Mining Multi-Faceted Overviews of Arbitrary Topics in a Text Collection

Xu Ling, Qiaozhu Mei, ChengXiang Zhai, Bruce Schatz
Department of Computer Science, Institute for Genomic Biology
University of Illinois at Urbana-Champaign
Urbana IL, 61801 USA
{xuling,qmei2,czhai,schatz}@uiuc.edu

ABSTRACT

A common task in many text mining applications is to generate a multi-faceted overview of a topic in a text collection. Such an overview not only directly serves as an informative summary of the topic, but also provides a detailed view of navigation to different facets of the topic. Existing work has cast this problem as a categorization problem and requires training examples for each facet. This has three limitations: (1) All facets are predefined, which may not fit the need of a particular user. (2) Training examples for each facet are often unavailable. (3) Such an approach only works for a predefined type of topics. In this paper, we break these limitations and study a more realistic new setup of the problem, in which we would allow a user to flexibly describe each facet with keywords for an arbitrary topic and attempt to mine a multi-faceted overview in an unsupervised way. We attempt a probabilistic approach to solve this problem. Empirical experiments on different genres of text data show that our approach can effectively generate a multi-faceted overview for arbitrary topics; the generated overviews are comparable with those generated by supervised methods with training examples. They are also more informative than unstructured flat summaries. The method is quite general, thus can be applied to multiple text mining tasks in different application domains.

Categories and Subject Descriptors

H.3.3 [**Information Search and Retrieval**]: Text Mining

General Terms

Algorithms

1. INTRODUCTION

Mining and extracting information from a text collection with ad hoc information needs is a common task in many applications. Currently, this is mainly achieved through a search engine. However, a search engine usually returns too many result pages, so how to help users digest the results remains a challenging task. Since users may take different perspectives to explore the information, ideally, it is appealing to automatically generate an overview of search results organized with multiple facets defined by a user. Such an overview enables a user to zoom into any specific facet of the topic. Moreover, interactive generation of such an overview would allow a user to probe a text collection from any ad hoc perspective.

From another perspective, such an overview serves as a semi-structured summary of text documents. This summary is much more informative than a standard unstructured summary which usually consists of just a set of sentences. For example, FlyBase[1] [21] provides a summary report for each Drosophila gene, including DNA sequence, functional description, mutant information, etc. However, such a gene summary is generated manually, which is extremely labor-intensive and impossible to keep up with the rapid growth of the literature data. As a result, automated summarizing gene information into multiple facets from biomedical literature has become an urgent task [11]. Another example of this type of information need is to find opinions about products from the Web. When people are exploring information about a product, *e.g.*, cameras, they tend to be interested in some natural semantic facets such as product features (battery, lens, and resolution), reviews, price, etc. An informative overview about product opinions should ideally organize and present information with these different facets. One can also easily imagine many other similar tasks in which a user would like an overview with a faceted structure and the summary sentences should be grouped according to this structure.

Therefore, an interesting research question is how to automatically generate such an overview for an arbitrary topic with ad hoc facets specified by a user. To the best of our knowledge, this problem has not been formally addressed, even though some related tasks have been studied such as clustering search results and supervised semi-structured summarization of biomedical literature, which we will further discuss in Section 6. Our task differs from clustering of search results in that we allow a user to specify the desired way of partitioning information, and it differs from supervised summarization in that we do not require any training example. From the perspective of mining multi-faceted overviews from a text collection, existing work has some lim-

KDD'08, August 24-27, 2008, Las Vegas, Nevada, USA.
Copyright 2008 ACM 978-1-60558-193-4/08/08 ...$5.00.

[1] http://flybase.bio.indiana.edu/.

itations. In complete unsupervised data-driven clustering, the resulted multi-faceted overviews do not necessarily reflect users's preferences. On the other hand, there are three problems with the supervised approaches: (1) All facets are predefined, which may not fit the need of a particular user. (2) Training examples for each facet are often unavailable. (3) Such an approach only works for a predefined domain.

In this paper, we break these limitations by studying a new and more realistic setup of the problem, in which we would allow a user to flexibly describe each facet with keywords for an arbitrary topic and would generate a multi-faceted overview without training examples (i.e., in a completely unsupervised way). For example, given a query like "Honda Accord," and a few keywords from the user which describes her interested facets like "engine," "design," and "price," our task is to generate a structured overview of all information about "Honda Accord." The resulted overview should be organized according to the user defined facets. From this overview, the user would be able to quickly find summarized information specific to each facet, and efficiently navigate to the original documents covering each aspect.

We propose a two-stage framework to solve this problem with probabilistic models. Specifically, in the first stage, we apply a bootstrapping method to expand the original facet keywords with additional correlated words in the document collection. Based on this expanded facet representation, we apply probabilistic mixture models to estimate the word distribution of every facet. This is done by simultaneously fitting the model to data and constraining a facet model so that it is close to the user specified definition. The basic idea is to "guide" the generative topic model with user defined facets.

We evaluate our method with two different tasks, summarizing literature information about a gene and mining consumers' car reviews. Both experiments show that the facet expansion in the first stage is effective and the regularized probabilistic model applied in the second stage performs better than other methods in most of the cases.

Our approach is quite general and has many potential applications. For example, it can serve as a generic overview mining system, in which the user only needs to provide a few keywords of her interested facets. Our method requires no training examples, thus can extract multiple facets for arbitrary topics and be applicable to any domains. Our approach can also serve as an starting point for any domain specific summarization system by alleviating the burden of acquiring large amounts of training examples. Although the proposed model works without needing training examples, it can also naturally incorporate training examples or external resources if any. Thus our system can be easily extended to support user feedback in interactive text mining.

2. PROBLEM FORMULATION

Following the definitions in [17], we formally define the key concepts of the problem of Multi-Faceted Overview Mining as follows:

Definition 1 (Document): We define a ***document*** d in a text collection $\mathcal{C}$ as a sequence of words $d = \{w_1, w_2, \ldots, w_{|d|}\}$. Following most existing work on topic modeling, we do not model the sequential order of words, thus the document is simply treated as a bag of words. We use $c(w,d)$ to denote the occurrences of word w in d.

