Topic Model Methods for Automatically Identifying Out-of-Scope Resources

Steven Bethard
Stanford University
353 Serra Mall
Stanford, CA 94305, USA
bethard@stanford.edu

Soumya Ghosh
University of Colorado
430 UCB
Boulder, CO 80309, USA
soumya.ghosh@colorado.edu

James H. Martin
University of Colorado
430 UCB
Boulder, CO 80309, USA
james.martin@colorado.edu

Tamara Sumner
University of Colorado
430 UCB
Boulder, CO 80309, USA
tamara.sumner@colorado.edu

ABSTRACT

Recent years have seen the rise of subject-themed digital libraries, such as the NSDL pathways and the Digital Library for Earth System Education (DLESE). These libraries often need to manually verify that contributed resources cover topics that fit within the theme of the library. We show that such scope judgments can be automated using a combination of text classification techniques and topic modeling. Our models address two significant challenges in making scope judgments: only a small number of *out-of-scope* resources are typically available, and the topic distinctions required for digital libraries are much more subtle than classic text classification problems. To meet these challenges, our models combine support vector machine learners optimized to different performance metrics and semantic topics induced by unsupervised statistical topic models. Our best model is able to distinguish resources that belong in DLESE from resources that don't with an accuracy of around 70%. We see these models as the first steps towards increasing the scalability of digital libraries and dramatically reducing the workload required to maintain them.

Categories and Subject Descriptors

H.3.7 [**Information Storage and Retrieval**]: Digital Libraries; I.2.7 [**Artificial Intelligence**]: Natural Language Processing

General Terms

Algorithms

JCDL'09, June 15–19, 2009, Austin, Texas, USA.
Copyright 2009 ACM 978-1-60558-322-8/09/06 ...$5.00.

Keywords

digital libraries, scope, relevance, topics, machine learning

1. INTRODUCTION

The last decade has witnessed significant activity in the development of digital repositories and libraries focused on specific disciplinary topics or communities. For instance, early pioneers in this area include SOSIG, the Social Science Information Gateway [12], and Renardus [13], which aimed to establish a pan-European subject gateway service. Other efforts have focused on creating e-print services to serve specific academic disciplines, such as cognitive science, cryptology, or physics, with arXiv [10] being a prominent example. In recent years, educational digital libraries have been developed for a variety of different topics. Often these libraries select a particular subject area, say, biological sciences or physics and astronomy, and then try to gather only the educational resources that are useful for learning or explaining these topics. Such subject-themed digital libraries are common, for example, the National Science Digital Library (NSDL) [26] calls them "pathways" and identifies a number of such pathways including BiosciEdNet [3], which covers biological sciences, and ComPADRE [7], which covers physics and astronomy.

To maintain useful, high quality, subject-themed collections, it is crucial for library efforts such as these to assess the new resources they receive, and include only those which fit within the scope of the collection. Currently, these scope assessments must be performed manually, considering resource content and metadata when available, but often relying heavily on the expertise of humans who are familiar with the library and understand its scope. However, as the quantity of user generated content continues to rise, manually assessing resource scope becomes less and less tractable. For instance, arXiv now has over a half million articles grouped into five subject areas; as the rate of user contributions grows, so does the effort at maintaining the integrity of these subject-based groupings. While user-generated content usually arrives a document at a time in e-print archives, it is not uncommon for digital libraries to receive a batch of several hundred or even thousands of resources, each of which must be manually sorted into "relevant" or "not relevant" groups. This is very time consuming, and typically requires expert subject knowledge and substantial familiarity with the digital library.

Thus, there is a need for tools that can automatically determine whether or not a digital resource is within the scope of a digital library, e-print archive, or other subject-themed repository. From a machine learning perspective, this seems like a standard text classification problem. That is, given a resource, the model need only examine the text, and then produce a classification of either "within-scope" or "out-of-scope". There is a large body of work on training and using such machine learning models, and in other domains, they have achieved high levels of accuracy [9, 19].

However, subject-themed digital libraries have some complexities that make them more challenging than classic text classification problems. First, the topic distinctions required for digital libraries are often more subtle than those in classic text classification. For example, the Reuters Corpus Volume I (RCV1) collection [19], which is commonly used for text classification experiments, asks for distinctions between topics like economics, sports and war. Because these topics share little in terms of the words used to discuss them, machine learning models can often distinguish between these topics by considering just a few important words in each document, such as *economy*, *basketball* or *military*. Digital libraries on the other hand, must make distinctions between resources that share many words in common. Consider for example, the Digital Library for Earth Systems Education (DLESE) [8], which is an educational digital library containing a variety of resources covering Earth systems topics like hurricanes, plate tectonics, and climate change. DLESE must make distinctions like:

- how tsunamis are formed (*within-scope*) vs.
  how tsunamis destroy ships (*out-of-scope*)

- how diamonds are formed (*within-scope*) vs.
  how diamonds are cut (*out-of-scope*)

Thus, words like *tsunami* or *diamond* are not as good clues for making digital library scope decisions because they may occur in both *within-scope* and *out-of-scope* resources. Having many words that can occur in both *within-scope* and *out-of-scope* resources makes it difficult for classic text classification models to learn the proper distinctions.

Another problem for classic text classification that is presented by digital libraries is the lack of training data. For a digital library, *within-scope* resources can be obtained by simply taking all resources already present in the library. For example, all 13,000+ resources in DLESE can be used as examples of *within-scope* resources for DLESE. However, resources rejected from a digital library are more difficult to obtain because records are often maintained only for the resources accepted into the collection. We have worked with curators of the DLESE collections to collect records for just over 600 resources that were rejected from this collection. While this is a decent number of resources in digital library terms, this is a very small number of resources in text classification terms – for comparison, the RCV1 corpus makes hundreds of thousands of negative examples available.

