Hierarchical Rhetorical Sentence Categorization for Scientific Papers

G H Rachman¹, M L Khodra², D H Widyantoro³

¹,²,³School of Electrical Engineering and Informatics, Institut Teknologi Bandung, Bandung, Indonesia

¹ghoziyahaitan@gmail.com, ²masayu@stei.itb.ac.id, ³dwi@stei.itb.ac.id

Abstract. Important information in scientific papers can be composed of rhetorical sentences that is structured from certain categories. To get this information, text categorization should be conducted. Actually, some works in this task have been completed by employing word frequency, semantic similarity words, hierarchical classification, and the others. Therefore, this paper aims to present the rhetorical sentence categorization from scientific paper by employing TF-IDF and Word2Vec to capture word frequency and semantic similarity words and employing hierarchical classification. Every experiment is tested in two classifiers, namely Naïve Bayes and SVM Linear. This paper shows that hierarchical classifier is better than flat classifier employing either TF-IDF or Word2Vec, although it increases only almost 2% from 27.82% when using flat classifier until 29.61% when using hierarchical classifier. It shows also different learning model for child-category can be built by hierarchical classifier.

1. Introduction

Actually domain of scientific papers become one kind of document that is often needed by academics and researchers. Then they use it as reference or comparison for their research by collecting relevant information [2]. This information can be composed of rhetorical sentences to be structured from some categories as needed by readers [3]. Rhetorical sentence itself is each sentences in document which is assigned with label of defined rhetorical categories based on its text features and rhetorical structure is a connected structure of texts, in which every segment of a text (i.e. sentence) has a meaningful category in the body of each section in the document [1].

Teufel [4] initially conducted rhetorical sentence categorization by assigning one of seven rhetorical categories to every sentence. These seven are “aim”, “textual”, “own”, “background”, “contrast”, “basis”, and “other”. After that, these seven is improved into 15 rhetorical categories [5], namely “aim”, “nov_adv”, “co_gro”, “othr”, “prev_own”, “own_mthd”, “own_fail”, “own_res”, “use”, “own_conc”, “codi”, “gap_weak”, “antisupp”, “support”, and “fut”. For seven rhetorical categories, Merity et al. [6] has improved its F-measure performance become 92.93% by using Maximum Entropy classifier from Teufel’s result [7]. Besides that, Widyantoro, et al [8] implemented 15 categories added with “textual” and employed multi-heterogeneous classifier. Its average of F-measure result is about 25%. They still used the technique of flat classification that only considers leaf nodes (child-category) without parent-category, as shown in Figure 1, in which no structure defines relationship between the existing categories [9] [10]. So based on our best knowledge, there is no hierarchical classification yet that is conducted for rhetorical sentence categorization in scientific papers. Due to this condition, in this paper, we develop hierarchical classifier for rhetorical sentence
categorization by employing rhetorical categories from Teufel, et al [5]. This paper is actually next results of our research in rhetorical sentence categorization.

The rest of this paper is organized as follows. The next section provides the related works on rhetorical sentence categorization from Teufel and implementation of Word2Vec semantic representation and hierarchical classification. Our method is explained in section 3. We also define various rhetorical categories and feature sets for classifying sentences. We describe our experiments in section 4. The results are analyzed in section 5. Finally, the conclusion and further work are described in section VI.

2. Related Work
Rhetorical sentence categorization is commonly used for scientific papers. Teufel and Moens [7] conducted this task by assigning one of seven rhetorical categories to every sentence. These seven are “aim”, “textual”, “own”, “background”, “contrast”, “basis”, and “other”. They employed Naïve Bayes classifier and got the F-measure result from 26% to 86%. Then, its result has risen up until 92.93% after employing Maximum Entropy [6]. These seven categories next become 15 rhetorical categories [5], namely “aim”, “nov_adv”, “co_gro”, “othr”, “prev_own”, “own_mthd”, “own_fail”, “own_res”, “use”, “own_conc”, “codi”, “gap_weak”, “antisupp”, “support”, and “fut”. The distribution detail of 15 categories from 7 rhetorical categories is shown in Table 1.

