Enhancing Text Clustering using Concept-based Mining Model

Shady Shehata Fakhri Karray Mohamed Kamel
Department of Electrical and Computer Engineering
University of Waterloo
Waterloo, Ontario, Canada N2L 3G1
{shady, karray, mkamel}@pami.uwaterloo.ca

Abstract

Most of text mining techniques are based on word and/or phrase analysis of the text. The statistical analysis of a term (word or phrase) frequency captures the importance of the term within a document. However, to achieve a more accurate analysis, the underlying mining technique should indicate terms that capture the semantics of the text from which the importance of a term in a sentence and in the document can be derived. A new concept-based mining model that relies on the analysis of both the sentence and the document, rather than, the traditional analysis of the document dataset only is introduced.

The proposed mining model consists of a concept-based analysis of terms and a concept-based similarity measure. The term which contributes to the sentence semantics is analyzed with respect to its importance at the sentence and document levels. The model can efficiently find significant matching terms, either words or phrases, of the documents according to the semantics of the text. The similarity between documents relies on a new concept-based similarity measure which is applied to the matching terms between documents.

Experiments using the proposed concept-based term analysis and similarity measure in text clustering are conducted. Experimental results demonstrate that the newly developed concept-based mining model enhances the clustering quality of sets of documents substantially.

1. Introduction

Natural Language Processing (NLP) is both a modern computational technology and a method of investigating and evaluating claims about human language itself. Text mining attempts to discover new, previously unknown information by applying techniques from natural language processing and data mining. Clustering, one of the traditional text data mining techniques, is unsupervised learning paradigm where clustering methods try to identify inherent groupings of the text documents so that a set of clusters are produced in which clusters exhibit high intra-cluster similarity and low inter-cluster similarity [1].

Usually, in text mining techniques, the frequency of a term (word or phrase) is computed to explore the importance of the term in the document. However, two terms can have the same frequency in a document, but one term might be contributing more to the meaning of its sentence than the other term. It is important to note that extracting the relations between verbs and their arguments in the same sentence has the potential for analyzing terms within a sentence. The information about who is doing what to whom clarifies the contribution of each term in a sentence to the meaning of the main topic of that sentence.

In this paper, a novel concept-based mining model is proposed. It captures the semantic structure of each term within a sentence and a document, rather than the frequency of the term within a document only. Each sentence is labeled by a semantic role labeler that determines the terms which contribute to the sentence semantics associated with their semantic roles in a sentence. Each term that has a semantic role in the sentence, is called a concept. Concepts can be either words or phrases and are totally dependent on the semantic structure of the sentence. When a new document is introduced to the system, the proposed mining model can detect a concept match from this document to all the previously processed documents in the data set by scanning the new document and extracting the matching concepts.

A new concept-based similarity measure called Conceptual Term Frequency (CTF) which makes use of concept-based term matching is proposed. This similarity measure outperforms other similarity measures that are based on term analysis models of the document dataset only. The similarity between documents is based on a combination of concept-based term analysis similarity within a sentence and concept-based term analysis similarity within a document. Similarity based on matching of concepts between document pairs, is shown to have a more significant effect

Proceedings of the Sixth International Conference on Data Mining (ICDM'06)
0-7695-2701-9/06 $20.00 © 2006 IEEE

on the clustering quality due to the similarity's insensitivity to noisy terms that can lead to an incorrect similarity. The concepts are less sensitive to noise when it comes to calculating document similarity. This is due to the fact that these concepts are originally extracted by the semantic role labeler. Thus, the matching among these concepts is less likely to be found in non-related documents.

The clustering results produced by the concept-based similarity combination has higher quality than those produced by a single-term analysis similarity only. The results are evaluated using two quality measures, the F-measure and the Entropy. Both of these quality measures showed improvement versus the use of the single-term method when the concept-based similarity measure is used to cluster sets of documents.

Following are the explanations of the important terms used in this paper:
- *Verb-argument structure*: (e.g John hits the ball). "hits" is the verb. "John" and "the ball" are the arguments of the verb "hits",
- *Label*: A label is assigned to an argument. e.g: "John" has subject (or Agent) label. "the ball" has object (or theme) label,
- *Term*: is either an argument or a verb. Term is also either a word or a phrase (which is a sequence of words),
- *Concept*: in the new proposed mining model, concept is a labeled term.

