Automatic summarization of results from clinical trials

Rodney L. Summerscales*, Shlomo Argamon*, Shangda Bai*, Jordan Hupert† and Alan Schwartz†‡
*Department of Computer Science, Illinois Institute of Technology, Chicago, IL 60616, USA
Email: {rsummers, argamon}@iit.edu
†Department of Pediatrics,‡Department of Medical Education,
University of Illinois at Chicago, Chicago, IL 60612, USA, Email: {jhupert, alansz}@uic.edu

***Abstract*—A central concern in Evidence Based Medicine (EBM) is how to convey research results effectively to practitioners. One important idea is to summarize results by key summary statistics that describe the effectiveness (or lack thereof) of a given intervention, specifically the absolute risk reduction (ARR) and number needed to treat (NNT). Manual summarization is slow and expensive, thus, with the exponential growth of the biomedical research literature, automated solutions are needed.**

In this paper, we present a novel method for automatically creating EBM-oriented summaries from research abstracts of randomly-controlled trials (RCTs). The system extracts descriptions of the treatment groups and outcomes, as well as various associated quantities, and then calculates summary statistics. Results on a hand-annotated corpus of research abstracts show promising, and potentially useful, results.

***Keywords*-information extraction; summarization; medical text processing**

I. Introduction

Evidence Based Medicine (EBM) is the philosophy that physicians should make treatment decisions based on the latest research findings. Unfortunately, it is difficult to keep up with the rapidly growing medical research literature. A classical solution is for teams of medical experts to compile comprehensive reviews of the medical literature on various topics, for example, in the Cochrane Collaboration[1]. Although quite useful, these must be manually researched and continually updated as new research is published.

Another approach is to develop tools to help physicians search databases of the medical literature to better decide on the best current intervention for a particular case. The PICO framework[12] is the most common such approach for clinical queries. A PICO query consists of the Patient/Problem, Intervention, Comparison intervention (when relevant), and clinical Outcome of interest. This framework can help practitioners find plausibly relevant research articles, but determining the true relevance of the research, and how to apply it to a particular case, is still rather difficult.

To alleviate this problem, standardized measures of effectiveness have been developed to help physicians evaluate the likely effects of possible interventions on specific clinical outcomes [8]. These are the *absolute risk reduction* (ARR), the percentage of control patients (those with the standard treatment) who would benefit from taking the new treatment (the experimental treatment), and the *number needed to treat* (NNT) with the new treatment to prevent one bad outcome that would happen with the control. While these statistics are sometimes published, often they are not, and physicians must calculate them directly. Online tools, such as the Risk Reduction Calculator [14], can help, but even so, calculating these statistics can be time-consuming, which limits the usefulness of the approach in practice.

We describe a system that automatically extracts the necessary information and calculates these summary statistics for a given research abstract. This system could be integrated into existing physician support systems and medical information retrieval systems. As far as we know, this is the first attempt to automatically build such summaries.

A. Summary statistics

ARR, first described by Laupacis et al. [8], is the difference between the Control Event Rate (CER) and the Experiment Event Rate (EER), where CER and EER are the rates of bad outcomes for participants in the control and experiment groups, respectively. To calculate ARR for a study, we need to know the number of bad outcomes for the control ($N^{bad}_{control}$) and experimental treatment groups (N^{bad}_{exp}) along with the sizes of the treatment groups ($N_{control}$ and N_{exp}). With this, we can calculate ARR:

$$\text{ARR} = \text{CER} - \text{EER} = \frac{N^{bad}_{control}}{N_{control}} - \frac{N^{bad}_{exp}}{N_{exp}} \qquad (1)$$

Given ARR, we can calculate NNT, the number of people that need to be given the experimental treatment in order to prevent one bad outcome. NNT is simply the reciprocal of ARR, rounded up to the nearest integer.

To calculate these summary stats, we must first find and intepret all the relevant quantities in an abstract or article. In some cases (though not all) the necessary information is found in a single sentence in an abstract such as:

> Mortality was higher in the quinine group than in the artemether group (10/52 v 6/51; relative risk 1.29 , 95% confidence interval 0.84 to 2.01)

From this sentence, we should get the following summary:

[1] http://www.cochrane.org

- Outcome: *Mortality*
- Control: *quinine group*
  - Number of bad outcomes: *10*
  - Number of participants in group: *52*
- Experiment: *artemether group*
  - Number of bad outcomes: *6*
  - Number of participants in group: *51*
- ARR: 7.5% [-6.4%, 21%]
- NNT: 14 [4.7, ∞]

II. Related work

While the specific task that we address here has not been previously addressed, we summarize here related work in biomedical text understanding.

