Investigating the Effectiveness of Thesaurus Generated Using Tolerance Rough Set Model

Gloria Virginia and Hung Son Nguyen

University of Warsaw, Faculty of Mathematics, Informatics and Mechanics
Banacha 2, 02-097 Warsaw, Poland

Abstract. We considered the tolerance matrix generated using tolerance rough set model as a kind of an associative thesaurus. The effectiveness of the thesaurus was measured using performance measures commonly used in information retrieval, recall and precision, where they were used for the *terms* rather than *documents*. A corpus consists of keywords defined as highly related with particular topic by human experts become the ground truth of this study. Analysis was conducted based on comparison values of all available sets created. Above all findings, this paper was thought as the fundamental basis that generating an automatic thesaurus using rough sets theory is a promising way. We also mentioned some directions for future study.

Keywords: rough sets, tolerance rough set model, thesaurus.

1 Introduction

Rough set theory is a mathematical approach to vagueness [12] that was introduced by Pawlak in 1982 [11]. It's relationship with other approaches has been studied for years and it has been successfully implemented in numerous areas of real-life applications [5]. Tolerance rough set model (TRSM) is one of its extension developed by Kawasaki, Nguyen, and Ho [4] based on the *generalized approximation space* as a tool to model document-term relation in text mining.

Hierarchical and non-hierarchical document clustering based on TRSM has been studied in [4] and [8] respectively and showed that the clustering algorithm being proposed could be well adapted to text mining. The study of TRSM implementation to search results clustering in [8] yielded a design of a Tolerance Rough Set Clustering (TRC) algorithm for web search results and proved that the new representation created had positive effects on clustering quality. For query expansion, the result of TRSM implementation showed that the approach was effective and high search precision was gained [3,8].

The potential of TRSM in automatic thesaurus construction has been revealed in [16]. By employing similar framework of study, this paper presents our investigation on the effectiveness of the thesaurus automatically created, both with and without stemming task on the process. The effectiveness of the thesaurus was calculated using performance measures commonly used in information retrieval which are recall and precision.

M. Kryszkiewicz et al. (Eds.): ISMIS 2011, LNAI 6804, pp. 705–714, 2011.

Brief explanation about rough sets theory, generalized approximation space and tolerance rough set model are presented on the next section and then followed by description of data and methodology used in the study. We report and discuss our findings in section 5.

2 Basic Notions on Tolerance Rough Set Model (TRSM)

Rough set theory was originally developed [12] as a tool for data analysis and classification. It has been successfully applied in various tasks, such as feature selection/extraction, rule synthesis and classification [5,10]. The central point of rough set theory is based on the fact that any concept (a subset of a given universe) can be approximated by its *lower* and *upper approximation.*

The classical rough set theory is based on equivalence relation that divides the universe of objects into disjoint classes. For some practical applications, the requirement for equivalent relation has showed to be too strict. The nature of the concepts in many domains are imprecise and can be overlapped additionally.

In [13], Skowron and Stepaniuk introduced a generalized approximation space (GAS) by relaxing the equivalence relation in classical rough sets to a tolerance relation, where transitivity property is not required. Formally, the generalized approximation space is defined as a quadruple $\mathcal{A} = (U, I, \nu, P)$, where

U is a non-empty universe of objects; let $\mathcal{P}(U)$ denote the power set of U,

$I : U \to \mathcal{P}(U)$ is an *uncertainty function* satisfying conditions: (1) $x \in I(x)$ for $x \in U$, and (2) $y \in I(x) \iff x \in I(y)$ for any $x, y \in U$. Thus the relation $xRy \iff y \in I(x)$ is a tolerance relation and $I(x)$ is a tolerance class of x,

$\nu : \mathcal{P}(U) \times \mathcal{P}(U) \to [0, 1]$ is a *vague inclusion function*, which measures the degree of inclusion between two sets. The function ν must be *monotone* w.r.t the second argument, i.e., if $Y \subseteq Z$ then $\nu(X, Y) \leq \nu(X, Z)$ for $X, Y, Z \subseteq U$,