Our approach to automatically assessing digital library scope combines classic text classification with techniques from statistical topic modeling. Statistical topic models automatically extract the topics or subjects that a collection is most concerned with. For example, in DLESE, topic models identify topics about oceans and currents, light and solar radiation, fossils and paleontology, etc. Our machine learning algorithms use these topics instead of simple word information to make their decisions about library scope. We show that using topics instead of words allows for the more subtle distinctions required by digital libraries to be made. It also turns out that classifiers based on topic models are better able to handle the lack of negative examples than word-based classifiers.

The rest of this article explains our approach. Section 2 discusses classic approaches to text classification and prior uses of statistical topic models. Section 3 describes the resources we have collected from DLESE for use in training and evaluating our models. Since DLESE serves as the testbed for our research, Earth system subjects will often provide the examples we use in this article to explain our approach. Section 4 presents our machine learning approach and the results of evaluating our models, and Section 5 discusses the implications of our findings and ideas for future work.

2. PRIOR WORK

2.1 Text Classification

The term *text classification* is used to describe any situation where the input is a set of texts or documents, and the output is an assignment of one or more categories to each of the documents. The categories assigned may be almost anything, for example, subject areas (e.g. *biology*, *chemistry*), approval ratings (e.g. *one-star*, *five-star*) or relevance judgments (e.g. *within-scope*, *out-of-scope*).

It is common practice to build text classifiers using machine learning [23, 29, 33, 34]. Under this paradigm, machine learning algorithms are presented with a large set of documents, where each document is accompanied by the category label that it should be assigned. These category labels are usually manually annotated by humans who are familiar with the domain. The machine learning algorithm then inspects the documents and their hand-annotated categories, and searches for patterns in the data which might predict the categories. Different learning algorithms do this in different ways, but in the end, they all produce models which are able to apply the patterns they have learned to new texts they are presented with.

A popular approach to text classification starts by representing documents based on *TF-IDF word weighting* [14, 16, 29, 30]. TF-IDF word weighting is a way of converting a document into a list of numbers that is easy for a machine learning algorithm to handle. It relies on the intuition that the words that are most predictive of the category of a document will occur many times within a single document and in only a small number of the total documents. For example, seeing the word *climate* only once is more predictive than seeing the word *the* 20 times, because *climate* occurs in a small number of documents, while *the* occurs in practically all documents. Formally, given a *word*, a *document* and a *corpus* of documents, TF-IDF assigns the weight:

$$\frac{|\{w \in \mathit{document} : w = \mathit{word}\}|}{|\{w \in \mathit{document}\}|} \cdot \log \frac{|\{d \in \mathit{corpus}\}|}{|\{d \in \mathit{corpus} : w \in d\}|}$$

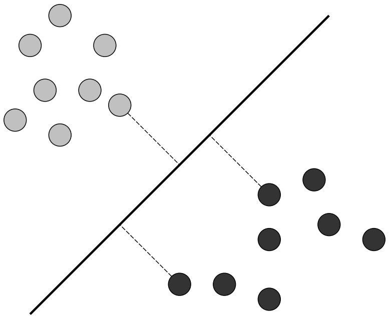

Figure 1: A separating hyperplane drawn by a support vector machine classifier. The dashed lines identify the support vectors.

In essence, this formula means that the weight of a word in a document increases with the number of times that word occurs in the document, and decreases with the number of documents in the corpus that contain that word.

Some of the most successful text classifiers combine this TF-IDF weighting with *support vector machine* (SVM) classifiers [14, 29, 30]. Support vector machines are a type of machine learning model which try to separate examples of one category from examples of another category by maximizing the margin of separation between the two categories. For example, Figure 1 shows how an SVM would separate an artificial dataset where the light points are of one category and the dark points are of another. SVMs can often be quite efficient because in order to make new classifications, they only need to store the examples that were closest to the separating hyperplane, known as the *support vectors*. Figure 1 identifies the example support vectors with dashed lines.

For many text classification datasets, SVMs using TF-IDF weighting achieve state of the art performance. For example, on the Reuters Corpus Volume I (RCV1) collection, an SVM with TF-IDF weighting was able to achieve an average F-measure[1]of 0.816, substantially outperforming other machine learning classifiers like k-Nearest Neighbor and Rocchio-Style Prototype classifiers [19]. As a result, SVMs using TF-IDF weighting are often the first approach applied to new text classification problems, and we use them as one of our baselines in the studies below.

2.2 Statistical Topic Models

Where the TF-IDF representation of a document encodes each word individually, *statistical topic models* try to characterize a document based on the *topics* it contains. Consider for example, a sentence like:

> *Bright surfaces like snow, ice, and clouds reflect the most energy, while dark surfaces like open ocean absorb the most.*

A TF-IDF model would simply describe this phrase as having two instances of *surfaces*, two instances of *most*, one instance of *bright*, and so on for each word in the sentence. In contrast, a statistical topic model might describe the sentence as being composed of 30% energy words (e.g. *bright*, *energy*, *dark*), 20% weather words (e.g. *snow*, *ice*, *clouds*), 10% marine words, etc. Note that the key difference here is that TF-IDF models think of documents as sets of words, while statistical topic models think of documents as mixtures of topics, where topics can be thought of as semantically meaningful groupings of words.

