Multilevel Measures of Document Similarity

Irina Matveeva
Department of Computer Science
University of Chicago
Chicago, IL 60637
matveeva@cs.uchicago.edu

Gina-Anne Levow
Department of Computer Science
University of Chicago
Chicago, IL 60637
levow@cs.uchicago.edu

Abstract

1 Introduction

Many applications such as document summarization, passage retrieval and question answering require a detailed analysis of semantic relations between terms within and across documents and sentences. Often one has a number of sentences or paragraphs and has to choose the candidate with the highest level of relevance for the topic or question. An additional requirement may be that the information content of the next candidate is different from the sentences that are already chosen.

Many approaches to information retrieval and document classification model the semantic similarity between documents using the relations between semantic classes of words. They include representing dimensions of the document vectors with distributional term clusters (**?**) and expanding the document and query vectors with synonyms and related terms as discussed in (**?**). Latent Semantic Analysis (LSA) (**?**) is one of the best known dimensionality reduction algorithms. It represents documents as vectors in the space of latent semantic concepts. Latent Dirichlet Allocation (LDA) (**?**) uses the latent semantic concepts as bottleneck variables in computing the term distributions for documents. The new representation captures overall semantic similarity between documents but is less sensitive to differences on the sentence level. Moreover, the methods include all vocabulary terms in their computations which limits their applicability.

Semantic similarity on the word level is targeted for word sense disambiguiation (WSD), e.g. Schütze (**?**), verb classification XXX(cite D. Lin). The research has shown that different measures of similarity may be required for different groups of terms such as nouns and verbs. It also reasonalbe to use different notions of similarity for content bearing general vocabulary words and named entities.

Methods of WSD are usually use co-occurrence statistics. Verb similarity measures is based on syntactic similarity.

In this project, we propose to use a combination of similarity measures between terms to model document similarity. We divide the vocabulary into general vocabulary terms and named entities and compute a separate similarity score for each of the group of terms. The overall similarity score is a function of these two scores. In addition, we use statistical co-occurrence as well as syntactic similarity to compute the similarity between the general vocabulary terms.

2 Approach

It is a difficult problem to define the notion of general vocabulary terms. In this paper, we adopt the following strategy to filtering the words that can be considered content bearing general vocabulary terms. First, we use a Named Entity (NE) recognition algorithm to identify the names of people, locations etc. These terms will not be considered general vocabulary terms. Part of the similarity score between documents will be the similarity based on the NE terms that they contain. We use a parser to identify nouns and adjectives that participate in three types of syntactic relations: subject, direct ob-

ject, the head of the noun phrase with an adjective or noun as a modifier for nouns and the modifier of a noun for adjectives. Currently, we consider this set of terms to be the content bearing words used to compute the similarity scores.

3 Co-occurence based Term Similarity

We use the Generalized Latent Semantic Analysis (GLSA) as a framework for computing semantically motivated term and document vectors. We begin with semantically motivated pair-wise term similarities and uses dimensionality reduction to compute a vector space representation for terms. Unlike other dimensionality reduction approaches that compute a dual term-document representation, we focus on term vectors because terms offer a much greater flexibility in exploring similarity relations than documents. The availability of large document collections such as the Web offers a great resource for statistical approaches. Recently, co-occurrence based measures of semantic similarity between terms have been shown to improve performance on such tasks as the synonymy test, taxonomy induction, and document clustering (**?**; **?**; **?**; **?**). Content bearing words, i.e. words which convey most semantic information, are often combined into semantic classes that correspond to particular activities or relations and contain synonyms and semantically related words. Therefore, it seems very natural to represent terms as low dimensional vectors in the space of semantic concepts.

3.1 GLSA Framework

We give a brief outline of the GLSA algorithm. We assume that we have a document collection C with vocabulary V. We also have a large, preferably Web based, corpus W.