$P : I(U) \to \{0, 1\}$ is a *structurality function.*

Together with uncertainty function I, vague inclusion function ν defines the *rough membership function* for $x \in U, X \subseteq U$ by $\mu_{I,\nu}(x, X) = \nu(I(x), X)$. Lower and upper approximations of any $X \subseteq U$ in $\mathcal{A}$, denoted by $\mathbf{L}_{\mathcal{A}}(X)$ and $\mathbf{U}_{\mathcal{A}}(X)$, are respectively defined as $\mathbf{L}_{\mathcal{A}}(X) = \{x \in U : P(I(x)) = 1 \wedge \nu(I(x), X) = 1\}$ and $\mathbf{U}_{\mathcal{A}}(X) = \{x \in U : P(I(x)) = 1 \wedge \nu(I(x), X) > 0\}$.

Let us notice that the classical rough sets theory is a special case of GAS. However, with given definition above, generalized approximation spaces can be used in any application where I, ν and P are appropriately determined.

Tolerance Rough Set Model (TRSM) [4] was developed as basis to model documents and terms in information retrieval, text mining, etc. With its ability to deal with vagueness and fuzziness, tolerance rough set seems to be promising tool to model relations between terms and documents. In many information retrieval problems, especially in document clustering, defining the similarity relation between document-document, term-term or term-document is essential.

Let $D = \{d_1, \ldots, d_N\}$ be a corpus of documents and $T = \{t_1, \ldots, t_M\}$ set of *index terms* for D. With the adoption of Vector Space Model [7], each document

d_i is represented by a weight vector $[w_{i1}, \ldots, w_{iM}]$ where w_{ij} denoted the weight of term t_j in document d_i. TRSM is an approximation space $\mathcal{R} = (T, I_\theta, \nu, P)$ determined over the set of terms T as follows:

Uncertainty function: $I_\theta(t_i) = \{t_j \mid f_D(t_i, t_j) \geq \theta\} \cup \{t_i\}$, where θ is a positive parameter and $f_D(t_i, t_j)$ denotes the number of documents in D that contain both terms t_i and t_j. The set $I_\theta(t_i)$ is called the *tolerance class* of term t_i,
Vague inclusion function: is defined as $\nu(X, Y) = \frac{|X \cap Y|}{|X|}$,
Structural function: $P(I_\theta(t_i)) = 1$ for all $t_i \in T$.

The membership function μ for $t_i \in T$, $X \subseteq T$ is then defined as $\mu(t_i, X) = \nu(I_\theta(t_i), X) = \dfrac{|I_\theta(t_i) \cap X|}{|I_\theta(t_i)|}$ and the lower, upper approximations and boundary regions of any subset $X \subseteq T$ can be determined – with the obtained tolerance $\mathcal{R} = (T, I, \nu, P)$ – in the standard way, i.e.,

$$L_{\mathcal{R}}(X) = \{t_i \in T \mid \nu(I_\theta(t_i), X) = 1\} \ . \tag{1}$$

$$U_{\mathcal{R}}(X) = \{t_i \in T \mid \nu(I_\theta(t_i), X) > 0\} \ . \tag{2}$$

$$BN_{\mathcal{R}}(X) = U_{\mathcal{R}}(X) - L_{\mathcal{R}}(X) \ . \tag{3}$$

In the context of information retrieval, tolerance class $I_\theta(t_i)$ represents the concept related to t_i. By varying the threshold θ, one can tune the preciseness of the concept represented by a tolerance class. For any set of terms X, the upper approximation $\mathbf{U}_{\mathcal{R}}(X)$ is the set of concepts that share some semantic meanings with X, while $\mathbf{L}_{\mathcal{R}}(X)$ is a "core" concept of X. The application of TRSM in document clustering was proposed as a way to enrich document and cluster representation with the hope of increasing clustering performance.

