A Cross-Collection Mixture Model for Comparative Text Mining

ChengXiang Zhai
Department of Computer Science
University of Illinois at Urbana Champaign

Atulya Velivelli
Department of Electrical Computer Engineering
University of Illinois at Urbana Champaign

Bei Yu
Graduate School of Library and Information Science
University of Illinois at Urbana Champaign

ABSTRACT

In this paper, we define and study a novel text mining problem, which we refer to as comparative text mining. Given a set of comparable text collections, the task of comparative text mining is to discover any latent common themes across all collections as well as summarize the similarity and differences of these collections along each common theme. This general problem subsumes many interesting applications, including business intelligence, summarizing reviews of similar products, and comparing different opinions about a common topic. We propose a generative probabilistic mixture model for comparative text mining. The model simultaneously performs cross-collection clustering and within-collection clustering, and can be applied to an arbitrary set of comparable text collections. The model can be estimated efficiently using the Expectation-Maximization (EM) algorithm. We evaluate the model on two different text data sets (i.e., a news article data set and a laptop review data set), and compare it with a baseline clustering method also based on a mixture model. Experiment results show that the model is quite effective in discovering the latent common themes across collections and performs significantly better than our baseline mixture model.

1. INTRODUCTION

Text mining is concerned with extracting knowledge and patterns from text [5, 6]. While there has been much research in text mining (see, e.g., [11, 2]), most existing research is focused on one single collection of text. The goals are often to extract basic semantic units such as named entities, to extract relations between information units, or to extract topic themes. In this paper, we study a novel problem of text mining referred to as comparative text mining. Given a set of comparable text collections, the task of comparative text mining is to discover any latent common themes across all collections as well as summarize the similarity and differences of these collections along each common theme. Specifically, the task involves: (1) discovering the different common themes across all the collections; (2) for each discovered theme, characterize what is in common among all the collections and what is unique to each collection. The need for comparative text mining exists in many different applications, including business intelligence, summarizing reviews of similar products, and comparing different opinions about a common topic.

In this paper, we study the comparative text mining problem and propose a generative probabilistic mixture model for comparative text mining. The model simultaneously performs cross-collection clustering and within-collection clustering, and can be applied to an arbitrary set of comparable text collections. The mixture model is based on component multinomial distribution models, each characterizing a different theme. The common themes and collection-specific themes are explicitly modeled. The model can be estimated efficiently using the Expectation-Maximization (EM) algorithm.

We evaluate the model on two different text data sets (i.e., a news article data set and a laptop review data set), and compare it with a baseline clustering method also based on a mixture model. Experiment results show that the model is quite effective in discovering the latent common themes across collections and performs significantly better than our baseline mixture model.

The rest of the paper is organized as follows. In Section 2, we briefly introduce the problem of comparative text mining. We then present a baseline simple mixture model and a new cross-collection mixture model in Section 3 and Section 4. We discuss the experiment results in Section 5.

2. COMPARATIVE TEXT MINING

2.1 A motivating example

With the popularity of e-commerce, online customer evaluations are becoming widely provided by online stores and third-party websites. Pioneers like amazon.com and epinions.com have accumulated large amounts of customer input including reviews, comments, recommendations and advice, etc. For example, the number of reviews in epinions.com is more than one million[4]. Given a product, there could be up to hundreds of reviews, which is impossible for the readers to go through. It is thus desirable to summarize a
collection of reviews for a certain type of products in order to provide the readers the most salient feedbacks from the peers. For review summarization, the most important task is to identify different semantic aspects of a product that the reviewers mentioned and to group the opinions according to these aspects to show similarities and differences in the opinions.

For example, suppose we have reviews of three different brands of laptops (Dell, IBM, and Apple), and we want to summarize the reviews. A useful summary would be a tabular representation of the opinions as shown in Table 1, in which each row represents one aspect (subtopic) and different columns correspond to different opinions.

<table>
<thead>
<tr>
<th>Subtopics</th>
<th>Dell</th>
<th>IBM</th>
<th>Apple</th>
</tr>
</thead>
<tbody>
<tr>
<td>Battery life</td>
<td>long</td>
<td>short</td>
<td>short</td>
</tr>
<tr>
<td>Memory</td>
<td>good</td>
<td>bad</td>
<td>good</td>
</tr>
<tr>
<td>Speed</td>
<td>slow</td>
<td>fast</td>
<td>fast</td>
</tr>
</tbody>
</table>

Table 1: The tabular representation of the summary

It is, of course, very difficult, if not impossible to automatically produce such a table automatically. However, we can still achieve something close to this goal – identifying the semantic aspects and identifying the common and specific characteristics of each product. This is what we meant by comparative text mining.

2.2 The general problem

The example above is only one of the many possible applications of comparative text mining. In general, the task of comparative text mining involves: (1) discovering the common themes across all the collections; (2) for each discovered theme, characterize what is in common among all the collections and what is unique to each collection.

Comparative text mining is challenging in several ways: (1) It is a completely unsupervised learning task, and we do not have training data available. All we have is a set of comparable text collections. It is for the same reason that comparative text mining can be very useful for many different purposes – the collections can be comparable in many different ways. (2) We need to identify theme across different collections, which is more challenging than identifying topic themes in one single collection. (3) The task involves a discrimination component – we want to distinguish the common information content said about a theme from the special information content specific to one particular collection. Such a discrimination task is difficult given that we do not have training data. In a way, comparative text mining goes beyond the regular one-collection text mining by requiring an “alignment” of multiple collections based on common themes.

