I.J. Information Technology and Computer Science, **2016, 11, 1-7**
Published Online November 2016 in MECS (http://www.mecs-press.org/)
DOI: 10.5815/ijitcs.2016.11.01

Clustering Undergraduate Computer Science Student Final Project Based on Frequent Itemset

Lusi Maulina Erman
Computer Science Department of Bogor Agricultural University, Bogor, 16680 Indonesia
E-mail: lusimaulina@gmail.com

Imas Sukaesih Sitanggang
Computer Science Department of Bogor Agricultural University, Bogor, 16680 Indonesia
E-mail: imas.sitanggang@ipb.ac.id

***Abstract*—**Abstract is a part of document has an important role in explaining the whole document. Words that frequently appear can be used as a reference in grouping the final project document into categories. Text mining method can be used to group the abstracts. The purpose of this study is to apply the method of association rule mining namely ECLAT algorithm to find most common terms combination and to group a collection of abstracts. The data used in this study is documents of final project abstract in English of undergraduate computer science student of IPB from 2012 to 2014. This research used stopwords about common computer science terminology, applied association rule mining with support of 0.1, 0.15, 0.2, 0.25, 0.3, and 0.35, and used k-Means clustering with number of cluster (k) of 10 because it gives the lowest SSE. This research compared the value of support, SSE, the number of cluster members, and purity value in each cluster. The best clustering result is data with additional stopwords and without applying association rule mining, and with k is 10. The SSE result is 23 485.03, and with purity of 0.512

Index Terms—Abstract, association rule mining, frequent itemset, K-Means, purity.

I. Introduction

Abstract is a part of document has an important role in explaining the whole document. Words that frequently appear can be used as a reference in grouping the final project document into categories. Text mining method can be used to group the abstracts. One of techniques for grouping text documents based on the frequency of occurrence of keywords is clustering. One grouping technique is clustering. Clustering is a process of grouping together entities from an ensemble into classes of entities that are similar in some sense [2]. Grouping the data set can be constituted by the relationship between keywords in a text document. The keywords are analyzed by gathering the keywords that coming together and finding the association relationship of them by using association rule mining techniques [8]. This research is grouping undergraduate Computer Science student final project based on frequent itemset. ECLAT algorithm of association rule mining is used to find most common terms combination (frequent itemset). And K-Means is used to group a collection of abstracts based on frequent itemset. The purpose of this study is to apply the method of association rule mining namely ECLAT algorithm to find most common terms combination (frequent itemset) and to group a collection of abstracts.

The present work is organized as follows. In section II, we briefly mention some existing works related to clustering documents and or with association rule mining. In Section III, we briefly describe the data and the concepts of the methods used in this study. In section IV, we describe the results and explanation of each method sequentially according to previous section. And finally we conclude the topic on section V.

II. Related Work

Major challenges in document clustering are very high dimensionality of the data and very large size of the databases. Few works of searching for documents relevant text of the many documents has been observed with text clustering and frequent itemsets.

Beil et al [1] use frequent item (term) sets for text clustering. Such frequent sets can be efficiently discovered using algorithms for association rule mining. To cluster based on frequent term sets, they measure the mutual overlap of frequent sets with respect to the sets of supporting documents. They present two algorithms for frequent term-based text clustering, FTC which creates flat clusterings and HFTC for hierarchical clustering. An experimental evaluation on classical text documents as well as on web documents demonstrates that the proposed algorithms obtain clusterings of comparable quality significantly more efficiently than state-of-the art text clustering algorithms.

Fung et al [6] use the notion of frequent itemsets, which comes from association rule mining, for document clustering. The intuition of clustering criterion is that each cluster is identified by some common words, called frequent itemsets, for the documents in the cluster. Frequent itemsets are used to produce a hierarchical topic tree for clusters. By focusing on frequent items, the

dimensionality of the document set is drastically reduced. This method outperforms best existing methods in terms of both clustering accuracy and scalability.

While Steinbach et al [14] presents the results of an experimental study of some common document clustering techniques. They compare the two main approaches to document clustering, agglomerative hierarchical clustering and K-Means. Hierarchical clustering is often portrayed as the better quality clustering approach, but is limited because of its quadratic time complexity. In contrast, K-means and its variants have a time complexity which is linear in the number of documents, but are thought to produce inferior clusters. Sometimes K-means and agglomerative hierarchical approaches are combined so as to get the best. They propose an explanation for these results that is based on an analysis of the specifics of the clustering algorithms and the nature of document data

III. Data and Method

This section consisting of two, describes the data and methods that were used to grouping undergraduate Computer Science student final project based on frequent itemset. The information of methods presented sequentially in this section.

