Conceptual Search And Text Categorization

Lev Ratinov
Department of Computer Science
University of Illinois
Urbana, IL, USA
ratinov2@uiuc.edu

Dan Roth
Department of Computer Science
University of Illinois
Urbana, IL, USA
danr@uiuc.edu

Vivek Srikumar
Department of Computer Science
University of Illinois
Urbana, IL, USA
vsrikum2@uiuc.edu

ABSTRACT

The most fundamental problem in information retrieval is that of interpreting information needs of users, typically expressed in a short query. Using the surface level representation of the query is especially unsatisfactory when the information needs are *topic specific* such as "US politics" or "Space Science", that seem to require understanding of what the query *mean* rather than what it is.

We suggest that a newly proposed semantic representation of words [4] can be used to support *Conceptual Search*. Namely, it allows retrieving documents on a given *topic* even when existing keyword-based search approaches fail. The method we develop allows us to categorize and retrieve documents topically on-the-fly, without looking at the data collection ahead of time, without knowing a-priori the topics of interest and without training topic categorization classifiers.

We compare our approach experimentally to state-of-the-art IR techniques and to machine learning based text categorization techniques and demonstrate significant improvement in performance. Moreover, as we show, our method is intrinsically adaptable to new text collections and domains.

Categories and Subject Descriptors

H.3.3 [**Information Systems**]: Information Search and Retrieval; I.2.6 [**Artificial Intelligence**]: Learning

General Terms

Experimentation, Concept search

Keywords

Information retrieval, Semantics,Feature Representation

1. INTRODUCTION

Information retrieval (IR) focuses on connecting queries to documents. A well formed query can convey a lot of information about the subject of interest in a few words. Hence, research in this field has traditionally focused on optimizing retrieval systems towards keyword queries. The implicit assumption has been that the user of the system is aware of the keywords in the domain that she is interested in. However, this may not always be possible. For example, consider the task of searching through one's old emails for some information. It is often the case that one does not remember the exact words in the email and hence, cannot find what one is looking for. Moreover, searching by specifying parameters like 'sender' may not be discriminative. Frequently, people are more interested in getting topic-specific information than retrieving specific documents. In other words, people might want to know about concepts, even though they might not be able to generate accurate keywords. We call this kind of search a *conceptual search*.

Conceptual searching can be seen as a text categorization problem, where the query defines the category of interest. Traditionally, text categorization has been studied as classification, which requires training of a classifier using labeled data. The use of annotated data necessitates that the categories are pre-defined. The need for annotated data and pre-defined categories prevented ideas from ideas from text categorization machine learning from being used for information retrieval.

On the other hand, humans can perform text categorization without seeing even one training example. For example, consider the task of deciding whether a Usenet post must be posted to *comp.sys.mac.os* or *sci.electronics*. Humans do this without any explicit training because we know the meaning of the labels. We also know the type of messages we might encounter if the label is posed to us as a search query. In other words, humans can perform text categorization *and* conceptual search because we know the semantics of the label/query. Traditionally, however, text categorization systems ignore the semantics of the label and treat them as atomic identifiers. However, training a classifier that performs the mapping from the document to these identifiers requires labeled data.

In this paper, we develop techniques for conceptual search and extend them to document categorization. There is a natural relationship between search and document categorization because queries are analogous to labels. We propose to solve both these problems without previously observed or any labeled data. This means that we can perform *on the fly* conceptual search and text categorization for previously unseen labels. Our approach is based on the use of an encyclopedic source (Wikipedia) to analyze both queries and documents from a semantic point of view. This analysis allows us to access the *concepts* contained in the document and the query. The semantic analysis used in this paper is described in Section 2.

We describe the datasets used for our experiments in Section 3. Performing retrieval based on intersection of concepts allows us to perform contextual search. Our experiments in information retrieval are described in Section 4,

WWW2008, April 21–25, 2008, Beijing, China.
.

where we show that conceptual search outperforms keyword based search. To evaluate the strength of our techniques, we study it in the context of text categorization and compare to standard methods that use training on pre-annotated data. To perform text categorization, we build classifiers for the label in the concept space and use machine learning ideas to improve the performance of our classifiers. Section 5 details our experiments and results for text categorization. With no labeled data at all, our approach can create a text categorization system because we focus on the meaning of the labels. For example, we can identify documents pertaining to *American Politics* because the system 'knows' what the term means. Since our classifier was not trained on any particular data set, one expects it to work well across different data sets. In this section, we also show that our method adapts classifiers from one domain in another. Section 6 offers a discussion about various features of our semantic analysis and its applications and concluding remarks are made in Section 7.

