# The Evaluation of Computer Science Curriculum for High School Education Based on Similarity Analysis

**Syaifudin Ramadhani[1,2], Mokhammad Amin Hariyadi[2], Cahyo Crysdian[2]**
[1]SMA Negeri 4 Malang, Indonesia
[2]Universitas Islam Negeri Maulana Malik Ibrahim, Malang, Indonesia

**Article Info**

***Article history:***

Received Sep 19, 2023
Revised Oct 15, 2023
Accepted Oct 30, 2023

***Keywords:***

Text Mining
Similarity Document
Computer Science Curriculum
Curriculum Document
Cosine Similarity
Word2Vec Similarity

## ABSTRACT

The government is currently developing regulations to regulate education curriculum For High School Students. In this regulation, curriculum standards have been created that can be developed by educators in schools. Computer science teachers at the school level develop a curriculum that has been set as a standard curriculum. However, measurable evaluation to optimize the development of the new curriculum has not been available yet. This research proposes a form of evaluation that can be used as a benchmark by analyzing the similarity of curriculum content developed by teachers using a text mining approach. This is conducted by comparing computer science documents with applicable documents, namely knowledge field documents. It is expected that the results of optimizing competency development in the computer science curriculum can be achieved better. The average similarity checking performance using Cosine Similarity and Word2Vec are 40.9850 and 97.3558 respectively. Meanwhile, in the process of fulfilling the knowledge sector, with Cosine Similarity an average percentage of 40.98% was obtained, and with Word2Vec an average percentage of 97.36% was obtained. The results of this trial will be used as a basis for measurable evaluation of teacher contributions to be able to develop the curriculum better according to the applicable curriculum. The results of this evaluation are also used by the government to make future curriculum evaluations more measurable and the standards used are clear and help facilitate curriculum development in schools.

***Corresponding Author:***

Cahyo Crysdian,
Magister Informatika, Universitas Islam Negeri Maulana Malik Ibrahim, Malang, Indonesia
Jl. Gajayana No.50, Dinoyo, Kec. Lowokwaru, Kota Malang, Jawa Timur 65144
Email: cahyo@ti.uin-malang.ac.id

## 1. INTRODUCTION

The new curriculum is a challenge for computer science teachers to be able to develop this curriculum to be implemented in each educational unit. Technical guidance or training on a scheduled and comprehensive basis is not provided by the government. The government has only provided one platform namely *Platform Merdeka Mengajar* (PMM) aiming to enable teachers practice independently, collaborate and share with other teachers and adopt all forms of good practice on that platform [1]. The government also issued regulations governing the implementation of learning at high schools which can be used as a basis for guiding curriculum development [2].

Competency development plays an important role in the education sector. The educational process is implemented in the form of a teaching and learning process with the noble aim of educating the younger generation [3]. Standard planning and rules are needed in the implementation of education curriculum development [1]. The standard rules are called the applicable curriculum. The development of the computer sciences Curriculum at the High School level (SMA) has been implemented since the Education Unit Curriculum (KTSP) was enacted in 2006. Where the computer sciences Curriculum Content presented contains basic competencies in computer science or computer sciences. The computer science curriculum which is a reference in developing this curriculum is also included in the 2020 computer science curriculum document with the topic Paradigms for Global Computing Education [4].

In 2022, the computer sciences curriculum was refined over time to be more specific, by developing the PMM, in which era the computer sciences curriculum was developed in the form of computer sciences competency elements including Generic Skills, Computational Thinking, Information and Communication Technology, Computer Systems, Computer Networks, and the Internet, Data Analysis, Algorithms and Programming, and the Social Impact of computer sciences. Of the eight competency elements in the computer sciences curriculum, the guidelines and standards are intended for alignment in the design, manufacture, and use of the applicable computer sciences curriculum, wherein standard curriculum development is used to provide direction and minimize development that is not following the applicable curriculum. Learning evaluation is also proposed to use two stages, including analysis of government policies and evaluation of learning practices based on the learning model system [5]. At the university level, the evaluation of policy MBKM (*Merdeka Belajar Kampus Merdeka*) is carried out using a qualitative approach. The data is collected by interviewing authorities to reveal that MBKM implementation can contribute to improve University independence and innovation.[6]. In the Arabic language curriculum at the university level, evaluation can be carried out through several stages, including the planning process, learning process, assessment process and learning evaluation.[7]. In the article released in the media, Indonesia, the government, has committed to developing aspects of education by increasing research and self-development supported by technology and information [8]. In terms of developing innovation at the university level, morality and integrity aspects are also needed in making a change in student paradigms. So far, the government merely prioritizes innovation but limits morality and integrity aspects. These aspects can also be a benchmark for government policy to change industrialist student paradigms to be more moral and characterful [9].

An investigation on measurements of text in Indonesian language documents utilizing different approaches, such as statistical methods and semantic analysis, is being conducted. The study focused on three parameters: Subjectivity Measurement (SM), Compression Rate (CR), and Processing Time (PT). The experimental results showed that the cosine similarity method achieved an SM of 83.46[10]. The research focused on processing keyword similarities in Indonesian online news documents to identify topics. The study utilized Bracewell's algorithm and the Top-N Keywords Selection algorithm. Three performance measurement scenarios were conducted, including Accuracy, Computing Time, and Human Evaluation. In the accuracy measurement experiment, the dataset consisted of articles from http://www.kompas.com published between 2011 and 2012. The training data included 979 items, and the testing data comprised 455 items from nine categories on the Kompas portal, covering a total of 559 topics. The experiment employed the cosine similarity method, which yielded a similarity performance of 95.26% [11]. The study focuses on using text mining techniques to analyze the performance of similarity measurements on novel documents. Three methods, namely cosine similarity, ISC Similarity, and Gaussian, are employed for this analysis. The dataset used in the study includes approximately 550 abstracts of technical reports published from 1991 to 2007 in computer science journals at the University of Rochester. Additionally, the DBLP dataset, consisting of titles from papers published by 552 active researchers in nine research fields, is included. These fields include databases, data mining, software engineering, computer theory, computer vision, operating systems, machine learning, networking, and natural language processing. The dataset also incorporates a collection of documents from the Reuters news agency in 1987 and 8280 web pages from various college computer science departments in the WebKB dataset. The experimental results reveal that the cosine similarity method achieves the best similarity performance, reaching 86.33% [12].