Since no training data is available, in general, we must rely on unsupervised learning methods, such as clustering, to perform comparative text mining. In this paper, we study how to use probabilistic mixture models to perform comparative text mining. Below we first describe a simple mixture model for clustering, which represents a straightforward application of an existing text mining method, and then present a more sophisticated mixture model specifically designed for comparative text mining.

3. CLUSTERING WITH A SIMPLE MIXTURE MODEL

A naive solution to comparative text mining is to treat the multiple collections as one single collection and perform clustering. Our hope is that some clusters would represent the common theme across the collections, while some others would represent themes specific to one collection. We now present a simple multinomial mixture model for clustering an arbitrary collection of documents. The basic idea is that we assume there are $k$ latent common themes in all collections. Each is characterized by a multinomial word distribution (also called a unigram language model [10]). We then assume that a document is a sample of a mixture model with these theme models as components. We fit such a mixture model to the union of all the text collections we have, and the obtained component multinomial models can be used to analyze the common themes and differences among the collections.

Formally, let $C = \{C_1, C_2, ..., C_m\}$ be m comparable collections of documents. Let $a_1, ..., a_k$ be $k$ theme aspects. Let $\theta_i$ be the unigram language model for aspect $a_i$, $\theta_B$ be the background model for all the collections. We assume that a document $d$ is a sample of the following mixture model:

$$p_d(w) = (1 - \lambda_B) \sum_{j=1}^{k} [\pi_{d,j} p(w|\theta_j)] + \lambda_B p(w|\theta_B)$$

where $\pi_{d,j}$ is a document-specific mixing weight for the $j$-th aspect theme, and $\sum_{j=1}^{k} \pi_{d,j} = 1$. $\lambda_B$ is the mixing weight of the background model $\theta_B$.

The reason why we want to use a background model is because it can force clustering to be done based on more discriminative words, leading to more informative and more discriminative component models. Clearly the log-likelihood
of the whole set of collections $C$ is

$$\log p(C|\Lambda) = \sum_{i=1}^{m} \sum_{d \in C_i} \sum_{w \in V} c(w, d) \times$$

$$\log[(1 - \lambda_B)\sum_{j=1}^{k} \pi_{d,j}p(w|\theta_j) + \lambda_B p(w|\theta_B)]$$

where $\Lambda = (\theta_1, ..., \theta_k)$ is the set of all the parameters. The model can be estimated using the Maximum Likelihood estimator

$$\hat{\Lambda} = \arg \max_{\Lambda} \log p(C|\Lambda)$$

The Expectation-Maximization (EM) algorithm can be used to compute this estimator. EM is an iterative optimization algorithm. For this simple model, the updating formulas are:

$$p(z_{d,w} = j) = \frac{\pi_{d,j}^{(m)}p^{(m)}(w|\theta_j)}{\sum_{j'=1}^{k} \pi_{d,j'}^{(m)}p^{(m)}(w|\theta_j)}$$

$$p(z_{d,w} = B) = \frac{\lambda_B p(w|\theta_B)}{\sum_{j=1}^{k} \pi_{d,j}^{(m)}p^{(m)}(w|\theta_j)}$$

$$\pi_{d,i}^{(m+1)} = \frac{\sum_{w \in V} p(z_{d,w} = j)}{\sum_{j=1}^{k} \sum_{w \in V} p(z_{d,w} = j)}$$

$$p(z_{d,w} = B) = \frac{\lambda_B p(w|\theta_B)}{\sum_{j=1}^{k} \pi_{d,j}^{(m)}p^{(m)}(w|\theta_j)}$$

$$\pi_{d,i}^{(m+1)} = \frac{\sum_{w \in V} p(z_{d,w} = j)}{\sum_{j=1}^{k} \sum_{w \in V} p(z_{d,w} = j)}$$

This mixture model is closely related to the probabilistic latent semantic indexing model (PLSI) proposed in [7], and represents a straightforward application of an existing single-collection text mining algorithm to the comparative text mining problem. Clearly such a simple model is insufficient at least for two reasons: (1) We have completely ignored the structure of collections. As a result, we may have clusters that represent only some of the collections, but not all of them. (2) There is no easy way to identify which theme cluster represents the common information across collections and which represents specific information to a particular collection. As we will show later in discussing the experiment results, this model is inadequate empirically either. Below we present a more sophisticated coordinated mixture model, which is specifically designed to perform comparative text mining and gives interesting text mining results in our experiments with two different comparative text mining tasks.

4. CLUSTERING WITH A CROSS-COLLECTION MIXTURE MODEL

4.1 The model

Our main idea for improving the simple mixture model for comparative text mining is to explicitly distinguish common theme clusters that characterize common information across all collections from special theme clusters that characterize collection-specific information. Thus we now consider $k$ latent common themes as well as a potentially different set of $k$ collection-specific themes. The model is thus significantly more complicated than the simple model, and has all the information we are interested in extracting explicitly modeled as a component in the mixture model. The sampling distribution of a word in a document is now collection-specific. Specifically, it would involve the background model, $k$ common theme models, and $k$ collection-specific theme models specifically defined for this particular collection.

