

Implementasi Metode Similaritas Pada Software Requirements Specification (SRS) Pengembangan Startup HayLingo Berdasarkan *Use Case Diagram* Menggunakan Text Mining

Ardiansyah Maulana Putra¹, Yudi Priyadi², Rosa Reska Riskiana³

^{1,2,3} Universitas Telkom, Bandung

¹ardiansyahmp@student.telkomuniversity.ac.id, ²whyphi@telkomuniversity.ac.id,

³rosareskaa@telkomuniversity.ac.id

Abstrak

Penelitian ini menggunakan Software Requirement Specification (SRS) untuk mengukur kesamaan kebutuhan perangkat lunak dari Pengembangan Startup HayLingo. Kesamaan diukur berdasarkan User Requirement. Secara umum penelitian ini bertujuan untuk mengetahui kesesuaian User Requirement dengan perangkat lunak yang akan dibangun melalui kesamaan antara Functional Requirements dan Use Case Diagram (Description) pada dokumen SRS Pengembangan Startup HayLingo. Kegiatan similaritas ini dilakukan melalui pendekatan Text Mining berupa kegiatan Text Pre-Processing pada dokumen kebutuhan SRS, yang kemudian dibandingkan. Tujuan khusus dari penelitian ini memiliki kegiatan sebagai berikut: mengklarifikasi antara model perangkat lunak, menentukan kesamaan, memvalidasi dan reliabilitas menggunakan Gwet AC1. Hasil pengukuran kemiripan mendapatkan nilai tertinggi sebesar 0.581422. Indeks kappa yang diperoleh adalah 0.26239, dimana proporsi kesepakatan yang diperoleh adalah Cukup (fair agreement).

Kata Kunci: Similarity, Use Case Description, Functional Requirement, Text Mining, Software Requirement Specification, Gwet's AC1

Abstract

This study uses the Software Requirement Specification (SRS) to measure the similarity of the software requirements of HayLingo Startup Development. Similarity is measured based on User Requirements. In general, this study aims to determine the suitability of User Requirements with the software to be built through the similarities between Functional Requirements and Use Case Diagrams (Description) in the HayLingo Startup Development SRS document. This similarity activity is carried out through a Text Mining approach in the form of Text Pre-Processing activities on SRS requirements documents, which are then compared. The specific purpose of this research has the following activities: clarifying between software models, determining similarities, validating and reliability using Gwet AC1. The results of the similarity measurement get the highest value of 0.581422. The kappa index obtained is 0.26239, where the proportion of agreements obtained is Fair (fair agreement).

Keywords: Use Case Description, Functional Requirement, Text Mining, Software Requirement Specification, Gwet's

1. Pendahuluan

1.1 Latar Belakang

Dalam aktivitas pengembangan suatu perangkat lunak, seperti: mengidentifikasi, mengumpulkan dan menyajikan, merupakan suatu hal yang rumit untuk dilakukan. Kerumitan ini terjadi, karena selalu terdapat kesalahan dalam menginterpretasikan requirement. Jika hal tersebut terjadi, maka dapat menurunkan kualitas dokumentasi *Software Requirements Specification* (SRS). [1].

Use Case Diagram membentuk dasar *Requirement* yang dibuat pengembang dalam kaitannya dengan *Requirement* fungsional dari sistem yang akan dikembangkan [2].

Pada penelitian ini digunakan suatu pendekatan berbasis *Requirement Statement* berupa *Functional Requirements* dan *Use Case Diagram (Description)* untuk mengukur kemiripan kebutuhan perangkat lunak pada Pengembangan Startup HayLingo. Penggunaan *Functional Requirements* dan *Use Case Diagram (Description)* pada penelitian ini digunakan secara dominan sebagai model dasar Artefak.

Secara Umum penelitian ini bertujuan untuk menentukan kesesuaian *User Requirement* dengan perangkat lunak yang akan dibangun, melalui *Similarity* antara *Functional Requirements* dan *Use Case Diagram (Description)* dengan pemodelan perangkat lunaknya. Untuk mendapatkan *Similarity* ini, dilakukan melalui pendekatan *Text Mining*, berupa kegiatan *Text Pre-Processing*, yang kemudian dibandingkan dengan artefak Unified Modeling Language (UML) pada *Functional Requirements* dan *Use Case Diagram (Description)*. Untuk mencapai hal-hal yang ada pada tujuan umum tersebut, maka tugas akhir ini Memiliki Tujuan Khusus sebagai berikut: Membandingkan *Requirement Statement* pada dokumen SRS Pengembangan Startup HayLingo berupa *Functional Requirements* dan *Use Case Description*, melakukan klarifikasi antara model perangkat lunak *Functional Requirements* dengan *Use Case Diagram (Description)*, menentukan *Similarity*, menggunakan *Text Mining*, melakukan validasi dan reliabilitas dari hasil *Similarity* menggunakan *Gwet AC1*.

1.2 Topik dan Batasan

Berdasarkan latar belakang di atas, maka permasalahan yang dapat dirumuskan sebagai berikut:

1. Menggunakan *Functional Requirements* dan *Use Case Diagram (Description)* pada SRS Pengembangan Startup HayLingo,
2. Menggunakan struktur bahasa Inggris dalam menganalisis *text*.
3. Dalam menerjemahkan kedalam Bahasa Inggris menggunakan tools google translate.
4. Setelah menganalisis *text* selanjutnya menghitung *Cosine Similarity* untuk mendapatkan hasil similaritas dari sebuah *text*, setelah mendapatkan hasil similaritas dari *text*
5. selanjutnya menghitung *Gwet's AC1* untuk mendapatkan hasil validasi.

1.3 Tujuan

Tugas akhir ini Secara Umum bertujuan untuk menentukan kesesuaian *User Requirement* dengan perangkat lunak yang akan dibangun, melalui *Similarity* antara *Functional Requirements* dengan *Use Case Diagram (Description)*. Untuk mendapatkan *Similarity* ini, dilakukan melalui pendekatan *Text Mining*, berupa kegiatan *Text Pre-Processing* berdasarkan hasil identifikasi dokumen SRS. Untuk mencapai hal-hal yang ada pada tujuan umum tersebut, maka tugas akhir ini memiliki tujuan khusus sebagai berikut:

1. Melakukan identifikasi dokumen SRS, bernama Pengembangan Startup HayLingo pada artefak *Functional Requirements* dan *Use Case Diagram (Description)*.
2. Menentukan *Similarity User Requirement* antara *Functional Requirements* dan *Use Case Diagram (Description)* menggunakan *Text Mining*.
3. Melakukan validasi dan reliabilitas dari hasil *Similarity* menggunakan *Gwet AC1*.

1.4 Organisasi Tulisan

Pada proses penelitian ini, diawali dengan pendahuluan yang menjelaskan latar belakang, batasan, dan tujuan. Berikutnya, dilanjutkan dengan rangkuman studi literatur mengenai konsep dan teori yang mendukung. Pada tulisan ini, bagian inti penelitian terdapat pada system yang dibangun dan pembahasan. Bagian terakhir berupa kesimpulan dan saran untuk pekerjaan selanjutnya.

