



TESIS - IF185401

**PEMBOBOTAN KATA MENGGUNAKAN METODE *HIRST & S-ONGE* UNTUK
KLASIFIKASI KEBUTUHAN NON-FUNGSIONAL BERBASIS ISO/IEC 25010**

**LUKMAN HAKIM
NRP. 05111750010004**

DOSEN PEMBIMBING :

Dr. Ir. Siti Rochimah, M.T.

NIP. 19681002 199403 2 001

Dr. Eng. Chastine Fatichah, S.Kom, M.Kom.

NIP. 19751220 200112 2 002

PROGRAM MAGISTER

BIDANG KEAHLIAN REKAYASA PERANGKAT LUNAK

DEPARTEMEN TEKNIK INFORMATIKA

FAKULTAS TEKNOLOGI INFORMASI DAN KOMUNIKASI

INSTITUT TEKNOLOGI SEPULUH NOPEMBER

SURABAYA

2019

Tesis disusun untuk memenuhi salah satu syarat memperoleh gelar
Magister Komputer (M.Kom.)
di
Institut Teknologi Sepuluh Nopember Surabaya

oleh:
LUKMAN HAKIM
Nrp. 05111750010004

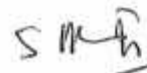
Dengan judul :

Pembobotan Kata Menggunakan Metode *Hirst & S-Onge* Untuk
Klasifikasi Kebutuhan Non-Fungsional Berbasis ISO/IEC 25010.

Tanggal Ujian : 10- 01- 2019
Periode Wisuda : 2019 Gasal

Disetujui oleh:

Dr. Ir. Siti Rochimah, MT
NIP. 196810021994032001



(Pembimbing 1)

Dr. Eng. Chastine Fatichah, S.Kom., M.Kom.
NIP. 197512202001122002



(Pembimbing 2)

Daniel Oranova Siahaan, S.Kom., M.Sc., PD.Eng.
NIP. 197411232006041001



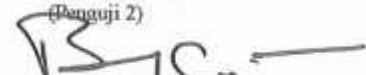
(Penguji 1)

Dr. Eng. Radityo Anggoro, S.Kom., M.Sc.
NIP. 1984101620081210002



(Penguji 2)

Bagus Jati Santoso, S. Kom., Ph. D.
NIP. 198611252018031001



(Penguji 3)



Dekan Fakultas Teknologi Informasi dan Komunikasi,


Dr. Agus Zainal Arifin, S.Kom, M.Kom
NIP. 19720809 199512 1 001

[Halaman ini sengaja dikosongkan]

PEMBOBOTAN KATA MENGGUNAKAN METODE *HIRST & S-ONGE* UNTUK KLASIFIKASI KEBUTUHAN NON-FUNGSIONAL BERBASIS ISO/IEC 25010

Nama : Lukman Hakim
NRP : 05111750010004
Pembimbing : Dr. Ir. Siti Rochimah, M.T.
Dr. Eng. Chastine Fatichah, S.Kom, M.Kom.

ABSTRAK

Aspek kualitas kebutuhan non-fungsional merupakan salah satu faktor penting yang berperan dalam kesuksesan pengembangan perangkat lunak. Aspek kualitas kebutuhan non-fungsional yang telah diketahui sejak tahap awal analisa kebutuhan diharapkan dapat meminimalisir hal-hal yang tidak diinginkan ketika produk perangkat lunak sudah dibuat. Namun, mengidentifikasi aspek kualitas kebutuhan non-fungsional merupakan hal yang sulit untuk dilakukan. Karena aspek kualitas kebutuhan non-fungsional sering ditemukan tidak lengkap, tersembunyi, dan tercampur di dalam kalimat-kalimat kebutuhan lain pada dokumen. Oleh karena itu dibutuhkan suatu cara untuk dapat mengidentifikasi aspek kualitas kebutuhan non-fungsional.

Penelitian yang ada telah mampu mengidentifikasi kebutuhan non-fungsional dengan menggunakan metode klasifikasi multi-label. Salah satunya adalah metode FSKKN dengan penambahan variabel semantik yang disebut semantik-FSKNN. Penambahan variabel tersebut membuat data yang dibentuk untuk pola pelatihan tersaring lebih ketat sehingga membuat kinerja klasifikasi lebih kecil dibandingkan metode FSKNN. Disamping itu standar kualitas yang digunakan tidak cocok untuk mengukur kualitas perangkat lunak. Oleh karena itu, penelitian ini mengusulkan pengukuran semantik antar *term-term* relevan setiap aspek kualitas dengan *term-term* pada dokumen yang akan dilabeli berbasis ISO/IEC 25010:2011. Proses pengukuran semantik dilakukan dengan menggunakan metode HSO. Metode HSO dipilih karena mampu mengukur keterkaitan semantik antara dua kata yang berbeda jenis katanya seperti *noun* dan *verb*. Sedangkan ISO/IEC 25010 dipilih sebagai acuan karena merupakan peningkatan dari ISO/IEC 9126.

Dari pengujian yang telah dilakukan dengan menggunakan dataset *voip* dan *geolocation* diketahui bahwa klasifikasi menggunakan metode FSKNN dengan pengembangan *term* sebanyak 1 level hipernim dan 20 sinonim dengan mempertimbangkan keterkaitan semantik dapat meningkatkan performa klasifikasi. Hal ini ditunjukkan dengan nilai *accuracy*, *precision*, *recall* dan F1 terbaik sebesar 97.10%, 25.05%, 21.35%, dan 14.73%.

Kata kunci: non-fungsional, iso/iec 25010, hipernim, hso, klasifikasi, fskn.

[Halaman ini sengaja dikosongkan]

TERM WEIGHTING USING *HIRST & S-ONGE* METHOD FOR CLASSIFICATION OF NON-FUNCTIONAL REQUIREMENT BASED ISO/IEC 25010

Student's Name : Lukman Hakim
Student ID : 05111750010004
Supervisor : Dr. Ir. Siti Rochimah, M.T.
Dr. Eng. Chastine Fatichah, S.Kom, M.Kom.

ABSTRACT

The quality aspect of non-functional requirements is one of the important factors that play a role in the success of software development. The quality aspects of non-functional requirements that have been known from the initial stage of requirements analysis are expected to minimize unwanted things when software products have been created. However, identifying aspects of quality of non-functional requirements is a difficult thing to do. Because the quality aspect of non-functional requirements is often found incomplete, hidden, and mixed in the sentences of other requirements in the document. Therefore we need a way to be able to identify aspects of the quality of non-functional requirements.

Existing research has been able to identify non-functional requirements using the multi-label classification method. One of them is the FSKKN method with the addition of semantic variables called semantics-FSKNN. The addition of these variables makes the data formed for the training pattern filtered tighter so as to make classification performance smaller than the FSKNN method. Besides that, the quality standards used are not suitable for measuring software quality. Therefore, this study proposes semantic measurement between relevant terms of each quality aspect with the terms of the document to be labeled based on ISO / IEC 25010:2011. The semantic measurement process is carried out using the HSO method. The HSO method was chosen because it was able to measure semantic linkages between two words of different types of words such as noun and verb. Whereas ISO / IEC 25010 was chosen as a reference because it is an increase from ISO / IEC 9126.

From the testing that has been performed using VoIP datasets and geolocation, it is known that the classification using the FSKNN method with the term development is 1 hypernym level and 20 synonyms by considering the semantic linkages can improve classification performance. This is indicated by the best *accuracy*, *precision*, *recall* and F1 values of 97.10%, 25.05%, 21.35%, and 14.73%.

Keywords: non-functional, iso/iec 25010, hypernym, hso, classification, fsknn.

[Halaman ini sengaja dikosongkan]

KATA PENGANTAR

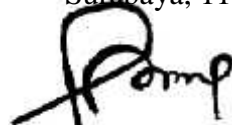
Segala puji bagi Allah SWT, Tuhan Yang Maha Pengasih dan Maha Penyayang yang telah melimpahkan segala rahmat, hidayah dan inayah-Nya sehingga tesis ini dapat diselesaikan dengan lancar dan baik. Pada kesempatan ini penulis ingin mengucapkan terima kasih sebanyak-banyaknya kepada :

1. Allah SWT atas limpahan nikmat iman, islam, kesehatan, waktu, serta berbagai kemudahan dari arah yang tidak pernah diduga sebelumnya, sehingga penulis dapat menyelesaikan penelitian ini dengan baik.
2. Ayahanda, Ibunda, Kakek, Nenek, Adik, Istri, dan Anak tercinta serta Bapak dan Ibu mertua yang tidak pernah berhenti memberikan dukungan materil, do'a yang tulus, serta percaya bahwa penulis harus senantiasa menjadi insan terbaik yang bermanfaat untuk orang sekitar dimanapun penulis berada.
3. Ibu Dr. Ir. Siti Rochimah, MT., dan Ibu Dr. Eng. Chastine Fatichah, S.Kom., M.Kom. selaku pembimbing yang telah banyak memberikan ilmu, arahan, saran dan motivasi selama pengerjaan tesis ini.
4. Bapak Daniel Oranova Siahaan, S.Kom., M.Sc., PD.Eng., Bapak Dr. Eng. Radityo Anggoro, S.Kom., M.Sc., dan Bapak Bagus Jati Santoso, S. Kom., Ph. D. selaku dosen penguji yang telah memberikan saran, arahan serta kritik yang membangun dalam tesis ini.
5. Bapak Waskitho Wibisono, S.Kom., M.Eng. Ph.D. selaku dosen wali dan Kaprodi S2 Teknik Informatika ITS Surabaya yang telah memberikan dorongan dan motivasi dalam kegiatan akademik.
6. Segenap dosen, staf TU dan karyawan perpustakaan Jurusan Teknik Informatika, Institut Teknologi Sepuluh Nopember.
7. Rekan-rekan S2 Teknik Informatika atas bantuan dan diskusi selama penelitian.
8. Semua pihak yang tidak dapat dituliskan satu per satu oleh penulis, terima kasih banyak atas doa dan dukungannya.

Semoga Allah SWT senantiasa menyayangi, menguatkan, memampukan, dan menunjukkan jalan yang terbaik atas semua kebaikan yang telah diberikan.

Penulis menyadari bahwa laporan penelitian ini tentunya masih jauh dari kesempurnaan. Oleh sebab itu, saran dan kritik sangat diharapkan untuk perbaikan dimasa yang akan datang. Semoga laporan penelitian ini dapat bermanfaat bagi penulis dan pembaca pada umumnya.

Surabaya, 11 Januari 2019

A handwritten signature in black ink, appearing to read 'Lukman Hakim', with a stylized, cursive script.

Lukman Hakim

DAFTAR ISI

LEMBAR PENGESAHAN	Error! Bookmark not defined.
ABSTRAK.....	iii
ABSTRACT.....	v
KATA PENGANTAR	vii
DAFTAR ISI.....	ix
DAFTAR GAMBAR	xiii
DAFTAR TABEL.....	xv
BAB 1 PENDAHULUAN	17
1.1. Latar Belakang	17
1.2. Perumusan Masalah.....	22
1.3. Tujuan Penelitian.....	23
1.4. Manfaat Penelitian.....	23
1.5. Kontribusi Penelitian.....	23
1.6. Batasan Masalah.....	24
BAB 2 KAJIAN PUSTAKA	25
2.1. Kebutuhan Non-Fungsional	25
2.2. ISO/IEC 25010:2011	26
2.3. Klasifikasi Teks.....	30
2.4. Klasifikasi <i>Multi-Label</i>	30
2.5. Algoritma Otomatisasi Data Pelatihan	31
2.5.1. Praproses	31
2.5.2. Pengembangan Term Gabungan antara Hipernim dan Sinonim.....	31
2.5.3. Relasi Hipernim	34
2.5.4. Pengukuran Keterkaitan Semantik.....	35

2.5.5.	Pembobotan TF-IDF	39
2.5.6.	Pengukuran Nilai Kemiripan	40
2.6	Algoritma <i>Fuzzy Similarity based K-Nearest Neighbor</i> (FSKNN)	40
2.6.1	Struktur Data Pelatihan.....	41
2.6.2	Proses Pelatihan	41
2.6.3	Proses <i>Testing</i>	45
2.7	Wordnet.....	46
2.8	Gwet's AC1	47
2.9	Evaluasi.....	50
2.10	Penelitian Sebelumnya	50
2.11	Kesimpulan Hasil Studi Komparasi	54
BAB 3 METODOLOGI PENELITIAN		55
3.1	Studi Literatur	55
3.2	Rancangan Arsitektur Deteksi dan Klasifikasi	56
3.2.1	Otomatisasi Data Pelatihan.....	58
3.2.2	Pengelompokan Pola Pelatihan	68
3.2.3	Penghitungan <i>Prior Probability</i> dan Nilai <i>likelihoods</i>	72
3.2.4	Pengklasifikasi FSKNN.....	76
3.3	Pembuatan Perangkat Lunak.....	78
3.4	Uji Coba Sistem	78
BAB 4 HASIL DAN PEMBAHASAN		81
4.1	Lingkungan Pengujian	81
4.2	Skenario Pengujian	82
4.3	Evaluasi Hasil Pengujian	83
4.3.1	Evaluasi Hasil Pengujian Skenario 1	83
4.3.2	Evaluasi Hasil Pengujian Skenario 2.....	84

4.3.3	Evaluasi Hasil Pengujian Skenario 3	86
4.4	Implementasi Sistem	87
BAB 5 KESIMPULAN DAN SARAN		101
5.1	Kesimpulan.....	101
5.2	Saran.....	102
DAFTAR PUSTAKA		103
LAMPIRAN.....		107
BIOGRAFI PENULIS		123

[Halaman ini sengaja dikosongkan]

DAFTAR GAMBAR

<i>Gambar 2. 1 ISO/IEC 25010:2011 untuk Quality Model</i>	27
Gambar 2.2 Pola Pengembangan Gabungan Hipernim dan Sinonim	32
Gambar 2.3 Representasi Relasi Hipernim dalam Wordnet (Chen, Tseng and Liang, 2010).....	34
<i>Gambar 2.4 Bagan Kelompok Knowledge-based Similarity Measures (Gomaa, 2013)</i>	35
Gambar 2.5 (a) Pola Jalur Diperbolehkan (b) Pola Jalur yang Tidak Diperbolehkan (Budanitsky and Hirst, 1998)	37
Gambar 2.6 Contoh Relasi Dengan Jalur yang Diperbolehkan dan Tidak Diperbolehkan (<i>Choudhari, 2012</i>).....	37
Gambar 2.7 Pseudo-code proses pengelompokan dan proses pendefinisian search set G_i (Jiang, 2012)	43
Gambar 3.1 Alur Metodologi Penelitian.....	55
Gambar 3. 2 Model deteksi dan klasifikasi non-fungsional	56
Gambar 3. 3Ilustrasi Perluasan Term dengan Gabungan Hipernim dan Sinonim	59
Gambar 3.4 Ilustrasi Struktur Relasi untuk <i>Term Functionality</i> dan <i>Term Suitability</i> pada kalimat kebutuhan KK2 Berbasis <i>Wordnet</i>	60
Gambar 4.1 Grafik rata-rata kinerja <i>accuracy</i> , <i>precision</i> , <i>recall</i> , dan F1 berdasarkan <i>ground truth</i> pada skenario 1	84
Gambar 4.2 Grafik tabulasi rata-rata kinerja <i>accuracy</i> , <i>precision</i> , <i>recall</i> , dan F1 menggunakan metode FSKNN pada skenario 1	84
Gambar 4.3 Grafik rata-rata kinerja <i>accuracy</i> , <i>precision</i> , <i>recall</i> , dan F1 berdasarkan <i>ground truth</i> pada skenario 2.....	85
Gambar 4.4 Grafik tabulasi rata-rata kinerja <i>accuracy</i> , <i>precision</i> , <i>recall</i> , dan F1 menggunakan metode FSKNN pada skenario 2	86
Gambar 4.5 Grafik tabulasi rata-rata kinerja <i>accuracy</i> , <i>precision</i> , <i>recall</i> , dan F1 menggunakan metode FSKNN pada skenario 3	86
Gambar 4.6 Potongan Praproses dalam sistem	89
Gambar 4.7 Pengembangan <i>term</i> pada sistem	90

Gambar 4.8 Hasil tahap pembobotan TF-IDF	90
Gambar 4.9 Hasil pengukuran derajat kemiripan dalam database	91
Gambar 4.10 Hasil Pengukuran Keterkaitan Semantik	92
Gambar 4.11 Hasil Pada Tahap Penentuan Distribusi <i>Term d</i> , <i>c</i> Pada Apek Kualitas <i>c_j</i>	92
Gambar 4.12 Hasil Pada Tahap Penentuan Distribusi <i>Term d</i> , <i>c</i> Pada Apek Kualitas <i>c_j</i>	93
Gambar 4.13 Hasil Pada Tahap Penentuan derajat keanggotaan μ , <i>c</i>	93
Gambar 4.14 Hasil Pada Tahap Penentuan derajat keanggotaan μ	93
Gambar 4.15 Hasil Pada Tahap Penentuan nilai similaritas <i>si</i> , <i>c</i>	94
Gambar 4.16 Hasil Pada Tahap Penentuan nilai derajat keanggotaan μ	95
Gambar 4.17 Hasil Pada Tahap Pengelompokan data pelatihan ke dalam beberapa kluster	95
Gambar 4.18 Hasil Pada Tahap Pencarian search set untuk setiap data Pelatihan	96
Gambar 4.19 Hasil Pada Tahap Penentuan Nilai Prior Probability.....	97
Gambar 4.20 Hasil Pada Tahap Penentuan jumlah k, (N^i), (n^i).....	97
Gambar 4.21 Hasil Pada Tahap Penentuan nilai Z , <i>j</i> dan Z , <i>j</i>	98
Gambar 4.22 Pada Tahap Penentuan nilai likelihoods <i>P</i>	98
Gambar 4.23 Hasil Klasifikasi Kalimat Uji 1.....	99
Gambar 4. 24 Hasil Klasifikasi Kalimat Uji 2.....	99
Gambar 4.25 Hasil Klasifikasi Kalimat Uji 3.....	100

DAFTAR TABEL

<i>Tabel 2.1 Definisi 31 Aspek Kualitas Berdasarkan ISO/IEC 25010</i>	28
Tabel 2.2 Daftar term relevan dengan 31 subatribut aspek kualitas ISO/IEC 25010:2011	32
Tabel 2.3 Tabel Penulisan Hasil Pengamatan	47
Tabel 2.4 Interpretasi Nilai Gwet AC1	49
Tabel 2.5 Data pengamatan Contoh Kasus Kappa.....	49
Tabel 2.6 Perbandingan Penelitian Berbasis Klasifikasi	53
Tabel 3.1 Tahap Praproses Kalimat Kebutuhan KK1	58
Tabel 3.2 Contoh hasil pengembangan term gabungan antara hipernim dan sinonim pada aspek kualitas functional correctness	59
Tabel 3.3 Ilustrasi Hasil Pengukuran keterkaitan Semantik Antara 10 Term Relevan dengan semua term pada kalimat kebutuhan KK2	61
Tabel 3.4 Ilustrasi Pencarian Term Frequency (TF) antara aspek kualitas functional correctness dengan kalimat kebutuhan KK2	62
Tabel 3.5 Ilustrasi Pencarian Term Frequency (TF) antara aspek kualitas functional correctness dengan kalimat kebutuhan KK1	63
Tabel 3.6 Ilustrasi Perhitungan IDF antara kualitas suitability dengan KK1	65
Tabel 3.7 Ilustrasi Hasil Perhitungan Bobot TF-IDF untuk aspek kualitas functional correctness dan kalimat kebutuhan KK1	66
Tabel 3.8 Daftar Nilai Derajat Kemiripan Antara Semua Aspek Kualitas dengan Kalimat Kebutuhan KK1	67
Tabel 3.9 Ilustrasi Hasil Otomatisasi Pelabelan untuk 4 Kalimat Kebutuhan	67
Tabel 3. 10 Ilustrasi Data Frekuensi 4 aspek kualitas terhadap 4 kalimat kebutuhan.....	67
Tabel 3.11 Hasil penentuan nilai dti, c_j dan $ddti, c_j$	68
Tabel 3.12 Hasil Penentuan Derajat Keanggotaan μ_{Rti, c_j}	69
Tabel 3.13 Hasil Penentuan Derajat Keanggotaan μ_{dti}	70
Tabel 3.14 Hasil Penentuan Similaritas Antara Kalimat Kebutuhan dengan Aspek Kualitas	71
Tabel 3.15 Hasil Penentuan Derajat Keanggotaan μ_{cjd}	71

Tabel 3.16 Hasil Penentuan Nilai Prior Probability	73
Tabel 3.17 Hasil Penentuan nilai N^i dan n^i	74
Tabel 3.18 Hasil Penentuan Nilai $Z_{e,j}$ dan $Z_{e,j}$	75
Tabel 3.19 Hasil Penentuan Nilai Likelihood PEH_j	75
Tabel 3.20 Daftar Frekuensi 37 Term Relevan dari Kalimat Uji	76
Tabel 3.21 Daftar Nilai Derajat Keanggotaan μ_{dti} Kalimat Uji	76
Tabel 3.22 Hasil Penentuan Nilai sim_d, c_j dan $\mu_{c_j dt}$ untuk data uji	77
Tabel 3.23 Data Search Set untuk Kalimat Uji	77
Tabel 3.24 Data tetangga terdekat dari dokumen uji (N^i) dan nilai vektor jumlah label (n^i) untuk Kalimat Uji	77
Tabel 4.1 Spesifikasi perangkat keras	81
Tabel 4.2 Spesifikasi perangkat lunak	81
Tabel 4.3 Tabulasi rata-rata kinerja accuracy, precision, recall, dan F1 berdasarkan <i>ground truth</i> pada skenario 1	83
Tabel 4.4 Tabulasi rata-rata kinerja accuracy, precision, recall, dan F1 menggunakan metode FSKNN dengan jumlah $K=10,20$, dan 30 pada skenario 184	84
Tabel 4.5 Tabulasi rata-rata kinerja accuracy, precision, recall, dan F1 berdasarkan <i>ground truth</i> pada skenario 2	85
Tabel 4.6 Tabulasi rata-rata kinerja accuracy, precision, recall, dan F1 menggunakan metode FSKNN dengan jumlah $K=30$ pada skenario 2	85
Tabel 4.7 Tabulasi rata-rata kinerja accuracy, precision, recall, dan F1 berdasarkan <i>ground truth</i> pada skenario 3	86
Tabel 4.8 Komposisi Kelas dari 225 Kalimat Kebutuhan	87

BAB 1

PENDAHULUAN

Pada Bab ini akan dijelaskan mengenai beberapa hal dasar dalam pembuatan proposal penelitian yang meliputi latar belakang, perumusan masalah, tujuan, manfaat, kontribusi penelitian, dan batasan masalah.