The study centers on a method for handling Indonesian text essays in order to simplify the scoring process for university essay exams. The method involves using Automated Essay Scoring

(AES) based on the Support Vector Regression (SVR) algorithm and the cosine similarity method to measure document similarity. Using a dataset of Indonesian language essay documents, consisting of 10 essay documents labeled as documents 1 to 10, the study determined the level of similarity between these documents. The results showed that the document ranked 3rd had the lowest similarity level, at 0.77%, while the document ranked 10th had the highest similarity level, at 39% [13]. In a research approach to text mining was explored, specifically focusing on automatic text summarization using the Vector Space Model. The study aimed to generate summaries, similar to abstracts, from scientific work documents or articles. The assessment data consisted of articles from computer sciences journals written in the Indonesian language. Certain criteria were applied to the journals, such as removing headers, footers, titles, sub-headings, tables, and figures. The articles were in PDF format. The experiment used five documents labeled D1 to D5 as test data. The experimental results revealed that the highest similarities were found between documents D1 and D4, with four similarities, while document D2 had no similarities with the other documents. Thus, the system's accuracy, particularly in generating automatic abstracts, was found to be lower compared to manual methods. The Vector Space Model method achieved a similarity level accuracy of only 40% in each tested document [14]. text mining techniques were utilized to identify essay tests. The research was further explored by Amalia et al. (2019) with slight content modifications on e-learning platforms and different methods. The method employed in the study was Latent Semantic Analysis (LSA). This approach aimed to assess the relevance of students' essay responses to the key answers provided by teachers. The trial involved three scenarios. Scenario 1 focused on calculating a 100% similarity level, indicating that the students' answers matched the answer key. Scenario 2 measured synonym similarity, where the terms used in the students' answers differed from the answer key but had the same meaning. Scenario 3 assessed a 0% similarity level, indicating that the students' answers were completely different from the answer key. The experimental results demonstrated that the system's accuracy surpassed the manual assessment conducted by teachers, achieving an accuracy rate of 83.3% [15].

In a study suggested a technique for identifying paraphrases in Malay with a text mining approach using the cosine similarity and Jaccard methods with the best similarity performance using the cosine similarity method of 86% [16], while a similar study to find out the similarity of an article document. in Indonesian from Wikipedia by calculating the cosine similarity of word vectors using the Word2Vec method with a similarity performance of 91% [17]. From the two studies with the best previous similarity results, this research will test the two methods and then compare the similarity optimization results in the 2013 computer science curriculum documents with computer science in senior high schools and evaluate the most effective factors, aspects and methods to be recommended to parties. related to evaluating the optimization of the computer science curriculum in the field so that good policies can be implemented in further curriculum development.

## 2. MATERIAL AND METHODS

### 2.1. CURRICULUM DATA

The data used in this study are teaching modules or RPP (*rencana pelaksanaan pembelajaran*/learning implementation plans) for computer science at the high school level. Teaching module data or lesson plans were obtained from computer science teaching teachers in Malang, for both public and private schools, and were also obtained from *Platform Merdeka Mengajar* (PMM). 11 data on state senior high schools in Malang city were obtained, while there are 1 data for private high schools in Malang. Eight (8) data from *Platform Merdeka Mengajar* (PMM) are used for teaching modules. Sample data can be seen in Table 1. In this table, it is shown that the input knowledge area sample data studied is in the knowledge area sub-topic category. Where the knowledge area consists of 18 knowledge areas and each knowledge area consists of several sub-topic categories. While the sample data used as input is as shown in Table 2.

Table 1. Topic-based knowledge area sample data

| No. | Input data sampel | Information |
|---|---|---|
| 1 | *Set dan bahasa o Bahasa reguler o Review deterministic finite automata (DFAs) o Nondeterministic finite automata (NFAs) o Kesetaraan DFA dan NFA o Review ekspresi reguler; • Persamaannya dengan finite automata o Properti penutupan o Membuktikan bahasa non-reguler, melalui lemma pemompaan atau cara alternatif • Bahasa bebas konteks o Push-down automata (PDA) o Hubungan PDA dan tata bahasa bebas konteks o Properti bahasa bebas konteks • Mesin Turing, atau model formal yang setara dari komputasi universal • Mesin Turing Nondeterministik • Hierarki Chomsky • Tesis Church-Turing • Komputabilitas • Teorema Rice • Contoh fungsi yang tidak dapat dihitung • Implikasi dari ketidakterhitungan* | Category Algorithm and Complexity Sub Topic Advanced Automata Theory and Computability |
| 2 | *Dasar-dasar I/O: handshaking, buffering, I/O terprogram, I/O berbasis interupsi • Struktur interupsi: divektor dan diprioritaskan, pengakuan interupsi • Penyimpanan eksternal, organisasi fisik, dan drive • Bus: protokol bus, arbitrase, direct- akses memori (DMA) • Pengantar jaringan: jaringan komunikasi sebagai lapisan lain dari akses jarak jauh • Dukungan multimedia • arsitektur RAID* | Category Architexture and Organization Sub Topic Interfacing and Communication |

Table 2. Computer Science Curriculum sample data

| No. | Input data sampel | Information |
|---|---|---|
| 1 | *Pada akhir fase E, peserta didik mampu memanfaatkan berbagai aplikasi secara bersamaan dan optimal untuk berkomunikasi, mencari sumber data yang akan diolah menjadi informasi, baik di dunia nyata maupun internet, serta mahir menggunakan fitur lanjut aplikasi perkantoran (pengolah kata, angka dan presentasi) beserta otomasinya untuk mengintegrasikan dan menyajikan konten aplikasi dalam berbagai representasi yang memudahkan analisis dan interpretasi konten tersebut….* | computer science Document of SMAN 6 Malang |
| 2 | *Unit pembelajaran TIK yang bersifat praktis (TIK sebagai tools) dan aplikatif seharusnya menjadi bagian dari program BimTIK sekolah, yang meliputi hal berikut.*<br>*1. Pengenalan pemakaian gawai untuk proses belajar-mengajar, yang tentunya sudah dilaksanakan di banyak sekolah dengan terpaksa saat pandemi melanda.*<br>*2. Pengenalan pemakaian aplikasi perkantoran untuk menunjang pelaporan, perhitungan, dan presentasi yang dibutuhkan di mata pelajaran apapun.*<br>*3. Pengenalan lingkungan e-learning yang diterapkan di sekolah: LMS dan kanal komunikasi yang ditetapkan sebagai standar komunikasi guru-siswa dalam belajar….* | computer science Document of SMAN 5 Malang |

Table 1 and Table 2 describes the sample data input for computer sciences documents at public and private high schools in Malang City. The data comes from the development of learning tools for computer sciences subjects from each school. In Table 1, in the input column the sample data taken is the source of the 2013 computer science curriculum document, while in Table 2 in the input column the sample data taken is a high school computer science document. The information column in Table 1 explains the Knowledge area categories in the 2013 computer science curriculum document, while the information column in Table 2 explains the school from which the document was taken.