2 Studi Terkait

2.1 Similarity

Semantics dan *Similarity* saling berkaitan dalam pembahasan *Natural Language Processing* (NLP). NLP membahas tentang cara mengukur kemiripan dua kata atau lebih yang direpresentasikan oleh persamaan konsep yang terkait. Ide awal dari *Semantics Similarity* ialah untuk mengidentifikasi konsep yang memiliki karakteristik sama atau *Similar*. Secara teknis, mengukur nilai kesamaan dapat menggunakan rumus dan kode program [3].

Requirement elicitation mencakup bagian penting dari rekayasa requirement, yaitu langkah-langkah dalam pengumpulan requirement, sebagai berikut: *elicitation requirement*, *analisis requirement*, *dokumentasi requirement*, *validasi requirement* dan *manajemen requirement* [4, 5].

Kesamaan semantik adalah ukuran spesifik dari kemiripan. Keterhubungan adalah ukuran yang lebih umum, yang juga mencerminkan keterkaitan. Kesamaan dan keterkaitan semantik dapat diterapkan untuk menyelesaikan statement yang tidak sama, seperti keambiguan dari arti kata, klasifikasi teks, mencari informasi, penerjemahan mesin, dan pengelompokan dokumen [6, 7].

2.2 Artefak SRS (Use case diagram, Use Case Scenario, Fungsional requirement)

Requirement Statement merupakan tahap dalam proses pengembangan perangkat lunak, untuk memastikan agar *requirement statement* dengan *Requirement* spesifikasi telah terpenuhi, karena lebih dari tiga perempat kegagalan sistem berasal dari proses analisis *requirement* [8].

Use case diagram digunakan untuk menggambarkan interaksi antara pengguna sistem (aktor) dengan use case yang berdasarkan pada skenario tertentu. Use case hanya menggambarkan alur aktor ketika berinteraksi dengan sistem. Oleh karena itu, use case dapat dihubungkan dengan use case lain melalui beberapa hubungan interdependensi: include, extend, alternative atau specialization, dan exception [9].

Use Case Scenario (Description) merupakan deskripsi singkat dari use case yang menjelaskan *requirement*, dijabarkan dalam bahasa alami [2]. *Fungsional requirement* digunakan untuk menentukan respons sistem yang diharapkan dapat diterima oleh aktor saat memberikan input tertentu, jika kondisi tertentu terpenuhi [6].

2.3 Text Mining

Text mining adalah proses mencari atau mengekstraksi informasi yang berguna dari data tekstual. Penambangan teks mencoba untuk menemukan pola yang menarik dari database besar [10, 11]. Secara umum, terdapat empat proses yang perlu dijalankan dalam melakukan *text mining* diantaranya : Pengumpulan data, *Text Pre-Processing*, merancang pemodelan yang akan melakukan proses *looping* melalui proses evaluasi serta validasi, dilakukannya presentasi dan interaksi untuk memverifikasi hasil pemodelan. Proses teks mining dimulai melalui kumpulan dokumen, alat penambangan teks akan mengambil dokumen tertentu dan memprosesnya terlebih dahulu dengan memeriksa format penulisan dan set karakter. Kemudian akan melalui fase analisis teks, terkadang perlu beberapa kali mengulang teknik analisis sampai informasi terekstraksi [12].

2.4 Cosine Similarity

Cosine Similarity adalah ukuran untuk menghitung pasangan kalimat yang memiliki keterkaitan satu sama lain, dan menentukan skor berdasarkan kata-kata yang mirip dalam kalimat [13]. *Cosine Similarity* juga dapat ditentukan oleh sudut antara dua vektor. Ini memungkinkan dokumen dengan komposisi yang sama diproses secara identik yang menjadikannya ukuran paling populer untuk dokumen teks. Untuk menghitung kemiripan suatu objek, salah satu perhitungannya bisa menggunakan *Cosine Similarity* [7, 12]. Melalui proses perhitungan tersebut, maka dapat dicari kesamaan antar dokumen. Berikut ini adalah perhitungan skor kesamaan *Cosine*:

$$\text{sim}(d1, d2) = \frac{d1 \cdot d2}{\|d1\| \cdot \|d2\|} = \frac{d1 \cdot d2}{\sqrt{\sum_{i=1}^t x_i^2} \cdot \sqrt{\sum_{i=1}^t y_i^2}} \quad (1)$$

Keterangan :

d1 = dokumen sumber

d2 = dokumen pembanding

x_i = vector term (kata) ke-i pada dokumen sumber

y_i = vector term (kata) ke-i pada dokumen pembanding

Untuk menghitung persamaan *Cosine* antara dua kalimat, maka kedua kalimat tersebut kemudian diubah menjadi sebuah istilah/kata. Kata tersebut kemudian ditransformasikan dalam bentuk vektor, dimana setiap kata dalam teks mendefinisikan dimensi dalam ruang Euclidean dan frekuensi dari setiap kata, sesuai dengan nilai dalam dimensi. Dikatakan identik, ketika hasil perhitungan kemiripannya mendekati satu (1) [12].

2.5 Gwet's AC1

Gwet's AC1 merupakan sebuah perhitungan statistika untuk mengukur hasil dari index kesepakatan dua ahli analisis atau pengamat dalam menguji sebuah model rekomendasi. Untuk menghitung nilai *Gwet's AC1* dari hasil pertanyaan pakar, dapat dicermati pada Tabel I [14, 15, 16].

Pengamat 1	Pengamat 2		
	Ya	Tidak	TOTAL
Ya	A	B	B1 = A+B
Tidak	C	D	B2 = C+D
TOTAL	A1 = A+C	A2 = B+D	N

Tabel 1 Hasil Elisitasi Oleh Dua Pengamat

Terdapat perhitungan oleh dua pengamat, dimana pengamat 1 mengklasifikasi A1, dan A2 itu menunjukkan jumlah *subject*. Pengamat 2 mengklasifikasi B1, dan B2 itu menunjukkan jumlah *subject*. Berikut ini adalah langkah untuk menghitung *Gwet's AC1*, yaitu:

- Langkah pertama perhitungan AC1 yang terobservasi:

$$P = \frac{A+D}{N} \quad (2)$$

- Langkah kedua Menghitung Probabilitas chance-agreement:

$$e(\gamma) = 2P_1(1 - P_1) \quad (3)$$

Dimana rumus P_1 sebagai berikut:

$$P_1 = \frac{(A1+B1)/2}{N} \quad (4)$$

- Langkah ketiga menghitung statistika AC1:

$$AC1 = \frac{p-e(\gamma)}{1-e(\gamma)} \quad (5)$$

Index kappa hasil perhitungan statistika *Gwet's AC1* berdasarkan dua pengamat beserta proporsi kesepakatannya dapat dilihat pada Table II.

Index Kappa	Proporsi Kesepakatan
< 0	Rendah (<i>less than chance-agreement</i>)
0.01 – 0.02	Sedikit (<i>slight agreement</i>)
0.21 – 0.40	Cukup (<i>fair agreement</i>)
0.41 – 0.60	Sedang (<i>moderate agreement</i>)
0.61 – 0.80	Substansi (<i>substantial agreement</i>)
0.81 - 1	Hampir Sempurna (<i>almost perfect</i>)

Tabel 2 Index Proporsi Kesepakatan *Gwet's AC1*

2.6 Studi Literatur

Pada Tabel 1, ditampilkan review dari aktivitas penelitian yang berhubungan dengan pembahasan SRS, *Requirement Statement*, Text Mining, Similarity, *Gwet AC1*.