Formally, let $C = \{C_1, C_2, ..., C_m\}$ be $m$ collections of documents. Let $a_1, ..., a_k$ be $k$ theme aspects. Let $\theta_i$, be the unigram language model for aspect $a_i$, $\theta_B$ be the background model for all the collections, and $\theta_{i,j}$ be the collection-specific theme model for aspect $a_i$ and collection $C_j$. A document $d$ from collection $C_i$ is assumed to be generated from a mixture model involving the following components:

- The background model $\theta_B$: This model supplies the general English words.
- $k$ common theme models $\theta_{1, \ldots, k}$: These models capture the common characteristics of the $k$ themes.
- $k$ collection-specific theme models $\theta_{1,i, \ldots, k,i}$: These models capture the special characteristics of the $k$ themes w.r.t. the particular collection $C_i$.

The sampling distribution of a word for document $d \in C_i$ using this mixture model is given by

$$p_d(w|C_i) = (1 - \lambda_B) \sum_{j=1}^{k} \pi_{d,j}(\lambda_C p(w|\theta_j) + (1 - \lambda_C)p(w|\theta_{j,i})) + \lambda_B p(w|\theta_B)$$

where $\lambda_B$ is the weight on the background model $\theta_B$ and $\lambda_C$ is the weight on the common theme model $\theta_j$ as opposed to the collection-specific theme model $\theta_{j,i}$. Intuitively, when we “generate” a word, we first decide whether to use the background model $\theta_B$ according to $\lambda_B$; the larger $\lambda_B$ is, the more likely we will use $\theta_B$. If we decide not to use $\theta_B$, then we need to decide which aspect to use. This is controlled by $\pi_{d,j}$, which is the probability of using aspect $j$ when generating words in $d$. Finally, once we decide which aspect to use, we still need to decide whether we should use...
the common theme model or the collection-specific theme model, and this is controlled by $\lambda_C$, the probability of using the common model. The weighting parameters $\lambda_B$ and $\lambda_C$ are expected to be set by the user, and their interpretation is as follows. $\lambda_B$ reflects our knowledge about how noisy the collections are. If we believe the text is verbose, then $\lambda_B$ should be set to a larger value. In our experiments, a value of 0.9–0.95 often works well. $\lambda_C$ indicates our emphasis on the commonality, as opposed to the speciality in comparative text mining. A larger $\lambda_C$ would allow us to learn a richer common theme model, whereas a smaller one would learn a weaker common theme model, but stronger special models.

The optimal value depends on the specific applications. The sampling distribution is illustrated in Figure 3.

According to this generative model, the log-likelihood of a document $d \in C_i$ would be

$$
\log p(d|C_i) = \sum_{w \in V} c(w,d) \log \left[ \lambda_B p(w|\theta_B) + (1 - \lambda_B) \sum_{j=1}^k \pi_{d,j} \lambda_C p(w|\theta_j) + (1 - \lambda_C) p(w|\theta_{j,i}) \right]
$$

and the log-likelihood of the whole set of collections is thus

$$
\log p(C) = \sum_{i=1}^m \sum_{d \in C_i} \sum_{w \in V} c(w,d) \log \left[ \lambda_B p(w|\theta_B) + (1 - \lambda_B) \sum_{j=1}^k \pi_{d,j} \lambda_C p(w|\theta_j) + (1 - \lambda_C) p(w|\theta_{j,i}) \right]
$$

### 4.2 Parameter estimation

The mixture model described above is extremely flexible with many parameters. It is thus necessary to regulate our model appropriately. First, we would like to have the flexibility for setting $\lambda_B$ and $\lambda_C$ as they depend on particular applications. Second, we can estimate the background model $\theta_B$ using all the available text in the $m$ text collections. That is,

$$
\hat{p}(w|\theta_B) = \frac{\sum_{i=1}^m \sum_{d \in C_i} c(w,d)}{\sum_{i=1}^m \sum_{d \in C_i} \sum_{w' \in V} c(w',d)}
$$

This leaves us with the following parameters to estimate:

- The common theme models: $\theta = \{\theta_1, ..., \theta_k\}$.
- The special theme models for each collection $C_i$: $\theta_{C_i} = \{\theta_{i,1}, ..., \theta_{i,k}\}$.
- The aspect mixing weights for each document $d$: $\pi_d = \{\pi_{d,1}, ..., \pi_{d,k}\}$.

We estimate these parameters using the Maximum Likelihood (ML) estimator, i.e.,

$$
\hat{\Lambda} = \arg \max_\Lambda \log p(C|\Lambda)
$$

where $\Lambda = (\theta, \theta_{C_1}, ..., \theta_{C_m}, \{\pi_d\}_{d \in C_1}, ..., \{\pi_d\}_{d \in C_m})$ represents all the parameters.

The Expectation-Maximization (EM) algorithm can be used to find a (local) maxima for $\hat{\Lambda}$. The updating formulas are shown in Figure 4.

### 4.3 Using the model

Once the model is estimated, we will have $k$ common theme models, and $k$ collection-specific models for each of the $m$ collections. Each of these models is a word distribution or unigram language model, corresponding to a cluster, thus we will have a total of $k$ common theme clusters. The high probability words can characterize the theme extracted. Such words can often be used directly as a special form of summary or indirectly to extract relevant sentences to form a summary. Actually, the word distributions can be used in many other ways, e.g., to classify other text documents or to link the related passages in the text collections so that a user can navigate the information space for comparative analysis.