Definition 2 (Facet Model): A ***facet model*** θ in a text collection $\mathcal{C}$ is a multinomial distribution of words $\{p(w|\theta)\}_{w \in V}$, which represents a semantic facet. The assumption of this model is that the words in text are sampled following word distributions corresponding to each facet, *i.e.*, $p(w|\theta)$. Therefore, the words with the highest probability in such a distribution would suggest the semantic topic represented by that facet. For example, a facet about car performance may assign high probability to words like "engine," "horsepower," and "speed."

Definition 3 (Multi-faceted Overview): A ***multi-faceted overview*** of a topic is a semi-structured summary of all information about the queried topic. Such an overview is structured in the way that sentences are grouped into the most relevant facets. We assume that a user would specify a facet she likes by a few keywords. Within each facet, sentences are ranked based on the relevance of their content to the corresponding facet. For example, a user searching for information about "Honda Accord" may want an overview with multiple facets such as "engine," "design," and "safety."

Based on the above concepts, we define the task of **Multi-Faceted Overview Mining (MuFOM)** from a text collection as follows: given an ad hoc topic and a few user specified keywords about the facets they are interested in, the goal is to generate a semi-structured overview of the query topic and present it with the user-specified facets.

The problem as defined above is challenging is many ways. First, we cannot rely on training examples to mine multi-faceted overview of arbitrary topics. The reasons are twofold: 1) it is impossible to create training examples for all ad hoc topics and facets; 2) it is usually too much burden for a user to label training examples for their interested facets. Therefore, a reasonable solution should not rely on any well-studied supervised-learning method. Whether a multi-faceted overview of arbitrary topics can be effectively generated without any training examples has not been proved in existing literature. Second, we alternatively assume that a user could specify facets by providing a few keywords. How to model and discriminate different facets using this limited information is not straightforward. In the next section, we will present a unified two-stage framework and probabilistic approaches to meet these challenges.

3. MINING MULTIPLE FACETS OF ARBITRARY TOPICS

We propose a novel system to generate multi-faceted overviews of arbitrary topics from text. It consists of two major components: a facet initialization module that initializes the representation of the user specified facets, and a facet modeling module that extracts and models the facets by statistical topic modeling.

3.1 Facet Initialization

In general, given an arbitrary topic, we expect that a user would guide the system by providing some information about the facets she wishes to use to organize the overview. To minimize the burden of the user, instead of asking the user to label training examples, we allow a user to specify the facets with keywords (*e.g.*, "price" and "performance" can be used to describe two interesting facets for the topic of buying vehicles). We initialize the facets with such keyword guidance from the user.

A facet can often be described in many different ways. For instance, in biomedical literature, the facet about genetical interaction can be described using different verbs such as "regulate," "inhibit," "promote" and "enhance." When writing a review about the interior design features of a car, the reviewer might describe it with different terms such as "air condition," "seat" *etc.*

In contrast to this richness of information, the guidance provided by a user is sparse: usually a couple of words. If we simply represent the facet of genetical interaction using the words "genetic" and "interaction", and represent the car's internal design features using the words "interior", "design", and "feature", we will end up with losing the information which is relevant to this facet but not described in such keywords. We solve this problem by borrowing the idea of query expansion commonly used in information retrieval. Specifically, we propose to iteratively expand/enrich the initial representation of a facet (*i.e.*, a few keywords) with additional related terms mined from the text collection, and generate an enriched initialization of facets. Such an representation would be used as a better "seed" to guide facet modeling.

Formally, we construct an undirected graph of terms, $G = (V, E)$, where each node in V is a term, and an edge $e = \langle v_i, v_j \rangle \in E$ indicates a similarity relationship between two terms. We weight e using $MI(i,j)$ $(i \neq j)$, where $MI(i,j) = \log \frac{p(v_i,v_j)}{p(v_i)p(v_j)}$. $MI(i,j)$ is known as the pointwise mutual information [3] of two terms, which measures the association of two terms based on their co-occurrences.

The current representation of a facet i is a set of keywords, or more generally a facet initial language model $\bar{\theta}_i$. Let us use A_i to denote the words with non-zero probability in $\bar{\theta}_i$. We then iteratively add new terms into A_i, and adjust $\bar{\theta}_i$ by

$$p(w|\bar{\theta}_i^{(t+1)}) = (1-\lambda)p(w|\bar{\theta}_i^{(0)}) + \lambda \sum_{w' \in N(w)} \frac{MI(w,w')}{Deg(w')} p(w|\bar{\theta}_i^{(t)}), \tag{1}$$

where $Deg(w) = \sum_{w' \in V} MI(w,w')$ and $N(w)$ is node w's maximum weighted neighbors.

The facet language models $\bar{\theta}_i^{(t)}$ after t iterations are passed to the next step as the initialization of the facets. There are two parameters associated with this step: n nearest neighbors in the graph G computed with Eqn. 1, and t iterations of adjusting $\bar{\theta}_i$.

3.2 Facet modeling

3.2.1 Statistical Topic Models

Statistical topic modeling [1, 8, 27, 18] is quite effective for mining topics in a text collection. In this kind of approaches, a document is often assumed to be generated from a mixture of k topic models. Probabilistic Latent Semantic Analysis (PLSA) [8] is a commonly used topic model. In this model, the log likelihood of a collection $\mathcal{C}$ is defined as:

$$L(\mathcal{C}) \propto \sum_{d \in \mathcal{C}} \sum_{w \in V} c(w,d) \log \sum_{j=1}^{k} p(\theta_j|d) p(w|\theta_j) \tag{2}$$