There are a variety of ways of automatically extracting such semantically meaningful groups. Typically these algorithms look at a large collection of texts, and try to condense all the words in all the texts into a smaller set of *topics* by considering which words appear together in the same or similar documents. Two popular algorithms for extracting topics are Non-negative Matrix Factorization (NMF) [17] and Latent Dirichlet Allocation (LDA) [5, 32]. NMF is based on viewing a collection of texts as a large term-by-document matrix (counting the number of times each term occurred in each document) and then applying a matrix factorization technique to split this into two smaller matrices: one which assigns topics to documents, and one which assigns topics to terms. LDA is based on an iterative process[2] where topics are first randomly assigned to each word, and then on each iteration until convergence, a new topic is assigned to each word based on how the topics are currently distributed across the collection.

Because algorithms like NMF and LDA can automatically extract semantically meaningful topics, they have obvious digital library applications, like identifying resource keywords and subject headings. For example, Newman and colleagues [24] applied LDA-based topic models to the OAIster collection [27], and hand labeled the resulting topics to create over 400 new subject headings. They suggested that, because these new subject headings could be automatically assigned with no human intervention, they could provide a substantially improved searching experience for users.

Mimno and McCallum also investigated using LDA-based topic models for assigning subject headings [22]. Mimno and McCallum pointed out that the size of the larger digital libraries was a problem for most existing topic model algorithms. (For example, Newman and colleagues were only able to use one third of the resources from OAIster.) Mimno and McCallum addressed this problem by designing a new topic model algorithm that could train models of individual books in parallel, and then cluster the book-level topics into corpus-level subject headings. They applied this model to a large set of books from the Open Content Alliance [28] and showed that they were often able to identify topics that matched Library of Congress subject headings.

Despite much active research in the areas of text classification and topic modeling, relatively little has been done to actively combine the two approaches. A few researchers have applied topic models to classification tasks, for example, Blei and McAuliffe's *Supervised Topic Model* [4] and Zhou and colleagues' *Latent Dirichlet Allocation Category Language Model* [35]. However, these models were compared against models which have historically performed poorer than SVMs with TF-IDF weighting, e.g. Naive Bayes or Rocchio classi-

[1] F-measure is a performance metric that balances credit between finding all texts about a particular category and not mislabeling other texts with that category. It is defined as the harmonic mean of *precision* and *recall*, where precision is the percent of the resources the model classified with a category that actually were of that category, and recall is the percent of resources known to be of a particular category that the model classified as such.

[2] There are actually many different ways of learning LDA models. This is a description of Gibbs sampling, which is easier to explain and widely implemented in machine learning libraries.

fiers. One of the major contributions of our paper is that it makes some direct comparisons between SVMs with TF-IDF weighting and SVMs with topic model features, and it does this by looking at real data obtained from digital libraries.

3. DIGITAL LIBRARY COLLECTIONS

Key to the evaluation of any tool based on machine learning models is gathering a set of human annotated data. In our case, this means gathering one group of resources that a human has identified as being *within-scope* of the digital library and another group of resources that a human has identified as being *out-of-scope* for the digital library.

As discussed in the introduction, gathering *within-scope* resources is relatively straightforward: every resource accepted into a digital library must be *within-scope* for that library. In our research, we used DLESE collections as our testbed because we were able to get access to human identified *out-of-scope* resources. Our colleagues at DLESE maintain records of resources that they have declined to include in the DLESE collections. Resources are declined for a number of reasons, including granularity concerns (e.g. the resource contains too many pages), poor quality of presentation, and inclusion of content which is out of the scope of the library. Fortunately, the DLESE team maintains comments for many of these declined resources, and using some simple keyword matching (e.g. *scope*, *oos* (out of scope), *Earth systems*) we were able to extract the resources that were declined primarily due to scope concerns.

Additional *out-of-scope* resources were also available from large collections contributed to DLESE. The Centers for Ocean Sciences Education Excellence (COSEE) offered up a few dozen resources that were a mix of marine biology and Earth science, and the Bridge Ocean Education Teacher Resource Center offered over 700 marine education resources. Catalogers inspected each resource, adding the resources relevant to Earth systems into DLESE, and declining the resources focusing primarily on the biological components of marine science, instead forwarding these resources to the National Science Digital Library (NSDL). From our colleagues at DLESE, we were able to obtain the records of which resources were added and which were declined.

In the end, we collected resources from three different DLESE sub-collections:

DCC: The DLESE Community Collection (DCC) is a high quality subset of the resources that have been contributed by the DLESE community. Some records were kept of contributed resources declined by DLESE, allowing us to use the keyword matching approach discussed above to select *out-of-scope* DCC resources.

COSEE: The resources from the Centers for Ocean Sciences Education Excellence (COSEE) cover a mix of marine biology and Earth systems topics, and, as mentioned above, were manually split by catalogers into accepted (*within-scope*) and declined (*out-of-scope*) sets.

BRIDGE: The resources from the Bridge Ocean Education Teacher Resource Center cover topics in marine education, and, like COSEE, were manually split by catalogers into accepted and declined resources.

We scraped the homepages of resources from these collections, gathering over 4000 resources, as shown in Table 1[3]. Because DCC covers a wide range of the topic areas that make up DLESE, we use this collection as our training and development data. Table 1 shows how the DCC data was split into a training section (DCC-train) and a development section (DCC-dev). While we were designing and training our scope classification models, we looked only at DCC-train and DCC-dev. The COSEE and BRIDGE data were reserved exclusively for our final evaluations.

Collection	Resources	*within-scope*	*out-of-scope*
DCC	3407	3331	76
DCC-train	3357	3306	51
DCC-dev	50	25	25
COSEE	39	14	25
BRIDGE	719	222	497

Table 1: Resources gathered from DLESE.

4. MACHINE LEARNING METHODS

As discussed earlier, the resource scope problem can be formulated as a text classification task. In this view of the problem, the resource scope model takes as input the text of a digital resource, and outputs a binary classification: either *within-scope* or *out-of-scope*. Our goal then, is to design an effective classifier that can serve as our resource scope model.

