Being *Erlang Shen*: Identifying Answerable Questions

Hong Yu
Columbia University
Department of Biomedical Informatics
622 West, 168th Street, VC-5, NY, NY 10032
yuh9001@dbmi.columbia.edu

Carl Sable
Cooper Union
Department of Electrical and Computer Engineering
51 Astor Place, NY, NY 10003
sable2@cooper.edu

Topics: language processing, reasoning aspects, knowledge representation and integration

Abstract

Research has shown that answers do not exist in biomedical corpora for many questions posed by physicians. We have therefore developed a *question filtering* component that determines whether or not a posed question is answerable. Using 200 clinical questions that have been annotated by physicians to be answerable or unanswerable, we have explored the use of supervised machine-learning algorithms to automatically classify questions into one of these two categories. We also have incorporated semantic features from a large biomedical knowledge terminology. Our results show that incorporating semantic features in general enhances the performance of question classification and the best system is a probabilistic indexing system that achieves an 80.5% accuracy. Our analysis also shows that stop words may play an important role for separating *Answerable* from *Unanswerable*.

1 Introduction

Chinese myth has long portrayed a powerful god *Erlang Shen*, who has a magical third eye in the middle of his forehead that sees truth. The real world mixes truth and falsehood and questions may be answerable or unanswerable. In the field of automatic question answering (QA), most QA systems implicitly assume that all questions are answerable. This study presents what we believe is the first attempt to separate answerable questions from unanswerable ones. We are essentially aiming to create the keen third eye to filter out unanswerable questions. The answerable questions can then be further processed for answer extraction and generation; the unanswerable questions may be further analyzed to determine the user's intentions.

Automatic question answering applies artificial intelligence and natural language processing techniques to extract information from corpora or databases in order to answer a user's question. Since no corpora or databases, no matter how large, can incorporate the entire universe of knowledge, they will not contain answers to certain questions. For example, research (Jacquemart and Zweigenbaum 2003) has found that the largest text collection, the World Wide Web, is not a good source for answering medical, domain-specific questions. On the other hand, biomedical literature and reputed online medical databases are useful for this task (Sackett et al. 2000, Straus and Sackett 1999). However, these same biomedical resources can not answer the question "What is causing her hives?"; this question was posed by a family physician (Ely et al. 2002). This study explores the use of supervised machine-learning approaches to automatically identify whether or not a question is answerable using biomedical corpora and databases.

Determining whether or not a question is answerable is a first step towards question answering. A question answering system needs first to identify a user's intentions, and then to generate a useful answer. Previously, studies have proposed models to offer explanations for failed queries or the results of the queries that are "unknown" (Chalupsky and Russ 2002). In this application, when a question is not answerable, the question answering system may further evaluate the question. For example, if the unanswerable question is not related to the medical domain, a system might return the question to user and provide the justification that the system only handles medical questions. If the unanswerable question is ambiguous, a system could use disambiguation to generate a list of non-ambiguous questions, from which the user can identify one or more according to his/her intentions. The efforts on identifying a user's intentions have been addresses in earlier work (Chalupsky and Russ 2002, Harabagiu et al. 2004, Gaasterland et. al. 1994, Grice 1975).

2 Related Work

Research on identifying a user's intentions starts with maxims of cooperative conversation (Grice 1975). A review is given by (Gaasterland et. al. 1994), who have analyzed cooperative answering as a specific application of Grice's maxims of cooperative conversation. According to these maxims, answers (and other contributions to a conversation) should not only be correct, but in addition, they should be useful, they should not be misleading, and they should not contain too much information. The overview provided by Gaasterland and his colleagues discusses how these maxims might be applied to query/answer systems, which they define to include not only question answering systems as defined in this paper, but also database systems and deductive databases that accept logical queries.

One interesting discussion in the work of Gaasterland and his colleagues involve general categories of reasons that a query or question might fail to have an answer. For example, the wording of a question might contain a false presupposition. An example in the medical domain might be, "What drug can fight the disease blindness?" The response "None" would be incorrect, since it seems to validate the false presupposition that blindness is a disease. A good response might be "Blindness is not a disease." Another interesting case involves questions with misconceptions, which are more general than false presuppositions. Questions with misconceptions can have correct answers that are still misleading. An example in the medical domain might be "What drug can a therapist prescribe to fight depression?" In this case, the answer "None" would technically be correct but misleading; a better response would be "Therapists can not prescribe drugs."

