

An Algorithm for Unsupervised Topic Discovery from Broadcast News Stories

Sreenivasa Sista
Department of Electrical and Computer Engineering,
Northeastern University
Boston, MA
ssista@ece.neu.edu

Richard Schwartz, Timothy R. Leek, and
John Makhoul
BBN Technologies,
Cambridge MA
schwartz,tleek,makhoul@bbn.com

ABSTRACT

The cost of annotating a large corpus with thousands of distinct topics is high. In addition, human annotators usually fail to indicate all of the relevant topics for each document. It would be desirable to determine the topics in any new domain or language automatically, given only a large corpus in that domain and language. We present an algorithm called Unsupervised Topic Discovery, which creates topics from a collection of news stories, provides a human understandable topic label and then assigns the topics to the news stories. Finally, we report our results on a collection of broadcast news stories in English and Arabic.

Keywords

Knowledge extraction, topic classification, thematic clustering, text summarization.

1. INTRODUCTION

Topics are subjects or themes that can be used to categorize a document (or a news story). Usually, a news story is about several topics. For example, a news story about the “Oklahoma City Bombing in 1995” can be said to contain topics “Bombings”, “Terrorism”, “Oklahoma”, “Deaths and Injuries”. The cost of human annotation of such topics for a large number of documents is very high. In this paper, we present an automatic text categorization method called “Unsupervised Topic Discovery” (UTD) that creates new topic categories.

Extracting themes from text is not a new problem. Word disambiguation was achieved by clustering words (noun and verb pairs) by themes [5][6]. The ITERATE algorithm [7] clusters documents into a hierarchical conceptual cluster tree based on inter-document similarity. Text excerpts from documents that are about a theme were found to be very useful [3] for generating document summaries. All the above approaches extracted thematic information from documents targeted at specific applications e.g., word disambiguation, document categorization and document summarization. In addition, we assert that topics cannot be arranged in a strict tree structure. For example, a group of documents about “illegal steroid usage in Olympics” can be represented in a hierarchical structure in two ways (a) cluster of documents about “drug-abuse”, where “drug-abuse” is a sub-category under the category “sports”, or (b) a cluster of documents about “sports”, where “sports” is a sub-category under the category “drug-abuse”.

The proposed method for UTD aims to create a flat list of topics, from a large collection of documents. Each discovered topic is automatically assigned an intelligible topic label and a set of support words, statistically associated with the topic. We employ OnTopic™[1] topic classification system, to generate statistical topic models. The resulting topic models can be used in applications like text categorization [1], document summarization and story segmentation [10].

In section 2, we define the term topic under the UTD framework. Section 3 describes the OnTopic topic classification model used for creating the topic support words. In Section 4, we describe the UTD system. Section 5 describes the evaluation methodology. Section 6 describes UTD experiments for broadcast news stories in English and Arabic.

2. TOPICS

There are many definitions for topics, so it’s worth defining what we mean. By “topics”, we mean subjects that can be used to categorize a document, much as used by Primary Source Media or by Reuters. Human annotators represent a topic by a single term (a phrase or a word) like “Inhalation Anthrax”. Often, the topic label (e.g., Inhalation Anthrax) does not appear, in its exact form, inside a document. The presence of the topic inside a document is judged by several words that are related to the main topic.

Thus, our definition of a topic has the following components:

1. A descriptive and human comprehensible label. ‘Birds,’ ‘aircraft accidents,’ ‘President Clinton,’ ‘White House,’ ‘dinosaurs,’ and ‘insurance industry’ are some topic labels.
2. A set of terms, known as support words, that supports the existence of the topic in a document or text. For example, terms “anthrax,” “anthrax spores,” “bacteria,” and “germ warfare agent” are some of the support terms for the topic “Inhalation Anthrax”.
3. A statistical measure of significance of the support term for a topic. Some support words carry more information about a topic than others. For example, the term “anthrax” gives a better indication of the presence of the topic “inhalation anthrax” than the term “bacteria”. Significance of a support word for a topic is represented by a statistical probability, $P(\text{support term} | \text{Topic})$.