| 7 Rhetoric Categories [4] | 15 Rhetoric Categories [5] |
|---------------------------|-----------------------------|
| Aim                       | Aim                         |
| Background                | Co_gro                      |
| Other                     | Othr                        |
|                           | Prev_own                    |
|                           | Own_mthd                    |
|                           | Own_fail                    |
| Own                       | Own_res                     |
|                           | Own_conc                    |
| Basis                     | Use                         |
|                           | Support                     |
|                           | Antisupp                    |
| Contrast                  | Codi                        |
|                           | Gap_weak                    |
| Textual*                  | Nov_adv**                   |
|                           | Fut**                       |

*This category is not continued in 15 rhetorical categories
**Teufel did not involve these categories to any 7 rhetorical categories

Widyantoro, et al. [8] implemented these 15 categories added with “textual” by adapting Teufel’s features and employing multi-heterogeneous classifier. Its average of F-measure result is about 25%. Actually, they used different corpus from Teufel’s [4]. This condition could affect the final rhetorical
categorization model, because Teufel employed meta-discourse feature, in which their existing grammars depend on word-patterns that always appear in sentences from corpus.

For the second, Word2Vec is used to group words which have similar meaning into vector representation. This model is proposed by Mikolov [12] and released by Google in 2013. There are two architectures of Word2Vec, namely the continuous bag-of-words (CBOW) and skip-gram. The CBOW predicts the current word based on the context and the skip-gram predicts surrounding words given the current word [13]. Heffernan and Teufel [14] employed Word2Vec representation to identifying problem statement in scientific text. They used 18,753,472 sentences from a biomedical corpus based on all full-text Pubmed articles and then built model from the 200 semantically similar words to only “problem”. Its result showed that Word2Vec model caused significant performance increase, because Word2Vec attributes have the greatest information gain compared the other features.

At last, flat classification is commonly conducted for text categorization. Actually, there is another technique named as hierarchical classification. The difference between them is flat classification only considers leaf nodes (child-category) without parent-category [9] [10], but hierarchical classification considers parent-category first then child-category from each parent. It assigns more suitable categories from the defined hierarchical category in document [15]. Tegegnie, et al [16] classified a heterogeneous collection of Amharic News Text by employing hierarchical structure. It shows that hierarchical classifier is better than flat classifier. The accuracy when using level three classifier with the top 15 features is 89.06%. It significantly improves 29.42% from the accuracy when using flat classifier.

3. Methodology
In this research, we use the data set of scientific paper from Khodra, et al. [3] added with 50 new scientific papers from ACL Anthology Reference Corpus (ACL-ARC). This data has 75 annotated scientific papers that have been split into 10,880 sentences. It is added with 50 new scientific papers, then our data set has 16,046 sentences. Training data is 100 and testing is 25 scientific papers. The number of training data set is 12,594 and testing data set is 3,452.

Every sentence in these papers has been annotated into one of 16 rhetorical categories from Teufel, et al. [5], namely “aim”, “nov_adv”, “co_gro”, “othr”, “prev_own”, “own_mthd”, “own_fail”, “own_res”, “own_conc”, “codi”, “gap_weak”, “antisupp”, “support”, “use”, “fut”, and “textual”. The description of these categories is explained in [8]. We adapt features from Teufel [4] and [3], namely content, absolute location, explicit structure, sentence length, syntax, citation, and formulaic & agentivity (meta-discourse). Its explanation can be seen in Table 2.

| Type                  | Name | Description                                      |
|-----------------------|------|--------------------------------------------------|
| Content               | Cont-1 | Occurrence of 10 significant terms of document using TF-IDF |
|                       | Cont-2a | Incidence of words occurring in document title |
|                       | Cont-3 | Occurrence of 10 significant terms of abstract using TF-IDF |
| Absolute location     | Loc | Sentence position in document to 10 segments |
|                       | Struct-1 | Sentence position within section |
| Explicit structure    | Struct-2 | Sentence position within paragraph |
|                       | Struct-3 | Prototypical type of section title |
| Sentence length       | Length | Whether sentence has longer than 15 words or not |
| Syntax                | Syn | Occurrence of 1st finite verb and auxiliary modal |
|                       | Adj | Occurrence of adjective |
| Citations             | Cit-1 | Occurrence of citation or self-citation |
|                       | Cit-2 | Citation location in sentence |
| Formulaic             | Formu | Occurrence of formulaic expression |
| Agentivity            | Ag-1 | Occurrence of agent type |
|                       | Ag-2 | Occurrence of action type |
|                       | Negation | Occurrence negation in sentence |
Our method is generally divided into four main processes. The first is pre-processing, and then constructing the vector representations of the vocabularies of data set by using TF-IDF and Word2Vec. The second is producing all the features. The third is building classification model. This will employ technique of hierarchical classification. The last is we conduct testing to get the F-measure by using some classifiers, namely Naïve Bayes [17] and SVM Linear utilizing LibSVM [18] library. The general process is shown in Figure 2.