Chalupsky and Russ (2002) propose to provide a list of plausible answers or explanations when exact answers cannot be found in a database in response to a user's query. Possible explanations deal with missing knowledge, limitations of resources, user misconceptions, and bugs in the system. Chalupsky and Russ have created a system called *WhyNot*, which accepts queries to the general knowledge base *Cyc*, and attempts to provide what they call partial proofs for failed queries. An example provided by the authors involves the question, "Is it true that anthrax lethally infects animals?" The answer, according to the system, is unknown, but WhyNot also determines that the answer would be known if an animal is a kind of mammal. *WhyNot* was built on a relational database and does not handle ad hoc questions.

Harabagiu and her associates (2004) have proposed methods to combine semantic and syntactic features for identifying a user's intentions. As stated in their paper, if a user asks "Will Primer Minister Mori survive the crisis?", the method detects the user's belief that the position of the Prime Minister is in jeopardy, since the concept DANGER is associated with the words "survive" and "crisis". In addition, they propose that the predicate-argument structures of a question can be used to coerce a user's intention when there exist questions with known intentions. The work discussed in (Harabagiu et al. 2004) derives intentions only from the questions, and does not involve human-computer dialogue.

Many research groups have developed either rule-based (Hughes 1986) or machine-learning approaches (Hermjakob 2001, Zhang and Lee 2003) to automatically classify questions into predefined question types (e.g., definitional questions such as "What is X"?) for the purpose of answer generation. However, they all assume that all questions can be answered. Our study presents a different dimension that demonstrates that not all questions can be answered, and that unanswerable questions can be automatically identified.

This study is a part of our ongoing effort involving the development of a domain-specific QA system, BioMedQA, which will automatically generate answers to questions posed by physicians and biomedical researchers. In the following sections, we first describe QA in general, as well as particular considerations relevant to the development of a domain-specific QA system. Next we describe our question collection and our approaches of classifying questions as *Answerable* or *Unanswerable*. We then present and evaluate our results. We close our paper with discussion, conclusions, and future work.

3 Question Answering

Question answering is an advanced form of information retrieval in which focused answers are generated for either user queries or ad hoc questions. Most research development in the area is in the context of open-domain, collection-based or web-based QA. Largely driven by the Text REtrieval Conference (TREC) QA track[1], technologies have been developed for generating short answers to factual questions (e.g., “Who is the president of the United States?”). Recently, the Advanced Research and Development Activity (ARDA)’s Advanced Question & Answering for Intelligence (AQUAINT) program[2] has supported QA techniques that generate long answers for scenario questions (e.g., opinion questions such as “What does X think about Y?” (Yu and Hatzivassiloglou 2003)). Most QA systems leverage techniques from several fields including *information retrieval* (Rigsbergen 1979), which generates query terms relevant to a question and selects documents that are likely candidates to contain answers; *information extraction,* which locates portions of a document (e.g., phrases, sentences, or paragraphs) that contain the specific answers; and *summarization* and *natural language generation*, which are used to generate coherent, readable answers.

Recently there has been growing interest in domain-specific question answering. For example, ACL 2004 dedicated a workshop to QA within restricted domains. Domain-specific, biomedical QA can differ from open-domain QA in at least two important ways. For one, it might be possible to have a list of question types that are likely to occur, and separate answer strategies might be developed for each one. Secondly, domain-specific resources such as knowledge bases and tools exist with a level of detail that might allow a deeper processing of questions than is not possible for open-domain questions.

4 Question Collection and Annotation

Ely and his colleagues (Ely et al. 1999, Ely et al. 2000) have collected thousands of clinical questions from more than one hundred family doctors. They have excluded requests for facts that could be obtained from the medical records (e.g., “What was her blood potassium concentration?”) or from the patient (e.g., “How long have you been coughing?”). The National Library of Medicine has made available a total of 4,653 clinical questions[3] over different studies (Alper et al. 2001, D'Alessandro et al. 2004, Ely et al. 1999, Ely et al. 2000, Gorman et al. 1994, Niu et al. 2003).

Although physicians tend to ask many questions when caring for patients, studies have found that many physicians cannot find satisfactory answers for their questions. Ely and his colleagues have identified 59 obstacles that prevent physicians from finding answers to some of those questions (Ely et al. 2002). They found that the most common class of obstacle preventing physicians from getting answers to their clinical questions is that the information resources do not always contain the answers. For example, biomedical information resources can not answer non-clinical questions such as “How do you stop somebody with five problems, when their appointment is only long enough for one?”

