Dictionary Requirements for Text Classification: A Comparison of Three Domains

Ellen Riloff
Department of Computer Science
University of Utah
Salt Lake City, UT 84112
riloff@cs.utah.edu

Abstract

The type of dictionary required for a natural language processing system depends on both the nature of the task and the domain. For example, an in-depth comprehension task probably requires more knowledge than an information retrieval task. Similarly, technical domains are fundamentally different from event-based domains and require different types of lexical knowledge. We explore these issues by comparing the performance of four text classification algorithms that use varying amounts of lexical knowledge. We tested the algorithms on three different domains: terrorism, joint ventures, and microelectronics. We found that the algorithms produced dramatically different results on each domain, suggesting that the nature of the domain strongly influences the types of knowledge required to achieve good performance.

Introduction

Knowledge-based natural language processing systems (NLP) use a dictionary to represent lexical knowledge associated with one or more domains. But what types of knowledge need to be represented? The answer to this question depends on both the task (the problem) and the domain (the subject). An information retrieval task may require substantially different types of lexical knowledge than an in-depth comprehension task. Similarly, domains have different properties that require varying levels of representation. For example, medical domains are typically dominated by technical words and jargon, while business-oriented domains generally use more common but varied language.

Text classification is an information retrieval task for which texts are assigned to one or more categories. We have developed three text classification algorithms that automatically categorize texts as either "relevant" to a given domain or "irrelevant". Our approach is based on a type of natural language processing called *information extraction*. Information extraction (IE) systems extract domain-specific information from text (see [MUC-3 Proceedings, 1991; MUC-4 Proceedings, 1992; MUC-5 Proceedings, 1993]). For example, an IE system developed for a terrorism domain might extract the names of perpetrators, victims, targets, weapons, dates, and locations of terrorist incidents. IE systems typically use a dictionary of domain-specific structures or patterns to identify and extract relevant information. The dictionary may be manually constructed, or automatically constructed using a training corpus (e.g., [Riloff, 1994; Riloff, 1993a; Kim and Moldovan, 1993]).

We conducted a set of experiments to compare the dictionary requirements for IE-based text classification in three different domains: terrorism, joint ventures, and microelectronics. First, we briefly describe three IE-based text classification algorithms that use varying amounts of extracted information to classify texts: the relevancy signatures algorithm, the augmented relevancy signatures algorithm, and a case-based text classification algorithm. The relevancy signatures algorithm classifies texts using simple phrases, the augmented relevancy signatures algorithm uses phrases and local semantic context, and the case-based algorithm uses larger pieces of context. Then we compare the results of these algorithms and a word-based algorithm across the three domains. The algorithms behaved very differently on each domain, suggesting that both the nature of the task and the domain play a crucial role in determining what types of lexical knowledge are most important.

IE-Based Text Classification

The IE-based text classification algorithms use a conceptual sentence analyzer called CIRCUS [Lehnert, 1991] to extract domain-specific information from text. CIRCUS uses a dictionary of structures called *concept nodes* to extract relevant information. The concept node dictionary used in the experiments with the terrorism domain was built manually, but the concept node dictionaries for the joint ventures and microelectronics domains were constructed automatically by a system called AutoSlog [Riloff, 1994; Riloff, 1993a] using a training corpus. In general, concept nodes recognize local linguistic patterns, including simple verb forms (active, passive, infinitive), and noun or verb phrases followed by prepositional phrases. Figure 1 shows a sample sentence and a concept node activated by the sentence. The concept node $MURDER-PASSIVE$ is activated by passive forms of verbs associated with murder, such as "X was murdered by Y", "X1 and X2 were murdered by Y", and "X has been murdered by Y". In this case, $MURDER-PASSIVE$ was activated by the verb "murdered" and extracted the three peasants as murder victims and the guerrillas as perpetrators.

Sentence:
Three peasants were murdered by guerrillas.