According to this mixture model, an article is "written" by following a stochastic process: first, the author would decide what topic to write about according to $p(\theta_j|d)$, which is the topic distribution of the article; then a word is sampled from the selected topic according to the word distribution of this topic $p(w|\theta_j)$. In this model, the parameters to be estimated are the word distributions of each topic and the topic distributions of each document $\Theta = \{\{p(\theta_j|d)\}_{j=1..k}, \{p(w|\theta_j)\}_{j=1..k}\}$. Generally, an Expectation-Maximization (EM) algorithm is applied to estimate the topic models. We introduce hidden variables $z(w,d,j)$, to represent the probability that a term w in document d is generated from topic j. To maximize the observed data likelihood, EM iteratively performs the following expectation (E) step and maximization (M) step. E-step:

$$\begin{aligned} z(w,d,j) &= p(w,d) \in \theta_j|\Theta_n \\ &= \frac{p_n(\theta_j|d)p_n(w|\theta_j)}{\sum_{j'=1}^{k} p_n(\theta_{j'}|d)p_n(w|\theta_{j'})} \end{aligned} \tag{3}$$

M-step:

$$p_{n+1}(w|\theta_j) = \frac{\sum_d c(w,d)z(w,d,j)}{\sum_d \sum_{w'} c(w',d)z(w',d,j)} \tag{4}$$

$$p_{n+1}(\theta_j|d) = \frac{\sum_w c(w,d)z(w,d,j)}{\sum_w \sum_{j'} c(w,d)z(w,d,j')} \tag{5}$$

Applying this model to our problem, facets are now the topics in the above mixture model. Without any constraints, estimating the facet models by maximizing the log likelihood of data alone may overfit the data and result in facet clusters which do not necessarily represent user's interested facets. That's essentially the problem with unsupervised clustering. However, in our task, we have user specified facet keywords, which serve as potentially good guidance. We want the resulted facet models to represent the user defined facets by constraining the estimated models to be close to the initial facet definitions. Therefore, rather than directly estimating the facet models by maximizing log likelihood of the data, we propose two alternative strategies to guide the facet model estimation: a generative model by applying prior distribution and a discriminative way of applying regularization. In the following, we discuss both approaches in detail and derive corresponding EM updating formulas.

3.2.2 Guidance with Dirichlet Model Priors

In this paper, we incorporate the prior knowledge of the facets by applying a prior distribution of the facet model. Ideally, such prior knowledge would be obtained from training data. However, producing training data for every facet of arbitrary topics is not realistic. Here we use previously initialized facet models to define a prior on the facets and estimate the models using the maximum a posterior (MAP) estimator.

Specifically, let $\bar{\theta}_j$ be the previously initialized facet models, we define the conjugate Dirichlet prior for the model θ_j as $Dir(\{1 + \mu_j p(w|\bar{\theta}_j)\}_{w \in V})$, where the parameter μ_j indicates how much confidence is put on this prior. The model prior is now defined as $p(\Theta) \propto \prod_{j=1}^{k} \prod_{w \in V} p(w|\theta_j)^{\mu_j p(w|\bar{\theta}_j)}$, and the impact of adding the prior is equivalent to adding $\mu_j p(w|\bar{\theta}_j)$ pseudo counts for word w in estimating $p(w|\theta_j)$. This way would allow us to adopt previously initialized facet models into our parameter estimation. For example, when the user defines one facet using "genetic interaction," we would "guide" the model estimation by constraining the resulted model to be as close as possible to the prior model which has high probability of generating these two words. Here, we can use a uniform μ for all μ_j if our confidences on different facets are roughly the same.

With this defined prior, we may now use the MAP estimator: $\hat{\Theta} = \mathrm{argmax}_{\Theta} p(\mathcal{C}|\Theta)p(\Theta)$ by applying the EM algorithm. Following [15, 17, 14], corresponding change in the M steps (Eqn. 4) is to incorporate the pseudo counts given by the prior:

$$p_{n+1}(w|\theta_j) = \frac{\mu_j p(w|\bar{\theta}_j) + \sum_d c(w,d)z(w,d,j)}{\mu_j + \sum_d \sum_{w'} c(w',d)z(w',d,j)}. \qquad (6)$$

The estimated $p(w|\theta_j)$ can then be assumed to be our facet models. Note that a user does not have to give keywords for every facet; indeed, if a user has not given keywords for the j-th facet, we could set $\mu_j = 0$, and the j-th facet would be discovered as an additional facet to those specified by the user.

3.2.3 Guidance with Regularization

An alternative solution of utilizing the user defined facets to guide the model estimation is to apply a discriminative approach. Here, we propose a regularized two-stage estimation framework. First, from previously initialized facets, we retrieve top ranked documents for each facet. We estimate the facet distribution of these documents by fixing the word distribution of each facet to the initialized models and estimating the facet distribution of the top retrieved documents using the PLSA EM updating formulas in Section 3.2.1. Again, as in the case of applying priors, a user does not have to give keywords for every facet. For the additional facets to those specified by the user, the corresponding facet distributions are estimated by initializing their word distribution from a random distribution and iteratively applying PLSA EM updating formulas. The goal here is to obtain a "training" document set $d \in \mathcal{C}_T$, whose facet distribution $\bar{p}(\theta_j|d)$ will be used to "constrain" the final estimation of facet models in a regularized framework. The second phase adopts the idea of applying network regularization to topic modeling in [16], which is to estimate facet models by maximizing the regularized data likelihood:

$$O(\mathcal{C}, \mathcal{C}_T) = (1-\alpha-\beta)L(\mathcal{C}) - \alpha R(\mathcal{C}) - \beta R_T(\mathcal{C}_T) \qquad (7)$$

In the objective function, $L(\mathcal{C})$ is the log likelihood of the collection $\mathcal{C}$ defined by Eqn.2, $R(\mathcal{C})$ is a regularizer defined on the collection $\mathcal{C}$ according to document similarity, and $R_T(\mathcal{C}_T)$ is the regularizer defined on the "training" document sets $\mathcal{C}_T$ based on the facet distribution. We define the regularizer similar to the graph harmonic function in [28], in which the graph nodes are documents and the edge weights represent document similarities. The underlying assumption is that two document similar in content should have similar facet distributions, and the final estimated facet distribution of the "training" documents should be close to the "constraint" distribution $\bar{p}(\theta_j|d)$.