A. Data

The data used in this study is documents of final project abstract in English of undergraduate Computer Science student of IPB from 2012 to 2014 as many as 346 documents.

B. Method

This research method consists of 4 steps: are data preprocessing, association rule mining, K-Means clustering, and clustering document analysis. The steps shown in Figure 1.

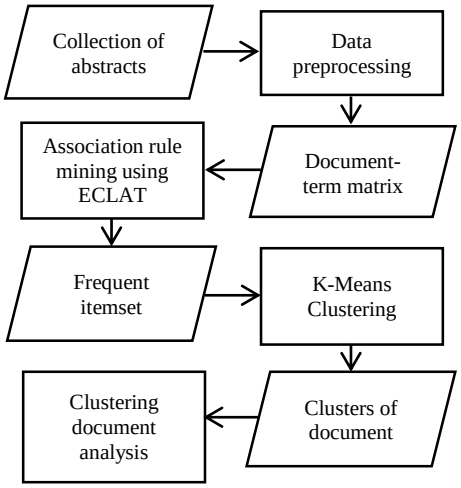

Fig.1. Steps of research

Data preprocessing consists of 6 steps: are case folding, remove punctuation and numbers, filtering, strip whitespace, stemming, and creating document-term matrix.

1. Case folding is changing all the letters in the document to lowercase [5].
2. Removing punctuation and number in the data aim to eliminate numbers and punctuations that have no analytic value.
3. Filtering is removing stopword (common words) that usually have no analytic value [5]. Stopwords used in this study come from stopwords in English which has been provided by the package tm in R. At this step also add stopwords related to computer science terms of a general nature.
4. Strip whitespace is removing extra whitespace as result of filtering step.
5. Stemming is changing the words to a derivative same representation by removing all affixes [5].
6. Document-term matrix gives the information about the frequency of occurrence of terms in the document collection of abstract data. Each row represents a document abstract data, while each column represents term in document collection. Overview of document-term matrix is shown in Figure 2.

	$term_1$	$term_2$	...	$term_n$
doc_1	$freq_{11}$	$freq_{12}$		$freq_{1n}$
doc_2	$freq_{21}$	$freq_{22}$		$freq_{2n}$
doc_3	$freq_{31}$	$freq_{32}$		$freq_{3n}$
...				...
doc_m	$freq_{m1}$	$freq_{m2}$	...	$freq_{mn}$

Fig.2. Document-term matrix

Association rule mining will generate $X \rightarrow Y$ shaped rules to determine how much the relationship between X and Y. It takes two measures to this rule, namely the support and confidence. Support is the possibility of X and Y appear in a transaction, while confidence is the possibility of Y when X also appears [8].

There are number of algorithms for finding frequent itemsets. Apriori is basic algorithm for finding frequent itemsets. But it takes more time for finding the frequent itemsets, It needs to scan the database again and again which is time consuming process. In this algorithm we need to calculate support and confidence, so ECLAT algorithm is developed to remove the limitations of Apriori, algorithm [10]. The proposed Equivalence CLASS Transformation (ECLAT) algorithm explores vertical data format [13]. By which it need to scan the database only once [10]. This research used algorithms ECLAT of association rule mining to find the combination of frequent itemset. A set of transactions is presented in vertical data format (TID, itemset), if TID is a transaction-id and itemset is the set of items bought in transaction TID [13]. A dataset consists of several items, followed by TID-list. Each item stated in the table *tid*-list vertically to form a vertical layout [15] shown in Figure 3.

TID	Item
1	A,B,E
2	B,C,D
3	C,E
4	A,B,C,D

→

A	B	C	D	E
1	1	2	2	1
4	2	3	4	3
	4	4		

Fig.3. Vertical layout [15]

ECLAT algorithm generate candidates with a depth-first search and use the intersection (point cut) tid-list an item of his [3]. Used intersection approach ECLAT algorithm is a bottom-up traversal approach illustrated in Figure 4 [15].

