2011 International Conference on Electrical Engineering and Informatics
17-19 July 2011, Bandung, Indonesia

Information Extraction from Scientific Paper Using Rhetorical Classifier

M.L. Khodra[#1], D.H. Widyantoro[#2], E.A. Aziz[*3], and R.T. Bambang[#4]
[#] *School of Electrical Engineering, Bandung Institute of Technology*
Ganesha 10 Bandung, Indonesia
[1] masayu@stei.itb.ac.id
[2] dwi@stei.itb.ac.id
[4] briyanto@lskk.ee.itb.ac.id
[*] *Faculty of Language and Arts, Indonesia University of Education*
Setiabudi 222 Bandung, Indonesia
[3] aminudin@upi.edu

***Abstract*—Time constraints often lead a reader of scientific paper to read only the title and abstract of the paper, but reading these parts is often ineffective. This study aims to extract information automatically in order to help the readers get structured information from a scientific paper. The information extraction is done by rhetorical classification of each sentence in a scientific paper. Rhetoric information is the intention to be conveyed to the reader by the author of the paper. This research used corpus-based approach to build rhetorical classifier. Since there was a lack of rethorical corpus, we constructed our own corpus, which is a collection of sentences that have been labeled with rhetorical information. Each sentence represented as a vector of content, location, citation, and meta-discourses features. This collection of feature vectors is used to build rhetorical classifiers by using machine learning techniques. Experiments were conducted to select the best learning techniques for rhetorical classifier. Training set consists of 7239 labeled sentences, and the testing set consists of 3638 labeled sentences. We used WEKA (Waikato Environment for Knowledge Analysis) and LibSVM libraries. Learning techniques being considered were Naive Bayes, C4.5, Logistic, Multi-Layer Perceptron, PART, Instance-based Learning, and Support Vector Machines (SVM). The best performers are the SVM and Logistic classifier with accuracy of 0.51. By applying one-against-all strategy, the SVM accuracy can be improved to 0.60.**

***Keywords*— rhetorical classifier, information extraction, scientific paper, rhetorical corpus, SVM classifier**

I. INTRODUCTION

Time constraints often lead a reader of scientific paper to read only the title and abstract of the paper. If the title and abstract are considered relevant to the needs of the reader, then the paper will be read in its entirety. This is conducted because there are only few relevant papers [1] of about 50 million of existing papers [2].

Although an abstract can be used to quickly identify the topic of a paper [3], reading only the abstract is often ineffective. First, the different background between authors and readers makes the contents of the paper are difficult to be understood by readers especially if the readers read only the abstract [4]. Second, the level of information between authors and readers may differ. Because the abstract is a summary of the contents of papers that are considered important by the author of the paper, the reader's required information may not be included in the abstract. Last, number of abstracts to be read is equivalent to number of papers. A reader has patience to read only 10 to 30 titles and abstracts of paper at a time [5].

As an alternative solution, this research extracts automatically structured information from a scientific paper in order to help the readers. This structured information is represented based on a rhetorical scheme called Rhetorical Document Profile (RDP) [6]. RDP is a representation of information that readers want to know from a paper so that the readers can determine relevancy of a paper just by reading the paper RDP. In addition, the RDP can also be used to compile a summary of information tailored to the needs of readers who are represented by a set of categories of rhetoric [6]. Rhetoric is the intention information to be conveyed to the reader by the author of the paper [6].

RDP is an instantiated template consisting of fifteen rhetorical slots. Each slot will be filled by a collection of sentences with a specific category. The process of determining the category of rhetorical for every sentence is called rhetorical classification, and the model used in this sentence classification is called rhetorical classifier..

By using corpus-based approach, a classification model is generated by performing supervised learning on a rhetorical corpus, which is a collection of sentences with rhetoric information. Since there was a lack of rhetorical corpus, we constructed our own corpus based on ACL-ARC paper collection [7]. We also investigated best performer of some rhetorical classifiers for our rhetorical corpus.