2. QUERY AND DOCUMENT SEMANTICS

Descriptive labels and search queries have semantic content which often goes beyond the words they contain. For example, though the phrase *'American politics'* can be treated as just the two words in it, it could connote discussion about a wide range of topics – Democrats, Republicans, abortion, taxes, homosexuality, guns, etc. This indicates that there are two representations for any query – the query might be treated as the set of words that it contains, or it might be treated as the set of concepts that it symbolizes. In fact, this duality of representation applies not only to queries, but also to documents. A document can be thought of as either a set of words or the set of concepts that it talks about.[1]

Exploiting the information in a short query has been the focus of the information retrieval community since its inception. The label is usually represented as a point in some high-dimensional feature space. We describe two different feature representations that we have used. One approach is to treat it as a vector in the space of words. In further discussion, we refer to this feature representation as the *bag of words (BOW)* representation.

The semantics of a text snippet could also be derived using an encyclopedic source of information. The idea of *Explicit Semantic Analysis* (ESA) (cf. [4]) proposes to use Wikipedia for this task. If each article in Wikipedia can be considered to be a concept, then Wikipedia represents a collection of naturally defined concepts. Text snippets can be represented as weighted vectors in this high dimensional space([4]). For set of words, we can construct a list of Wikipedia concepts in which they appear. This defines the Explicit Semantic Analysis (ESA) of the words. In other words, the ESA of a set of words is the list of Wikipedia articles that discuss them. This list is ranked by the similarity of the words with the text of the corresponding article.

For example, consider the query *Science Electronics*. The top ten ESA concepts for this query is the following –

```
Institute of Electrical and Electronics Engineers
Materials science
Electrical engineering
Transmission electron microscopy
Electrochemistry
Scientific journal
Computer software
Electronics
Computer
Computational chemistry
```

The concepts are ranked by the TF-IDF score of the query in the Wikipedia article about that concept. Note that these concepts are not synonyms of the query. This means that ESA provides us with a tool to measure *relatedness* between queries and documents in terms of concepts that discuss them.

Since both the Bag of Words and ESA representations can be seen as vectors in a high dimensional space, we can define closeness between two vectors as the distance between them. This means that in both representations, we can use the simple k-Nearest Neighbors algorithm (Algorithm 1) to identify most related documents.

Algorithm 1 NN-R: *Find k-Nearest Neighbors of some text t, represented in some vector space as $R(t)$*

1: **for all** Documents d with vector representation $R(d)$ **do**
2: Distance(d) = Distance between $R(d)$ and $R(t)$
3: **end for**
4: Return k documents that have the shortest distance from t

3. DATASETS

To implement our ideas, we needed datasets that have labels with rich semantic information. Unfortunately, most datasets do not provide descriptive label names, since the label names are not expected to be used. Fortunately, there are several applications on the Web, where label names are critical. The datasets used both for search and for text categorization are presented in this section.[2]

3.1 20 Newsgroups Dataset

The 20 Newsgroups Dataset is a common benchmark used for testing classification algorithms. The dataset, introduced in [7], contains approximately 20,000 newsgroup posts, partitioned (nearly) evenly across 20 different newsgroups. Some of the newsgroups are very closely related to each other (e.g. *comp.sys.ibm.pc.hardware* and *comp.sys.mac.hardware*), while others are highly unrelated (for example, *misc.forsale* and *soc.religion.christian*). The 20 topics are organized into broader categories: computers, recreation, religion, science, forsale and politics. This dataset has been used for transductive learning ([10]), learning hierarchical classifiers ([8]) and other classification settings ([2]).

Since our approach focuses on the label names, the label names are essential for good performance. We cleaned the label names by expanding the newsgroup names that that were used in the original data. For example, we expanded *os* into *operating system* and *mac* to *macintosh apple*). We also removed some stop words such as *misc, alt* and *talk*. The amended label names given for each class are summarized in Table 1.

[1]This distinction between words and concepts is similar to that between *denotation* – the literal meaning of words, and *connotation* – their implied meaning.

[2]The Yahoo Answers dataset was collected for these tasks. The data will be made available online soon.

Newsgroup Name	Expanded Label
talk.politics.guns	politics guns
talk.politics.mideast	politics mideast
talk.politics.misc	politics
alt.atheism	atheism
soc.religion.christian	society religion christianity christian
talk.religion.misc	religion
comp.sys.ibm.pc.hardware	computer systems ibm pc hardware
comp.sys.mac.hardware	computer systems mac macintosh apple hardware
sci.electronics	science electronics
comp.graphics	computer graphics
comp.windows.x	computer windows x windowsx
comp.os.ms-windows.misc	computer os operating system microsoft windows
misc.forsale	for sale discount
rec.autos	cars
rec.motorcycles	motorcycles
rec.sport.baseball	baseball
rec.sport.hockey	hockey
sci.crypt	science cryptography
sci.med	science medicine
sci.space	science space

Table 1: *Newsgroup label names. We expanded the names of the newsgroups to full words and removed some words like misc. This table lists the expanded newsgroup names.*

3.2 Yahoo Answers

The second dataset that we used for our experiments is based on Yahoo Answers. We extracted 189,467 question and answer pairs from 20 top-level categories from the Yahoo Answers website (that is, about 10,000 question/answer pairs per category). These top-level categories have a total of 280 subcategories which refine the labels. For our experiments, we used the original subcategory names to as label names. Table 2 shows a sample of category and subcategory names.