3. TOPIC MODEL

There are many methods for determining the topics in a document. The OnTopic™ system at BBN uses a Hidden Markov Model (HMM) to model multiple topics in documents

explicitly [1]. The model is pictured in Figure 1. We assume (make believe) that the story is generated by this model. According to this model, when an author decides to write a story, the first thing he does is to pick a set of topics. The topics are chosen according to the prior distribution for topics. There is a HMM with one state for each of the chosen topics, plus an additional state for the General Language topic. Each of the states has a distribution for the language about that topic. In the simplest case, this language model is a unigram distribution on words. However, the state could also contain a higher order n-gram language model for word sequences for that topic. According to the model, the author writes the story one word at a time. Before choosing each word, he first chooses which of the topics that word will be about. This is chosen according to the probability of each topic, given the set of topics in the story. Once the topic is chosen, the author chooses a word from the corresponding topic state according to the distribution of words for the topic. Then the author must choose the topic for the next word, and so on until the story is finished. (If an n-gram model is used, the process is slightly more complicated, but the same principle can be used.)

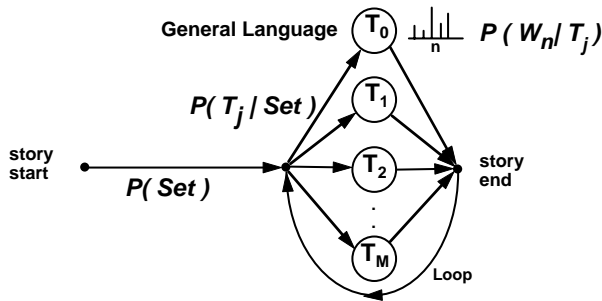


Figure 1: OnTopic topic classification model - HMM for how a story is generated from its topics.

Typically, we are not given the word distributions for the topics. We must determine them from a corpus of stories annotated with topics. We use a modified version of the EM method to determine the distributions. Given a new document, we could determine the most likely set of topics by considering all possible sets of topics exhaustively. Of course, this is too expensive. Instead, we first determine which individual topics are likely enough and then we consider all combinations of those few topics.

These topic models can be used to provide a very concise summary of a story, or as a common set of key terms to be used in searching for documents about a desired combination of subjects (without having to know what words are actually in the document).

4. SYSTEM DESIGN

In this section, we describe the process of Unsupervised Topic Discovery (UTD). UTD is a process by which a system or a machine gains knowledge of or ascertains the existence of previously unknown topics in a collection of documents. The input to the UTD system, typically, is a collection of documents and the output, are topics. The algorithm for finding the themes or topics in the corpus has two high-level steps. First (in section 4.1), we find descriptive phrases (using the MDL criterion) in the

corpus. Then, we augment each document with the phrases they contain and identify names of people, places, and organizations. Next, in section 4.2, we determine an initial set of topics, based on the key-phrases that occur in each document. These key-phrases are assigned as the initial topics for each document. We then use the Estimate-Maximize algorithm to determine the full set of words and phrases that are statistically associated with each of the topics.

4.1 Feature Selection

One of the goals of the topic discovery process is to create meaningful topic labels. We choose topic labels from the features of a document. In order to create intelligible topic labels, we first augment a document with names and phrases. Then, we choose words, names and phrases as potential names.

4.1.1 Proper Names

Proper nouns are usually key terms for any document or news story e.g., Bill Clinton. We use Identifinder [2], a name extraction system to locate names of persons, places, and organizations. We enhance the person name feature by replacing a partial name of a person with the most recent full name inside the same document. For example, we replace all the later occurrences of the name “Clinton” with the most recent occurrence of the full name “Bill Clinton”. We reason that the full name of a person is most often mentioned at the beginning of the document and the later references are only partial references.