Penulis Dan Judul	Metode	Hasil Penelitian
Kartika, M. D et al	• <i>Gwet's AC1</i> • SDLC • SWOT	Hasilnya berupa mockup tampilan halaman web TB.Purnama. dengan memperhitungkan nilai <i>Gwet's AC1</i> dengan Menguji validasi dan reliability menggunakan dan mendapatkan kesimpulan dari dua pakar. Nilai statistika AC1 mendapatkan nilai “Substansi”

Fauzan, R., & Pramono, D. et al (2013)	• Pemproses-an Bahasa Alami • <i>Cohen's Kappa</i>	• Mengkstraksi diagram kasus penggunaan • Pemilihan kalimat yang mirip dengan kasus penggunaan • Mengekstraksi kalimat Mendapatkan nilai kemiripan
Daniela, U et al (2018)	Kualitatif	Menghasilkan website untuk Sidone GKJ dari hasil perancangan SRS.
Priyadi, Y et al (2020)	• Text Mining • idVar4CL	Similaritas yang di dapat dari perancangan program python 3 menggunakan text pre-processing akan mendapatkan nilai cosine similarity berupa matrix.
S, Rosa. (2019)	Kualitatif	Menyusun dokumen SRS tentang Manajemen sistem informasi perpustakaan hal yang di susun meliputi : Use Case, Diagram Kelas dll
Purbo, Onno. (2019)	Text Mining	Memvisualkan graph pada Bahasa pemrograman python dari data yang telah di dapat dari Twitter dengan web browser selenium.
D. Siahaan et al (2017)	• Klasifikasi • Pemprosesan Bahasa alami • Gwet's AC1 • Adaboost • C4.5	Menganalisis dengan mencocokkan rule model yang di dihasilkan pada saat latih. Diambil beberapa <i>rule</i> yang mempengaruhi prediksi. Dari <i>rule</i> tersebut didapat fitur yang paling dominan yaitu kata sifat, kata keterangan, kata ganti, jmlah kata penghubung, dan
Y. Priyadi et al (2019)	• Text Processing • Semantik Similarity • Gwet AC1 • WS4J	Graph ketergantungan kebutuhan pada SRS, terdapat hubungan antara hasil proses <i>Requirements Dependency Graph</i> dari SRS dengan jenis ketergantungan Dahlstedt yaitu pada contoh similar, Part-Of, dan elaborate.
Dehak, N et al (2010)	Semantik Similarity	Mendapatkan nilai Similarity pada suara dari beberapa percobaan.
Setiadi, W. (2018)	• Gwet's AC1 • <i>n-gram</i>	• Pengumpulan Dataset • Pemprosesan awal dataset • Repositori Aturan rekomendasi • Uji coba Metode Rekomendasi • Analisis Dan Evaluasi Metode Rekomendasi Menggunakan Gwet's AC1

Artina, N. (2006)	Structural	Aliran data Use Case lebih terstruktur
Mujilahwati, S. (2016)	Pre-Processing	Hasil Text Pre-Processing yang telah di klasifikasikan
Karnik, A et al (2007)	<i>Cosine Similarity</i>	Sebuah matrix berupa Nilai • Avg-Cosin Similarity • Source Program Windows • Target Program
Kurniawan, R. G et al (2020)	Distributi-nal Semantik Similarity	• Nilai Similarity per kata Bahasa Inggris dan Indonesia • Visualisasi <i>vector</i> dengan inputan Bahasa Inggris dan Indonesia
Rundupandang, R et al (2018)	SWTO	Rancangan UML berdasarkan Business Model Canvas
Komputer, F. I. et al (2017)	• <i>Cosine Similarity</i> • <i>Text Mining</i>	Sebuah aplikasi pendeteksi perbandingan kemiripan metode Cosin Similarity dan Smith-Waterman
Permana, F. B et al (2016)	• <i>Cosine Similarity</i> • Struktur	Grafik nilai kappa untuk setiap nilai threshold • nilai threshold • “Withdraw Cash” dan “Program Switch • Kueri deskripsi
Dreisbach, C. et al (2019)	• NLP • text mining • symptoms	Nilai studi proporsi dari nilai hasil dari text mining
Martiana, E et al (2011)	• Machine learning • Text Mining • Naïve Bayes	• Aplikasi Desktop client chatting • Nilai Probabilitas
Harfianto, H et al (2018)	Algoritma FP-Growth	nilai akurasi Kappa
Horkoff, J. (2019)	• Machine Learning • FR and NFR	Algoritma ML untuk menentukan akurasi dan performenc dari NFR
Samosir, H et al (2019)	• Semantic Similarity • Gwet's AC1	Hasil Pendekatan yang andalan dalam Mengidentifikasi Kebutuhan Kelas RealisasiRequire menggunakan Kappa AC1

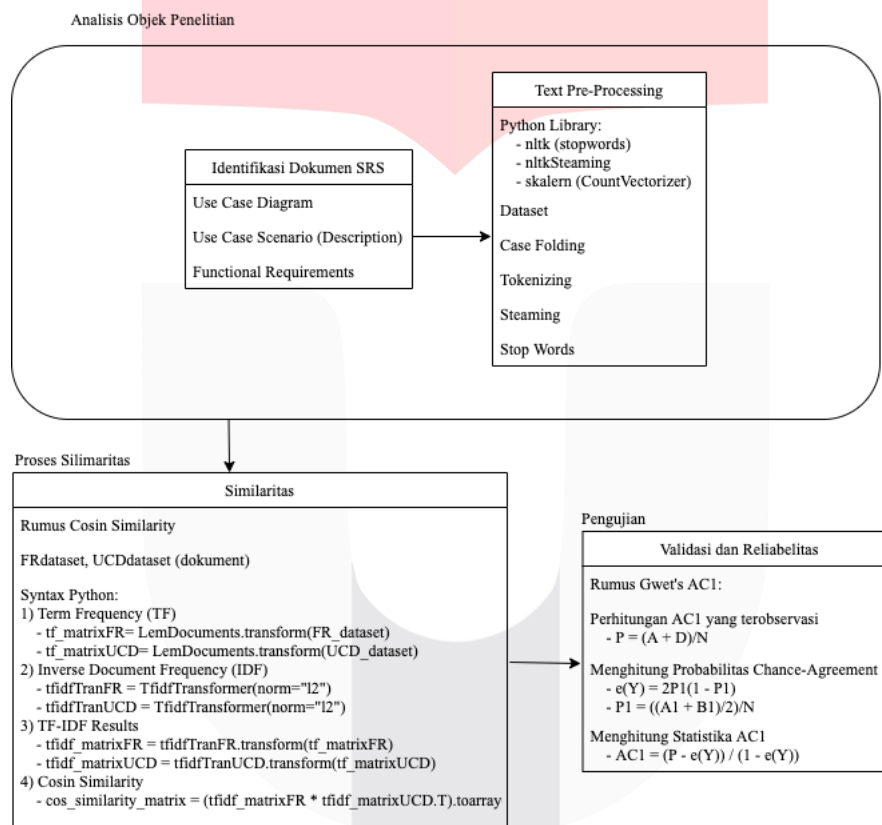
Tabel 3 Studi Literatur

3 Sistem Yang Dibangun

Merujuk pada Gambar 1, disajikan mengenai perancangan *Research Framework*. Secara garis besar, dalam perancangan ini terdiri dari tiga kegiatan utama, yaitu:

1) Analisis Objek Penelitian, dibagi menjadi tiga yaitu:

- a) Pemodelan Artefak. Pemodelan yang dibuat mencakup pembuatan *Use Case Diagram*, *Use Case Description* Dan *Activity Diagram*.
 - b) *Text Pre-Processing*. Dilakukan untuk menerapkan metode *Text Mining*, yang akan digunakan untuk menentukan similaritas dari sebuah *text*.
- 2) Proses *Similarity*
- Pada tahap proses *Similarity*, digunakan rumus *Cosine Similarity* untuk mendapatkan nilai *Similarity* dari dataset yang telah ditentukan dalam proses *Text Pre-Processing*. Perhitungan *Cosine Similarity* menggunakan library Python, sebagai berikut:
- a) Syntax untuk menentukan Tokenizing, Steaming, Stop Words.
 - b) Sintaks untuk menentukan *Term Frequency* (TF).
 - c) Syntax untuk menentukan *Inverse Document Frequency* (IDF).
 - d) Syntax untuk menentukan TF-IDF Results.
 - e) Syntax menentukan *Cosin Similarity*.
- 3) Pengujian
- Tahap pengujian, merupakan tahap untuk memvalidasi dari hasil *Similarity* yang sebelumnya telah dilakukan. Proses validasinya menggunakan Rumus Gwet's AC1 yang dibagi menjadi tiga tahap perhitungan, yaitu: perhitungan AC1 yang terobservasi, menghitung Probabilitas chance-agreement, dan menghitung statistika AC1.



Gambar 1 Research Framework

4 Pembahasan

Berikut ini adalah penjelasan dari rincian aktivitas, berdasarkan research framework yang telah dibuat sebelumnya, terdiri dari:

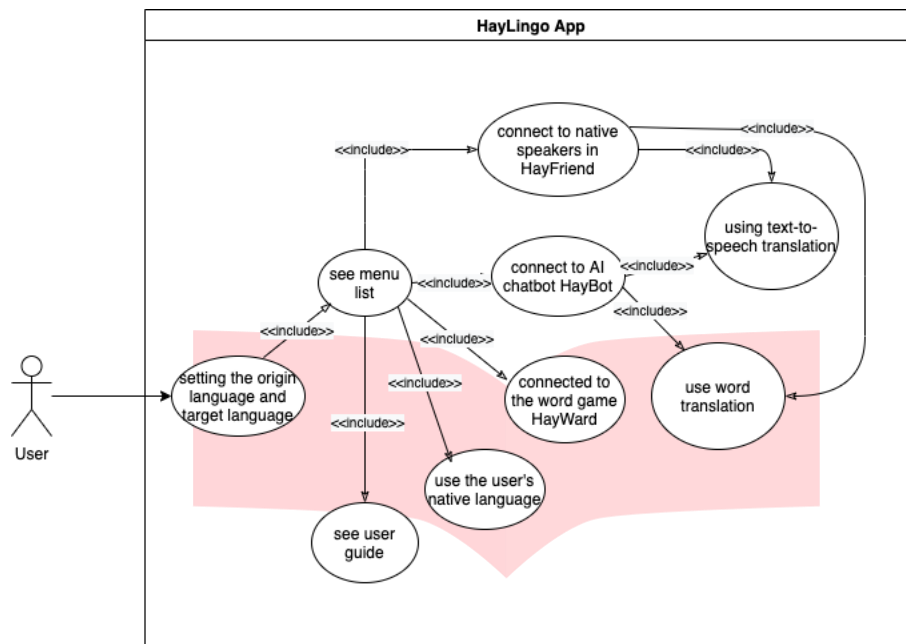
1. Studi Literatur. Konsep penulisan ini ditunjang dengan penelitian sebelumnya, melalui suatu pendekatan referensi, yang mencakup *Use Case Diagram*, *Use Case Scenario (Description)* dan *Functional Requirement*.
2. *Text Pre-Processing*, Dilakukan untuk menerapkan metode *Text Mining*, yang digunakan untuk menentukan similaritas dari sebuah *text*. *Pre-Processing* sangat penting dilakukan untuk menganalisis *text* yang tidak sesuai, seperti kata yang tidak formal, ambigu, dan tidak terstruktur.
3. Kesesuaian artefak pada dokumen SRS. Jika artefak tidak sesuai, maka proses akan terus berulang sampai menemukan kesesuaian. Selanjutnya dilakukan *Text Pre-Processing* dan melakukan similaritas.

4. *Similarity*. Untuk mendapatkan hasil similaritas, maka pada proses ini menggunakan rumus *Cosine Similarity* dari proses yang telah dilakukan sebelumnya, pada *text* yang sudah diproses melalui metode *Text Pre-Processing*.
5. Pengujian. Pengujian dilakukan dengan menggunakan rumus Gwet's AC1 untuk menvalidasi data yang sudah didapat.



4.1 Requirement Statement

Requirement yang diidentifikasi dari dokumen SRS Pengembangan Startup HayLingo berupa *Functional Requirements* dan *Use Case (Description)* yang nantinya akan diproses untuk menentukan similaritasnya menggunakan python. lihat gambar 2 untuk use case diagram dan tabel 4 *Use Case Scenario (Description)* dan tabel 5 *Functional Requirement*.



Gambar 2 Use Case Diagram

No	Nama Fungsi	Deskripsi
1	Language Settings	The actor user sets the language when first logging into the HayLingo chatbot. The system stores the origin language of the actor user and adjusts the language in the application such as the actor user's home language and sets the artificial intelligence language of the HayBot feature and other target users according to the target language being studied.
2	View Menu List	Actor users who have entered the main page of the HayLingo application will be given an option to be able to view other main features such as HayBot, HayFriend, and HayWord as well.
3	See User Help & Commands	The actor user has seen the menu list display and the menu list displays several feature menus that can be selected and wants to see a help list that can help if the user needs certain assistance in the chatbot.
4	Changing Language	Actor users have seen the menu list display and the list menu displays several feature menus that can be selected, including a menu to change the language in the settings where there is a button to change the language.
5	Unlock Premium View	Actor users have seen the menu list display and the list menu displays several feature menus that can be selected, including the settings menu card with buttons to view premium profiles and status.
6	Connect To AI Bot On HayBot	The actor user has seen the menu list display and the menu list displays several feature menus that can be selected, including the main menu which contains the HayBot feature. This feature has a talkative chatbot function to be able to talk or converse with the user. The actor user enters the HayBot menu and starts chatting with the HayBot chatbot.
7	Connect to other users in HayFriend	Actor users have seen the menu list display and the list menu displays several feature menus that can be selected, including the main menu which includes the HayFriend feature. This feature has a function to bring together actor users with other users who are studying the same language to be able to talk and/or learn to better understand certain languages according to the target language they want to learn (currently only English).