At the system design stage, the system to be designed in this study consists of three stages, namely preprocessing text, similarity process, and calculating process for fulfilling SMA computer sciences Curriculum Documents with Knowledge Area can be shown at Figure 2.

Figure 2 explains the stages in system design when the first document data input section is an English text document that has been translated into Indonesian language text, there are eighteen Knowledge Areas including AL-Algorithms and Complexity, SP-Social Issues and Professional Practice, SF-Systems Fundamentals, SE-Software Engineering, SDF-Software Development Fundamentals, PL-Programming Languages, PD-Parallel and Distributed Computing, PBD-Platform-based Development, OS-Operating Systems, NC-Networking and Communication, IS-Intelligent Systems, IM-Information Management, IAS-Information Assurance and Security, HCI-Human-Computer Interaction, GV-Graphics and Visualization, DS-Discrete Structures, CN-Computational Science, and AR-Architecture and Organization. The second document data input is a text document in Indonesian. The document was obtained from computer sciences teachers in public high schools in Malang. The document is a computer science curriculum tool document that has been developed by each computer science teacher at the computer sciences Subject Teacher Consultation (MGMP) of Public and Private High Schools throughout the City of Lang.

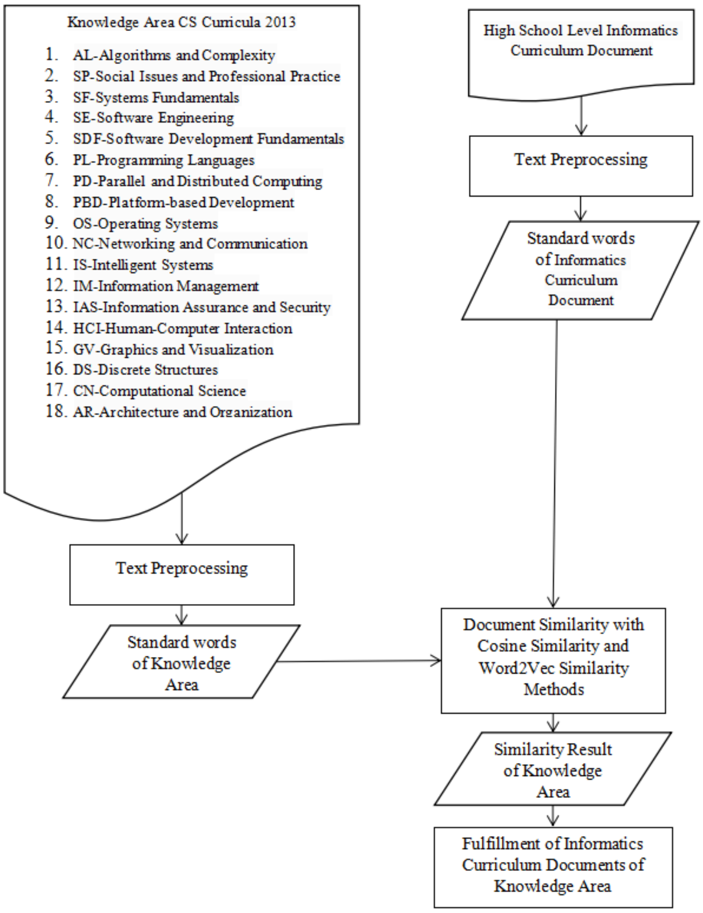
Knowledge Area CS Curricula 2013
1. AL-Algorithms and Complexity
2. SP-Social Issues and Professional Practice
3. SF-Systems Fundamentals
4. SE-Software Engineering
5. SDF-Software Development Fundamentals
6. PL-Programming Languages
7. PD-Parallel and Distributed Computing
8. PBD-Platform-based Development
9. OS-Operating Systems
10. NC-Networking and Communication
11. IS-Intelligent Systems
12. IM-Information Management
13. IAS-Information Assurance and Security
14. HCI-Human-Computer Interaction
15. GV-Graphics and Visualization
16. DS-Discrete Structures
17. CN-Computational Science
18. AR-Architecture and Organization
High School Level Informatics Curriculum Document
Text Preprocessing
Standard words of Informatics Curriculum Document
Text Preprocessing
Standard words of Knowledge Area
Document Similarity with Cosine Similarity and Word2Vec Similarity Methods
Similarity Result of Knowledge Area
Fulfillment of Informatics Curriculum Documents of Knowledge Area

Figure 2. System Design

The stages of text processing are carried out from the data input of the first document and the second document. These stages include the case folding process, tokenizing process, filtering process, and stemming process [13]. The purpose of this processing stage is to normalize the source text to be processed in the document similarity checking so that documents to be processed in the similarity process can be minimized in the form of basic terms [18]. The text processing process for Senior High School computer science curriculum can be seen in Figure 3.

## 2.2. TEXT PROCESSING

In Figure 3 the document source used is the Computer Science Curriculum at high school, while in Figure 4 the document source used is the 2013 Computer Science Curriculum. In the next stage in Figure 3 and Figure 4 is the Case Folding stage, the steps taken include: change the text to lower case (lower case) then carry out the process of removing numbers, removing punctuation marks, and finally the process of removing empty characters (spaces). In the filtering stage, the steps taken are to remove unnecessary words that often appear in large numbers (stop words). In the stemming stage, the steps taken are removing prefixes and word affixes to become standard terms or basic words. At the tokenizing stage, sentences are separated into word chunks. The result after processing is a group of basic words in Bahasa Indonesia [19]. The source document used as a comparison document is the 2013 Computer Science Curricula document that can be seen in Figure 4. The process stages and results obtained are the same as in Figure 3, namely group of basic words in Bahasa Indonesia.