Technique of hierarchical classification is conducted to the child category after doing the first classification in parent-category. So, there are two step of flat classification for parent-category and child-category. In this research, we use this technique for category “contrast”, “own”, “basis”, and “other” because only these categories are divided into some child-categories [4] [5]. This hierarchical rhetorical category schema that will be done is shown in Figure 3. As shown in Table 1, “nov_adv” and “fut” are not derivative from any category, but in this paper we assume that these two are elaborated from category “own”. It is due to the definition of “nov_adv” and “fut” that is mostly related to the definition of “own”.

![Figure 2. General process of our methodology](https://github.com/medallia/Word2VecJava)

![Figure 3. Hierarchical rhetorical category](https://github.com/medallia/Word2VecJava)

We consider four pre-processing that consist of case folding, tokenization, stemming, and stop word elimination. These are done using Apache Lucene library and Weka [19]. All of them will always be applied in all experiments. Then this paper employs Word2Vec representation algorithm from Medallia\(^1\) library. We use 20,157 scientific papers taken from ACL Anthology Reference Corpus (ACL-ARC) to build the word vector learning model itself. After that, we use our annotated pre-processed sentences to output word vectors as feature of words. The weight of word is calculated through TF-IDF and model from Word2Vec representation (Skip-Gram). We compare the usage of them. For Word2Vec, we set its layer size into the vector length of 300. In the end, all classification models are produced from the last process.

Especially for hierarchical classification, we employ SMOTE [20] and SpreadSubsample from Weka to try dealing with imbalance data set. Actually, the difference of this paper from our previous work [11], namely: employing meta-discourse feature from Teufel’s; not using previous label; using Naïve Bayes and SVM; doing hierarchical classification technique; and adding data set for 16 rhetorical categories and data set for Word2Vec.

3.1. Experiment

\(^1\)https://github.com/medallia/Word2VecJava
In addition, there are four experiments that is conducted. It is related to implementation of TF-IDF, Word2Vec of Skip-Gram, and hierarchical classification. These are as follows:

- **Baseline.** The features to use are content, absolute location, explicit structure, sentence length, syntax, formulaic & agentivity (meta-discourse), and citation. These are adapted from [4] and [3];
- **Scenario 1.** In this experiment, all the features from baseline are used. Then, we employ TF-IDF to give score for bag of words from data set;
- **Scenario 2.** This experiment will employ all the features and involve Word2Vec to give score of semantic similarity words by using Skip-Gram algorithm. This scenario is conducted to compare the result between using Word2Vec and TF-IDF;
- **Scenario 3.** This experiment will employ all features from baseline and Word2Vec representation, then we conduct the hierarchical classification from seven rhetorical categories until 16 categories. In second step of hierarchical classification, we also employ SpreadSubsample and SMOTE.

4. Result and Analysis

First, we built the word vector learning model itself for the 20,157 scientific papers taken from ACL Anthology Reference Corpus (ACL-ARC). Besides that, we use 100 scientific papers for training and 25 for testing. We extracted all the features regarding scenario we planned. This was evaluated by two classifiers, namely Naïve Bayes and SVM Linear.

