Discourse Parsing of Sociology Dissertation Abstracts Using Decision Tree Induction

Shiyan Ou, Christopher S.G. Khoo, Dion H. Goh, Hui-Ying Heng

Division of Information Studies,
School of Communication & Information
Nanyang Technological University
Singapore, 637718
{pg00096125, assgkhoo, ashlgoh, ps7610453J}@ ntu.edu.sg

Abstract. In this study, we investigated the use of decision tree induction to parse the macro-level discourse structure of sociology dissertation abstracts. We treated discourse parsing as a sentence categorization task. The attributes used in constructing the decision tree models were stemmed words that occurred in at least 35 sentences (out of 3694 sentences in 300 sample abstracts). Sentence location information was also used. The model obtained an accuracy rate of 71.3% when applied to a test sample of 100 abstracts. Another model that made use of information regarding the presence of 31 indicator words in neighboring sentences was also developed. Although this model did not obtain better results, a comparison of the two models suggests that an improvement in the classification of sentences in *problem statement* and *research method* section is possible by combining the models.

1. Introduction

This paper reports our initial effort to develop an automatic method for parsing the discourse structure of sociology dissertation abstracts. In a previous study (Khoo, Ou & Goh, 2002), we determined that the macro-level structure of dissertation abstracts typically has five sections: *background*, *problem statement*, *research method*, *research results*, and *concluding remarks*. All the sentences in the abstracts analyzed could be subsumed under these five categories, although not every abstract contains all of the categories (Khoo, Ou & Goh, 2002).

In this study, we treat discourse parsing as a text categorization problem – assigning each sentence in a dissertation abstract to one of the five predefined sections or categories. Decision tree induction, a machine-learning method, was applied to word tokens found in the abstracts to construct a decision tree model for the categorization purpose. Decisions tree induction was selected primarily because decision tree models are easy to interpret and can be converted to rules that can be incorporated in other computer programs. Other machine-learning methods, such as support vector machines and Bayesian learning, will be explored in the future.

Three models were investigated in this study. Model 1 made use of word tokens found in the sentence itself to categorize the sentence – without considering the context. Model 2 took into consideration the position of the sentence in the abstract. Model 3 made use of indicator words found in other sentences before and after the sentence being categorized.

This study is part of broader study to develop a method for multi-document summarization. Accurate discourse parsing will make it easier to perform automatic multi-document summarization of dissertation abstracts.

2. Previous studies

Discourse structure usually has the form of a tree structure, resulting from the recursive embedding and sequencing of discourse units (Kurohashi & Nagao, 1994). According to Mann & Thompson (1988), a discourse unit has an independent functional integrity, and can be a clause in a sentence, a single sentence, a text segment containing several sentences, or a paragraph. To understand a text, it is important to parse the discourse structure, and identify how discourse units are combined and what kind of relations they have.

A few algorithms for discourse parsing have been proposed. Kurohashi & Nagao (1994) detected discourse structure in scientific and technical text, where each sentence is considered a discourse unit, using three types of surface clues -- clue words, occurrence of identical/synonymous words/phrases, and certain similarity between two sentences. Marcu (1997) constructed his discourse parser based on cue phrases to split text into discourse units (clause or sentence). Le & Abeysinghe (2003) improved Marcu's discoursing system by combining syntactic information, cue phrases and cohesive devices.

There has been an increasing interest in applying machine learning to discourse parsing, including supervised and unsupervised methods. Nomoto & Matsumoto (1998) used C4.5 decision tree induction program to develop a model of sentential dependencies (contextual links and cohesive links between a pair of sentences) for parsing the discourse of news articles from a Japanese economics daily. Marcu (1999) used C4.5 to develop a rhetorical parser to identify the discourse units of unrestricted texts. This kind of learning-based approach produces an impressive result but requires a large training corpus.

Supervised learning requires manual assignment of predefined category labels to the training dataset. An alternative is to derive the set of labels themselves using an unsupervised learning technique (e.g. a clustering algorithm). Marcu & Echihabi (2002) used an unsupervised learning technique to extract pairs of lexical cue words from a very large corpus. However, it is not yet clear whether unsupervised learning methods of categorization are practically useful.

3. Data preparation

A sample of 300 abstracts were obtained systematically from the set of PhD dissertation abstracts indexed under Sociology in the Dissertation Abstracts International database, with a publication year of 2001. The sentences in the abstracts were manually assigned to one of the five predefined categories:

- *Background* -- introduces the general area of the study, explains why it is an important or interesting problem, and refers to other studies related to the current study.
- *Problem Statement* -- includes research objectives, research questions, hypotheses, and the adopted theoretical framework. The expected results are sometimes indicated. Definitions or explanations of concepts are sometimes provided.
- *Research Method* -- outlines how the study was carried out.
- *Research Results* -- reports the results of the data analysis and research conclusion.
- *Concluding Remarks* -- presents recommendations, future work, or implications of the research results.