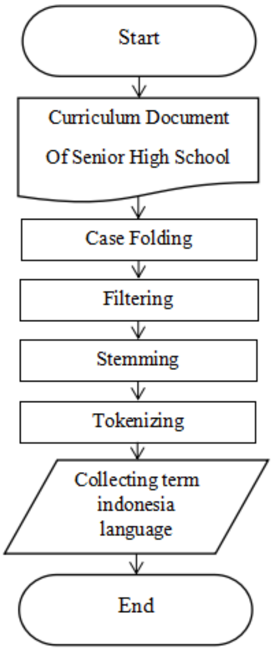
Start
Curriculum Document Of Senior High School
Case Folding
Filtering
Stemming
Tokenizing
Collecting term indonesia language
End

Figure 3. Flowchart of text preprocessing for Senior High School Computer Science Curriculum

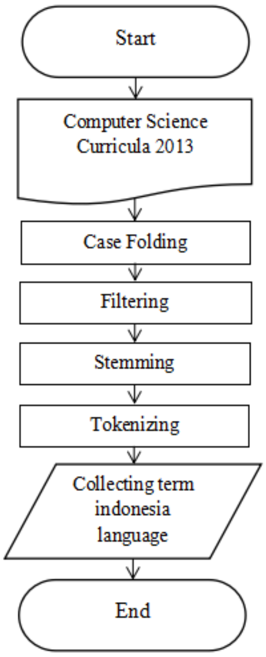
Start
Computer Science Curricula 2013
Case Folding
Filtering
Stemming
Tokenizing
Collecting term indonesia language
End

Figure 4. Flowchart of text preprocessing for Computer Science Curricula 2013

## 2.3. DOCUMENT SIMILARITY

Figure 4 explains the stages of the document similarity process using the cosine similarity method, while Figure 5 explains the stages of the document similarity process using the Word2Vec method. In Figure 4 and Figure 5 shows that the document similarity process is carried out by calculating the scalar multiplication result between the computer sciences curriculum document (Doc 1) and the knowledge area document (Doc 2) then calculating the sum of the results of the scalar multiplication of the two documents. The next step is to calculate the length of each document vector by squaring the weight of each term in each document, adding up the squared values and finally taking the squared, calculating the similarity of document 1 to document 2 with the formula adjusted to the actual variables in the document. The final step is sorting the results of the similarity calculations in Docs 1 and 2. The results of the document similarity process are in numerical form with a range of 0 to 1[16].

The Word2Vec similarity checking method is a technique used to represent each word in a context as a vector with N dimensions. In representing a word, Word2Vec uses a neural network to calculate the contextual and semantic similarity of each word represented in the form of a one-dimensional vector. hot encoded [17]. The document similarity process using the Word2Vec method explained in Figure 5. Figure 5 shows the process of document similarity checking using the Word2Vec method. At the first stage, the terms and the number of terms found are arranged using computer science document vectors and knowledge area documents. Then they are processed using the CBOW method. The output from CBOW is used as input at the Skip-gram stage. The results of the two techniques in this method produce document similarity levels in numerical from 0 to 1[17].

At the stage of fulfilling computer sciences curriculum documents with Knowledge Area is the final stage for calculating the fulfillment of computer sciences curriculum documents in the knowledge area. At this stage, the knowledge area documents that are taken into account are the sub-knowledge area topics. Where each knowledge area has several sub-competencies. The process of compliance calculation in each of these sub-knowledge areas can be seen in Figure 6.

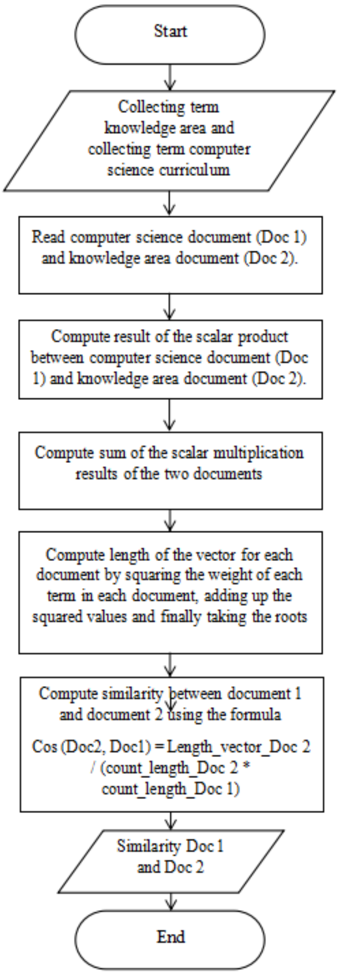
Start
Collecting term knowledge area and collecting term computer science curriculum
Read computer science document (Doc 1) and knowledge area document (Doc 2).
Compute result of the scalar product between computer science document (Doc 1) and knowledge area document (Doc 2).
Compute sum of the scalar multiplication results of the two documents
Compute length of the vector for each document by squaring the weight of each term in each document, adding up the squared values and finally taking the roots
Compute similarity between document 1 and document 2 using the formula
Cos (Doc2, Doc1) = Length_vector_Doc 2 / (count_length_Doc 2 * count_length_Doc 1)
Similarity Doc 1 and Doc 2
End

Figure 4. Flowchart of cosine similarity

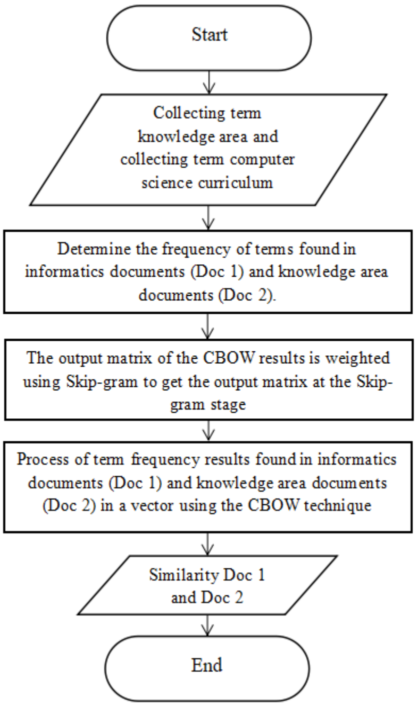
Start
Collecting term knowledge area and collecting term computer science curriculum
Determine the frequency of terms found in informatics documents (Doc 1) and knowledge area documents (Doc 2).
The output matrix of the CBOW results is weighted using Skip-gram to get the output matrix at the Skip-gram stage
Process of term frequency results found in informatics documents (Doc 1) and knowledge area documents (Doc 2) in a vector using the CBOW technique
Similarity Doc 1 and Doc 2
End