8	Connect To HayWord Game	The actor user has seen the menu list display and the list menu displays several feature menus that can be selected, including the main menu which contains the HayWord feature. This feature has a function as a feature that can help increase vocabulary and find some unique words that were previously unknown to the user through a 'guess word' game scheme. The user actor when entering this feature is faced with a word which the user has to guess whether the word is included in a certain word setting (such as verb, noun, adjective).
---	-------------------------	--

Tabel 4 Use Case Scenario (Description)

No	Fungsional Requirement
1	As a user, you can chat with HayBot's artificial intelligence chatbot
2	As a user, you can connect and chat with other users from the target foreign language you want to learn
3	As a user, you can play a guessing mini game on the HayWord feature
4	As a user, you can check, activate or deactivate premium feature functions
5	As a user, you can use the phrase ideas and topic ideas in HayBot and HayFriend
6	As a user, you can activate bot audio replies and get a voice response from the bot
7	As a user, you can translate various words with HayLingo's word-by-word translation feature
8	As a user, you can do text-to-speech conversions while talking

Tabel 5 Fungsional Requirement

4.2 Text Pre-Processing

Inisialisasi awal merupakan hasil untuk pertanyaan *Requirement* yang telah ditentukan sebelumnya, terdapat dua belas dokumen terdiri dari enam dokumen dari *Functional Requirements (Description)* dan enam dokumen dari *Use Case Scenario (Description)*. Pada Tabel III akan digunakan dalam proses Text Mining. Mengacu pada keduabelas Dokumen dataset tersebut selanjutnya akan dilakukan *Text Pre-Processing* dengan menggunakan tool *Natural Language Toolkit* (NLTK). NLTK merupakan proses yang digunakan untuk memproses Bahasa alami, tahapan *Text Processing* yang dilakukan pada penelitian ini meliputi : *Case Folding* (untuk menghapus angka, tanda baca, menghapus karakter kosong dan mengubah *text* menjadi *lowercase*), *Tokenizing* (untuk memisahkan setiap dokumen menjadi potongan-potongan token per-kata maupun kalimat), *Stemming* proses infleksi ke bentuk kata dasar, *Stopwords Removal* (untuk menghilangkan beberapa kata yang tidak memiliki arti dalam setiap dokumen).

Pada Gambar 3 dan Gambar 4 merupakan cuplikan layar proses *Text Processing* menggunakan *Case Folding*, semua Dokumen diubah menjadi huruf kecil *Lowercase*. Kedua Dokumen dijadikan file .txt berbeda, bernama "FR_beforecase.txt" untuk *Functional Requirements* dan "UCD_beforecase.txt" untuk *Use Case Scenario (Description)* dan hasil dari *Case Folding* disimpan dalam file .txt bernama "FR_aftercase.txt" dan "UCD_aftercase.txt"

```
[ "as a user, you can chat with haybot's artificial intelligence chatbot. as a user, you can connect and chat with other users from the target foreign language you want to learn. as a user, you can play this game, guess the word on the hayword feature. as a user, you can check, activate or deactivate premium feature functions. as a user, you can use the phrase ideas and topic ideas in haybot and hayfriend. as a user, you can activate bot audio replies and get a voice response from the bot. as a user, you can translate various words with haylingo's word-by-word translation feature. as a user, you can do text-to-speech conversions while talking." ]
```

Gambar 3 Hasil Pengolahan Case Folding Functional Requirements

["actor user sets the language when first logging into the haylingo chatbot. the system stores the origin language of the actor user and adjusts the language in the application such as the actor user's home language and sets the artificial intelligence language of the haybot feature and other target users according to the target language being studied. actor users who have entered the main page of the haylingo application will be given an option to be able to view other main features such as haybot, hayfriend, and hayword as well. actor user has seen the menu list display and the menu list displays several feature menus that can be selected and wants to see a help list that can help if the user needs certain assistance in the chatbot. actor users have seen the menu list display and the list menu displays several feature menus that can be selected, including a menu to change the language in the settings where there is a button to change the language. actor users have seen the menu list display and the list menu displays several feature menus that can be selected, including the settings menu card with buttons to view premium profiles and status. the actor user has seen the menu list display and the menu list displays several feature menus that can be selected, including the main menu which contains the haybot feature. this feature has a talkative chatbot function to be able to talk or talk to the user. the actor user enters the haybot menu and starts chatting with the haybot chatbot. actor users have seen the menu list display and the list menu displays several feature menus that can be selected, including the main menu which includes the hayfriend feature. this feature has a function to bring together actor users with other users who are studying the same language to be able to talk and/or learn to better understand certain languages according to the target language they want to learn (currently only english). the actor user has seen the menu list display and the list menu displays several feature menus that can be selected, including the main menu which contains the hayword feature. this feature has a function as a feature that can help increase vocabulary and find some unique words that were previously unknown to the user through a 'guess word' game scheme. the user actor when entering this feature is faced with a word which the user has to guess whether the word is included in a certain word setting (such as verb, noun, adjective)."]

Gambar 4 Hasil Pengolahan Case Folding Use Case Scenario (Description)

Pada Gambar 5 terdapat enambelas Dokumen, yang didapatkan dari *Functional Requirements* dan *Use Case Scenario (Description)*, Dokumen tersebut nantinya akan dijadikan dataset, terdapat d1, d2, d3, d4, d5, d6, d7 dan d8 untuk dataset dari *Functional Requirements* dan d9, d10, d11, d12, d13, d14, d15, dan d16 untuk dataset dari *Use Case Scenario (Description)*.

```
#FR And UCD Documentation
d1 = "as a user, you can chat with haybot's artificial intelligence chatbot."
d2 = "as a user, you can connect and chat with other users from the target foreign language you want to learn."
d3 = "as a user, you can play this game, guess the word on the hayword feature."
d4 = "as a user, you can check, activate or deactivate premium feature functions."
d5 = "as a user, you can use the phrase ideas and topic ideas in haybot and hayfriend."
d6 = "as a user, you can activate bot audio replies and get a voice response from the bot."
d7 = "as a user, you can translate various words with haylingo's word-by-word translation feature."
d8 = "as a user, you can do text-to-speech conversions while talking."
d9 = "Actor user sets the language when first logging into the HayLingo chatbot. The system stores the origin language
d10 = "Actor users who have entered the main page of the HayLingo application will be given an option to be able to view
d11 = "Actor user has seen the menu list display and the menu list displays several feature menus that can be selected
d12 = "Actor users have seen the menu list display and the list menu displays several feature menus that can be selected
d13 = "Actor users have seen the menu list display and the list menu displays several feature menus that can be selected
d14 = "Actor user has seen the menu list display and the menu list displays several feature menus that can be selected
d15 = "Actor users have seen the menu list display and the list menu displays several feature menus that can be selected
d16 = "User aktor telah melihat tampilan list menu dan list menu menampilkan beberapa menu fitur yang dapat dipilih, terdapat
FR_dataset = [d1,d2,d3,d4,d5,d6,d7,d8]
UCD_dataset = [d9,d10,d11,d12,d13,d14,d15,d16]
```

Gambar 5 Dokumentasi

Setelah dokumen disiapkan, maka selanjutnya dokumen tersebut diproses untuk Tokenisasi, proses ini dilakukan untuk memecah kalimat per kalimat disertai dengan proses Stemming sehingga semua hasil menjadi kata-kata penting, proses dapat dilihat pada Gambar 6 dan Gambar 7.

```
{'user': 38, 'chat': 7, 'haybot': 18, 'artificial': 4, 'intelligence': 23, 'chatbot': 8, 'connect': 10, 'users': 39, 'target': 32, 'foreign': 14, 'language': 24, 'want': 42, 'learn': 25, 'play': 27, 'game': 16, 'guess': 17, 'word': 43, 'hayword': 21, 'feature': 13, 'check': 9, 'activate': 3, 'deactivate': 12, 'premium': 28, 'functions': 15, 'use': 37, 'phrase': 26, 'ideas': 22, 'topic': 34, 'hayfriend': 19, 'bot': 6, 'audio': 5, 'replies': 29, 'voice': 41, 'response': 30, 'translate': 35, 'various': 40, 'words': 45, 'haylingo': 20, 'word-by-word': 44, 'translation': 36, 'text-to-speech': 33, 'conversions': 11, 'talking': 31}
```