A. Finding clinical entity mentions

Rosario and Hearst [13] developed a probabilistic graphical model for identifying treatments and diseases in sentences from medical texts and classifying their relationships, using orthographic and syntactic features, as well as the Medical Subject Headings (MeSH) hierarchy.

Paek et al. [11] used shallow semantic parsing to identify agent, patient and effect (i.e. treatment, group, and outcome) entities in sentences containing one of five key verbs in the conclusion sections of abstracts of randomized controlled trials. Sentences were parsed into their constituents and a classifier was used to identify the constituents that were arguments for the predicate in the sentence.

Leaman and Gonzalez [9] developed BANNER, a biomedical CRF-based NER system. They applied their system to various publicly available biomedical data sets and achieved good results compared with existing NER systems.

Chowdhury and Lavelli [2] describe a CRF-based NER system for recognizing disease mentions. Their system uses lexical features (e.g. POS tags), orthographic features, token bigrams and trigrams, syntactic dependency features, and dictionary lookup features from the UMLS Metathesaurus.

B. Finding quantities

Until now, quantity finding appears to have been limited to finding the total number of participants in a trial. Demner-Fushman and Lin [3] use a pattern-based approach to find and extract population sizes. Xu et al. [16] developed a method to extract trial sizes as well as subject demographic information from medical abstracts. They use text classification augmented with a Hidden Markov model to identify sentences containing demographics, and then parse the sentences to extract the information. Hansen et al. [4] focus on finding the original number of participants in the trial, before subjects drop out or are allocated to different treatment groups. They use a variety of features to classify integers found in an abstract, selecting the largest candidate as the trial size.

C. Summarizing clinical results

The most similar work to ours is that of Kiritchenko et al. [6]. Their system, ExaCT, is a tool to help human reviewers compile a database of clinical trials and their characteristics. It first automatically identifies text fragments in a journal article that best describe the trial characteristics. A human reviewer then assesses and modifies the selections. The information found by ExaCT consists of 21 different elements describing the trial participants, the interventions assigned to them, the outcomes measured in the trial, and information about the article (e.g. authors, data of publication). However, it does not attempt to extract the number of bad outcomes or to calculate summary statistics. ExaCT uses a sentence classifier to find sentences most likely to contain desired information elements; elements are then extracted from candidate sentences using hand-crafted rules.

III. Methods

Our system automatically calculates summary statistics for an abstract by first identifying treatment group and outcome mentions. Given these, the system labels each integer as a group size, outcome number, or some other value. Group sizes and outcome numbers are then associated with treatment groups and outcomes. Finally, the system identifies the outcome numbers corresponding to the same outcome and calculates its summary statistics.

A. Finding candidate sentences

The system starts by finding sentences likely to contain relevant information. Identifying candidate sentences is a common step when analyzing medical documents [11], [3], [6], [4], [16], significantly reducing the amount of text that must be processed in order to find needed information. This lowers the likelihood of false positives and also enables the use of more time-consuming methods when searching for mentions and quantities. In the current study, we select sentences that contain at least one integer, since group sizes and outcome numbers are always integers. This approach is used by Hansen et al. [4] to look for clinical trial sizes, and guarantees we will not miss any sentences that contain group sizes and/or outcome numbers. Such sentences are also likely to contain treatment group and outcome mentions, as they are needed to identify the numbers in the sentences.

B. Mention finding

Finding mentions referring to treatment groups, outcomes, and times may be viewed as a sort of named entity recognition (NER). The goal of named entity recognition is to automatically identify the sections of a text that name entities such as people, organizations, locations, or specific types of information such as email addresses, dates/times, or monetary values.

While much research has been devoted to finding named entities in biomedical research papers, the focus has been

on identifying the names of genes, proteins, and drugs. Relatively little work has been done on finding treatment groups or outcomes. Recognizing these entities can be quite challenging, since treatments, for instance, may be anything from short drug names to complex phrases such as:

> *conventional coronary artery bypass grafting surgery using cardiopulmonary bypass*

As well, some may only be referred to indirectly, as in:

> *half had additional advice on anxiety management and half* ***did not***

Here, the second treatment, *no additional advice on anxiety management*, is not even explicitly mentioned, but is merely implied.