1.1. Latar Belakang

Kebutuhan non-fungsional yang berkaitan erat dengan aspek kualitas perangkat lunak merupakan salah satu faktor penting untuk mendukung kesuksesan pengembangan perangkat lunak (Ullah, Iqbal and Khan, 2011)(Ramadhani, Rochimah and Yuhana, 2015). Non-fungsional merupakan kendala pada layanan atau fungsi yang ditawarkan oleh sistem secara keseluruhan seperti *usability, flexibility, efficiency, availability, and portability* (Sommerville, 2010). Non-fungsional berfokus pada produktivitas yang buruk, pemrosesan yang lambat, biaya yang relatif tinggi, kualitas yang rendah, dan ketidakpuasan pelanggan (Chung and Leite, 2009).

Mengidentifikasi aspek kualitas kebutuhan non-fungsional merupakan hal yang sulit untuk dilakukan. Karena aspek kualitas kebutuhan non-fungsional tersebut ditemukan tidak lengkap, tersembunyi, dan tercampur di dalam kalimat-kalimat kebutuhan lain pada dokumen. Permasalahan tersebut menyebabkan aspek kualitas kebutuhan non-fungsional sering terlewatkan selama proses pengembangan sistem perangkat lunak (Rahimi, Mirakhorli and Cleland-Huang, 2014)(Casamayor, Godoy and Campo, 2010). Kalimat kebutuhan yang memiliki kaitan dengan aspek kualitas kebutuhan non-fungsional pada umumnya ditulis dalam bentuk bahasa alami. Kalimat kebutuhan yang ditulis dalam bahasa alami biasanya memiliki kerancuan. Kerancuan akan mempersulit analisis dalam mengidentifikasi aspek kualitas kebutuhan non-fungsional yang terdapat di dalamnya(Rashwan, Ormandjieva and Witte, 2013). Oleh karena itu dibutuhkan suatu cara untuk dapat mengidentifikasi aspek kualitas kebutuhan non-fungsional, salah satunya dengan cara melakukan klasifikasi dari kalimat-kalimat kebutuhan yang tertulis dalam dokumen kebutuhan tersebut (Ramadhani, Rochimah and Yuhana, 2015).

Ada empat metode yang digunakan untuk menyelesaikan permasalahan identifikasi aspek kualitas kebutuhan non-fungsional (Casamayor, Godoy and Campo, 2010)(Cleland-Huang *et al.*, 2006)(Suharso and Rochimah, 2012)(Ramadhani, Rochimah and Yuhana, 2015). Keempat metode tersebut dibagi menjadi dua pendekatan pembelajaran yaitu pendekatan pembelajaran semi-diawasi (*semi-supervised learning*) dan pembelajaran diawasi (*supervised learning*). Pendekatan yang paling sering digunakan adalah pendekatan pembelajaran diawasi (*supervised learning*).

Penelitian yang menggunakan pendekatan pembelajaran semi-diawasi untuk mengidentifikasi aspek kualitas kebutuhan non-fungsional berbasis algoritma *Naive Bayes* adalah Casamayor dkk (2010) (Casamayor, Godoy and Campo, 2010). Algoritma *Naive Bayes* digunakan untuk proses pelatihan yang digabungkan dengan metode *Expectation Maximization* (EM) yang bertujuan untuk mengurangi jumlah data latih. Proses pertama yaitu dengan melakukan praproses terhadap kalimat - kalimat kebutuhan untuk mendapatkan *term* kunci. Proses kedua adalah melakukan klasifikasi dengan metode gabungan tersebut. Metode EM terdiri dari 2 proses yaitu proses *E-step* dan *M-step*, dimana proses *E-step* berfungsi untuk mengisi data yang hilang berdasarkan estimasi pada parameter dan pada proses *M-step* melakukan re-estimasi parameter untuk mendapatkan kemungkinan termaksimal. Dalam penelitian menggunakan metode gabungan *Naive Bayes* dengan EM ini, kalimat kebutuhan yang tidak terlabeli dianggap kehilangan data karena kurangnya label kelas tersebut.

Metode *Naive Bayes* dengan menggunakan pendekatan semi-diawasi telah terbukti mampu melakukan identifikasi aspek kualitas kebutuhan non-fungsional dari kalimat kebutuhan dengan baik, tetapi metode ini hanya mampu mengklasifikasikan kalimat kebutuhan non-fungsional terbatas ke dalam satu aspek kualitas (*single-label*). Sedangkan menurut Huang dkk (2010) dan Rashwan dkk (2013) (Cleland-Huang *et al.*, 2006)(Rashwan, Ormandjieva and Witte, 2013) ada kemungkinan bahwa satu opsi kalimat kebutuhan dalam dokumen kebutuhan mengandung lebih dari satu aspek kebutuhan non-fungsional (*multi-label*). Oleh karena itu dibutuhkan suatu cara untuk dapat mengidentifikasi aspek kualitas kebutuhan non-fungsional ke dalam lebih dari satu aspek kualitas (*multi-label*).

Ada tiga penelitian utama terkait pengidentifikasian aspek kualitas kebutuhan non-fungsional yang menggunakan pendekatan pembelajaran diawasi. Penelitian pertama adalah mengidentifikasi aspek kualitas kebutuhan non-fungsional berbasis algoritma klasifikasi model probabilistik (Cleland-Huang *et al.*, 2006). Penelitian ini terdiri dari dua tahap yaitu tahap awal dan tahap pengembangan. Tahap awal pada penelitian ini terdiri dari dua proses. Proses pertama adalah menyediakan sekumpulan kata kunci baku. Kata kunci baku tersebut diambil dari katalog metode operasionalisasi softgoal keamanan dan kinerja pada *Softgoal interdependency graphs* (SIG) (Chung and Leite, 2009). Sedangkan proses kedua adalah menggunakan kata kunci baku tersebut untuk mengklasifikasi setiap tipe non-fungsional dari kebutuhan yang tidak terklasifikasi ke dalam lebih dari satu aspek kualitas (*multi-label*). Tahap awal pada penelitian ini diidentifikasi memiliki masalah yaitu kesulitan dalam menemukan katalog yang dapat diterima dan terstandarisasi pada tipe non-fungsional lain, sehingga tidak dievaluasi lebih lanjut (Cleland-Huang *et al.*, 2006).

Tahap pengembangan pada penelitian ini terdiri dari tiga proses yaitu pelabelan data latih, penambahan *term* indikator, dan klasifikasi. Tahap pengembangan pada penelitian ini memiliki tingkat subjektivitas yang tinggi karena proses pelabelan data latih dilakukan secara manual. Disamping itu, banyaknya aspek kualitas yang ditawarkan oleh model-model kualitas seperti model *McCall*, *Boehm*, *Dromey*, *FURPS*, *BNN*, *Star*, dan *ISO* menyebabkan kebingungan dalam menentukan aspek kualitas (Behkamal, Kahani and Akbari, 2009).

Penelitian kedua adalah pelabelan otomatis aspek kualitas kebutuhan non-fungsional berbasis ISO/IEC 9126:2001 (Suharso and Rochimah, 2012). Penelitian ini terdiri dari tiga proses yaitu pelabelan otomatis data pelatihan, penambahan *term* indikator, dan klasifikasi. Penelitian ini belum mempertimbangkan faktor hubungan semantik antara *term-term* relevan dengan setiap aspek kualitas yang diproses selama pelatihan. Faktanya faktor semantik dapat meningkatkan hasil klasifikasi menjadi lebih baik (Jensen and Martinez, 2000) (Çelik, 2013).

Penelitian ketiga adalah klasifikasi aspek kualitas kebutuhan non-fungsional menggunakan semantic-FSKNN berbasis ISO/IEC 9126:2001 (Ramadhani, Rochimah and Yuhana, 2015). Penelitian ini terdiri dari tiga proses yaitu pelabelan otomatis data pelatihan, pengukuran keterkaitan semantik, dan klasifikasi. Proses pelabelan otomatis data pelatihan dilakukan untuk menghemat lebih banyak waktu dibandingkan proses pelabelan dilakukan secara manual, proses ini dilakukan dengan pembobotan TF-IDF dengan pencarian tingkat kemiripan menggunakan *Cosine Measure* (Suharso and Rochimah, 2012). Proses pengukuran semantik dilakukan dengan menggunakan metode HSO (Hirst and Stone, 1995) untuk mendapatkan hubungan semantik antar *term-term* relevan setiap aspek kualitas dengan setiap aspek kualitas. Sedangkan untuk proses klasifikasi, menggunakan metode FSKNN yang telah diperkenalkan oleh Jiang dkk (2012) (Jiang, Tsai and Lee, 2012). Pada proses ini dilakukan penambahan variabel baru dalam fase pengelompokan pola pelatihan ke dalam kluster. Variabel tersebut diperoleh dari nilai keterkaitan semantik dalam proses sebelumnya. Proses penambahan variabel semantik pada metode FSKNN disebut semantic-FSKNN. Kelas yang digunakan untuk proses klasifikasi berdasarkan Aspek kualitas ISO/IEC 9126:2001.

Penelitian ketiga diidentifikasi mempunyai tiga permasalahan. Permasalahan pertama adalah aspek kualitas yang digunakan sebagai acuan masih menggunakan standart lama yaitu ISO/IEC 9126:2001. Menurut Al-Kilidar dkk (2005) (Al-Kilidar, Cox and Kitchenham, 2005), ISO/IEC 9126:2001 tidak cocok untuk mengukur kualitas perangkat lunak. Ini menimbulkan keraguan serius tentang validitas standar secara keseluruhan. Buruknya kualitas perangkat lunak dapat menyebabkan kerugian (Al-Qutaish, 2010). Disamping itu, evaluasi kritis dari penelitian yang dilakukan oleh Al-Kilidar dkk (2005) (Al-Kilidar, Cox and Kitchenham, 2005) dan Al-Qutaish (2010) (Al-Qutaish, 2010) mengungkapkan *ambiguity* pada enam sub-atribut *compliance* pada struktur hirarki ISO/IEC 9126:2001. Sejalan dengan evaluasi ini, Behkamal dkk (2009) (Behkamal, Kahani and Akbari, 2009) hanya menggunakan 21 sub-atribut dari total 27 sub-atribut yang dimiliki oleh ISO/IEC 9126:2001.

Permasalahan kedua adalah pengembangan *term* menggunakan 2 level hipernim dapat mengurangi kinerja akurasi. Berdasarkan hasil penelitian yang dilakukan oleh Ramadhani dkk (2015) (Ramadhani, Rochimah and Yuhana, 2015) permasalahan tersebut terjadi karena penggunaan hipernim yang terlalu banyak dapat menghilangkan makna asli dari *term* yang dikembangkan. Menurut Rozi dkk (2015) (Rozi, Faticah and Purwitasari, 2015) penggunaan hipernim mampu meningkatkan akurasi. Namun, penggunaan hipernim dalam beberapa dataset hanya akan menambah jumlah *term* yang akan dianggap sebagai noise sehingga mengurangi akurasi.

Permasalahan ketiga adalah pengukuran keterkaitan semantik dilakukan antar *term-term* relevan setiap aspek kualitas dengan setiap aspek kualitas. Hasil dari pengukuran semantik tersebut akan dijadikan variabel baru berupa faktor semantik ke dalam proses algoritma FSKNN. Variabel baru tersebut akan ditambahkan pada tahap pengelompokan pola training ke dalam kluster. Evaluasi kritis dari penelitian Ramadhani dkk (2015) (Ramadhani, Rochimah and Yuhana, 2015) mengungkapkan perubahan metode dari FSKNN ke semantik-FSKNN tidak memiliki perbedaan yang signifikan. Hal ini dikarenakan penambahan variabel tersebut membuat data yang dibentuk untuk pola pelatihan tersaring lebih ketat sehingga membuat kinerja klasifikasi lebih kecil dibandingkan metode FSKNN.

Penelitian ini mengajukan tiga solusi dari permasalahan di atas. Solusi pertama adalah menggunakan ISO/IEC 25010:2011 sebagai acuan untuk mengukur aspek kualitas. ISO/IEC 25010:2011 dipilih sebagai acuan karena merupakan peningkatan dari ISO/IEC 9126:2001 (Saptarini, Rochimah and Yuhana, 2016). ISO/IEC 25010:2011 adalah bagian dari rangkaian *standart international SquaRE* dan telah menggantikan ISO/IEC 9126:2001 pada tahun 2011. Disamping itu, ISO/IEC 25010:2011 menambahkan beberapa amandemen seperti *security* dari sub-atribut dirubah menjadi atribut dan memiliki lima sub-atribut baru. ISO/IEC 25010:2011 mendefinisikan model kualitas perangkat lunak, yang terdiri dari delapan atribut meliputi *functional suitability*, *reliability*, *performance efficiency*, *usability*, *security*, *compatibility*, *maintainability* dan *portability*. Setiap atribut terbagi menjadi beberapa sub-atribut (Kadi, Idri and Ouhbi, 2016).

Solusi kedua adalah pengembangan *term* gabungan akan menggunakan dua skenario. Skenario pertama menggunakan 2 level hipernim dan 9 sinonim pada masing-masing hipernim seperti penelitian sebelumnya (Ramadhani, Rochimah and Yuhana, 2015). Sedangkan skenario kedua pengembangan *term* menggunakan 1 level hipernim dan 20 sinonim. Skenario kedua diharapkan dapat meningkatkan kinerja klasifikasi.

Solusi ketiga adalah pengukuran keterkaitan semantik akan dilakukan pada proses otomatisasi data pelatihan. Penerapan Pengukuran semantik pada proses otomatisasi data pelatihan ini akan menggunakan dua skenario. Skenario pertama adalah menghasilkan data pelatihan secara otomatis tanpa melakukan pengukuran keterkaitan semantik menggunakan 1 level hipernim dan 20 sinonim. Sedangkan skenario kedua adalah data pelatihan pada skenario pertama ditambahkan pengukuran keterkaitan semantik antar *term-term* relevan setiap aspek kualitas dengan *term-term* pada dokumen yang akan dilabeli. Kedua data latih tersebut akan digunakan pada proses klasifikasi. Penambahan pengukuran semantik pada data latih yang diperoleh dari proses otomatisasi data pelatihan ini dilakukan dengan tujuan untuk menghasilkan data latih yang lebih baik. Skenario kedua diharapkan dapat menghasilkan akurasi yang lebih tinggi dibandingkan dengan skenario pertama.

Hasil dari penelitian ini dapat membantu para analis kebutuhan untuk mengetahui aspek kualitas kebutuhan non-fungsional apa saja yang diperlukan untuk mendukung kesuksesan proses pengembangan perangkat lunak. Aspek kualitas kebutuhan non-fungsional yang telah diketahui sejak tahap awal analisa kebutuhan diharapkan dapat meminimalisir hal-hal yang tidak diinginkan ketika produk perangkat lunak sudah dibuat.

1.2. Perumusan Masalah

Rumusan masalah yang diangkat dalam penelitian ini adalah sebagai berikut:

1. Bagaimana cara menghasilkan data latih secara otomatis berdasarkan aspek kualitas ISO/IEC 25010:2011?
2. Bagaimana mengetahui performa otomatisasi data pelatihan berdasarkan *ground truth* dari dua skenario antara data latih yang diperoleh dari

pengembangan *term* menggunakan 2 level hipernim dengan 9 sinonim dan pengembangan *term* menggunakan 1 level hipernim dengan 20 sinonim?

3. Bagaimana menambahkan pengukuran keterkaitan semantik antara *term-term* relevan setiap aspek kualitas dengan *term-term* pada dokumen yang akan dilabeli ke dalam data latih yang diperoleh pada proses otomatisasi data pelatihan?

1.3. Tujuan Penelitian

Tujuan yang akan dicapai dalam penelitian ini adalah membangun kerangka kerja untuk melakukan otomatisasi data pelatihan berdasarkan aspek kualitas ISO/IEC 25010:2011. Hasil dari proses ini adalah semua kalimat kebutuhan yang sudah terlabeli berdasarkan aspek kualitas ISO/IEC 25010:2011. Pelabelan yang dilakukan meliputi pengembangan *term* menggunakan hipernim dan sinonim serta pengukuran keterkaitan semantik menggunakan metode HSO. Pengukuran semantik dilakukan antara *term-term* relevan setiap aspek kualitas dengan *term-term* pada dokumen yang akan dilabeli. Selanjutnya yaitu menghitung bobot berdasarkan term frekuensi(*tf*), inverse dokumen frekuensi(*idf*), *tf-idf*, dan nilai *similarity*. Proses terakhir akan dilakukan klasifikasi untuk mengetahui perbedaan data latih yang diperoleh dari dua level hipernim dan data latih baik yang memperhatikan pengukuran keterkaitan semantik maupun tidak.

1.4. Manfaat Penelitian

Manfaat dari penelitian ini adalah membantu para analis kebutuhan dalam menganalisa aspek kualitas kebutuhan non-fungsional apa saja yang harus dipenuhi untuk mendukung kesuksesan proses pengembangan perangkat lunak. Aspek kualitas kebutuhan non-fungsional yang telah diketahui sejak tahap awal analisa kebutuhan diharapkan dapat meminimalisir hal-hal yang tidak diinginkan ketika produk perangkat lunak sudah dibuat.

1.5. Kontribusi Penelitian

Kontribusi yang diharapkan dari penelitian ini adalah melakukan klasifikasi untuk mengetahui aspek kualitas kebutuhan non-fungsional apa saja yang harus dipenuhi untuk mendukung kesuksesan proses pengembangan perangkat lunak. Sebelum melakukan pengklasifikasian, dibutuhkan otomatisasi data pelatihan

menggunakan aspek kualitas ISO/IEC 25010:2011. Proses otomatisasi data pelatihan akan menghasilkan data latih yang diperoleh dari pengembangan *term* menggunakan 1 level hipernim dengan 20 sinonim. Selain itu, data latih yang diperoleh dari proses otomatisasi data pelatihan akan dilakukan penambahan *term* frekuensi yang diperoleh dari pengukuran semantik antara *term-term* relevan setiap aspek kualitas dengan *term-term* pada dokumen yang akan dilabeli.