We note that there are two parameters we need to set $\lambda_B$ and $\lambda_C$. This is intentional since we need them to control the bias in comparative text mining. With $\lambda_B$ we can input our knowledge about the noise(stop words) in the data. For example, if we know the text data is verbose, then we should set $\lambda_B$ to a high value, whereas if the data is concise and mostly content-bearing keywords, then we need to set $\lambda_B$ to a smaller value. Similarly, with $\lambda_C$, we can input our bias our “threshold” for similarity across collections, which is related to our emphasis on extracting common theme models (setting $\lambda_C$ to a higher value) vs. emphasis on extracting collection-specific models (setting $\lambda_C$ to a smaller value). These biases cannot be learned by the maximum likelihood estimator. Indeed, maximizing the data likelihood is not really our ultimate goal, which is why we do not intend for our model to be as free as possible. Instead, we want to regularize our model in a meaningful way so that we can impose certain preferences while maximizing the data likelihood. The flexibility and control provided by $\lambda_B$ and $\lambda_C$ make it possible for a user to control the focus of the results of comparative text mining.
5. EXPERIMENTS AND RESULT ANALYSIS

We have evaluated the proposed mixture models on two domains – war news and laptop reviews. We discuss the results in the following subsections.

5.1 War news

The War news data consists of news excerpts on two comparable events: (1) Iraq war and (2) Afghanistan war, both of which occurred in the last two years. The Iraq war excerpts were a combination of 30 articles from the CNN and BBC web sites over the last one year span. The Afghanistan war data consists of 26 news articles downloaded from the CNN and BBC web sites for one year starting from Nov. 2001. Our goal is to compare these two wars and find out their common and specific characteristics.

5.1.1 Cross-Collection Mixture Model Results

The results of the proposed cross-collection mixture model are arranged in Table 2, where we show the top 5 words along with their probabilities from the common theme models and the top 5 words from the Iraq-specific theme models, and Afghanistan-specific theme models, respectively, in the descending order of probabilities. These results are obtained by fixing the number of clusters to five and setting $\lambda C = .25$ and $\lambda B = .91$. Variations of these parameters are discussed later.

These clusters can be interpreted as follows. Note that while interpreting the clusters, we may refer to some high probability words in a model that are not included in the table.

cluster1: In common theme words category of cluster1, us, nation, action are the top ranking words. We can make a semantic understanding that these words indicate the U.S military action in Iraq and Afghanistan. In the Iraq theme words category god, saddam, baghdad, live and victorious are among the words. The semantic context of these words is the speech of Saddam Hussein referring to God and in defending the Iraqi nation in the event of U.S attack. In the Afghanistan theme words category, paper, afghan, meeting, euro, highway and refugees are the top ranking words. The semantic understanding of these words is the strife in afghanistan and European union playing an official role in helping the refugees and highway work.

cluster2: In cluster2, mr, marines, defense, key, dead, general are the top ranking words in the common theme words category. These words give a semantic context that in both the wars the U.S marines are involved and there is a key role for a defense department general. In the Iraq theme words category the top words iraq,us, baghdad, nato, kuwait, and in the Afghanistan theme words category story, full, rabbani, mazar, sharif are some of the top ranking words. Rabbani refers to a leader of a group in Afghanistan, mazar, sharif refer to the place Mazar-e-Sharif in Afghanistan were this group had its first victory.

cluster3: In cluster3, killed, month, deaths, died are some of the top ranking words. The semantic context inferred from these words is that in both these wars there has been a huge loss of life, which could mean both civilian and military. In the Iraq theme words category, troops, hoon, sanchez, billion, spokesman, soldier are the top ranking words. Hoon is the last name of the british defence secretary and Sanchez is the last name of the U.S General in Iraq. These words quite clearly point to the semantic category of the important defence people of the allied forces in the Iraq war. The top ranking words in the Afghan theme words category are , taleban, runsfeld, hotel, front, dropped, bombing, afghanistan. These words refer to the U.S Defence secretary who had an important role in the Afghan war and to the bombs being dropped in this war in Afghanistan.

cluster4: The top ranking words in the common theme words category are, monday, official, do, political, spokesman, administration. These words refer to the Monday briefings by an official spokesman of a political administration during both the wars. The top ranking words in the Iraq theme words category are , intelligence, weapons, inquiry, commission, independent, hutton, destruction, mass. We can infer that the semantic context of these words is the inquiry into
Table 2: cross-collection mixture model results on War news data