4.1.2 MDL Phrases

A Phrase is often more descriptive than a single word. For example, the phrase “aircraft accidents” is more descriptive than the word “accidents”. We find those word sequences that have high mutual information, such that they decrease the description length (DL, the number of bits needed to code the corpus) for the corpus [4].

The description length (DL) of a corpus is defined by [11] a sum of the number of bits necessary to represent the coded terms in the corpus plus the number of bits required to represent the mapping between the original and the term for the code. In the UTD system, we represent each term by a 32-bit integer. We join two consecutive terms into a phrase (bigram) if the DL of the system decreases by more than some number of bits. The phrase creation process is iterative. First, we find pairs of words and create a new corpus replacing the individual words with newly created phrases. The process of searching for bigrams in the new corpus and creation of another version of corpus with new bigrams is repeated until the search for bigrams results in a null set. For example, in the first iteration, we create a bigram “RODHAM CLINTON”, then, in the next iteration, we create “HILLARY [RODHAM CLINTON]”.

4.2 Topic Discovery

Topics are themes that are discussed extensively within a document and that are shared among several documents. The TFIDF measure, used in the information retrieval field, coarsely reflects the characteristics of a topic term. Our algorithm for topic discovery employs the TFIDF measure [12] for selecting the topic labels and the OnTopic training procedure for creating topic support words. The algorithm is shown below:

1. Find the K(=5) key terms for each document. The key terms are the top K terms ranked by their TFIDF score,

$$TFIDF(w, D) = n(w, D) \log\left(\frac{P}{df(w)}\right)$$

where $n(w, D)$ is the frequency of the term (word or phrase) w in document D , P is number of documents in the corpus, $df(w)$ is the number of documents that have the term w .

2. Prune the key term w if it is not a key term for more than Q (=4) documents. The surviving key terms over all the document collection are the topic categories for the document collection. Pruning of key-terms prevents creation of topics from terms that are extremely rare. In addition, pruning also eliminates errors in text due to transcription and errors in feature selection by subsystems like Identifinder and MDL phrase finder.
3. Using the OnTopic training procedure [1], create a topic model. The input to the trainer is the list of documents and the initial topic categories (surviving key terms) for each document. The OnTopic trainer employs the Estimate-Maximize procedure to obtain topic distributions for each topic category by maximizing the posterior probability, $P(\text{Set of topics given a Document})$. The training results in 100 support words on an average for each topic.

Table 1: Support words and Likelihood scores for the UTD topic “inhalation_anthrax”

Support Words	P (Term Topic)
anthrax	0.2671
inhalation_anthrax	0.1346
Lundgren	0.0525
anthrax_spores	0.0228
John_Rowland	0.0215
postal_workers	0.0184
mailroom	0.0175
germ_warfare_agent	0.0168
Kathy_Nguyen	0.0155
Miesel	0.0101
anthrax_attacks	0.0101

The support words along with the topic label completely define a topic. Some of the support words for the topic “inhalation anthrax”, created by the UTD system, on a collection of news stories (in English), are shown in Table 1.

5. TOPIC EVALUATION

To evaluate the topics, we assign the topics to the documents by using the OnTopic topic classification system [1]. The topic classification system assigns an expanded set of topics for each document. The expanded set includes the topics we failed to identify at the beginning and topics that are related to the document but the label did not appear in the document. For each document, the topic classification system provides a list of topics ranked by the log-posterior probability, $P(\text{topic} | \text{document})$. We

choose a set of five topics at the top-rank for each document. For each document, to eliminate duplicate topic sets and improve visual quality of the topics, a sub-string search for duplicate topics is done within the assigned set.

A subjective evaluation of a small sample (=100) of documents, randomly chosen from each test document collection was done. During the evaluation, the evaluators mark the topics as “correct” or “wrong”. To determine the quality of topics, the following questions were asked:

- Does the topic label relate to the document? Reject if the label is not related to the document.
- What is the quality of the topic label? We reject topic labels like “British Prime” (if they are created as topic labels).
- We also reject the topic labels if they are too generic e.g., “people”.
- Is the topic label similar to another topic label for the same document? We reject one of the “President Clinton” and “Bill Clinton” if they are assigned to the same document.
- If the topic label is too ambiguous and doesn’t convey the information from the document, even after placing the topic label in the context of the other labels, then reject the topic. For example, if the topics for a document are STOCK, ZERO, MARKET, then reject the topic ZERO.