1.6. Batasan Masalah

Batasan masalah pada penelitian ini adalah:

1. Sebagaimana penelitian sebelumnya yang dilakukan oleh Suharso dkk (2012) dan Ramadhani dkk (2012) (Suharso and Rochimah, 2012)(Ramadhani, Rochimah and Yuhana, 2015), dalam penelitian ini aspek kualitas yang digunakan hanya sub-atribut dari aspek kualitas ISO/IEC 25010:2011.
2. Dokumen yang digunakan dalam proses klasifikasi berada pada tingkatan *sentences level* pada dokumen kebutuhan.
3. Dataset berupa gabungan domain daftar spesifikasi kebutuhan perangkat lunak *voip* dan *geolocation* dalam Bahasa Inggris.

BAB 2

KAJIAN PUSTAKA

Bab ini merupakan pembahasan dari referensi terkait yang telah dilakukan dalam menyelesaikan permasalahan sesuai dengan uraian pada latar belakang. Bab ini diawali dengan menjabarkan hal-hal yang diterapkan pada metode yang diusulkan, kelemahan yang terdapat pada penelitian sebelumnya, komparasi penelitian sebelumnya. Selanjutnya dilanjutkan dengan kelebihan dari metode yang akan digunakan untuk menyelesaikan permasalahan deteksi aspek kualitas kebutuhan non-fungsional, metode usulan.

2.1 Kebutuhan Non-Fungsional

Kebutuhan non-fungsional dapat dianggap sebagai kriteria kualitas atau kinerja yang harus dipenuhi oleh sistem perangkat lunak (Xiang *et al.*, 2015). Non-fungsional adalah elemen kunci yang perlu ditangani selama proses pengembangan. Non-fungsional dapat bersifat subjektif, relatif, beragam dan interaktif (Pavel Kruliš, 2012). Kebutuhan non-fungsional sering juga disebut aspek kualitas dari sebuah sistem. Kebutuhan non-fungsional atau aspek kualitas merupakan sebuah konsep penting dalam rekayasa kebutuhan. Konsep tersebut menentukan kesuksesan atau kegagalan sebuah sistem perangkat lunak (Khatler and Kalia, 2013).

Kebutuhan non-fungsional dapat digolongkan menjadi 2 bagian, yaitu kebutuhan non-fungsional primer dan kebutuhan non-fungsional spesifik. Yang dimaksud kebutuhan non-fungsional primer adalah abstraksi dari tingkat tertinggi suatu kebutuhan non-fungsional yang dapat diurai menjadi kebutuhan non-fungsional spesifik, sedangkan kebutuhan non-fungsional spesifik memiliki tingkat yang lebih detail dan menunjukkan aspek yang terhubung dengan kebutuhan non-fungsional primer (Khatler and Kalia, 2013). Jika mengacu pada ISO/IEC 25010:2011, yang tergolong kebutuhan non fungsional primer seperti *functionality suitability, performance efficiency, compatibility, usability, reliability, security, maintainability, portability*, sedangkan kebutuhan non-fungsional spesifik, salah satu contoh yang dapat diambil dari *usability* yaitu *appropriateness recognizability, learnability, operability, user error protection*,

user interface aesthetics, dan *accessibility*. Detail kebutuhan non-fungsional primer dan spesifik menurut ISO/IEC 25010:2011 dijabarkan pada Gambar 2.1.

Walaupun kebutuhan non-fungsional dipertimbangkan selama proses pengembangan sistem perangkat lunak, tetapi belum ada kesepakatan dalam komunitas rekayasa kebutuhan tentang bagaimana kebutuhan non-fungsional dapat dielisitasi, direpresentasikan dan diintegrasikan. Kondisi inilah yang memicu permasalahan utama dalam kebutuhan non-fungsional yaitu ketiadaan ketentuan formal untuk melakukan elisitasi atau identifikasi kebutuhan non-fungsional (Khatter and Kalia, 2013).

Oleh karena itu fokus dalam penelitian ini adalah untuk mencari cara melakukan elisitasi atau identifikasi kebutuhan non-fungsional dengan hasil yang dapat memberikan akurasi yang tinggi.

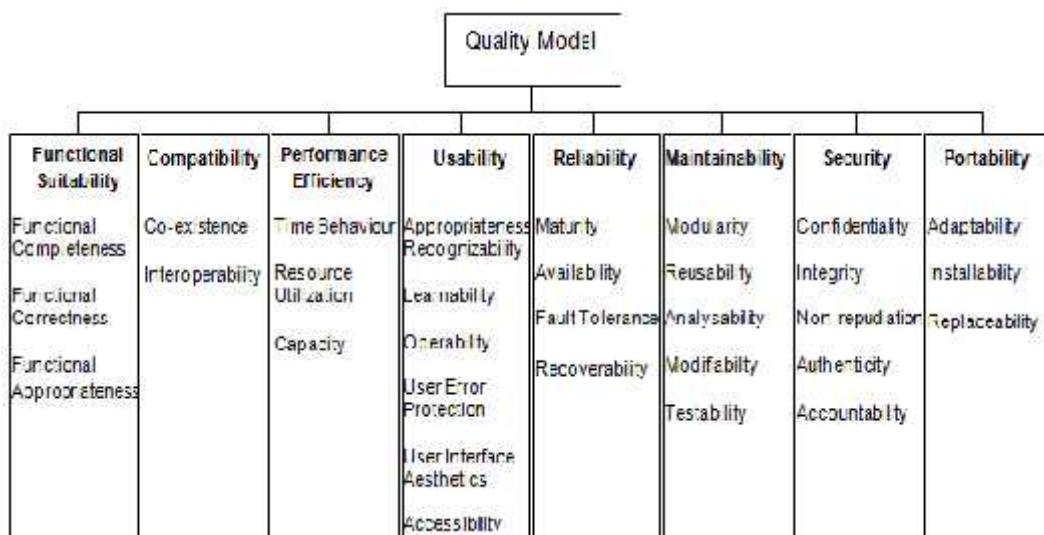
2.2 ISO/IEC 25010:2011

ISO 25010:2011 disiapkan oleh *Joint Technical Committee ISO/IEC JTC 1, Information technology, subcommittee SC 7, Software and system engineering*. Edisi pertama ISO/IEC 25010:2011 membatalkan dan menggantikan ISO/IEC 9126-1: 2001, yang telah direvisi secara teknis. ISO/IEC 25010:2011 adalah bagian dari rangkaian *Standart International SquaRE*, yang terdiri dari divisi-divisi berikut :

1. *Quality Management Division* (ISO/IEC 2500n).
2. *Quality Model Division* (ISO/IEC 2501n).
3. *Quality Measurement Division* (ISO/IEC 2502n).
4. *Quality Requirements Division* (ISO/IEC 2503n).
5. *Quality Evaluation Division* (ISO/IEC 2504n).
6. *SquaRE Extension Division* (ISO/IEC 25050 - ISO/IEC 25099).

Mengukur kualitas sistem dengan cara seberapa baik suatu sistem dapat memenuhi berbagai kebutuhan pemangku kepentingan. Oleh karena itu, perlu untuk mempertimbangkan kualitas dari perspektif pemangku kepentingan yang berbeda. Model-model kualitas menyediakan kerangka untuk mengumpulkan kebutuhan pemangku kepentingan. Di antara dokumen dalam standar ISO / IEC, 25010:2011 "Model Kualitas" mendefinisikan model kualitas produk yang terdiri dari delapan atribut yang selanjutnya dibagi lagi menjadi sub-atribut (Gambar

2.1). Model ini dipahami sebagai model struktural yang didefinisikan oleh SQuaRE sebagai; "Kualitas model: set karakteristik yang ditetapkan, dan hubungan di antara mereka.". Standar ini juga memberikan definisi dari masing-masing atribut kualitas dan sub-atribut (Febrero, Calero and Moraga, 2016).



Gambar 2. 1 ISO/IEC 25010:2011 untuk Quality Model

Strategi Dekomposisi Hierarki ini dimaksudkan untuk menyediakan sarana untuk menangani kompleksitas konseptual. Penanganan kompleksitas tersebut bertujuan untuk menggambarkan Kualitas sebagai properti multidimensi. Selain itu, Dekomposisi Hierarki ini juga menyediakan sarana untuk memperoleh atribut terukur sederhana yang cocok untuk kombinasi lebih lanjut ke dalam indeks kualitas (Febrero, Calero and Moraga, 2016).

Dalam penelitian ini yang digunakan sebagai dasar acuan dalam model aspek kualitas adalah dokumen ISO 25010:2011 yang berisi 1 model kualitas yaitu kualitas produk perangkat lunak. Model tersebut terdiri dari delapan atribut kualitas yang terbagi lagi menjadi 31 sub-atribut kualitas, detail sub-atribut tersebut dapat dilihat pada Gambar 2.1. Pada penelitian ini model kualitas produk ISO 25010:2011 yang digunakan sebagai dasar acuan model kualitas untuk melakukan identifikasi aspek kualitas dari kalimat kebutuhan, tetapi hanya 31 sub-atribut yang digunakan karena sub-atribut lebih spesifik dari atribut. Disamping itu, beberapa penelitian sebelumnya lebih cenderung menggunakan sub-atribut dibandingkan atribut (Suharso and Rochimah, 2012)(Ramadhani,

Rochimah and Yuhana, 2015). Definisi dari 31 sub-atribut tersebut ditunjukkan pada Tabel 2.1.

Tabel 2.1 Definisi 31 Aspek Kualitas Berdasarkan ISO/IEC 25010

No.	SubAtribut	Definisi
1	<i>Functional completeness</i>	<i>The set of functions covers all the specified tasks and user objectives.</i>
2	<i>Functional correctness</i>	<i>The correct results with the needed degree of precision.</i>
3	<i>Functional appropriateness</i>	<i>The functions facilitate the accomplishment of specified tasks and objectives.</i>
4	<i>Time behaviour</i>	<i>The response and processing times and throughput rates of a product or system, when performing its functions, meet requirements.</i>
5	<i>Resource utilization</i>	<i>The amounts and types of resources used by a product or system, when performing its functions, meet requirements.</i>
6	<i>Capacity</i>	<i>The maximum limits of a product or system parameter meet requirements.</i>
7	<i>Co-existence</i>	<i>perform its required functions efficiently while sharing a common environment and resources with other products, without detrimental impact on any other product.</i>
8	<i>Interoperability</i>	<i>Exchange information and use the information that has been exchanged.</i>
9	<i>Appropriateness recognizability</i>	<i>Users can recognize whether a product or system is appropriate for their needs.</i>
10	<i>Learnability</i>	<i>a product or system can be used by specified users to achieve specified goals of learning to use the product or system with effectiveness, efficiency, freedom from risk and satisfaction in a specified context of use.</i>
11	<i>Operability</i>	<i>Easy to operate and control.</i>
12	<i>User error protection</i>	<i>Protects users against making errors.</i>
13	<i>User interface aesthetics</i>	<i>Enables pleasing and satisfying interaction for the user.</i>
14	<i>Accessibility</i>	<i>System can be used by people with the widest range of characteristics and capabilities to achieve a specified goal in a specified context of use.</i>
15	<i>Maturity</i>	<i>Meets needs for reliability under normal operation.</i>

No.	SubAtribut	Definisi
16	<i>Availability</i>	<i>Operational and accessible when required for use.</i>
17	<i>Fault tolerance</i>	<i>Operates as intended despite the presence of hardware or software faults.</i>
18	<i>Recoverability</i>	<i>The event of an interruption or a failure, a product or system can recover the data directly affected and re-establish the desired state of the system.</i>
19	<i>Confidentiality</i>	<i>System ensures that data are accessible only to those authorized to have access.</i>
20	<i>Integrity</i>	<i>Prevents unauthorized access to, or modification of, computer programs or data.</i>
21	<i>Non-repudiation</i>	<i>Proven to have taken place, so that the events or actions cannot be repudiated later.</i>
22	<i>Accountability</i>	<i>The actions of an entity can be traced uniquely to the entity.</i>
23	<i>Authenticity</i>	<i>The identity of a subject or resource can be proved to be the one claimed.</i>
24	<i>Modularity</i>	<i>Computer program is composed of discrete components such that a change to one component has minimal impact on other components.</i>
25	<i>Reusability</i>	<i>An asset can be used in more than one system, or in building other assets.</i>
26	<i>Analysability</i>	<i>Effectiveness and efficiency with which it is possible to assess the impact on a product or system of an intended change to one or more of its parts, or to diagnose a product for deficiencies or causes of failures, or to identify parts to be modified.</i>
27	<i>Modifiability</i>	<i>A product or system can be effectively and efficiently modified without introducing defects or degrading existing product quality.</i>
28	<i>Testability</i>	<i>Effectiveness and efficiency with which test criteria can be established for a system, product or component and tests can be performed to determine whether those criteria have been met.</i>
29	<i>Adaptability</i>	<i>A product or system can effectively and efficiently be adapted for different or evolving hardware, software or other operational or usage environments.</i>
30	<i>Installability</i>	<i>Effectiveness and efficiency with which a product or system can be successfully installed and/or uninstalled in a specified environment.</i>

No.	SubAtribut	Definisi
31	<i>Replaceability</i>	<i>Replace another specified software product for the same purpose in the same environment.</i>

2.3 Klasifikasi Teks

Klasifikasi teks adalah suatu proses otomatisasi untuk menetapkan suatu masukan teks ke dalam label atau kelas yang memiliki hubungan dengan teks tersebut. Klasifikasi teks dibagi menjadi dua yaitu berdasarkan jumlah pelabelan dan tingkatan masukan teks yang digunakan. Berdasarkan jumlah pelabelan sendiri dibagi menjadi dua yaitu klasifikasi *single-label* dan *multi-label*, sedangkan berdasarkan tingkatan masukan teks yang digunakan juga dibagi menjadi dua yaitu *sentence level* dan *document level* (Pan, 2006).

Dalam penelitian ini jenis klasifikasi yang digunakan untuk melakukan identifikasi kebutuhan non-fungsional adalah klasifikasi *multi-label* dengan masukan teks berada pada *sentence level* karena berupa opsi-opsi kalimat kebutuhan dalam bentuk kalimat singkat yang tertulis dalam dokumen kebutuhan.

2.4 Klasifikasi *Multi-Label*

Dalam penjelasan sebelumnya klasifikasi *multi-label* merupakan proses klasifikasi teks yang dapat mengklasifikasikan teks ke dalam beberapa label atau kelas secara bersamaan. Klasifikasi *multi-label* dalam penyelesaiannya memiliki dua pendekatan yang berbeda yaitu metode transformasi masalah dan metode adaptasi algoritma (Prajapati, Thakkar and Ganatra, 2012).

Dalam penelitian ini jenis pendekatan yang digunakan adalah metode adaptasi algoritma. Metode adaptasi algoritma digunakan untuk menyelesaikan permasalahan *multi-label*. Metode yang digunakan mengadaptasi pada algoritma yang ada untuk menyelesaikan permasalahan *single-label*. Algoritma tersebut dikembangkan agar dapat menyelesaikan permasalahan *multi-label* tersebut (Prajapati, Thakkar and Ganatra, 2012). Beberapa algoritma yang digunakan dalam metode ini seperti ML-KNN, BPMLL, dan FSKNN. Dalam penelitian ini algoritma yang digunakan untuk klasifikasi *multi-label* pada proses identifikasi kebutuhan non-fungsional adalah algoritma *FSKNN* (Jiang, Tsai and Lee, 2012).

2.5 Algoritma Otomatisasi Data Pelatihan

Pemrosesan bahasa alami merupakan teknik untuk mengajarkan komputer dalam memahami maksud dari kata-kata yang digunakan oleh manusia. Metode ini lah yang kemudian diadaptasi oleh bidang Rekaya Perangkat Lunak salah satunya dalam memproses aspek perubahan dari suatu produk perangkat lunak.

2.5.1. Praproses

Salah satu teknik pemrosesan bahasa alami yang kerap digunakan pada pra proses adalah teknik-teknik berikut, yaitu:

1. Tokenisasi

Pada tahap ini, input kalimat kebutuhan dipecah menjadi unit terkecil. Biasanya unit tersebut berupa kata-kata atau kalimat atau paragraf. Contoh dari tokenisasi misalnya salah satu kalimat kebutuhan yaitu “*the system must be maintainable*” akan ditokenisasi menjadi “*the*”, “*system*”, “*must*”, “*be*”, dan “*maintainable*”.

2. Case Folding

Merubah semua huruf menjadi huruf kecil. Huruf yang diterima dalam proses case folding adalah huruf “a” hingga “z”, karakter selain huruf yang diterima akan dihilangkan dan dianggap delimiter.

3. Stopwords Removal

Stopwords Removal merupakan proses menghilangkan kata, tanda baca, simbol dan bilangan yang tidak berpotensi menjadi indikator topik pada kalimat kebutuhan. Contoh dari stopwords removal yaitu “*a*”, “*an*”, “*and*”, “*as*”, “*be*”, tanda baca koma dan sebagainya.

4. Stemming

Stemming memiliki peran untuk menjadikan teks menjadi kata dasar. Misal dari contoh kalimat kebutuhan pada tahap sebelumnya maka kata “*maintainable*” akan diubah menjadi “*maintan*”.

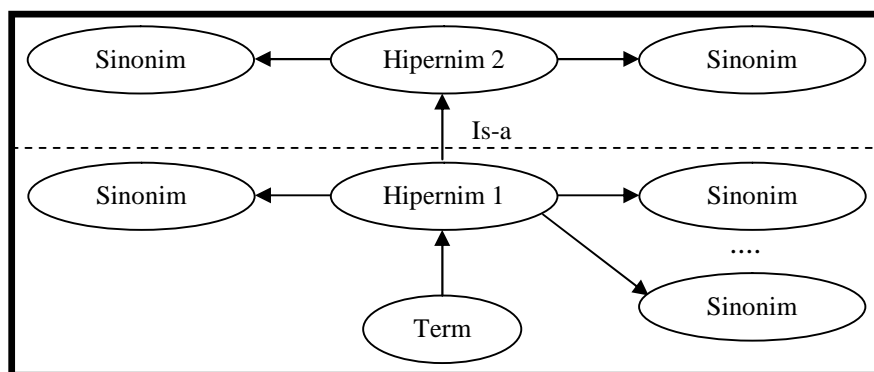
2.5.2. Pengembangan Term Gabungan antara Hipernim dan Sinonim

Pada tahap ini pengembangan akan dilakukan dengan menggunakan *term* gabungan antara hipernim dan sinonim. Pengembangan *term* tersebut disesuaikan dengan penelitian sebelumnya yang dilakukan oleh Suharso dkk (2013) dan

Ramadhani dkk (2015) (Suharso and Rochimah, 2012) (Ramadhani, Rochimah and Yuhana, 2015).

Pengembangan *term* gabungan antara hipernim dan sinonim hanya dilakukan pada *term* yang berkaitan langsung dengan ISO/IEC 25010:2011. Hal ini dikarenakan *term-term* tersebut akan dijadikan acuan dasar untuk melakukan proses klasifikasi kalimat kebutuhan. Disamping itu, pengembangan *term* pada penelitian ini disesuaikan dengan evaluasi kritis yang dilakukan oleh Ramadhani dkk (2015) bahwa penggunaan hipernim terlalu banyak dikhawatirkan akan menghilangkan makna asli dari *term* yang dikembangkan. Untuk membuktikan hipotesa ini maka pengembangan *term* gabungan dilakukan menggunakan dua skenario. Skenario pertama menggunakan 2 level hipernim dan 9 sinonim pada masing-masing hipernim seperti penelitian sebelumnya (Ramadhani, Rochimah and Yuhana, 2015). Sedangkan skenario kedua pengembangan *term* menggunakan 1 level hipernim dan 20 sinonim.

Pola pengembangan term gabungan antara hipernim dan sinonim ditunjukkan pada Gambar 2.2. Sedangkan *term-term* relevan dengan 31 sub-atribut aspek kualitas dari ISO/IEC 25010:2011 yang akan digunakan dalam penelitian ini ditunjukkan pada Tabel 2.2.