4.1 Establishing Baselines

Both to design our models and to evaluate their performance, it is crucial to get an idea of how difficult the task is. In machine learning research, this is typically achieved by designing a few *baseline* systems. Baseline systems range from very simple systems to re-implementations of other state-of-the-art systems. For our task, we consider four baseline systems:

Majority Class The simplest classifier possible, the Majority Class classifier simply looks at the training data, determines which category is the most common, and classifies all new resources with that category. In our case, the DCC training data indicates that *within-scope* is the most common category, and so the Majority Class classifier assumes that all resources are *within-scope*. This clearly a very poor classifier, so any decent classifier should outperform it.

Oracle Majority In most machine learning tasks, the distribution of categories in the training data looks much like the distribution of categories in the test data. For example, if 60% of categories in the training data are *within-scope*, then it is usually assumed that 60% of the categories in the testing data will also be *within-scope*. However, this is not true for our data: over 98% of the DCC training resources are annotated as *within-scope*, while only 36% of COSEE resources and only 30% of BRIDGE resources are annotated this way. The Oracle Majority classifier works much like the Majority Class classifier, but it cheats and looks at the test data to determine the most common category. This model

[3] We restricted ourselves to resources targeted at high school students. This reduced the number of *within-scope* resources somewhat, but allowed us to reuse the data in another related project.

Model	COSEE	BRIDGE
Majority Class	35.9	30.9
Multinomials	35.9	30.9
TF-IDF SVM	35.9	30.9
Oracle Majority	64.1	69.1

Table 2: Accuracy of baseline systems

serves as a comparison to a more ideal situation: how well could we do if we already knew approximately how many resources should be *within-scope* and how many should be *out-of-scope*?

Mixture of multinomials To compare against a model of intermediate complexity, we use a well known generative mixture model, the mixture of multinomials [25]. This model views each document as a mixture of words (all of which have some probability of appearing) and, given a large set of texts, can learn the probability of any particular text being part of that set. We train one mixture of multinomials model on the *within-scope* resources and one on the *out-of-scope* resources. To classify a new resource, we ask each model for the probability that our resource was generated by that model. If the probability is higher for the *within-scope* model, we classify the new resource as *within-scope*, and if the probability is higher for the *out-of-scope* model, we classify the new resource as *out-of-scope.* While such a model is expected to perform worse than an SVM with TF-IDF weighting, it should give us an idea of how well a basic machine learning model can perform on our digital library data.

SVM with TF-IDF weighting To determine the performance of state-of-the-art text classification techniques, we trained a support vector machine classifier[4] using the standard TF-IDF approach. We followed the usual procedures of removing all stop words (e.g. *the*, *in*, *to*, etc.), stemming the remaining content words (e.g. converting *dogs* to *dog*), and then applying the TF-IDF word weighting as discussed in the prior work section. The resulting model serves as a realistic approximation of how well current text classification techniques perform on a digital library scope classification task.

We evaluated these baseline models on the COSEE and BRIDGE data. When testing our models on the COSEE data, we used DCC-train to train the models, and then tuned the model parameters using the DCC-dev set. When testing our models on the BRIDGE data, we used DCC-train to train the models, and then tuned the model parameters using the DCC-dev and COSEE sets. Note that in both cases, the final testing data was never used to tune the models.

Table 2 shows how each of these models performed on our data. Surprisingly, both machine learning models performed only at the level of the Majority Class classifier, simply identifying all new resources as *within-scope.* This poor performance of even a state-of-the-art classifier shows that the digital library scope classification problem is much more challenging than some of the traditional text classification problems. A couple likely culprits for this poor performance are: the massive bias towards *within-scope* in the training data, and the complexity of using hundreds of thousands of words as unique features. The following sections explore techniques for addressing these problems.

Model	COSEE	BRIDGE
Majority Class	35.9	30.9
Multinomials	56.4	64.5
TF-IDF SVM	61.5	67.7
Oracle Majority	64.1	69.1

Table 3: Accuracy of systems when trained on only 51 *within-scope* and 51 *out-of-scope* resources

4.2 Unskewing Label Distributions

As discussed earlier, one of the things that makes working with digital library scope judgments difficult is that while there are many examples of resources that are within the scope of the library (namely, all the resources in the library itself), there are often very few resources that have been manually identified as being out of the scope of the library. For example, in our DCC data, over 98% of the resources we obtained were *within-scope*, and only the small remaining fraction (only 76 resources in total) were *out-of-scope.* Note however that this does not mean that in general 98% of resources contributed to DLESE were *within-scope.* Rather, this skewed ratio is a result of both unavailable records, and implicit pre-filtering by contributors who weeded out obviously *out-of-scope* resources before submitting to DCC. In particular, it doesn't match the *within-scope* proportion of larger contributed collections like COSEE or BRIDGE, where closer to 30% of resources were *within-scope.*

The large fraction of *within-scope* resources in DCC could be a problem for machine learning. Most machine learning algorithms are based on the assumption that the training data is a rough approximation of the future testing data. Yet, the fraction of *within-scope* resources in DCC is much larger than the fraction of *within-scope* resources in COSEE or BRIDGE. Because these differences violate the machine learning assumption, many machine learning algorithms will perform on these datasets as we saw earlier: simply defaulting to the clearly dominant majority class. Even sophisticated models like SVMs, which are designed to focus only on the most important examples (the support vectors), still run into difficulties with highly skewed datasets [20, 21].