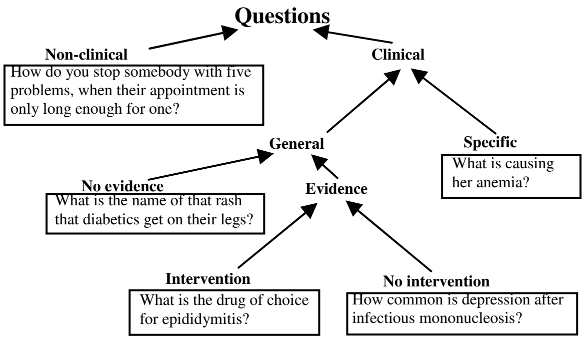
Questions
Non-clinical
How do you stop somebody with five problems, when their appointment is only long enough for one?
Clinical
General
Specific
What is causing her anemia?
No evidence
What is the name of that rash that diabetics get on their legs?
Evidence
Intervention
What is the drug of choice for epididymitis?
No intervention
How common is depression after infectious mononucleosis?

Figure 1: “Evidence taxonomy” created by Ely and his colleagues (Ely et al. 2002) with examples.

In addition, in the medical domain, physicians are urged to practice *Evidence Based Medicine* when faced with questions about how to care for their patients (Gorman et al. 1994, Straus and Sackett 1999, Bergus et al. 2000). Evidence based medicine refers to the use of the best evidence from scientific and medical research to make decisions about the care of individual patients. The needs of evidence based medicine have also driven biomedical researchers to provide evidence in their research reports. With this in mind, Ely and his colleagues have created an “evidence taxonomy” to organize medical questions into five hierarchical categories (shown in Figure 1).

In addition, Ely and his colleagues have manually annotated 200 clinical questions, placing them into the five leaf categories shown in Figure 1. Those 200 questions were randomly selected from the thousands that they collected (Ely et al. 2002). After searching for answers to these questions in biomedical literature and online medical databases, Ely and his colleagues have concluded that the *Non-clinical*, *Specific,* and *Non-evidence* questions are not answerable, while both subcategories of *Evidence* (i.e., *Intervention* and *No-intervention* questions) are potentially answerable with evidence. *Non-clinical* questions do not deal with the specific domain, *Specific* questions require information from a patient’s record, and *Non-evidence* questions are questions for which the answer is generally unknown. This results in a total of 83 unanswerable questions and 117 answerable questions. These 200 questions have been used in our

[1] http://trec.nist.gov/

[2] http://www.informedia.cs.cmu.edu/aquaint/

[3] http://clinques.nlm.nih.gov/JitSearch.html

study to automatically classify a question as either *Answerable* or *Unanswerable*.

5 Supervised Machine-Learning

Separating *Answerable* from *Unanswerable* is a task of document categorization. We have explored supervised machine-learning approaches to automatically classify a question into one of these two categories. In the following subsections, we will describe the machine-learning systems, the learning features, the cross-validation methodology, and the evaluation metrics used for our classification.

5.1 Systems

We have applied seven text categorization systems using a variety of approaches. Five of the seven systems comprise the publicly available Rainbow package (McCallum 1996). The approaches used by these systems are Rocchio/TF*IDF, K-nearest neighbors (kNN), maximum entropy, probabilistic indexing, and naïve Bayes. All of these approaches have been used successfully for text categorization tasks (Sebastiani 2002). We have also applied support vector machines[4] because it has shown to be successful for text categorization tasks (Yang and Liu 1999, Sebastiani 2002). Additionally, we have explored the machine-learning system, BINS (Sable and Church 2001), which is a generalization of Naive Bayes. Brief descriptions of these approaches are given in the following subsections; see (Sable 2003) for more detailed descriptions of these machine-learning algorithms.

Rocchio/TF*IDF

A Rocchio/TF*IDF system (Rocchio 1971) adopts TF*IDF, the vector space model typically used for information retrieval, for text categorization tasks. Rocchio/TF*IDF represents every document and category as a normalized vector of TF*IDF values. The term frequency (TF) of a token (typically a word) is the number of times that the token appears in the document or category, and the inverse document frequency (IDF) of a token is a measure of the token's rarity (usually calculated based on the training set). For test documents, scores are assigned to each potential category by computing the similarity between the document to be labeled and the category, often computed to be the cosine measure between the document vector and the category vector; the category with the highest score is than chosen.

K-Nearest Neighbors (kNN)

A K-nearest neighbors system determines which training questions are the most similar to each test question, and then uses the known labels of these similar training questions to predict a label for the test question. The similarity between two questions can be computed as the number of overlapping features between them, as the inverse of the Euclidean Distance between feature vectors, or according to some other measure. The kNN approach has been successfully applied to a variety of text categorization tasks (Sebastiani 2002, Yang and Liu 1999).

