KINETIK, Vol. 2, No. 4, November 2017, Pp. 299-308
ISSN : 2503-2259
E-ISSN : 2503-2267

The Analysis of Proximity Between Subjects Based on Primary Contents Using Cosine Similarity on Lective

Muhammad Andi Al-rizki[*1], Galih Wasis Wicaksono[2], Yufis Azhar[3]
[1,2,3]Universitas Muhammadiyah Malang
andialrizki@webmail.umm.ac.id[*1], galih.w.w@umm.ac.id[2], yufis@umm.ac.id[3]

Abstract

In education world, recognizing the relationship between one subject and another is imperative. By recognizing the relationship between courses, performing sustainability mapping between subjects can be easily performed. Moreover, detecting and reducing any duplicated contents in several subjects will be also possible to execute. Of course, these conveniences will benefit lecturers, students and departments. It will ease the analysis and discussion processes between lecturers related to subjects in the same domain. In addition, students will conveniently choose a group of subjects they are interested in. Furthermore, departments can easily create a specialization group based on the similarity of the subjects and combine the courses possessing high similarity. In this research, given a good database, the relationship between subjects was calculated based on the proximity of the primary contents of the subjects. The feature used was term feature, in which value was determined by calculating TF-IDF (Term Frequency Inverse Document Frequency) from each term. In recognizing the value of proximity between subjects, cosine similarity method was implemented. Finally, testing was done utilizing precision, recall and accuracy method. The research results show that the precision and accuracy values are 90,91% and the recall value is 100%.

Keyword: *TF-IDF, Cosine Similarity, Primary Content, Lective*

1. Introduction

In considering curriculum as a set of plans to improve learning outcomes in higher education, there has been a continuous curriculum transformation in order to match the current and future industry needs. In Indonesia, the current curriculum for higher education is transformed from Competency Based Curriculum (KBK) to Higher Education Curriculum (KPT) affecting the development and the modification of teaching tools such as Subject Teaching Plan (RPS) and Teaching Execution Plan (RPP) [1].

RPS and RPP are important tools that should be generated by teachers as required by a rule issued by Indonesian Ministry of Education number 44 in 2015 (Permenristekdikti no. 44, year 2015) about National Standard for Higher Education (SN Dikti). Therefore, Lective (www.lective.id) was developed to help teachers developing RPS and RPP, so it provides easy maintenance to its cohesion and relationship. Furthermore, by Lective, the opportunity to develop RPS and RPP collaboratively by a group of teachers in the same or similar subjects will be available.

RPS as an important element in learning tools has been set in Permenristekdikti no. 44 in 2015 on article 12 about SN Dikti. RPS contains at least elements such as 1) the name of the department, 2) the name and code of the subject, 3) the number of credit unit (SKS), 4) semester number, 5) lecturer's name, 6) subject's learning outcome (CPMK), 7) planned final ability that has been obtained (KAD), 8) primary content, 9) learning method, 10) time allocation, 11) learning experiences obtained by students, 12) assessment criteria, 13) assessment indicators, 14) weighting assessment, and 15) references.

When designing RPS, lecturers often refer to the same study materials and literatures. Therefore, it is possible that two or more courses show similar primary content, either partially or fully. It is certainly not a problem in the parallel course model in the previous curriculum modes [2]. However, within the KPT, each subject is required to have their own CPMK; thus, subjects with the primary contents being interspersed or even identical with other subjects must be reviewed in 2016 KPT structure [1].

Al-rizki, M., Wicaksono, G., & Azhar, Y. (2017). The Analysis of Proximity Between Subjects Based on Primary Contents Using Cosine Similarity on Lective. Kinetik : Game Technology, Information System, Computer Network, Computing, Electronics, and Control, 2(4). doi:http://dx.doi.org/10.22219/kinetik.v2i4.271
Paper submitted on July 14, 2017; Revision on October 18, 2017; Received October 23, 2017

In order to assist the detection of the subject similarity, a feature of proximity analysis in Lective was developed based on the primary contents in the RPS. The primary contents were used because it can be easily validated by experts. This new feature was generally very useful for department, lecturers, and students. For the department, this feature can be a reference when restructuring KPT by combining the subjects having high proximity. For lecturers, this feature becomes a reference to avoid a proximity of the primary content. As for students, this feature can show the level of relevance between subjects.