There are a number of ways to address this problem. The simplest way of "unskewing" the data is to randomly remove *within-scope* resources until the number of *within-scope* resources equals the number of *out-of-scope* resources. In general, this is a bad idea, because it throws away the useful information contained in the removed resources. In the case of the DCC data, this means throwing away information for over 3000 resources. Still, if there is enough information left in the remaining resources, the models may be able to learn something. Table 3 shows the result of downsampling the training data to only 51 *within-scope* resources and 51 *out-of-scope* resources. This approach is surprisingly effective: performance of both machine learning models rises, with the SVM model achieving accuracies in the 60s for both the COSEE and BRIDGE data. Still, the model is unable to beat the performance of the oracle model that knows the majority class for the test data.

[4] We use the svm-perf implementation [15].

Model	COSEE	BRIDGE
Majority Class	35.9	30.9
TF-IDF SVM	64.1	63.8
Oracle Majority	64.1	69.1

Table 4: Accuracy of systems when optimizing for the area under the ROC curve.

Another approach to handling the massive skew in the data is to apply a more advanced machine learning algorithm. For example, the SVM-perf [15] formulation of support vector machines can optimize to a variety of different performance metrics including accuracy (the standard metric), the point at which precision and recall break even, and the area under the receiver operating characteristic (ROC) curve[5].

Optimizing to accuracy, as many machine learning algorithms do, is probably a bad idea for our data because the model can achieve over 98% accuracy on the training data by simply memorizing the majority class, and thus getting the *out-of-scope* resources correct contributes little to the overall accuracy. The other performance metrics can give a higher penalty for getting *out-of-scope* items wrong, and thus can encourage the learning algorithm to pay more attention to these examples.

We tried all the alternate performance metrics available through SVM-perf, and selected the metric that performed best on the development data. For both the COSEE and BRIDGE data, the area under the ROC curve outperformed the other metrics. Table 4 shows the accuracies of the models based on this performance metric. Performance rises into the 60s for both the COSEE and BRIDGE data but fails to outperform the oracle baseline, similar to what was seen when the number of *within-scope* and *out-of-scope* resources was balanced.

As we have seen, both adjusting the training data distribution and optimizing to different performance metrics can help address the problem of skew toward *within-scope* resources in our data. Therefore, in our remaining experiments, we try each of these techniques and select the one that performs best on our development data.

4.3 Topic-Based Features

While the techniques of the previous section were able to address the category skew problem present in our digital library scope task, they do not address the subtle topic distinctions that digital libraries must make. Recall from the introduction that DLESE must, for example, distinguish resources about how tsunamis are formed from resources about the effects of tsunamis on ships and watercraft. These subtle distinctions are much harder to make than the distinctions of standard text classification, e.g. differentiating economics from sports or sports from war.

In an attempt to provide a more semantic description of the text than the TF-IDF bag of words provides, we investigated building scope classifiers based on features derived from statistical topic models. In essence, we treat the topic models as a dimensionality reduction technique, reducing the hundreds of thousands of word features to just hundreds of topic features. To achieve this, we train statistical topic models on our digital library resources, asking for up to a few hundred topics. These models can estimate the distribution of topics for any given resource, e.g. a resource might be estimated to consist of 30% topic 1, 10% topic 2, 20% topic 3, etc. These few hundred percentages then replace the hundreds of thousands of TF-IDF scores that were previously used. Machine learning algorithms then look for patterns in these topic percentages in the same way that they look for patterns in the TF-IDF scores.

We consider two types of models for learning the topics, Non-negative Matrix Factorization (NMF) and Latent Dirichlet Allocation (LDA). Both schemes have been shown to have some correlations to intuitive human notions of topics, and so should be helpful for our digital library scope classification task. Their approaches to identifying topics are quite different however, so it would not be surprising to see that one serves better for our task than the other.

4.3.1 Non-negative Matrix Factorization (NMF)

NMF constructs a matrix where the rows are documents, the columns are words, and the entry in each cell is the number of times the given word occurred in the given document. NMF takes this term-document matrix and factors it into a term-topic matrix and a topic-document matrix, from which one can read off the topics associated with each word and the topics associated with each document. Formally, the NMF problem may be stated as:

$$V \simeq WH \tag{1}$$

where V is an $n \times d$ term-document matrix, W is an $n \times k$ term-topic matrix, and H is a $k \times d$ topic-document matrix. NMF requires that all elements of the W and H matrices are non-negative, and as a result the topics correspond roughly to intuitive human notions of topics [17]. NMF also has the nice property that the percentage of each topic observed in a document can be read directly off the H topic-document matrix.

NMF factorization is carried out by minimizing the divergence $F = D(V||WH)$ such that $W, H \geq 0$. Here, D measures the distance between the V matrix and the WH matrix and is defined as:

$$D(A||B) = \sum_{ij} A_{ij} log \frac{A_{ij}}{B_{ij}} - A_{ij} + B_{ij}$$

Following [18] we minimize F using an iterative scheme with the following multiplicative updates for W and H.

$$H_{jk} \leftarrow H_{jk} \frac{\sum_i W_{ij} V_{ik}/(WH)_{ik}}{\sum_p W_{pj}} \tag{2}$$

$$W_{ij} \leftarrow W_{ij} \frac{\sum_k H_{jk} V_{ik}/(WH)_{ik}}{\sum_l H_{jl}} \tag{3}$$

We start with V being the same TF-IDF matrix used in the previous experiments. To speed the convergence of the algorithm, we then run a *k-means* clustering on the V matrix, and initialize the W matrix with the cluster centroids. The NMF factorization is then allowed to run until it reaches convergence.

4.3.2 Latent Dirichlet Allocation (LDA)

LDA [5] is a hierarchical extension to the previously introduced mixture of multinomials model, where unlike the

[5] The ROC curve graphs true positives on the Y axis against false positives on the X axis.

mixture of multinomials model, documents may simultaneously belong to many different components (topics). LDA is usually presented as a generative model, as an imagined process that someone might go through when writing a text. This generative process looks something like:

1. Decide what kind of topics you want to write about.
2. Pick one of those topics.
3. Imagine words that might be used to discuss that topic.
4. Pick one of those words.
5. To generate the next word, go back to 2.