Enriching document representation: With TRSM, the "richer" representation of document $d_i \in D$ is achieved by simply representing document with its upper approximation, i.e. $\mathbf{U}_{\mathcal{R}}(d_i) = \{t_i \in T \mid \nu(I_\theta(t_i), d_i) > 0\}$
Extended weighting scheme: In order to employ approximations for document, the weighting scheme need to be extended to handle terms that occurs in document's upper approximation but not in the document itself. The extended weighting scheme is defined from the standard TF*IDF by:

$$w_{ij}^* = \frac{1}{S} \begin{cases} (1 + \log f_{d_i}(t_j)) \log \frac{N}{f_D(t_j)} & \text{if } t_j \in d_i \\ \min_{t_k \in d_i} w_{ik} \frac{\log \frac{N}{f_D(t_j)}}{1 + \log \frac{N}{f_D(t_j)}} & \text{otherwise} \end{cases}$$

where S is a normalization factor.

The use of upper approximation in similarity calculation to reduce the number of zero-valued similarities is the main advantage TRSM-based algorithms claimed to have over traditional approaches. This makes the situation, in which two documents have a non-zero similarity although they do not share any terms, possible.

3 Data of the Study

This study used two corpora: ICL-corpus and WORDS-corpus. Both of them adopt the Text REtrieval Conference (TREC) format [9], i.e. every document is marked up by <DOC></DOC> tags and has a unique document identifier which is marked up by <DOCNO></DOCNO> tags.

The ICL-corpus consists of 1,000 documents which came from Indonesian Choral Lovers Yahoo! Groups, a mailing list of Indonesian choral community, hence the body text is the body of email. During annotation process, each document of ICL-corpus has been assigned a topic, or more, by choral experts and concurrently they were expected to determine words that highly related with the topics given [15]. We then treated those keywords as the body text of document in WORDS-corpus. Therefore, both corpora are basically correlated in the sense that WORDS-corpus contains keywords defined by human experts for the given topic(s) of each document in ICL-corpus. Hence, the WORDS-corpus also consists of 1,000 documents and the identifier of WORDS-corpus' document is in accordance with identifier of ICL-corpus' document.

We take an assumption that each topic given by the human experts in annotation process is a concept, therefore we consider the keywords determined by them as the term variants that semantically related with particular concept. These keywords are in WORDS-corpus, hence the WORDS-corpus contains important terms of particular concept selected by human expert. In automatic process of the system, these terms should be selected, therefore WORDS-corpus become the ground truth of this study.

Topic assignment yielded 127 topics which many of them has few document frequency; 81.10 % have document frequency less than 10 and 32.28 % of them have document frequency 1.

4 Methodology

A thesaurus is a type of lightweight ontology which provides additional relationships, and does not provide an explicit hierarchy [6]. By definition, a thesaurus could be represented technically in the form of a term-by-term matrix. In this study, we considered the *tolerance matrix* generated using TRSM as the representation of the intended thesaurus, which is technically a term-by-term matrix and contains tolerance classes of all index terms.

Figure 1 shows the main phases of the study, which were performed twice: with stemming task and without stemming task.

4.1 Extraction Phase

The main objective of extraction phase was preprocessing both corpora. A version of Indonesian stemmer, called *CS stemmer*, was employed in stemming task. In [1], it was introduced as a new confix-stripping approach for automatic Indonesian stemming and was showed as the most accurate stemmer among other

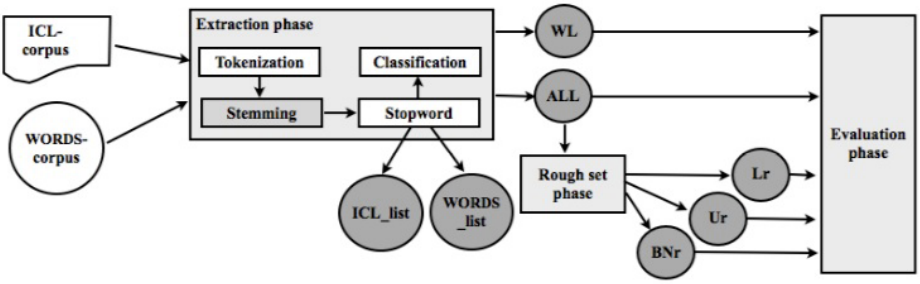

Fig. 1. Main phases of the study: extraction phase, rough set phase, and evaluation phase. A rectangle represents a phase while a circle represent a result.

automated Indonesian stemmer. For stopword task, Vega's stopword [14] was applied as in [2] the use of the stopword gave highest precision and recall.