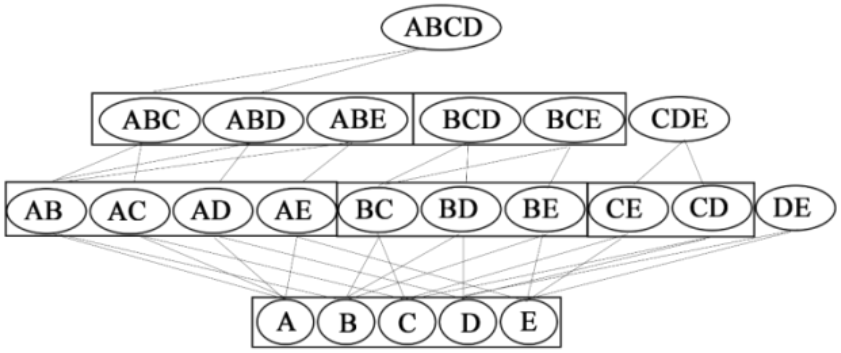

Fig.4. Bottom-up traversal [15]

Algorithm 1: ECLAT – Frequent Itemset Mining	
Input:	A transaction database *D*, A user specified threshold min_{sup} A set of atoms of a sublattice *S*
Output:	Frequent itemsets *F*

Procedure:
Eclat(S)
	for all atoms $A_i \in S$
	$T_i=\emptyset$
	for all atoms $A_j \in S$, with $j > i$ *do*
	$R= A_i \cup A_j$;
	$L(R)=L(A_i) \cap L(A_j)$;
	If $support(R) \geq min_{sup}$ *then*
	$T_i=T_i \cup \{R\}$;
	$F_{\|R\|}=F_{\|R\|} \cup \{R\}$;
	end
	end
	end
	for all $T_i \neq \emptyset$ do Eclat(T_i);

Fig.5. Pseudocode of ECLAT algorithm [7]

Pseudocode of ECLAT algorithm is shown in Figure 5. In this pseudocode, atom is term.

Clustering is a division of data into groups of similar objects. Each cluster consists of objects that are similar to each other and dissimilar to objects of other clusters. The goal of a good document clustering scheme is to minimize intracluster distances between documents, while maximizing inter-cluster distances [9]. In this research, K-Means clustering implemented in R language. K-means algorithm is popular because of its simplicity and efficiency. The complexity of each iteration is O(kn) similarity comparisons, and the number of necessary iterations is usually quite small [12].

The input of this K-Means in R is data results from dimensions reduction of the document-term matrix after ECLAT algorithm is executed and the value of k.

Sum of squared error (SSE) is used to determine the best value of k. Clustering with the smallest SSE is the best clustering result. SSE is defined as follows (1) [8]:

$$\mathrm{SSE} = \sum_{i=1}^{k} \sum_{x \in C_i} \mathrm{dist}(c_i, p)^2 \quad (1)$$

where *k* is the number of classes, *p* is a data object, C_i are objects in the cluster i, c_i is the centroid or center point of the cluster *i*, and *dist* is the distance function, Euclidean distance.

In this research, analysis of document clustering to measure the quality of the final clustering is using purity evaluation. Purity is one measure to measure the quality of external criterion-based clustering [11]. External criterion is a method for evaluating how well the clustering results by using a set of reference class as the representative user ratings, and the reference class is obtained from human judgment. Class tags in this

research were adopted from Fhattiya [4] is presented in Appendix A. The greater the purity value (closer to 1) shows the better the quality of the cluster. The formula to calculate the purity as in (2) [11]

$$\text{purity}(\Omega, \text{K}) = \frac{1}{N}\sum_{\text{k}} \max_{\text{j}} \left|\omega_k \cap c_j\right| \quad (2)$$

with $\Omega = \{\omega_1, \omega_2, \dots, \omega_k\}$ is cluster, $\text{K} = \{c_1, c_2, \dots, c_j\}$ is class, N is the number of documents, ω_k is objects in cluster ω_k , and c_j is objects in class c_j.

IV. Result and Discussion

This section describes the results and explanation of each method sequentially according to previous section.