Figure 5. Flowchart of Word2Vec

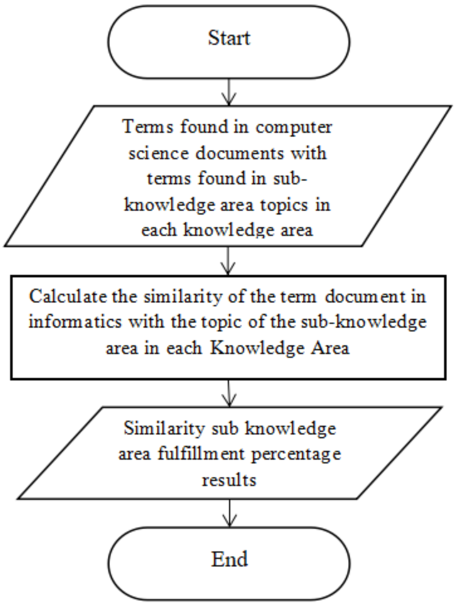
Start
Terms found in computer science documents with terms found in sub-knowledge area topics in each knowledge area
Calculate the similarity of the term document in informatics with the topic of the sub-knowledge area in each Knowledge Area
Similarity sub knowledge area fulfillment percentage results
End

Figure 6. Fulfillment results of similarity

In Figure 6, the process of calculating the fulfillment of the contents of computer science curriculum documents in the field of knowledge resulting from document similarity checks is carried out by calculating the similarity of documents between the terms contained in documents in the field of computer science and the topics of subfields of science in each Knowledge Area. At this stage, the process of calculating document similarity from the results of grouping computer sciences documents with sub-knowledge area topics in each knowledge area is done by calculating the similarity of each term. The result of calculating this similarity is the value of content fulfillment in each knowledge area sub-topic. The final results of calculating the fulfillment of computer sciences curriculum document content with Knowledge areas is the percentage of content fulfillment in each sub-knowledge area.

## 3. RESULTS AND DISCUSSION

The experiment was held on several stages. First, to help implement this system, the author uses the TKinter tool and programming language. Second, the type of analysis explained in this chapter, including the results of document similarity checking with input data based on terms, combinations of terms and sentences. The result of document similarity checking trial using cosine similarity and word2Vec can be seen in Table 3.

Table 3. Document similarity result

| No. | Document Name | Similarity result | |
|---|---|---|---|
| | | Cosine Similarity | Word2Vec |
| 1 | Doc. 1 vs. Doc. KA | 0.5040 | 0.9966 |
| 2 | Doc. 2 vs. Doc. KA | 0.4390 | 0.9977 |
| 3 | Doc. 3 vs. Doc. KA | 0.6040 | 0.9970 |
| 4 | Doc. 4 vs. Doc. KA | 0.2570 | 0.9730 |
| 5 | Doc. 5 vs. Doc. KA | 0.3610 | 0.9988 |
| 6 | Doc. 6 vs. Doc. KA | 0.5880 | 0.9965 |
| 7 | Doc. 7 vs. Doc. KA | 0.2970 | 0.9206 |
| 8 | Doc. 8 vs. Doc. KA | 0.4010 | 0.9443 |
| 9 | Doc. 9 vs. Doc. KA | 0.3360 | 0.9926 |
| 10 | Doc. 10 vs. Doc. KA | 0.4850 | 0.9990 |
| 11 | Doc. 11 vs. Doc. KA | 0.3710 | 0.9508 |
| 12 | Doc. 12 vs. Doc. KA | 0.3360 | 0.9498 |
| 13 | Doc. 13 vs. Doc. KA | 0.3690 | 0.9776 |
| 14 | Doc. 14 vs. Doc. KA | 0.5710 | 0.9739 |
| 15 | Doc. 15 vs. Doc. KA | 0.3210 | 0.9962 |
| 16 | Doc. 16 vs. Doc. KA | 0.3330 | 0.9990 |
| 17 | Doc. 17 vs. Doc. KA | 0.2800 | 0.9781 |
| 18 | Doc. 18 vs. Doc. KA | 0.4450 | 0.9436 |
| 19 | Doc. 19 vs. Doc. KA | 0.4020 | 0.9237 |
| 20 | Doc. 20 vs. Doc. KA | 0.4970 | 0.9625 |

Table 3 explains that the best similarity performance between each computer sciences document and knowledge area documents is obtained using the method Word2Vec and that the computer sciences document with the best similarity performance is Doc. 16. The best similarity result with the cosine similarity method was obtained in Doc. 3. Meanwhile, in the process of fulfilling the knowledge area in each document can be seen in Table 4.

Table 4. Results of fulfillment of knowledge area

| No. | Document Name | Fulfillment result | |
|---|---|---|---|
| | | Cosine Similarity (%) | Word2Vec (%) |
| 1 | Doc. 1 vs. Doc. KA | 50.4 | 99.65788 |
| 2 | Doc. 2 vs. Doc. KA | 43.9 | 99.76635 |
| 3 | Doc. 3 vs. Doc. KA | 60.4 | 99.6982 |
| 4 | Doc. 4 vs. Doc. KA | 25.7 | 97.30118 |
| 5 | Doc. 5 vs. Doc. KA | 36.1 | 99.88402 |
| 6 | Doc. 6 vs. Doc. KA | 58.8 | 99.65035 |
| 7 | Doc. 7 vs. Doc. KA | 29.7 | 92.05786 |
| 8 | Doc. 8 vs. Doc. KA | 40.1 | 94.42995 |
| 9 | Doc. 9 vs. Doc. KA | 33.6 | 99.25563 |
| 10 | Doc. 10 vs. Doc. KA | 48.5 | 99.89637 |
| 11 | Doc. 11 vs. Doc. KA | 37.1 | 95.08197 |
| 12 | Doc. 12 vs. Doc. KA | 33.6 | 94.9827 |
| 13 | Doc. 13 vs. Doc. KA | 36.9 | 97.7642 |
| 14 | Doc. 14 vs. Doc. KA | 57.1 | 97.3877 |
| 15 | Doc. 15 vs. Doc. KA | 32.1 | 99.6177 |
| 16 | Doc. 16 vs. Doc. KA | 33.3 | 99.8994 |
| 17 | Doc. 17 vs. Doc. KA | 28 | 97.8102 |
| 18 | Doc. 18 vs. Doc. KA | 44.5 | 94.3593 |
| 19 | Doc. 19 vs. Doc. KA | 40.2 | 92.3695 |