Documents were tokenized based on character other than alphabetic. The resulted tokens were stemmed using the CS stemmer and then compared to the Vega's stopword. It yielded list of unique terms and its frequency. There were 9,458 unique terms extracted from ICL-corpus and 3,390 unique terms extracted from WORDS-corpus; called *ICL_list* and *WORDS_list* respectively. When it was run without stemming process, we identified 12,363 unique terms in ICL_list and 4,281 unique terms in WORDS_list.

Both corpora were classified based on 127 topics yielded in preliminary process. Taking the assumption that keywords determined by human experts are the term variants of a concept then aggregation of all terms appeared in each class were taken as the terms of representative vector of each class. The resulted classes of ICL-corpus was called *ALL* while the resulted classes of WORDS-corpus was called *WL*. The frequency matrix of topic-term needed in rough set phase was created based on these classes.

4.2 Rough Set Phase

This phase was conducted in order to generate the lower set *Lr*, upper set *Ur*, and boundary set *BNr* of each class; *RS* refers to all three sets. These sets were possible to be created using (1), (2), and (3) when tolerance matrix was ready.

The tolerance matrix was created based on algorithm explained in [8]. It needed topic-term frequency matrix as the input, then the occurrence binary matrix *OC matrix*, co-occurrence matrix *COC matrix*, and tolerance binary matrix *TOL matrix* were generated in sequence manner by employing (4), (5), and (6) respectively. Note that $tf_{i,j}$ denotes the frequency of term j in topic i and θ is the co-occurrence threshold of terms.

$$oc_{i,j} = 1 \quad \Leftrightarrow \quad tf_{i,j} > 0 \ . \tag{4}$$

$$coc_{x,y} = \text{card}(OC^x \text{ AND } OC^y) \ . \tag{5}$$

$$tol_{x,y} = 1 \quad \Leftrightarrow \quad coc_{x,y} \geq \theta \ . \tag{6}$$

4.3 Evaluation Phase

In this phase, all resulted sets were compared across the other, i.e. ICL_list vs. WORDS_list, ALL vs. WL, ALL vs. RS, WL vs. RS, and between RS (Lr vs. BNr, Ur vs. Lr, and Ur vs. BNr). The objective is to get the amount of terms appeared in both compared sets. These comparisons were conducted for co-occurrence threshold θ between 1 to 75. From each comparison at particular θ value, we got 127 values which were the value of each class. The average value was then computed for each θ value as well as for all θ value.

Recall and precision are measures commonly used in information retrieval field to evaluate the system performance. Recall R is the fraction of relevant documents that are retrieved while precision P is the fraction of retrieved documents that are relevant [7]. Suppose *Rel* denotes relevant documents and *Ret* denotes retrieved documents, then recall R and precision P are defined as follow

$$R = \frac{|Rel \bigcap Ret|}{|Rel|} \qquad P = \frac{|Rel \bigcap Ret|}{|Ret|} \ . \tag{7}$$

In this study, both measures were used for the *terms* rather than *documents*. That is to say, by considering WL as the ground truth, then recall R is the fraction of relevant terms that are retrieved while precision P is the fraction of retrieved terms that are relevant. Based on the definition, better recall value is preferred than better precision value because better recall value will ensure the availability of important terms in the set.

5 Analysis

With regard to the process of developing WORDS_list, the fact that ICL_list could cover almost all WORDS_list terms was not surprising. It was interesting though that there were some terms of WORDS_list did not appear in ICL_list; 17 terms yielded by the process without stemming task and 11 terms yielded by the process with stemming task. By examining those terms, we found that the *CS stemmer* could only handle the formal terms (6 terms) and left the informal terms (5 terms) as well as the foreign term (1 term); the other terms caused by typographical error (5 terms) in ICL_corpus.