$MURDER-PASSIVE$
victim = "three peasants"
perpetrator = "guerrillas"

Figure 1: An instantiated concept node

The first text classification algorithm is the *relevancy signatures algorithm* [Riloff and Lehnert, 1994; Riloff and Lehnert, 1992]. A *signature* is a word paired with a concept node; together, this pair represents a unique set of linguistic expressions. For example, the word "murdered" could be paired with one concept node to recognize passive verb forms (i.e., "X was murdered by Y") and a different concept node to represent active verb forms (i.e., "X murdered Y").

The first step of the algorithm is to analyze a preclassified training corpus using CIRCUS. The instantiated concept nodes produced by CIRCUS are transformed into signatures, and statistics are calculated to determine how often each signature appears in relevant texts. Thresholds are used to identify signatures that occur much more frequently in relevant texts than irrelevant texts, and these are labeled as *relevancy signatures.* For example, we might assume that a signature represents an expression that is a good indicator for the domain if 90% of the occurrences of the signature are in relevant texts.

The second algorithm uses both linguistic expressions and local semantic context to classify texts. The *augmented relevancy signatures algorithm* collects frequency statistics for signatures as well as semantic features associated with extracted information. Figure 2 shows the signatures and slot triples that represent the instantiated concept node $MURDER-PASSIVE$ in Figure 1. Each slot triple contains the event type, slot type, and semantic feature associated with a piece of extracted information. The first slot triple in Figure 2 represents the peasants, and the second slot triple represents the guerrillas. A domain-specific semantic feature dictionary is used to find the features associated with a noun phrase.

Signature:
<murdered, $MURDER-PASSIVE$>
Slot Triples:
(murder, victim, CIVILIAN)
(murder, perpetrator, TERRORIST)

Figure 2: Signatures and Slot Triples

The augmented relevancy signatures algorithm identifies signatures and slot triples that are highly correlated with relevant texts in the training corpus [Riloff and Lehnert, 1994; Riloff and Lehnert, 1992]. A new text is classified as relevant if it produces an instantiated concept node that represents both a signature and a slot triple that were highly correlated with relevance, independently. Intuitively, a text is classified as relevant if it contains a linguistic expression and a piece of extracted information that are both strongly associated with the domain.

The third text classification algorithm uses multiple pieces of extracted information to represent larger natural language contexts. The *case-based text classification algorithm* [Riloff and Lehnert, 1994; Riloff, 1993b] collects all of the information extracted from a sentence and merges it into a single case structure. Figure 3 shows a sample sentence and the resulting case structure.

SENTENCE: Two vehicles were destroyed and an unidentified office of the agriculture and livestock ministry was heavily damaged following the explosion of two bombs yesterday afternoon.

CASE
Signatures: (<destroyed, $DESTRUCTION-PASSIVE$>,
<damaged, $DAMAGE-PASSIVE$>,
<bombs, $WEAPON-BOMB$>)
Perpetrators: nil
Victims: nil
Targets: (GOVT-OFFICE-OR-RESIDENCE VEHICLE)
Instruments: (BOMB)

Figure 3: A sample sentence and corresponding case

In the training phase, the training corpus is analyzed by CIRCUS, the concept nodes produced by each sentence are merged into a case, and each case is stored in the case base. In the classification phase, a new text is processed by CIRCUS, a case is generated for each sentence, and the case base is consulted to determine whether there are any similar cases in the case base. A case is considered to be similar if it shares a signature,