A. Data Preprocessing

This study used 346 data abstract final documents in PDF file format. This PDF files are converted into txt file format manually by copying and pasting it. Data preprocessing consists of 6 steps: are case folding, remove punctuation and numbers, filtering, strip whitespace, stemming, and creating document-term matrix. The first step is case folding. This step is needed to be done to be processed easily in the next steps without noticing whether the capital letter or not. The next steps are remove punctuation and number, then filtering. Filtering is removing stopword (common words) that usually have no analytic value. This research used additional stopwords about common Computer Science terminology. The detail of additional stopword is presented in Appendix B.

After filtering, stopwords will disappear and leave a lot of whitespaces, so the strip whitespace step is needed to be done. Next is stemming. Stemming is removing affixes to generate the basic word. At this step, not all the basic word generated perfectly. Some generated words transformed into unlisted English dictionary words as shown in Table 1. However, this result will not bother the next steps: Association Rule Mining and Clustering Analysis. All data that has the same words will be generated as the same basic word.

Table 1. Samples of stemming step

Before stemming	After stemming
temporal	tempor
analyze	analyz
queries	queri

The last step of the preprocessing is creating the document-term matrix. This matrix gives information about the frequency of occurrence of a term in the document collection of abstract data, so that this matrix has a large dimension, as many as the number of document data multiplied by the number of constituent words (terms). To reduce the dimension of this matrix, removing low frequency terms is needed, using removeSparseTerm() with sparse value of 0.95. This value is considered quite well from another sparse value, because it leaves a considerable number of terms. It aims to get a term with many variations as the input of the association step. This sparse parameter of 0.95 indicates that the omitted terms are the ones that appear with 0 frequencies at 95% of the documents.

B. Association Rule Mining (ARM)

Frequent itemset is searched in the document-term matrix using ECLAT algorithm on package arules in R. Using the support parameter of 0.1, 0.15, 0.2, 0.25, 0.3, and 0.35, as limitation in comparison clustering data. The comparison between the support value and the number of term generated can be seen in Table 2.

Table 2. The comparison between the support value and the number of term generated

Data	Before adding *stopwords*		After adding *stopwords*	
	Frequent itemset	*Term*	*Frequent itemset*	*Term*
Without ARM	-	254	-	235
Support = 0.1	1089	101	122	84
Support = 0.15	337	47	38	33
Support = 0.2	143	26	15	14
Support = 0.25	86	21	11	11
Support = 0.3	51	14	5	5
Support = 0.35	31	12	4	4

Table 2 shown that the higher the value of the support, the less frequent itemset is generated. This result is used as a reference in the reduction of the term document matrix dimensions as clustering input.

C. K-Means Clustering

Input of this step is document-term matrix before and after association step with and without additional stopwords about common Computer Science terminology. The smallest SSE value is the k value of 10. SSE value for each clustering with k=2 to 10 shown in Figure 6 and 7. Therefore, the function of the K-Means given input k of 10. Before applying kmeans(), the set.seed() function is needed to determine how the producers of random numbers should be initialized (seeded) and achieve the results remain even if the program code is executed continuously. In this study the seed value used is 346, 122, 300, 255, and 50. Selection of the value of the function set.seed() can be used as correction material for further research and as a limitation in the comparison document clustering. To evaluate the results of clustering, purity evaluation calculation on the next step of clustering document analysis.

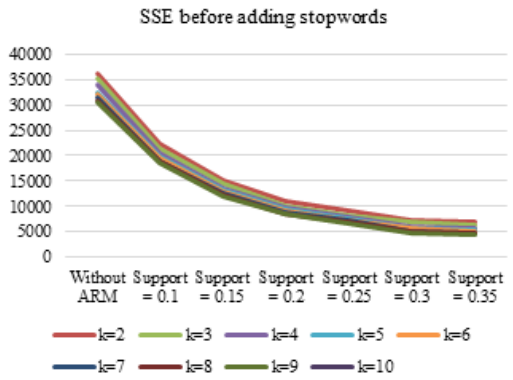

Fig.6. SSE value before adding stopwords

Fig.7. SSE value after adding stopwords

D. Clustering Document Analysis

In this step, the final clusters before and after applying association rule mining are compared by using support of 0.1, 0.15, 0.2, 0.25, 0.3, and 0.35. It aims to look at the influence of the association rule mining on document clustering. Comparison of the results of clustering is also done by calculating the value of the purity evaluation of each data clustering with k of 10. Purity is used to measure the quality of the clustering. The resulting purity evaluation value is the average of the results of the phase purity grades K-Means clustering using seed values mentioned above. The comparison of the purity value is given in Figure 8.