<table>
<thead>
<tr>
<th>Common theme words</th>
<th>Cluster1</th>
<th>Cluster2</th>
<th>Cluster3</th>
<th>Cluster4</th>
<th>Cluster5</th>
</tr>
</thead>
<tbody>
<tr>
<td>us</td>
<td>0.042</td>
<td>mr</td>
<td>killed</td>
<td>Monday</td>
<td>united</td>
</tr>
<tr>
<td>nation</td>
<td>0.0299</td>
<td>marines</td>
<td>month</td>
<td>official</td>
<td>nations</td>
</tr>
<tr>
<td>will</td>
<td>0.0238</td>
<td>dead</td>
<td>deaths</td>
<td>i</td>
<td>with</td>
</tr>
<tr>
<td>action</td>
<td>0.022</td>
<td>general</td>
<td>one</td>
<td>would</td>
<td>is</td>
</tr>
<tr>
<td>re</td>
<td>0.0216</td>
<td>defense</td>
<td>died</td>
<td>where</td>
<td>it</td>
</tr>
<tr>
<td>border</td>
<td>0.0194</td>
<td>key</td>
<td>been</td>
<td>do</td>
<td>they</td>
</tr>
<tr>
<td>its</td>
<td>0.0171</td>
<td>since</td>
<td>drive</td>
<td>spokesman</td>
<td>diplomatic</td>
</tr>
<tr>
<td>ve</td>
<td>0.0161</td>
<td>first</td>
<td>according</td>
<td>political</td>
<td>blair</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Iraq theme words</th>
<th>god</th>
<th>iraq</th>
<th>troops</th>
<th>intelligence</th>
<th>n</th>
</tr>
</thead>
<tbody>
<tr>
<td>saddam</td>
<td>0.022</td>
<td>us</td>
<td>hoon</td>
<td>weapons</td>
<td>weapons</td>
</tr>
<tr>
<td>baghdad</td>
<td>0.0129</td>
<td>baghdad</td>
<td>sanchez</td>
<td>inquiry</td>
<td>inspectors</td>
</tr>
<tr>
<td>your</td>
<td>0.0124</td>
<td>nato</td>
<td>billion</td>
<td>commission</td>
<td>council</td>
</tr>
<tr>
<td>live</td>
<td>0.01</td>
<td>iraq</td>
<td>spokeswoman</td>
<td>independent</td>
<td>declaration</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Afghan theme words</th>
<th>paper</th>
<th>story</th>
<th>taleban</th>
<th>bin</th>
<th>northern</th>
</tr>
</thead>
<tbody>
<tr>
<td>afghan</td>
<td>0.019</td>
<td>full</td>
<td>rumsfeld</td>
<td>laden</td>
<td>alliance</td>
</tr>
<tr>
<td>meeting</td>
<td>0.0139</td>
<td>saturday</td>
<td>hotel</td>
<td>stemberg</td>
<td>kabul</td>
</tr>
<tr>
<td>euro</td>
<td>0.0121</td>
<td>e</td>
<td>front</td>
<td>taliban</td>
<td>taleban</td>
</tr>
<tr>
<td>highway</td>
<td>0.0118</td>
<td>rabbani</td>
<td>dropped</td>
<td>chat</td>
<td>aid</td>
</tr>
</tbody>
</table>

the presence of weapons of mass destruction in Iraq. Lord Hutton whose last name is one of the top ranking words is the judge making this probe in Britain. In the Afghan theme words category the top ranking words are, bin, laden, steinberg, taliban, afghanistan. James Steinberg is the head of Foreign Policy Studies at the Brookings Institution, his last name is one of the top ranking words. He was interviewed on many occasions by CNN on the war strategy in Afghanistan. The other words refer to taliban, which was ruling Afghanistan prior to the war and bin laden who had a strong support base from the Taliban in Afghanistan.

5.1.2 Simple Mixture Model Results

The results of using the simple mixture model are shown in Table 3. The value of $\lambda_B = 0.95$. The number of clusters is 5.

Cluster1: The top ranking words in this cluster are, will, let, united, god, inspectors, your, nation, n. These words are semantically incoherent. There is no semantic theme for this cluster.

Cluster2: The top ranking words in this cluster are, british, soldiers, baghdad, air, basra, mosque, southern, fired. The words british, basra, southern lets us infer that the semantic theme is the presence of british soldiers in Basra, a townin southern iraq.

Cluster3: The top ranking words in this cluster are, weapons, kay, rumsfeld, commission, senate, survey, paper. From the words weapons, kay, rumsfeld, senate we can loosely infer that the semantic theme is the American senate enquiry into the presence of weapons. This is a loose semantic
Table 3: Results on War news data using simple mixture model