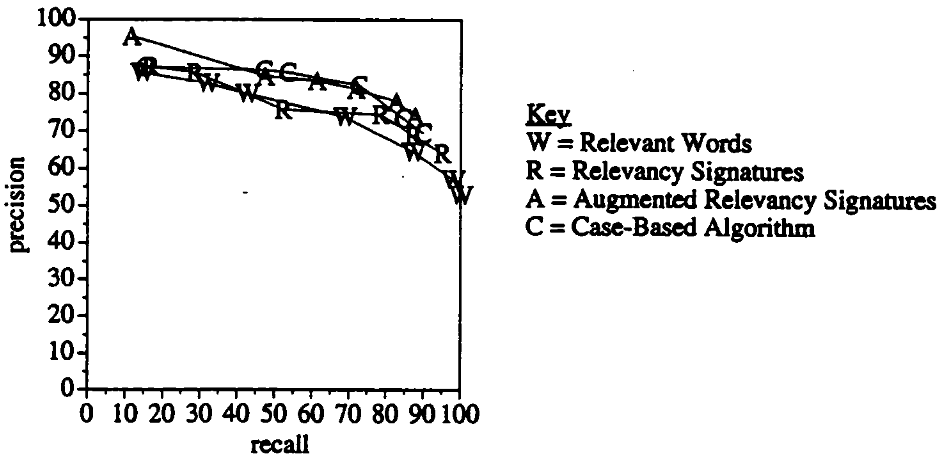

Figure 4: Text classification results for the terrorism domain

a semantic feature representing extracted information, and the same types of information (e.g., a victim and a perpetrator). If a high percentage of similar cases came from relevant texts, then the case probably contains relevant information and is classified as relevant.

The case-based algorithm is similar to the augmented relevancy signatures algorithm except that the frequency statistics are computed for several pieces of information *together*. In contrast, the augmented relevancy signatures algorithm computes statistics for signatures and slot triples separately. By using rich natural language contexts, the case-based algorithm can classify texts that are inaccessible to the previous algorithms because some sentences contain multiple pieces of information that are relevant *together* but not independently.

Experimental Results in Three Domains

We compared the performance of these algorithms, and a fourth word-based algorithm, on three different domains. The fourth algorithm, called the *relevant words algorithm*, is exactly the same as the relevancy signatures algorithm except that the statistics are computed for individual words instead of signatures.

Figure 4 shows the results in the terrorism domain on 1700 texts from the MUC-4 corpus. 53% of the texts were relevant to the MUC-4 domain guidelines [MUC-4 Proceedings, 1992]. Each algorithm was evaluated using a 10-fold cross-validation design (see [Riloff, 1994; Riloff and Lehnert, 1994] for details). We used 7 different sets of threshold values to produce a spectrum of recall/precision results, so each data point represents one recall/precision point achieved by the algorithm. Recall and precision are standard metrics used in the information retrieval community. *Recall* measures the percentage of relevant texts that were correctly classified as relevant by the algorithm. *Precision* measures the percentage of texts classified as relevant that actually are relevant.

The augmented relevancy signatures algorithm and the case-based algorithm achieved considerably higher combined recall and precision scores than the word-based and relevancy signatures algorithms. This is because context is crucially important for the terrorism domain. The definition of terrorism for this experiment stated that the perpetrator must be a terrorist and the target must be civilian.[1] To make these distinctions, two types of semantic information are essential: (1) the role of the object, e.g. whether someone is the perpetrator or victim of a crime, and (2) the semantics of the object itself, e.g., whether someone is a guerrilla or a civilian.

The role of an object can be identified through local linguistic expressions, for example "X was murdered" indicates that X is a murder victim. Isolated keywords (e.g., the word "murdered" alone) do not provide enough context to identify role objects, which explains why the word-based algorithm had difficulty. Furthermore, role objects can only be identified with context. For example, it is impossible to identify a victim or perpetrator simply by looking at their name: "John was murdered" indicates that John is a victim while "John murdered Steve" indicates that John is a perpetrator.

Relevancy signatures also had some trouble because they don't represent the semantic features associated with role objects, but the augmented relevancy signatures algorithm and the case-based algorithm do. For example, both algorithms used a semantic feature dictionary to recognize "peasants" as CIVILIAN, and "guerrillas" as TERRORIST. The additional context allows these algorithms to distinguish terrorist perpetrators (e.g., "a guerrilla") from civilian perpetrators ("a burglar"), and

[1] The guidelines for the terrorism domain were specified by the MUC-4 organizers (see [MUC-4 Proceedings, 1992]).