Gambar 2.2 Pola Pengembangan Gabungan Hipernim dan Sinonim

Tabel 2.2 Daftar term relevan dengan 31 subatribut aspek kualitas ISO/IEC 25010:2011

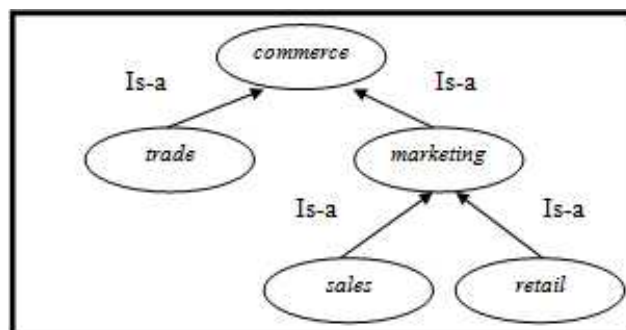
No.	SubAtribut	Term Relevan
1	<i>Functional completeness</i>	<i>Functional, suitability, completeness, covers, specified, tasks, objectives.</i>
2	<i>Functional correctness</i>	<i>Functional, suitability, correctness, results, needed, degree, precision.</i>

No.	SubAtribut	Term Relevan
3	<i>Functional appropriateness</i>	<i>Functional, suitability, appropriateness, facilitate, accomplishment, tasks, objectives.</i>
4	<i>Time behaviour</i>	<i>Performance, efficiency, time, behaviour, response, processing, throughput, rates, functions.</i>
5	<i>Resource utilization</i>	<i>Performance, efficiency, resource, utilization, amounts, functions.</i>
6	<i>Capacity</i>	<i>Performance, efficiency, capacity, maximum, limits, parameter.</i>
7	<i>Co-existence</i>	<i>Compatibility, coexistence, required, sharing, environment, detrimental, impact.</i>
8	<i>Interoperability</i>	<i>Compatibility, interoperability, exchange, information.</i>
9	<i>Appropriateness recognizability</i>	<i>Usability, appropriateness, recognizability, users, system, needs.</i>
10	<i>Learnability</i>	<i>Usability, learn, achieve, goals, freedom, risk, satisfaction.</i>
11	<i>Operability</i>	<i>Usability, operability, easy, control.</i>
12	<i>User error protection</i>	<i>Usability, user, error, protection, making.</i>
13	<i>User interface aesthetics</i>	<i>Usability, user, interface, aesthetics, enables, pleasing, satisfying, interaction.</i>
14	<i>Accessibility</i>	<i>Usability, access, system, people, wide, range, characteristics, capability.</i>
15	<i>Maturity</i>	<i>Reliability, maturity, under, normal, operation.</i>
16	<i>Availability</i>	<i>Reliability, available, accessible, required.</i>
17	<i>Fault tolerance</i>	<i>Reliability, fault, tolerance, intended, despite, presence, hardware, software.</i>
18	<i>Recoverability</i>	<i>Reliability, recover, event, interruption, failure, data, directly, affected, desired.</i>
19	<i>Confidentiality</i>	<i>Security, confidentiality, ensures, data, accessible, authorized.</i>
20	<i>Integrity</i>	<i>Security, integrity, prevents, unauthorized, access, modify, data.</i>
21	<i>Non-repudiation</i>	<i>Security, repudiation, verify, place, events, actions, repudiated, later.</i>
22	<i>Accountability</i>	<i>Security, accountability, traced, uniquely, entity.</i>
23	<i>Authenticity</i>	<i>Security, authenticity, identity, subject, resource, proved, claimed.</i>
24	<i>Modularity</i>	<i>Maintain, modular, composed, discrete, components,</i>

No.	SubAtribut	Term Relevan
		<i>change, minimal, impact.</i>
25	<i>Reusability</i>	<i>Maintain, reusable, asset, more, than, system, building, other.</i>
26	<i>Analysability</i>	<i>Maintain, analyze, possible, assess, impact, intended, change, diagnose, deficiencies, causes, failures, identify, modified.</i>
27	<i>Modifiability</i>	<i>Maintain, modified, defects, degrading, existing, quality.</i>
28	<i>Testability</i>	<i>Maintain, tested, test, criterion, established, determine.</i>
29	<i>Adaptability</i>	<i>Portability, adaptability, different, evolving, hardware, software, operational, environments.</i>
30	<i>Installability</i>	<i>Portability, installed, effectiveness, efficiency, successfully, remove.</i>
31	<i>Replaceability</i>	<i>Portability, replaceability, another, product, same, purpose.</i>

2.5.3. Relasi Hipernim

Secara umum hipernim merupakan suatu kata yang memiliki ruang lingkup makna yang mencakup hal - hal umum atau mencakup aspek yang luas. Relasi Hipernim dalam Wordnet menurut Miller (Miller, 1995) memiliki kata lain *supername*, yaitu hubungan transitif antar *synsets*, hubungan semantik ini mengatur makna kata dalam struktur hirarki. Struktur hirarki yang dimaksud adalah struktur hirarki pohon *IS-A*, dimana kata Y merupakan hipernim dari suatu kata X, jika setiap X adalah (*IS-A*) Y (Koirala & Rasheed, 2008). Berikut ilustrasi dari relasi hipernim berdasarkan leksikal Wordnet pada Gambar 2.3.

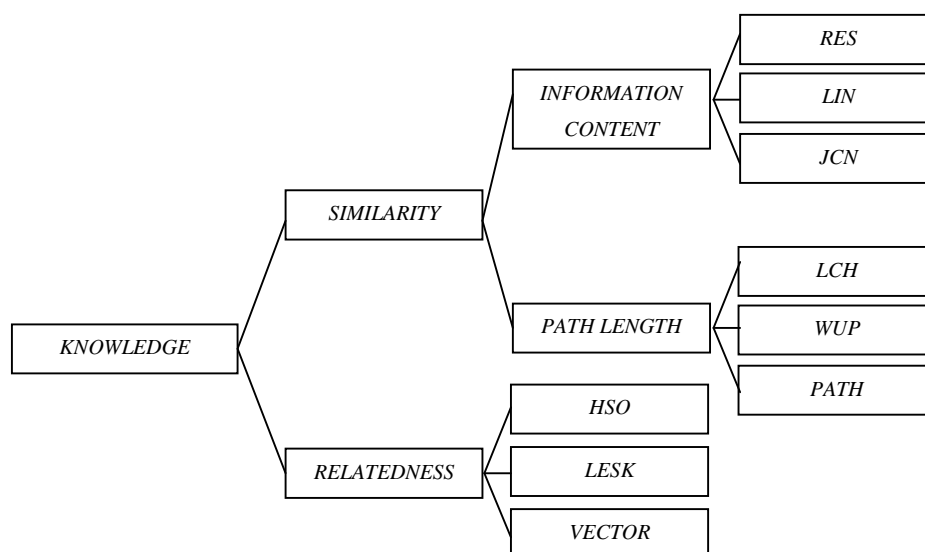


Gambar 2.3 Representasi Relasi Hipernim dalam Wordnet (Chen, Tseng and Liang, 2010)

Pada Gambar 2.3 diatas dapat diketahui bahwa berdasarkan hubungan *IS-A* pada struktur hirarki pohon diatas, kata *marketing* merupakan hipernim untuk kata *sales* dan *retail* pada level 1, sedangkan kata *commerce* yang berada pada level 2 merupakan hipernim untuk kata *sales*, *retail*, *marketing*, dan *trade*.

2.5.4. Pengukuran Keterkaitan Semantik

Pengukuran semantik menurut Gomaa dan Fahmy digolongkan menjadi tiga kelompok yang berbeda yaitu *String-based similarity measure*, *Corpus-based similarity measure*, dan *Knowledge-based similarity measure*. Pengukuran semantik yang berhubungan dengan pemanfaatan informasi yang terdapat dalam wordnet termasuk ke dalam kelompok *Knowledge-based similarity measure*. Kelompok *Knowledge-based similarity measure* terbagi lagi menjadi dua bagian yang berbeda yaitu *semantic similarity measure* dan *semantic relatedness measure*, detail pembagian kelompok dapat dilihat pada Gambar 2.4 (Gomaa & Fahmy, 2013).



Gambar 2.4 Bagan Kelompok Knowledge-based Similarity Measures (Gomaa, 2013)

Berdasarkan Gambar 2.4, pada penelitian ini akan menggunakan metode *Hirst & S-Onge (HSO)* yang berada pada bagian *semantic relatedness measure*. Metode HSO digunakan karena memiliki beberapa kelebihan dibandingkan metode pada bagian *semantic similarity measure*, seperti salah satunya memiliki kemampuan untuk dapat mengukur keterkaitan semantik antara dua kata yang berbeda jenis katanya, misalnya kata berjenis *noun* dengan kata berjenis *verb*.

Korelasi persepsi manusia dengan metode HSO berbasis tes acuan Miller-Charles dan Rubenstein-Goodenough juga menunjukkan bahwa metode HSO memiliki korelasi yang lebih tinggi dibandingkan pada metode - metode yang ada pada bagian *semantic similarity measure* (Wan & Angryk, 2007).

Ide dasar dari metode HSO adalah menentukan keterkaitan semantik antara dua kata yang dibandingkan menggunakan relasi kohesi untuk menghitung jalur yang diperbolehkan antara dua kata tersebut. Metode HSO memiliki tiga tipe relasi kohesi berbeda yang secara langsung menghubungkan keterkaitan semantik antara dua kata sebagai berikut (Choudhari, 2012)(Hirst & St-Onge, 1998):

1. Extra Strong Relation

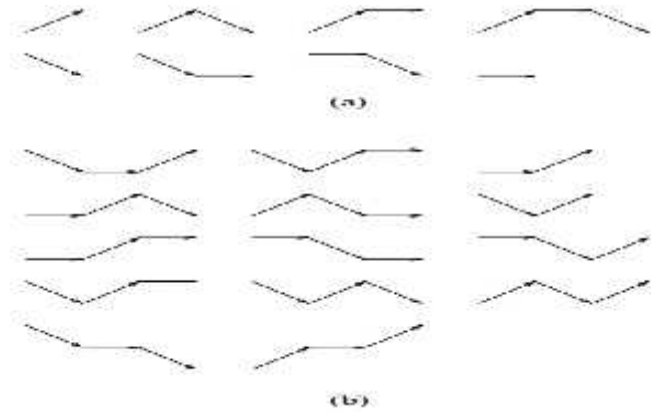
Extra Strong Relation merupakan relasi antara dua kata yang memiliki bobot tertinggi dari semua jenis hubungan lainnya dan menghasilkan keterkaitan yang tinggi. Contoh dari *extra strong relation* adalah misal perbandingan kata yang sama yaitu **man** dan **man** (Hirst & St-Onge, 1998).

2. Strong Relation

Kata dikatakan *Strong Relation* apabila memenuhi tiga kriteria. Kriteria pertama adalah dua buah kata yang memiliki kata induk yang sama atau didapatkan dari induk yang sama. Kriteria kedua adalah relasi yang digunakan dalam konsep kata induk yang memiliki bentuk *horizontal link* (antonim, *similar to*, *see also*, *attribute*). Misalnya kata **man** dan **woman** memiliki hubungan *strong relation* karena keduanya memiliki *horizontal link* dalam bentuk antonim. Sedangkan kriteria ketiga adalah ketika ada link apapun antara kata induk dari setiap kata, jika salah satu kata merupakan kata majemuk atau frase yang mencakup kata lainnya. Misalnya kata **color** dan **water-color**.

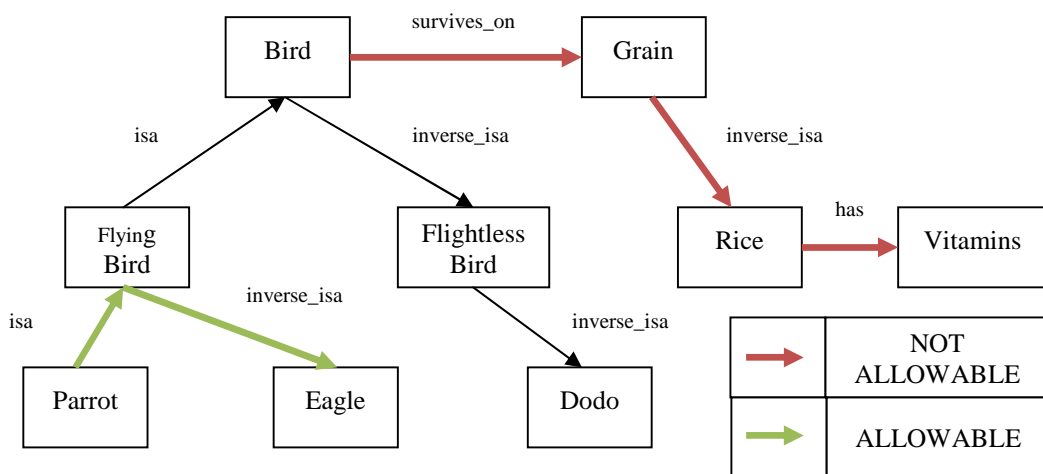
3. Medium Strong Relation

Dalam *medium strong relation*, pengukuran keterkaitan semantik dilakukan dengan mempertimbangkan jalur yang diperbolehkan (*allowable path*) dan jumlah perubahan arah jalur (*number of change of direction*). Jalur yang diperbolehkan merupakan penggabungan jalur dari kata sumber ke suatu kata lainnya yang tidak menyimpang dari arti kata sumber, detail jalur yang diperbolehkan dan tidak diperbolehkan dapat dilihat pada Gambar 2.5.



Gambar 2.5 (a) Pola Jalur Diperbolehkan (b) Pola Jalur yang Tidak Diperbolehkan (Budanitsky and Hirst, 1998)

Pada Gambar 2.6 menunjukkan contoh keterkaitan semantik antara dua kata yang saling terhubung. Keterkaitan semantik tersebut terhubung dengan salah satu jalur yang diperbolehkan dan juga contoh dua kata yang terhubung dengan jalur yang tidak diperbolehkan. Dalam Gambar 2.6 kata *parrot* dan *eagle* menunjukkan hubungan *strong relation* antara satu dengan yang lainnya, dan keduanya terhubung dengan salah satu pola jalur yang diperbolehkan. Pada kata *bird* dan *vitamins* hubungan secara semantik sangat jauh antara satu dengan yang lainnya dan terhubung dengan pola jalur yang tidak diperbolehkan sehingga sangat masuk akal apabila tidak mempertimbangkan hubungan *bird* dengan *vitamins* karena kedua kata ini sangat berbeda.



Gambar 2.6 Contoh Relasi Dengan Jalur yang Diperbolehkan dan Tidak Diperbolehkan (Choudhari, 2012)

Metode HSO memberikan penjelasan tentang kenapa hanya pola jalur yang ditunjukkan pada Gambar 2.5 (a) saja yang diperbolehkan, karena jika jalur multilink antara dua kata menjadi indikasi beberapa pertimbangan kedekatan secara semantik maka semantik untuk setiap relasi harus dipertimbangkan (Hirst & St-Onge, 1998). Seperti yang ditunjukkan pada Gambar 2.6, hubungan keatas (*upward link*) dari sebuah kata menunjukkan arti sebuah generalisasi. Kata *flying bird* merupakan generalisasi untuk kata *parrot* dan *eagle* yang berarti bahwa *flying bird* secara semantik lebih general dibandingkan kata *parrot* dan *eagle*. Demikian pula untuk hubungan ke bawah (*downward link*) yang menunjukkan arti spesialisasi. Pada kata yang terhubung dengan kedua hubungan *upward* dan *downward link* tidak menunjukkan banyak penyimpangan dari makna pada kata aslinya. Sedangkan untuk hubungan yang ditunjukkan oleh *horizontal link* bisa memiliki dua arah yang berbeda tergantung pada makna *horizontal link* antara dua kata tersebut, seperti pada contoh pada Gambar 2.6 yang menunjukkan hubungan antara *bird* dengan *vitamin* melalui dua buah *horizontal link*, yang mengakibatkan penyimpangan jauh secara semantik antara kata *bird* dengan *vitamins*. Berbeda halnya untuk *horizontal link* yang menghubungkan antara kata *man* dan *woman* yang secara semantik memiliki makna antonim antara keduanya. Oleh sebab itu metode HSO memberikan dua aturan untuk memastikan bahwa suatu jalur sesuai dengan hubungan antara kata sumber dan kata sasaran, sebagai berikut :

- a. Aturan 1 : tidak ada *link* yang mendahului *upward link*.

Setelah suatu kata dipersempit ke bawah dengan menggunakan *downward link* maupun *horizontal link*, tidak diperbolehkan untuk menggeneralisasi kata tersebut lagi dengan menggunakan *upward link*.

- b. Aturan 2 : paling banyak satu perubahan arah yang diperbolehkan.

Suatu tindakan merubah arah merupakan suatu langkah besar dalam penentuan keterkaitan semantik antara dua kata, oleh karena itu perubahan arah harus dibatasi. Tetapi ada pengecualian untuk aturan 2 ini yaitu :

Dijijinkan untuk menggunakan *horizontal link* untuk membuat transisi dari atas ke arah bawah. Sebagai contoh kata *heat* dan *hot*, dimana keduanya terhubung dengan *horizontal link* dalam bentuk relasi *attribute*.

Dalam *medium strong relation* untuk mengukur keterkaitan semantik antara dua kata metode HSO menggunakan persamaan sebagai berikut :

$$w_{ht} = C - p \cdot h \cdot l \cdot h - k * m \quad o \quad chan \quad o \quad d \quad (2.28)$$

Dimana C dan k merupakan suatu konstanta.

Penelitian sebelumnya (Ramadhani, Rochimah and Yuhana, 2015) melakukan pengembangan dengan memperbolehkan *path length* sepanjang 6 jalur sehingga nilai konstanta $C = 9$ dan k tetap bernilai 1. Nilai untuk *strong relation* di dapatkan dari $2 * C$, sehingga jika dua kata menunjukkan keterkaitan semantik yang kuat maka akan menerima nilai 18. Untuk dua kata yang memiliki hubungan *medium strong relation* dapat diukur dengan persamaan 2.28. Nilai yang didapatkan dari pengukuran keterkaitan semantik tersebut akan disesuaikan ke dalam konsep fuzzy yaitu jarak nilai hanya dari 0 hingga 1, kemudian nilai tersebut akan dibagi dengan 18. Nilai 18 merupakan nilai tertinggi yang bisa didapatkan ketika dua kata dilakukan pengukuran keterkaitan semantiknya.

2.5.5. Pembobotan TF-IDF

Tahapan pembobotan TF-IDF dilakukan untuk mendapatkan bobot setiap *term* terhadap suatu kalimat kebutuhan. Nilai dari hasil pembobotan ini merepresentasikan pengaruh *term* terhadap isi dari kalimat kebutuhan tersebut. Proses pembobotan TF-IDF terdiri dari tiga proses pencarian *term frequency* (TF) dari setiap *term*, pencarian kemunculan tiap *term* terhadap seluruh kalimat kebutuhan atau disebut *inverse document frequency* (IDF), dan yang terakhir adalah proses perkalian TF dan IDF dengan dilakukan normalisasi agar nilai bobot yang didapatkan berada dalam rentang nilai 0 hingga 1. Berikut penjelasan masing - masing proses beserta contohnya :

1. Term Frequency (TF)

Pencarian *term frequency* baik dari setiap *term* atribut aspek kualitas maupun dari setiap *term* kalimat kebutuhan, dilakukan berdasarkan persamaan :

$$T = (t_i, D) \quad (2.1)$$

Dimana t_i adalah *term* ke-i dan D adalah suatu kalimat kebutuhan yang mengandung *term* ke-i.

2. Inverse Document Frequency (IDF)

Berdasarkan Tabel 2.2, maka dapat dihitung nilai IDF untuk setiap *term* dengan menggunakan persamaan :

$$IDF = \log \frac{N}{d(t_i, D)} + 1 \quad (2.2)$$

Dimana N adalah jumlah total kebutuhan dalam koleksi kalimat kebutuhan dan $d(t_i, D)$ adalah jumlah kebutuhan yang mengandung term t_i .

3. Pembobotan TF-IDF

Pada tahap ini, melakukan pembobotan TF-IDF dengan menggunakan persamaan :

$$\omega_K(t_i) = \frac{t(t_i, K) \times \log \frac{N}{d(t_i, K)} + 1}{\sqrt{\sum_{t_i} (t(t_i, K) \times \log \frac{N}{d(t_i, K)} + 1)^2}} \quad (2.3)$$

2.5.6. Pengukuran Nilai Kemiripan

Setelah dilakukan tahapan pembobotan TF-IDF maka dapat dilakukan pengukuran nilai kemiripan antara kalimat kebutuhan K dengan suatu atribut aspek kualitas dengan persamaan :

$$C(Q, K) = \sum_{i=1}^M \omega_Q(t_i) \times \omega_K(t_i) \quad (2.4)$$

Dimana $\omega_Q(t_i)$ merupakan bobot dari term t_i yang berasal atribut aspek kualitas, sedangkan $\omega_K(t_i)$ merupakan bobot yang didapatkan dari term t_i yang berasal suatu kalimat kebutuhan K . Dengan menggunakan penentuan nilai *threshold* tertentu maka akan dapat dihasilkan kalimat - kalimat kebutuhan K yang terlabeli ke dalam satu atau lebih atribut aspek kualitas yang kemudian dapat digunakan sebagai data latih dalam algoritma FSKNN.