1. Construct the weighted term document matrix D based on C

2. For the vocabulary words in V, obtain a matrix of pair-wise similarities, S, using the large corpus W

3. Obtain the matrix U^T of low dimensional vector space representation of terms that preserves the similarities in S, $U^T \in R^{k \times |V|}$

In this paper, we used a large document collection to obtain the matrix of semantic associations between all pairs of vocabulary terms using a number of well-established co-occurrence based methods, such as point-wise mutual information (PMI), likelihood ratio, χ^2 test etc. (**?**). In our experiments, we mainly use PMI because it has been successfully applied to such semantic proximity tests as the synonymy test (**?**; **?**) and taxonomy induction (**?**). It was also successfully used as a measure of term similarity to compute document clusters (**?**), and to extract semantic relations between verbs (**?**).

PMI between random variables representing two words, w_1 and w_2, is computed as

$$PMI(w_1, w_2) = \log \frac{P(W_1 = 1, W_2 = 1)}{P(W_1 = 1)P(W_2 = 1)}. \quad (1)$$

We used the singular value decomposition (SVD) and the Laplacian Eigenmaps Embedding algorithm to computed GLSA term vectors. SVD preserves all similarities in the term similarities matrix S. The Laplacian Eigenmaps Embedding algorithm preserves the similarities only locally, based on the neighborhood graph of the terms, since local information is often more reliable. For details on these algorithms, see (**?**) and (**?**).

4 Vocabulary Classes

5 Combined Similarity Measure

We generated a number of different document representations. One of them is the traditional bag-of-words representation, denoted as $TDT2 - all$. For this representation we used all 112,710 vocabulary words (after low-casing, stop words removal and stemming). The second representation was similar but excluded all words and expressions that were tagged as named entities. The vocabulary space is 55,729. The third representation also excluded all named entities and used only the nouns and adjectives from the set of the content bearing words as described above. Here the vocabulary size is 13,818. We also had a representation where the documents were indexed with named entities only, with the vocabulary of 129,550. The vocabulary space for this representation is larger than for the first one which contains all words. The reason is that in the first

All Words	no NE	no NE, content words	only NE
112,710	55,729	13,818	29,550

Table 1:

Topic	Topic Description	N Docs
20012	Pope visits Cuba	133
20013	1998 Winter Olympics	514
20015	Current Conflict with Iraq	1405
20039	India Parliamentary Elections	119
20044	National Tobacco Settlement	252
20070	India A Nuclear Power	444
20071	IsraeliPalestinian Talks (London)	203
20076	AntiSuharto Violence	326

Table 2:

representation we considered individual words as indexing elements. For the representation with the named entites, however, we considered phrases as a whole, so that "michael blassie" and "blassie" are two different vocabulary entries.

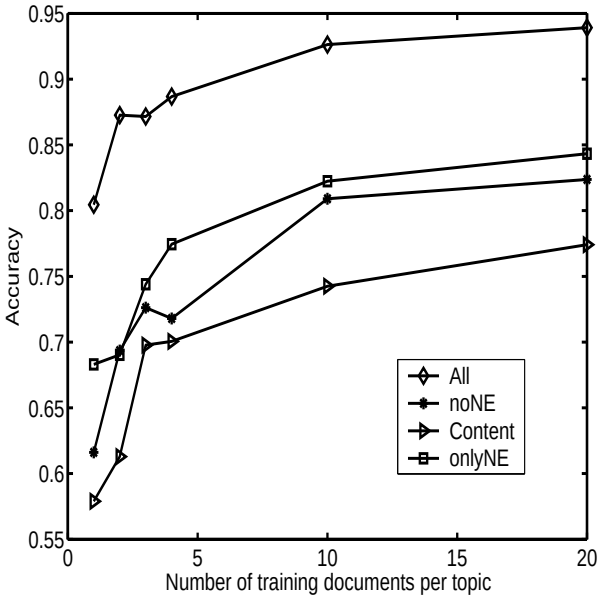

Figure 1:

6 Experiments

6.1 Document Collection

6.1.1 TDT2

We used the version TDT2 collection that included named entities tags. There are 100 topics that were used for the TDT2 collection. Many documents in the TDT2 collection are assigned to one or more topics. We used only the documents that are assigned to a single topic. We used only the English sources and removed documents that did not contain any named entities. We had 57,816 documents.