Top-level Category	Subcategory
Arts And Humanities	Theater Acting
Business And Finance	Advertising Marketing
Business And Finance	Taxes
Computers And Internet	Security
Consumer Electronics	Play Station
Entertainment And Music	Jokes Riddles
Games And Recreation	Video Online Games
Sports	German Football Soccer
Sports	Rugby League

Table 2: *Sample Yahoo Answers categories and subcategories. In all, we collected 20 top level categories and 280 subcategories from Yahoo Answers.*

4. TRADITIONAL IR AND CONCEPTUAL SEARCH

Information Retrieval (IR) systems typically address the problem of identifying the most relevant documents given a user's queries. The underlying assumption in building such systems is that users can generate set of keywords that captures their information need accurately. However, formulating accurate keywords is not always possible for users. In such a situation, people describe their requirements with related words and browse till they find the information that they are looking for. In this section, we address this problem of *Conceptual Search* and compare it with a traditional IR system.

The ESA representation described in Section 2 provides a tool that can represent queries and documents as a collection of concepts. This way, we can compute semantic relatedness of documents and queries (cf. [4]). facilitating conceptual search for documents.

4.1 Methodology

Our experiments in this section were conducted with the Lemur Toolkit[3], which is a standard platform for conducting experiments in information retrieval. The toolkit has been used successfully for many information retrieval tasks, notably the TREC tasks. We used the TFIDF retrieval model for all our retrieval tasks. All the other parameters were set at their default values.

We used the Newsgroup and Yahoo Answers datasets in this setting. For the Newsgroup dataset, the Usenet posts were the documents that were to be retrieved and the names of the newsgroups were used as search queries (after processing them as detailed in Table 1). Similarly, for the Yahoo Answers dataset, the posts were the documents and the names of the categories were used as queries.

4.1.1 Baseline 1: Standard IR

As a baseline for comparison, for each dataset, we created an index of all the documents using Lemur's indexer. We used Lemur's retrieval engine to return a list of relevant documents using the queries described above. This represents the standard information retrieval setting.

4.1.2 Baseline 2: Query Expansion

For our second baseline, we expanded the query with the ESA representation of the query. The words of the ESA expansion were appended to the words of the query. Again, the documents were indexed with Lemur and Lemur's retrieval engine was used to perform retrieval using the expanded query.

4.1.3 Conceptual Search

In this setting, we used the ESA representation of both the queries and the documents. The words of the ESA representation of the documents were indexed and the ESA representation of the query was used for retrieval.

4.2 Results

Table 3 shows the results of experiments. We report the mean average precision of our retrieval experiments. Figures 1 and 2 show the precision-recall curves of the three methods on the Newsgroup and the Yahoo Answers datasets respectively. We used TREC's evaluation software to compute the metrics. We observed that the performance of the retrieval engine in the ESA space is significantly better for all the standard metrics.

[3]http://www.lemurproject.com

Experiment	Mean Average Precision	
	Newsgroups Dataset	Yahoo Answers Dataset
Traditional IR	0.1504	0.0660
Query Expansion	0.1741	0.0209
Conceptual Search	0.3681	0.1152

Table 3: *Results of information retrieval experiments:Searching in the ESA space is significantly better than using the words of the query. This is because the words need not appear in the messages that are posted in it.*

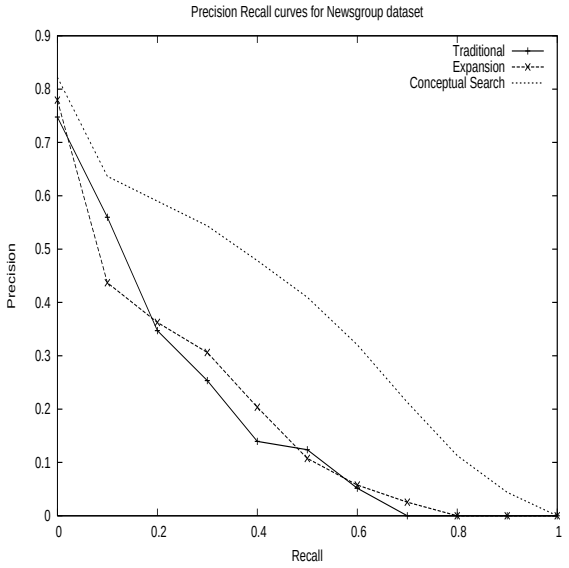

Figure 1: Precision vs. Recall plot for the Newsgroup dataset.