$$R(\mathcal{C}) = \sum_{\langle u,v\rangle \in E} w(u,v) \sum_{j=1}^{k} (p(\theta_j|u) - p(\theta_j|v))^2 \qquad (8)$$

$$R_T(\mathcal{C}_T) = \sum_{u \in \mathcal{C}_T} (\sum_{v \in \mathcal{C}} w(u,v)) \sum_{j=1}^{k} (p(\theta_j|u) - \bar{p}(\theta_j|u))^2 \qquad (9)$$

The basic idea of this framework is to "supervise" statistical topic modeling using the discriminative regularization. Intuitively, $L(\mathcal{C})$ in Eqn.7 measures how likely the data is generated from the facet models, and $R_T(\mathcal{C}_T)$ constrains the estimated facet models to be close to the initial facet models. Utilizing $R(\mathcal{C})$ essentially propagates this constraint through the entire collection according to document similarities. The parameter α and β can then be set between 0 and 1, which indicates how much we want to follow the "training" facet distributions. When $\alpha + \beta = 0$, the objective function boils down to the log likelihood of PLSA. Thus maximizing $O(\mathcal{C}, \mathcal{C}_T) = L(\mathcal{C})$ will lead to the facets which best fit the contents of the collection. When $\alpha + \beta = 1$, it boils down to the general graph-based semi-supervised learning. Formally, the objective function is:

$$\begin{aligned} & O(\mathcal{C}, \mathcal{C}_T) \\ = & -(1-\alpha-\beta) \sum_{d \in \mathcal{C}} \sum_{w \in V} c(w,d) \log \sum_{j=1}^{k} p(\theta_j|d)p(w|\theta_j) \\ + & \alpha \sum_{\langle d,d'\rangle \in E} w(d,d') \sum_{j=1}^{k} (p(\theta_j|d) - p(\theta_j|d'))^2 \qquad (10) \\ + & \beta \sum_{d \in \mathcal{C}_T} (\sum_{d' \in \mathcal{C}} w(d,d')) \sum_{j=1}^{k} (p(\theta_j|d) - \bar{p}(\theta_j|d))^2 \end{aligned}$$

Note, this model can handle arbitrary number of facets, and applicable to the more general situation in which the number of facets can be larger than the number of facets specified by user's keywords, so that the users can discover new facets other than what they have in mind.

Now, the remaining task is to estimate model parameters by maximizing the objective function in Eqn.10 with the constraints that $\sum_j p(\theta_j|d) = 1$ and $\sum_w p(w|\theta_j) = 1$. Please note that Eqn.10 is closely related to the regularized PLSA model used in [16]. The difference is that with the additional regularizer $R_T(\mathcal{C}_T)$, this model now becomes a semi-supervised method. Following [16], we can use a generalized EM algorithm (GEM) [19] with the similar EM updating procedure. We outline every iteration of the GEM algorithm as follows:

1. In E step (Iteration n), computing the hidden variables using Equation 3. This is exactly the same with PLSA.
2. In M step (Iteration n):
   - 2.1 Compute the updated $p_{n+1}(w|\theta_j)$ using Equation 4.
   - 2.2 Compute the $p_{n+1}^{(0)}(\theta_j|d)$ using Equation 5.
   - 2.3 Iteratively compute: $p_{n+1}^{(t+1)}(\theta_j|d)$

     $$\begin{aligned} = & \ (1-\gamma)p_{n+1}^{(t)}(\theta_j|d) + \gamma \cdot \frac{\alpha}{\alpha+\beta}\bar{p}(\theta_j|d) \\ + & \ \gamma \cdot \frac{\beta}{\alpha+\beta} \frac{\sum_{d'} w(d,d')p_{n+1}^{(t)}(\theta_j|d)}{\sum_{d'} w(d,d')} \qquad (11) \end{aligned}$$

     until $O_{n+1}(\mathcal{C}, \mathcal{C}_T)$ cannot be improved.
3. Stop EM iteration when $O_n(\mathcal{C}, \mathcal{C}_T)$ converges.

Applying this algorithm, the estimated $p(w|\theta_j)$ can then be assumed to be our facet models.

4. GENERATING OVERVIEW

Our strategy for generating multi-faceted overviews essentially resembles the strategy in [12], i.e., using the estimated facet models to categorize the sentences about the query topic into appropriate semantic facets. The process is as follows. First, we retrieve the documents relevant to the query topic, and segment each document into sentences. Secondly, we compute the relevance score S between each sentence-facet pair. To ensure reliable association between sentences and facets, for each sentence, we rank all facets based on S and keep only the facets with highest scores. Essentially, the sentences assigned to a certain facet compose a summary of the facet. The underlying rationale is to empirically only consider the most dominant facets in one sentence. Then, for each facet, we present it in a ranked list of sentences according to S. Such multi-faceted overview is similar to the "attribute data" report in FlyBase, and the online reviews of a product, *e.g.*, the editor review of cars at edmunds.com[2].

Given facet models, we can use the negative KL-divergence [10] function to measure the similarity between the sentence s and the estimated facet θ_j: $S = -D(\theta_j||\theta_s) = \sum_w p(w|\theta_j) \log \frac{p(w|\theta_s)}{p(w|\theta_j)}$, where θ_s, θ_j represents the language model of the sentence and the facet respectively. The sentence model is computed using relative frequency of words in sentence after Dirichlet smoothing: $p(w|\theta_s) = \frac{c(w,s)+\mu p(w|\mathcal{C})}{|s|+\mu}$, where $c(w,s)$ is the count of word w in sentence s and $p(w|\mathcal{C}) = c(w,\mathcal{C})/|V|$ is the collection background model.

5. EXPERIMENTS AND RESULTS

The framework we proposed is quite general, and can be applied to many domains. In this section, we evaluate our system in two completely different domains and show that the strategy of facet expansion is quite effective, and the regularized topic model approach performs the best among almost all compared methods.

The experiments are conducted on a relatively "clean" set of documents retrieved for the given topic. We made this assumption because the focus here is not to study the retrieval performance. For example, the Gene Summarizer will first tag all abstracts with gene names utilizing the gene name entity recognizer [9] developed for related project. Since our method can discover additional facets that a user has not specified, it can still work if the retrieval results are not very accurate as the non-relevant documents (e.g., those matching a distracting sense of an ambiguous query word) likely will be separated as forming additional facets. In such a case, our overview enables a user to quickly zoom into relevant facets. In the following experiments, in order to evaluate the results, we set the number of facets to be identical to that specified by a user, even though in a real application we may use a larger number of facets to accommodate any additional facets not specified by the user.