Naïve Bayes

The naïve Bayes approach is commonly used for machine learning and text categorization tasks. Naïve Bayes is based on Bayes' Law and assumes conditional independence of features. For text categorization, this “naive” assumption amounts to the assumption that the probability of seeing one word in a document is independent of the probability of seeing any other word in a document, given a specific category. Although this is clearly not true in reality, Naive Bayes has been useful for many text classification and other information retrieval tasks (Lewis 1998). The label of a question is the category that has the highest probability given the “bag of words” in the document. To be computationally feasible, log likelihood is generally maximized instead of probability.

Probabilistic Indexing

This is another probabilistic approach that chooses the category with the maximum probability given the words in a document. Probabilistic indexing stems from Fuhr’s probabilistic indexing paradigm (Fuhr 1988), which was originally intended for relevance feedback and was generalized for text categorization by Joachims (Joachims 1997), who considered it a probabilistic version of a TF*IDF classifier, although it more closely resembles naïve Bayes. Unlike naïve Bayes, the number of times that a word occurs in a document comes into play, because the probability of choosing each specific word, if a word were to be randomly selected from the document in question, is used in the probabilistic calculation. Although this approach is less common in the text categorization literature, one author of this paper has seen that it is very competitive for many text categorization tasks (Sable 2003).

Maximum Entropy

This is another probabilistic approach that has been successfully applied to text categorization (Nigam et. al. 1999). A maximum entropy system starts with the initial assumption that all categories are equally likely. It then iterates through a process known as improved iterative scaling that updates the estimated probabilities until some stopping criterion is met. After the process is complete, the category with the highest probability is selected.

[4] We have applied Libsvm, which is available at http://www.csie.ntu.edu.tw/~cjlin/libsvm/

Support Vector Machines (SVMs)

A support vector machine system is a binary classifier that learns a hyperplane in a feature space that acts as an optimal linear separator which separates (or nearly separates) a set of positive examples from a set of negative examples with the maximum possible margin (the margin is defined as the distance from the hyperplane to the closest of the positive and negative examples). SVMs have been widely tested to be one of the best machine-learning classifiers, and previous studies have shown that SVMs outperform other machine learning algorithms for open-domain sentence classification (Zhang and Lee 2003) and other text categorization tasks (Yang and Liu 1999, Sebastiani 2002).

BINS

The BINS system (Sable and Church 2001) uses a generalization of Naive Bayes. BINS places words that share common features into a single bin. Estimated probabilities of a token appearing in a document of a specific category are then calculated for bins instead of individual words, and this acts as a method of smoothing which can be especially important for words with scarce evidence. BINS has proven to be very competitive for many text categorization tasks (Sable 2003, Yu and Sable 2005).

5.2 Learning Features

We have explored bag of words as learning features. Since our collection consists of biomedical, domain-specific questions, we have also incorporated concepts and semantic types from the largest biomedical knowledge resource Unified Medical Language System (UMLS), as additional learning features for question classification. Including the UMLS features represents a method of *class-based* smoothing (Resnik, 1993) where the probabilities of individual or sparse words are smoothed by the probabilities of larger or less sparse semantic classes. In the following subsection, we will describe UMLS concepts and semantic types.

5.3 The Unified Medical Language System

The National Library of Medicine (NLM) has created the Unified Medical Language System (UMLS)[5] (Humphreys and Lindberg 1993) to aid in the development of computer systems that process text in the biomedical domain. The UMLS includes the Metathesaurus, a large database that incorporates more than one million biomedical concepts plus synonyms and concept relations. For example, the UMLS links the following synonymous terms as a single concept: *Achondroplasia*, *Chondrodystrophia, Chondrodystrophia fetalis*, and *Osteosclerosis congenita.*

The UMLS also consists of the Semantic Network, which contains 135 semantic types; each semantic type represents a more general category to which certain specific UMLS concepts can be mapped via is-a relationships (e.g., *Pharmacologic Substance*). The Semantic Network also describes a total of 54 types of semantic relationships (e.g., hierarchical *is-a* and *part-of* relationships). Each specific UMLS concept in the Metathesaurus is assigned one or more semantic types. For example, *Arthritis* is assigned to one semantic type, *Disease or Syndrome*; *Achondroplasia* is assigned to two semantic types, *Disease or Syndrome* and *Congenital Abnormality*.