| No. | Document Name | Fulfillment result | |
|---|---|---|---|
| | | Cosine Similarity (%) | Word2Vec (%) |
| 1 | Doc. 1 vs. Doc. KA | 50.4 | 99.65788 |
| 2 | Doc. 2 vs. Doc. KA | 43.9 | 99.76635 |
| 3 | Doc. 3 vs. Doc. KA | 60.4 | 99.6982 |
| 4 | Doc. 4 vs. Doc. KA | 25.7 | 97.30118 |
| 5 | Doc. 5 vs. Doc. KA | 36.1 | 99.88402 |
| 6 | Doc. 6 vs. Doc. KA | 58.8 | 99.65035 |
| 20 | Doc. 20 vs. Doc. KA | 49.7 | 96.2454 |

In Table 4, the similarity score and fulfillment of the best performance knowledge areas with the cosine similarity method is produced in Doc. 3 with a similarity score of 60.4 and with the Word2Vec similarity method generated in Doc. 10 and Doc. 16 with a similarity score of 99.8994% which can be seen in Table 3 and Table 4. The comparison between two similarity checking methods results in a significant difference between the two methods. Table 3 can be seen that the results of comparing documents with the best similarity results in the cosine similarity method are obtained in the 4th document, and the best similarity results in the Word2Vec method are obtained in the 9th document. Of course, the results of the similarity greatly affect the content of the material processed by similarity whether the content is identical or not.

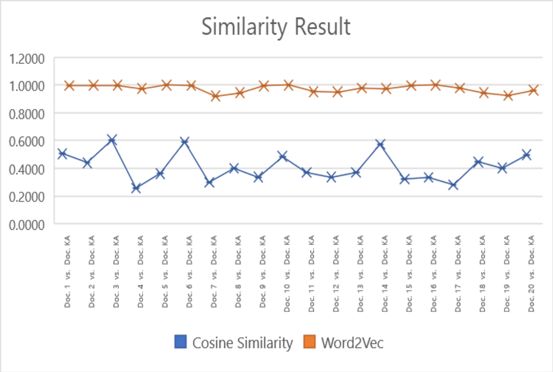

Figure 7. Similarity result

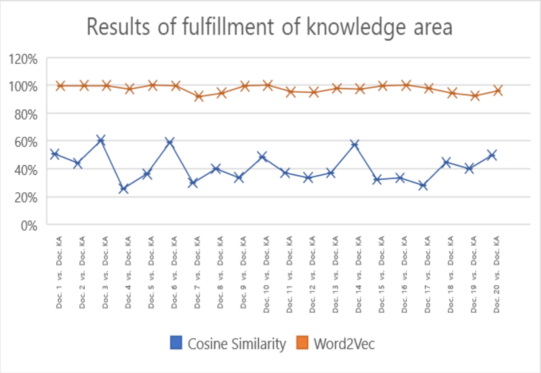

Figure 8. Fulfillment of knowledge area

In Figure 7 shown the results of the method process, meanwhile in Figure 8 shown the results of the fulfillment of computer sciences documents on knowledge area documents. The similarity results for the first document (a computer science document from SMAN 1 Malang), similarity generated using cosine similarity is 0.5040, and using Word2Vec is 0.9966. The resulting similarity tends to be low because one of the factors is that the material elements developed at this school are only 3 elements, namely computational thinking, computer networks, and general skills. For the second document (computer science document from SMAN 2 Malang), the similarity generated using cosine similarity was 0.4390, and using Word2Vec the similarity was 0.9977. The resulting similarity tends to be low because the material elements developed at this school are still 2 elements of computer science material from 18 knowledge areas. Meanwhile, for the 3rd document (computer science document from SMAN 3 Malang) the similarity results obtained using cosine similarity were 0.6040 and by using Word2Vec the similarity obtained was 0.9970. The resulting similarity tends to be better than Docs 1 and 2 because the material elements developed at this school have 5 material elements, including computer systems, information and communication technology, computer networks, and the internet, and data analysis. In the 4th document (computer science document from SMAN 4 Malang) the similarity results obtained using cosine similarity was 0.2570 and using Word2Vec the similarity obtained was 0.9730. The resulting similarity tends to be low because the material elements developed at this school are still 2 material elements, namely computational thinking and information and communication technology elements. In the 5th document (computer science document from SMAN 5 Malang), the similarity generated using cosine similarity was 0.3610 and by using Word2Vec the similarity was 0.9988. The similarity produced in this document tends to be low also because the material elements developed at this school are still 2 elements, namely computational thinking and information and communication technology elements. In the 6th document (computer science document from SMAN 6 Malang) the similarity results obtained using cosine similarity were 0.5880 and using Word2Vec the similarity was 0.9965. The similarity produced in this document tends to be good because the material elements developed at this school have 9 elements, namely computational thinking, information and communication technology elements, computer system elements, cross-field practice elements, informatics elements and general skills, algorithm and programming elements, elements social impact of informatics, data analysis and elements of computer networks and the internet. In contrast to the 7th document (computer science document from SMAN 7 Malang) the similarity results obtained by using cosine similarity were 0.2970 and by using Word2Vec the similarity obtained was 0.9206. The similarity produced in this document tends to be low because the material elements developed at this school are still 2 elements, namely computational thinking and data analysis elements.