It can be seen using ESA improves the relevance of the documents that are retrieved over both the baselines. Expanding *only* the query with the ESA is not necessarily beneficial – for the Yahoo Answers dataset, the mean average precision falls. This behavior is often seen with query expansion, where recall is improved at the cost of precision. The improvement in the precision of the Newsgroup dataset indicates that it is an 'easier' collection of documents in terms of retrieval.

One reason for the better performance of the conceptual search is that the keywords in the query do not necessarily appear in the messages that are posted to a forum. For example, in the Newsgroup domain, a post to *sci.electronics* need not contain either *science* or *electronics*. These documents will not be retrieved by simple keyword search. In addition, a search in the ESA representation is effectively a search for documents that discuss related concepts. For example, from the sample ESA of this query in Section 2, we see that some of the concepts related to this query are *Transmission electron microscopy* and *Scientific journal*. One can expect that the ESA representation of the relevant documents will also contain these concepts. As a result, these documents will be retrieved when we search in the ESA space.

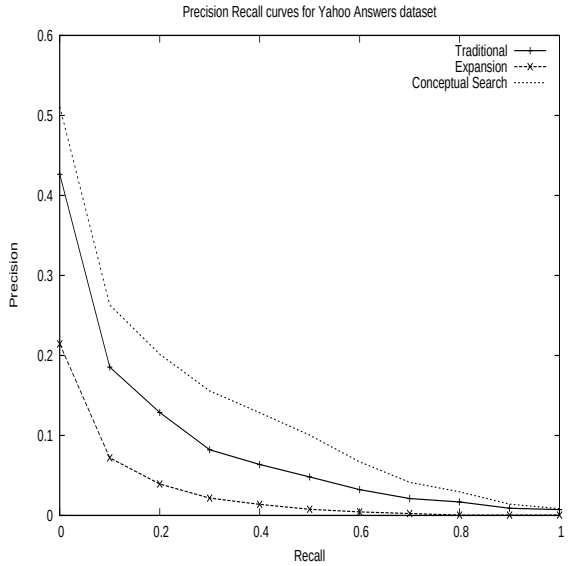

Figure 2: Precision vs. Recall plot for the Yahoo Answers dataset.

4.3 Expressivity of ESA

The representation of documents and queries as a set of concepts outperforms simple keyword search. To compare the expressively of the ESA representation with keywords, we computed the top 20 TF-IDF words of each category in the Newsgroup dataset. In some sense, these represent the most expressive keywords of each category. From these sets, we drew random subsets of different lengths and used them as our queries for retrieval. We compared the performance of the retrieval with the that of their ESA representation. Figure 3 shows how the precision with different keyword lengths. The ESA representation is more useful when the user queries are short because longer queries are often more specific. This means that the expressive power of ESA is comparable to that of the most expressive query.

5. DATALESS CATEGORIZATION

5.1 On-the-fly Categorization

From the previous discussion on conceptual search, it follows that conceptual search can be seen as a categorization problem of assigning topics to documents. In the previous sections, we considered the problem of deciding whether a document is relevant to the given query or not. In this section, the problem is the following – given multiple queries, decide the one to which the document belongs. This is essentially a classification problem, which has been studied extensively in machine learning from multiple perspectives. For example, the work in [11] approaches the problem of web page classification as a one-class classification problem, assuming that only positive examples are available.

Unlike the traditional categorization approaches, we do not use any labeled data. Given the analogy between queries and labels, we demonstrate the expressive capacity of the representations discussed earlier. We present several algorithms that use only the information present within the label to classify the data.

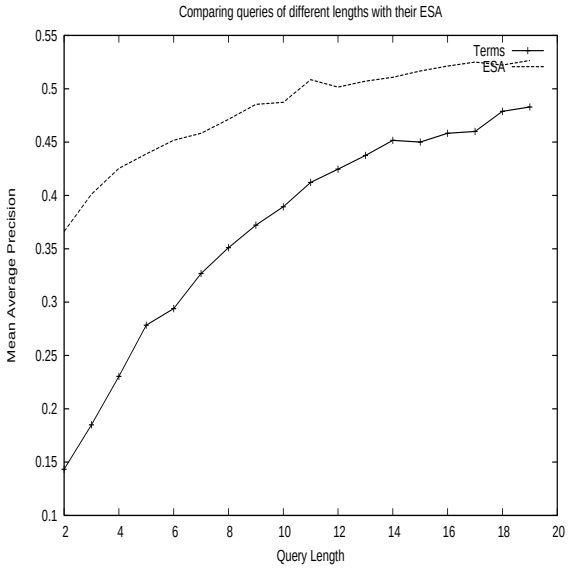

Figure 3: Effect of query length on average precision

The simplest approach is to perform Nearest Neighbor (NN) classification using the words of the label. In further discussion, we call this approach the **NN-BOW** algorithm. The pseudo code for this is given by Algorithm 1, where the Bag of Words representation is used for the documents and for the labels. It is evident that the NN-BOW algorithm can classify a document correctly, only if the document contains a word that appeared in one of the label names. Therefore, the recall of the NN-BOW classifier is very limited. In our experiments, only around 20% of the documents contain the words appearing in the labels.