5.1 Gene summarization

Automated generating semi-structured gene summaries from biomedical literature is a very important task in modern biomedical research. A supervised solution has been proposed to extract relevant information about a gene from literature and present it in six predefined generic facets [11]. We experimented our methods in this problem, and adopted the same strategy to generate the gene summaries proposed by [11]. Basically, for each retrieved sentence of the query gene, we rank all the facets based on the sentence-facet relevance score, and assign it to two most relevant facets. Then each facet is summarized by a ranked list of assigned sentences based on that score.

Table 2: Precision of the top-5 extracted sentence comparing different level of facet expansion

Asp.	**UpBd**	**Orig.**	**n10u1**	**n10u10**	**n50u1**	**n50u10**
SI	0.75	0.49	0.45	0.43	0.44	0.45
GI	0.81	0.41	0.37	0.36	0.47	0.47
GP	0.49	0.22	0.20	0.18	0.18	0.22
EL	0.45	0.18	0.24	0.25	0.25	0.25
MP	0.51	0.20	0.26	0.23	0.23	0.25
WFPI	0.64	0.15	0.25	0.20	0.17	0.19
Avg.	0.61	0.28	0.30	0.28	0.29	0.31

In this experiment, we retrieved 22590 PubMed abstracts about fruit fly as our document collection by matching the keyword "Drosophila melanogaster" in the MESH[3] field. We used Lemur Toolkit[4] to implement the system. Adopting the six facets defined in [11] (see Table 1), our system started from the facet keywords and estimated facet models based on the entire collection. Different methods are evaluated based on their final generated gene summaries.

The experiment was done on 19 randomly selected fruit fly genes. We retrieved 463 sentences relevant to these testing genes from our fruit fly document collection, and asked an insect biologist to annotate these sentences with the predefined six facets in Table 1 to construct a gold standard. A sentence is assigned a facet label if and only if it contains information on this facet, regardless of whether it contains any extra information. To study how different methods affect the final generated summary, we evaluated them based on the precision of best five sentences for each facet separately. The results are shown in Table 2 and 4.

We first evaluated the facet expansion module in Table 2. In this experiment, we fixed the facet model estimation step to the regularized approach in Section 3.2.3 with parameters $\alpha = 0.1, \beta = 0.5$ and top-5 "training" document per facet, and measured precision@5 using the human annotated gold standard for results with 5 different levels of facet expansion. The facet models is constructed by (1) **Orig**: relative frequency of the original facet keywords; (2) **n10u1**: one iteration adjusting the facet models through 10 neighbors in the MI graph; (3) **n10u10**: 10 iterations through 10 neighbors; (4) **n50u1**: one iteration through 50 neighbors; (5) **n50u10**: 10 iterations through 50 neighbors. Among these runs, run 1 did not expand the original representation of the facets, and run 5 went through the most extensive expansion. Note, column **UpBd** indicates the upper bound precision@5 scores as some testing genes with relatively few references do not have 5 sentences per facet in our gold standard annotations. As can be seen from Table.2, along with more expansion on the facet representation, the generated summary achieves better score. The summarization performance varies across facets. In general, sequence information (SI) and genetic interaction (GI) get best scores, and Wild-type Function & Phenotypic Information (WFPI) have the lowest precision. This may be because these two best facets are the most specific thus easier to be discriminated from others. While WFPI is too broad and may be described

[2] http://www.edmunds.com/toyota/corolla/2009/review.html

[3] MeSH (http://www.nlm.nih.gov/mesh/meshhome.html) is the U.S. National Library of Medicine's controlled vocabulary used for indexing articles for MEDLINE/PubMed.

[4] The Lemur Toolkit (http://www.lemurproject.org/) is a open-source toolkit designed to facilitate research in language modeling and information retrieval.

Table 1: facets definition and keywords for gene summary

Name	**Keywords**	**Definition**
Sequence Information (SI)	sequence, similarity	Describing the sequence information of the target gene and its product.
Genetical Interaction (GI)	interaction	Describing the genetical interactions of the target gene with other molecules.
Gene Product (GP)	protein, product	Describing the product (protein, rRNA, *etc.*) of the target gene.
Expression Location (EL)	expression	Describing where the target gene is mainly expressed.
Mutant Phenotype (MP)	mutation,phenotype	Describing the information about the mutant phenotypes of the target gene.
Wild-type Function & Phenotypic Information (WFPI)	wild-type, function	Describing the wild-type functions and the phenotypic information about the target gene and its product.

Table 3: Example facet models for gene summary after facet expansion

SI	GI	GP	EL	MP	WFPI
sequence	interaction	protein	expression	mutant	type
similar	between	product	gene	mutation	wild
amino	interact	gene	express	phenotype	function
protein	bind	encode	we	allele	mutant
acid	domain	we	regulate	defect	mutation
gene	protein	function	transcription	wild	we
region	we	are	protein	lethal	these
encode	complex	expression	develop	type	phenotype

Table 4: Precision of the top-5 extracted sentence comparing different facet modeling methods

Asp.	**Sup**	**Pri**	**Reg**	**MQR**	**MQR+FB**
SI	0.29	0.44	**0.45**	0.47	0.43
GI	0.49	**0.51**	0.47	0.41	0.39
GP	0.20	0.20	**0.22**	0.20	0.20
EL	0.15	0.22	**0.25**	0.18	0.20
MP	0.17	**0.25**	**0.25**	0.20	0.22
WFPI	0.27	0.09	**0.19**	0.15	**0.19**
Avg.	0.26	0.29	**0.31**	0.27	0.27

using various terms, thus it is harder to model its word distribution. This trend is also observed in Table 4 when comparing different facet modeling methods.