The rest of this paper is organized as follows. Section 2 introduces the thematic roles background. The concept based mining model which includes concept-based term analysis and concept-based similarity measure is presented in section 3. Experimental results are presented in section 4. The last section summarizes the conclusions and suggests future work.

2. Thematic Roles Background

Generally, the semantic structure of a sentence can be characterized by a form of verb argument structure. The study of the roles associated with verbs is referred to a thematic role or case role analysis [7]. Thematic roles, first proposed by Gruber and Fillmore [3], are sets of categories that provide a shallow semantic language to characterize the verb arguments.

Recently, there have been many attempts to label thematic roles in a sentence automatically. Gildea and Jurafsky [5] were the first to apply a statistical learning technique to the FrameNet database. They presented a discriminative model for determining the most probable role for a constituent, given the frame, predicator, and other features. A machine learning algorithm for shallow semantic parsing was proposed in [11]. It is an extension of the work in [5]. Their algorithm is based on using Support Vector Machines (SVM) which results in improved performance over that of earlier classifiers by Gildea and Jurafsky [5].

To the best of our knowledge, there is no research work that employs the full potential of the output of the role labeling task in mining text based on the semantics analysis of the text.

3. Concept-based Mining Model

The proposed concept-based mining model consists of concept-based term analysis and concept-based similarity measure.A raw text document is the input to the proposed model. Each document has well defined sentence boundaries. Each sentence in the document is labeled automatically based on the PropBank notations [8]. After running the semantic role labeler, each sentence in the document might have one or more labeled verb argument structures. The number of generated labeled verb argument structures is entirely dependent on the amount of information in the sentence. The sentence that has many labeled verb argument structures includes many verbs associated with their arguments. The labeled verb argument structures, the output of the role labeling task, are captured and analyzed by the concept-based mining model.

In this model, both the verb and the argument are considered as *terms*. One term can be an argument to more than one verb in the same sentence. This means that this term can have more than one semantic role in the same sentence. In such cases, this term plays important semantic roles that contribute to the meaning of the sentence. In the concept-based mining model, a labeled term either word or phrase is considered as *concept*.

3.1. Concept-Based Term Analysis

The objective of this task is to achieve a concept-based term analysis (word or phrase) on the sentence and document levels rather than a single-term analysis in the document set only.

To analyze each concept at the sentence-level, a concept-based frequency measure, called the conceptual term frequency (ctf) is proposed. The ctf is the number of occurrences of concept c in verb argument structures of sentence s. The concept c, which frequently appears in different verb argument structures of the same sentence s, has the principal role of contributing to the meaning of s.

To analyze each concept at the document-level, the term frequency tf, the number of occurrences of a concept (word or phrase) c in the original document, is calculated. The process of calculating tf and ctf measures in a set of documents is attained by the proposed algorithm which is called (Concept-based Term Analyzer).

3.1.1 Algorithm: Concept-based Term Analyzer

---

1. d_{doci} is a new Document
2. L is an empty List (L is a matching concept list)
3. **for** each sentence s in d **do**
4. c_i is a new concept in s
5. **for** each concept $c_i \in \{c_1, c_2, ..., c_n\}$ in s **do**
6. compute tf_i of c_i in d
7. compute ctf_i of c_i in s in d
8. **end for**
9. **for** each d_k, where $k = \{0, 1, .., doci - 1\}$, c_i exist **do**
10. **for** each concept $c_j \in \{c_1, c_2, ..., c_m\}$ in s **do**
11. **if** $(c_i == c_j)$ **then**
12. compute $tfweight = avg(tf_i, tf_j)$
13. compute $ctfweight = avg(ctf_i, ctf_j)$
14. add new concept matches to L
15. **end if**
16. **end for**
17. **end for**
18. **end for**
19. output the matched concepts list L

The concept-based analyzer algorithm describes the process of calculating the tf and the ctf of the matched concepts in the documents. The procedure begins with processing a new document (at line 1) which has well defined sentence boundaries. Each sentence is semantically labeled according to [8]. The lengths of the matched concepts and their verb argument structures are stored for the concept-based similarity calculations in section 3.2.

For each sentence (in the for loop at line 3) the concepts of the verb argument structures which represent the semantic structures of the sentence are processed sequentially. Each concept in the current document is matched with the other concepts in the previously processed documents. To match the concepts in previous documents is accomplished by keeping a concept list L that holds the entry for each of the previous documents that shares a concept with the current document.