In the study of document proximity based on text, there are two components that are usually used. Those are term weighting and similarity measure. A number of previous research has already used Term Frequency Inverse Document Frequency (TF-IDF) [3][4][5][6] for term weighting. Moreover, to measure similarity, the most popular method was the application of vector space model by utilizing cosine similarity measurement [7][8]. In cosine similarity, each document is considered as a vector composed by term weight. In order to measure similarity between two vectors, cosine measurement is performed [9]

The analysis of proximity between subjects in Lective was based on primary content in RPS. The primary content, afterwards, was extracted, resulting weighted terms by utilizing Term Frequency Inverse Document Frequency (TF-IDF). Furthermore, based on the weighted term, the similarity between subjects was measured using cosine similarity. The test was conducted for 10 subjects in Informatics Department of Universitas Muhammadiyah Malang. Consecutively, the result was validated and confirmed by experts in the subjects used.

2. Research Method

2.1 Requirement Analysis

In this phase, an analysis of the RPS structure was performed. Of the various elements in the RPS, it was finally determined that the primary content was suitable element to measure the proximity of the subject. In addition, the analysis was also performed on preprocessing techniques and document measurement techniques. Not all preprocessing techniques were implemented, only those which required by the extraction of information in the Lective were selected. As for the measurement of document similarity, the method having a good performance in Lective system was chosen.

To facilitate the user in analyzing the subject proximity, the value of proximity needed to be visualized in a certain way to provide better understanding.

2.2 System Design

The system design phase aims to design the flow of the proximity analysis as an additional feature of the Lective system. This phase began with the acquisition of primary content in the database, considered as documents. Subsequently, each document was preprocessed which included tokenizing, stopword removal and normalization. The next step was the calculation of term weight using TF-IDF followed by proximity measurement between documents using cosine similarity method. The result of the proximity calculation between the documents was stored to the Lective database and will be visualize in Lective system. Flow chart design of the proximity analysis between subject based on the primary content can be seen in Figure 1.

2.2.1 Preprocessing Design

As can be seen in Figure 1, the document preprocessing in the system consists of tokenization, stopword removal, and normalization. The result of document preprocessing is a set of standardize words stored into database, in which each word is called as term. The details of each preprocessing process are described below:

1. Tokenization
   Tokenization process is a process of dividing text data into piece or token. The token in text data is word which usually divided by white space. In the case that there are special symbols such as dot, question mark, and asterisk, these symbols are also considered as white space [10].
2. Stopword removal
   Stopword removal is a process for removing tokens that are identified as non-substantial token. Non-substantial tokens usually are types of word that appear very frequently and not be able to differentiate between one text data and other text data. [11]. In this research,

stopword is defined based on the structure of primary content which has been already stored in Lective.

3. Normalization
   The aim of normalization process modifies token which is not in standard form, such as not correctly spelled. For example, the existing web and website having same meaning, but with different writing.