Finally, another challenge in recognizing treatments and outcomes, in particular, is that they lack common orthographic features such as numbers, special characters (e.g. ':', '-', '@'), or uppercase letters that aid in recognizing entities such as dates, email addresses, or genes/proteins.

In this study, we use a Conditional Random Field (CRF) classifier [7] to find entity mentions. CRFs have been successfully applied to many different natural language segmentation tasks including that of extracting the names of diseases [2], treatments and diseases [9], and treatments, treatment groups, and outcomes [15].

We build on our previous work [15], where we found that the most useful features for determining if a word was part of a treatment or outcome were the word itself, its part of speech, context features (features from neighboring words), and the label from the section of the abstract that the word appears in (where the abstract has section labels). For identifying group mentions, the word itself and its context features were most useful. Word shape features (character n-grams and various binary word shape features), while often used for named entity recognition, did not help in finding treatments, groups, and outcomes.

We use a first-order linear-chain CRF where the label of the current token is partially dependent on the labels of the tokens immediately before and after it. The classifier is trained on a collection of prelabeled tokens. The features used as input to the classifier are:

- Features based on the token itself: the actual token, its POS tag, and if it's inside parentheses;
- Features based on the phrase containing the token:
  - the type of phrase (noun phrase, verb phrase, etc);
  - UMLS semantic type (if any) for the phrase containing the token.
  - whether it is the first or last token in the phrase;
- Features based on the four nearest tokens on each side of the token in question:
  - the tokens themselves and their part of speech tags;
  - semantic tags for each token;
  - whether each token is in the same phrase as the token in question;
- The section (if known) containing the token (e.g. "Intervention", "Results");

UMLS Metathesaurus semantic types for phrases are found using MetaMap [1]. Other semantic tags for words include *people* and *measurements*. The lists for measurement and people words were created manually. Measurement words include common units of measurement (length, volume, weight, etc.) and their abbreviations. People words are words used to refer to groups of people (e.g. *people*, *participants*, *subjects*, *men*, *women*, *children*, etc).

After token classification we apply some simple rules to clean up results and find additional mentions that were missed. First we apply a group label to all tokens in noun phrases that end with the token "group", as this is usually an indicator of a group mention. Then we look for the longest token sequences that match other detected mentions (ignoring order).

For a CRF classifier we used the MALLET SimpleTagger [10]. The OpenNLP [2] tokenizer, part of speech tagger, and chunker were used to segment sentences into words, generate POS tags for each word, and parse sentences into phrases. Since the corpus consists of medical abstracts, we used models trained on the PennBioIE biomedical corpus[3], obtained from the JULIE Lab[4].

C. Finding quantities

Treatment group sizes and outcome numbers are found in similarly. A vector of features is computed for each integer in the text, and a CRF classifier labels each as a *group size*, *outcome number*, or *other* based on these features. Features include:

- Is the integer small (< 5)?
- Features based on the four nearest tokens on each side:
  - the tokens themselves and their part of speech tags;
  - semantic tags for each token;
  - mention labels (as in Section III-B) for each token;
- Whether specific patterns match the occurrence;
- syntactic/semantic context features;
- the abstract section label (if present);

The patterns used were "(n = *INTEGER*)", which often indicates a group size or population size, and "*INTEGER* / *INTEGER*" or "*INTEGER* of *INTEGER*", where the first integer is usually an outcome number and the second is usually a group size.

Syntactic/semantic context features were constructed by first chunking the sentence to a sequence of integers, special tokens ("/", "v", "vs", ";", "(", ")"), group/outcome mentions, and noun/verb phrases (that do contain a group or outcome mentions). Features for an integer are labels (e.g. special token, mention label, phrase label) of the four context items on either side.

[2]http://opennlp.sourceforge.net/
[3]http://bioie.ldc.upenn.edu/
[4]http://www.julielab.de/

D. Computing Summary Statistics

After mentions and quantities have been identified, we need to determine what can be calculated with them.

1) Templates: We approach determining what can be calculated with the detected quantities as an information extraction problem where groups, outcomes, outcome numbers, and summary statistics are viewed as "events". Our task is now to identify all of the relevant information related to each event. To keep track of the information related to each event, we use *templates* with slots for all of the necessary information for the event, as follows:

- *Group templates*: name of the treatment group.
- *Outcome templates*: name of the outcome.
- *Group size templates*: number of people in the group; the relevant group.
- *Outcome number templates*: number of bad outcomes; the relevant outcome; the relevant group;
- *Summary statistic templates*: links to outcome number templates for the experimental and control treatments; group role conflict? (is there uncertainty as to which group is the control and which is the experiment?).