After the document is processed, L contains all the matching concepts between the current document and any previous document that shares at least one concept with the new document. Finally, L is output as the list of documents with the matching concepts and the necessary information about them. The concept-based term analyzer algorithm is capable of matching each concept in a new document (d) with all the previously processed documents in $O(m)$ time, where m is the number of concepts in d.

Consider the following sentence:
*"We have **noted** how some electronic techniques, **developed** for the defense effort, have eventually been **used** in commerce and industry".*
In this sentence, the semantic role labeler identifies three target words (verbs), marked by bold, which are the verbs that represent the semantic structure of the meaning of the sentence. These verbs are *noted*, *developed*, and *used*. Each one of these verbs has its own arguments as follows:
- [ARG0 We] [TARGET **noted**] [ARG1 how some electronic techniques developed for the defense effort have eventually been used in commerce and industry]
- We have noted how [ARG1 some electronic techniques] [TARGET **developed**] [ARGM-PNC for the defense effort] have eventually been used in commerce and industry
- We have noted how [ARG1 some electronic techniques developed for the defense effort] have [ARGM-TMP eventually] been [TARGET **used**] [ARGM-LOC in commerce and industry]

Arguments labels[1] are numbered Arg0, Arg1, Arg2, and so on depending on the valency of the verb in sentence. The meaning of each argument label is defined relative to each verb in a lexicon of Frames Files [8]. generality, Arg0 is very consistently assigned an Agent-type meaning, while Arg1 has a Patient or Theme meaning almost as consistently [8]. Thus, this sentence consists of the following three verb argument structures:
- First verb argument structure: [ARG0 We], [TARGET noted], and [ARG1 how some electronic techniques developed for the defense effort have eventually been used in commerce and industry]
- Second verb argument structure: [ARG1 some electronic techniques], [TARGET developed], and [ARGM-PNC for the defense effort]
- Third verb argument structure: [ARG1 some electronic techniques developed for the defense effort], [ARGM-TMP eventually], [TARGET used], and [ARGM-LOC in commerce and industry]

A cleaning step is performed to remove stop-words that have no significance, and to stem the words using the popular Porter Stemmer algorithm [10]. The terms generated after this step are called *concepts* as follows:
- Concepts in the first verb argument structure: "note", "electron techniqu develop defens effort evenut commerc industri"
- Concepts in the second verb argument structure: "electron techniqu", "develop", and "defens effort"
- Concepts in the third verb argument structure: "electron techniqu develop defens effort", "eventu", and "commerc industri".

It is imperative to note that these concepts are extracted from the same sentence. Thus, the concepts mentioned in this example sentence are: "note", "electron techniqu develop defens effort evenut commerc industri", "electron techniqu", "develop", "defens effort", "electron techniqu

[1] Because the meaning of each argument number is defined on a per-verb basis, there is no straightforward mapping of meaning between arguments with the same number [8].

Table 1. Concept-based Term Analysis

Row Number	**Sentence Concepts**	**CTF**
(1)	note	1
(2)	electron techniqu develop defens effort evenut commerc industri	1
(3)	electron techniqu	3
(4)	develop	3
(5)	defens effort	3
(6)	electron techniqu develop defens effort	2
(7)	eventu	2
(8)	commerc industri	2
	Individual Concepts	**CTF**
(9)	electron	3
(10)	techniqu	3
(11)	defens	3
(12)	effort	3
(13)	evenut	2
(14)	commerc	2
(15)	industri	2

develop defens effort", "eventu", and "commerc industri".

The traditional analysis methods assign same weight for the words that appear in the same sentence. However, the concept-based term analysis discriminates among terms that represents the sentence concepts. This analysis is entirely based on the semantic analysis of the sentence. In this example, some concepts have higher conceptual term frequency ctf than others as shown in Table 1. In such cases, these concepts (with high ctf) contribute to the meaning of the sentence more than other concepts (with low ctf).

As shown in Table 1, the concept-based term analysis computes the ctf measure for: the concepts extracted from the verb argument structures of the sentence, which are in Table 1 from row (1) to row (8), the concepts overlapped with other concepts that are in Table 1 from row (3) to row (8), and the individual concepts in the sentence, which are in Table 1 from row (9) to row (15).