<table>
<thead>
<tr>
<th>Common Theme Words</th>
<th>Cluster1</th>
<th>Cluster2</th>
<th>Cluster3</th>
<th>Cluster4</th>
<th>Cluster5</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>will</td>
<td>british</td>
<td>weapons</td>
<td>inquiry</td>
<td>countries</td>
</tr>
<tr>
<td></td>
<td>0.0189</td>
<td>0.0172</td>
<td>0.0215</td>
<td>0.052</td>
<td>0.026</td>
</tr>
<tr>
<td></td>
<td>let</td>
<td>soldiers</td>
<td>lay</td>
<td>intelligence</td>
<td>contracts</td>
</tr>
<tr>
<td></td>
<td>0.0119</td>
<td>0.01529</td>
<td>0.0211</td>
<td>0.0355</td>
<td>0.0234</td>
</tr>
<tr>
<td></td>
<td>united</td>
<td>baghdad</td>
<td>runsfeld</td>
<td>dossier</td>
<td>allawi</td>
</tr>
<tr>
<td></td>
<td>0.0118</td>
<td>0.0152</td>
<td>0.017</td>
<td>0.0236</td>
<td>0.0123</td>
</tr>
<tr>
<td></td>
<td>god</td>
<td>air</td>
<td>commission</td>
<td>hutton</td>
<td>boon</td>
</tr>
<tr>
<td></td>
<td>0.0111</td>
<td>0.0112</td>
<td>0.0144</td>
<td>0.0207</td>
<td>0.0117</td>
</tr>
<tr>
<td></td>
<td>inspectors</td>
<td>basra</td>
<td>group</td>
<td>claim</td>
<td>russian</td>
</tr>
<tr>
<td></td>
<td>0.0109</td>
<td>0.0108</td>
<td>0.0144</td>
<td>0.0188</td>
<td>0.0103</td>
</tr>
<tr>
<td></td>
<td>your</td>
<td>mosque</td>
<td>senate</td>
<td>wmd</td>
<td>international</td>
</tr>
<tr>
<td></td>
<td>0.0103</td>
<td>0.0104</td>
<td>0.0111</td>
<td>0.0187</td>
<td>0.0097</td>
</tr>
<tr>
<td></td>
<td>nation</td>
<td>southern</td>
<td>survey</td>
<td>mps</td>
<td>russia</td>
</tr>
<tr>
<td></td>
<td>0.0102</td>
<td>0.01</td>
<td>0.0101</td>
<td>0.018</td>
<td>0.0091</td>
</tr>
<tr>
<td></td>
<td>n</td>
<td>fired</td>
<td>paper</td>
<td>committee</td>
<td>reconstruction</td>
</tr>
<tr>
<td></td>
<td>0.0097</td>
<td>0.0097</td>
<td>0.0096</td>
<td>0.0173</td>
<td>0.00915</td>
</tr>
</tbody>
</table>

theme.

Cluster4: The top ranking words in this cluster are, inquiry, intelligence, dossier, button, claim, wmd, committe. Hutton is the last name of Lord Hutton who is heading the inquiry of presence of WMD in Iraq. It is evident from the other words that the semantic theme is the British inquiry into presence of WMD in Iraq.

Cluster5: The top ranking words in this cluster are, countries, contracts, allawi, hoon, russian, international, russia, reconstruction. We can loosely infer from these words that the semantic theme is the denial of contracts to some countries like Russia.

Even in the case of War news the results of the simple mixture model are bad, compared to our cross-collection mixture model. In this case we could only loosely infer 3 semantic themes for the 5 clusters.

5.2 Laptop Customer Reviews

This data set was constructed to test our models for comparing opinions of customers on different laptops. We manually downloaded the following 3 review sets from opinions.com [4], filtering out the misplaced ones: Apple iBook (MS598LL/A) Mac Notebook (34 reviews), Dell Inspiron 8200 (8TWORH) PC Notebook (22 reviews), IBM ThinkPad T20 2647 (264744U) PC Notebook (42 reviews).

5.2.1 Cross-Collection Mixture Model Results

The results of the proposed cross-collection mixture model are arranged in Table 4, where we show the top 8 words along with their probabilities from the common theme models and the top 5 words from the three laptop-specific theme models, respectively, in the descending order of probabilities.

These results are obtained by fixing the number of clusters to eight and setting $\lambda_C=.7$ and $\lambda_B=.96$.

cluster1: In cluster1, sound, speakers, playback, feel, pros, cons, market are the top ranking words in the common theme words category. The semantic context of these words is that in all the customer reviews audio devices or their characteristics affect the market depending on their pros and cons. The top ranking words in dell category do not have a strong correlation, but they all are some of the features of dell laptops. The top two words in apple laptop category are magazine, ipod. It is understood that in the apple theme category, ipod an apple music player compatible with the apple laptop is being described in a magazine. In the IBM category, the semantic theme in this case appears to be the good features of trackpoint device on IBM laptops in terms of reducing the stress for a user.

cluster2: The top ranking words in the common theme words category are, port, jack, ports, will, your, warm, keep, down. Port and jack both refer to I/O device terminals in a laptop. Hence the common theme semantic category associated with this cluster is I/O device terminals in a laptop. In the Dell category the top ranking word is banias, which indicates that the banias mobile platform recently released a feature in dell laptops. In the apple category, exx and quartz are the top two ranking words. Mac OS X is the an advanced operating system introduced in apple laptops and quartz is feature available in its architecture. The words OSX and quartz together refer to this operating system, which is the semantic context of the apple category. In the IBM category, it is unclear what the top ranking words capture.

cluster3: The top ranking words in the common theme category are, ram, mb, memory, 256mb, 128mb, tech. These words give us a semantic understanding that all the three laptop reviews have a description of memory devices such as RAM and their configurations such as 256mb. In the dell category the top ranking words appear to be related to some options available on dell laptop like eraser, sodimm, sdram. The top ranking words in the apple category include macos and airport. Analyzing these words macos is the MAC operating system on apple laptops, and airport is a wireless card slot in apple laptops. Again the semantic context of these words is some special features of apple laptops not mentioned previously. In the IBM category, again it is unclear what the top words capture.

cluster4: The top ranking words in the common theme words category are, n, trackpad, chip, improved, volume, did, latch, make, intel. In this case there is no strong semantic correlation between the top ranking words of the common theme category, though we have some prominent common laptop features like a trackpad getting a high rank. In the Dell category the top ranking words are, inspiron, pentium, 8200. It is quite evident that the semantic context
of these words is the Dell Inspiron 8200 laptop. The top ranking words in the Apple category are, macos, netchap, apache, i5. The semantic context of these words is the type of browser available on Apple laptops along with the MAC OS. Netchap and i5 are both compatible. In the IBM category, again it is unclear what the top words capture.