A precision value is reported for each topic rank,

$$\text{Precision}(\text{rank } n) = \frac{\# \text{correct topics at rank } n}{\# \text{evaluated documents}}$$

Another measure of performance that is used in conjunction with precision is the recall measure. Recall is the percentage of topics inside a document that have been hypothesized by a system. To measure recall, human annotation of topics for a document is necessary and the annotated topics should have various levels of granularity. Thus, we refrained from including the recall measure in the UTD system evaluation.

Table 2: 95% confidence interval values for a random sample size of 100

Precision	95% confidence interval
95%	4.1%
90%	5.7%
85%	6.8%
80%	7.7%
70%	8.8%

We believe that evaluating randomly sampled 100 documents provides a reasonable understanding of the performance of the UTD system. Using the binomial model, the 95% confidence interval is given by

$$c = 1.92 \cdot \sqrt{\frac{p \cdot (1-p)}{n}},$$

where, c is the confidence interval, p is the probability of error, and n is the sample size (=100). Table 2 shows the 95% confidence intervals for various precision values.

Table 3: UTD Topic Precision for English and Arabic Documents

Topic Rank	English documents	Arabic documents
1	96%	88%
2	96%	78%
3	89%	73%
4	82%	83%
5	82%	73%

Table 4: Comparison of UTD system performance with TFIDF key term selection procedures

Topic rank	TFIDF terms ^δ (baseline performance)	UTD topics
1	92%	96%
2	83%	96%
3	87%	89%
4	81%	82%
5	68%	82%

^δ Topics are top 5 key terms for the document, ranked according to the TFIDF score (defined in step 1 of UTD algorithm described in section 4.2)

Table 5: Average number of topics assigned to each document

System	UTD on English	UTD on Arabic	TFIDF key-terms
Average number of topics per document	4.887	4.321	4.315

The confidence interval (shown in Table 2) indicates that for several evaluations, each evaluation done on a randomly chosen sample of 100 documents, 95% of the time the variation in the precision is limited by the confidence interval i.e., at $95\% \pm 4.1\%$ or $90\% \pm 5.7\%$ or $70\% \pm 8.8\%$. In addition, the evaluation sample size has to be quadrupled to reduce the confidence interval by half, which means a very time consuming subjective evaluation.

6. EXPERIMENTAL RESULTS

The UTD system was tested on both English and Arabic documents. For the test-run, 95,530 English documents were chosen from the Primary Source Media (PSM) data for the duration July 1995 through June 1996 and January 1997 through December 1997. We also added 350 documents from Yahoo daily news during the month of November 2001. The Arabic language UTD run was carried out on 37,057 Arabic documents from the Al-Hayat news source for the period January 2001 to November 2001. We ignored the stop words. The remaining words were stemmed to their common roots using the Porter stemming algorithm [8] for English and the Buckwalter stemming algorithm [9] for Arabic.

UTD run on English produced 24124 topics, with an average of 4.887 topics per document. The evaluation was done on a random sample of 100 documents from each of the English and Arabic document collections. From the precision scores are shown in Table 3, it can be seen that the topics discovered by the UTD system are very good.

For the Arabic data, 21737 topics were created with an average of 4.321 topics per document. The lower precision values (as indicated in Table 3) for the Arabic data, when compared to the results for English data, were mainly due to the presence of orthographic variations, errors in stemming due to multiple inflections of the same word, different character usage for similar looking glyphs and an incomplete stop word list.

In order to determine how valuable it was to find proper nouns, we performed an experiment on English data without the IdentiFinder names as features. We found that the degradation in precision value for the topics at rank 1 and 2 was only 2%.