The 300 sample abstracts were partitioned into a training set of 200 abstracts used to construct the classifier, and a test set of 100 abstracts to evaluate the accuracy of the constructed classifier.

In the manual categorization of sentences, two problems were encountered. Some of the abstracts were found to be unstructured and difficult to code into the five categories. These were mostly descriptive qualitative studies. There were 29 such abstracts in the training set and 16 in the test set. The unstructured abstracts were deleted both from the training and test sets in this study.

Another problem was that some sentences could be said to belong to more than one category. For example, a sentence could state both the research objective and the research method to be used. To simplify the classification problem, each sentence was assigned to only one category, though actually some sentences could arguably be assigned to multiple categories or no category at all.

To prepare the abstracts for the experiments, the abstracts were tokenized and words were stemmed using the Conexor parser (Conexor Functional Dependency Grammar 3.7-User's Manual, 2002). A small stoplist comprising prepositions, articles and auxiliary verbs were used. The word frequency was calculated for each unique word. 1326 words that occurred in at least five sentences were retained in the study, and used as features in the decision-tree construction.

The abstracts were also segmented into sentences using a program, and each sentence was converted into a vector of term weights. Binary weighting is used, which means that a value of "1" is assigned to a word if it occurs in the sentence, "0" otherwise. The dataset is formatted as a table with sentences as rows and words as columns.

4. Decision tree induction method

Decision tree induction is one of the most widely used and practical methods for learning text classifiers from examples. The constructed decision tree is used like a flow chart. Each internal node denotes a test on an attribute, each branch represents an outcome of the test, and leaf notes represent categories. A well-known decision-tree induction program C5.0 (Quinlan, 1993) was used in the study.

The basic procedure for constructing a decision tree consists of two phrases: *tree construction* and *tree pruning*. During tree construction, C5.0 uses a divide-and-conquer process to derive a decision tree. The training set is split based on the attribute that provides the maximum *information gain*. Each subgroup defined by the first split is then split again, usually based on a different attribute, and the process repeats until the subgroups cannot be split any further. To help prevent overtraining, we limit the size of each subgroup to at least 5.

After a decision tree is build, some of the branches may reflect noise or outliers in the training data. Tree pruning attempts to identify and remove such branches. The pruning severity determines the extent to which the generated decision tree will be pruned. A higher value will result in a smaller, more concise tree which is more generalizable to new data, while a lower value will result in a more complex tree which might "overfit" the training data. Therefore, it is important to determine the optimal amount of pruning. In this study, a range of pruning severity is tried and the accuracy of each model constructed with different amounts of pruning is estimated using 10-fold cross-validation.

10-fold cross-validation works by dividing the training data into 10 subsets, and then building 10 decision trees with each subset "held out" in turn. Each of those trees is tested on the hold-out sample, and the average accuracy of the trees on those hold-out samples is used as an estimate of the accuracy of the decision tree built using the whole training set. 10-fold cross-validation thus avoids making use of the testing sample, which can be reserved for the evaluation of the final model.

C5.0 has a special method for improving the accuracy rate called *boosting*. It works by building a series of decision trees in a sequence, each decision tree attempting to improve on the errors made by the previous tree. New cases are classified by applying the whole set of trees and a weighted voting procedure is used to combine the separate predictions into an overall prediction. In this study, boosting was found to contribute little to the accuracy of discourse parsing, and results will not be reported in this paper.

4. Experiments

As mentioned earlier, three models were investigated in this study:

- Model 1 made use of word tokens found in the sentence.
- Model 2 took into consideration the position of the sentence in the abstract.
- Model 3 made use of indicator words found in other sentences before and after the sentence being categorized.

For model 2, the sentence locations are normalized by dividing the sentence number by the total number of sentences in the abstract.

4.1. Model 1 and Model 2 -- words present in the sentence and sentence location

Table 1 gives the estimated accuracy of models 1 and 2 using 10-fold cross validation, for various amounts of pruning, and with and without sentence location. We mentioned earlier that we made use only of words that met the threshold value of 5 – i.e. occurred in at least 5 sentences. We experimented with a range of threshold values. The results for threshold values 5, 10 and 35 are reported in Table 1.

The best result was obtained by Model 2 (using sentence location), using 95% pruning severity and word frequency threshold of 35. The estimated accuracy rate was 64.8%. This is substantially better than the best result obtained by Model 1 (55.2%).