Using the ESA representation, we define the **NN-ESA** classifier. This is similar to the earlier case, except that we use the ESA representation for labels and documents. This allows correct categorization of documents even if they share no common words with the label name. Since the ESA representation maps a text snippet into high-dimensional semantic space, two documents may be very close in the ESA space even if they share no common terms.

Our experiments show that, while the NN-BOW classifier performs poorly, the NN-ESA classifier often gives satisfactory performance without any training. This results in an on-the-fly classifier that can use dynamic, ad-hoc labels.

5.2 Modeling unlabeled data

5.2.1 Bootstrapping

The NN-BOW and the NN-ESA classifiers can classify new data based only on the semantic information available in the label name. In many cases, however, we are interested in retrieving a document that belong to a particular category in a specific collection of documents. This allows us to take advantage of the previous work in semi-supervised classification (Refer, for example, [9]).

One straightforward way to do this is to bootstrap the learning process using only the label names as examples. Algorithm 2 presents the pseudo code for a learning a bootstrapped semi-supervised classifier using a feature representation R. Note that though we do perform training, we do not use any explicitly labeled data and use only the label names as the starting point in the training. (Steps 1 to 4 indicate this.)

Algorithm 2 Bootstrap-R. *Training a bootstrapped classifier for a feature representation R, where R could be Bag of Words or ESA.*

1: Let training set $T = \emptyset$
2: **for all** l_i^R, the feature representation of label l_i **do**
3: $T = T \cup \{< l_i^R, i >\}$
4: **end for**
5: **repeat**
6: Train a Naive Bayes classifier NB on T
7: **for all** d_i, a document in the document collection **do**
8: If $NB.classify(d_i^R)$ with confidence above θ
9: $T = T \cup \{< d_i^R, NB.classify(d_i^{BOW}) >\}$
10: **end for**
11: **until** No new training documents are added.
12: **for all** d_i, document in the document collection **do**
13: Label d_i with $NB.classify(d_i^R)$
14: **end for**

Using the two feature representations – BOW and ESA – with Algorithm 2 gives us two more algorithms, called **Boot-BOW** and **Boot-ESA** respectively.

5.2.2 Co-training

The classifiers Boot-BOW and Boot-ESA are learned by bootstrapping on the bag of words and the ESA representations of the data respectively. The fact that BOW and ESA are parallel representation of the same data is ignored. Prior work ([1]) has studied the scenario when two independent feature representations (or views, as they are called in [1]) $\phi_1(d)$ and $\phi_2(d)$ are available for the same data and that if each feature representation is sufficient for correct classification of the data. In such a case, we can train two classifiers $c_{\{\phi_1\}}$ and $c_{\{\phi_2\}}$ that classify data better than chance. These two classifiers can train one another in a procedure called Co-training. We can apply this idea for our task. The algorithm for co-training is summarized in Algorithm 3. While the ESA representation is a function of the BOW representation, violating the 'view independence' assumption, we show that in practice, this procedure leads to a satisfactory performance. [4]

5.3 Experimental Methodology

In [10], 20 newsgroups dataset is used to construct 10 binary classification problems. These problems, along with the baseline used in that work for comparing their results, are shown in Table 4. It is to be noted that these results were generated with fully supervised training.

In our work, we are interested in 'on-the-fly' categorization, where the user may at first be interested in discriminating between 'sports' and 'health', but later may be interested in finer granularity categorization, for example: 'sports. We report the average performance on all the problems, and

[4]We also experimented with concatenating the BOW and the ESA representations into a single BOW+ESA view as was done in [3] for supervised classification. Then we bootstrapped the Naive Bayes classifier on the BOW+ESA view, but it consistently performed worse than co-training.

Algorithm 3 Co-training *We use the fact that BOW and ESA can independently classify the data quite well.*

Let training set $T^{BOW} = \emptyset$,$T^{ESA} = \emptyset$.
for all Labels l_i **do**
  $T^{BOW} = T^{BOW} \cup \{< i_i^{BOW}, i >\}$
  $T^{ESA} = T^{ESA} \cup \{< l_i^{ESA}, i >\}$
end for
repeat
  Train a Naive Bayes classifier NB^{BOW} on T^{BOW}.
  Train a Naive Bayes classifier NB^{ESA} on T^{ESA}.
  for all d_i, a document in the document collection **do**
    if $(NB^{BOW}.classify(d_i^{BOW})$ with confidence $> \theta)$ and $(NB^{ESA}.classify(d_i^{ESA})$ with confidence $> \theta)$ and $(y^{BOW} = y^{ESA})$ **then**
      $T^{BOW} = T^{BOW} \cup \{< d_i^{BOW}, y^{BOW} >\}$
      $T^{ESA} = T^{ESA} \cup \{< d_i^{ESA}), y^{ESA} >\}$
    end if
    $y^{BOW} = NB^{BOW}.classify(d_i^{BOW})$
    $y^{ESA} = NB^{ESA}.classify(d_i^{ESA})$
  end for
until No new training documents are added
for all d_i, document in the document collection **do**
  Label d_i with $NB^{ESA}.classify(d_i^{ESA}))$
end for