Figure 8 shown the purity value differences were not very significant. This means that the use of association rule mining cluster does not give better results than without association rule mining viewed from the value of the resulting purity evaluation. Although the data using association rule mining has the lowest value of SSE.

Stopword additions related to the term general Computer Science at the data also do not provide better results than the cluster in which the data are not given additional stopword when viewed from the value of the resulting purity evaluation. Although SSE value of the data with the addition of SSE stopword has a lower value than the data without adding stopword.

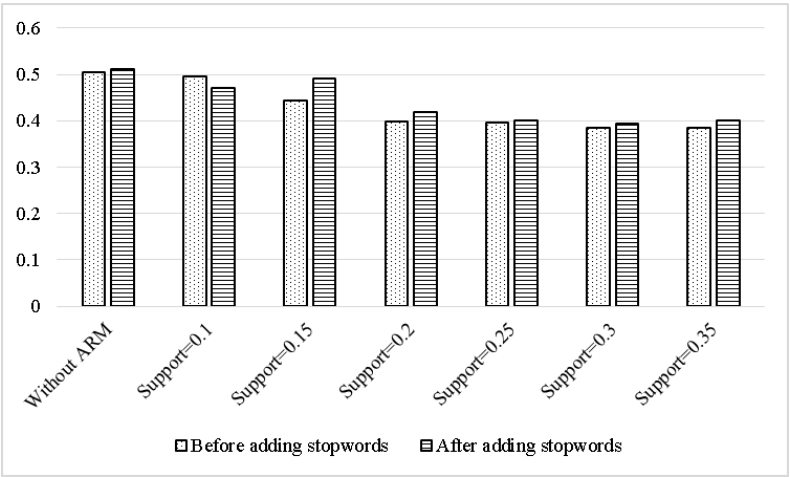

Fig.8. Purity evaluation

V. Conclusion

The conclusion of this study is a grouping of the final project document based on an abstract can be done by analyzing the association on terms that often appear in the abstract. This research applies association rule mining with support for 0.1, 0.15, 0.2, 0.25, 0.3, and 0.35. This research used K-Means clustering with number of cluster (k) of 10 because it gives the lowest SSE. This research compared the value of support, SSE, the number of cluster members, and purity value in each cluster. The best clustering result is data with additional stopwords and without applying association rule mining, and with k is 10. The SSE result is 23 485.03, and with purity of 0.512.

This study still has the deficiency that can be seen from the high SSE value and the purity value that is not even close to 1. Suggestion for further research is to be able to use other algorithms that can combine stages of association and clustering documents. Moreover, it can be done adding stopwords more complete and uses a variant of the K-Means algorithm for document clustering.

APPENDIX A CLASS TAGS WERE ADOPTED FROM FHATTIYA [4]

BI (Bioinformatics)	
DNA secuencing error correction	DNA sequence assembly
Dynamic programming	Graph algorithm
Metagenome fragment binning	Network pharmacology
SNP identification	
CI (Computer Intelligent)	
Artificial intelligence	Computational intelligence
Computer vision	Decision support system
Expert system	Fuzzy system
Genetic algorithm	Haar wavelet
Image processing	Machine learning
Neural network	Pattern recognition
Sound Processing	PNN
Prediction	Semantic web
SOM	Speech recognition
SVM	
DM (Data Mining)	
Associaton rule mining	Data warehouse
Clustering	Classification
Detection of outliers	Sequantial pattern mining
OLAP	Spatio temporal data mining
Spatial data mining	Web mining
Text mining	
IR (Information Retrieval)	
Ant colony algorithm	Belief Revison
Direct term feedback	Compression
Document clustering	Learning management system
Expert system	Naive bayes
IDF	Phonetic search
Indexing	Query expansion
Information retrieval	Question answering system
Neural network	Segmentation
Knowledge graph	Semantic indexing
Recall	Semantic smoothing
Recognition	Shortest path
Text summarization	Speling correction
NCC (Net Centric Computing)	
Arduino	Dynamic tagging
E-voting	Hexapod
HTTP	Security service
Network Security	Spatial query
Cryptography	Steganography
Obstacle Avoidance	Vector space model
Operating system	Watermarking
Parallel	Wireless ad hot networks
Peer-to-peer	Wireless microcontroller
RSA algorithm	
SEIS (Software Engineering & Information Science)	
Database	Software engineering
E-campaign	Recommendation systems
E-commerce	Software project management
E-goverment	Software quality assurance
E-learning	Software testing
Human computer interaction	Software testing
Interaction design	Test data generation
Knowledge management system	Usability
Digital library	Visualization information
Information systems	