Gambar 6 Hasil Pengolahan Tokenization dan Stemming Functional Requirements

```
'actor': 7, 'user': 115, 'ses': 94, 'language': 61, 'logging': 65, 'haylingo': 50, 'chatbot': 24, '': 4, 'stores': 99, 'origin': 82, 'adjusts': 9, 'application': 13, 's": 0, 'home': 53, 'artificial': 14, 'intelligence': 57, 'haybo': 48, 'feature': 41, 'target': 105, 'users': 116, 'according': 6, 'studied': 100, 'entered': 39, 'main': 66, 'pag': 83, 'given': 46, 'option': 81, 'able': 5, 'view': 118, 'features': 42, '': 3, 'hayfriend': 49, 'hayword': 51, 's een': 91, 'menu': 75, 'list': 64, 'display': 35, 'displays': 36, 'menus': 76, 'selected': 92, 'wants': 121, 'help': 5 2, 'needs': 78, 'certain': 22, 'assistance': 15, 'including': 55, 'change': 23, 'settings': 95, 'button': 19, 'card': 21, 'buttons': 20, 'premium': 86, 'profiles': 87, 'status': 98, 'contains': 26, 'talkative': 103, 'function': 44, 'ta lk': 102, 'enters': 40, 'starts': 97, 'chatting': 25, 'includes': 54, 'bring': 18, 'studying': 101, 'and/or': 11, 'le arn': 63, 'better': 17, 'understand': 113, 'languages': 62, 'want': 120, '(': 1, 'currently': 27, 'english': 38, '):' 2, 'aktor': 10, 'telah': 107, 'melihat': 69, 'tampilan': 104, 'dan': 28, 'menampilkan': 73, 'beberapa': 16, 'futura': 43, 'yang': 122, 'dapat': 29, 'dipilih': 34, 'termasuk': 109, 'didalamnya': 31, 'terdapat': 108, 'ini': 56, 'memilik i': 71, 'fungsi': 45, 'sebagai': 88, 'membantu': 70, 'menambah': 72, 'vocabulary': 119, 'menemukan': 74, 'kata': 58, 'unik': 114, 'sebelumnya': 89, 'tidak': 112, 'diketahui': 33, 'oleh': 80, 'melalui': 68, 'sebuah': 90, 'skema': 96, 'permainan': 85, '': 123, 'tebak': 106, '': 124, 'ketika': 60, 'masuk': 67, 'kedalam': 59, 'dihadapkan': 32, 'denga n': 30, 'nanti': 77, 'harus': 47, 'ditebak': 37, 'apakah': 12, 'tersebut': 110, 'pengaturan': 84, 'tertentu': 111, 's eperti': 93, 'verb': 117, 'noun': 79, 'adjective': 8}
```

Gambar 7 Hasil Pengolahan Tokenization dan Steaming Use Case Scenario (Description)

4.3 TF-IDF Processing Documents Value

Proses perhitungan matrix TF-IDF menggunakan library CountVectorizer untuk mendapatkan matrix dengan hasil 0 dan 1 yang dimana library tersebut merupakan hasil dari pemrosesan yang hasilnya disimpan pada variable LemDocuments dapat dilihat pada Gambar 8. setelah mendapatkan hasil matrix TF-IDF selanjutnya menghitung similarity antar dokumen, dan jika nilai dari hasil pemrosesan similarity mendekati 1 dapat disimpulkan kedua dokumen tersebut memiliki kesamaan yang identik.

Sintaks dan hasil eksekusi untuk menentukan *Term Frequency* (TF) dapat dicermati pada Gambar 8 dan Gambar 9, yaitu:

- Dataset dari *Functional Requirements (Description)* ditandai dengan “FR” `tf_matrixFR=LemDocuments.transform(FR_dataset)`.
- Dataset dari *Use Case Scenario (Description)* ditandai dengan “UCD” `tf_matrixUCD=LemDocuments.transform(UCD_dataset)`.

[illegible]

Gambar 8 Hasil Pengolahan Term Frequency (TF) Functional Requirements

[illegible]

Gambar 9 Hasil Pengolahan Term Frequency (TF) Use Case Scenario (Description)

Syntax dan hasil eksekusi untuk menentukan *Inverse Document Frequency* (IDF) dapat dicermati pada Gambar 10 dan Gambar 11, yaitu:

- tfidfTranFR = TfidfTransformer(norm="l2")
- tfidfTranUCD=TfidfTransformer(norm="l2")

2.09861229	3.19722458	3.19722458	1.	1.	3.19722458
3.19722458	3.19722458	3.19722458	3.19722458	3.19722458	3.19722458
3.19722458	3.19722458	3.19722458	3.19722458	3.19722458	3.19722458
3.19722458	3.19722458	3.19722458	3.19722458	3.19722458	3.19722458
2.5040774	3.19722458	3.19722458	3.19722458	3.19722458	3.19722458
3.19722458	3.19722458	3.19722458	3.19722458	3.19722458	3.19722458
3.19722458	3.19722458	3.19722458	3.19722458	3.19722458	3.19722458
3.19722458	3.19722458	2.5040774	3.19722458	3.19722458	3.19722458
2.09861229	2.5040774	2.5040774	2.5040774	3.19722458	3.19722458
3.19722458	3.19722458	3.19722458	3.19722458	3.19722458	3.19722458
3.19722458	3.19722458	3.19722458	2.5040774	3.19722458	3.19722458
3.19722458	3.19722458	3.19722458	3.19722458	3.19722458	3.19722458
3.19722458	3.19722458	3.19722458	3.19722458	3.19722458	3.19722458
3.19722458	3.19722458	3.19722458	3.19722458	3.19722458	3.19722458
3.19722458	3.19722458	2.5040774	3.19722458	3.19722458	3.19722458
3.19722458	3.19722458	3.19722458	3.19722458	3.19722458	3.19722458
3.19722458	3.19722458	3.19722458	3.19722458	3.19722458	3.19722458
2.5040774	3.19722458	3.19722458	2.5040774	3.19722458	3.19722458
3.19722458	3.19722458	3.19722458	3.19722458	3.19722458	3.19722458
3.19722458	1.	3.19722458	3.19722458	3.19722458	3.19722458
2.5040774	3.19722458	3.19722458	3.19722458	3.19722458	3.19722458

Gambar 10 Hasil Pengolahan Inverse Document Frequency (IDF) Functional Requirements