Despite the fact that CS stemmer succeeded in reducing the number of terms of ICL_list (23.50%) as well as of WORDS_list (20.81%), it reduced the average of recall in each class of ALL about 0.64% from 97.39%. We noticed that the average of precision in each class of ALL increased about 0.25%, however the values themselves were very small (14.56% for process without stemming task and 14.81% for process with stemming task). From these, we could say that the ICL_list was still too noisy of containing many unimportant terms in describing particular topic.

5.1 ALL vs. RS

Table 1 shows the average values of comparison process between *ALL vs. RS* and *WL vs. RS* in percentage. The values of ALL-Ur for process with and without

Table 1. Average of Co-occurrence Terms Between Sets

	With Stemming			Without Stemming		
	Ur (%)	Lr (%)	BNr (%)	Ur (%)	Lr (%)	BNr (%)
(1)	(2)	(3)	(4)	(5)	(6)	(7)
ALL	100.00	5.00	95.00	100.00	4.43	95.57
WL$_{Recall}$	97.64	5.55	92.08	97.55	4.64	92.91
WL$_{Precision}$	13.77	27.49	13.50	14.13	26.30	13.75

stemming task, which are 100%, made us confident that the TRSM model has been employed correctly.

The low values of ALL-Lr (5% and 4.43%) and the high values of ALL-BNr (95% and 95.57%) compared with the low values of WL$_{Recall}$-Lr (5.55% and 4.64%) and the high values of WL$_{Recall}$-BNr (92.08% and 92.91%) indicate that rough sets theory seemed to work in accordance with the natural way of human thinking. From the values of WL$_{Recall}$, we could learn that it was possibly the case happened during topic assignment, that only limited number of terms could be considered precisely belong to a particular topic while numerous of others could not, e.g. were in uncertain condition. It was supported by the fact that many times the human experts seemed to encounter difficulty in determining keywords during annotation process. We came into this from the data that rather than listing the keywords, they chose sentences on the text or even made their own sentences. By doing this, they did not define specific terms as the highly related terms with particular topic but mentioning many other terms in the form of sentences instead. From this, we can say that the rough set theory is able to model the natural way of topic assignment conducted by human.

From Table 1, we can see that all values in column 3 are higher than all values in column 6 while all values in column 4 are lower than all values in column 7. Hence, it seems that employing stemming task could retrieve more terms considered as the "core" terms of a concept and at the same time reduce the number of uncertain terms retrieved.

5.2 WL vs. RS

Table 1 shows us that value of WL$_{Recall}$-Ur of process with stemming is higher than the process without stemming. It supports our confidence so far that stemming task with CS stemmer would bring more benefit in this framework of study.

Despite the fact that better recall is preferred than better precision, as we explained in 4.3, we noticed that the values of WL$_{Precision}$-Ur are small (13.77% and 14.13%). With regard to (7), they were calculated using equation $P = \frac{|WL \bigcap Ur|}{|Ur|}$. Based on the equation, we can expect to improve the precision value by doing one, or both, of these: (1) increasing the co-occurence terms of WL and Ur or (2) decreasing the total number of Ur. Suppose we have a constant number of Ur (after setting up the θ at a certain value), then what we should do to improve the precision is increasing the number of co-occurence terms, i.e. increasing the availability of relevant terms in Ur.

It has been explained in section 4.2 that the topic-term frequency matrix was used as the input of generating the tolerance class. It means, the weighting scheme was solely based on the term frequency of occurrence in particular topic. By this fact, the precision value is possible to be enhanced by improving the weighting scheme.