In this example, the topic of the sentence is about the *electronic techniques*. These concepts have the highest ctf value with 3. In addition, the concept *note* which has the lowest ctf, has no significant effect on the topic of the sentence. Thus, the concepts with high ctf such as *electronic, techniques, developed, defense, and effort* present indeed the topic of the sentence.

3.2. A Concept-Based Similarity Measure

Concepts convey local context information, which is essential in determining an accurate similarity between documents. A new concept-based similarity measure, based on matching concepts at the sentence and document levels rather than on individual terms (words) only, is devised. The concept-based similarity measure relies on two critical aspects. First, the analyzed labeled terms are the concepts that capture the semantic structure of each sentence. Secondly, the frequency of a concept is used to measure the contribution of the concept to the meaning of the sentence, as well as to the main topics of the document. These aspects are measured by the proposed concept-based similarity measure which measures the importance of each concept at the document-level by the tf measure and at the sentence-level by the ctf measure. The concept-based measure exploits the information extracted from the concept-based term analyzer algorithm to better judge the similarity between the documents.

This similarity measure is a function of the following factors: the number of matching concepts (m) in the verb arguments structures in each document (d), the total number of sentences (s) in each document d, the total number of the labeled verb argument structures (v) in each sentence s, the tf_i of each concept c_i in each document d where $(i = 1, 2, ..., m)$, the ctf_i of each concept c_i in s for each document d where $(i = 1, 2, ..., m)$, the length (l) of each concept in the verb argument structure in each document d, and the length (s) of each verb argument structure which contains a matched concept.

The conceptual term frequency (ctf) is an important factor in calculating the concept-based similarity measure between documents. The more frequent the concept appears in the the verb argument structures of a sentence in a document, the more conceptually similar the documents are.

The concept-based similarity between two documents d_1 and d_2 is calculated by:

$$sim_c(d_1, d_2) = \sum_{i=1}^{m} max(\frac{l_i}{S_{i_1}}, \frac{l_i}{S_{i_2}}) * weight_{i_1} * weight_{i_2}, \quad (1)$$

where
$weight_{i_1} = tfweight_{i_1} + ctfweight_{i_1}$
$weight_{i_2} = tfweight_{i_2} + ctfweight_{i_2}$

The concept-based weight of concept i_1 in document d_1 is presented by $weight_{i_1}$. In calculating $weight_{i_1}$, the $tfweight_{i_1}$ value presents the weight of concept i in the first document d_1 at the document-level and the $ctfweight_{i_1}$ value presents the weight of the concept i in the first document d_1 at the sentence-level based on the contribution of concept i to the semantics of the sentences in d_1. The sum between the two values of $tfweight_{i_1}$ and $ctfweight_{i_1}$ presents an accurate measure of the contribution of each concept to the meaning of the sentences and to the topics mentioned in a document. The term $weight_{i_2}$ is applied to the second document d_2.

Equation 1 assigns a higher score, as the matching concept length approaches the length of its verb argument structure, because this concept tends to hold more conceptual information related to the meaning of its sentence.

In equation 2, the tf_{ij_1} value is normalized by the length

of the document vector of the term frequency tf_{ij} in the first document d_1, where $j = 1, 2, ..., cn_1$

$$tfweight_{i_1} = \frac{tf_{ij_1}}{\sqrt{\sum_{j=1}^{cn_1} (tf_{ij_1})^2}}, \quad (2)$$

cn_1 is the total number of the concepts which has a term frequency value in the document d_1. In equation 3, the ctf_{ij_1} value is normalized by the length of the document vector of the conceptual term frequency ctf_{ij} in the first document d_1 where $j = 1, 2, ..., cn_1$

$$ctfweight_{i_1} = \frac{ctf_{ij_1}}{\sqrt{\sum_{j=1}^{cn_1} (ctf_{ij_1})^2}}, \quad (3)$$

cn_1 is the total number of concepts which has a conceptual term frequency value in the document d_1. The same normalization equations are applied to the weights of the concepts in the second document d_2 as shown in equations 4 and 5

$$tfweight_{i_2} = \frac{tf_{ik_2}}{\sqrt{\sum_{k=1}^{cn_2} (tf_{ik_2})^2}}, \quad (4)$$

$$ctfweight_{i_2} = \frac{ctf_{ik_2}}{\sqrt{\sum_{k=1}^{cn_2} (ctf_{ik_2})^2}}. \quad (5)$$