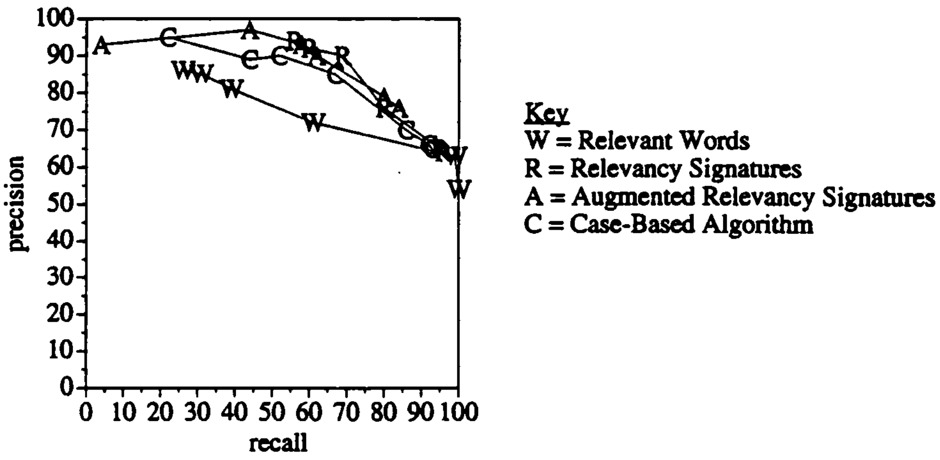

Figure 5: Text classification results for the joint ventures domain

military targets (e.g., "an army base") from civilian targets (e.g., "the U.S. embassy").

Figure 5 shows the results for the joint ventures domain using a training corpus of 1200 texts.[2] 54% of the 1200 texts were relevant to the domain.[3] In the joint ventures domain, all three IE-based algorithms performed well, and considerably better than the word-based algorithm.

The joint ventures domain is characterized by key words and phrases that are associated with joint venture activities, such as "joint-venture" and "tie-up". These words are strong indicators of joint venture activities, but the classification task was to label a text as relevant only if it mentioned a specific joint venture between two named partners. Therefore finding the word "joint-venture" was not enough to warrant a relevant classification, and consequently the word-based algorithm did not perform well.

To illustrate, we probed the corpus with several keywords that are associated with joint ventures and calculated recall and precision scores based upon how many relevant and irrelevant texts were retrieved. Figure 6 shows that some keywords and phrases performed well, for example the phrase "joint venture"[4] achieved 93.3% recall and 88.9% precision. However, many seemingly good keywords produced poor results. For example, the hyphenated word "joint-venture" achieved only 73.2% precision because it is often used a modifier where the main concept is not a joint venture (as in "joint-venture legislation"). Interestingly, the plural word "ventures" produced only 58.8% precision while the singular form "venture" achieved 82.8% precision. This is because the plural form often refers to joint ventures in general (as in "there have been many joint ventures this year") and the singular form is usually used to describe a specific joint venture (as in "Toyota formed a joint venture with Nissan"). We observed this phenomenon in the terrorism domain as well, where words such as "assassinations" and "bombings" referred to these types of events in general but did not describe any specific incident.

Words	Recall	Precision
joint, venture	93.3%	88.9%
tie-up	2.5%	84.2%
venture	95.5%	82.8%
jointly	11.0%	78.9%
joint-venture	6.4%	73.2%
consortium	3.6%	69.7%
joint, ventures	19.3%	66.7%
partnership	7.0%	64.3%
ventures	19.8%	58.8%

Figure 6: Recall and precision scores for joint venture keywords

The relevancy signatures algorithm, however, represents linguistic expressions that include prepositions, such as "venture with" and "project between". Figure 7 shows the precision rates associated with similar patterns recognized by signatures. The most interesting result was that the presence of the prepositions produced much more effective classification terms. For example, "venture between" achieved 100% precision while "venture" only achieved 82.8% precision, and "tie-up with" achieved 100% precision while "tie-up" alone achieved only 84.2% precision. In this domain, the prepositions often indicate that a partner is

[2]The corpus contained 719 texts from the MUC-5 corpus [MUC-5 Proceedings, 1993] and 481 texts from the Tipster detection corpus [Tipster Proceedings, 1993] (see [Riloff, 1994] for details of how these texts were selected).