**cluster5:** The top ranking words in the common theme words category are, battery, hours, life, 5, end, 3, high, processor. The semantic theme of this category is Battery life. The words battery, hours, life, high and the numbers 5, 3 make it evident that in all three reviews battery life is being discussed in terms of the number of hours a charged battery lasts. The word “dells” is the top ranking word in the dell category, which may indicate the presence of a dell theme. The top ranking words in the Apple category are, iphoto, itunes, import, quicktime, ipod. Apple-iPhoto is a software for storing and accessing digital video files. The top ranking words in the dell category seem to be related to shipping and tracking of products. The top ranking words in the IBM category are, thinkpad, ibm, covers, lightest, 3000, composite. It can be inferred from these words that the semantic theme is the physical nature of IBM thinkpad laptop. The physical nature is characterized by the words lightest, composite.

**cluster6:** The top ranking words in the common theme words category are, t, modem, internet, later, configuration, free, vga. From these words we infer that the semantic theme in this category is about communication devices like modem and communication medium like internet. The top ranking words in the dell category seem to be related to shipping and tracking of products. The top ranking words in the Apple category are, iphoto, itunes, import, book, quicktime, imovie. The semantic theme associated with this category is listing of softwares for storing and accessing multimedia file formats. Apple-iPhoto is a software for storing and accessing digital images, Apple-iTunes is a music jukebox and storage software for music files. Quicktime and iMovie are used for playing and storing digital video files. The top ranking words in the IBM category are, thinkpad, ibm, connector, lightest, bluetooth, device, cap. These words indicate that the semantic context of these words is the compatibility of IBM Thinkpads and related accessories.
mon theme in all the customer reviews. The top ranking words in the Dell category do not seem to indicate a semantic theme. While some of the top ranking words in the Apple category, such as airport, are clearly Apple-specific, it is unclear how to interpret other top words. The top ranking words in the IBM category are, t20, ultrabay, tells, device, number, 600x, t23. We infer from the words t20 and t23 that they are describing the IBM Thinkpad series and from the words ultrabay and device that IBM laptop compatible devices are also being described.

cluster8: The top ranking words in the common theme words category are, office, microsoft, little, basic, 6, under, word. The semantic theme of this category is Microsoft products like office, basic and word in laptops. We cannot infer any semantic theme from the top words in Dell category or IBM category. But the top ranking words in the Apple category all refer to utility softwares available on apple laptops.

5.2.2 Simple Mixture Model Results
The results of using the simple mixture model are shown in Table 5. The value of $\lambda_2 = 0.95$. The number of clusters is 8.

Cluster1 The top ranking words are, port, ports, usb, modem, firewire, 56k, ethernet, jack. These words do not have a strong semantic theme in common, because the words port, ports, usb, jack describe I/O devices in laptop. While the words modem, firewire, 56k, ethernet describe communication devices or medium.

Cluster2 The top ranking words are, m, support, feel tech, athlon, me, cons, told. It is evident that these words are in no semantic theme in common. Athlon is the name of a processor while the other words like support, me, cons, told and tech are not semantically related to it.

Cluster3 The top ranking words in this cluster are, ram, mb, memory, 256mb, 128, uxga, oes, multi. A majority words give us a semantic understanding that all the three laptop reviews have a description of memory devices such as RAM and their configurations such as 256mb. Words such as uxga, oes, multi are not conveying any semantic theme. Hence, even this cluster does not have a good semantic theme.

Cluster4 The top ranking words in this cluster are, chip, radeon, wil, processor, 9000, far, 440, chipset. The words radeon and processor indicate a processor brand, but other words have nothing in common with this theme. Hence this cluster does not have a semantic theme.

Cluster5 The top ranking words in this cluster are, display, he, sleep, lid, real, worse, open, ugly. It is evident that we do not get a semantic theme from these words.

Cluster6 The top ranking words in this cluster are, os, box, old, application, warranty, operating, speeds, 2002. We cannot infer a semantic theme from these words.

Cluster7 The top ranking words in this cluster are, battery, hours, 5, life, word, weight, 6 and 8. We can infer from the high probability words in this cluster that, the semantic theme is battery life.

Cluster8 The top ranking words in this cluster are, speakers, sturdy, nice, really, keep, if, things and lets. From the words speakers, sturdy and nice we infer that the semantic theme in physical nature of speakers in a laptop.

In the case of the cross-collection mixture model results almost all the 8 clusters had a semantic theme. While using the simple mixture model we have got only 2 semantic theme clusters.

5.3 Discussion
The results shown above are obtained from a specific setting of parameters. When we vary the parameters, the results are generally different. When $\lambda_2$ is set to a small value, non-informative stop words tend to show up in common themes. A reasonable value for $\lambda_2$ is generally higher than 0.9 – in that case, the model automatically eliminate the non-informative words from the theme clusters, allowing for more discriminative clustering. Indeed, in all our experiments, we have retained all the stop words. The parameter $\lambda_C$ affects the vocabulary allocation between the common and collection-specific themes. In the news data experiments, when we change $\lambda_C$ to a value above 0.4, the collection-specific terms would dominate the common theme models. In the laptop data experiments, when $\lambda_C$ is less than 0.7, we lose many content keywords of the common themes to the corresponding collection-specific themes, as expected.