The rest of the paper is organized as follows. Section 2 provides an overview of previous works related to information extraction of scientific papers and predefined rhetorical categories. Section 3 describes our own rhetorical corpus, and section 4 describes our rhetorical classifier. Experiments are discussed in Section 5, followed by some concluding remarks on Section 6.

II. RELATED WORK

978-1-4577-0752-0/11/$26.00 ©2011 IEEE

Early works on information extraction system from scientific papers were usually specific to a particular domain, such as enzyme interactions and protein structures [8], genetic interaction [9], and biomedical information [10][11]. Moreover, there is also generic information extraction system from scientific papers, such as extracting common fields of header and references [12], and extracting sentences to generate document surrogates based on rhetorical information [6]. This rhetorical-based representation has been applied for some domains: computational linguistics [6][13][14], chemistry [13], astronomy [14], and medical [15].

Each rhetorical category states the purpose to be conveyed by the author of the paper. For analysis of rhetorical structure of an abstract of a scientific paper, some researches only defined four rhetorical categories (introduction, method, result, conclusion) [15] or five categories (background, problem statement, research method, research result, and concluding remarks) [16].

TABLE I
TEUFEL'S (2009) ARGUMENTATIVE ZONING WITH 15 CATEGORIES [13] + TEXTUAL CATEGORY [6]

Category	Description
AIM	Statement of specific research goal, or hypothesis of current paper
NOV_ADV	Novelty or advantage of own approach
CO_GRO	No knowledge claim is raised (or knowledge claim not significant for the paper)
OTHR	Knowledge claim (significant for paper) held by somebody else. Neutral description
PREV_OWN	Knowledge claim (significant) held by authors in a previous paper. Neutral description.
OWN_MTHD	New Knowledge claim, own work: methods
OWN_FAIL	A solution/method/experiment in the paper that did not work
OWN_RES	Measurable/objective outcome of own work
OWN_CONC	Findings, conclusions (non-measurable) of own work
CODI	Comparison, contrast, difference to other solution (neutral)
GAP_WEAK	Lack of solution in field, problem with other solutions
ANTISUPP	Clash with somebody else's results or theory; superiority of own work
SUPPORT	Other work supports current work or is supported by current work
USE	Other work is used in own work
FUT	Statements/suggestions about future work (own or general)
TEXTUAL	Indication of paper's textual structure.

For rhetorical structure of full paper, fifteen rhetorical categories (AZ-II scheme) are defined [13], as an improvement scheme of AZ-I that consists of seven rhetorical slots [6]. In comparison with AZ-I, AZ-II is considered to be more informative, better in recognizing the structure of problem solving, and more subtle in describing a difference [13]. Since our information extraction system processes a full paper, this research employed AZ-II scheme for Rhetorical Document Profile (RDP) as shown by Table I.

Filler of each slot of RDP is a collection of sentences that have the same rhetorical category corresponding to slot category. These sentences are extracted from the scientific paper by a rhetorical classifier.

All previous researches on AZ applied a corpus-based approach [6][14]. A rhetorical classifier is built by performing supervised learning such as Naive Bayes [6], Maximum Entropy [14], Support Vector Machines (SVM) [15], and decision tree [16]. We investigated performance comparison among these classifiers for our rhetorical corpus.

III. OUR RHETORICAL CORPUS

We constructed a rhetorical corpus of computational linguistics domain based on ACL-ARC collection. This paper collection is proposed as standard corpus for natural language processing researches [7].

For computational linguistics, there are two rhetorical corpus used in [6][13] but both cannot be accessed. Corpus in [6] consists of 12471 AZ-I labeled sentences of 79 papers, and corpus in [13] consists of 1629 AZ-II labeled sentences of 9 papers.