[3]According to the MUC-5 domain guidelines [MUC-5 Proceedings, 1993].

[4]Not necessarily in adjacent positions.

mentioned, so the phrase usually refers to a specific joint venture. In some sense, prepositions such as "with" and "between" act as implicit pointers to the partners.

Signature Pattern	Precision
venture between <entity>	100.0%
venture with <entity>	95.9%
venture of <entity>	95.4%
venture by <entity>	90.9%
project between <entity>	100.0%
project with <entity>	75.0%
set up with <entity>	94.7%
set up by <entity>	66.7%
tie-up with <entity>	100.0%
<entity> construct	100.0%
<facility> was constructed	63.6%
<entity> form company	100.0%
<entity> set up company	100.0%
<entity> join forces	100.0%
<entity> agreed to form	100.0%

Figure 7: Relevancy percentages for joint venture signature patterns

The signatures also represented several expressions that did not contain any words that are strongly associated with joint ventures individually, but *are* strongly associated with joint ventures collectively. For example, the phrases "form company", "set up company", "join forces", and "agreed to form" all achieved 100% precision. Consequently, the relevancy signatures algorithm performed much better than the word-based algorithm. However, the additional context used by the augmented relevancy signatures algorithm and the case-based algorithm did not greatly improve performance. For this classification task, it doesn't matter *who* the partners are, as long as partners exist.

Figure 8 shows the results for the microelectronics domain on a corpus of 500 texts of which 57% were relevant.[5] We did not evaluate the augmented relevancy signatures algorithm and the case-based algorithm on this domain because we did not have a semantic feature dictionary for microelectronics. In contrast to the previous experiments, the word-based algorithm outperformed the relevancy signatures algorithm in the microelectronics domain. In microelectronics, and presumably other technical domains, specialized words and jargon play a crucial role in representing the meaning of a text. For example, words like "multichip", "megabit", and "bipolar" are strongly indicative of microelectronics and performed very well as keywords. Furthermore, technical jargon is usually unambiguous and self-contained; that is, other words surrounding them do not usually change their meaning. As a result, many microelectronics concepts can be recognized by looking for individual words.

[5]According to the MUC-5 guidelines. The texts were part of the MUC-5 corpus [MUC-5 Proceedings, 1993].

Conclusions

Traditional information retrieval systems (e.g., [Turtle and Croft, 1991; Salton, 1971]) and text classification systems [Maron, 1961; Borko and Bernick, 1963; Hoyle, 1973] use isolated words or phrases to classify texts, but some classification tasks can benefit from using additional linguistic information. Our experiments indicate that both the domain and the task influence the kinds of knowledge that a system needs. In both the terrorism and joint ventures domains, the IE-based text classification algorithms recognized role objects using simple linguistic patterns, which produced substantially better results than isolated words and phrases. Furthermore, semantic information associated with role objects must be represented if the task places semantic constraints on the domain. For example, the classification task in the terrorism domain specified that only certain type of perpetrators and victims were relevant. Finally, experiments with the microelectronics domain suggest that isolated words can be effective in technical domains. We conclude that the dictionary requirements of an NLP system are strongly influenced by the domain and task to which it will be applied. Shallow lexical knowledge can produce good results for some problems, but more complex syntactic and semantic information must be represented to achieve good performance on others.

References

Borko, H. and Bernick, M. 1963. Automatic Document Classification. *J. ACM* 10(2):151–162.