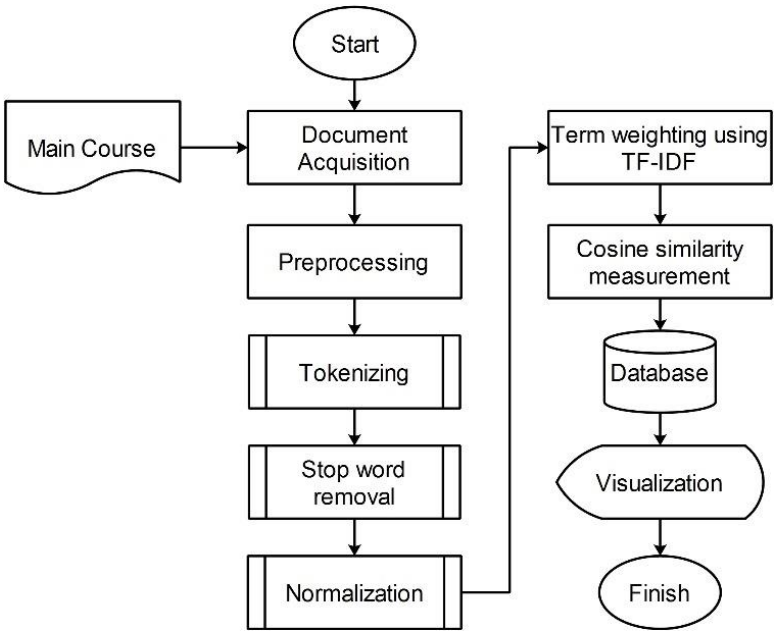

Figure 1. The Flowchart of Proximity Analysis to Measure Similarity between Subjects Based on Primary Content

2.2.2 Term Weighting Method

The calculation of term weight is conducted for all terms in the primary content in Lective. The aim of weighting is to measure the important of a particular term, related with primary content and subject. Moreover, the method that used is TF-IDF combining the term frequency and inverse document frequency [3][6]. Term frequency (TF) is related with the number of term appearance in a particular subject's primary content. Inverse document frequency (IDF) is related with the number of subject's primary content having a particular term. Hence, if there is a term appearing in a large number of subject's primary content, the IDF value becomes small. Equation 1 illustrates TF-IDF formula.

$$W_{d,t} = tf_{d,t} \times IDF_t \quad (1)$$

Where:
d = subject's primary content
t = term (word)
$tf_{d,t}$ = number of term appearance in subject's primary content
IDF_t = inverse document frequency

2.2.3 Subject Similarity Measurement

In order to measure the similarity between subjects' primary content, cosine similarity was utilized. Cosine similarity works by calculating cosine value between vectors. Each subject's primary content was considered as a vector [12]. Take for example, there were subject A and subject B, then there were vector A and vector B consecutively. Hence, the similarity between vector A and vector B was the cosine value of vector A to vector B. Moreover, since the method implements cosine concept, value 1 means vector A and vector B are exactly the same, but value 0 means vector A and vector B are completely different. Cosine similarity measurement is presented by Equation 2.

$$S_{D_i D_j} = \frac{\sum_{k=1}^{L}(W_{ik}\, W_{jk})}{\sqrt{\sum_{k=1}^{L} W_{ik}^2} \times \sqrt{\sum_{k=1}^{L} W_{jk}^2}} \quad (2)$$

Where:
S = similarity between subject *i* and *j*
D = subject's primary content (consider as a document)
W = weight
L = number of subjects in Lective
D_i = subject's primary content *i*
D_j = subject's primary content *j*

2.3 Testing Plan

Testing was conducted in order to quantify the efficiency and the effectiveness of the analysis of similarity between subjects in Lective. The utilized methods were precision, recall, and accuracy, which are widely utilized in measuring similarity level. Precision was useful for measuring the relevancy level related to the presented information. Recall was utilized to measure the ability of system to obtain all relevant information. Finally, accuracy measured the accuracy of the retrieved information [13]. The formula of precision, recall, and accurac*y* was presented by Equation 3, 4, and 5.

$$precision = \frac{TP}{TP + FP} \quad (3)$$

$$recall = \frac{TP}{TP + FN} \quad (4)$$

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (5)$$

Where:
FP = valid by system but invalid by experts
TP = valid by system and valid by experts
FN = invalid by system but valid by experts
TN = invalid by system and invalid by experts

3. Result and Discussion

3.1 Pre-processing of the Primary Content

As previously stated, each subject's primary content was preprocessed by tokenizing, removing stopwords, and normalizing. Table 1 contains the example of raw data, and Table 2 presents the preprocessing data which refers to Table 1.

Preprocessing implementation in Lective was written in PHP language. The list of stopwords was stored in p_stopword_list.txt file, so it was easy to maintain its content. Following this stage, normalization was performed by replacing words with different name but having same meaning. The list of words for normalization process was stored in p_wordreplae_list.json, Therefore, a new word list could be added. The implementation is shown in Figure 2.

3.2 Term Weighting

the process was followed by the calculation of TF-IDF value. It was executed by calculating the TF and IDF values of each term in each subject's primary content, followed by multiplying the results as TF-IDF value. The calculation example is presented in Table 3, and the implementation in PHP is illustrated in Figure 3.