2) Template filling: For each detected group, outcome, group size, or outcome number, the system creates a template, whose slots are filled by finding the group and outcome mentions that should be associated with the group size and outcome numbers. A classifier is used to find the most likely group or outcome for each group size or outcome number. With this approach features are computed for each possible pair of (group size, group), (outcome number, outcome), and (outcome number, group) in a sentence. Three classifiers, one trained for each of the pair types, compute the probability that each pair should be associated in each way. We use the MegaM v0.92 [5] maximum entropy classifier to compute these probabilities.

The features for a given (quantity, mention) pair are:

- Is the mention the closest one to the quantity
- Are other detected mentions/quantities between them?
- Does the mention occur after the quantity?
- The tokens on either side of each element in the pair
- Do both elements appear in similar positions in the sentence? E.g., for a given (size, mention) pair, are they both the first (size, mention) in the sentence?
- Do both elements appear in the same "constituent" in the sentence? The boundaries for this type of constituent are tokens in the set {'v', 'and', 'or', ','} ('v' is a common abbreviation for "versus" in abstracts).

After probabilities are computed for each possible (quantity, mention) pairing of the same type within a sentence (e.g. all possible (outcome number, outcome) pairs in a sentence), quantities and mentions are linked using the following rules, starting with the highest probability pairing and considering pairs in order of descending probability:

- (group size, group): If the group size and group have yet to be associated, link them.
- (outcome number, outcome): If the outcome number is not linked to an outcome, link it to this one.
- (outcome number, group): If the outcome number is not linked to a group, link it to this one.

The system also ensures that outcome numbers and group sizes that appear in the common patterns "*OUTCOME NUMBER* / *GROUP SIZE*" or "*OUTCOME NUMBER* of *GROUP SIZE*" are associated with the same group.

3) Summary statistic templates: Once group, outcome, group size, and outcome number templates are filled, we need to determine if we have enough information to calculate summary statistics. For this we need two outcome number templates (one control and one experiment). For each sentence, we pair outcome number templates that are linked to the same outcome template or to outcomes with identical names. For each pair create a new summary statistic template. If an outcome number template is incomplete because the size of a treatment group is not mentioned in the sentence that contains the outcome number, the system will look in previous sentences to see if a size is mentioned for the treatment group. If multiple sizes are mentioned (subjects may have dropped out), the current system just rejects the outcome number template and does not compute statistics for the outcome.

Currently, if the group name associated with the outcome number contains words/phrases referring to a control group ('control', 'standard care', 'usual care', 'placebo') or to an experimental group ('experiment(al)', 'new treatment/therapy/intervention') it is labeled it as control or experiment respectively and the opposite label is assigned to the other group name. Otherwise, the system reports that group roles are uncertain and calculates the statistics assuming that the group with the lower bad outcome event rate is the experimental group. Statistics are calculated using the formula given in Section I-A.

IV. Results and discussion

We evaluate performance on a sample corpus of 263 British Medical Journal (BMJ) abstracts obtained via PubMed[5]. They describe randomized controlled trials published between 2005 and 2009. Articles not evaluating treatments were ignored. We annotated treatment groups, outcomes, group sizes, and outcome numbers. For longer treatment groups and outcome, there are two boundaries one that defines the minimal string that needs to be recognized to uniquely identify the entity and another that defines the largest string that could be considered acceptable

Table I shows accuracy results for finding and associating mentions and quantities. 10-fold cross-validation was performed over abstracts, and recall, precision, and F-score

[5]http://www.ncbi.nlm.nih.gov/pubmed/

Table I
ACCURACY OF MENTION FINDING, QUANTITY FINDING, AND MENTION-QUANTITY ASSOCIATION

	Current system			Baselines		
	Rec.	**Prec.**	**F**	**Rec.**	**Prec.**	**F**
Groups	0.71	0.82	0.76	0.67	0.84	0.74
Outcomes	0.34	0.56	0.42	0.28	0.61	0.38
Group sizes	0.82	0.77	0.80	0.61	0.72	0.66
Outcome numbers	0.73	0.69	0.71	0.41	0.70	0.52
(number, outcome)	0.62	0.82	0.71	0.62	0.82	0.71
(number, group)	0.86	0.86	0.86	0.69	0.73	0.71
(size, group)	0.88	0.89	0.89	0.93	0.68	0.78

(the harmonic mean of precision and recall) were computed. A mention detected by the system matches an annotated mention if the detected mention contains all the words in the short version and does not contain any words outside the long version of the annotated mention..