Id	Problem Description	Accuracy ([10])
1	Motorcycles Vs Ms-Windows	82.86
2	Baseball Vs Politics.misc	77.21
3	Religion Vs Politics.guns	65.48
4	Atheism Vs Autos	79.89
5	IBM hardware Vs Forsale	69.82
6	Politics.mideast Vs Sci.med	74.52
7	Christianity Vs Hockey	89.0
8	Space Vs Mac.Hardware	77.54
9	Windows.X Vs Electronics	68.13
10	Sci.Cryptography Vs Comp.graphics	79.73

Table 4: *The set of 10 binary classification problems used in Raina et. al. 2006 for the 20 newsgroups data, with the results reported in Raina et. al. 2006 for logistic regression classifier trained on 10 samples.*

then focus on specific cases which highlight the strengths and the weaknesses of our approach.

The problem sets we worked with are as follows – For the 20 newsgroups, we used the problems introduced by [10], which are shown in Table 4. For the Yahoo Answers dataset, we generated 20 random binary classification problems at *subcategory* level. Some of these the problems are shown in Table 5.

Intuitively, these are 'easy' classification problems for humans. Typically, with 10 training samples, [10] reported the error rates as high as 20% in 8 out of 10 problems, and even with 100 labeled samples, the error rate on the *religion vs. politics.guns* problem was above 20%. We achieve error rates below 11.5% on 9 out of 10 problems with *no* labeled data at all.

5.4 Results on binary categorization problems

The results of the binary categorization tasks for the Newsgroups and the Yahoo Answers datasets are summarized in Tables 6 and 7 respectively.

Id	Description
⋮	⋮
14	Health Diet Fitness Health Allergies
15	Business And Finance Small Business Consumer Electronics Other Electronics
16	Consumer Electronics DVRs Pets Rodents
17	Business And Finance India Business And Finance Financial Services
18	Sports Mexican Football Soccer Social Science Dream Interpretation
19	Sports Scottish Football Soccer Pets Horses
20	Health Injuries Sports Brazilian Football Soccer

Table 5: *Binary categorization problems for the Yahoo Answers dataset- subcategory level*

We have compared our Newsgroup results with two baseline results on the same dataset, both of which were reported in [10].[5] In both the baselines, 10 labeled documents were used for training the classifiers. The first one represents the traditional supervised scenario, where binary classifiers are built using labeled training data.

For the second baseline, in addition to the training data for the binary problem at hand, data from related domains was used to improve the performance. Table 6 5.4 shows that while the Nearest Neighbors algorithm with the bag of words representation does not perform as well as the baselines, using the ESA representation with the same algorithm outperforms the first supervised baseline for every problem. In fact, NN-ESA algorithm is slightly better than the second supervised baseline, which uses sophisticated machine learning ideas.

Using bootstrapping improves the performance of the classifiers and in most cases, co-training does even better. The key result is that we achieve an average accuracy of 90.96% *without* any labeled data.

For the Yahoo Answers dataset, we report the results of some of the categorization tasks in Table 7 5.4 and the average accuracy. As a baseline, we trained a supervised classifier on 10 documents, which achieved an average accuracy of 85.67%. The performance of our algorithms on this dataset is similar to the performance on the Newsgroups dataset. Again, even the "on the fly" Nearest Neighbors classifier in the ESA space beats the performance of the supervised classifier and the classifier that is trained with co-training is the best on an average.

5.5 Domain Adaptation

In this section, we compare the robustness of the BOW and the ESA representations in the context of text categorization across domains. The problem of discriminating two concepts across several domains is an important open problem in machine learning called *domain adaptation*, which recently has received a lot of attention [5, 6]. As our running example, we choose to focus on discriminating documents pertaining to *baseball* and *hockey*. The task is to perform

[5]We wish to thank Rajat Raina for providing the code and the results of his work.