[	2.5040774	2.09861229	2.09861229	1.25131443	1.	3.19722458
	3.19722458	1.11778304	3.19722458	3.19722458	2.5040774	2.5040774
	2.5040774	3.19722458	3.19722458	3.19722458	2.5040774	2.5040774
	2.5040774	2.09861229	3.19722458	2.5040774	2.09861229	3.19722458
	1.81093022	3.19722458	3.19722458	3.19722458	2.5040774	2.5040774
	2.5040774	2.5040774	2.5040774	2.5040774	2.5040774	1.40546511
	3.19722458	2.5040774	2.5040774	3.19722458	3.19722458	3.19722458
	3.19722458	2.5040774	2.09861229	2.5040774	2.5040774	3.19722458
	1.81093022	2.09861229	2.09861229	2.09861229	2.5040774	2.5040774
	3.19722458	3.19722458	2.5040774	3.19722458	2.5040774	2.5040774
	2.5040774	3.19722458	3.19722458	2.5040774	1.25131443	3.19722458
	1.58778666	2.5040774	2.5040774	2.5040774	2.5040774	2.5040774
	2.5040774	2.5040774	2.5040774	1.25131443	3.19722458	2.5040774
	3.19722458	2.5040774	2.5040774	2.5040774	2.5040774	2.5040774
	2.5040774	2.5040774	2.5040774	3.19722458	2.5040774	2.5040774
	2.5040774	1.40546511	3.19722458	2.5040774	3.19722458	3.19722458
	2.5040774	3.19722458	3.19722458	3.19722458	3.19722458	3.19722458
	2.09861229	3.19722458	2.5040774	2.09861229	2.5040774	2.5040774
	2.5040774	2.5040774	2.5040774	2.5040774	2.5040774	2.5040774
	2.5040774	1.	3.19722458	2.5040774	2.09861229	3.19722458
	2.09861229	3.19722458	2.5040774	2.5040774	2.5040774	]

Gambar 11 Hasil Pengolahan Inverse Document Frequency (IDF) Use Case Scenario (Description)

Syntax dan hasil eksekusi untuk menentukan TF-IDF Results, yaitu:

- `tfidf_matrixFR=tfidfTranFR.transform(tf_matrixFR).`
- `tfidf_matrixUCD=tfidfTranUCD.transform(tf_matrixUCD).`

Transformasinya menggunakan *Library tfidfTran*, yang dapat dicermati pada Gambar 12 dan Gambar 13.

[illegible]

Gambar 12 Hasil Pengolahan TF-IDF Results Functional Requirements

```
[[0.28238208 0. 0. 0. 0.22553782 0.
0. 0.37815353 0. 0. 0. 0.
0. 0. 0. 0. 0. 0.
0. 0. 0. 0. 0. 0.
0.20421663 0. 0. 0. 0. 0.
0. 0. 0. 0. 0. 0.
0. 0. 0. 0. 0. 0.
0. 0. 0. 0. 0. 0.
0.20421663 0. 0.23665822 0. 0. 0.28238208
0. 0. 0. 0. 0. 0.
0. 0. 0. 0. 0. 0.
0. 0. 0. 0. 0. 0.
0. 0. 0. 0. 0. 0.
0. 0. 0. 0. 0.28238208 0.
0. 0. 0. 0. 0. 0.
0. 0. 0. 0. 0. 0.
0. 0. 0. 0. 0. 0.
0. 0. 0. 0.47331645 0. 0.
0. 0. 0. 0. 0. 0.]
```

Gambar 13 Hasil Pengolahan TF-IDF Results Use Case Scenario (Description)

4.4 Similarity Value

Syntax menentukan *Cosin Similarity*, yaitu: $\text{cos_similarity_matrix} = (\text{tfidf_matrixFR} * \text{tfidf_matrixUCD.T}).\text{toarray}()$. Sintaks ini untuk menentukan hasil perhitungan dari proses yang dilakukan sebelumnya, dari hasil perhitungan *Cosine Similarity* tersebut telah menghasilkan sebuah array, yang dapat dicermati pada Gambar 14.

Dari hasil pengujian, nilai yang didapat dari hasil pemrosesan antara kedua dokumen memiliki similarity tertinggi adalah dokumen ke-5 yaitu "as a user, you can use the phrase ideas and topic ideas in haybot and hayfriend." dan dokumen ke-7 yaitu "as a user, you can activate bot audio replies and get a voice response from the bot." Dengan dokumen ke-13 yaitu "Actor users have seen the menu list display and the list menu displays several feature menus that can be selected, including the settings menu card with buttons to view premium profiles and status." dan dokumen ke-15 yaitu "Actor users have seen the menu list display and the list menu displays several feature menus that can be selected, including the main menu which includes the HayFriend feature. This feature has a function to bring together actor users with other users who are studying the same language to be able to talk and/or learn to better understand certain languages according to the target language they want to learn (currently only English).", dengan hasil dari pengukuran similaritas mendapatkan nilai tertinggi sebesar 0.581422 pada matrix ke-[5,7], artinya tingkat similarity yang dihasilkan tertinggi atau yang hampir mendekati 1.



	0	1	2	3	4	5	6
0	0.519571	0.464339	0.193157	0.157136	0.298867	0.390643	0.503482
1	0.253578	0.143451	0.454496	0.211260	0.476103	0.337569	0.342648
2	0.196317	0.228082	0.094878	0.077185	0.089875	0.191883	0.089875
3	0.107698	0.120208	0.181038	0.147277	0.123833	0.264382	0.123833
4	0.093172	0.103994	0.156620	0.304848	0.107130	0.228722	0.107130
5	0.501466	0.158291	0.186691	0.245744	0.387957	0.322000	0.150820
6	0.118114	0.520981	0.172067	0.238024	0.249974	0.289950	0.135808
7	0.078456	0.091881	0.175011	0.104516	0.090210	0.192597	0.090210

	7
0	0.222224
1	0.192032
2	0.109156
3	0.150398
4	0.130112
5	0.581422
6	0.303601
7	0.109562

Gambar 14 Hasil Pengolahan Cosin Similarity value

4.5 Validasi Gwet's AC1

Langkah validasi menggunakan rumus Gwet's AC1. Pada python menggunakan library "cohen_kappa_score" untuk menghitung index kappa Gwet's AC1 pada hasil pengukuran similaritas dengan nilai tertinggi yaitu pada matrix ke-[5,4]. Index kappa yang diperoleh ialah 0.26239 yang artinya masuk ke dalam range 0.21 – 0.40 dimana proporsi kesepakatan yang didapat Cukup (*fair agreement*).

5 Kesimpulan

Terdapat kesamaan antara dokumen *Fungsional Requirement* dan *Use Case Scenario (Description)*. Hasil dari pengukuran similaritas mendapatkan nilai tertinggi sebesar 0.581422 pada dokumen ke-5, dokumen ke-7, dokumen ke-13, dan dokumen ke-15. Untuk Index kappa yang diperoleh ialah 0.26239 dikatakan Valid karena masuk ke dalam range 0.21 – 0.40, dapat disimpulkan kesamaan antara kedua dokumen yang dibandingkan memiliki proporsi kesepakatan Cukup (*fair agreement*).

Pada penelitian selanjutnya, untuk validasi dan realibilitas disarankan melibatkan dua orang pengamat dalam kegiatan validasi hasil similaritas menggunakan konsep Gwet's AC1, yang menghitung dua dokumen kebutuhan agar dapat sesuai dengan yang diharapkan. Untuk proses validasi dengan pyhon menggunakan "cohen_kappa_score".