As a baseline, we used the biomedical named entity recognizer BANNER[9] that has been shown to be effective at identifying treatment and disease mentions. To boost recall, we used the same post-processing rules for finding additional mentions as in our own mention finder.

We see that our mention finder is more effective at finding both group and outcome mentions than BANNER. BANNER has better precision, but its recall and F-score are noticeably worse. This is not surprising since BANNER does not have any semantic features, such as UMLS semantic types, but instead relies mainly on lexical and orthographic features, which have been shown to be less useful for finding groups and outcomes [15]. Overall, outcomes are harder to detect, because they have more variability.

For group sizes and outcome numbers, we compare our method with a baseline that labels all integers matching the pattern "(n = *INTEGER*)" as group sizes and those matching "*INTEGER* / *INTEGER*", or "*INTEGER* of *INTEGER*" as outcome numbers (first number) and group sizes (second number). Our proposed number finder significantly outperforms the baseline which only uses the patterns. Hence, while the patterns are useful, they are not enough to find most of the quantities that we want. To our knowledge this is the first attempt at extracting outcome numbers and group sizes. As a rough comparison, Hansen et al. [4] achieve an F-score of 0.84 for the simpler task of extracting the total number of trial participants from a corpus of 223 abstracts.

Regarding associating mentions and quantities, we use a baseline that associates a quantity with the nearest detected mention of the appropriate type. In this context, precision is the proportion of quantity-mention associations made by the system that are correct, and recall is the proportion of quantities that were able to be correctly associated with mentions by the system. False misses are quantities that were not associated with any mention, due to the mention finder missing some mentions. We see that the classifier-based approach outperforms the simple baseline except when it comes to associating outcome numbers with outcome mentions. This indicates that additional features, although useful for association in the other cases, are not helpful for associating outcome numbers with outcomes.

Table II
ACCURACY OF CALCULATING SUMMARY STATISTICS FROM SENTENCES

	Recall	**Precision**	**F-score**
Detected data	0.39	0.82	0.53
Annotated data	0.90	0.90	0.90

Table II gives results for finding and calculating summary statistics. Results are given for both detected and annotated mentions and quantities; the latter, in effect, shows how the system would do with perfect mention and quantity finding. As before, 10-fold cross-validation was performed over abstracts, and metrics were computed for individual summary statistics. A summary statistic is considered correct if the outcome numbers and group sizes are associated with the correct group and outcome mentions and, in the case where working with detected mentions, if these mentions match annotated mentions using the previously described matching criteria. If there is an error in any of these pieces of information, the entire statistic is considered incorrect.

Of the 263 abstracts in our corpus, there are a total 59 in which summary statistics can be calculated just from the information given in a single sentence. Overall, the system gets decent precision, at 82%, though recall is only 39%; note that precision is more important, since incorrect results may be more dangerous than missing potentially useful results. We also see that the system performs much better when working with annotated data, as expected, showing that the main area for improvement is in mention detection. The main difficulty appears to lie in extracting outcomes.

In addition to the 59 summary stats that may be calculated from single sentence, there are another 41 summary stats that may be calculated if we obtain the group size from an earlier sentence. When we extended our system system to look in previous sentences for group sizes, we were only able to calculate an additional two summary statistics (both correct). The reason for this is that there is more variation in how group sizes are reported in sentences that do not contain outcome numbers. Demographic information (e.g. how many people were men or women) is often reported in a similar manner to group sizes in these cases, making classification more difficult.

Another challenge is that a particular group can be referred to with various names throughout an abstract. For instance, all of "intensive rehabilitation programme", " rehabilitation", and "control group" may refer to the same treatment group. The system currently considers two group names to refer to the same group only if the mentions consist of the same set of words. This approach is too strict and

should take coreference into account.

The 30 summary statistics found by our system were independently evaluated by authors JH and AS, who are EBM experts. They evaluated the generated statistics and classified each as *correct* (no errors), *qualitatively correct* (contains a minor error, but still useful), or *wrong* (not useful at all). One considered 24 (80%) to be correct, 3 (10%) qualitatively correct, and 3 (10%) wrong; the other author considered 24 (80%) correct, 1 (3%) qualitatively correct, and 5 (17%) wrong. Disagreement arose regarding outcomes that were not the main outcome of interest (e.g. number of people who found their treatment acceptable), and the correctness of detected *per-protocol* results (ignores those who drop out of the trial) when *intention to treat* results (analysis includes those who dropped out) were missed. While there was less agreement on what both considered questionable, 19 (63%) summary stats were considered by both to be fully correct. Thus even the questionable summary stats found by the system may still be useful.