Hoyle, W. 1973. Automatic Indexing and Generation of Classification Systems by Algorithm. *Information Storage and Retrieval* 9(4):233–242.

Kim, J. and Moldovan, D. 1993. Acquisition of Semantic Patterns for Information Extraction from Corpora. In *Proceedings of the Ninth IEEE Conference on Artificial Intelligence for Applications*, Los Alamitos, CA. IEEE Computer Society Press. 171–176.

Lehnert, W. 1991. Symbolic/Subsymbolic Sentence Analysis: Exploiting the Best of Two Worlds. In Barnden, J. and Pollack, J., editors 1991, *Advances in Connectionist and Neural Computation Theory, Vol. 1.* Ablex Publishers, Norwood, NJ. 135–164.

Maron, M. 1961. Automatic Indexing: An Experimental Inquiry. *J. ACM* 8:404–417.

Proceedings of the Third Message Understanding Conference (MUC-3), San Mateo, CA. Morgan Kaufmann.

Proceedings of the Fourth Message Understanding Conference (MUC-4), San Mateo, CA. Morgan Kaufmann.

Proceedings of the Fifth Message Understanding Conference (MUC-5), San Francisco, CA. Morgan Kaufmann.

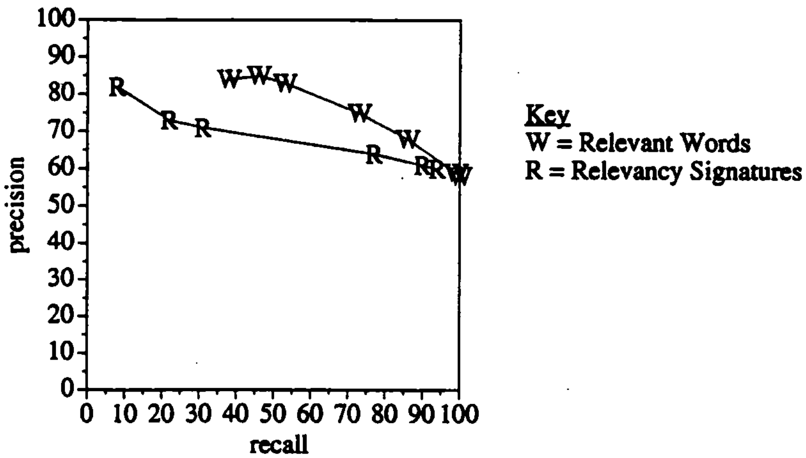

Figure 8: Text classification results for the microelectronics domain

Riloff, E. and Lehnert, W. 1992. Classifying Texts Using Relevancy Signatures. In *Proceedings of the Tenth National Conference on Artificial Intelligence.* AAAI Press/The MIT Press. 329–334.

Riloff, E. and Lehnert, W. 1994. Information Extraction as a Basis for High-Precision Text Classification. *ACM Transactions on Information Systems* 12(3):296–333.

Riloff, E. 1993a. Automatically Constructing a Dictionary for Information Extraction Tasks. In *Proceedings of the Eleventh National Conference on Artificial Intelligence.* AAAI Press/The MIT Press. 811–816.

Riloff, E. 1993b. Using Cases to Represent Context for Text Classification. In *Proceedings of the Second International Conference on Information and Knowledge Management (CIKM-93),* New York, NY. ACM Press. 105–113.

Riloff, E. 1994. *Information Extraction as a Basis for Portable Text Classification Systems.* Ph.D. Dissertation, Department of Computer Science, University of Massachusetts Amherst.

Salton, G., editor 1971. *The SMART Retrieval System: Experiments in Automatic Document Processing.* Prentice Hall, Englewood Cliffs, NJ.

Proceedings of the TIPSTER Text Program (Phase I), San Francisco, CA. Morgan Kaufmann.

Turtle, Howard and Croft, W. Bruce 1991. Efficient Probabilistic Inference for Text Retrieval. In *Proceedings of RIAO 91.* 644–661.