| No | Rhetorical Category | Baseline | Scenario 1 (TF-IDF) | Scenario 2 (Word2Vec) |
|----|---------------------|----------|---------------------|-----------------------|
|    | Naïve Bayes | SVM Linear | Naïve Bayes | SVM Linear | Naïve Bayes | SVM Linear |
| 1  | othb        | 13.40  | 3.50 | 10.50 | 23.20 | 4.60 | 15.30 |
| 2  | prev_own    | 16.00  | 22.00 | 21.80 | 9.70 | 27.50 | 22.00 |
| 3  | fut         | 32.30  | 35.30 | 17.80 | 9.10 | 13.70 | 34.60 |
| 4  | nov_adv     | 15.50  | 0.00 | 4.70 | 0.00 | 2.80 | 10.50 |
| 5  | own_conc    | 25.30  | 26.40 | 15.50 | 27.30 | 11.70 | 33.80 |
| 6  | own_fail    | 4.50   | 0.00 | 2.90 | 0.00 | 0.00 | 0.00 |
| 7  | own_mthd    | 68.10  | 68.00 | 32.80 | 69.20 | 17.20 | 71.50 |
| 8  | own_res     | 10.90  | 0.00 | 29.50 | 13.60 | 9.40 | 24.60 |
| 9  | antisupp    | 0.00   | 0.00 | 0.00 | 0.00 | 1.40 | 0.00 |
| 10 | codi        | 0.00   | 0.00 | 1.80 | 0.00 | 2.50 | 0.00 |
| 11 | gap_weak    | 23.40  | 7.70 | 15.80 | 7.50 | 12.60 | 31.20 |
| 12 | aim         | 46.00  | 49.00 | 29.70 | 46.40 | 15.70 | 58.70 |
| 13 | textual     | 17.90  | 1.00 | 36.00 | 51.70 | 23.30 | 65.20 |
| 14 | support     | 21.40  | 18.80 | 12.00 | 19.60 | 8.00 | 23.40 |
| 15 | use         | 16.80  | 17.70 | 12.60 | 14.20 | 16.30 | 17.30 |
| 16 | co_gro      | 30.90  | 33.70 | 18.60 | 28.20 | 17.10 | 39.00 |
| **Total** | 21.40 | 17.69 | 16.38 | 19.98 | 11.49 | 27.82 |

Table 3 shows the result of first three experiments. After employing bag of words TF-IDF feature in Scenario 1, the performance of baseline experiment using only Teufel’s features is still higher than adding bag of words TF-IDF. It can be because co-occurrence of significant words using TF-IDF has been applied for feature of content (Cont-1 and Cont-3). This feature applies TF-IDF more specifically, in which the significant words produced for each document are different. Meanwhile bag of words TF-IDF is more general for all documents. So that it does not have significant impact for the performance.

In baseline, Naïve Bayes produced the highest F-measure result that is 21.40%, but in Scenario 1 and 2 it keeps decreasing when using bag of words. It is proven that Naïve Bayes is not suitable to handle high dimensional feature caused by bag of words, either employing TF-IDF or Word2Vec. This result is different when using SVM Linear. From baseline until Scenario 2, the F-measure result has
risen up from 17.69% to 27.82%. This indicates that the classification method is suitable for high dimensional feature though.

Besides that, the increasing performance is because either TF-IDF or Word2Vec can capture some important words that often appear in sentences but do not exist in the dictionary of meta-discourse (formulaic & agentivity) from Teufel’s [4]. Meanwhile only Word2Vec can capture words that is similar to those important words. Co-occurrence of these similar important words can affect the decision of assigning category to sentences. Some of these words are align, annotate, argument, aspect, concept, consist, context, data, depend, differ, direct, distribute, document, domain, embed, express, extract, feature, function, generate, grammar, include, inform, knowledge, label, language, learn, and many others.

Actually, as shown in Table 3, some categories do not improve at all after applying either TF-IDF or Word2Vec representation. These are “own_fail”, “antisupp”, and “codi”. It can be because these categories have low number of sentences and no related important words appear often in sentences from these categories. This result is different from category “textual”. This category significantly improves from experiment of baseline until Scenario 2. It means there are some important words that almost always appear in sentence from this category. Scenario 2 can capture these words, while baseline experiment cannot. Besides that, some categories have fairly stable performance since baseline experiment, just like “own_mthd”, “aim”, “fut”, and “co_gro”. It means that sentences from these categories have patterns score of baseline features that often appear. So it has been good enough to define these categories without TF-IDF or Word2Vec.