For get a low personal bias rhetorical corpus of AZ-II scheme, each paper was annotated by three independent annotators (graduate students who were knowledgeable in computational linguistics). Differences in annotations were resolved by discussion among the annotators until they reached an agreement. This scenario is an alternative scenario to remove personal bias. In the standard model, two independent annotators annotate each sentences in collection, and then the third annotator, as an expert, reconciles the differences [17].

By using GATE 4.0 (General Architecture for Text Engineering), each sentence is annotated with a rhetorical category of AZ-II scheme. Each paper is saved as an xml file. **Fig. 1** shows an example of annotated paragraph in xml format.

```
<P><Sentence><Nov_Adv>Finally we also introduce a baseline that has yet not been introduced in the literature of Japanese compound noun analysis.</Nov_Adv></Sentence><Sentence> <Own_Conc> The baseline works fairly well, and the text scanning method will turn out to be much better than the baseline.</Own_Conc> </Sentence></P>
```

Fig. 1. An example of annotated paragraph

IV. OUR RHETORICAL CLASSIFIER

A rhetorical classifier is built to classify rhetorical categories of new sentences with high accuracy. Supervised learning infers a model from a collection of labeled training sentence by performing induction.

Given a corpus of n examples ($\vec{x}_1$,y_1),...($\vec{x}_n$,y_n). Each example ($\vec{x}_i$,y_i) consists of a sentence vector $\vec{x}_i$ and a category label y_i. The sentence vector represents a vector of

sentence features to classify a rhetorical category. This task is in multi-class setting and called multi-class classification [18].

A. Sentence Features

Table II shows all features involved. We combined features in [6], [14], and our additional features as the sentence representations. There are eight types of features of rhetorical sentences: content, absolute location, explicit structure, sentence length, verb syntax, citations, formulaic expression, and agentivity [6]. Merity [14] proposed different values of some features like straight counter for section, location, and paragraph (shown by * in Table II). We added two additional features: abstract content and qualifying adjective incidence (shown by ** in Table II).

TABLE II
OUR FEATURE POOL BASED ON TEUFEL'S FEATURE TYPES [6]

Type	Name	Description	Values
Content	Cont-1	Significant terms incidence determined by tf.idf	0,1
	Cont-2	Incidence of title or headline words determined by tf.idf	0,1
	Cont-3**	Incidence of significant terms in abstract, determined by tf.idf	0,1
Absolute location	Loc	Sentence position within document relation to 10 segments	1-10
Explicit structure	Struct-1	Sentence position within section	1-7
	Struct-2	Sentence position within paragraf	1-3
	Struct-3	Headline type	0-16
	SectCount*	Section counter	1-10
	SectLoc*	Sentence position within section (straight counter)	1-10
	ParLoc*	Sentence position within paragraf (straight counter)	1-10
Sentence length	Length	Is the sentence longer than 15 words?	0,1
Syntax	Syn	Is the 1st finite verb modified by modal auxiliary ?	0.1
	Adj**	Inicidence of qualifying adjective	0,1
Citations	Cit-1	Citation or self citation incidence	0,1,2
	Cit-2	Citation location in sentence	0,1,2,3
Formulaic expressio n	$Formu_{1..21}$**	Incidence of each formulaic expression in sentence	0,1
Agentivit y	$Ag\text{-}1_{1..16}$**	Incidence of each agent type	0,1
	$Ag\text{-}2_{1..9}$**	Incidence of each action type	0,1
	Negation	Incidence of negation in sentence	0,1

Content features are general features in sentence extraction for determining global sentence relevance. Teufel employed TF-IDF to identify concepts that are characteristic for the contents of the document, and the n top-scoring words are chosen as content words. Sentence scores are computed as a weighted count of the content words in a sentence meaned by sentence length. Since an abstract consists of important sentences that can be a part of important concepts of the paper, we also incorporate it as an additional feature.

Qualifying adjectives are used to state conclusion as author's opinion based on experiment facts. Its incidence is an important features to identify a conclusion sentence. If there is a qualifying adjective, the sentence score is 1.