Clearly sentence location is important in identifying which category or section a sentence belongs to. A common sequence for the five categories in a dissertation abstract is: *background* -> *problem statement* -> *research method* -> *research findings* -> *concluding remarks.*

Table 1. Estimated accuracy of Model 1 and Model 2 for various pruning severity and various word frequency threshold values

Word frequency threshold values	Number of words used as attributes	Sentence location as an additional attribute	Pruning Severity				
			80%	85%	90%	95%	99%
35	243	No (Model 1)	54.1	54.1	**55.2**	55.1	54.6
		Yes (Model 2)	64.5	63.5	63.8	**64.8**	63.7
10	888	No (Model 1)	52.6	54.7	55.1	54.1	53.1
		Yes (Model 2)	62.9	63.3	63.7	63.8	63.1
5	1326	No (Model 1)	-	54.4	53.7	53.1	53.9
		Yes(Model 2)	64.5	64.3	63.9	63.6	63.6

* *"-" could not run*

* *The values are estimated accuracy using 10-fold cross validation*

The high word frequency threshold of 35 indicates that only high frequency words are useful for categorizing the sentences. Using a threshold of 35 and a pruning severity of 95%, the decision tree contains 31 words for Model 2 (with sentence location) and 22 words Model 1 (without sentence location).

After building the final decision tree for Model 2, we applied it to the test sample of 100 abstracts. The accuracy rate obtained was 71.3% (see Table 3).

4.2. Model 3 -- indicator words found in neighboring sentences

The dissertation abstract is a continuous discourse with relations between sentences. Sentences before and after the sentence being processed can help to determine the category of the sentence. If the previous sentence is the first sentence in the *research results* section, then the current sentence is likely to be under *research results* as well.

Furthermore, sentences which are easy to classify, because they contain clear indicator words, can be used to help identify the categories of other sentences that do not contain clear indicator words. For example, the *research results* section often begins with a sentence containing clear indicator words, e.g.

- *Results showed that …*
- *The result indicated that …*
- *The analysis revealed that …*
- *The study suggested that …*
- *This study found that …*

Subsequent sentences will amplify on the results but may not contain a clear indicator word.

To test this assumption, we extracted indicator words from the decision tree of Model 2. For each sentence, we then measured the distance between the sentence and the nearest sentence before and after containing each indicator word. Table 2 illustrates this. Sentence 13 in document 4 is being processed. The indicator word "*study*" is found in sentence 4 (9 sentences earlier) as well as in sentence 14 (1 sentence after).

Then, we used these indicator words as additional attributes (distance as the attribute values) to construct Model 3. The evaluation results of Model 2 and Model 3 (based on the 100 test sample) are shown in Table 3. Table 3 shows that Model 3 does not give better results than Model 2.

Table 2. Indicator words in neighboring sentences

Doc_id	Sentence_id	Neighboring sentence_id	Indicator word	Distance
4	13	4	study	-9
4	13	7	analysis	-6
4	13	14	study	1

* Negative means that the sentence containing indicator word is before the sentence being processed.
* Positive means that the sentence containing indicator word is after the sentence being processed.

Table 3: Test results for Model 2 and Model 3 based on the test sample of 100 abstracts

Section	No. of sentences	Model 2 % correctly classified	Model 3 % correctly classified
1	173	71.7%	68.8%
2	183	53.0%	54.1%
3	189	45.0%	43.9%
4	468	90.0%	88.7%

5	29	55.2%	55.2%
Total	1042	**71.3%**	**70.3%**

We compared the categories assigned to sentences by Model 2 and Model 3 (see Table 4). Model 2 and 3 disagreed on 70 (6.7%) of the sentences. Most of the agreements were in section 4 (*research results*) and section 5 (*concluding remarks*) with 96.8% and 100% agreement respectively. Section 2 (*problem statement*) and section 3 (*research method*) each had about 12% disagreement.

We noticed that for the 70 sentences where Model 2 and Model 3 disagreed, most of the sentences were correctly classified by either Model 2 or Model 3 (see Table 5). Out of the 70 cases, Model 2 classified 35 (50.0%) sentences correctly and Model 3 classified 23 (32.9%) correctly. This means that some improvement in categorization can be obtained by combining Model 2 and Model 3 – developing an accurate method to choose between Model 2 and Model 3. This may be useful for section 2 and 3 where Model 2 and Model 3 had an equal number of correct classifications (see Table 5). Interestingly, both models did not have any correct classification for section 5. Developing a method for combining Model 2 and 3 is possible future work.