The effectiveness of facet expansion on this task also varies across facets. As we should expect, when the original keywords from the user work well (i.e., SI, GI and GP), expansion is not so effective as in the case where the original keywords do not work well (i.e., EL, MP and WFPI). In Table.3, we show the top 10 words of each facet model after ten iterations of expansion with 50 MI neighbors. (Due to space limit, the probabilities of these terms are not shown.) In facet GP, we see terms like "encode" now ranked very high, which is actually a very informative term indicating gene product information. In facet MP, terms like "allele," "defect," "lethal" indicating important information of mutant phenotype are now expanded into the original facet. By adjusting facet model with these informative terms, the model is more accurate and closer to actual word usage of the facets.

Secondly, we evaluated the effectiveness of different facet modeling approaches in Table.4. In this experiment, we compared 5 runs: **Pri** and **Reg** are our prior-based and regularizer-based approaches (see Section 3.2.2 and 3.2.3) with most extensive facet expansion; **Sup** represents the result of the system by [11] using the supervised approach; **MQR** casts this task as a multi-query retrieval problem, where it treats the original query and keywords of each facet as independent queries and generate final summary following our strategy in Section 4; **MQR+FB** is a variation of **MQR** with pseudo feedback. We made three observations from Table. 4.

First, **Sup** performs worse than other methods in facet SI, EL and MP. In the beginning, this seems unexpected, since **Sup** applies the supervised method which exploits training examples and presumably should work better. However, after looking at the procedure how the training examples are generated in [11], it is not surprising at all. Training examples for facets SI, EL and MP are extracted from the "Summary" paragraph in FlyBase, which are actually generated from a common template by filling in the key information term in their underline database. For example, sentences about sequence information is always described using the template: "It has been sequence and its amino acid sequence contains a ...". While in real abstracts, the same information would be described in variety of ways using different terms. Apparently, using these "faked" training examples would hurt the summarization. The training examples for other three facets were real sentences heuristically extracted from abstracts, thus provides good supervision to the summarization. We can see that our proposed methods (**Pri** and **Reg**) perform comparable with **Sup** in facets GI and GP, and only worse than **Sup** in WFPI which is a difficult facet to model.

Secondly, both of our proposed methods (**Pri** and **Reg**) perform better than the multi-query retrieval methods with or without pseudo feedback (**MQR** and **MQR+FB**). The regularized approach performs the best for 5 out of 6 facets and achieves best average score over all six facets. Compared with it, the prior-based approach is slightly worse. One interesting thing is that the prior-based approach has lowest score for the facet WFPI. The reason might be again related to the difficulty of this facet. As the prior-based approach constrains the final model to be close to the prior expanded from original keywords, when the facet expansion could not improve much (comparing column **u0** and **n50u10** in Table 2) for this difficult facet, trusting too much in this prior will certainly hurt. However, the regularizer approach can overcome this problem since it constrains the final model on the facet distribution instead of the facet model itself.

Third, for the multi-query retrieval methods, using pseudo feedback improved the four relatively difficult facets like GP, EL, MP, and WFPI, while performed worse than that without pseudo feedback in the two easier facets SI and GI. This

Table 5: facets for consumer reviews of cars

Name	**Keywords**
Body Styles (BS)	exterior, design, body, style
Powertrains & Performance (PP)	performance, fuel, powertrain
Safety (SF)	safety, reliability
Interior Design & Features (IF)	interior, design, features
Driving Impressions (DI)	comfort, fun

Table 6: Example facet models for car reviews after facet expansion

BS	PP	SF	IF	DI
body	powertrain	safety	interior	fun
style	performance	reliable	design	comfort
exterior	fuel	feature	feature	drive
design	economy	mercedes	safety	seat
interior	9	preferred	exterior	very
panel	please	term	standard	are
attract	torque	cheap	space	long
different	tribute	sporty	panel	ride
roof	72	basic	roof	well

is not surprising, as pseudo feedback plays a similar role in retrieval as in facet expansion and probabilistic mixture models in our system. For easy facets, the original keywords already captured majority of the information, while pseudo feedback may introducing other noisy terms thus shift the facet model away from what it should be. After averaging over all facets, using pseudo feedback or not achieved similar scores.

5.2 Overview of consumer reviews of cars

In this section, we test our system in another domain, *i.e.*, online car reviews. We crawled the consumers' reviews from edmunds.com on 15 car models like "chevrolet, malibu, 2006," "honda, accord, 2006," as our document collection (1156 reviews in total). Among these queries, 12 have comprehensive editor reviews, which are used as our gold standard overviews for evaluation. We applied different methods on generating overviews of the consumer reviews, and evaluated the final performance using ROUGE[5]. The generated overviews consist of ten best sentences per facet. We evaluated different methods with all the metrics provided by ROUGE, and report the ROUGE-1 Average_R score averaging over all 12 test queries in Table 7 (performance on other metrics are all consistent and not presented here). In this evaluation, the five facets used in editor reviews are picked as our test facet set (see Table 5).

In this experiment, the effectiveness of facet expansion on this task is demonstrated by the word distribution of facet models after initialization using the MI graph. The top words of each facet model expanded by one iteration of adjustment through ten MI neighbors are displayed in Table 6. In the facet Powertrains & Performance (PP), we see terms like "economy" is ranked very high, which is actually a very informative term indicating fuel economy of the car.

Table 7: ROUGE-1 Average_R scores

Asp.	**Pri**	**Reg**	**MQR**	**MQR+FB**
BS	0.193	**0.200**	0.174	0.197
PP	0.273	**0.278**	0.207	0.239
SF	0.235	**0.243**	0.208	0.230
IF	0.309	**0.324**	0.294	0.287
DI	0.316	**0.319**	0.264	0.271
Avg.	0.265	0.**273**	0.229	0.245

In the facet Driving Impressions (DI), terms like "drive," "seat" indicating driving experience are now expanded into the original model.

In Table.7, **Pri** and **Reg** represent the prior-based and regularizer-based model estimation methods based on the expansion above. **MQR** and **MQR+FB** represent the multi-query retrieval method without or with pseudo feedback. The regularizer-based method performed best for all five facets. Consistent with the above experiments on the gene summarization task, both of our proposed methods (**Pri** and **Reg**) performed better than the multi-query retrieval methods. We also presented one example of our generated overview (with top-2 sentences per facet) for the query "honda, accord, 2006" and its corresponding editor's review in Table 8. Especially, our extracted sentence for the facet interior design matches excellently well to the editor's review.