3.3 Subject Similarity Measurement

Based on term weight at Table 3, the following step can be implemented that is measuring the similarity between subjects using cosine similarity. The example of similarity measurement is

shown in Table 4. In addition, the implementation of cosine similarity measure in PHP is shown in Figure 4.

After cosine similarity was calculated, the result was stored in the database. For efficiency reason, there was no need to save the similarity value between 2 documents which were absolutely not similar (when the similarity value was 0). Furthermore, only 4 closest subjects was displayed in Lective. As an example, if Table 5 shows the result of similarity measurement for Jaringan Komputer subject to other subjects, then the displayed subjects were 1) Manajemen Jaringan (0.2133 similar), 2) Komunikasi Data (0.1270 similar), 3) Pemrograman Berorientasi Obyek (0.0610 similar), and 4) Algoritma Pemrograman (0.0385 similar).

Table 1. List of Subject's Primary Contents Before Preprocessing

Subjects	Primary Contents
Jaringan Komputer (D1)	Konsep dasar switching dan konfigurasi. VLANs. Konsep Mekanisme Routing. Inter-VLAN Routing. Mekanisme Routing Static. Mekanisme Routing Dynamic. Routing Static dan Dynamic. Access Control Lists. Dynamic Host Configuration Protocol. Network Address Translation (NAT)
Manajemen Jaringan (D2)	Konsep Manajemen Jaringan dan Pengenalan Router Mikrotik. Dynamic Host Configuration Protocol (DHCP). Bridging. Routing. Wireless. Firewall. Quality of Service (QoS). Tunelling. Packet Flow Diagram. Web proxy
Komunikasi Data (3)	Model Komunikasi, Komunikasi Data, Jaringan Data. Protokol Layer Stack, Arsitektur Protokol Layer, Fungsi Pelayeran Protokol. Konsep, Prinsip Kerja, dan Fungsi Physical Layer, Teknik Encoding Decoding, Teknik Modulasi Demodulasi. Konsep, Fungsi dan Prinsip Kerja dari Data Link Layer, Data link Control Protocol (Control Flow & Error Control), HDLC. Konsep, Fungsi dan Prinsip Kerja dari Medium Access Layer, Multiplexing, Multiple Access, Topology (LAN dan WAN), Congestion Control.

Table 2. List of Subject's Primary Contents After Preprocessing

Subjects	Primary Contents
Jaringan Komputer (D1)	Switching konfigurasi vlans mekanisme routing inter vlan routing mekanisme routing static mekanisme routing dynamic routing static dynamic access control lists dynamic host configuration protocol network address translation nat
Manajemen Jaringan (D2)	Manajemen jaringan pengenalan router mikrotik dynamic host configuration protocol dhcp bridging routing wireless firewall quality service qos tunelling packet flow diagram web proxy
Komunikasi Data (D3)	Model komunikasi komunikasi data jaringan data protokol layer stack arsitektur protokol layer fungsi pelayeran protokol prinsip kerja fungsi physical layer teknik encoding decoding teknik modulasi demodulasi fungsi prinsip kerja data link layer data link control protocol control flow error control hdlc fungsi prinsip kerja medium access layer multiplexing multiple access topology lan wan congestion control

```
// tokenizing
$materi = str_replace($this->preprocessing->tokenList(),' ',strtolower(trim($d->materi)));
// stopword
$materi = preg_replace($this->preprocessing->stopwordList(), "", $materi);
// wordreplace
$materi = str_replace(array_keys($this->preprocessing->replaceList()), array_values($this-
>preprocessing->replaceList()), $materi);
```

Figure 2. Preprocessing Source Code

3.4 The Visualization of Analysis Result

Visualization became the last phase which was highly useful to present the analysis of subject' s similarity result on the Lective page. The page consisted of table showing the similarity result of a particular subject to other subjects. The technology used on each page was bootstrap template and jQuery. Figure 5 shows the example of table based page.

Beside the page based interface, there was also node based interface which presented a graph of a particular subject with its relation to other subjects. This page utilized vis.js technology. Figure 6 presents the example of the node based page.