APPENDIX B THE LIST OF ADDITIONAL STOPWORDS ABOUT COMMON COMPUTER SCIENCE TERMINOLOGY

accuracy	format	optimal	source
algorithm	formula	output	summary
application	function	perform	system
base	generate	power	technique
browser	implement	previous	technology
calculate	input	problem	tool
code	installation	process	transfer
compile	measure	program	unit
computer	method	provide	use
develop	model	purpose	utility
environment	need	require	research
execute	operate	research	
file	operation	result	

REFERENCES

[1] F. Beil, M. Ester, X. Xu. “Frequent term-based text clustering,” In Proceedings of the Eight ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2002, New York, NY, USA. pp. 436-442.

[2] R. Bordawekar, B. Blainey, R. Puri, *Analyzing Analytics*. San Rafael, CA: Morgan & Claypool, 2016.

[3] C. Borgelt, “Efficient implementations of Apriori and ECLAT” In Proceedings of the Workshop on Frequent Itemset Mining Implementations (FIMI-03), 2003, Melbourne, FL, USA. pp. 26-34.

[4] SR. Fhattiya. “Development of Data Warehouse and OLAP Application for Monitoring the Achievement of IPB Computer Science Students”. unpublished.

[5] R. Feldman, and J. Sanger. *The Text Mining Handbook Advanced Approaches in Analyzing Unstructured Data*. Cambridge, UK: Cambridge University Press, 2007.

[6] BCM. Fung, K. Wang, M. Ester, "Hierarchical document clustering using frequent itemsets," In Proceedings of the 2003 SIAM International Conference on Data Mining, 2003, San Francisco, CA, USA. pp. 59-70.
[7] X. Guandong, Z. Yanchun, L. Lin. *Web Mining and Social Networking: Techniques and Applications.* New York (US): Springe Science & Business Media. 2010.
[8] J. Han, M. Kamber, J. Pei. *Data Mining Concepts and Techniques*. Waltham, USA: Morgan Kaufmann Publisher, 2012.
[9] N.P. Katariya and M.S. Chaudhari. "Bisecting k-means algorithm for text clustering", International Journal of Advanced Research in Computer Science and Software Engineering, vol. 5 issue 2, pp.221-223, February 2015.
[10] M. Kaur and U. Grag. "ECLAT Algorithm for Frequent Itemsets Generation", International Journal of Computer Systems, vol.1 issue 3, pp. 82-84, December 2014.
[11] CD. Manning, P. Raghavan, H. Schutze. *An Introduction to Information Retrieval*. Cambridge, UK: Cambridge University Press, 2009.
[12] P. Shinde and S. Govilkar. "A systematic study of text mining techniques", International Journal on Natural Language Computing (IJNLC), vol. 4 no.4, pp. 54-62, August 2015.
[13] T. Slimani and A. Lazzez. "Efficient Analysis of Pattern and Association Rule Mining Approaches", I.J. Information Technology and Computer Science, vol.6 no.3, 2014, pp. 70-81.
[14] M. Steinbach, G. Karypis, V. Kumar. "A comparison of document clustering techniques," In KDD Workshop on Text Mining, 2000, Boston, MA, USA. pp. 1-20.
[15] MJ. Zaki, S. Parthasarathy, M Ogihara, W. Li. "New algorithms for fast discovery of association rules". In 3rd International Conference on Knowledge and Data Engineering, 1997, California, USA. pp. 283-286.

Authors' Profiles

Lusi Maulina Erman: Bachelor degree of Computer Science in Bogor Agricultural University. Her main interests include.

Imas Sukaesih Sitanggang: Doctor of Computer Science in Bogor Agricultural University. Her main research interests include data mining and spatial data processing.

How to cite this paper: Lusi Maulina Erman, Imas Sukaesih Sitanggang,"Clustering Undergraduate Computer Science Student Final Project Based on Frequent Itemset", International Journal of Information Technology and Computer Science(IJITCS), Vol.8, No.11, pp.1-7, 2016. DOI: 10.5815/ijitcs.2016.11.01