While this isn't a totally realistic description of the process of writing, it does at least get at the idea that the words in a document are usually topically coherent. More formally, the process above can be described as:

1. For each doc d select a topic distribution $\theta^d \sim Dir(\alpha)$
2. Select a topic $z \sim \theta^d$
3. For each topic select a word distribution $\phi^z \sim Dir(\beta)$
4. Select a word $w \sim \phi^z$

The goal of the LDA learning algorithm then is to maximize the likelihood of our documents, where the likelihood of an individual document d is $p(d|\alpha,\beta) = \prod_{i=1}^{N} p(w_i|\alpha,\beta)$. Marginalizing over the hidden variables z,θ and ϕ we get the following intractable integral.

$$p(w_i|\alpha,\beta) = \int\int\sum_z p(w_i|z,\phi)p(z|\theta)p(\theta|\alpha)p(\phi|\beta))d\theta d\phi \quad (4)$$

The integral can be approximated in a few different ways, but in this paper we use Gibbs sampling as it has been widely implemented and was available in the LingPipe toolkit [1].

Gibbs sampling starts by randomly assigning topics to all words in the corpus. Then the word-topic distributions and document-topic distributions are estimated using the following equations:

$$P(z_i|z_{i-}, w_i, d_i, w_{i-}, d_{i-}, \alpha, \beta) = \frac{\phi_{ij}\theta_{jd}}{\sum_{t=1}^{T}\phi_{it}\theta_{td}} \quad (5)$$

$$\phi_{ij} = \frac{C_{word_{ij}} + \beta}{\sum_{k=1}^{W} C_{word_{kj}} + W\beta} \quad \theta_{jd} = \frac{C_{doc_{dj}} + \alpha}{\sum_{k=1}^{T} C_{doc_{dk}} + T\alpha} \quad (6)$$

$C_{word_{ij}}$ is the number of times word i was assigned topic j, $C_{doc_{dj}}$ is the number of times topic j appears in document d, W is the total number of unique words in the corpus, and T is the number of topics requested. In essence, the equations above mean that we count the number of times that a word is assigned a topic and the number of times a topic appears in a document, and we use these numbers to estimate word-topic and document-topic probabilities. Once topics have been assigned and distributions have been calculated, Gibbs sampling repeats the process, this time selecting a new topic for each word by looking at the calculated probabilities. The process is repeated until the distributions become stable or a set number of iterations is reached.

In our experiments, we tried out a few different settings for the LDA parameters. In particular, we tried topic models with 50, 100, 200 and 400 topics, and trained them for a number of iterations between 100 and 1000[6].

Model	COSEE	BRIDGE
Majority Class	35.9	30.9
NMF SVM	64.1	60.8
LDA SVM	69.2	69.7
Oracle Majority	64.1	69.1

Table 5: Accuracy of systems based on topic models.

	within-scope	*out-of-scope*
within-scope (model)	172	166
out-of-scope (model)	64	356

Table 6: Accuracy of systems based on topic models.

4.3.3 Topic Model Results

Both the NMF topics and the LDA topics were used to train SVM models, using the development data to set the various parameters. The best NMF model used 50 topics trained until convergence, with the SVM trained on all of the data and optimizing against the zero/one loss function. The best LDA model used 50 topics trained for 100 iterations, with the SVM trained on all of the data and optimizing for the precision-recall break-even point.

Table 5 shows the performance of the these classifiers. The LDA-based model not only out-performs all of our previous models, but it also out-performs the oracle model which peeked at the test data label distribution. For both COSEE and BRIDGE, the LDA-based model achieves almost 70% accuracy, an impressive result on such a difficult task. The NMF-based model performs somewhat poorer, suggesting that, at least for digital library scope judgments, LDA topic models may be a better choice than NMF models.

As a simple error analysis, we looked the confusion table for the errors made by the LDA-based model, as shown in Table 6. This shows, for example, that if our current models were used as a filter for library curators, 68% ($\frac{356}{166+356}$) of *out-of-scope* resources would be automatically thrown away, at the cost of incorrectly throwing away 27% ($\frac{64}{172+64}$) of *within-scope* resources.

5. DISCUSSION

The models developed here are the first steps towards tools that can help curators of digital libraries maintain useful, high quality, subject-themed collections. Given a digital library, our models can automatically distinguish resources that are *within-scope* from resources that are *out-of-scope* with accuracies near 70%. Our models achieve a substantial improvement over a state-of-the-art TF-IDF SVM model, which is unable to identify even a single *out-of-scope* resource.

Our models took advantage of two key insights into the digital library scope assessment problem. First, due to the small number of *out-of-scope* resources and the large number of *within-scope* resources, most machine learning algorithms failed to pay attention to the *out-of-scope* items. We found two ways of addressing this problem: downsampling

[6] We did not experiment with the α and β parameters, leaving them at their LingPipe defaults of 0.1 and 0.01, respectively.

Model	COSEE	BRIDGE
Majority Class	35.9	30.9
Multinomials	35.9	30.9
TF-IDF SVM	35.9	30.9
Multinomials (downsampled)	56.4	64.5
TF-IDF SVM (downsampled)	61.5	67.7
TF-IDF SVM (alt metric)	64.1	63.8
NMF SVM (alt metric)	64.1	60.8
LDA SVM (alt metric)	**69.2**	**69.7**
Oracle Majority	64.1	69.1

Table 7: Summary of all model accuracies.

the training data by throwing away *within-scope* resources, and using Support Vector Machine (SVM) algorithms that could optimize to alternate performance metrics. Both of these approaches performed similarly when used alone, but when combined with topic based features (which generally led to better performance) the alternate performance metrics were the clear winners.