3.5 Test Result

The test was executed by measuring the similarity of 10 subjects to other subjects in the database; subsequently, 4 most similar subjects to the selected test subjects were retrieved. The selected test subjects were belonging to a different level of study and a different subject's domain in Informatics Department of Universitas Muhammadiyah Malang. After performing measurement, the results were assessed by 2 experts. In order to avoid unfair evaluation perception, the 4 most similar subjects were sequenced randomly. In the existing of same idea by those 2 experts about similarity or dissimilarity between 2 subjects, it could be assured that the 2 subjects are similar or dissimilar. however, in facing dissimilar perception between the experts, another decision from the third expert will be required.

The validation result was written as decision matrices, shown by Table 6. Furthermore, it can be seen that the precision and the accuracy show identical values of 90.91%, and the recall value was 1%. On the other hand, the low recall value did not represent a negative issue since there were only 4 subjects which were considered in the evaluation; nevertheless, in fact there were many subjects which were also similar to the selected test subjects.

```
foreach ($data as $d) {
  // pre-processing
  // tokenizing
  $materi = str_replace($this->preprocessing->tokenList(), ' ', strtolower(trim($d->materi)));
  // stopword
  $materi = preg_replace($this->preprocessing->stopwordList(), "", $materi);
  // wordreplace
  $materi = str_replace(array_keys($this->preprocessing->replaceList()),
  array_values($this->preprocessing->replaceList()), $materi);

  $hit = array_filter(array_count_values(str_word_count($materi, 1)));
  $dokumen[] = array(
        'id_mk' => $d->id_matakuliah,
        'matakuliah' => strtolower($d->nama_matakuliah),
        'materi' => $materi,
        'tf' => $hit);
}
// load materi (array)
foreach ($docs as $d) {
        $kata = array_filter(explode(' ', $d['materi']));
        foreach ($kata as $key => $value) {
        $idf[$kata[$key]] = log(count($docs) / count($df[$kata[$key]]));
        }
}
foreach ($d['tf'] as $key => $value) {
        if(!empty($idf[$key])){
                $tfidf_d[$key] = $value * $idf[$key];
        }
}
```

Figure 3. TF-IDF Implementation in PHP

Table 3. The Example of Term Weighting Using TFIDF

Term	*TF*			*IDF*	*TFIDF*		
	D1	D2	D3		D1	D2	D3
switching	1	0	0	0.477121255	0.477121255	0	0
konfigurasi	1	0	0	0.477121255	0.477121255	0	0
vlans	1	0	0	0.477121255	0.477121255	0	0
mekanisme	3	0	0	0.477121255	1.431363764	0	0
routing	5	1	0	0.176091259	0.880456295	0.176091259	0
inter	1	0	0	0.477121255	0.477121255	0	0
vlan	1	0	0	0.477121255	0.477121255	0	0
static	2	0	0	0.477121255	0.954242509	0	0
dynamic	3	1	0	0.176091259	0.528273777	0.176091259	0
access	1	0	2	0.176091259	0.176091259	0	0.352182518

Table 4. The Example of Subject Similarity Measurement

Subject	D1 - D2	D1 - D3	D2 - D3
Similarity	0.062574859	0.014815754	0.006392416

```
public static function similarity(array $vec1, array $vec2)
{
    $vectorKey = array_keys(array_merge($vec1, $vec2));
    $dotProduct = 0;
    $magnitudeVec1 = 0;
    $magnitudeVec2 = 0;
    foreach ($vectorKey as $key)
    {
    $keyVec1Val = isset($vec1[$key])?$vec1[$key]:0;
    $keyVec2Val = isset($vec2[$key])?$vec2[$key]:0;
    $dotProduct += ($keyVec1Val * $keyVec2Val);
    $magnitudeVec1 += ($keyVec1Val * $keyVec1Val);
    $magnitudeVec2 += ($keyVec2Val * $keyVec2Val);
    }
    $magnitudeVec1 = sqrt($magnitudeVec1);
    $magnitudeVec2 = sqrt($magnitudeVec2);
    // hitung a / b
    $similarity = $dotProduct / ($magnitudeVec1 * $magnitudeVec2);
    return $similarity;
}
```