Problem Id	Supervised Baseline 1[10]	Supervised Baseline 2[10]	NN-BOW	NN-ESA	Boot-BOW	Boot-ESA	Co-train
1	82.86	90.46	71.23	87.80	**97.89**	94.36	97.72
2	77.21	90.94	60.73	83.42	94.06	97.08	**96.23**
3	65.48	72.11	63.05	82.55	84.61	**89.81**	89.59
4	79.89	90.32	69.92	89.47	96.92	96.27	**97.95**
5	69.82	72.41	71.48	67.48	**86.12**	68.76	69.53
6	74.52	85.46	62.86	87.73	59.84	**95.50**	92.14
7	89.0	96.31	62.42	87.70	86.59	92.79	**96.43**
8	77.54	88.17	71.476	98.08	99.58	99.24	**99.66**
9	68.13	75.93	69.598	89.41	94.36	93.16	**96.84**
10	79.73	81.28	54.56	79.22	88.42	82.26	**90.96**
Average	76.41	84.33	65.73	85.29	88.84	90.92	**92.70**

Table 6: *Results of categorization algorithms on the binary problems of the Newsgroup dataset. The supervised baseline is for training set of 10 documents.*

Prob. Id	NN-BOW	NN-ESA	Boot-BOW	Boot-ESA	Co-train
14	63.38	94.98	96.37	**97.49**	96.74
15	64.54	78.96	58.08	**87.85**	70.9
16	56.83	98.93	99.69	99.69	**99.84**
17	58.92	39.28	**91.07**	37.5	89.28
18	93.81	92.14	98.23	99.01	**99.21**
19	73.67	94.92	93.96	**97.34**	96.37
20	50.63	92.34	49.78	97.02	**99.14**
Avg.	66.79	88.62	90.70	92.73	95.30

Table 7: *Results of the different categorization algorithms on the binary problems of the Yahoo Answers dataset. The average accuracy of a traditional supervised classifier for this dataset was 85.67%. In comparison, our best approach achieveld an accuracy of 95.3%*

well both on the 20 Newsgroups dataset and the Yahoo Answer dataset.

Traditional approaches to text categorization, which require observing data from the target collection beforehand and make use of data labeled with the appropriate category, suffer from the problem of domain adaptation. That is, if a classifier for a category of interest is trained in a given domain, it may not categorize documents well in a new domain. Informally, this is because the classifier has "learned" the vocabulary used to express the category's documents in a specific domain. In this paper we show that our method is intrinsically robust across domains. Our interpretation is that it categorize documents as belonging to a category based on they *meaning* rather than the surface representation.

In view of the approaches presented in this paper, the domain adaptation task becomes easy. When the documents for the new domain are represented with ESA vectors, this 'universal' representation, allows either immediate NN-ESA algorithm, or a seamless application of a classifier learned of one domain to another domain.

The primary focus of this work is conceptual search and categorization without data. However, it is often the case that a related, auxiliary labeled dataset is available. We stress that no labeled data for the primary categorization task is available. We use labeled data from an auxiliary domain to check whether it can be used to improve the performance on the primary categorization task.

To check our hypothesis, we first trained a traditional Naive Bayes *baseball vs. hockey* classifier using the BOW representation for the 20 Newsgroup (20NG) and the Yahoo Answers domain. Then we performed 5-fold cross validation within the same domain. The categorization accuracy was 0.97 for the Newsgroup domain , and 0.93 for the Yahoo domain. This indicates that the classifier can be learned successfully for the same domain. However, when we applied the classifier trained on Newsgroup data to Yahoo data , the accuracy dropped down to 0.89, and when we applied the classifier trained on Yahoo to the Newsgroup domain, the accuracy dropped down significantly to 0.60. This shows that the BOW classifiers are very sensitive to data distribution.

Next, we performed the same experiment on the ESA representation. The within-domain 5-fold cross validation performance was 0.96 for 20NG and 0.97 for Yahoo. When the Naive Bayes classifier trained on the ESA representation of Yahoo documents was applied to 20NG, the performance dropped only slightly to 0.90. The performance stayed the same when we applied the classifier trained on ESA representation of 20NG documents to Yahoo – the accuracy dropped down only to 0.96.

However, the most significant result is that when we applied the dataless approach, NN-ESA, presented in Section 5.1 (we used only the label name), the performance was 0.94 on the 20NG dataset and 0.96 on the Yahoo dataset. It is also worth stating that the performance of NN-BOW (see section5.1) was poor: 0.66 on the 20NG data and 0.65 on the Yahoo data.

These result are summarized below:
Train(BOW,20NG)→Test(BOW,20NG)=0.97
Train(BOW,Yahoo)→Test(BOW,20NG)=0.60
Train(BOW,Yahoo)→Test(BOW,Yahoo)=0.93
Train(BOW,20NG)→Test(BOW,Yahoo)=0.89
Train(ESA,20NG)→Test(ESA,20NG)=0.96
Train(ESA,Yahoo)→Test(ESA,20NG)=0.90
Train(ESA,Yahoo)→Test(ESA,Yahoo)=0.97
Train(ESA,20NG)→Test(ESA,Yahoo)=0.96
NN-ESA(20NG)= 0.94
NN-ESA(Yahoo)=0.96
NN-ESA(20NG)=0.66
NN-ESA(Yahoo)=0.65

These results demonstrate the good adaptation properties of the ESA-representation-based approaches in general, and the dataless NN-ESA approach presented in this paper, which uses the universal cross-domain semantic information

present in the label to classify data across domains.