In addition, we also modified feature extraction of formulaic expression and agentivity. Formu1..21, Ag-11..16, and Ag-21..9 are meta-discourse features extracted by using Teufel's defined patterns [6]. Teufel only used the first occurrence of a pattern in the sentence. Since a sentence can match no pattern, one pattern, or more than one pattern, we implemented each pattern incidence as one boolean feature.

Fig. 2 provides an illustrating example of how a sentence matches five types of formulaic expression, four action types, and two agent types. In particular, this sentence matches with pattern "^ look at how" so that the system extracted action type interest shown by v3. Character ^ in a pattern means the word look is a trigger word for the pattern. A pattern will be considered if the trigger word is found in the sentence.

While[f1] it may be worthwhile to base[v1] such a model on preexisting sense classes (Resnik, 1992), in the[a1] work[f2] described[v2] here[f3] we[a2] look at how[v3] to derive[v4] the classes directly from distributional data.

Types of formulaic expression:	Actions types:	Agent types:
f1: comparison_formulaic	v1: continue	a1: ref_agent
f2: method_formulaic	v2: presentation	a2: us_agent
f3: here_formulaic	v3: interest	
	v4: change	

Fig. 2. An example of meta-discourse features of a sentence; text in the top row is the example sentence, and the three columns below it contain the corresponding meta-discourse features of the sentence.

B. Machine Learning Techniques

This research investigated the following algorithms: Naive Bayes (NB), C4.5, Support Vector Machines (SVM), Logistic, Multilayer Perceptron (MLP), PART, and k-Nearest Neighbours (kNN). We used Weka's implementation of these algorithms, except for SVM from LibSVM.

Naive Bayes (NB) provides a simple approach using probabilistic knowledge with two simplifying assumptions: conditional independence of features, and no hidden attributes influence the prediction [19]. The NB model contains each class probability and conditional probability of each attribute value given a class. Classification uses the model to find a class with maximum probability given an instance.

C4.5 produces decision tree by top-down induction derived from the divide-and-conquer algorithm. Each node in the tree is the best attribute selected based on information gain criterion. [20]

SVM is a learning algorithm that constructs a hyper plane with maximal margin between classes. It finds some support vectors, which are the training data that constrain the margin width.

Logistic and Multi-Layer Perceptron (MLP) are included in the same package of classifiers. Logistic uses a multinomial logistic regression model with a ridge estimator, a restricted

maximum likelihood estimator. Ridge estimator is used to improve the parameter estimates and to reduce the error of predictions [21]. MLP uses backpropagation for classification [20].

PART generates rules by combining rules created from decision trees and the separate-and conquer rule-learning [22]. It learns one rule at a time and avoids postprocessing for efficiency.

k-Nearest Neighbours (kNN) is a lazy learning algorithm that only stores the verbatim training examples. There is no set of abstractions model derived from training examples [23]. In classification, it searches k closest members of the training data and the prediction is majority class of those neighbours.

Some machine learning techniques can handle multi-class problems directly like Naive Bayes and C4.5. LibSVM used one-against-one strategy to handle multi-class [24].

V. EXPERIMENTS

A. Data

Our corpus consists of 10877 annotated sentences in xml format. For experiments, it is split randomly into a training set and a test set. The training set consists of sentences from two third of the total number of papers in the corpus (50 papers), and sentences of the remaining papers are used as the test set.

Table III depicts the distribution of sentences in each rhetorical category. As shown in the table, the corpus poses an imbalanced dataset, and about a half number of sentences is OWN_MTHD sentences. OWN_FAIL and CODI are among categories with smallest sizes.