The above analysis is based on the facets defined by Edmunds's editor reviews. As our system is able to generate overviews for any user defined facets, one important feature that our system provides is allowing a user to customize the overview using her own interested facets. To illustrate this, we experimented our system on another set of facets with the same parameter setting as above, and generated the overview in Table.9 for the same query "honda, accord, 2006". This facet definition represents a different user information need. In this case, the user do not want to discriminate between interior and exterior design features, but is interested in another facet about price. Now, the returned sentences about exterior and interior all come to the facet "Design." In the "Finance" facet, the sentences about price are extracted. This example shows that our system is effective for generating multi-faceted overviews according to users' ad hoc information interests without requiring any training examples.

6. RELATED WORK

As a primary strategy for presenting search results, generating an overview by organizing search results has attracted a lot of attention in both web industry and academia research communities. Current research is mostly conducted in either unsupervised (e.g., data-driven clustering), or supervised manner by exploiting extra resources. In one line of work [7, 20, 25, 26], clustering algorithms are used to cluster the top documents returned from a traditional information retrieval system based on the assumption that relevant documents tend to form clusters [23]. However these works generate overviews solely based on unsupervised clustering of the search results, thus the obtained clusters do not necessarily correspond to users' real interests. Supervised methods by exploiting external resources such as Web directory, search logs, and WordNet are also studied [2, 4,

[5] ROUGE (http://berouge.com) is a commonly used evaluation package to automatically evaluate summarization systems. It provides a suite of evaluation metrics to measure the similarity between system generated summaries and the gold standard.

Table 8: Example of generated overview for query “honda, accord, 2006” and corresponding editor’s review

facets	Generated Overview	Editor’s Review
Body Styles, Exterior Design	Like the minor exterior styling changes from 2005 to 2006. Tried the Camry XLE first, nice ride, but lacked a few features i wanted, like dual zone A/C, and didn’t like the wood trim.	... Available trim levels include ... The VP provides air conditioning, power windows ...
Powertrains, Performance	I am very pleased with fuel economy. I was amazed at the performance of the 4cyl engine, great pick-up and great fuel economy.	... you’ll get a combined 253 hp and 232 lb-ft of torque and a 25 city/34 highway EPA rating (best in the lineup). Four-cylinder engines are available with ...
Safety	I enjoy honda cars because they are reliable, and have a good resale value, safety features, and they make a quality product. Take it from me...my original Honda Accord Coupe did not give me any problems at all.	... In IIHS testing, the Honda Accord earned a Good rating (the best possible) for frontal-offset crash safety; in side-impact tests, it received a Good rating when equipped with side airbags ...
Interior Design	The interior is beautiful - I got all of the features and the navigation is extremely easy to use. Accord’s interior is top notch, nice design, clear gauges, comfy seats, lots of storage space.	Honda tailored the Accord’s interior to meet the needs of the American family. The seating arrangements are top-notch, and the interior design and materials quality continue the high-caliber standards ... The car’s backseat is among the roomiest in the segment...
Driving Impressions	There are sportier-handling rides, but it is still very responsive, yet comfortable on long trips. Drives very nicely & is comfortable.	... The Accord’s steering has a slick, precise feel and the suspension provides a comfortable ride as well as decent levels of road grip ... Brake feel is reassuring...

Table 9: Example of overview by a different another set of facet definitions for query “honda, accord, 2006”

facets	Definition	Generated Overview
Design	design, style	Like the minor exterior styling changes from 2005 to 2006. Accord’s interior is top notch, nice design, clear gauges, comfy seats, lots of storage space.
Engine	engine, fuel	I was amazed at the performance of the 4cyl engine, great pick-up and great fuel economy. Fuel economy is better than the sticker,(35+ on a recent Boston trip) as was the CRV.
Finance	finance, price	When I bought it I was amazed at the trim level for the price. It is extremely fun to drive, fit and finish is fantastic, the oversteer could easily be corrected, at the price, it has no peer and is 10k less then a comparable BMW
Safety	safety	I enjoy honda cars because they are reliable, and have a good resale value, safety features, and they make a quality product. For the price of the car, there are many safety features with airbags front, side and rear side bags.
Driving	comfort, fun	There are sportier-handling rides, but it is still very responsive, yet comfortable on long trips. Drives very nicely & is comfortable.

24, 22]. However their generality is often limited due to the labor required to build the external resources (e.g., training data); more importantly, they cannot accommodate a customized structure of organizing search results for a user.

In bioinformatics, people have studied how to automatically generate gene summaries to facilitate biologists in finding gene-centered information from biomedical literatures [11]. Existing solutions are summarizing all gene information from MEDLINE abstracts into six predefined generic facets. This approach adopted a training data set by utilizing the annotated data from the model organism databases (e.g., FlyBase).

Another related problem is mining and summarizing opinions in Weblogs [13, 6, 5], where the goal is mainly to mine user opinions by identifying and extracting positive and negative opinions or analyzing and extracting topical contents of blog articles. None of this body of work would allow a user to impose an ad hoc structure of the summary. The simultaneous topic-sentiment analysis work in [17] is more related to our work as the idea of applying prior to mining topics from blog articles is similar to one of the approaches for modeling facets discussed in Section 3.2.2. However, the effectiveness of such a method is not quantitatively evaluated. And the sentiment model is still launched in a completely supervised manner which requires training examples. Our regularized topic model is based on the general regularization framework proposed in [16], and the MAP estimator of PLSA has also previously been used in [14] for opinion integration.

7. CONCLUSIONS

In this paper, we studied a novel problem in text mining: automated generation of multi-faceted overviews for arbitrary topics from a text collection. We developed a system which combines both generative and discriminative topic mining techniques to automatically summarize information about a query topic into multiple facets. Empirical experiments demonstrated that our proposed system, especially the regularized PLSA model, is quite effective in mining multi-faceted overviews for applications in two completely different domains: the gene summarization task in biomedical literature and the car review mining task for online customer reviews. Given our general setup of the problem (i.e., no need for training examples and a user can flexibly specify facets with keywords), our proposed methods can be applied to many domains.