6. DISCUSSION

The main theme of this work is that conceptual search and text categorization are two closely related tasks and both can be immensely helped by the use of encyclopedic knowledge. In Sections 4 and 5, we showed that ESA is a very expressive representation for describing documents and queries as a set of concepts. The most appealing aspect of ESA is that it is domain independent because of its dependence on Wikipedia. A natural question that raises is whether ESA can be used universally for all search problems. In this section, we discuss features and applicability of the ESA representation.

Firstly, in datasets considered in this paper, the documents we used dealt with a single subject. However, documents might discuss several unrelated topics. For example, news articles sometimes report unrelated events in the same documents – these documents often summarize different events over a period of time. Such documents are often found in the TREC datasets. For example, a document in one of the collections of the dataset describes different disasters that occurred in a particular year. This documents includes diverse concepts like names of places, natural disasters, man-made events, etc. Not surprisingly, the ESA representation of this documents is a combination of many irrelevant concepts, which presents a difficulties for retrieval algorithms.

Secondly, since Wikipedia is an encyclopedia discussing a very broad range of topics, queries that are very specific cannot be handled effectively. This suggests that the idea of conceptual search can complement keyword search instead of replacing it. In addition, in some circumstances, we could replace Wikipedia with a more specific source of world knowledge and define ESA using this source. Thus, we can create a topic specific conceptual search engine. Topic-specific conceptual search is useful if we are interested in searching for very specific information related to a particular area.

7. CONCLUSIONS AND FUTURE WORK

This paper demonstrates that classification and information retrieval are closely related and techniques from one can be used in the other. We show that classification tasks can benefit immensely by using the semantic content of the class labels.

Furthermore, we believe that the idea of creating *on-the-fly* classifiers without using any labeled data can be applied to information retrieval. Today's search engines are primarily keyword based and information retrieval techniques focus on improving the precision of document retrieval. A query specific classifier can be seen as an expert in identifying relevance of documents to the query. However, people rarely search for specific documents and, in fact, browse for concepts that they are interested in. In this setting, we show that the use of encyclopedic knowledge broadens the scope of search and returns *information* rather than just documents.

8. REFERENCES

[1] A. Blum and T. Mitchell. Combining labeled and unlabeled data with co-training. In *COLT: Proceedings of the Workshop on Computational Learning Theory, Morgan Kaufmann Publishers*, pages 92–100, 1998.

[2] E. Gabrilovich and S. Markovitch. Text categorization with many redundant features: using aggressive feature selection to make svms competitive with c4.5. In *ICML '04: Proceedings of the twenty-first international conference on Machine learning*, page 41, New York, NY, USA, 2004. ACM Press.

[3] E. Gabrilovich and S. Markovitch. Feature generation for text categorization using world knowledge. In *Proceedings of The 19th International Joint Conference on Artificial Intelligence*, Scotland, UK, 2005.

[4] E. Gabrilovich and S. Markovitch. Computing semantic relatedness using wikipedia-based explicit semantic analysis. In *Proceedings of The Twentieth International Joint Conference for Artificial Intelligence*, Hyderabad, India, 2007.

[5] H. D. III and D. Marcu. Domain adaptation for statistical classifiers. In *J. Artificial Intelligence, 26:101-126*, 2006.

[6] J. Jiang and C. Zhai. Instance weighting for domain adaptation in nlp. In *Proceedings of ACL*, 2007.

[7] K. Lang. NewsWeeder: learning to filter netnews. In *Proceedings of the 12th International Conference on Machine Learning*, pages 331–339. Morgan Kaufmann publishers Inc.: San Mateo, CA, USA, 1995.

[8] A. K. McCallum, R. Rosenfeld, T. M. Mitchell, and A. Y. Ng. Improving text classification by shrinkage in a hierarchy of classes. In J. W. Shavlik, editor, *Proceedings of ICML-98, 15th International Conference on Machine Learning*, pages 359–367, Madison, US, 1998. Morgan Kaufmann Publishers, San Francisco, US.

[9] K. Nigam, A. K. McCallum, S. Thrun, and T. M. Mitchell. Text classification from labeled and unlabeled documents using EM. *Machine Learning*, 39(2/3):103–134, 2000.

[10] R. Raina, A. Y. Ng, and D. Koller. Constructing informative priors using transfer learning. In *ICML '06: Proceedings of the 23rd international conference on Machine learning*, pages 713–720, New York, NY, USA, 2006. ACM Press.

[11] H. Yu, J. Han, and K. C.-C. Chang. Pebl: Web page classification without negative examples. *IEEE Transactions on Knowledge and Data Engineering*, 16(1):70–81, 2004.