2.6 Algoritma Fuzzy Similarity based K-Nearest Neighbor (FSKNN)

Algoritma FSKNN menggunakan pendekatan *fuzzy similarity measure* (FS) untuk menghitung similaritas antara *term*, dokumen dan aspek kualitas dalam data pelatihan digunakan untuk mendapatkan suatu pola pelatihan. Pola pelatihan tersebut dikelompokkan ke dalam beberapa kluster berdasarkan nilai *threshold* tertentu. Hasil kluster tersebut digunakan untuk mendapatkan *search set* pada setiap data latih. *Search set* inilah yang digunakan untuk menemukan *K-Nearest Neighbor* (KNN) dari suatu dataset baik dalam proses pelatihan maupun proses klasifikasi. Suatu dokumen uji akan dilabeli atau dimasukkan ke dalam aspek

tertentu berdasarkan pada *K-Nearest Neighbor*-nya dengan menggunakan estimasi *maximum a posteriori* (MAP).

Algoritma *fuzzy* yang digunakan untuk mengukur similaritas dalam pendekatan FSKNN ini berdasarkan pada formula khusus berbasis *fuzzy information retrieval* yang diajukan oleh Widyantoro dan Yen, dimana formula ini terbukti memiliki performa yang baik (Widyantoro and Yen, 2000). Berikut detail proses algoritma FSKNN (Jiang, Tsai and Lee, 2012):

2.6.1 Struktur Data Pelatihan

Dalam klasifikasi dokumen *multi-label* diberikan suatu rangkaian (D,T,C) dimana :

$$D = \{(d_1, y_1), (d_2, y_2), \dots, (d_l, y_l)\} \quad (2.5)$$

D merupakan satu set l pola pelatihan, $T = \{t_1, t_2, \dots, t_m\}$ merupakan satu set *term*, dan $C = \{c_1, c_2, \dots, c_p\}$ merupakan satu set p aspek kualitas. T memuat semua *term* kunci yang telah dipilih untuk dokumen d_1, d_2, \dots, d_l . Pola pelatihan ke i terdiri dari dua komponen yaitu dokumen d_i dan label-vektor y_i . Sebuah dokumen d_i , $1 \leq i \leq l$, direpresentasikan sebagai sebuah vektor $\langle w_{1i}, w_{2i}, \dots, w_{mi} \rangle$ dimana w_j adalah frekuensi *term* dari *term* t_j , yaitu jumlah kemunculan dari t_j dalam dokumen d_i . Setiap dokumen dapat dikelompokkan pada satu atau lebih dari satu aspek kualitas di C . Untuk setiap d_i , $1 \leq i \leq l$, y_i adalah sebuah vektor dengan p komponen, yaitu $y_i = \langle y_{1i}, y_{2i}, \dots, y_{pi} \rangle$ dimana :

$$y_j = \begin{cases} 1, & \text{jika } d_i \text{ terdapat } t_j \\ 0, & \text{jika } d_i \text{ tidak terdapat } t_j \end{cases} \quad (2.6)$$

Untuk $1 \leq j \leq p$. Dalam klasifikasi *single-label*, hanya satu komponen dalam y_i yang bernilai 1, tetapi dalam klasifikasi *multi-label*, beberapa komponen mungkin bernilai 1 dalam y_i . Sebagai contoh $y_i = \langle 0, 1, 0, 1 \rangle$ mengindikasikan bahwa d_i termasuk dalam aspek kualitas c_2 dan c_4 .

2.6.2 Proses Pelatihan

Proses Pelatihan dalam FSKNN terdiri dari dua bagian yaitu bagian pertama adalah pengelompokan pola pelatihan ke dalam kluster, dan bagian kedua adalah perhitungan nilai *prior probability* dan nilai *likelihood*.

1. Pengelompokan Pola Pelatihan ke Dalam Kluster

Pengelompokan dokumen pelatihan d_1, d_2, \dots, d_t ke dalam p kluster berbasis pada *fuzzy similarity measure*. Diberikan $d(t_i, c_j)$ dan $d(t_i, c_j)$ yang merupakan distribusi *term* t_i pada aspek kualitas c_j , yang didefinisikan sebagai berikut:

$$d(t_i, c_j) = \frac{\sum_{i=1}^t w_{ti} y_{ji}}{\sum_{i=1}^t w_{ti}} \quad (2.7)$$

$$d(t_i, c_j) = \frac{\sum_{i=1}^t s_i(w_{ti}) y_{ji}}{\sum_{i=1}^t y_{ti}} \quad (2.8)$$

Untuk $1 \leq i \leq m$ dan $1 \leq j \leq p$, dimana :

$$s_i(x) = \begin{cases} 1, & \text{jiika } x > 0 \\ 0, & \text{jiika } x = 0 \end{cases} \quad (2.9)$$

Sehingga tentunya akan didapatkan $0 \leq d(t_i, c_j) \leq 1$.

Tahapan selanjutnya mendefinisikan derajat keanggotaan t_i terhadap aspek kualitas c_j sebagai berikut :

$$\mu_R(t_i, c_j) = \frac{d(t_i, c_j)}{\sum_{1 \leq u \leq m, 1 \leq v \leq p} d(t_u, c_v)} \times \frac{d(t_i, c_j)}{\sum_{1 \leq u \leq m, 1 \leq v \leq p} d(t_u, c_v)} \quad (2.10)$$

Untuk $1 \leq i \leq m$ dan $1 \leq j \leq p$.

Tahapan selanjutnya adalah menentukan *fuzzy similarity* dari setiap dokumen d , $d = w_1, w_2, \dots, w_m$, terhadap aspek kualitas c_j sebagai berikut :

$$st(d, c_j) = \frac{\sum_{i=1}^m \mu_R(t_i, c_j) \otimes \mu_{t_i}(t_i)}{\sum_{i=1}^m \mu_R(t_i, c_j) \oplus \mu_{t_i}(t_i)} \quad (2.11)$$

Dimana \otimes dan \oplus merupakan *fuzzy t-norm* dan *t-conorm* yang secara berurutan didefinisikan sebagai berikut :

$$x \otimes y = x \times y \quad (2.12)$$

$$x \oplus y = x + y - x \times y \quad (2.13)$$

Dan

$$\mu_d(t_i) = \frac{w_i}{\sum_{1 \leq v \leq m} w_v} \quad (2.14)$$

$\mu_d(t_i)$ merupakan derajat keanggotaan dari *term* t_i terhadap d . Tahapan terakhir adalah mendefinisikan derajat keanggotaan dari dokumen d terhadap aspek kualitas c_j sebagai berikut :

$$\mu_{c_j}(d) = \frac{st(d, c_j)}{\sum_{1 \leq v \leq p} st(d, c_v)} \quad (2.15)$$

Untuk $1 \leq j \leq p$.

Untuk setiap dokumen pelatihan $d_i, 1 \leq i \leq l$, akan dilakukan perhitungan $\mu_{c_j}(d_i), 1 \leq j \leq p$, dengan menggunakan persamaan 2.11. Untuk mendefinisikan p kluster, S_1, S_2, \dots, S_p , adalah sebagai berikut :

$$S_v = \{d_u | \mu_{c_v}(d_u) \geq \alpha, 1 \leq u \leq l\} \quad (2.16)$$

Untuk $1 \leq v \leq p$, dimana α adalah threshold yang didefinisikan oleh user untuk digunakan dalam proses pelatihan. Pertimbangan dilakukannya pengelompokan pola pelatihan yang sudah terlabeli ke dalam kluster karena ada kemungkinan bahwa dokumen pelatihan belum secara sempurna diklasifikasikan ke dalam aspek kualitas yang benar. Selain itu *error* mungkin juga terjadi selama proses pencatatan, pemunculan *term*, dan pelabelan data latih, sehingga dengan menggunakan pengelompokan dapat menghilangkan dan memperbaiki kesalahan tersebut (Jiang, *et al.*, 2012). Untuk setiap $d_i, 1 \leq i \leq l$, didefinisikan *search set* G_i untuk setiap $S_v \in G_i$ jika dan hanya jika $d_i \in S_v, 1 \leq v \leq p$. Proses berikutnya akan dijelaskan pada *pseudo-code* yang ditunjukkan pada Gambar 2.7, dengan catatan pada mulanya $S_v = \emptyset, 1 \leq v \leq p$, dan $G_u = \emptyset, 1 \leq u \leq l$.

```

for untuk setiap dokumen pelatihan  $d_i, 1 \leq i \leq l$ 
  for untuk setiap kategori  $c_j, 1 \leq j \leq p$ 
    if  $(\mu_{c_j}(d_i) \geq \alpha)$ 
      then  $S_{j_c} = S_{j_c} \cup \{d_i\}$ 
  for untuk setiap dokumen pelatihan  $d_u, 1 \leq u \leq l$ 
    for untuk setiap kluster  $S_v, 1 \leq v \leq p$ 
      if  $(d_u \in S_v)$ 
        then  $G_u = G_u \cup S_v$ 

```

Gambar 2.7 Pseudo-code proses pengelompokan dan proses pendefinisian search set G_i (Jiang, 2012)

Keluaran yang berupa *search set* G_1, G_2, \dots, G_l selanjutnya akan digunakan untuk menentukan data tetangga terdekat yang dapat membantu melakukan perhitungan nilai *prior probability* dan nilai *likelihood* pada tahapan berikutnya.

2. Perhitungan Nilai Prior Probability dan Nilai Likelihood

Diberikan $P(H_j)$ yang merupakan *prior probability* yang harus diketahui nilainya terlebih dahulu sebelum melanjutkan ke dalam setiap observasi, sedangkan $P(E|H_j)$ merupakan suatu kelas *likelihood* dan merupakan sebuah *conditional probability* bahwa H_j memiliki keterkaitan dengan observasi E . Perhitungan probabilitas ini dilakukan dari pola pelatihan yang didapatkan sebelumnya, sebagai berikut :

$$P(H_j = 1) = \frac{s + \sum_{i=1}^l y_j}{2s + l} \quad (2.17)$$

$$P(H_j = 0) = 1 - P(H_j = 1) \quad (2.18)$$

Dimana s merupakan nilai *smoothing constant*, yang biasanya bernilai real positif kecil. Tahapan berikutnya melakukan perhitungan kelas *likelihood* $P(E|H_j)$. E dapat bernilai 0,1, ... , atau k . Untuk setiap dokumen pelatihan \mathbf{d}_i , $1 \leq i \leq l$, dimana $N^i = \{d_{v1}, d_{v2}, \dots, d_v\}$ merupakan *k-nearest neighbor* yang diambil dari *search set* G_i dan $\mathbf{n}^i = \{n_1^i, n_2^i, \dots, n_p^i\}$, yang merupakan vektor jumlah label yang didefinisikan :

$$n_j^i = \sum_{v=v_1}^{v_k} y_j \quad (2.19)$$

Untuk $1 \leq j \leq p$. Tahap selanjutnya didefinisikan :

$$Z(\mathbf{e}, j) = \sum_{i=1}^l y_j \delta_{\mathbf{e}}(j) \quad (2.20)$$

$$\bar{Z}(\mathbf{e}, j) = \sum_{i=1}^l \bar{y}_j \delta_{\mathbf{e}}(j) \quad (2.21)$$

Dimana $\bar{y}_j = 1 - y_j$ dan

$$\delta_{\mathbf{e}}(j) = \begin{cases} 1, & \text{jika } \mathbf{e} = \mathbf{n}_j^i \\ 0, & \text{jika } \mathbf{e} \neq \mathbf{n}_j^i \end{cases} \quad (2.22)$$

Kemudian didefinisikan *likelihoods*, sebagai berikut :

$$P(E = \mathbf{e} | H_j = 1) = \frac{s + Z(\mathbf{e}, j)}{(k+1)s + \sum_{v=0}^k Z(\mathbf{v}, j)} \quad (2.23)$$

$$P(E = \mathbf{e} | H_j = 0) = \frac{s + \bar{Z}(\mathbf{e}, j)}{(k+1)s + \sum_{v=0}^k \bar{Z}(\mathbf{v}, j)} \quad (2.24)$$

Untuk setiap $e = 0, 1, \dots, k$ dan $j = 1, 2, \dots, p$, karena ukuran dari $G_{v_i}^t$, $1 \leq i \leq l$ selalu lebih kecil daripada jumlah pola pelatihan awal, menghitung *likelihoods* dapat dilakukan secara efisien.

2.6.3 Proses Testing

Proses *testing* dalam algoritma FSKNN dilakukan dengan menggunakan estimasi *maximum a posteriori* (MAP). Dimisalkan $N^t = \{d_{v1}, d_{v2}, \dots, d_v\}$ merupakan satu set *k-nearest neighbors* untuk dokumen testing d^t , dan $n^t = n_1^t, n_2^t, \dots, n_p^t$ merupakan vektor jumlah label untuk d^t (persamaan 2.15), untuk menentukan aspek kualitas mana yang memiliki hubungan dengan d^t yaitu dengan cara menghitung vektor label $y^t = y_1^t, y_2^t, \dots, y_p^t$ dari dokumen d^t dengan menggunakan estimasi *maximum a posteriori* (MAP) sebagai berikut :

$$y_j^t \begin{cases} 1, & \text{ji} \quad P(H_j = 1|E = n_j^t) > P(H_j = 0|E = n_j^t) \\ 0, & \text{ji} \quad P(H_j = 0|E = n_j^t) > P(H_j = 1|E = n_j^t) \\ R[0,1], & \text{o he} \end{cases} \quad (2.25)$$

Untuk $1 \leq j \leq p$, dimana H_j adalah variabel acak untuk mengetahui masuk ke dalam aspek kualitas c_j atau tidak ($H_j = 1$ untuk iya, dan $H_j = 0$ untuk tidak), E merupakan variabel untuk jumlah dokumen dalam N^t yang berhubungan dengan aspek kualitas c_j , dan $R[0,1]$ mengindikasikan 0 atau 1 dipilih secara acak. Dengan menggunakan *Bayes Rule* didapatkan :

$$P(H_j = b|E = n_j^t) = \frac{P(H_j=b)P(E=n_j^t|H_j=b)}{P(E=n_j^t)} \quad (2.26)$$

Untuk $b = 0, 1$. Oleh karena itu persamaan 2.25 akan berubah menjadi :

$$y_j^t \begin{cases} 1, & \text{ji} \quad P(H_j = 1)P(E = n_j^t|H_j = 1) > P(H_j = 0)P(E = n_j^t|H_j = 0) \\ 0, & \text{ji} \quad P(H_j = 0)P(E = n_j^t|H_j = 0) > P(H_j = 1)P(E = n_j^t|H_j = 1) \\ R[0,1], & \text{o he} \end{cases} \quad (2.27)$$

Untuk $1 \leq j \leq p$. Tentunya untuk menghitung y_j^t harus ditemukan N^t , dan menghitung $P(H_j)$ (Persamaan 2.13 dan 2.14) dan juga $P(E|H_j)$ (Persamaan 2.19 dan 2.20). Perhitungan yang tidak tergantung pada d^t dapat dilakukan pada proses pelatihan dan sisanya dilakukan ketika proses *testing*. Berikut tahapan selama proses *testing* :

1. Hitung $\mu_{c_j}(d^t)$, $1 \leq j \leq p$ dengan Persamaan 2.11 menggunakan informasi atau hasil perhitungan yang dilakukan dengan Persamaan 2.6, 2.7 dan 2.10.
2. Periksa jika $\mu_{c_j}(d^t) \geq \alpha$ untuk $1 \leq j \leq p$, maka akan didapatkan *search set* untuk d^t yaitu $G^t = S_{j_1} \ S_{j_2} \ \dots \ S_{j_n}$.
3. Temukan set N^t dari *k-nearest neighbors* terhadap d^t dari G^t , dan dapatkan vektor jumlah label n^t .
4. Hitung y_j^t , $1 \leq j \leq p$ menggunakan Persamaan 2.23 dengan menggunakan informasi yang telah dihitung dalam proses pelatihan dengan Persamaan 2.13, 2.14, 2.19 dan 2.20.
5. Jika y_j^t yang didapatkan bernilai 1, maka d^t termasuk ke dalam aspek kualitas c_j , jika sebaliknya maka d^t tidak termasuk ke dalam c_j .

Algoritma FSKNN yang digunakan pada penelitian sebelumnya (Jiang, Tsai and Lee, 2012)(Ramadhani, Rochimah and Yuhana, 2015) diidentifikasi memiliki dua keterbatasan yaitu (1) *Term* kunci yang digunakan pada proses pelatihan hanya terbatas pada *term* yang muncul dalam dokumen pelatihan saja. (2) Penambahan variabel baru berupa faktor semantik ke dalam proses algoritma FSKNN membuat peningkatan kinerja klasifikasi tidak signifikan. Oleh karena itu, pada penelitian ini akan menambahkan pengukuran semantik antara *term-term* relevan setiap aspek kualitas dengan *term-term* pada dokumen yang akan dilabeli. Penambahan pengukuran semantik ini akan meningkatkan jumlah *term* frekuensi pada data latih yang diperoleh pada proses sebelumnya. Peningkatan jumlah *term* frekuensi tersebut diharapkan mampu meningkatkan jumlah tetangga terdekat. Peningkatan jumlah tetangga terdekat akan mempengaruhi vektor yang dihasilkan. Perubahan vektor tersebut diharapkan mampu meningkatkan kinerja klasifikasi.

2.7 Wordnet

Wordnet adalah sistem referensi leksikal bahasa Inggris yang dikembangkan oleh para ilmuwan di Universitas Princeton pada tahun 1990 (Christiane, 1998)(Miller, 1995). Wordnet dikembangkan untuk menyediakan kamus online yang dibangun tidak hanya dalam urutan abjad tetapi dalam cara yang lebih konseptual yang menunjukkan hubungan semantik.

Wordnet telah meningkatkan jumlah kata dan menambahkan hubungan kata dalam 17 jenis berbeda diantaranya sinonim, antonim, hyponyms, dan mironim. Sekarang ini, wordnet menampung lebih dari 150.000 *term*. *Term* tersebut diorganisasikan didalam 115.000 arti kata atau set sinonim yang disebut *synsets*, dan setiap *synsets* dapat terhubung dengan *synsets* yang lainnya. Menggunakan wordnet dapat untuk mengukur keterkaitan *term* dari kata - kata, sehingga dapat dihitung nilai similaritas dan keterkaitan semantik antar *term* yang ditemukan dalam wordnet. Wordnet memiliki 41 kategori leksikal sebagai *nouns*, *verbs*, *adjectives*, dan *adverb*. Wordnet juga dapat digunakan untuk mendapatkan hubungan sinonim, hipernim, hiponim, meronim dan informasi topik yang didapatkan dari suatu kata. (Chua & Kulathuramaiyer, 2004)(Çelik & Güngör, 2013)(Patil & Atique, 2013).

Dalam penelitian ini, wordnet digunakan sebagai referensi leksikal untuk melakukan pengembangan *term* atau memperkaya *term* dari data latih dengan menggunakan relasi hipernim yang didapatkan dari wordnet tersebut.

2.8 Gwet's AC1

Gwet's AC1 adalah salah satu metode yang digunakan untuk mengukur indeks kesepakatan antara dua pengamat. Gwet's AC1 menunjukkan pendekatan yang lebih dapat diandalkan dibandingkan dengan Cohen Kappa(Gwet, 2013). Penghitungan nilai AC1, hasil dari observasi pengamat dimasukkan dalam matrik 2x2 seperti yang ditunjukkan pada Tabel 2.3 Penulisan Hasil Pengamatan di bawah ini:

Tabel 2.3 Tabel Penulisan Hasil Pengamatan

Pengamat 1	Pengamat 2		Total
	Ya	Tidak	
Ya	a	b	B1=a+b
Tidak	c	d	B2=c+d
Total	A1=a+c	A2=b+d	N

Pada Tabel 2.3 terdapat dua pengamat yang mengklasifikasikan N subjek ke dua kemungkinan kategori. Dua kategori dilabelkan sebagai “ya” dan “tidak”. Variabel a diklasifikasikan oleh dua pengamat sebagai ya, Variabel b diklasifikasikan oleh pengamat 2 tidak, Variabel d diklasifikasikan oleh dua

pengamat sebagai tidak, dan Variabel c diklasifikasikan oleh pengamat 1 sebagai tidak. Variabel B1 menunjukkan jumlah dari variabel a dan variabel b, Variabel B2 menunjukkan jumlah dari variabel c dan variabel c, Variabel A1 menunjukkan jumlah dari variabel a dan variabel c, dan Variabel A2 menunjukkan jumlah dari variabel b dan variabel d.