The general problem of mining multi-faceted overviews for arbitrary topics represents a new research direction in text mining. Our work can be extended in several directions: (1) We have not considered the removal of redundant information in generated overviews. One future improvement would be to integrate other text features for redundancy removal. (2) The user specified facets might lie in different granularities, which is not tackled in this work, but will certainly be an interesting future topic to explore. (3) In our model of estimating facets, incorporating training examples and user feedbacks is a very natural extension. Extensive experiments and studies on utilizing interactive user feedback would bring important insight into how to leverage additional information in user activities. (4) There are many types of online resources that can be utilized to improve the facet modeling, research on how to integrate them into our framework would be another interesting future direction.

8. ACKNOWLEDGEMENT

We thank all the anonymous reviewers for their valuable comments. We also thank Moushumi Sen Sarma for her great effort in annotating the gene summary sentences. This work is in part supported by the National Science Foundation under award numbers 0425852, 0713571, 0428472, as well as a Yahoo! PhD Fellowship.

9. REFERENCES

[1] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *J. Mach. Learn. Res.*, 3:993–1022, 2003.

[2] H. Chen and S. Dumais. Bringing order to the web: automatically categorizing search results. In *Proceedings of CHI '00*, pages 145–152, 2000.

[3] K. W. Church and P. Hanks. Word association norms, mutual information, and lexicography. *Comput. Linguist.*, 16(1):22–29, 1990.

[4] S. T. Dumais, E. Cutrell, and H. Chen. Optimizing search by showing results in context. In *Proceedings of CHI '01*, pages 277–284, 2001.

[5] D. Gruhl, R. Guha, R. Kumar, J. Novak, and A. Tomkins. The predictive power of online chatter. In *Proceedings of KDD '05*, pages 78–87, 2005.

[6] D. Gruhl, R. Guha, D. Liben-Nowell, and A. Tomkins. Information diffusion through blogspace. In *Proceedings of WWW '04*, pages 491–501, 2004.

[7] M. A. Hearst and J. O. Pedersen. Reexamining the cluster hypothesis: Scatter/gather on retrieval results. In *Proceedings of SIGIR '96*, pages 76–84, Zürich, CH, 1996.

[8] T. Hofmann. Probabilistic latent semantic indexing. In *Proceedings of SIGIR '99*, pages 50–57, 1999.

[9] J. Jiang and C. Zhai. Exploiting domain structure for named entity recognition. In *Proceedings of HLT-NAACL '06*, pages 74–81, 2006.

[10] Kullback, S. and Leibler, R. A. On information and sufficiency. *The Annals of Mathematical Statistics*, 22(1):79–86, mar 1951.

[11] X. Ling, J. Jiang, X. He, Q. Mei, C. Zhai, and B. Schatz. Automatically generating gene summaries from biomedical literature. In *Proceedings of PSB '06*, pages 41–50, 2006.

[12] X. Ling, J. Jiang, X. He, Q. Mei, C. Zhai, and B. R. Schatz. Generating gene summaries from biomedical literature: A study of semi-structured summarization. *Inf. Process. Manage.*, 43(6):1777–1791, 2007.

[13] B. Liu, M. Hu, and J. Cheng. Opinion observer: analyzing and comparing opinions on the web. In *Proceedings of WWW '05*, pages 342–351, 2005.

[14] Y. Lu and C. Zhai. Opinion integration through semi-supervised topic modeling. In *Proceedings of WWW '07*, pages 121–130, 2008.

[15] G. J. McLachlan and T. Krishnan. *The EM Algorithm and Extensions*. Wiley, 1997.

[16] Q. Mei, D. Cai, D. Zhang, and C. Zhai. Topic modeling with network regularization. In *Proceedings of WWW '08*, pages 101–110, 2008.

[17] Q. Mei, X. Ling, M. Wondra, H. Su, and C. Zhai. Topic sentiment mixture: modeling facets and opinions in weblogs. In *Proceedings of WWW '07*, pages 171–180, 2007.

[18] Q. Mei and C. Zhai. A mixture model for contextual text mining. In *Proceedings of KDD '06*, pages 649–655, 2006.

[19] R. M. Neal and G. E. Hinton. A view of the em algorithm that justifies incremental, sparse, and other variants. pages 355–368, 1999.

[20] P. Pirolli, P. Schank, M. Hearst, and C. Diehl. Scatter/gather browsing communicates the topic structure of a very large text collection. In *Proceedings of CHI '96*, pages 213–220, 1996.

[21] M. A. C. R. A. Drysdale and T. F. Consortium. Flybase: genes and gene models. *Nucleic Acids Res.*, 33:390–395, 2005.

[22] E. Stoica, M. Hearst, and M. Richardson. Automating creation of hierarchical faceted metadata structures. In *Proceedings of NAACL/HLT '2007*, pages 244–251, 2007.

[23] C. J. Van Rijsbergen. *Information Retrieval, 2nd edition.* Dept. of Computer Science, University of Glasgow, 1979.

[24] X. Wang and C. Zhai. Learn from web search logs to organize search results. In *Proceedings of SIGIR '07*, pages 87–94, 2007.

[25] O. Zamir and O. Etzioni. Grouper: a dynamic clustering interface to Web search results. *Computer Networks (Amsterdam, Netherlands: 1999)*, 31(11–16):1361–1374, 1999.

[26] H.-J. Zeng, Q.-C. He, Z. Chen, W.-Y. Ma, and J. Ma. Learning to cluster web search results. In *Proceedings of SIGIR '04*, pages 210–217, 2004.

[27] C. Zhai, A. Velivelli, and B. Yu. A cross-collection mixture model for comparative text mining. In *Proceedings of KDD '04*, pages 743–748, 2004.

[28] X. Zhu, Z. Ghahramani, and J. D. Lafferty. Semi-supervised learning using gaussian fields and harmonic functions. In *ICML*, pages 912–919, 2003.