The National Library of Medicine makes available MMTx[6], a programming implementation of MetaMap (Aronson 2001), which maps free text to UMLS concepts and their associated semantic types. The MMTx program first parses text, separating the text into noun phrases. Each noun phrase is then mapped to a set of possible UMLS concepts, taking into account spelling and morphological variations, and each concept is weighted, with the highest weight representing the most likely mapped concept. The UMLS concepts are then mapped to semantic types according to definitive rules as described in the previous paragraph. MMTx can be used either as a standalone application or as an API that allows systems to incorporate its functionality. In our study, we have applied MMTx to map terms in a question to appropriate UMLS concepts and semantic types; we have added the resulting concepts and semantic types as additional features for question classification.

5.4 Cross-Validation

To evaluate the performance of each system, we have performed four-fold cross-validation. Specifically, we have randomly divided our corpus into four subsets of 50 questions each for four-fold cross-validation experiments; i.e., we train on 150 questions and test on the other 50, and perform four such experiments with each of the text-categorization system that we have tested. We have performed these experiments using bag of words alone as well as bag of words plus combinations of the other features discussed in the previous subsection.

5.5 Evaluation Metrics

Results are reported according to two metrics. The first metric is overall accuracy, which is simply the percentage of questions that are categorized correctly (i.e., they are correctly labeled as *Answerable* or *Unanswerable*). A simple baseline system that

[5] http://www.nlm.nih.gov/research/umls/

[6] http://mmtx.nlm.nih.gov/

automatically categorizes all questions as *Answerable* (something that most automatic QA systems assume anyway) would achieve an overall accuracy of 117/200 = 58.5%.

The second metric is the F1 measure (Rigsbergen 1979) for the *Answerable* category. The F1 measure combines the precision (P) for the category (the number of documents correctly placed in the category divided by the total number of document placed in the category) with the recall (R) for the category (the number of documents correctly placed in the category divided by the number of documents that actually belong to the category). The metric is calculated as F1 = (2 * P * R) / (P + R); the result is always in between the precision and the recall but closer to the lower of the two, thus requiring a good precision and recall in order to achieve a good F1 measure.

6 Results

Since we have applied MMTx for identifying appropriate UMLS concepts and semantic types for each question, which are then included as features for question classification, we have evaluated the precision of MMTx for this task. One of the authors (Dr. Carl Sable) has manually examined the 200 questions comprising our corpus as described in Section 3. MMTx assigns 769 UMLS Concepts and 924 semantic types to the 200 questions (remember that some UMLS concepts are mapped to more than one semantic type, as described in Section 5.3). Our analysis has indicated that 164 of the UMLS Concept labels and 194 of the semantic type labels are wrong; this indicates precisions of 78.7% and 79.0%, respectively. An example of a case that MMTx gets wrong is the abbreviation "pt", which, in this corpus, is often used as an abbreviation for "patient"; MMTx typically assigns this to the UMLS concept *pint* and the semantic type *Quantitative Concept.* Note that manually estimating the recall of MMTx would be difficult, since it would require an expert that is familiar with all possible UMLS concepts and the ways to express them.

We have compared the performance of the machine-learning systems specified in Section 5.1 used to label questions as *Answerable* or *Unanswerable* with feature combinations described in Sections 5.2 and 5.3. Table 1 shows the results of all systems tested using the cross-validation procedure explained in Section 5.4. For four of the six feature combinations, the system that achieves the best performance is the Probabilistic Indexing system; the overall accuracy is as high as 80.5% and the F1 measure for the *Answerable* category is as high as 83.0%. We have also found that incorporating UMLS concepts or semantic types often improves performance compared to using bag-of-words only.

Table 2 lists six questions that are predicted incorrectly by the best machine-learning classifier (i.e., probabilistic indexing with bag-of-words and UMLS concepts as features). Questions are presented exactly as they were expressed by physicians, including bad grammar and incorrect spellings. Since we can not control what physicians will type, these represent complexities that will have to be dealt with by a real-world system.

Answerable: 1) What is best time to get OB ultrasound for dating and to see other things? 2) What are long-term options for hemorrhagic gastritis beyond H2 blockers? 3) Does Zoloft cause stomach upset?
Unanswerable: 4) What is the cause of this patient's tremor? 5) What dose the HMO formulary say I can use for this patient's nasal condition? 6) How long shall I treat knee injury w conservative measures before referring?

Table 2: Three *Answerable* and three *Unanswerable* questions that the classifier predicts incorrectly.