Figure 4. The Implementation of Source Code for Cosine Similarity

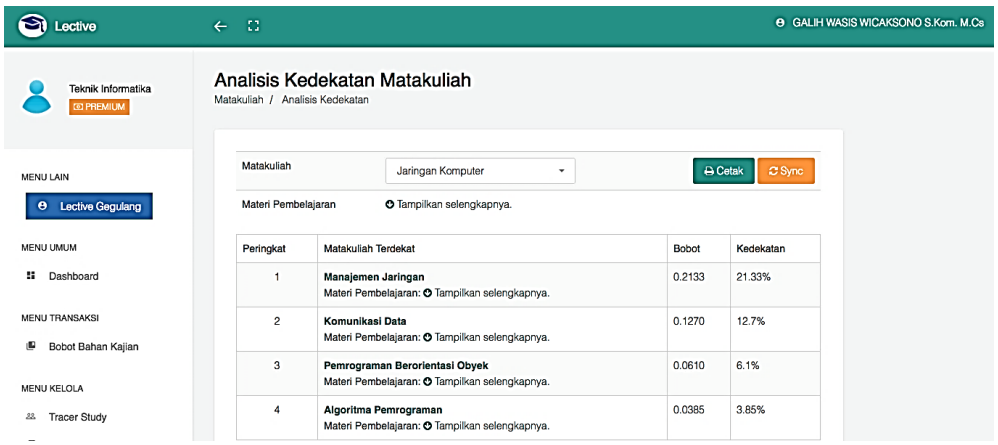

Figure 5. The Interface of Subject Similarity

Table 5. List of Similarity on the Analysis of Jaringan Komputer Subject in Informatics Department

No	Subjects	Closest Subjects	Closest Values	%	Departments
1	Jaringan Komputer	Manajemen Jaringan	0.2133	21.3%	Teknik Informatika
2	Jaringan Komputer	Komunikasi Data	0.1270	12.7%	Teknik Informatika
3	Jaringan Komputer	Pemrograman Berorientasi Obyek	0.0610	6.1%	Teknik Informatika
4	Jaringan Komputer	Algoritma Pemrograman	0.0385	3.85%	Teknik Informatika
5	Jaringan Komputer	Temu Kembali Informasi	0.0382	3.82%	Teknik Informatika
6	Jaringan Komputer	Sistem Terdisribusi	0.0140	1.4%	Teknik Informatika
7	Jaringan Komputer	Pemrograman Paralel	0.0113	1.13%	Teknik Informatika
8	Jaringan Komputer	Keamanan Jaringan	0.0062	0.62%	Teknik Informatika
9	Jaringan Komputer	Struktur Data	0.0060	0.6%	Teknik Informatika
10	Jaringan Komputer	Manajemen Proyek Perangkat Lunak	0.0058	0.58%	Teknik Informatika