5.3 Tolerance Value

From ALL-Ur comparison with stemming, we also found that there was indication that Ur set enriched ALL set; it was based on the average value of co-occurrence terms over Ur that was 70.79% for θ value 1 to 75. In fact, the average value was started from 4.02% for $\theta = 1$ and getting higher up to 99.33% for $\theta = 75$. Note that the smaller the average value means the possibility of Ur set enrichs ALL set is higher. With regard to (6), it is reasonable that increasing the θ value will reduce the number of total terms in Ur set and increase the average value of co-occurrence terms between ALL-Ur over Ur. The important point of this is the possibility of enriching a concept getting lower by increasing the θ value.

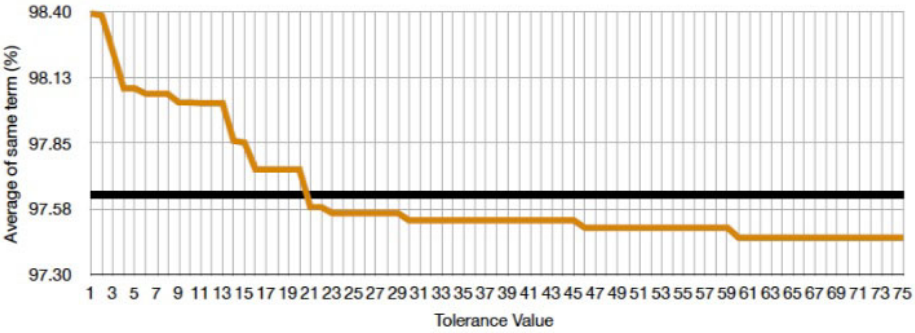

Fig. 2. The WL_{Recall}-Ur comparison

Figure 2 is the graph of co-occurrence terms between WL set and Ur set over WL for θ value 1 to 75. It is clear that after dramatic changes the graph starts to stable at tolerance value 21. Looking at the average number of terms in Ur set at $\theta = 21$ was also interesting. It is 733.79 terms, which means reducing 92.24% of the average number of terms in Ur set at $\theta = 0$ which is 9.458 terms. From Fig. 2, we can also see that the average number of co-occurrence terms at $\theta = 21$ is 97.58%, which is high. By this manual inspection, we are confident to propose $\theta \geq 21$ to be used in similar framework of study. However, automatically setting the tolerance value is suggested, especially while considering the nature of mailing list, i.e. growing over the time.

5.4 ICL_list vs. Lexicon

Lexicon is vocabulary of terms [7]. The lexicon used by CS stemmer in this study consists of 29,337 Indonesian base words. Comparison between ICL_list and Lexicon showed that there was 3,321 co-occurrence terms. In other words, 64.89% of ICL_list was different from Lexicon. Out of 6,137 terms, we analyzed the top 3,000 terms with respect to the document frequency.

We identified that the biggest problem (37.3% of terms) was caused by foreign language; most of them was English. Next problems were the colloquial terms which was 26.1% of terms and proper nouns which was 22.73% of terms. Combination of foreign and Indonesian terms, e.g. *workshopnya*, was considered as colloquial terms. We also found that the CS stemmer should be improved as there were 19 formal terms left unstemmed in ICL_list. Finally, we suggested 5 terms to be added into Lexicon and 8 terms into stopword-list.

6 Conclusion

This paper was thought as the fundamental basis that generating an automatic thesaurus using rough sets theory is a promising way. There was indication that it could enrich a concept and proved to be able to cover the important terms that should be retrieved by automatically process of system, even though foreign languages, colloquial terms and proper nouns were identified as big problems in main corpus. We noticed that CS stemmer as a version of Indonesian stemming algorithm was able to reduce the total number of index terms being processed and improved the recall of Ur as it was expected, however it should be upgraded. In this paper, we also proposed θ value ≥ 21 for similar framework of study, as well as suggesting some terms to be added into Lexicon and stopword-list.

There is much work to do related with this study, such as (1) to improve the weighting scheme hence it is not only based on term frequency of occurrence, (2) to upgrade the CS stemmer to handle the formal terms better, (3) to find a way in dealing with the foreign terms and colloquial terms, and (4) to set the θ value automatically, particularly by considering the nature of mailing list.