For the single-term similarity measure, the cosine correlation similarity measure in [13] is adopted with the popular TF-IDF [2] (Term Frequency/Inverse Document Frequency) term weighting. Recall that the cosine measure calculates the cosine of the angle between the two document vectors d_1 and d_2. Accordingly, the single-term similarity measure (sim_s) is $sim_s(d_1, d_2) = \cos(x, y) = \frac{d_1 \cdot d_2}{\|d_1\| \ \|d_2\|}$

4. Experimental Results

The experimental setup consisted of three datasets. The first data set consisted of 23,115 ACM abstract articles collected from the ACM digital library. The ACM articles are classified according to the ACM computing classification system into five main categories: general literature, hardware, computer systems organization, software, and data. The second data set has 12,902 documents from the Reuters 21578 dataset. There are 9,603 documents in the training set, 3,299 documents in the test set, and 8,676 documents are unused. Out of the 5 category sets, the topic category set contains 135 categories, but only 95 categories have at least one document in the training set. These 95 categories were used in the experiment. The third dataset consisted of 361 samples from the Brown corpus [4]. Each sample has 2000+ words. The Brown corpus main categories used in the experiment were: press: reportage, press: reviews, religion, skills and hobbies, popular lore, belles-letters, learned, fiction: science, fiction: romance, and humor.

The similarities which are calculated by the concept-based model are used to compute a similarity matrix among documents. Three standard document clustering techniques are chosen for testing the effect of the concept-based similarity on clustering [6]: (1) Hierarchical Agglomerative Clustering (HAC), (2) Single Pass Clustering, and (3) k-Nearest Neighbor (kNN)[2].

In order to evaluate the quality of the clustering, two quality measures widely used in the text mining literature for the purpose of document clustering [12] are adopted. The first is the ***F-measure***, which combines the Precision and Recall measures from the Information Retrieval literature. The precision P and recall R of a cluster j with respect to a class i are defined as $P = Precision(i, j) = \frac{M_{ij}}{M_j}$ and $R = Recall(i, j) = \frac{M_{ij}}{M_i}$ where M_{ij} is the number of members of class i in cluster j, M_j is the number of members of cluster j, and M_i is the number of members of class i. The F-measure of a class i is defined as $F(i) = \frac{2PR}{P+R}$.

The second measure is the ***Entropy***, which measures how homogeneous a cluster is. The higher the homogeneity of a cluster, the lower the entropy is, and vice versa. For every cluster j in the clustering result C, the probability p_{ij} that a member of cluster j belongs to class i is computed.

Basically, the aim is to maximize the F-measure, and minimize the Entropy of clusters to achieve high quality clustering. The results listed from Table 2 to Table 8 show the improvement on the clustering quality using the concept-based model. The ward and the complete linkages were used as the cluster distance measures for the HAC method since they tend to produce tight clusters with small diameter as shown in Tables 2, 3, 4, and 5. A document-to-cluster similarity threshold of 0.3 was used in the single pass clustering method as depicted in Tables 6 and 7. A k of 5 and a cluster similarity threshold of 0.35 were used in the kNN method as illustrated in Tables 8 and 9. The parameters chosen for the different algorithms were the ones that produced best results.

The percentage of improvement ranges from +27.94% to +98.74% increase in the F-measure quality, and -23.03% to -95.68% drop in Entropy (lower is better for Entropy).

5. Conclusions

This work bridges the gap between natural language processing and text mining disciplines. A new concept-based mining model composed of two components, is proposed to improve the text clustering quality. By exploiting the semantic structure of the sentences in documents, a better text clustering result is achieved. The first component is the new concept-based term analysis which an-

[2]Though kNN is mostly known to be used for classification, it has also been used for clustering (example could be found in [9]).