In order to examine useful features for the classification, we have calculated log likelihood ratios of word occurrences in each of our two categories (i.e., *Answerable* and *Unanswerable*). For each word/category pair, the level of indication of the word for the category is computed as the log likelihood of seeing the word in a question of the specified category minus the log likelihood of seeing the word in the other category. Thus, the strength of the word for a category will only be positive if it is the more likely category of the two, given the word, and the magnitude of the strength will depend on the likelihood of the other category. For each question, the strength of all words in the question have been computed for both categories based on evidence from the other questions (one category will have a positive strength and the other category will have a negative strength for each word), and the top words for both categories have been examined. For example, consider the following *Answerable* question:

"How soon should you ambulate a patient with a deep vein thrombosis?"

The top three words indicating the *Answerable* and *Unanswerable* categories, with scores calculated as described above (higher scores representing stronger indications of a category), are:

Answerable: you (1.8), should (1.0), how (0.5)
Unanswerable: a (1.6), patient (0.2), with (-0.2)

ML Approach	Performance Using Features (C means UMLS Concepts, ST means semantic types)					
	Bag of Words	Words+C	Words+ST	Words+C+ST	C only	ST only
*Rocchio/TF*IDF	74.0 (77.4)	72.5 (75.8)	74.5 (77.5)	74.0 (77.2)	67.6 (70.3)	65.0 (68.5)
*kNN	68.5 (71.7)	69.0 (73.5)	65.5 (69.9)	65.5 (70.1)	65.0 (66.0)	61.5 (61.6)
*MaxEnt	66.0 (69.6)	68.0 (73.1)	70.5 (76.1)	69.5 (74.9)	65.0 (67.6)	65.5 (70.9)
*Prob Indexing	78.0 (81.7)	80.5 (83.0)	80.0 (82.9)	79.0 (82.1)	70.0 (70.8)	66.5 (70.0)
*Naïve Bayes	68.0 (74.8)	74.5 (77.9)	73.5 (77.6)	73.0 (76.7)	71.0 (76.0)	64.0 (69.2)
**SVMs	67.5 (74.9)	68.0 (74.6)	69.0 (75.4)	67.0 (73.6)	62.5 (70.1)	67.0 (69.8)
BINS	72.0 (74.5)	72.0 (75.2)	68.5 (72.2)	66.5 (69.1)	66.0 (70.7)	58.5 (64.4)

Table 1: Percentages for overall accuracy and F1 scores (in parentheses) of machine-learning systems with different combinations of learning features for classifying *Answerable* versus *Unanswerable* biomedical questions.

"*" indicates Rainbow implementation
"**" indicates libsvm implementation

Note that the word "with" has a negative weight; this means that it is really an indicator of an *Answerable* question. So this question contains only two words that are indicative of an *Unanswerable* question. Note that the words "ambulate" and "thrombosis" are infrequent and do not show up in either list; it is likely that these words do not occur in any other question, in which case there is no evidence for them and their scores would be 0 for both categories.

We have observed that many stop words have high scores and we have therefore hypothesized that stop words may play an important role for this classification task. Studies have found that for some non-content based categorization tasks, stop words, have proven to be useful; one example is authorship attribution (Mosteller and Wallace 1963). Table 3 shows the change in classification performance when we remove the stop words from the questions. (These results have only been computed for the Rainbow systems, which provide a simple mechanism to do this.) Our results show that when we exclude stop words, this tends to decrease performance, and in particular this is true for the naïve Bayes and probabilistic indexing systems. These results provide evidence that stop words may play an important role for classifying a question posed by a physician as either *Answerable* or *Unanswerable*.

7 Discussion

Based on overall accuracy results, all systems beat random guessing (50.0%) and the simple baseline system that is described in Section 5.5. (58.5%). Furthermore, the F1 measure for the *Answerable* category is higher than the overall accuracy for each system; this indicates that all systems have a slight disposition towards the *Answerable* category (based on the training documents). Compared to typical text categorization tasks, our task is more challenging because our data set is small (only 150 short questions are used for training at one time) which leads to a small feature space. Nevertheless, most systems achieve reasonable performance with several feature combinations, and the probabilistic indexing system achieves and overall accuracy that is up to 22.0% higher than the simple baseline system.

A manual inspection of the questions that are classified incorrectly reveals the problem of data sparseness; for most of these questions, the majority of words do not occur in any other question in the data set. For example, in Table 2, the word "hemorrhagic" in question 2 does not appear in any other question. We speculate that a larger training set could potentially alleviate this problem and boost our results. We have also found that some questions may be mislabeled. For example, the question "Would it be better to put her on a potassium sparing diuretic or just potassium" has been labeled as *Answerable*, but it seems to us that this is a patient-specific question that should be labeled as *Unanswerble.*

Our results show a moderate increase of performance when including the UMLS features. We have observed that many UMLS concepts in these questions, when labeled correctly, represent information that was already present in the bag of words representation. We also found that some semantic types tend to be very general, appearing in both *Answerable* and *Unanswerable* questions commonly. For example, the semantic type *Disease or Syndrome* occurs 44 times in 37 *Answerable* questions and 30 times in 26 *Unanswerable* questions. Therefore, these tokens will not play an important role for classification. However, we believe that this same information will be indispensable for potential future work discussed in Section 8.