V. Conclusion

We have presented a method for accomplishing the novel task of automatically extracting information and calculating summary statistics from peer-reviewed medical research articles describing randomized controlled trials. Such structured summaries are needed to support effective evidence-based medicine. To our knowledge, this is the first attempt at extracting outcome numbers and associating mentions with quantities for the purpose of calculating summary statistics.

Future work includes improving the detection of outcome mentions, implementing a more sophisticated method for identify mentions that refer to the same treatment group, and developing a method to classify outcome mentions as good or bad. Currently the system does not look for the number of participants that drop out of a study. As this affects the group size calculation in situations where the size is not mentioned in the same sentence as the outcome numbers, we will need to add support for this in the future. Finally, the system needs to identify the type of analysis used when reporting results (intention to treat or per-protocol).

References

[1] A Aronson. Effective mapping of biomedical text to the umls metathesaurus: the metamap program. *Proc. AMIA Symposium*, Jan 2001.

[2] Md. Faisal Mahbub Chowdhury and Alberto Lavelli. Disease mention recognition with specific features. In *Proc. 2010 Workshop on Biomedical Natural Language Processing*, pages 83–90, Uppsala, Sweden, July 2010. Association for Computational Linguistics.

[3] Dina Demner-Fushman and Jimmy Lin. Answering clinical questions with knowledge-based and statistical techniques. *Computational Linguistics*, 33(1), 2007.

[4] Marie J. Hansen, Nana Ø Rasmussen, and Grace Yuet-Chee Chung. Extracting number of trial participants from abstracts of randomized controlled trials. pages 1–5, Jul 2008.

[5] Hal Daumé III. Notes on cg and lm-bfgs optimization of logistic regression. Paper: `http://pub.hal3.name#daume04cg-bfgs`, Implementation: `http://hal3.name/megam/`, 2004.

[6] S Kiritchenko, B de Bruijn, S Carini, J Martin, and I Sim. Exact: automatic extraction of clinical trial characteristics from journal publications. *BMC Med Inform Decis Mak*, 10(1):56, 2010.

[7] John Lafferty, Andrew McCallum, and Fernando Pereira. Conditional random fields: probabilistic models for segmenting and labeling sequence data. In *International Conference on Machine Learning*, 2001.

[8] Andreas Laupacis, David L. Sackett, and Robin S. Roberts. An assessment of clinically useful measures of the consequences of treatment. *New England Journal of Medicine*, 318(26):1728–1733, 1988.

[9] Rober Leaman and Graciela Gonzalez. Banner: An executable survey of advances in biomedical named entity recognition. *Pacific Symposium on Biocomputing*, 13:652–663, 2008.

[10] Andrew Kachites McCallum. Mallet: A machine learning for language toolkit. http://mallet.cs.umass.edu, 2002.

[11] Hyung Paek, Yacov Kogan, Prem Thomas, Seymour Codish, and Michael Krauthammer. Shallow semantic parsing of randomized controlled trial reports. In *AMIA Annual Symp Proc. 2006*, pages 604–608, 2006.

[12] W. Scott Richardson, Mark C. Wilson, Jim Nishikawa, and Robert S. A. Hayward. The well-built clinical question: A key to evidence-based decisions. *ACP Journal Club*, 123:A–12, 1995.

[13] Barbara Rosario and Marti A. Hearst. Classifying semantic relations in bioscience texts. In *Proc. 42nd Annual Meeting on Association for Computational Linguistics (ACL 2004)*, page 430, 2004.

[14] Alan Schwartz. Evidence based medicine (ebm) and decision tools. *MedEdPORTAL*, 2006. Available from: `http://www.aamc.org/mededportal`, ID = 209.

[15] Rodney Summerscales, Shlomo Argamon, Jordan Hupert, and Alan Schwartz. Identifying treatments, groups, and outcomes in medical abstracts. In *The Sixth Midwest Computational Linguistics Colloquium (MCLC 2009)*, 2009.

[16] Rong Xu, Yael Garten, Kaustubh S. Supekar, Amar K. Das, Russ B. Altman, and Alan M. Garber. Extracting subject demographic information from abstracts of randomized clinical trial reports. *MEDINFO 2007*, 2007.