Table 2. F-measure of the HAC (Ward)

	Single-Term	Concept-based	Improvement
Reuters	0.723	0.925	+27.94%
ACM	0.697	0.918	+31.70%
Brown	0.581	0.906	+55.93%

Table 3. Entropy of the HAC (Ward)

	Single-Term	Concept-based	Improvement
Reuters	0.251	0.012	-95.21%
ACM	0.317	0.043	-86.43%
Brown	0.385	0.018	-95.32%

Table 4. F-measure of the HAC (Complete)

	Single-Term	Concept-based	Improvement
Reuters	0.623	0.907	+45.58%
ACM	0.481	0.895	+86.07%
Brown	0.547	0.901	+64.71%

Table 5. Entropy of the HAC (Complete)

	Single-Term	Concept-based	Improvement
Reuters	0.315	0.025	-92.06%
ACM	0.362	0.135	-62.7%
Brown	0.401	0.021	-94.76%

Table 6. F-measure of the Single Pass

	Single-Term	Concept-based	Improvement
Reuters	0.411	0.816	+98.54%
ACM	0.398	0.791	+98.74%
Brown	0.437	0.804	+83.98%

Table 7. Entropy of the Single Pass

	Single-Term	Concept-based	Improvement
Reuters	0.523	0.067	-87.18%
ACM	0.608	0.152	-75%
Brown	0.551	0.045	-91.83%

Table 8. F-measure of the kNN

	Single-Term	Concept-based	Improvement
Reuters	0.511	0.917	+79.45%
ACM	0.491	0.891	+81.46%
Brown	0.462	0.902	+95.23%

Table 9. Entropy of the kNN

	Single-Term	Concept-based	Improvement
Reuters	0.348	0.015	-95.68%
ACM	0.402	0.111	-29.1%
Brown	0.316	0.023	-23.03%

alyzes the semantic structure of each sentence to capture the sentence concepts. Then, the component analyzes each concept at the sentence and document levels. The second component is the concept-based similarity measure which allows measuring the importance of each concept with respect to the semantics of the sentence, and the topic of the document. This allows performing concept matching and concept-based similarity calculations among documents in a very robust and accurate way. The quality of text clustering achieved by this model significantly surpasses the traditional single-term based approaches.

There are a number of possibilities for extending this work. One direction is to improve the accuracy of the similarity calculations of the documents by employing different similarity calculation strategies. Another future direction is to link the presented work to web document clustering.

References

[1] K. J. Cios, W. Pedrycz, and R. W. Swiniarski. *Data Mining Methods for Knowledge Discovery*. Kluwer Academic Publishers, Boston, 1998.

[2] R. Feldman and I. Dagan. Knowledge discovery in textual databases (kdt). In *Proceedings of First International Conference on Knowledge Discovery and Data Mining*, pages 112–117, 1995.

[3] C. Fillmore. *The case for case. Chapter in: Universals in Linguistic Theory*. Holt, Rinehart and Winston, Inc., New York, 1968.

[4] W. Francis and H. Kucera. *Manual of information to accompany a standard corpus of present-day edited american english, for use with digital computers*, 1964.

[5] D. Gildea and D. Jurafsky. Automatic labeling of semantic roles. *Computational Linguistics*, 28(3):245–288, 2002.

[6] A. K. Jain and R. C. Dubes. *Algorithms for Clustering Data*. Prentice Hall, Englewood Cliffs, N.J., 1988.

[7] D. Jurafsky and J. H. Martin. *Speech and Language Processing*. Prentice Hall Inc., 2000.

[8] P. Kingsbury and M. Palmer. Propbank: the next level of treebank. In *Proceedings of Treebanks and Lexical Theories*, 2003.

[9] S. Y. Lu and K. S. Fu. A sentence to sentence clustering procedure for pattern analysis. *IEEE Transactions on Systems Mans and Cybernetics*, 8:381–389, 1978.

[10] M. F. Porter. An algorithm for suffix stripping. *Program*, 14(3):130–137, July 1980.

[11] S. Pradhan, W. Ward, K. Hacioglu, J. Martin, and D. Jurafsky. Shallow semantic parsing using support vector machines. In *Proceedings of the Human Language Technology/North American Association for Computational Linguistics (HLT/NAACL)*, 2004.

[12] M. Steinbach, G. Karypis, and V. Kumar. A comparison of document clustering techniques. In *Knowledge Discovery and Data Mining (KDD) Workshop on TextMining*, August 2000.

[13] A. Strehl, J. Ghosh, and R. Mooney. Impact of similarity measures on web-page clustering. In *Proceedings of 17th National Conference on Artificial Intelligence: Workshop of Artificial Intelligence for Web Search (AAAI)*, pages 58–64, 2000.