TABLE III
DESCRIPTION OF DATA PER CATEGORY

Category	Training set		Test set		Total
AIM	136	1.88%	77	2.12%	213
NOV_ADV	179	2.47%	68	1.87%	247
CO_GRO	271	3.74%	113	3.11%	384
OTHR	528	7.29%	444	12.20%	972
PREV_OWN	471	6.51%	150	4.12%	621
OWN_MTHD	3608	49.84%	1717	47.20%	5325
OWN_FAIL	46	0.64%	24	0.66%	70
OWN_RES	264	3.65%	155	4.26%	419
OWN_CONC	385	5.32%	193	5.31%	578
CODI	69	0.95%	42	1.15%	111
GAP_WEAK	241	3.33%	124	3.41%	365
ANTISUPP	36	0.50%	24	0.66%	60
SUPPORT	284	3.92%	109	3.00%	393
USE	244	3.37%	196	5.39%	440
FUT	113	1.56%	38	1.04%	151
TEXTUAL	364	5.03%	164	4.51%	528
Total	7239	100%	3638	100%	10877

B. Results

Table IV shows the accuracies of various rhetorical classifiers. SVM and Logistic classifiers are the best performer with accuracy of 0.51.

TABLE IV
ACCURACIES OF VARIOUS RHETORICAL CLASSIFIERS

Classifier	Accuracy
NB	0.48
C4.5	0.47
SVM	**0.51**
Logistic	**0.51**
MLP	0.31
PART	0.41
1NN	0.38

We tried to improve the SVM accuracy by applying one-against-all strategy to handle multi-class. In this strategy, one binary classifier is built for classify a rhetorical category. For each classifier, training data has two classes: positive and negative. For example, AIM classifier learned from 7239 labeled sentences, and only 136 sentences assigned as positive examples.

Table V shows performances of each SVM binary classifiers. In these classifiers, true negatives are high and the accuracies are also high. If F-measure is used for performance measure, there are some classifiers that can not extract the sentences of some categories like NOV_ADV, OWN_FAIL, OWN_RES, and CODI.

TABLE V
PERFORMANCES OF SVM BINARY CLASSIFIERS

Category	Accuracy	F-measure
AIM	0.98	0.48
NOV_ADV	0.98	0.00
CO_GRO	0.97	0.30
OTHR	0.87	0.01
PREV_OWN	0.96	0.26
OWN_MTHD	0.66	0.66
OWN_FAIL	0.99	0.00
OWN_RES	0.96	0.00
OWN_CONC	0.94	0.01
CODI	0.99	0.00
GAP_WEAK	0.97	0.05
ANTISUPP	0.99	0.08
SUPPORT	0.96	0.08
USE	0.95	0.05
FUT	0.99	0.40
TEXTUAL	0.96	0.25

If we compare the accuracy of ensemble SVM classifier, we cannot improve the performance significantly. Table VI shows two different combining strategies of one-against-all. The second row is performance of classifier with combining strategy using all positive probabilities of all binary classifiers. The last row is performance of classifier with combining strategy using positive probabilities of some classifiers that classify positive for an input sentence. The last classifier is the best performer with accuracy of 0.6 but it cannot classify 1417 sentences (no positive classes).

TABLE VI
ACCURACIES OF VARIOUS SVM CLASSIFIERS

Multi-class strategy	**Accuracy**
one-against-one	0.51
one-against-all with all positive probabilities	0.50
one-against-all with positive probabilities of some classifiers	**0.60**

VI. CONCLUSION

In this paper we have described our rhetorical corpus and various rhetorical classifiers built from full scientific papers. We provide several machine learning techniques and conduct experiments to evaluate their effectiveness on our corpus. The experiment results that SVM and Logistic are the best performer. By applying one-against-all, we can improve the one-against-one SVM classifier. For future work, we will investigate if more features can improve the current performance.

REFERENCES

[1] R.R.Powell, L.M. Baker, and J.J.Mika. 2002. *Library and information science practitioners and research*, Library & Information Science Research, 24 (2002) 49–72

[2] A.E.Jinha. 2010. *Article 50 Million: An Estimate of the Number of Scholarly Articles in Existence*.

[3] Luhn, H.P. (1958). The automatic creation of literature abstracts. IBM Journal of Research and Development.