8 Conclusions and Future Work

This paper describes what we believe is the first attempt in the field of question answering to automatically identify answerable questions, i.e., the questions for which answers can be found in the

ML Approach	Performance Difference when Stop Words are Excluded			
	Bag of Words	Words+C	Words+ST	Words+C+ST
*Rocchio/TF*IDF	-3.0 (-3.1)	-6.5 (-6.4)	-5.5 (-4.2)	-4.5 (-3.4)
*kNN	+1.5 (+1.4)	-1.0 (-2.1)	-1.5 (-1.2)	-3.0 (-3.1)
*MaxEnt	+0.5 (-2.2)	-7.5 (-7.9)	-2.5 (-1.5)	-2.0 (-0.8)
*Prob Indexing	-3.0 (-4.4)	-6.5 (-7.5)	-7.5 (-6.7)	-4.0 (-3.5)
*Naïve Bayes	-6.0 (-3.7)	-9.5 (-7.8)	-5.0 (-5.4)	-6.5 (-7.6)

Table 3: Increase (+) or decrease (-) of overall accuracy and F1 scores (in parentheses) after we remove stop words for classifying *Answerable* versus *Unanswerable* biomedical questions.

"*" indicates Rainbow implementation

available corpora. Our results are promising; the best system achieves an 80.5% overall accuracy for separating *Answerable* from *Unanswerable* questions based on a small training set. We consider this result to represent an important proof-of-concept. In the future, we expect that biomedical QA systems such as BioMedQA will be able to accurately distinguish *Answerable* from *Unanswerable* questions relying on more advanced processing described in the following paragraph.

We believe that it will eventually be possible to automatically decompose biomedical questions according to component question types, which are described in (Ely et al. 1999); for example, "What are the affects of <*drug*> on <*disease*>?". We speculate that the recognition of UMLS concepts and semantic types using tools such as MMTx will play a key role in this type of question classification. If questions can be accurately mapped to component question types, then the filtering of unanswerable questions will become straight-forward; an even greater benefit will be that specific answer strategies could be developed for each answerable component question type.

Acknowledgements

We thank Dr. John Ely, Dr. Jerry Osheroff, and Dr. James Cimino for valuable discussion and for making their annotation available to us. This project is supported by JDRF 6-2005-835. Any opinions, findings, or recommendations are those of the authors and do not necessarily reflect JDRF's views.

References

Allen, J.F. and C.R. Perrault. 1986. Analyzing intention in utterances. In B.J. Grosz, K.S. Jones, and B.L. Weber, editors, Readings in Natural Language Processing, Pages 441-458. Morgan Kaufmann Publishers, Inc., Los Altos, California, 1986.

Alper, B., J. Stevermer, D. White, and B. Ewigman. 2001. Answering family physicians' clinical questions using electronic medical databases. *J Fam Pract* 50: 960-965.

Aronson, A. 2001. Effective Mapping of Biomedical Text to the UMLS Metathesaurus: The MetaMap Program. *American Medical Information Association*.

Bergus, G.R., Randall, C.S., Sinift, S.D. and D.M. Rosenthal. 2000. Does the structure of clinical questions affect the outcome of curbside consultations with specialty colleagues? Arch Fam Med. 9(6): 541-7.

Chalupsky, H. and T.A. Russ. 2002. WhyNot: Debugging Failed Queries in Large Knowledge Bases. In Proceedings of the fourteenth innovative applications of artificial intelligence, pages 870-877, AAAI Press.

D'Alessandro, D.M., Kreiter, C.D., and M.W. Peterson. 2004. An evaluation of information seeking behaviors of general pediatricians. Pediatrics 113: 64-69.

Ely, J., J. Osheroff, M. Ebell, G. Bergus, B. Levy , M. Chambliss, and E. Evans. 1999. Analysis of questions asked by family doctors regarding patient care. *BMJl*: 358-361.

Ely, J., J. Osheroff, M. Ebell, M. Chambliss, D. Vinson, J. Stevermer, and E. Pifer. 2002. Obstacles to answering doctors' questions about patient care with evidence: qualitative study. *BMJ* 324: 710-713.