Acknowledgments. This work is partially supported by the National Centre for Research and Development (NCBiR) under Grant No. SP/I/1/77065/10 by the Strategic scientific research and experimental development program: "Interdisciplinary System for Interactive Scientific and Scientific-Technical Information", specific Grant Agreement Number-2008-4950/001-001-MUN-EWC from European Union Erasmus Mundus "External Cooperation Window" EMMA, and grants from Ministry of Science and Higher Education of the Republic of Poland (N N516 368334 and N N516 077837). We also thank Faculty of Computer Science, University of Indonesia, for the permission of using the CS stemmer.

References

1. Adriani, M., Asian, J., Nazief, B., Tahaghogi, S.M.M., Williams, H.E.: Stemming Indonesian: A Confix-Stripping Approach. ACM Transactions on Asian Language Information Processing 6(4), 1–33 (2007), Article 13
2. Asian, J.: Effective Techniques for Indonesian Text Retrieval. Doctor of Philosophy Thesis. School of Computer Science and Information Technology. RMIT University (2007)
3. Gaoxiang, Y., Heling, H., Zhengding, L., Ruixuan, L.: A Novel Web Query Automatic Expansion Based on Rough Set. Wuhan University Journal of Natural Sciences 11(5), 1167–1171 (2006)
4. Kawasaki, S., Nguyen, N.B., Ho, T.B.: Hierarchical Document Clustering Based on Tolerance Rough Set Model. In: 4th European Conference on Principles of Data Mining and Knowledge Discovery, pp. 458–463. Springer, London (2000)
5. Komorowski, J., Pawlak, Z., Polkowski, L., Skowron, A.: Rough Sets: A Tutorial. In: Rough Fuzzy Hybridization: A New Trend in Decision-Making, pp. 3–98. Springer, Singapore (1998)
6. Lassila, O., McGuinness, D.: The Role of Frame-Based Representation on the Semantic Web. Technical Report KSL-01-02, Knowledge System Laboratory, Standford University (2001)
7. Manning, C.D., Raghavan, P., Schutze, H.: An Introduction to Information Retrieval. Cambridge University Press, England (2009)
8. Nguyen, H.S., Ho, T.B.: Rough Document Clustering and the Internet. In: Pedrycz, W., Skowron, A., Kreinovich, V. (eds.) Handbook of Granular Computing, pp. 987–1003. John Wiley & Sons Ltd., Chichester (2008)
9. National Institute of Standards and Technology, `http://www.nist.gov/srd/nistsd23.cfm`
10. Nguyen, H.S.: Approximate Boolean Reasoning: Foundations and Applications in Data Mining. In: Peters, J.F., Skowron, A. (eds.) Transactions on Rough Sets V. LNCS, vol. 4100, pp. 334–506. Springer, Heidelberg (2006)
11. Pawlak, Z.: Rough Sets. International Journal of Computer and Information Science 11(5), 341–356 (1982)
12. Pawlak, Z.: Some Issues on Rough Sets. In: Peters, J.F., Skowron, A., Grzymała-Busse, J.W., Kostek, B.z., Świniarski, R.W., Szczuka, M.S. (eds.) Transactions on Rough Sets I. LNCS, vol. 3100, pp. 1–58. Springer, Heidelberg (2004)
13. Skowron, A., Stepaniuk, J.: Tolerance Approximation Spaces. Fundam. Inf. 27(2-3), 245–253 (1996)
14. Vega, V.B.: Information Retrieval for the Indonesian Language. Master thesis. National University of Singapore (2001) (unpublished)
15. Virginia, G., Nguyen, H.S.: Automatic Ontology Constructor for Indonesian Language. In: 2010 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology, pp. 440–443. IEEE Press, Los Alamitos (2010)
16. Virginia, G., Nguyen, H.S.: Investigating the Potential of Rough Sets Theory in Automatic Thesaurus Construction. In: 2011 International Conference on Data Engineering and Internet Technology, pp. 882–885. IEEE, Los Alamitos (2011)