6. RELATED WORK
The most related work to our work is the coupled clustering method presented in [8], which appears to be one of the very few studies considering the clustering problem in multiple collections. They extend the information bottleneck approach [12] to discover common clusters across different collections. Comparative text mining goes beyond this by analyzing both the similarities and collection-specific differences. We also use a completely different approach based on probabilistic mixture models. The aspect models studied in [7, 3] are also related to our work but they are more close to our baseline model and are not designed for comparing multiple collections. There are lots of studies in document clustering [1]. Again, the difference lies in that they consider only one collection and thus are similar to the baseline model.

Our work is also related to document summarization, especially multiple document summarization (e.g., [9, 13]). Indeed, we can regard comparative text mining as a special form of summary of multiple text collections. However, an important difference is that while a summary intends to retain the explicit information in text (to maintain fidelity), comparative text mining aims at extracting non-obvious implicit patterns.

7. CONCLUSIONS AND FUTURE WORK
In this paper, we define and study a novel text mining problem referred to as comparative text mining. It has to do with discovering any latent common themes across a set of comparable collections of text as well as summarizing the similarity and differences of these collections along each common theme.

We propose a generative cross-collection mixture model for performing comparative text mining. The model simultaneously performs cross-collection clustering and within-collection clustering, and can be applied to an arbitrary set of comparable text collections. We define the model and present the EM algorithm that can estimate the model efficiently. We evaluate the model on two different text data sets (i.e., a news article data set and a laptop review data set), and compare it with a baseline clustering method based on a simple aspect mixture model. Experiment results show
that the cross-collection mixture model is quite effective in discovering the latent common themes across collections and performs significantly better than the baseline simple mixture model. The proposed model is directly usable for many different purposes, e.g., comparing the course web pages from the major computer science department web sites to discover the core computer science topics.

The work reported in this paper is just an initial step toward a promising new direction. There are many interesting future research directions. First, it may be interesting to explore the Maximum A Posterior (MAP) estimation of the proposed mixture model, which would allow us to incorporate more prior knowledge in a principled way. For example, a user may already have certain thematic aspects in mind. With MAP estimation, we can easily add that bias to the component models. Second, we can generalize our model to model semi-structured data to perform more general comparative data mining. One way to achieve this goal is to introduce additional random variables in each component model so that we can model any structured data. Finally, it would be very interesting to explore how we could exploit the learned theme models to provide additional help to a user who wants to perform comparative analysis. For example, the learned common theme models can be used to construct a hidden Markov model (HMM) to identify the parts in the text collections about the common themes, and to connect them through automatically generated hyperlinks. This would allow a user to easily navigate through the common themes.

### 8. REFERENCES


---

Table 5: Results from the Customer reviews laptops using simple mixture model

<table>
<thead>
<tr>
<th>common theme words</th>
<th>Cluster1</th>
<th>Cluster2</th>
<th>Cluster3</th>
<th>Cluster4</th>
<th>Cluster5</th>
<th>Cluster6</th>
<th>Cluster7</th>
<th>Cluster8</th>
</tr>
</thead>
<tbody>
<tr>
<td>port</td>
<td>0.0419</td>
<td>0.0166</td>
<td>0.0316</td>
<td>0.0125</td>
<td>0.012</td>
<td>0.013</td>
<td>0.0504</td>
<td>0.0106</td>
</tr>
<tr>
<td>ports</td>
<td>0.0304</td>
<td>0.0161</td>
<td>0.0169</td>
<td>0.0118</td>
<td>0.0087</td>
<td>0.0089</td>
<td>0.0314</td>
<td>0.0104</td>
</tr>
<tr>
<td>sub</td>
<td>0.0286</td>
<td>0.01529</td>
<td>0.0146</td>
<td>0.0106</td>
<td>0.007</td>
<td>0.00736</td>
<td>0.0224</td>
<td>0.0007</td>
</tr>
<tr>
<td>modem</td>
<td>0.0257</td>
<td>0.0111</td>
<td>0.0143</td>
<td>0.0084</td>
<td>0.0073</td>
<td>0.0067</td>
<td>0.0218</td>
<td>0.00843</td>
</tr>
<tr>
<td>firewire</td>
<td>0.0179</td>
<td>0.009</td>
<td>0.0099</td>
<td>0.0081</td>
<td>0.0071</td>
<td>0.00672</td>
<td>0.0116</td>
<td>0.00834</td>
</tr>
<tr>
<td>56k</td>
<td>0.015</td>
<td>0.009</td>
<td>0.0099</td>
<td>0.0076</td>
<td>0.0066</td>
<td>0.00661</td>
<td>0.01</td>
<td>0.008</td>
</tr>
<tr>
<td>ethernet</td>
<td>0.015</td>
<td>0.0084</td>
<td>0.0091</td>
<td>0.0061</td>
<td>0.0065</td>
<td>0.0059</td>
<td>0.0091</td>
<td>0.0078</td>
</tr>
<tr>
<td>jack</td>
<td>0.0129</td>
<td>0.0083</td>
<td>0.009</td>
<td>0.0058</td>
<td>0.0064</td>
<td>0.0059</td>
<td>0.0087</td>
<td>0.0076</td>
</tr>
</tbody>
</table>