In the 8th document (computer science document from SMAN 8 Malang) the similarity results obtained using cosine similarity were 0.4010 and by using Word2Vec the similarity obtained was 0.9443. The similarity produced in this document tends to be low because the material elements developed at this school are still 2 elements, namely computational thinking and data analysis elements. In the 9th document (computer science document from SMAN 9 Malang) the similarity results obtained using cosine similarity were 0.3360 and by using Word2Vec the similarity obtained was 0.9926. The similarity produced in this document tends to be low because the material elements developed at this school have 2 elements, namely computational thinking and information and communication technology elements. In the 10th document (computer science document from SMAN 10 Malang) the similarity results obtained using cosine similarity were 0.4850 and using Word2Vec the similarity obtained was 0.9990. The similarity produced in this document tends to be low because the material elements developed at this school are still 2 elements, namely computational thinking elements and information and communication technology elements. For the 11th document (computer science document from SMAN Taruna Nala, East Java, Malang) the similarity result obtained by using cosine similarity was 0.3710 and by using Word2Vec the similarity obtained was 0.9508. The similarity produced in this document tends to be low because the material elements developed at this school are still 8 elements, namely elements of computational thinking, elements of information and communication technology, elements of computer systems, elements of informatics and general skills, elements of algorithms and programming, elements of the social impact of informatics, analysis of data and elements of computer networks and the internet, but the content of the material developed does not fully cover the elements of the material developed. In the 12th document (computer science document from Cor Jesu Malang Christian High School), the similarity produced by using cosine similarity is 0.3360 and by using Word2Vec the similarity obtained is 0.9498. The similarity produced in this document tends to be low because the material elements developed at this school are still 8 elements, namely elements of computational thinking, elements of information and communication technology, elements of computer systems, elements of cross-field practice, elements of informatics and general skills, elements of algorithms and programming, elements of the social impact of informatics, data analysis and elements of computer networks and the internet, but the content of the material developed does not fully cover the elements of the material developed.

In the 13th document (computer science document from SMAN 2 Kalitidu) the similarity result obtained by using cosine similarity was 0.3690 and by using Word2Vec the similarity obtained was 0.9776. The similarity produced in this document tends to be low because the material elements developed at this school are still 1 element, namely the computer network and internet elements. Meanwhile, in the 14th document (computer science document from SMAN 2 Kisaran) the similarity result obtained by using cosine similarity was 0.5710 and by using Word2Vec the similarity obtained was 0.97387654. The similarity produced in this document tends to be low because the material elements developed at this school are still 3 elements, namely computer system elements, data analysis elements, and computer network and internet elements. In the 15th document (computer science document from SMAN 12 Bekasi) the similarity result obtained using cosine similarity was 0.321 and using Word2Vec the similarity obtained was 0.9961769. The similarity produced in this document tends to be low because the material elements developed at this school are still 1 element, namely the computer network and internet elements. For the 16th document (computer science document from SMAN 1 Batangan) the similarity result obtained using cosine similarity was 0.333 and using Word2Vec the similarity obtained was 0.99899405. The similarity produced in this document tends to be low because the material elements developed at this school are still 1 element, namely the information and communication technology element. For the 17th document (computer science document from SMAN 2 Batu Malang) the similarity result obtained using cosine similarity was 0.280 and using Word2Vec the similarity obtained was 0.9781024. The similarity produced in this document tends to be low because the material elements developed at this school are still 4 elements, namely computational thinking elements, information and communication technology elements, computer system elements, and computer network and internet elements. In the 18th document (computer science document from SMAN 2 Lamongan) the similarity result obtained by using cosine similarity was 0.445 and by using Word2Vec the similarity obtained was 0.943593. The similarity produced in this document tends to be low because the material elements developed at this school are still 4 elements, namely introductory informatics elements, information and communication technology elements, computer network and internet elements, data analysis elements, and cross-field practice elements. In the 19th document (computer science document from SMAN 2 Playen) the similarity result obtained by using cosine similarity was 0.402 and by using Word2Vec the similarity obtained was 0.92369515. The similarity produced in this document tends to be low because the material elements developed at this school are still 1 element, namely algorithm and programming elements. For the last document, the 20th document (computer science document from SMAN 114 North Jakarta), the similarity result obtained using cosine similarity was 0.497, and using Word2Vec, the similarity obtained was 0.96245396. The similarity produced in this document tends to be low because the material element developed at this school is still 1 element, namely the computer system element.

In Figure 8, the similarity results obtained have significant differences between documents 1 to 20. The difference in similarity is based on the content that has been developed by each teacher at a high school in Malang. Documents 3 and Documents 16 have a fairly good level of similarity from the results of the similarity process. It can be said that the material content contained in the document has fulfilled the competencies or main topics contained in the knowledge area document. Whereas in documents 1, 3, 6 and 14 the similarity results are quite good above 0.5. In this process it can be said that the document has quite good similarity results. This result can be said that the content of the processed documents has similarities that are less satisfactory than the similarity process that is carried out. In Figure 8 for the results of the fulfillment of computer sciences documents on knowledge area documents also in documents 1, 3, 6 and 14 have the highest percentage level in terms of fulfillment of the contents of the material. Whereas documents 2, 4, 5, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19 and 20 have a moderate percentage of fulfillment. In fulfilling the percentage of document fulfillment, it can be said to be good if it is close to 100%. From the results of the process using the cosine similarity and Word2Vec methods, the similarity indicator used is word-based similarity. For further research, deeper similarity can be carried out by using scenarios by carrying out similarity based on words, combinations of words, sentences and paragraphs. So, the results obtained will be more detailed and diverse.

## 4. CONCLUSION

Document similarity can be analyzed based on the document input. The document contains terms that are identical to the document knowledge area, so the similarity of the documents produced is also good. In this study, the approach to measuring document similarity using the cosine similarity method resulted in the fulfillment of the knowledge area sub-topic on an average of 40.9850%, while using the Word2Vec method resulted in the fulfillment of the knowledge area sub-topic on an average of 97.3558%. The best similarity performance can be demonstrated by the Word2Vec method. This research can be developed further by using other methods that have more similar performance with computer science document data input from all schools in Indonesia starting from phase C, D to E until phase F.

## ACKNOWLEDGEMENTS

The author would like to thank computer sciences teachers who are members of the Malang City High School computer sciences teacher community, especially High Schools in Malang City and curriculum developer that upload in PMM who have helped or provided support related to research conducted such as assistance in providing data and information regarding the development of the computer sciences curriculum in each school.