Perhitungan AC1-statistic :

$$\text{Kesepakatan yang terobservasi, } P = \frac{a+d}{N} \quad (2.28)$$

Dimana variabel a adalah banyaknya data yang dikelompokkan ke dalam kategori “Ya” oleh kedua pengamat sedangkan variabel d adalah banyaknya data yang dikelompokkan ke dalam kategori “tidak” oleh kedua pengamat.

Sedangkan untuk menghitung probabilitas acak antara kedua pengamat yaitu :

- Pengamat 1 mengatakan Ya sebanyak 2 dan Tidak sebanyak 0.
- Pengamat 2 mengatakan Ya sebanyak 2 dan Tidak sebanyak 0.

Diketahui bahwa kedua pengamat mengatakan Ya sebanyak 100 %. Jadi, probabilitas yang diharapkan dari kedua pengamat mengatakan Ya secara acak ditunjukkan pada persamaan 2.29, 2.30 dan untuk probabilitas keseluruhan ditunjukkan pada persamaan 2.31. Sedangkan untuk menghitung nilai indeks Kappa menggunakan Cohen Kappa statistik dengan persamaan 2.32.

$$P_{ya} = \frac{a+b}{a+b+c+d} \quad \frac{a+c}{a+b+c+d} \quad (2.29)$$

$$P_{tidak} = \frac{c+d}{a+b+c+d} \quad \frac{b+d}{a+b+c+d} \quad (2.30)$$

$$P_e = P_{ya} + P_{tidak} \quad (2.31)$$

$$k = \frac{P - P_e}{1 - P_e} \quad (2.32)$$

$$\text{Probabilitas } \textit{chance-agreement}, e() = 2P_1(1 - P_1) \quad (2.33)$$

$$\text{AC1-statistic, } A_1 = \frac{P - e()}{1 - e()} \quad (2.34)$$

Tabel 2.4 berikut merupakan interpretasi nilai dari Gwet AC1 yang mengukur indeks kesepakatan antar dua pengamat.

Tabel 2.4 Interpretasi Nilai Gwet AC1

Index Kappa	Proporsi Kesepakatan
< 0	Rendah
0.01 - 0.20	Sedikit
0.21 - 0.40	Cukup
0.41 - 0.60	Sedang
0.61 - 0.80	Substansial
0.81 - 1	Hampir sempurna

Berikut adalah contoh perhitungan *ACI-static*:

Terdapat dua orang pengamat yang dimintai pendapatnya terhadap satu kalimat kebutuhan berdasarkan satu aspek kualitas ISO/IEC 25010:2011, apakah kalimat kebutuhan tersebut masuk ke dalam aspek kualitas *Functional Completeness* atau tidak. Hasil pengamatan terlihat seperti Tabel 2.5 di bawah ini:

Tabel 2.5 Data pengamatan Contoh Kasus Kappa

Pengamat 1	Pengamat 2		Total
	Ya	Tidak	
Ya	2	0	2
Tidak	0	0	0
Total	2	0	2

$$P_o = \frac{a+d}{N} = \frac{2+0}{2} = 1$$

Jika dihitung menggunakan indeks Cohen Kappa statistik maka didapat nilai indeks Kappa sesuai persamaan 2.29, 2.30, 2.31, dan 2.32 sebagai berikut :

$$P = \left(\frac{2}{2}\right) \times \left(\frac{2}{2}\right) = 1$$

$$P = \left(\frac{0}{2}\right) \times \left(\frac{0}{2}\right) = 0$$

$$P = 1 + 0 = 1$$

$$Kappa = (1 - 1) / (1-1) = 0$$

Nilai indeks Cohen Kappa yang dihasilkan pada data pengamatan Tabel 2.5 sangatlah rendah yaitu 0, hal ini dikarenakan pengamatan kedua pakar hanya berdasarkan satu aspek kualitas saja. Hasilnya akan tetap bernilai 0 meskipun kedua pengamat sudah menyatakan persetujuannya dengan memilih kategori Ya. Oleh karena itu, penelitian ini menggunakan Gwet's AC1. Gwet's AC1 terbukti

mampu menghasilkan nilai yang benar sesuai dengan kesepakatan kedua pengamat. Berikut perhitungan dari Gwet's AC1 sesuai persamaan 2.33, dan 2.34 :

$$(\) = 2 \left(\frac{2+2}{2 \times 2} \right) \left(1 - \frac{2+2}{2 \times 2} \right) = 2 * 0 = 0$$

$$AC1 = (1 - 0) / (1 - 0) = 1$$

Berdasarkan nilai AC1, maka dapat disimpulkan bahwa kalimat kebutuhan masuk ke dalam aspek kualitas *Functional Completeness*.

2.9 Evaluasi

Terdapat banyak macam-macam metode evaluasi yang dapat digunakan untuk mengukur kehandalan dari metode yang diusulkan. Proposal ini mengusulkan metode pengukuran dengan menggunakan *accuracy*, *precision*, *recall*, dan F1. Adapun proses perhitungan dari *accuracy*, *precision*, dan *recall* ditentukan dari prediksi informasi terhadap nilai sebenarnya yang direpresentasikan dengan True Positive (TP), True Negative (TN), False Positive (FP), dan False Negative (FN).

- *Accuracy*

$$A = \frac{T + I}{T + F + P + I} \quad (2.29)$$

- *Precision* adalah tingkat ketepatan antara informasi yang diminta oleh pengguna dengan jawaban yang diberikan oleh sistem.

$$precision = \frac{T}{T + F} \quad (2.29)$$

- *Recall* adalah tingkat keberhasilan sistem dalam menemukan kembali sebuah informasi.

$$recall = \frac{T}{T + F} \quad (2.30)$$

- F1 adalah rata-rata harmonik dari *precision* dan *recall*.

$$F1-Score = 2 \times \frac{P \times R}{P + R} \quad (2.31)$$

2.10 Penelitian Sebelumnya

Pada perbandingan penelitian ini, penulis akan membandingkan metode-metode yang berada pada ruang lingkup ekstraksi aspek kualitas perangkat lunak berbasis algoritma klasifikasi. Secara garis besar, metode ekstraksi bisa dilakukan menggunakan dua pendekatan yaitu pendekatan pembelajaran semi-diawasi (*semi-supervised learning*) dan diawasi (*supervised learning*).

(Casamayor, Godoy and Campo, 2010) Casamayor dkk (2010) pada penelitiannya yang berjudul '*Identification of non-functional requirement in textual specification a semi supervised learning approach*' mengusulkan sebuah cara untuk menyelesaikan permasalahan identifikasi aspek kualitas kebutuhan non-fungsional dengan pendekatan pembelajaran semi-diawasi (*semi-supervised learning*) berbasis algoritma Naive Bayes. Pendekatan ini memiliki kelemahan yaitu proses klasifikasi terbatas pada satu aspek kualitas (*single-label*).

Beberapa penelitian lainnya yang mengusulkan sebuah cara untuk menyelesaikan permasalahan identifikasi aspek kualitas kebutuhan non-fungsional dengan pendekatan pembelajaran diawasi (*supervised learning*) dengan menggunakan algoritma klasifikasi model probabilistik adalah penelitian yang dilakukan oleh Huang dkk (2006) (Cleland-Huang *et al.*, 2006). Pada penelitiannya yang berjudul '*The detection and classification of non-functional requirement with application to early aspect*' mengusulkan proses identifikasi aspek kualitas kebutuhan non-functional dengan menggunakan kata kunci yang dihasilkan dari proses penambangan term indikator berbobot. *Term indicator* tersebut digunakan untuk mengklasifikasi setiap tipe non-fungsional dari kebutuhan yang tidak terklasifikasi. Pendekatan ini diidentifikasi memiliki dua kelemahan. Kelemahan pertama adalah kesulitan dalam menemukan katalog yang dapat diterima dan terstandarisasi pada tipe non-fungsional lain, sehingga tidak dievaluasi lebih lanjut. Kelemahan kedua adalah penentuan label menggunakan model yang tidak terstandarisasi.

Suharso dkk (2012) pada penelitiannya yang berjudul '*Sistem penambangan term indikator pada pelatihan kebutuhan non-fungsional berbasis ISO/IEC 9126*' mengusulkan kerangka kerja untuk melakukan otomatisasi pelabelan data latih. Penelitian ini memanfaatkan pengukuran similaritas dokumen dengan menggunakan *cosine measure* untuk melakukan proses pelabelan data latih (Suharso and Rochimah, 2012). Pendekatan ini memiliki kelemahan yaitu belum memperhatikan hubungan keterkaitan semantik.

Ramadhani dkk (2015) pada penelitiannya yang berjudul '*Classification of Non-Functional Requirements Using Semantic-FSKNN Based ISO/IEC 9126*' mengusulkan metode semantik-FSKNN untuk melakukan identifikasi aspek

kualitas kebutuhan non-fungsional. Penelitian ini terdiri dari tiga proses, yaitu (1) Pelabelan otomatis data pelatihan, (2) Pengukuran keterkaitan semantik, dan (3) Klasifikasi aspek kualitas kebutuhan non-fungsional (Ramadhani, Rochimah and Yuhana, 2015).

Proses pelabelan otomatis data pelatihan dilakukan dengan mencari bobot TF-IDF dengan pencarian tingkat kemiripan menggunakan *cosine measure* (Suharso and Rochimah, 2012). Proses pengukuran semantik dilakukan dengan menggunakan metode HSO (Hirst and St-onge, 1995) untuk mendapatkan hubungan semantik antar *term-term* relevan setiap aspek kualitas dengan setiap aspek kualitas. Sedangkan untuk proses klasifikasi, menggunakan metode FSKNN yang telah diperkenalkan oleh (Jiang, Tsai and Lee, 2012). Pada proses ini dilakukan penambahan variabel baru dalam fase pelatihan sehingga nilai keterkaitan semantik yang diperoleh dalam proses sebelumnya dapat dipertimbangkan selama proses pelatihan. Kelas yang digunakan untuk proses klasifikasi berdasarkan Aspek kualitas ISO/IEC 9126:2001.

Pendekatan ini diidentifikasi memiliki tiga kelemahan. Kelemahan pertama adalah aspek kualitas yang digunakan sebagai acuan masih menggunakan standart lama yaitu ISO/IEC 9126:2001. Permasalahan kedua adalah pengembangan *term* menggunakan 2 level hipernim dapat mengurangi kinerja akurasi. Berdasarkan hasil penelitian yang dilakukan oleh Ramadhani dkk (2015) (Ramadhani, Rochimah and Yuhana, 2015) permasalahan tersebut terjadi karena penggunaan hipernim yang terlalu banyak dapat menghilangkan makna asli dari *term* yang dikembangkan. Menurut Rozi dkk (2015) (Rozi, Fatichah and Purwitasari, 2015) penggunaan hipernim mampu meningkatkan akurasi. Namun, penggunaan hipernim dalam beberapa dataset hanya akan menambah jumlah term yang akan dianggap sebagai noise sehingga mengurangi akurasi.

Permasalahan ketiga adalah penambahan variabel baru ke dalam fase pelatihan pada metode FSKNN membuat kinerja klasifikasi menurun. Hal ini disebabkan karena data yang dibentuk untuk pola pelatihan tersaring lebih ketat (Ramadhani, Rochimah and Yuhana, 2015).

Tabel 2.6 Perbandingan Penelitian Berbasis Klasifikasi

Item	Penelitian				
Perbandingan	Casamayor 2010	Huang 2006	Suharso 2012	Ramadhani 2015	Pendekatan diusulkan
Metode	Naive Bayes + EM	Probabilistik	Cosine Measure	Semantic-FSKNN	FSKNN
Tipe Klasifikasi	<i>Single-label</i>	<i>Multi-label</i>	<i>Multi-label</i>	<i>Multi-label</i>	<i>Multi-label</i>
Wordnet	Tidak ada	Tidak ada	Ada	Ada	Ada
Keterkaitan Semantik	Tidak ada	Tidak ada	Tidak ada	Ada (HSO)	Ada (HSO)
Tipe Non-Fungsional	11 NFRs	9 NFRs	21 NFRs	21 NFRs	31 NFRs
Kelebihan	Metode yang digunakan bisa mengurangi jumlah data latih dalam jumlah besar.	Klasifikasi multi-tabel, dan term indikator yang diperoleh bisa meningkatkan akurasi.	Menghasilkan data pelatihan secara otomatis, pengembangan term menggunakan sinonim, dan menggunakan ISO/IEC 9126:2001	Pengembangan term gabungan 2 level hipernim dan 9 sinonim pada masing2x hipernim, dan mempertimbangkan pengukuran faktor semantik antara term relevan dengan setiap aspek kualitas.	Menggunakan ISO/IEC 25010 : 2011, pengembangan term gabungan 1 level hipernim dan 20 sinonim, dan mempertimbangkan keterkaitan semantik antara term relevan dengan term pada dokumen yang akan dilabeli.
Kekurangan	Klasifikasi single label, dan Aspek kualitas tidak standar.	Data latih diperoleh secara manual, dan aspek kualitas yang digunakan tidak standar.	Belum mempertimbangkan faktor hubungan semantik antara term relevan dengan setiap aspek kualitas.	Aspek kualitas ISO/IEC 9126 : 2001 merupakan Aspek kualitas lama. Belum mempertimbangkan faktor hubungan semantik antara term relevan dengan term pada dokumen yang akan dilabeli.	

2.11 Kesimpulan Hasil Studi Komparasi

Tabel 2.6 menunjukkan perbandingan keempat pendekatan klasifikasi diatas dengan pendekatan yang diusulkan dalam penelitian ini. Pendekatan yang diusulkan memiliki tiga kelebihan. Kelebihan pertama adalah aspek kualitas yang digunakan sebagai acuan untuk melakukan pelabelan merupakan standar kualitas ISO/IEC 25010:2011 menghasilkan data latih berdasarkan ISO/IEC 25010:2011. ISO/IEC 25010:2011 merupakan peningkatan dari ISO/IEC 9126:2001 (Saptarini, Rochimah and Yuhana, 2016). Kelebihan kedua adalah pengembangan term yang dilakukan hanya menggunakan 1 level hipernim saja. Pembatasan ini dilakukan karena terlalu banyak penggunaan hipernim dikhawatirkan akan menghilangkan makna asli dari *term* yang dikembangkan.

Kelebihan ketiga adalah menambahkan pengukuran keterkaitan semantik antara *term-term* relevan setiap aspek kualitas dengan *term-term* pada dokumen yang akan dilabeli. Pengukuran semantik menggunakan metode HSO yang terinspirasi dari paper(Budanitsky and Hirst, 1998).

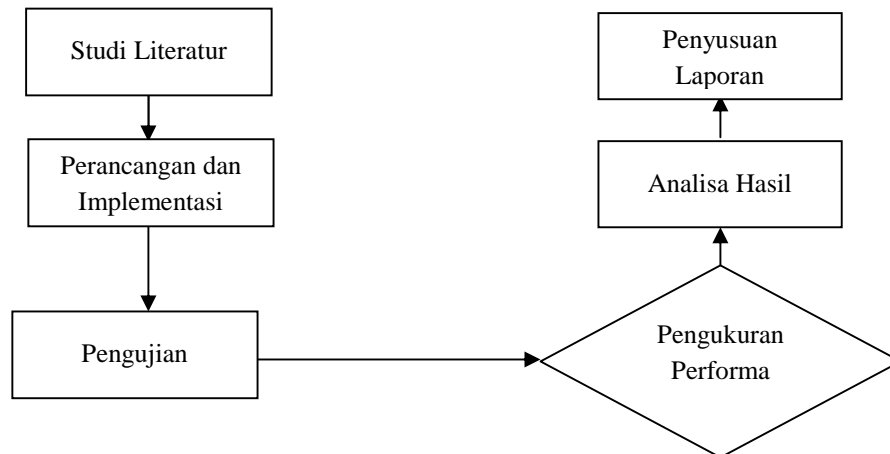
Pada proses otomatisasi data pelatihan, kalimat kebutuhan digeneralisasi menjadi kelas label untuk setiap kalimat kebutuhan. Kelas label terdiri dari kebutuhan fungsional dan non-fungsional. Kelas non-fungsional sendiri terdiri dari 31 fitur yang diambil dari standar internasional ISO/IEC 25010:2011. Kalimat kebutuhan yang telah terlabeli selanjutnya dapat digunakan sebagai input pada proses klasifikasi. Metode klasifikasi akan menggunakan metode FSKNN.

Pada proses pengklasifikasian kalimat kebutuhan, terdapat banyak jenis aspek kualitas yang dapat digunakan. Penelitian ini akan mengidentifikasi aspek kualitas kebutuhan non-fungsional pada kalimat-kalimat kebutuhan seperti yang diusulkan pada paper Huang dkk (2006), Suharso dkk (2012), dan Ramadhani dkk (2015). Fokus penelitian ketiga paper tersebut untuk mengidentifikasi aspek kualitas kebutuhan non-fungsional. Aspek kualitas kebutuhan non-fungsional yang telah diketahui sejak tahap awal analisa kebutuhan diharapkan dapat meminimalisir hal-hal yang tidak diinginkan ketika produk perangkat lunak sudah dibuat.

BAB 3

METODOLOGI PENELITIAN

Bab ini akan memaparkan tentang metodologi penelitian yang digunakan pada penelitian ini, yang terdiri dari (1) studi literatur, (2) penelitian, dan (3) dokumentasi. Ilustrasi alur metodologi penelitian dapat dilihat pada Gambar 3.1



Gambar 3.1 Alur Metodologi Penelitian

Penjelasan tahapan metode penelitian pada Gambar 3.1 akan diterangkan secara terperinci pada sub-bab berikut.

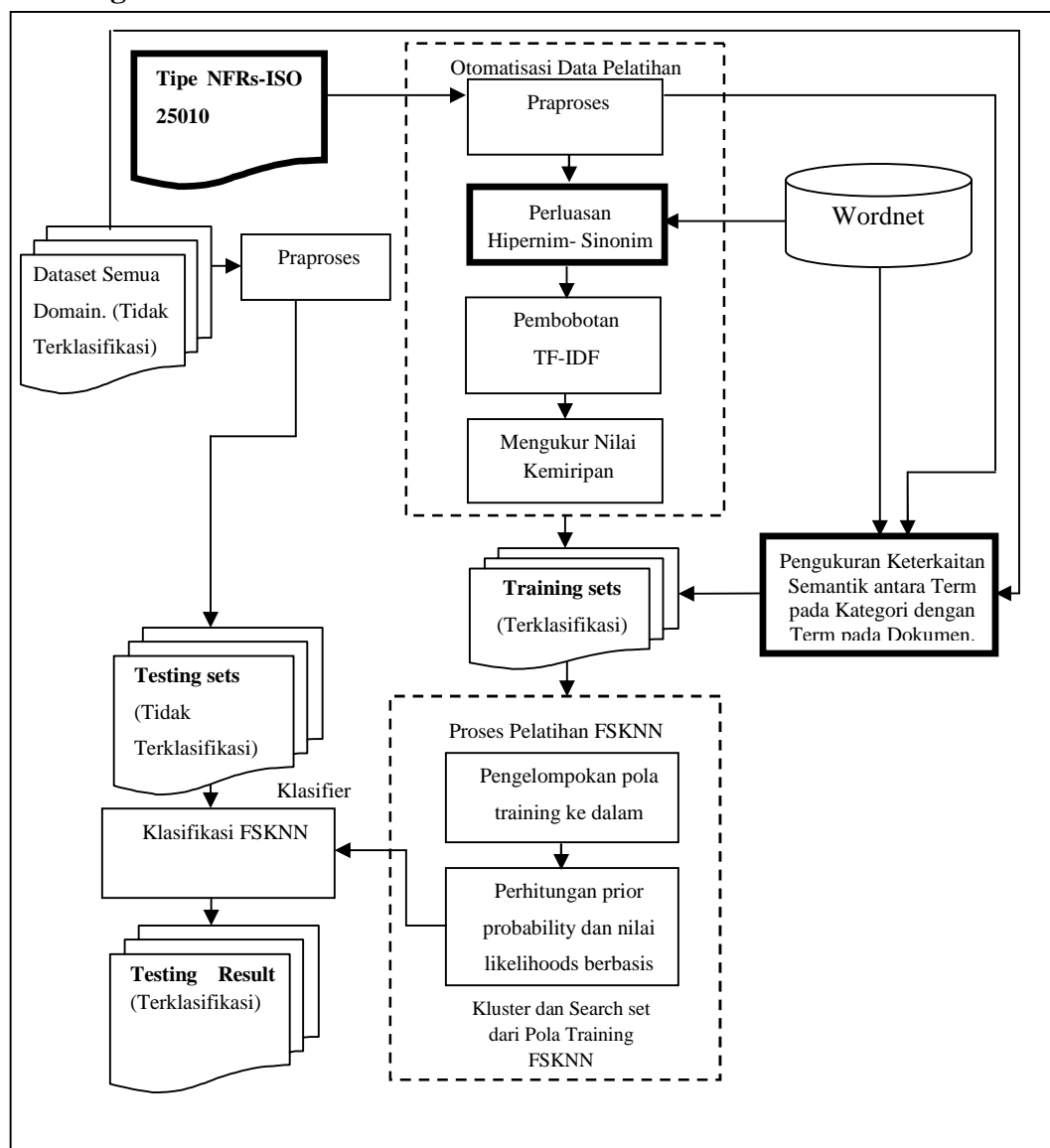
3.1 Studi Literatur

Studi literatur dilakukan untuk mendapatkan informasi tentang perkembangan penelitian terkait identifikasi atau elisitasi aspek kualitas kebutuhan non-fungsional perangkat lunak, baik dalam perkembangan pendekatan metode yang telah digunakan, dan informasi tentang kelebihan dan kekurangan dari masing - masing penelitian pendahulu yang telah dilakukan. Studi literatur yang dilakukan diharapkan dapat memberikan gambaran secara lengkap dan dapat memberikan dasar kontribusi tentang identifikasi aspek kualitas kebutuhan non-fungsional perangkat lunak yang akan dilakukan pada penelitian ini. Studi literatur yang dilakukan yaitu mencari dan mempelajari beberapa referensi sebagai berikut :

1. Penelitian terdahulu yang telah melakukan pembahasan identifikasi aspek kualitas kebutuhan non-fungsional perangkat lunak.