Referensi

- [1] Setiadi, W. (2018). *Rekomendasi Perbaikan Pernyataan Kebutuhan Yang Rancu Dalam Spesifikasi Kebutuhan Perangkat Lunak Menggunakan Teknik Berbasis Aturan Depandi. 2*, 227–249.
- [2] Karnik, A., Goswami, S., & Guha, R. (2007). Detecting Obfuscated Viruses Using Cosine Similarity Analysis. *Proceedings - 1st Asia International Conference on Modelling and Simulation: Asia Modelling Symposium 2007, AMS 2007*, 165–170. <https://doi.org/10.1109/AMS.2007.31>
- [3] Kurniawan, R. G., & Bijaksana, M. A. (2020). Building related words in Indonesian and English translation of Al-Qur'an vocabulary based on distributional similarity. *Jurnal Teknologi Informasi Dan Terapan*, 7(1), 46–53.
- [4] Dar, H., Lali, M. I., Ashraf, H., Ramzan, M., Amjad, T., & Shahzad, B. (2018). A systematic study on software requirements elicitation techniques and its challenges in mobile application development. *IEEE Access*, 6, 63859–63867. <https://doi.org/10.1109/ACCESS.2018.2874981>
- [5] C. Dreisbach, T. A. Koleček, P. E. Bourne, and S. Bakken, “A systematic review of natural language processing and text mining of symptoms from electronic patient-authored text data,” *Int. J. Med. Inform.*, vol. 125, no. February, pp. 37–46, 2019, doi: 10.1016/j.ijmedinf.2019.02.008.
- [6] Cai, Y., Zhang, Q., Lu, W., & Che, X. (2018). A hybrid approach for measuring semantic similarity based on IC-weighted path distance in WordNet. *Journal of Intelligent Information Systems*, 51(1), 23–47. <https://doi.org/10.1007/s10844-017-0479-y>
- [7] Almousa, M., Benlamri, R., Khoury, R., Almousa, M., Benlamri, R., & Khoury, R. (n.d.). *Knowledge-Based Systems xxx (xxxx) xxx Exploiting non-taxonomic relations for measuring semantic similarity and relatedness in WordNet*. <https://doi.org/10.1016/j.knosys.2020.106565>
- [8] Gelu, P., Sarno, R., & Siahaan, D. (2018). Requirements Association Extraction based on Use Cases Diagram. *Lontar Komputer: Jurnal Ilmiah Teknologi Informasi*, 9(1), 11. <https://doi.org/10.24843/lkjiti.2018.v09.i01.p02>.
- [9] Sabharwal, S., Sibai, R., & Kaur, P. (2015). Deriving Complexity Metric based on Use Case Diagram and its validation. *2014 IEEE International Symposium on Signal Processing and Information Technology, ISSPIT 2014*, 102–107. <https://doi.org/10.1109/ISSPIT.2014.7300571>
- [10] Nayak, A. S., & Kanive, A. P. (2016). Survey on Pre-Processing Techniques for Text Mining. *International Journal Of Engineering And Computer Science*, June 2016. <https://doi.org/10.18535/ijecs/v5i6.25>
- [11] Kannan, S., Gurusamy, V., Vijayarani, S., Ilamathi, J. & Nithya, M. (2016). *Preprocessing Techniques for Text Mining Preprocessing Techniques for Text Mining*. 5(October 2014), 7–16.
- [12] Priyadi, Y., Kusumahadi, K., & Lyanda, P. S. (2021). *Causal Loop Variable Identification Method (IdVar4CL) For Systems Thinking Based On Text Mining Approach*. 19(2), 1–2.
- [13] Zec, S., Soriani, N., Comoretto, R., & Baldi, I. (2017). High Agreement and High Prevalence: The Paradox of Cohen's Kappa. *The Open Nursing Journal*, 11(1), 211–218. <https://doi.org/10.2174/1874434601711010211>
- [14] Y. Priyadi, A. Djunaidy, and D. Siahaan, “Requirements Dependency Graph Modeling on Software Requirements Specification Using Text Analysis,” *2019 1st Int. Conf. Cybern. Intell. Syst. ICORIS 2019*, pp. 221–226, 2019, doi: 10.1109/ICORIS.2019.8874920.
- [15] Dehak, N., Dehak, R., Glass, J., Reynolds, D., & Kenny, P. (2010). Cosine similarity scoring without score normalization techniques. *Odyssey 2010: Speaker and Language Recognition Workshop*, 71–75.
- [16] Davis, A., Dieste, O., Hickey, A., Juristo, N., & Moreno, A. M. (2006). Effectiveness of requirements elicitation techniques: Empirical results derived from a systematic review. *Proceedings of the IEEE International Conference on Requirements Engineering*, 179–188. <https://doi.org/10.1109/RE.2006.17>
- [17] Setiadi, W. (2018). *Rekomendasi Perbaikan Pernyataan Kebutuhan Yang Rancu Dalam Spesifikasi Kebutuhan Perangkat Lunak Menggunakan Teknik Berbasis Aturan Depandi. 2*, 227–249.
- [18] Mujilawati, S. (2016). Pre-Processing Text Mining Pada Data Twitter. *Seminar Nasional Teknologi Informasi Dan Komunikasi, 2016(Sentika)*, 2089–9815.
- [19] Fauzan, R., & Pramono, D. (2013). Pemeriksaan Kemiripan Diagram Kasus Penggunaan Terhadap Skenario. *JUTI: Jurnal Ilmiah Teknologi Informasi*, 11(2), 51. <https://doi.org/10.12962/j24068535.v11i2.a11>
- [20] Permana, F. B., & Siahaan, D. O. (2016). Pendekatan Kesamaan Semantik dan Struktur dalam Kasus Penggunaan untuk Mendapatkan Kembali Spesifikasi Kebutuhan Perangkat Lunak. *Journal of Information Systems Engineering and Business Intelligence*, 2(2), 57. <https://doi.org/10.20473/jisebi.2.2.57-66>
- [21] H. Samosir and D. Siahaan, “Identifying Requirements Association Based on Class Diagram Using Semantic Similarity,” vol. 10, no. 1, pp. 19–28, 2019.
- [22] S, Rosa. 2016. *Rekayasa Perangkat Lunak Terstruktur Dan Berorientasi Objek*. Bandung: Informatika Bandung.

- [23]Horkoff, J. (2019). Non-functional requirements for machine learning: Challenges and new directions. *Proceedings of the IEEE International Conference on Requirements Engineering, 2019-September*, 386–391. <https://doi.org/10.1109/RE.2019.00050>
- [24]Artina, N. (2006). Penerapan Analisis Kebutuhan Metode Use Case pada Metode Pengembangan Terstruktur. *Jurnal Ilmiah STIMIK GI MDP*, 2(3), 1–6. https://www.academia.edu/10468545/Jurnal_Penerapan_Analisis_Kebutuhan
- [25]Kartika, M. D., & Priyadi, Y. (2020). Pengembangan Sistem Penjualan Menggunakan UML dan Proses Bisnis E-Commerce Pada TB.Purnama Banjarnegara. *JATISI (Jurnal Teknik Informatika Dan Sistem Informasi)*, 7(3), 480–497. <https://doi.org/10.35957/jatisi.v7i3.416>
- [26]C. Dreisbach, T. A. Koleck, P. E. Bourne, and S. Bakken, “A systematic review of natural language processing and text mining of symptoms from electronic patient-authored text data,” *Int. J. Med. Inform.*, vol. 125, no. February, pp. 37–46, 2019, doi: 10.1016/j.ijmedinf.2019.02.008.