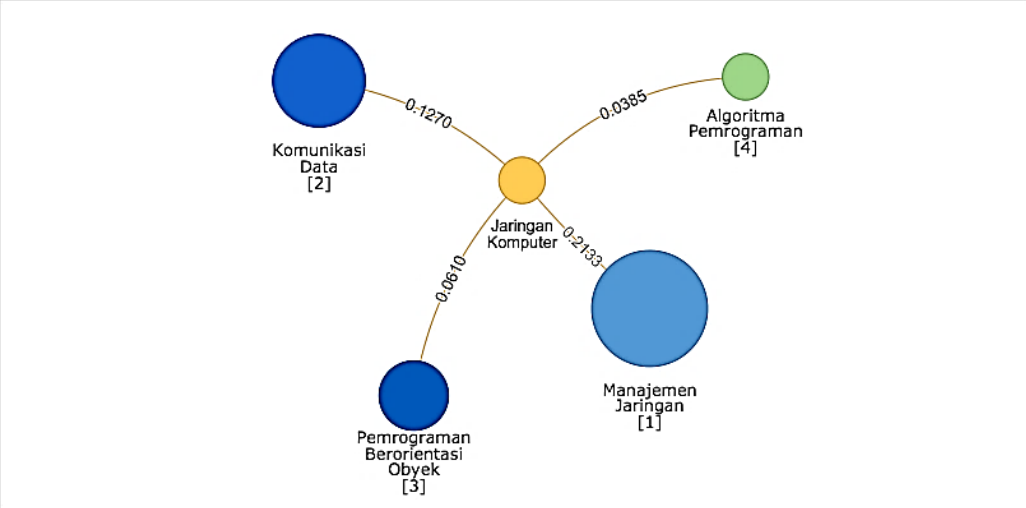

Figure 6. The Visualization of Subject's Similarity Result

Table 6. Decision Matrix of Expert's Validation

	Actual Valid	Actual Invalid
Predicted Valid	30 (TP)	3 (FP)
Predicted Invalid	0 (FN)	0 (TN)
Number of data/ Pair		33
Precision		90.91%
Recall		1%
Accuracy		90.91%

4. Conclusion

Based on research and test result related to the analysis of subject's similarity in Lective, it can be concluded that:

1. The development of subject similarity analysis feature in Lective based on primary content using TFIDF and cosine similarity is possible.

2. Based on the test result, the system performs high ability by reaching the precision and accuracy values of 90.91%. Although the recall value is low, it does not mean that the system is underperformed since there is limitation in retrieving similar subjects.

References

[1] Menristekdikti, "*Guide of Curriculum Plan for Higher Education*, *2nd ed*.," Jakarta: Indonesian Directorate General of Learning and Student Affairs Ministry of Research, Technology and Higher Education, 2016.
[2] Sutrisno & Suyadi, "*Curriculum Design of Higher Education; Referring to Indonesian National Framework,"* Bandung: PT Remaja Rosdakarya, 2016.
[3] O. Karmayasa, "*Implementation of Vector Space Model and Some Notation Methods on Term Frequency Inverse Document Frequency (TF-IDF),*" JELIKU (Electronic Journal of Computer Science of Universitas Udayana, 2012.
[4] M. Fitri, "*Designing Information Retrieval System using Combination of TF-IDF Weighing Method in Document Searching based on Bahasa Indonesia,"* Jurnal Sistem dan Teknologi Informasi, 2013.
[5] M. N. Saadah, R. W. Atmagi, D. S. Rahayu, and A. Z. Arifin, "*Text Document Retrieval System using TF-IDF and LCS Weighing,*" JUTI: Scientific Journal for Information Technology, Vol. 11, No. 1, Pp. 19, Jan. 2013.
[6] M. Nurjannah, H. Hamdani, and I. F. Astuti, "*Application of Term Frequency-Inverse Document Frequency (TF-IDF) Algorithm for Mining Text,"* Informatics Journal of Mulawarman, Vol. 8, No. 3, pp. 110–113, Jun. 2016.
[7] R. V. Imbar et al., "*Implementation of Cosine Similarity dan Algoritma Smith-Waterman to Detect Text Similarity*," Informatics Journal, Vol. 10, No. 1, Pp. 31–42, 2014.
[8] Sugiyamto, B. Surarso, and A. Sugiharto, "*Performance Analisys of Cosine Method and Jacard on Document Similarity testing*," Journal of Informatics Society, Vol. 5, No. 10, Pp. 1–8, 2014.
[9] W. H. Gomaa and A. A. Fahmy, "*A Survey of Text Similarity Approaches*," International Journal of Computer Applications, Vol. 68, No. 13, Pp. 975–8887, 2013.
[10] S. M. Weiss, N. Indurkhya, F. J. Damerau, and T. Zhang, "*Text Mining: Methods for Analyzing Unstructured Information*." Springer : New York, 2004.
[11] B. Liu, "*Web Data Mining*." Berlin, Heidelberg: Springer Berlin Heidelberg, 2011.
[12] R. V. Imbar, A. Adelia, M. Ayub, and A. Rehatta, "*Implementation Cosine Similarity and Smith-Waterman Algorithm to Detect Text* Similarity," Informatics Journal, Vol. 10, No. 1, 2015.
[13] M. Ridwan, H. Suyono, and M. Sarosa, "*Application Data Mining to Evaluate College Students' Academic Performance using Naïve Bayes Classifier Algorithm*," EECCIS, Vol. 7, Pp. 59–64, 2013.