To assess the importance of support words for finding topics, we compared the performance of UTD system against a baseline system. Our baseline system defines topics as the TFIDF key-terms for a document. For each document, five top ranked TFIDF

terms are selected as topics. The evaluation results for the baseline system and the UTD system are shown in Table 4. At the top ranked topic, both the baseline system and the UTD method work well, although the UTD method has half the errors. The 95% confidence interval for the top-1 result is 4%. Thus, these results are significant. At the lower ranks, the difference between the methods is larger. We believe this is because the lower topics, which are often more general, are less likely to include the topic name, or need the supporting evidence of the other words. Thus, the UTD method is better able to find them, while the baseline method using TFIDF makes more mistakes.

From the UTD experiments, we also observed that on an average, 30% of the topics assigned to a document (and subjectively judged as correct) are not in the initial topic set used in training. In addition, for 9% of the correct topics (after subjective evaluation) the topic labels are not present in the document, but the support words identified the topic. These two results indicate that (a) the UTD process finds good topic terms that would not have been selected by the TFIDF key-term selection procedure, and (b) the UTD process, by using the support words, finds topics for which the topic label is not in the document. These results reinforce our belief that support words are strong indicators of the presence of a topic inside the documents.

7. CONCLUSIONS

We have described a new method for discovering topics from a document collection. The UTD system performed well on English and Arabic corpora. Our approach of using support words as an additional evidence of a topic, resulted in improved performance when compared with a simple key-term selection approach. We were able to identify topics in documents even if the topic name is not in the document at all.

Our results have shown that names as features improve the performance by a small fraction. Thus, the same system can be used on a new language with minimum resources.

8. ACKNOWLEDGEMENTS

We thank Alex Fraser, Jinxi Xu and Mohammed Noamany for helping us with the Arabic pre-processing (stemming and stop word removal). We thank Mohammed Noamany for the great help in subjective evaluation of the Arabic data.

9. REFERENCES

[1] Schwartz, R., Imai, T., Kubala, F., Nguyen, L. and Makhoul, J. A Maximum Likelihood Model for Topic

Classification of Broadcast News, *Eurospeech-97*, Greece, 1997.

- [2] Kubala, F., Schwartz, R., Stone, R., and Weischedel, R. Named Entity Extraction from Speech, DARPA Broadcast News Transcription and Understanding Workshop, 1998.
- [3] Salton, G., and Singhal, A. Automatic Text Theme Generation and the Analysis of Text Structure, Technical Report 94-1438, Dept. of Computer Science, Cornell University, NY, 1994.
- [4] Carl de Marcken. The Unsupervised Acquisition of Lexicon from Continuous Speech, MIT Artificial Intelligence Laboratory, A.I. Memo No. 1558, 1995.
- [5] Pereira, F., Tishby, N., and Lee, L. Distributional clustering of English words, *Proceedings of ACM*, 183-190, 1993.
- [6] Hang Li, and Abe, N. Word clustering and disambiguation based on co-occurrence data, *Proceedings of COLING-ACL'98*, 749-755, 1998.
- [7] Biswas, G., Weinberg, J.B., and Fisher, D.H. ITERATE: A Conceptual Clustering Algorithm for Data Mining, *IEEE Transactions on Systems, Man, and Cybernetics – Part C: Applications and Reviews*, Vol. 28, 219-230, 1998.
- [8] Porter, M.F. An algorithm for suffix stripping, *Program*, 14(3), 130-137, 1980.
- [9] Buckwalter, T. Personal Communications, 2001.
- [10] Srivastava, A. Story Segmentation in Audio Indexing, M.S. Thesis, Northeastern University, Boston, 1999.
- [11] Leek, T.R. Minimum Description Length (MDL) Phrases. BBN Technologies, Technical Report 010405TRL, 2001.
- [12] Salton, G. The SMART retrieval system: Experiments in Automatic Document Processing. Prentice Hall, 1971.