### Table 4. Comparison of f-measure result from scenario 3

| No | Rhetorical Category | Hierarchical Classifier Step 1 | Rhetorical Category | Hierarchical Classifier Step 2 |
|----|---------------------|-------------------------------|---------------------|-------------------------------|
|    |                     | SVM Linear                    |                     | SVM Linear                    |
| 1  | other              | 18.20                         | othr                | 14.20                         |
| 2  | prev_other         | 21.10                         | prev_own            | 21.10                         |
| 3  | own                | 80.60                         | own_adv             | 40.00                         |
| 4  | own_conc           | 20.10                         | own_conc            | 20.10                         |
| 5  | own                | 23.10                         | own      | 8.30  |
| 6  | own_conc           | 8.30  | own_fut | 23.10 |
| 7  | own                | 71.30                         | own_mthd           | 31.50                         |
| 8  | own_res            | 31.50                         | own_res            | 31.50                         |
| 9  | contrast           | 6.95  | antisupp | 9.50  |
| 10 | contraste          | 4.00  | gap_weak | 20.00 |
| 11 | aim                | 59.50                         | aim                | 59.50                         |
| 12 | textual            | 62.50                         | textual            | 62.50                         |
| 13 | support            | 32.30                         | support            | 32.30                         |
| 14 | use                | 17.30                         | use                | 17.30                         |
| 15 | background         | 39.10                         | co_gro             | 39.10                         |
| 16 | Total              | 45.09                         | Total              | 29.61                         |

In Scenario 3, we only employ SVM Linear because it has been proven in most previous experiment that this is the best classifier so far for high dimensional feature. Then we conduct hierarchical classification for category “contrast”, “own”, “basis”, and “other” because only these categories are divided into some child-categories [4] [5]. We build new learning model for child-category from these categories. Its total is four model. The advantage of applying hierarchical classification, we can build learning model for child-category by using the features we select. Every model can be different. Actually, for child-category from “own”, “basis”, and “other”, we employ all features from Teufel [4] and [3], Word2Vec representation, and SMOTE. We employ SMOTE to know that whether it can improve some categories that have F-measure 0.00% in Scenario 2 with SVM Linear or not. Then learning model of child-category from “contrast” employs all features from Teufel, Word2Vec representation, and SpreadSubsample.

The result of Scenario 3 is shown in Table 4. There is no F-measure 0.00% for “own_fail”, “antisupp”, and “codi”. The F-measure result for “own_fail” is 8.30%, “antisupp” 9.50%, and “codi”
4.00%. It is proven that SMOTE and SpreadSubsample can improve performance of categories with the low number sentences. Meanwhile, as shown in Fig 4, the performance of some categories has decreased quite highly, just like “own_conc” and “gap_weak”. They decrease about 10% until 11%. It can happen because employing SMOTE in parent-category “own” and SpreadSubsample in parent-category “contrast”. In category “own”, we only employ SMOTE for “own_fail”, “nov_adv”, and “own_res” that have the low number of sentences. Besides that, in category “contrast”, we only employ SpreadSubsample for “antisupp” and “codi”. This means that SMOTE and SpreadSubsample can decrease the performance of categories that is not set.

![Figure 4. Comparison between f-measure result from scenario 2 and 3](image)

Actually, the result between Scenario 2 using flat classification and Scenario 3 using hierarchical classification in Figure 4 shows that hierarchical classifier is better than flat classifier, although its difference is not significant. It increases only almost 2% from 27.82% until 29.61%. Hierarchical classifier can be better because we can build different learning model for child-category by using the features we select. For example, we employ SpreadSubsample only for parent-category “contrast” which is divided into “antisupp”, “gap_weak”, and “codi”. We can select features as needed and suitable for group of child-category. In flat classification, we cannot conduct this technique, because we only build one learning model for all categories. Therefore, this hierarchical classification model is considered.

5. Conclusion and Further Work

The implementation of Word2Vec can improve the performance of some rhetorical sentence classification from baseline scenario. It can be because Word2Vec model can capture some related important words that do not exist in meta-discourse dictionary from [4]. The result is different from implementation of TF-IDF. This feature does not significantly affect the performance, because co-occurrence of significant words using TF-IDF has been applied for feature of content (Cont-1 and Cont-3). This feature applies TF-IDF more specifically, in which the significant words produced for each document are different. Besides that Word2Vec is also better than TF-IDF, because TF-IDF only capture co-occurrence of important words not related words.

Furthermore our work shows that the implementation of hierarchical rhetorical sentence categorization improves the performance of rhetorical sentence categorization, although is not significant. Hierarchical classifier is better than flat classifier because we can build different learning model for child-category. We can select features as needed and suitable for group of child-category. In flat classification, we cannot conduct this technique, because we only build one learning model for all categories. In addition, we do not use previous label in this paper, because in second step of hierarchical classification, we have to build model for group of child-category, in which its label is not structured any more like 16 rhetorical categories. So it is difficult to predict previous label and define “start” label like in [11].
For the future work, we plan to conduct research for every child-category in 16 rhetorical categories from Teufel [5]. It is to know that whether every child-category from same parent-category have high similar category or not.