2. Dasar teori tentang kebutuhan non-fungsional, ISO/IEC 25010:2011, klasifikasi dokumen dan klasifikasi *multi-label*.
3. Metode pengukuran keterkaitan semantik *Hirst-St.Onge (HSO)* berbasis wordnet.
4. Metode klasifikasi FSKNN.
5. Dasar teori tentang sistem referensi leksikal wordnet.
6. Metode evaluasi klasifikasi *multi-label* dengan *accuracy, precision, recall*, dan F1.

3.2 Rancangan Arsitektur Deteksi dan Klasifikasi



Gambar 3. 2 Model deteksi dan klasifikasi non-fungsional

Pada Gambar 3.2 terdiri dari dua proses utama yaitu otomatisasi data pelatihan, dan proses klasifikasi yang terdiri dari langkah-langkah pelatihan dan klasifikasi. Proses otomatisasi data pelatihan sendiri terdiri dari tujuh sub-proses yaitu praproses, perluasan gabungan hipernim dan sinonim, pengukuran keterkaitan semantik, *term frequency (tf)*, *inverse document frequency (idf)*, *tf-idf* dan pengukuran kemiripan menggunakan *cosine measure*. Proses ini bertujuan untuk menghasilkan sejumlah daftar kebutuhan terklasifikasi secara otomatis sebagai data latih berdasarkan standart kualitas ISO/IEC 25010:2011.

Sedangkan untuk proses klasifikasi terdiri dari tiga proses utama yaitu pengelompokan pola pelatihan ke dalam kluster, perhitungan prior probability dan nilai likelihoods, dan klasifikasi FSKNN. Proses pertama terdiri dari enam sub-proses yaitu penentuan distribusi term t_i pada aspek kualitas c_j , penentuan derajat keanggotaan $\mu_R(t_i, c_j)$ dan $\mu_d(t_i)$, penentuan nilai similaritas $si(d, c_j)$ dengan fuzzy similarity measure, penentuan nilai derajat keanggotaan $\mu_{c_j}(d)$, pengelompokan data latih ke dalam beberapa kluster, dan pencarian search set untuk setiap data lath dari setiap kluster.

Proses kedua terdiri dari empat sub-proses yaitu penentuan nilai prior probability, penentuan jumlah tetangga terdekat (k), (N^i), dan vektor jumlah label (n^i), penentuan nilai $Z(e, j)$ dan $\bar{Z}(e, j)$, dan penentuan nilai likelihoods $P(E|H_j)$. Sedangkan proses ketiga terdiri dari empat sub-proses yaitu penentuan nilai similaritas $si(d, c_j)$ dan derajat keanggotaan $\mu_{c_j}(d^t)$, pengecekan $\mu_{c_j}(d^t)$ berdasarkan threshold dan menemukan search set, penentuan tetangga terdekat dari dokumen uji (N^i) dan nilai vektor jumlah label (n^i), dan penentuan nilai MAP untuk menentukan aspek kualitas pada dokumen uji.

Pada sub-bab ini akan diberikan contoh eksperimen awal dari setiap fase yang terdapat dalam desain sistem yang telah dijelaskan pada sub-bab 3.2 dan digambarkan pada Gambar 3.2 diatas. Pada eksperimen awal ini dibatasi melakukan klasifikasi ke dalam 4 aspek kualitas yaitu *functional correctness*, *capacity*, *time behaviour*, *maturity* dengan 4 kalimat kebutuhan yang akan dilabeli sebagai data latih, sebagai contoh penjabaran proses hanya dilakukan pada 1 kalimat kebutuhan saja. Ketiga kalimat kebutuhannya adalah sebagai berikut :

KK1. *The system must not charge a diversity user when a call to administrator is placed.*

KK2. *The system must be suitability with user requests.*

KK3. *The maximum time it takes for sound to travel from speaker to receiver should be 500 milliseconds.*

KK4. *The system shall refresh the display every 60 seconds.*

3.2.1 Otomatisasi Data Pelatihan

Fase pertama adalah proses otomatisasi data pelatihan. Fase ini terdiri dari tahap praproses, pengembangan term relevan dengan hipernim dan sinonim, pengukuran keterkaitan semantik, pembobotan TF-IDF, dan pengukuran nilai derajat kemiripan dari setiap kalimat kebutuhan yang akan dijadikan data latih seperti yang telah dijelaskan pada subbab 2.6. Suatu kalimat kebutuhan masuk ke dalam aspek kualitas tertentu berdasarkan penerapan threshold atau ambang batas tertentu.

1. Tahap Praproses

Pada fase ini 1366 kalimat kebutuhan akan dilakukan praproses untuk mendapatkan *term* atau kata dari setiap kalimat kebutuhan yang akan digunakan sebagai data latih. Hasil dari tahap praproses melalui sistem dapat dilihat pada Tabel 3.1 dibawah.

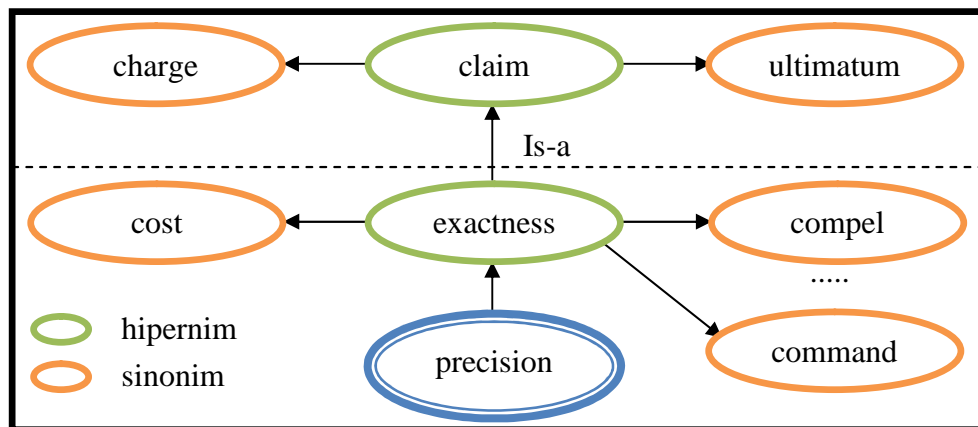
Tabel 3.1 Tahap Praproses Kalimat Kebutuhan KK1

Proses	Hasil
1. Case folding	the system must not charge a diversity user when a call to administrator is placed.
2. Tokenisasi	the, system, must, not, charge, a, diversity, user, when, a, call, to, administrator, is, placed.
3. Filtering	system, must, not, charge, diversity, user, when, call, administrator, placed.
4. Stemming	system, must, charge, diversity, user, call, administr, place.

2. Pengembangan Term Relevan (Hipernim dan Sinonim)

Berdasarkan daftar term relevan pada Tabel 2.2 pada sub-bab 2.5.2 akan dikembangkan term relevan *precision* untuk aspek kualitas *functional correctness*, maka proses pertama adalah mencari hipernim dari term *precision* hingga mencapai 1 level di atasnya dengan 20 term sinonimnya. Penggambaran proses pencarian dapat dilihat pada Gambar 3.3.

Dari Gambar 3.3 dapat diketahui hipernim untuk term *precision* adalah *exactness* untuk level 1 dan *claim* untuk level 2, dimana kemudian masing - masing hipernim tersebut dicari sinonimnya hingga mencapai sejumlah sinonim. Perluasan gabungan hipernim-sinonim ini menggunakan dua skenario. Skenario pertama adalah menggunakan 2 level hipernim dan 9 sinonim pada masing-masing hipernim. Sedangkan skenario kedua menggunakan 1 level hipernim dan 20 sinonim. Total perluasan term yang didapatkan pada skenario satu sebanyak 20 term dan 21 pada skenario dua. Dengan langkah yang sama maka dapat dicari perluasan term untuk semua term relevan pada keempat aspek kualitas. Contoh pengembangan term gabungan antara hipernim dan sinonim pada aspek kualitas *functional correctness* ditunjukkan pada Tabel 3.2.



Gambar 3. 3Ilustrasi Perluasan Term dengan Gabungan Hipernim dan Sinonim
Tabel 3.2 Contoh hasil pengembangan term gabungan antara hipernim dan sinonim pada aspek kualitas *functional correctness*

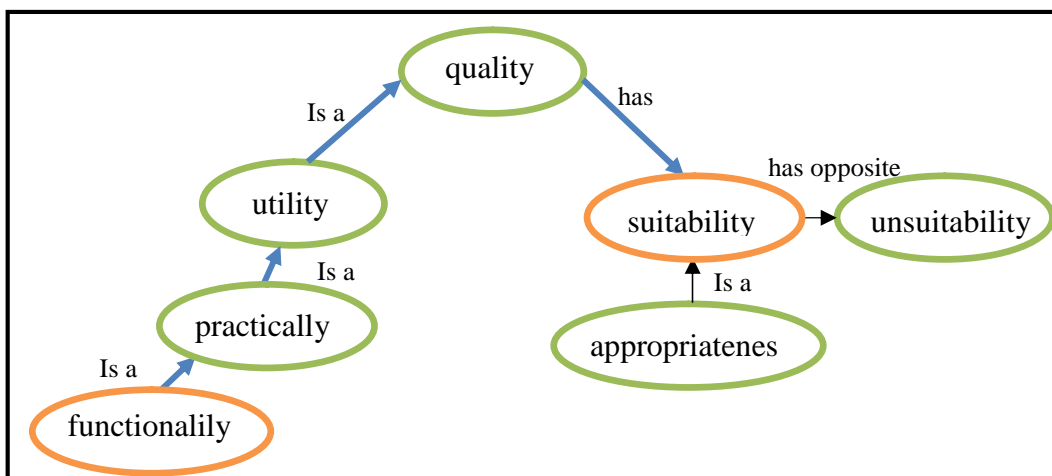
Aspek Kualitas	Term Relevan	Hasil Pengembangan Term dengan Gabungan Hipernim dan Sinonim
Functional Correctness	functionality	operable, functioning, operational, serviceable, useful, usable, operative, practical, useable, structural, working, utilitarian, running
	result	phenomenon, turn, upshot, chelation, shape, variation, fortune, metempsychosis, levitate, loop, process, impeach,
	needed	status, diversity , death, danger, situation, feeling, Order, dishabille, full, ...
	degree	property, division, shiner, fact, salty, viable, rule, tract, treasure, liability, possession, pyromania, ...

Aspek Kualitas	Term Relevan	Hasil Pengembangan Term dengan Gabungan Hipernim dan Sinonim
	precision	<u>exactness</u> , cost, draw, call , command, demand, claim, compel, take, govern, <u>claim</u> , charge, ultimatum, pre emption, seat, allege, contention, pretension, enjoyment, admission

3. Pengukuran keterkaitan semantik antara term pada aspek kualitas dengan term pada dokumen

Berdasarkan daftar *term-term* relevan dan aspek kualitas pada contoh proses otomatisasi data pelatihan, maka dapat diketahui bahwa ada 4 aspek kualitas yaitu *functional correctness*, *capacity*, *time behaviour*, dan *Maturity*. Dimisalkan total keseluruhan term relevan pada keempat aspek kualitas tersebut adalah 10. Term relevan tersebut ditunjukkan dalam Tabel 3.3. Dari data tersebut kemudian akan dilakukan pengukuran keterkaitan semantik antara setiap *term-term relevan* dan *term-term* pada dokumen yang akan dilabeli, sebagai contoh diambil proses pengukuran keterkaitan semantik antara *term functional* dengan *term suitability* pada kalimat kebutuhan KK2.

Memfaatkan referensi leksikal wordnet maka diketahui struktur relasi *horizontal link*, *upward link* dan *downward link* antara *term functional* dengan *term suitability* pada aspek kualitas *functional correctness* seperti yang ditunjukkan pada Gambar 3.4.



Gambar 3.4 Ilustrasi Struktur Relasi untuk *Term Functionality* dan *Term Suitability* pada kalimat kebutuhan KK2 Berbasis *Wordnet*

Berdasarkan struktur relasi pada Gambar 3.4 diketahui bahwa keterkaitan semantik antara *term functionality* dengan *term suitability* pada aspek kualitas *functional correctness* merupakan hubungan relasi *medium strong relation*, dimana jalur berwarna biru merupakan salah satu pola dari jalur yang diperbolehkan.

Berdasarkan Gambar 3.4 diatas dapat diketahui juga nilai dari *number of change of direction* adalah 1, nilai untuk panjang jalur atau *path length* dari kedua kata adalah 4, dengan diketahui nilai - nilai tersebut maka nilai keterkaitan semantik antara *functional* dan *suitability* dapat dihitung dengan persamaan 2.28 sebagai berikut :

$$w_{ht(f,s)} = 9 - 4 - (1 \cdot 1) = 4$$

Sesuai penjelasan pada sub-bab 2.9 dimana nilai yang didapatkan akan disesuaikan dengan konsep fuzzy, maka nilai 4 yang didapatkan pada contoh diatas akan dibagi dengan 18 sehingga nilai yang didapatkan adalah 0,222.

Dengan langkah yang sama seperti penjelasan diatas maka dapat diketahui nilai keterkaitan semantik antara 10 *term* relevan dengan semua term pada kalimat kebutuhan KK2 seperti yang dapat dilihat pada Tabel 3.3.

Tabel 3.3 Ilustrasi Hasil Pengukuran keterkaitan Semantik Antara 10 Term Relevan dengan semua term pada kalimat kebutuhan KK2

Term Set	system	suitability	user	request
(t1) <i>functionality</i>	0.2222	0.2222	0.1667	0.0
(t2) <i>suitability</i>	0.1667	1.0	0.2778	0.1111
(t3) <i>correctness</i>	0.0	0.0	0.0	0.0
(t4) <i>result</i>	0.0	0.0	0.0	0.0
(t5) <i>needed</i>	0.0	0.0	0.0	0.0
(t6) <i>degree</i>	0.0	0.0	0.0	0.1667
(t7) <i>precision</i>	0.2222	0.2222	0.1667	0.0
(t8) <i>reliability</i>	0.3333	0.2778	0.2778	0.1111
(t9) <i>maturity</i>	0.1667	0.1667	0.1111	0.2222
(t10) <i>under</i>	0.0	0.0	0.0	0.0

Penambahan faktor semantik merupakan pengukuran keterkaitan semantik antara semua term relevan pada aspek kualitas dengan seluruh term pada dokumen yang akan dilabeli, dimana hasil dari pengukuran ini akan ditambahkan pada tahapan otomatisasi data pelatihan, proses ini memiliki tujuan untuk

menghasilkan data latih baru. Data latih baru tersebut akan dibandingkan dengan data latih yang diperoleh tanpa pengukuran semantik. Hal ini dilakukan karena ada kemungkinan data latih tanpa pengukuran semantik masih memiliki kesalahan (Jiang, et al., 2012). Contoh berikut memberikan gambaran mengapa penambahan pengukuran keterkaitan semantik diperlukan dalam proses otomatisasi data pelatihan.

Dalam proses pencarian TF pada kalimat kebutuhan KK2. Jika tanpa adanya pengukuran keterkaitan semantik, maka hasil pengukuran frekuensi term antara term relevan aspek kualitas *functional correctness* dengan kalimat kebutuhan KK2 hanya akan menghasilkan nilai 1 jumlah *term* frekuensi yaitu *term suitability*, sedangkan untuk keseluruhan *term* yang lain akan menghasilkan nilai 0. Hal ini dikarenakan *term* relevan yang memiliki kesamaan dengan kalimat kebutuhan KK2 hanya satu *term* saja. Sedangkan jika term relevan pada aspek kualitas *functional correctness* tersebut telah dilakukan pengukuran keterkaitan semantik, maka hasil pengukuran jumlah frekuensi akan menghasilkan hasil yang berbeda seperti pada Tabel 3.4 dibawah :

Tabel 3.4 Ilustrasi Pencarian Term Frequency (TF) antara aspek kualitas *functional correctness* dengan kalimat kebutuhan KK2

Bentuk Kebutuhan	Term Relevan	TF
Aspek kualitas <i>functional correctness</i>	<i>functional</i>	1
	<i>suitability</i>	1
	<i>correctness</i>	1
	<i>result</i>	1
	<i>needed</i>	1
	<i>degree</i>	1
	<i>precision</i>	1
Kalimat kebutuhan KK2	<i>functional</i>	2
	<i>suitability</i>	2
	<i>correctness</i>	0
	<i>result</i>	0
	<i>needed</i>	0
	<i>degree</i>	0
	<i>precision</i>	2

Dari Tabel 3.4 diatas diketahui bahwa untuk term relevan *functional* memiliki jumlah term relevan 2, hal ini dikarenakan dari pengukuran keterkaitan

semantik untuk *term* relevan *functional* memiliki hubungan semantik dengan tiga term yaitu *system* sebesar 0.2222, *suitability* sebesar 0.2222, dan *user* sebesar 0.1667. Jika *threshold* yang digunakan adalah 0.2 maka term *functional* dipastikan memiliki hubungan semantik dengan *term system* dan *suitability*.

4. Pembobotan TF-IDF

Pembobotan TF-IDF terdiri dari tiga proses pencarian *term frequency (TF)* dari setiap *term*, pencarian kemunculan tiap *term* terhadap seluruh kalimat kebutuhan atau disebut *inverse document frequency (IDF)*, dan yang terakhir adalah pembobotan TF-IDF dengan normalisasi nilai.

a. Term Frequency

Dalam proses pencarian TF, frekuensi yang muncul dari term hasil pengembangan pada tahap sebelumnya diakumulasikan ke dalam *term* relevan. Jika tanpa adanya pengembangan term dengan hipernim dan sinonim, maka hasil pengukuran frekuensi term antara term relevan aspek kualitas *functional correctness* dengan kalimat kebutuhan KK1 akan menghasilkan jumlah frekuensi 0 untuk keseluruhan term relevan dikarenakan tidak ada satupun term relevan yang muncul pada kalimat kebutuhan KK1 diatas. Sedangkan jika term relevan pada aspek kualitas *functional correctness* tersebut telah dilakukan pengembangan dengan hipernim dan sinonim, maka hasil pengukuran jumlah frekuensi akan menghasilkan hasil yang berbeda seperti pada Tabel 3.5 dibawah :

Tabel 3.5 Ilustrasi Pencarian Term Frequency (TF) antara aspek kualitas *functional correctness* dengan kalimat kebutuhan KK1

Bentuk Kebutuhan		Term Relevan	TF
Aspek Kualitas <i>Functional</i> <i>Correctness</i>		<i>functional</i>	1
		<i>suitability</i>	1
		<i>correctness</i>	1
		<i>result</i>	1
		<i>needed</i>	1
		<i>degree</i>	1
		<i>precision</i>	1
Kalimat KK1	Kebutuhan	<i>functional</i>	0
		<i>suitability</i>	0
		<i>correctness</i>	0

Bentuk Kebutuhan	Term Relevan	TF
	<i>result</i>	0
	<i>needed</i>	1
	<i>degree</i>	0
	<i>precision</i>	1

Dari Tabel 3.5 diatas diketahui bahwa untuk *term* relevan *needed* dan *precision* memiliki masing - masing jumlah *term* relevan 1, hal ini dikarenakan dari pengembangan term dengan menggunakan hipernim dan sinonim untuk *term* relevan *needed* menghasilkan term *diversity* dan untuk term relevan *precision* menghasilkan term *call*, dimana term *diversity* dan *term call* muncul pada kalimat kebutuhan KK1 sebanyak satu kali. Keseluruhan hasil pengembangan *term* hipernim dan sinonim pada aspek kualitas *functional correctness* dapat dilihat pada Tabel 3.2. Pengembangan ini adalah term yang dihasilkan oleh penelitian yang dilakukan oleh Ramadhani dkk (2015) (Ramadhani, Rochimah, dan Yuhana, 2015).

b. Invers Document Frequency (IDF)

Berdasarkan Tabel 3.5, maka dapat dihitung nilai IDF untuk setiap *term* dengan menggunakan persamaan 2.25. Sebagai contoh akan dihitung nilai IDF dari *term correct* dari hubungan aspek kualitas *functional correctness* pada kalimat kebutuhan 1, sebagai berikut :

$$\begin{aligned}
 \text{IDF} &= \log \left(\frac{2}{1} \right) + 1 \\
 &= 0.301 + 1 \\
 &= \mathbf{1.3}
 \end{aligned}$$

Serta term *needed* dari hubungan aspek kualitas *functional correctness* pada kalimat kebutuhan 1, sebagai berikut :

$$\begin{aligned}
 \text{IDF} &= \log \left(\frac{2}{2} \right) + 1 \\
 &= 0 + 1 \\
 &= \mathbf{1}
 \end{aligned}$$

Dengan cara yang sama maka didapatkan nilai IDF untuk *term-term* relevan yang lain seperti yang ditunjukkan pada Tabel 3.6.