The success of these alternate performance metrics is interesting because when we evaluate our models in the end, we only care about simple accuracy. Yet allowing the model to optimize to a different performance metric was still beneficial, letting the model better deal with differences in category distribution between the training data and the testing data. An interesting future extension to our work would be to more thoroughly explore this phenomenon and determine if some performance metrics are just generally more forgiving of category distribution changes than others.

Our other key insight that boosted the performance of the scope classifiers was that statistical topic models can provide a set of features more useful for making the subtle distinctions required by digital libraries. We found that SVM classifiers based on topics derived from Non-negative Matrix Factorization (NMF) performed much like SVMs based on TF-IDF features. Thus, even though NMF was condensing several hundred thousand features into less than a hundred, it was still maintaining most of the meaning. However, it wasn't really capturing anything more than the words themselves provided. Latent Dirichlet Allocation (LDA), on the other hand, seems to have done just that. We found that SVM classifiers based on topics derived from LDA performed better than SVM classifiers using TF-IDF features, and even performed better than an oracle baseline that cheated by looking at the label distribution in the test data. This suggests that the topics derived by LDA aren't just condensing down the massive number of word features, but they're condensing them down into the most important topics for characterizing the resources. This could be an important claim for LDA models, and deserves to be tested on a broader set of text classification problems.

LDA-based models are also probably more appropriate for online tools which need to respond in real-time. NMF does not provide a natural way to deal with previously unseen resources. For example, to produce NMF topics for our test data, we had to run the NMF factorization over the combined resources from DCC, COSEE and BRIDGE. This means that each time an NMF-based model wanted to classify a new resource as *within-scope* or *out-of-scope*, it would have to add the new resource to the dataset and then rerun the factorization over all the resources[7]. This can get quite time consuming - as it was, it took several days for each of our NMF factorizations to complete. On the other hand, LDA has a clear probabilistic framework to deal with new resources. The topic distribution of a new resource can be estimated by taking the current LDA model and running a few iterations of Gibbs sampling over the new resource by itself. (We found that as little as 10 iterations was sufficient in our experiments.) This estimates the topic distribution of the new resource without any need to modify the existing LDA model. This makes LDA a better model for real-time use, because there is no need for long retraining sessions between resources.

By combining alternate performance metrics that can handle the lack of *out-of-scope* resources with LDA-based features that can make the necessary subtle topic distinctions, we have created scope classification models that perform well at a very difficult task. While not yet perfect, we see our models as the first step towards automating some of the work of maintaining a coherent, subject-themed digital library. Even scope classifiers with imperfect performance can be useful for prioritizing work, e.g. moving the most relevant candidates to the front of the queue and the most problematic to the end, thereby increasing throughput by getting in-scope resources into the library sooner. We expect that even our current models, which achieve about 70% accuracy, could be used in this way. And while it is true that a new scope classifier would need to be trained for each new repository, we have shown that this can be done by just using the resources already present in the library and only a very small number of additional *out-of-scope* resources. In our case, we just used the 76 *out-of-scope* resources that were already available, but it is likely that by carefully selecting *out-of-scope* resources to illustrate the important distinctions the library must make, higher performance could be achieved with a very small number of resources.

5.1 Limitations

Though our models represent some significant first steps, work still remains to confirm the effectiveness of our approach. Perhaps most importantly, we have focused on DLESE because manually identified *out-of-scope* resources were readily available. But DLESE is restricted to Earth system topics, so additional work is still needed to verify that our algorithms perform as well in other subject areas, like biology or physics.

6. CONCLUSIONS

We have presented a novel approach for building computational models which, given a subject-themed digital library, can automatically classify new resources contributed to that library as either *within-scope* or *out-of-scope*. These models address two significant challenges: digital libraries often have only a small number of *out-of-scope* resources available, and digital library themes often require subtle topic distinctions. We trained support vector machine (SVM) classifiers, optimizing for alternate performance metrics and

[7] There are some alternatives to rerunning the factorization each time. Guillamet and Vitria [11] propose only partially re-running the NMF process while Buciu [6] suggests an approach for projecting the new resource onto the old term-topic matrix. However, these methods are *ad-hoc* and known to be suboptimal [31].

providing as features topics derived from Latent Dirichlet Allocation (LDA), and produced scope classification models that could distinguish between *within-scope* and *out-of-scope* resources with almost 70% accuracy. These models do not need large volumes of hand annotated data for training, and perform well enough to underpin the next generation of repository management tools and perhaps eventually tools for end-users as well.

We envision future scope classifiers to be developed as follows. First, all resources in the digital library are collected as examples of *within-scope* resources. Next, *out-of-scope* resources are gathered either from records if available, or by manually identifying resources that are close topically, but still not quite right for the library. Then, our LDA-SVM classifier is retrained on this new data, producing a model which can make the kind of scope judgments necessary. The main time spent would be in finding the small number of *out-of-scope* resources, and the return for this effort would be a substantial increase in efficiency as reviewers or curators focus on the most promising resources first. As the performance of our model improves, less and less manual review of the scope judgments will be required. Digital libraries based on such automated tools will be substantially more scalable, and will cope better with the massive increases in the amount of information available on the web.

7. ACKNOWLEDGMENTS

This material is based in part upon work supported by the National Science Foundation under Grant Number CISE/IIS-0741326. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

8. REFERENCES

[1] Alias-i. LingPipe 3.7.0. http://alias-i.com/lingpipe/, Oct. 2008.