References
[1] Taboada M and Mann W C 2006 Rhetorical structure theory: Looking back and moving ahead Discourse studies vol 8 no 3 pp 423-59
[2] Schwegler R A and Shamoon L K 1982 The Aims and Process of the Research Paper College English vol 44 no 8 pp 817-24
[3] Khodra M L, Widyantoro D H, Aziz E A and Trilaksono B R 2012 Automatic Tailored Multi-Paper Summarization based on Rhetorical Document Profile and Summary Specification Journal of ICT Research and Applications vol 6 no 3 pp 220-239
[4] Teufel S 1999 Argumentative Zoning: Information Extraction from Scientific Text (Edinburgh: University of Edinburgh)
[5] Teufel S, Siddharthan A and Batchelor C 2009 Towards discipline-independent argumentative zoning: evidence from chemistry and computational linguistics Proc. of the 2009 Conf. on Empirical Methods in Natural Language Processing (Association for Computational Linguistics) pp 1493-1502
[6] Merity S, Murphy T, and Curran J 2009 Accurate Argumentative Zoning with Maximum Entropy models, Proc. of the 2009 Workshop on Text and Citation Analysis for Scholarly Digital Libraries (Suntec: World Scientific Publishing Co Pte Ltd) pp 19-26
[7] Teufel S and Moens M 2002 Summarizing scientific articles: experiments with relevance and rhetorical status (Computational Linguistics - Summarization vol 28) issue 4 pp 409-445
[8] Widyantoro D H, Khodra M L, Trilaksono B R and Aziz E A 2013 A multiclass-based classification strategy sentence categorization from scientific papers Journal of ICT Research and Applications vol 7 no 3 pp 235-49
[9] D’Alessio S, Murray K, Schiaffino R and Kershenbaum A 2000 The effect of using hierarchical classifiers in text categorization (Content-Based Multimedia Information Access-Volume 1) pp 302-13
[10] Silla Jr C N and Freitas A A 2011 A survey of hierarchical classification across different application domains (Data Mining and Knowledge Discovery vol 22) no 1-2 pp 31-72
[11] Rachman G H, Khodra M L, Widyantoro D H 2017 Rhetorical sentence categorization for scientific paper using word2vec semantic representation IOP Publishing, Journal of Physics: Conference Series vol 801 no 1
[12] Rong X 2014 word2vec Parameter Learning Explained (Cornell University Library) Preprint cs/1411.2738
[13] Mikolov T, Chen K, Corrado G and Dean J 2013 Efficient estimation of word representations in vector space (Cornell University Library) Preprint cs/1301.3781
[14] Heffernan K and Teufel S 2016 Identifying problem statements in scientific text, foundations of the language of argumentation, Proc. of COMMA 2016 Workshop (University of Potsdam) p 18
[15] Sun A and Lim E P 2001 Hierarchical text classification and evaluation, Proc. IEEE International Conference on Data Mining, ICDM 2001 (IEEE) pp 521-528
[16] Tegegne A K, Tarekegn A N and Alemu T A 2017 A comparative study of flat and hierarchical classification for amharic news text using svm Int. Journal of Information Engineering and Electronic Business (IJIEEB) vol 9 no 3 pp 36-42
[17] Rish I 2001 An empirical study of the naive Bayes classifier, IJCAI 2001 workshop on
empirical methods in artificial intelligence (IBM New York)

[18] Chang C C and Lin C J 2011 LIBSVM: a library for support vector machines *ACM Transactions on Intelligent Systems and Technology (TIST)* vol 2 no 3

[19] Hall M, Frank E, Holmes G, Pfahringer B, Reutemann P and Witten I H 2009 The WEKA data mining software: an update *ACM SIGKDD explorations newsletter* vol 11 no 1 pp 10-18

[20] Chawla N V, Bowyer K W, Hall L O and Kegelmeyer W P 2002 SMOTE: synthetic minority over-sampling technique *Journal of artificial intelligence research* vol 16 pp 321-57.

Acknowledgement

This research was partially supported by the Master’s Program toward Doctoral Degree for Excellent Graduate (*Program Pendidikan Magister Menuju Doktor untuk Sarjana Unggul/PMDSU*) from Kemenristekdikti Indonesia within research entitled “Automatic Guided Summarization”.