Table 4: Comparison of categories assigned by Model 2 and Model 3

Correct category	No. of sentences	Agree		Disagree	
		No. of sentences	%	No. of sentences	%
1	173	162	93.6%	11	6.4%
2	183	161	88.0%	22	12.0%
3	189	167	88.4%	22	11.6%
4	468	453	96.8%	15	3.2%
5	29	29	100%	0	0%
Total	1042	972	93.3%	70	6.7%

Table 5: Cases where Model 2 and Model 3 disagree

Correct category	No. of sentences	No. of sentences where Model 2 and Model 3 disagree	Model 2		Model 3	
			No. correct	%	No. correct	%
1	173	11	8	72.7%	3	27.3%
2	183	22	9	40.9%	11	44.5%
3	189	22	10	45.5%	7	31.8%
4	468	15	8	53.3%	2	13.3%
5	29	0	0	0%	0	0%
Total	1042	70	35	50.0%	23	32.9%

5. Conclusion and future work

In this study, we investigated the use of decision tree induction to parse the macro-level discourse structure of sociology dissertation abstracts. We treated discourse parsing as a sentence categorization task. The attributes used in constructing the decision tree models were stemmed words that occurred in at least 35 sentences (out of 3694 sentences in 300 sample abstracts). Sentence location information was also used. The model (Model 2) obtained an accuracy rate of 71.3% when applied to a test sample of 100 abstracts.

We also developed a model (Model 3) that made use of information regarding the presence of 31 indicator words in neighboring sentences. Although Model 3 did not obtain better results, a comparison of Model 2 and Model 3 suggests that an improvement in the classification accuracy for section 2 and 3 (*problem statement* and *research method*) is possible by combining the 2 models.

Possible future work includes developing a method to combine Model 2 and Model 3. We also plan to carry out more in-depth error analysis to determine whether some inference method can be used to improve the categorization. Other machine-learning methods such as support vector machine (SVM) and Bayesian learning will also be investigated. Finally, the manual categorization of the sample abstracts was done by one person. We plan to have two more codings so that inter-indexer consistency can be calculated, and compared with the performance of the automatic categorization.

References

1. Conexor Functional Dependency Grammar 3.7 - User's Manual. (2002). Retrieved August 25, 2003 from http://www.conexor.com/m_syntax.html

2. Khoo, Christopher, Ou, Shiyan, & Goh, Dion. (2002). A hierarchical framework for multi-document summarization of dissertation abstracts. In *Proceedings of the 5th Conference on Asian Digital Libraries (pp.99-110)*. Singapore.

3. Kurohashi, Sadao & Nagao, Makoto. (1994). Automatic detection of discourse structure by checking surface information in sentences. In *Proceedings of the 15th International Conference on Computational Linguistics (COLING--94) (vol. 2, pp.1123-1127)*. Kyoto, Japan.

4. Le, Huong T. & Abeysinghe, Greetha. (2003). A study to improve the efficiency of a discourse parsing system. In *Proceedings of the 4th International Conference on Intelligent Text Processing and Computational Linguistics (CICLing-2003) (pp.356-369)*. Mexico City, Mexico.

5. Mann, W.C. & Thompson, S.A. (1988). Rhetorical Structure Theory: Toward a functional theory of text organization. *Text,* 8(3), 243-281.

6. Marcu, D. (1997). The rhetorical parsing, summarization, and generation of natural language texts. PhD Dissertation, Department of Computer Science, University of Toronto.

7. Marcu, D. (1999). A decision-based approach to rhetorical parsing. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics (ACL-99) (pp.365-372)*. Maryland.

8. Marcu, D. & Echihabi, A. (2002). An unsupervised approach to recognizing discourse relations. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL-2002).* Philadelphia.

9. Nomoto, Tadashi & Matsumoto, Yuji (1998). Discourse parsing: a decision tree approach. In *Proceedings of the 6th Workshop on Very Large Corpora (WVLC-98).* Montreal, Quebec, Canada. Retrieved August 25, 2003, from http://acl.ldc.upenn.edu/W/W98/W98-1125.pdf

10. Quinlan, J.R. (1993). *C4.5: programs for machine learning.* San Mateo: Morgan Kaufmann Publishers.

Discussion Points

61. What does the author mean by classification?

62. Does the research involve creation or implementation of a classification scheme?

63. How does the researcher use classification to improve the automated approach?

64. How do these methods compare to current human-generated approaches to classification?

65. How does the reported research expand our understanding of classification?

66. Does the research suggest an improvement over human-generated classification?

67. What do you think are the most important lessons learned in this research?

68. What do you think are the best practices reported in this research?

69. What would you recommend to the researcher as the next step in this approach?

70. Is there other related research that you would recommend the researcher become acquainted with?