## REFERENCES

[1] Astawa, Ida Bagus Made dan Adnyana. 2018. Belajar dan Pembelajaran. Depok: Rajawali Grafindo Persada.

[2] "SNPT. (2020 The National Standard on Higher Education (Standar Nasional Pendidikan Tinggi) 3 (Indonesia Ministry of Education and Culture (Kementerian Pendidikan dan Kebudayaan)) Accessed 20 September 2023, https://peraturan.bpk.go.id/Home/Details/163703/ permendikbud-no-3-tahun-2020 BERITA NEGARA REPUBLIK INDONESIA No.47, 2020, 1-75."

[3] "Mubai, Akrimullah & Jalinus, Nizwardi & Ambiyar, Ambiyar & Wakhinuddin, Wakhinuddin & Abdullah, Rijal & Rizal, Fahmi & Waskito, Waskito. (2021). Implementasi Model Cipp Dalam Evaluasi Kurikulum Pendidikan Teknik Informatika. EDUKATIF : JURNAL ILMU PENDIDIKAN. 3. 1383-1394. 10.31004/edukatif.v3i4.549.".

[4] Cc2020 Task Force, Computing Curricula 2020: Paradigms for Global Computing Education. New York, NY, USA: ACM, 2020. doi: 10.1145/3467967.

[5] C. Crysdian, "The evaluation of higher education policy to drive university entrepreneurial activities in information technology learning," Cogent Education, vol. 9, no. 1, p. 2104012, Dec. 2022, doi: 10.1080/2331186X.2022.2104012.

[6] S. Anggara, "Exploring the Effectiveness of Merdeka Belajar Kampus Merdeka Policy in Indonesian Higher Education Institutions: An In-depth Case Study Analysis," AIJP, vol. 15, no. 2, pp. 1563–1570, May 2023, doi: 10.35445/alishlah.v15i2.3885.

[7] H. Mahliatussikah and S. Kuswardono, "Merdeka Belajar Kampus Merdeka (MBKM) Curriculum Design in Arabic Language Education Study Program," in Proceedings of the Unima International Conference on Social Sciences and Humanities (UNICSSH 2022), R. Harold Elby Sendouw, T. Pangalila, S. Pasandaran, and V. P. Rantung, Eds., in Advances in Social Science, Education and Humanities Research, vol. 698. Paris: Atlantis Press SARL, 2023, pp. 587–595. doi: 10.2991/978-2-494069-35-0_72.

[8] "CIPG and Nesta. (2019). Understanding Indonesia's inno_vation system. Downloaded from https://media. nesta.org.uk 20 September 2023."

[9] M. Pabbajah, I. Abdullah, R. N. Widyanti, H. Jubba, and N. Alim, "Student demoralization in education:The industrialization of university curriculum in 4.0.Era Indonesia," Cogent Education, vol. 7, no. 1, p. 1779506, Jan. 2020, doi: 10.1080/2331186X.2020.1779506.

[10] P. P. Tardan, A. Erwin, K. I. Eng, and W. Muliady, "Automatic text summarization based on semantic analysis approach for documents in Indonesian language," in 2013 International Conference on Information Technology and Electrical Engineering (ICITEE), Yogyakarta, Indonesia: IEEE, Oct. 2013, pp. 47–52. doi: 10.1109/ICITEED.2013.6676209.

[11] A. Fuddoly, J. Jaafar, and N. Zamin, "Keywords Similarity Based Topic Identification for Indonesian News Documents," in 2013 European Modelling Symposium, Manchester, United Kingdom: IEEE, Nov. 2013, pp. 14–20. doi: 10.1109/EMS.2013.3.

[12] S. Sohangir and D. Wang, "Improved sqrt-cosine similarity measurement," J Big Data, vol. 4, no. 1, p. 25, Dec. 2017, doi: 10.1186/s40537-017-0083-6.

[13] A. R. Lahitani, A. E. Permanasari, and N. A. Setiawan, "Cosine similarity to determine similarity measure: Study case in online essay assessment," in 2016 4th International Conference on Cyber and IT Service Management, Bandung, Indonesia: IEEE, Apr. 2016, pp. 1–6. doi: 10.1109/CITSM.2016.7577578.

[14] C. Slamet, A. R. Atmadja, D. S. Maylawati, R. S. Lestari, W. Darmalaksana, and M. A. Ramdhani, "Automated Text Summarization for Indonesian Article Using Vector Space Model," IOP Conf. Ser.: Mater. Sci. Eng., vol. 288, p. 012037, Jan. 2018, doi: 10.1088/1757-899X/288/1/012037.

[15] A. Amalia, D. Gunawan, Y. Fithri, and I. Aulia, "Automated Bahasa Indonesia essay evaluation with latent semantic analysis," J. Phys.: Conf. Ser., vol. 1235, no. 1, p. 012100, Jun. 2019, doi: 10.1088/1742-6596/1235/1/012100.

[16] P. P. Gokul, B. K. Akhil, and K. K. M. Shiva, "Sentence similarity detection in Malayalam language using cosine similarity," in 2017 2nd IEEE International Conference on Recent Trends in Electronics,

Information & Communication Technology (RTEICT), Bangalore: IEEE, May 2017, pp. 221–225. doi: 10.1109/RTEICT.2017.8256590.

[17] N. R. Ramadhanti and S. Mariyah, "Document Similarity Detection Using Indonesian Language Word2vec Model," in 2019 3rd International Conference on Informatics and Computational Sciences (ICICoS), Semarang, Indonesia: IEEE, Oct. 2019, pp. 1–6. doi: 10.1109/ICICoS48119.2019.8982432.

[18] Z. Jingling, Z. Huiyun, and C. Baojiang, "Sentence Similarity Based on Semantic Vector Model," in 2014 Ninth International Conference on P2P, Parallel, Grid, Cloud and Internet Computing, Guangdong, China: IEEE, Nov. 2014, pp. 499–503. doi: 10.1109/3PGCIC.2014.101.

[19] D. A. Diartono, I. Nugroho, and J. A. Razaq, "PENINGKATAN HASIL SISTEM TEMU KEMBALI INFORMASI BERBASIS PADA KATA MAJEMUK MENGGUNAKAN JACCARD SIMILARITY," JDI, vol. 14, no. 1, pp. 1–10, Mar. 2022, doi: 10.35315/informatika.v14i1.9160.