6.1.2 GigaWord English

6.2 Classification Experiments

To validate our approach we conducted document classification experiments.

We selected the top 10 topics and excluded the top 2, which gave us 8 topics. Table X shows the topics and the number of documents for each of them. It gave us 3396 documents that we used for classification.

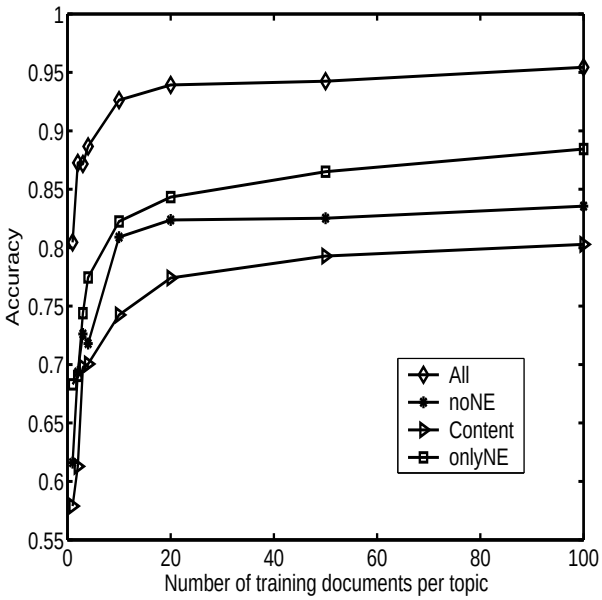

Figure 2:

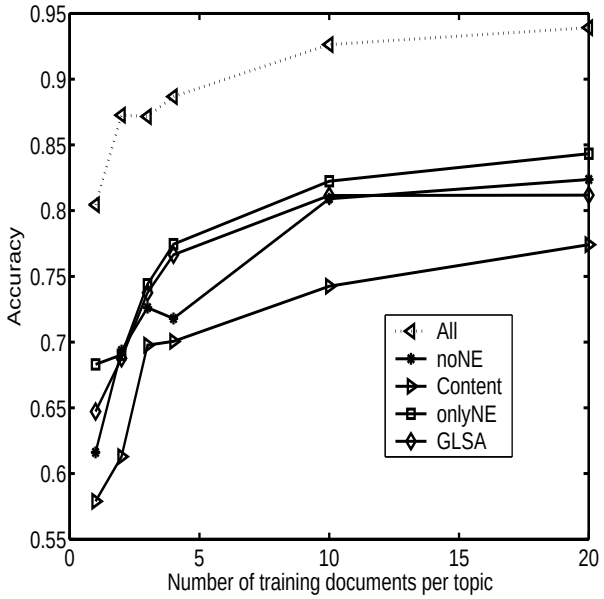

Figure 3:

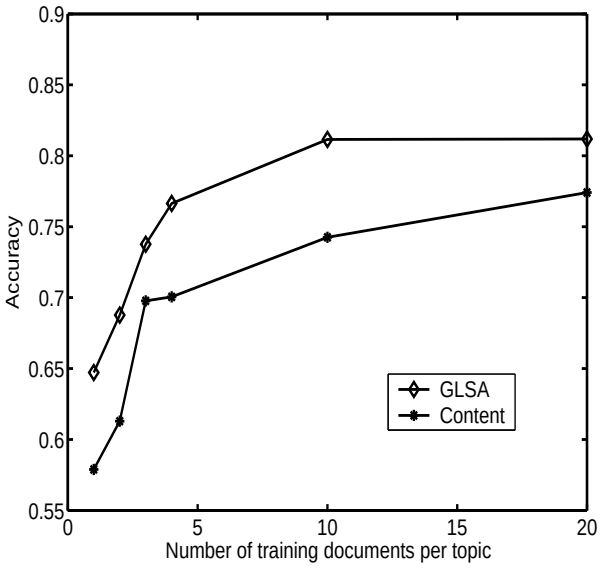

Figure 5:

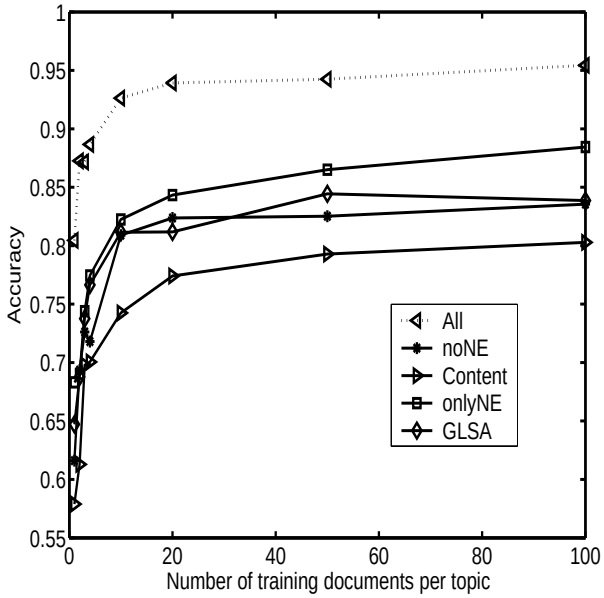

Figure 4:

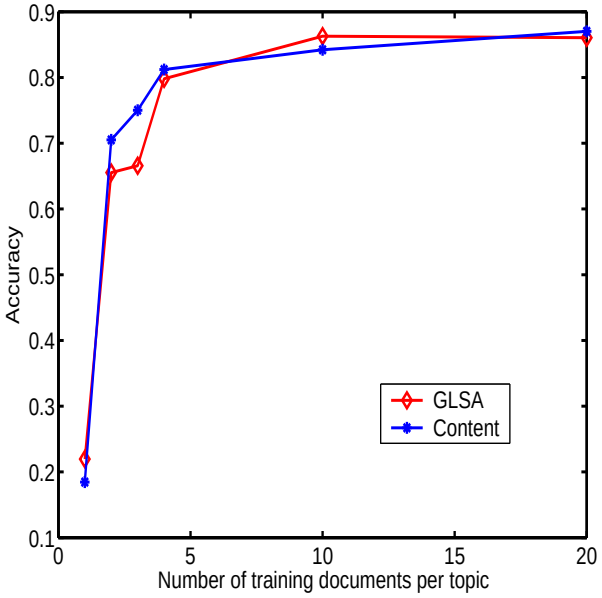

Figure 6: Combination of onlyNe and GLSA/Content representations.

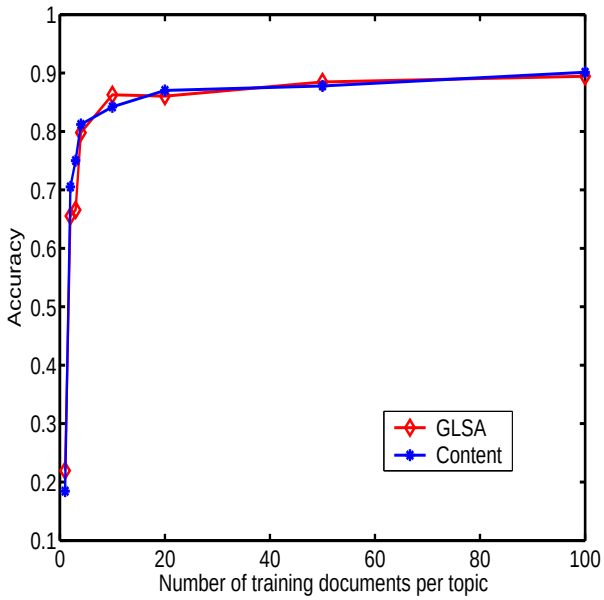

Figure 7: Combination of onlyNe and GLSA/Content representations.