[4] Jarrett, C. 2007. *Problems and Joys of Reading Research Papers for Practitioner Purposes*, Journal of Usabilitiy Studies, Vol. 3, Issue 1, November 2007, pp. 1-6

[5] S. Ou , C. Khoo, and D. Goh. 2006. Automatic multi-document summarization for digital libraries. In C. Khoo, D. Singh & A.S. Chaudhry (Eds.), Proceedings of the Asia-Pacific Conference on Library & Information Education & Practice 2006 (A-LIEP 2006), Singapore, 3-6 April 2006 (pp. 72-82). Singapore: School of Communication & Information, Nanyang Technological University.

[6] Teufel, S. 1999. *Argumentative zoning: Information Extraction from Scientific Text*, PhD Dissertation, University of Edinburgh.

[7] The ACL Anthology Reference Corpus (ACL ARC). [Online]. Available: http://acl-arc.comp.nus.edu.sg/

[8] Humphreys, K., Demetriou, G. and Gaizauskas, R. 2000. *Two Applications of Information Extraction to Biological Science Journal Articles: Enzyme Interactions and Protein Structures*, Pacific Symposium on Biocomputing 5:502-513

[9] Proux, D., Rechenmann, F., and Julliard, L. 2000. *A Pragmatic Information Extraction Strategy for gathering Data on Genetic Interactions*, in Proc. Of International Conference on Intelligent System on Molecular Biology, 2000

[10] Corney, D.P.A., Buxton, B.F., Langdon, W.B. and Jones, D.T. 2004. *BioRAT: extracting biological information from full-length papers*, Vol. 20 no. 17 2004, pages 3206–3213

[11] Yakushiji, A., Tateisi, Y., and Miyao, Y. 2001. *Event Extraction From Biomedical Papers Using A Full Parser*, Pacific Symposium on Biocomputing 6:408-419

[12] Peng, F., and McCallum, A. 2006. *Accurate Information Extraction from Research Papers using Conditional Random Fields.* Information processing & management, Elsevier

[13] Teufel, S., Siddhartan, A., Batchelor, C. 2009. *Towards Discipline-Independent Argumentative zoning Evidence from Chemistry and Computational linguistics*, in Proc. Of the 2009 Conference on Empirical Methods in Natural Language Processing.

[14] S Merity, T Murphy. 2009. Accurate Argumentative Zoning with Maximum Entropy models. ACL-IJCNLP 2009

[15] L. McKnight, and P. Srinivasan. 2003. Categorization of Sentence Types in Medical Abstracts, in Proc. AMIA 2003 Symposium.

[16] S. Ou , C. Khoo, D. Goh, and H. Heng. 2003. Discourse Parsing of Sociology Dissertation Abstracts Using Decision Tree Induction. In Proc. Of the 13th Annual ASIST SIG CR Workshop.

[17] Petrillo, M., Baycroft, J. ,2010 , *Introduction to Manual Annotation*, Fairview Research

[18] T. Joachims. 2001. Learning to Classify Text Using Support Vector Machines: Methods, Theory, and Algorithms. Kluwer Academic Publishers.

[19] John,G.H., Langley, P. 1995 Estimating Continuous Distributions in Bayesian Classifiers. In *Eleventh Conference on Uncertainty in Artificial Intelligence*, San Mateo, 338-345.

[20] Witten, I.H., Frank, E. 2005. Data mining: practical machine learning tools and techniques.

[21] Cessie, S., van Houwelingen, J.C. (1992). Ridge Estimators in Logistic Regression. *Applied Statistics*. 41(1):191-201.

[22] Frank, E., Witten, I.H. 1998 Generating Accurate Rule Sets Without Global Optimization. In: *Fifteenth International Conference on Machine Learning*, 144-151.

[23] Aha, D., Kibler, D. 1991. Instance-based learning algorithms. *Machine Learning*. 6:37-66

[24] C.-C. Chang and C.-J. Lin. 2009. LIBSVM: a Library for Support Vector Machines.