[2] R. B. Allen and Y. Wu. Metrics for the scope of a collection. *Journal of the American Society for Information Science and Technology*, 56(12):1243–1249, 2005.

[3] BEN. BiosciEdNet. http://www.biosciednet.org/, 2009.

[4] D. Blei and J. McAuliffe. Supervised topic models. In J. Platt, D. Koller, Y. Singer, and S. Roweis, editors, *Advances in Neural Information Processing Systems 20*, pages 121–128. MIT Press, Cambridge, MA, 2008.

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

[6] I. Buciu. Learning sparse non-negative features for object recognition. In *Intelligent Computer Communication and Processing, 2007 IEEE International Conference on*, pages 73–79, 2007.

[7] comPADRE. Resources for physics and astronomy education. http://www.compadre.org/, 2009.

[8] DLESE. Digital library for earth system education. http://www.dlese.org/, 2009.

[9] H. Drucker, D. Wu, and V. Vapnik. Support vector machines for spam categorization. *Neural Networks, IEEE Transactions on*, 10(5):1048–1054, 1999.

[10] P. Ginsparg. Winners and losers in the global research village. In *Proceedings of the Joint ICSU Press/UNESCO Expert Conference on Electronic Publishing in Science*, 1996.

[11] D. Guillamet and J. Vitria. Evaluation of distance metrics for recognition based on non-negative matrix factorization. *Pattern Recogn. Lett.*, 24(9-10):1599–1605, 2003.

[12] D. Hiom. The social science information gateway: putting theory into practice. *Information Research*, 4(1), 1998.

[13] L. Huxley. *Follow the Fox to Renardus: An Academic Subject Gateway Service for Europe*, pages 157–171. Lecture Notes in Computer Science. Springer, Berlin / Heidelberg, 2000.

[14] T. Joachims. *Learning to Classify Text Using Support Vector Machines: Methods, Theory and Algorithms.* Kluwer Academic Publishers, 2002.

[15] T. Joachims. A support vector method for multivariate performance measures. In *Proceedings of the 22nd international conference on Machine learning*, pages 377–384, Bonn, Germany, 2005. ACM.

[16] A. Kolcz and W. Yih. Raising the baseline for high-precision text classifiers. In *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 400–409, San Jose, California, USA, 2007. ACM.

[17] D. D. Lee and H. S. Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401(6755):788–791, Oct. 1999.

[18] D. D. Lee and H. S. Seung. Algorithms for non-negative matrix factorization. In *NIPS*, pages 556–562, 2000.

[19] D. D. Lewis, Y. Yang, T. G. Rose, and F. Li. RCV1: a new benchmark collection for text categorization research. *J. Mach. Learn. Res.*, 5:361–397, 2004.

[20] Y. LI, K. BONTCHEVA, and H. CUNNINGHAM. Adapting SVM for data sparseness and imbalance: A case study in information extraction. *Natural Language Engineering*, 15(02):241–271, 2009.

[21] Y. Li and J. Shawe-Taylor. The SVM with uneven margins and chinese document categorization. COLIPS PUBLICATIONS, 2003.

[22] D. Mimno and A. McCallum. Organizing the OCA: learning faceted subjects from a library of digital books. In *Proceedings of the 7th ACM/IEEE-CS joint conference on Digital libraries*, pages 376–385, Vancouver, BC, Canada, 2007. ACM.

[23] A. Moschitti and R. Basili. *Complex Linguistic Features for Text Classification: A Comprehensive Study*, pages 181–196. Springer Berlin / Heidelberg, 2004.

[24] D. Newman, K. Hagedorn, C. Chemudugunta, and P. Smyth. Subject metadata enrichment using statistical topic models. In *Proceedings of the 7th ACM/IEEE-CS joint conference on Digital libraries*, pages 366–375, Vancouver, BC, Canada, 2007. ACM.

[25] K. P. Nigam. *Using unlabeled data to improve text classification.* PhD thesis, Carnegie Mellon University, 2001. Chair-Tom M. Mitchell.

[26] NSDL. National science digital library. http://nsdl.org/, 2009.

[27] OAIster. Open archives initiative (OAI)ster. http://www.oaister.org/, 2009.

[28] OCA. Open content alliance. http://www.opencontentalliance.org/, 2009.

[29] F. Sebastiani. Machine learning in automated text categorization. *ACM Comput. Surv.*, 34(1):1–47, 2002.

[30] P. Soucy. Beyond TFIDF weighting for text categorization in the vector space model. *In Proceedings of the Proceedings of the 19th International Joint Conference on Artificial Intelligence (IJCAI 2005)*, pages 1130—1135, 2005.

[31] D. Soukup and I. Bajla. Robust object recognition under partial occlusions using NMF. *Computational Intelligence and Neuroscience*, 2008:857453, 2008. PMC2396239.

[32] M. Steyvers and T. Griffiths. Probabilistic topic models. In T. Landauer, Mc, S. Dennis, and W. Kintsch, editors, *Latent Semantic Analysis: A Road to Meaning*. Lawrence Earlbaum, 2007.

[33] S. Veeramachaneni, D. Sona, and P. Avesani. Hierarchical dirichlet model for document classification. In *ICML '05: Proceedings of the 22nd international conference on Machine learning*, pages 928–935, New York, NY, USA, 2005. ACM.

[34] O. Yilmazel, N. Balasubramanian, S. C. Harwell, J. Bailey, A. R. Diekema, and E. D. Liddy. Text categorization for aligning educational standards. In *Proceedings of the 40th Annual Hawaii International Conference on System Sciences*, page 73. IEEE Computer Society, 2007.

[35] S. Zhou, K. Li, and Y. Liu. *Text Categorization Based on Topic Model*, pages 572–579. Springer Berlin / Heidelberg, 2008.