Ely, J., J. Osheroff, P. Gorman, M. Ebell, M. Chambliss, E. Pifer, and P. Stavri. 2000. A taxonomy of generic clinical questions: clasification study. *BMJ* 321: 429-432.

Fuhr, N. 1998. Models for retrieval with probabilistic indexing. *Information Processing and Management* 25(1):55-72.

Gaasterland, T., P. Godfrey, and J. Minker. 1994. An overview of cooperative answering. In *Nonstandard Queries and Nonstandard Answers*, pages 1-40, Clarendon Press.

Gorman, P., J. Ash, and L. Wykoff. 1994. Can primary care physician's questions be answered using hte medical journal literature? *Bull Med Libr Assoc* 82: 140-146.

Grice, H. 1975. Logic and conversation. In *Syntax and Semantics*, Academic Press.

Harabagiu, S.M., Maiorano, S.J., Moschitti, A, and C.A. Bejan. 2004. Intentions, implicatures and processing of complex questions. In *HLT-NAACL Workshop on Pragmatics of Question Answering*.

Hermjakob, U. 2001. Parsing and question classification for question answering. In *Proceedings of ACL Workshop on Open-Domain Question Answering*.

Hovy, E., Gerber, L., Hermjakob, U., Junk, M., and C.Y. Lin. Question answering in Webclopedia. In *Proceedings of the TREC-9 Conference*.

Hughes, S. 1986. Question classification in rule-based systems. In *Annual Technical Conference of the British Computer Society Specialist Group on Expert Systems*.

Humphreys, B. L., and D. A. Lindberg. 1993. The UMLS project: making the conceptual connection between users and the information they need. *Bull Med Libr Assoc* 81: 170-7.

Jacquemart, P., and P. Zweigenbaum. 2003. Towards a medical question-answering system: a feasibility study. *Stud Health Technol Inform* 95: 463-8.

Joachims, T. 1997. A probabilistic analysis of the Rocchio algorithm with TFIDF for text categorization. In Proceedings of the 14th International Conference on Machine Learning.

Lewis, D. 1998. Naive (Bayes) at forty: the independence assumption in information retrieval. In Proceedings of the European Conference on Machine Learning.

McCallum, A. 1996. A toolkit for statistical language modeling, text retrieval, classification, and clustering. *http://www.cs.cmu.edu/~mccallum/bow*.

Mosteller, F. and D. Wallace. 1963. Inference in an authorship problem. Journal of the American Statistical Association 58:275-309.

Nigam, K.; Lafferty, J.; and McCallum, A. 1999. Using maximum entropy for text classification. In Proceedings of the IJCAI-99 workshop on machine learning for information filtering.

Niu, Y., G. Hirst, G. McArthur, and P. Rodriguez-Gianolli. 2003. Answering clinical questions with role identification. *ACL workshop on natural language processing in biomedicine*.

Resnik, P. 1993. Selection and information: A class-based approach to lexical relationships. PhD thesis. Department of Computer and Information Science, University of Pennsylvania.

Rigsbergen, V. 1979. *Information Retrieval, 2nd Edition*. Butterworths, London.

Rocchio, J. 1971. Relevance feedback in information retrieval. In The Smart Retrieval System: Experiments in Automatic Document Processing, pages 313-323, Prentice Hall.

Sable, C. 2003. Robust Statistical Techniques for the Categorization of Images Using Associated Text. Columbia University, New York.

Sable, C., and K. Church. 2001. Using Bins to empirically estimate term weights for text categorization. *EMNLP*, Pittsburgh.

Sackett, D., S. Straus, W. Richardson, W. Rosenberg, and R. Haynes. 2000. *Evidence-Based Medicine: How to practice and teach EBM*. Harcourt Publishers Limited, Edinburgh.

Sebastiani, F. 2002. Machine learning in automated text categorization. *ACM Computing Surveys*. 34: 1-47.

Straus, S., and D. Sackett. 1999. Bringing evidence to the point of care. *Journal of the American Medical Association* 281: 1171-1172.

Yang, Y., and X. Liu. 1999. A re-examination of text categorization methods. In Proceedings in the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval.

Yu, H., and V. Hatzivassiloglou. 2003. Towards answering opinion questions: Separating facts from opinions and identifying the polarity of opinion sentences. *EMNLP*.

Yu, H., and C. Sable, and H. R. Zhu. 2005. Classifying medical questions based on an evidence taxonomy. Forthcoming.

Zhang, D. and Lee, WS. 2003. Question classification using support vector machines. In *Proceedings of the 26th Annual International ACM SIGIR conference, pages 26-32.*