Tabel 3.6 Ilustrasi Perhitungan IDF antara kualitas suitability dengan KK1

Term	IDF
<i>functional</i>	1.301
<i>suitability</i>	1.301
<i>correctness</i>	1.301
<i>result</i>	1.301
<i>needed</i>	1
<i>degree</i>	1.301
<i>precision</i>	1

c. Pembobotan TF-IDF

Berdasarkan hasil pencarian dan perhitungan TF dan IDF antara aspek kualitas *functional correctness* dengan kalimat kebutuhan KK1 pada Tabel 3.5 dan 3.6, maka langkah selanjutnya adalah melakukan pembobotan TF-IDF dengan menggunakan persamaan 2.26. Sebagai contoh dilakukan perhitungan pembobotan untuk *term correct* dan *term needed* pada kalimat kebutuhan KK1 sebagai berikut:

$$\omega_K(t_2) = \frac{t_i(t_2, K) \times l_i \frac{N}{d(t_2, K)} + 1}{\sqrt{t_i(t_i, K) \times l_i \frac{N}{d(t_i, K)} + 1}^2}$$

$$= (tf_{i1,K} \times idf_{i1}) / \text{sqrt} ((tf_{i1,K} \times idf_{i1})^2 + (tf_{i2,K} \times idf_{i2})^2 + (tf_{i3,K} \times idf_{i3})^2 + (tf_{i4,K} \times idf_{i4})^2 + (tf_{i5,K} \times idf_{i5})^2 + (tf_{i6,K} \times idf_{i6})^2 + (tf_{i7,K} \times idf_{i7})^2)$$

$$= (0 \times 1.301) / \text{sqrt} ((0 \times 1.301)^2 + (0 \times 1.301)^2 + (0 \times 1.301)^2 + (0 \times 1.301)^2 + (1 \times 1)^2 + (0 \times 1.301)^2 + (1 \times 1)^2)$$

$$= \mathbf{0.0}$$

“Hasil TFxIDF untuk Term **correctness**”

$$= (tf_{i1,K} \times idf_{i1}) / \text{sqrt} ((tf_{i1,K} \times idf_{i1})^2 + (tf_{i2,K} \times idf_{i2})^2 + (tf_{i3,K} \times idf_{i3})^2 + (tf_{i4,K} \times idf_{i4})^2 + (tf_{i5,K} \times idf_{i5})^2 + (tf_{i6,K} \times idf_{i6})^2 + (tf_{i7,K} \times idf_{i7})^2)$$

$$= (1 \times 1) / \text{sqrt} ((0 \times 1.301)^2 + (0 \times 1.301)^2 + (0 \times 1.301)^2 + (0 \times 1.301)^2 + (1 \times 1)^2 + (0 \times 1.301)^2 + (1 \times 1)^2)$$

$$= \mathbf{0.707}$$

“Hasil TFxIDF untuk Term **needed**”

Dengan cara yang sama seperti proses diatas, maka dapat dicari bobot TF-IDF untuk semua *term* pada setiap bentuk kebutuhan seperti yang ditunjukkan pada Tabel 3.7.

Tabel 3.7 Ilustrasi Hasil Perhitungan Bobot TF-IDF untuk aspek kualitas functional correctness dan kalimat kebutuhan KK1

Bentuk Kebutuhan	Term Relevan	TF-IDF
Aspek kualitas <i>functional correctness</i>	<i>functional</i>	0.402
	<i>suitability</i>	0.402
	<i>correctness</i>	0.402
	<i>result</i>	0.402
	<i>needed</i>	0.309
	<i>degree</i>	0.402
	<i>precision</i>	0.309
Kalimat Kebutuhan KK1	<i>functional</i>	0
	<i>suitability</i>	0
	<i>correctness</i>	0
	<i>result</i>	0
	<i>needed</i>	0.707
	<i>degree</i>	0
	<i>precision</i>	0.707

5. Pengukuran derajat Kemiripan

Setelah dilakukan tahapan pembobotan TF-IDF maka dapat dilakukan pengukuran nilai kemiripan antara suatu kalimat kebutuhan dengan suatu atribut aspek kualitas dengan persamaan 2.27. Meneruskan contoh pada tahap sebelumnya akan diukur nilai kemiripan antara kalimat kebutuhan KK1 dengan atribut aspek kualitas *functional correctness* sebagai berikut :

$$C(Q, K) = \sum_{i=1}^M \omega_Q(t_i) \times \omega_K(t_i)$$

$$= (suit_{t1} \times K_{t1}) + (suit_{t2} \times K_{t2}) + (suit_{t3} \times K_{t3}) + (suit_{t4} \times K_{t4}) + (suit_{t5} \times K_{t5}) + (suit_{t6} \times K_{t6}) + (suit_{t7} \times K_{t7})$$

$$= (0.402 \times 0) + (0.402 \times 0) + (0.402 \times 0) + (0.402 \times 0) + (0.309 \times 0.707) + (0.402 \times 0) + (0.309 \times 0.707)$$

$$= \mathbf{0.219}$$

Dengan tahapan proses yang sama dapat diperoleh nilai kemiripan antara kalimat kebutuhan KK1 dengan aspek kualitas lainnya seperti yang ditunjukkan pada Tabel 3.8. Nilai *threshold* tertentu digunakan untuk melabeli kalimat kebutuhan KK1 ke dalam satu atau lebih aspek kualitas.

Tabel 3.8 Daftar Nilai Derajat Kemiripan Antara Semua Aspek Kualitas dengan Kalimat Kebutuhan KK1

Aspek Kualitas	Nilai Derajat Kemiripan
<i>Functional Correctness</i>	0.219
<i>Capacity</i>	0.0
<i>Time behaviour</i>	0.0
<i>Maturity</i>	0.359

Dimisalkan untuk nilai derajat kemiripan pada Tabel 3.8 diterapkan *threshold* 0.1 maka dapat disimpulkan kalimat kebutuhan KK1 termasuk ke dalam aspek kualitas *functional correctness* dan *maturity*.

Dengan cara yang sama seperti yang dilakukan pada kalimat kebutuhan KK1, maka kalimat kebutuhan KK2, KK3 dan KK4 juga dapat dilabeli secara otomatis untuk digunakan sebagai data latih. Hasil atau keluaran dari fase ini adalah semua kalimat kebutuhan yang sudah terlabeli seperti Tabel 3.9 dan juga data TF untuk semua kalimat kebutuhan seperti pada Tabel 3.10. Kedua data tersebut nantinya digunakan untuk proses pelatihan pada metode FSKNN.

Tabel 3.9 Ilustrasi Hasil Otomatisasi Pelabelan untuk 4 Kalimat Kebutuhan

Kalimat Kebutuhan	Aspek Kualitas	Binary Label
KK1	<i>Functional correctness, maturity</i>	1,0,0,1
KK2	<i>Functional correctness, time behaviour.</i>	1,0,1,0
KK3	<i>Functional correctness, time behaviour</i>	1,0,1,0
KK4	<i>Functional correctness, time behaviour</i>	1,0,1,0

Tabel 3. 10 Ilustrasi Data Frekuensi 4 aspek kualitas terhadap 4 kalimat kebutuhan

Kalimat Kebutuhan	Data Frekuensi (TF) untuk 4 aspek kualitas
KK1	1,2,1,0,1,0,0,1,0,1

KK2	0,0,0,1,0,1,1,0,1,1
KK3	1,0,2,0,0,0,0,1,0,1
KK4	0,0,0,0,1,0,1,0,0,0

3.2.2 Pengelompokan Pola Pelatihan

Berdasarkan hubungan similaritas antara dokumen dan aspek kualitas kemudian dilakukan pengelompokan ke dalam beberapa kluster dengan menggunakan suatu nilai *threshold* yang telah didefinisikan sebelumnya. Tujuan pada fase ini adalah untuk mengelompokkan data latih ke dalam kluster berdasarkan nilai similaritas dan menentukan *search set* pada setiap data latih untuk mendukung fase pelatihan berikutnya.

1. Penentuan distribusi $term t_i$ pada aspek kualitas c_j

Penentuan distribusi $d(t_i, c_j)$ dan $dd(t_i, c_j)$ berdasarkan pada Tabel 3.9 dan Tabel 3.10 dengan diketahui :

- a. 1 i 10 yang merepresentasikan jumlah term dan 1 j 4 yang merepresentasikan jumlah aspek kualitas.
- b. Dimisalkan penentuan $d(t_i, c_j)$ dan $dd(t_i, c_j)$ untuk t_5 (*needed*) pada aspek kualitas c_1, c_2, c_3 , dan c_4 :

$$\begin{aligned}
 d(t_5, c_1) &= \frac{(1x1)+(0x1)+(0x1)+(1x1)}{1+0+0+1} = \mathbf{1} & d(t_5, c_1) &= \frac{(1x1)+(0x1)+(0x1)+(1x1)}{1+1+1+1} = \mathbf{0.5} \\
 d(t_5, c_2) &= \frac{(1x0)+(0x0)+(0x0)+(1x0)}{1+0+0+1} = \mathbf{0} & d(t_5, c_2) &= \frac{(1x0)+(0x0)+(0x0)+(1x0)}{0+0+0+0} = \mathbf{0} \\
 d(t_5, c_3) &= \frac{(1x0)+(0x1)+(0x1)+(1x1)}{1+0+0+1} = \mathbf{0.5} & d(t_5, c_3) &= \frac{(1x0)+(0x1)+(0x1)+(1x1)}{0+1+1+1} = \mathbf{0.3} \\
 d(t_5, c_4) &= \frac{(1x1)+(0x0)+(0x0)+(1x0)}{1+0+0+1} = \mathbf{0.5} & d(t_5, c_4) &= \frac{(1x1)+(0x0)+(0x0)+(1x0)}{1+0+0+0} = \mathbf{1}
 \end{aligned}$$

Tabel 3.11 Hasil penentuan nilai $dt(t_i, c_j)$ dan $dd(t_i, c_j)$

<i>Term Set</i>	$d(t_i, c_j)$				$dd(t_i, c_j)$			
	<i>C1</i>	<i>C2</i>	<i>C3</i>	<i>C4</i>	<i>C1</i>	<i>C2</i>	<i>C3</i>	<i>C4</i>
<i>(t1) functional</i>	1	1	0	0	1	1	0	0
<i>(t2) suitability</i>	1	1	0	0	0.5	0.5	0	0
<i>(t3) correctness</i>	1	1	0	0	1	1	0	0
<i>(t4) result</i>	0	0	1	1	0	0	1	0.5
<i>(t5) needed</i>	1	0	0.5	0.5	0.5	0	0.3	1
<i>(t6) degree</i>	0	0	1	1	0	0	1	0.5
<i>(t7) precision</i>	0	0	0.5	1	0	0	1	1

<i>(t8) reliability</i>	1	1	0.0	0	1	1	0	0
<i>(t9) maturity</i>	0	0	1	1	0	0	1	0.5
<i>(t10) under</i>	0.667	0.667	0.334	0.334	1	1	1	0.0

2. Penentuan derajat keanggotaan $\mu_R(t_i, c_j)$ dan $\mu_d(t_i)$

Data dari Tabel 3.11 maka akan digunakan untuk penentuan derajat keanggotaan $\mu_R(t_i, c_j)$ berdasarkan persamaan 2.10 dan penentuan derajat keanggotaan $\mu_d(t_i)$ berdasarkan persamaan 2.14. Dimisalkan penentuan $\mu_R(t_i, c_j)$ untuk $t_5(\text{needed})$ pada aspek kualitas c_1, c_2, c_3 , dan c_4 :

$$\begin{aligned} \mu_R(t_5, c_1) &= \frac{1}{1} \times \frac{0.5}{1} = \mathbf{0.5} & \mu_R(t_5, c_3) &= \frac{0.5}{1} \times \frac{0.3}{1} = \mathbf{0.1} \\ \mu_R(t_5, c_2) &= \frac{0}{1} \times \frac{0}{1} = \mathbf{0} & \mu_R(t_5, c_4) &= \frac{0.5}{1} \times \frac{1}{1} = \mathbf{0.5} \end{aligned}$$

Tabel 3.12 Hasil Penentuan Derajat Keanggotaan $\mu_R(t_i, c_j)$

<i>Term Set</i>	Aspek Kualitas			
	<i>F. Correctness</i> (c1)	<i>Capacity</i> (c2)	<i>TB</i> (c3)	<i>Maturity</i> (c4)
<i>(t1) functional</i>	1	1	0	0
<i>(t2) suitability</i>	0.5	0.5	0	0
<i>(t3) correctness</i>	1	1	0	0
<i>(t4) result</i>	0	0	1	0.5
<i>(t5) needed</i>	0.5	0	0.165	0.5
<i>(t6) degree</i>	0	0	1	0.5
<i>(t7) precision</i>	0	0	0.5	1
<i>(t8) reliability</i>	1	1	0	0
<i>(t9) maturity</i>	0	0	1	0.5
<i>(t10) under</i>	0.667	0.667	0.334	0

Sedangkan untuk penentuan $\mu_d(t_i)$ menggunakan data yang ditunjukkan pada Tabel 3.9, dimisalkan penentuan $\mu_d(t_i)$ untuk kalimat kebutuhan KK1 sebagai berikut :

$$\begin{aligned} \mu_{K1}(t_1) &= \mu_{K1}(t_3) = \mu_{K1}(t_5) = \mu_{K1}(t_8) = \mu_{K1}(t_{11}) = \frac{1}{2} = 0.5 \\ \mu_{K1}(t_4) &= \mu_{K1}(t_6) = \mu_{K1}(t_7) = \mu_{K1}(t_9) = \frac{0}{2} = 0 \end{aligned}$$

$$\mu_{K1}(t_2) = \frac{2}{2} = 1$$

Sehingga hasil keseluruhan derajat keanggotaan $\mu_d(t_i)$ seperti yang ditunjukkan pada Tabel 3.13.

Tabel 3.13 Hasil Penentuan Derajat Keanggotaan $\mu_d(t_i)$

<i>Term Set</i>	Kalimat Kebutuhan			
	<i>KK1</i>	<i>KK2</i>	<i>KK3</i>	<i>KK4</i>
<i>(t1) functional</i>	0.5	0	0.5	0
<i>(t2) suitability</i>	1	0	0	0
<i>(t3) correctness</i>	0.5	0	1	0
<i>(t4) result</i>	0	1	0	0
<i>(t5) needed</i>	0.5	0	0	1
<i>(t6) degree</i>	0	1	0	0
<i>(t7) precision</i>	0	1	0	1
<i>(t8) reliability</i>	0.5	0	0.5	0
<i>(t9) maturity</i>	0	1	0	0
<i>(t10) under</i>	0.5	1	0.5	0

3. Penentuan nilai similaritas $si(d, c_j)$ dengan *fuzzy similarity measure*

Derajat keanggotaan $\mu_R(t_i, c_j)$ dan $\mu_d(t_i)$ dari data yang didapatkan pada Tabel 3.12 dan Tabel 3.13 dapat digunakan untuk penentuan nilai similaritas dengan *fuzzy similarity measure* berdasarkan persamaan 2.11, 2.12, 2.13. Dimisalkan penentuan similaritas untuk kalimat kebutuhan KK1 terhadap aspek kualitas c_1 adalah sebagai berikut :

$$\begin{aligned}
 si(d_1, c_1) &= \frac{\sum_{i=1}^n \mu_R(t_i, c_1) \otimes \mu_d(t_i)}{\sum_{i=1}^n \mu_R(t_i, c_1) \oplus \mu_d(t_i)} \\
 &= \frac{1 \quad 0.5 + 0.5 \quad 1 + 1 \quad 0.5 + \dots + 0.667 \quad 0.5}{1 \quad 0.5 + 0.5 \quad 1 + 1 \quad 0.5 + \dots + 0.667 \quad 0.5} \\
 &= \frac{2.584}{9.167} = 0.282
 \end{aligned}$$

Dengan langkah yang sama maka didapatkan nilai similaritas antara setiap kalimat kebutuhan dengan setiap aspek kualitas yang ditunjukkan pada Tabel 3.14.

Tabel 3.14 Hasil Penentuan Similaritas Antara Kalimat Kebutuhan dengan Aspek Kualitas

Aspek Kualitas	Kalimat Kebutuhan			
	KK ₁	KK ₂	KK ₃	KK ₄
c ₁	0.282	0.042	0.304	0.048
c ₂	0.269	0.044	0.350	0
c ₃	0.018	0.371	0.014	0.062
c ₄	0.022	0.227	0	0.214

Nilai yang bercetak tebal pada Tabel 3.14 diatas merupakan nilai tertinggi dari pengukuran setiap kalimat kebutuhan terhadap setiap aspek kualitas, nilai tersebut nantinya digunakan untuk proses pada tahap selanjutnya.

4. Penentuan nilai derajat keanggotaan $\mu_{c_j}(d)$

Setelah proses penentuan similaritas antara kalimat kebutuhan dengan aspek kualitas, tahapan selanjutnya adalah menentukan derajat keanggotaan antara kalimat kebutuhan dengan aspek kualitas $\mu_{c_j}(d)$ berdasarkan persamaan 2.15. Menggunakan data pada Tabel 3.14, diambil contoh penentuan $\mu_{c_j}(d)$ untuk beberapa kalimat kebutuhan dan ketegori yaitu kk₃ dengan c₁, kk₄ dengan c₂, kk₁ dengan c₄, kk₂ dengan c₄, dan k₄ dengan c₄ sebagai berikut :

$$\mu_{c_1}(k_3) = \frac{si(d_1, c_1)}{\sum_{1 \leq v \leq p} si(d_1, c_v)} = \frac{0.022}{0.282} = 0.0$$

$$\mu_{c_2}(k_4) = \frac{si(d_2, c_2)}{\sum_{1 \leq v \leq p} si(d_2, c_v)} = \frac{0}{0.214} = 0$$

$$\mu_{c_4}(kk_2) = \frac{si(d_2, c_4)}{\sum_{1 \leq v \leq p} si(d_2, c_v)} = \frac{0.227}{0.067} = 0.6$$

$$\mu_{c_4}(k_4) = \frac{si(d_4, c_4)}{\sum_{1 \leq v \leq p} si(d_4, c_v)} = \frac{0.214}{0.214} = 1$$

$$\mu_{c_1}(k_3) = \frac{si(d_3, c_1)}{\sum_{1 \leq v \leq p} si(d_3, c_v)} = \frac{0.304}{0.350} = 0.8$$

Hasil penentuan $\mu_{c_j}(d)$ untuk semua kalimat kebutuhan ditunjukkan pada Tabel 3.15.

Tabel 3.15 Hasil Penentuan Derajat Keanggotaan $\mu_{c_j}(d)$

Aspek Kualitas	Kalimat Kebutuhan			
	KK ₁	KK ₂	KK ₃	KK ₄

Aspek Kualitas	Kalimat Kebutuhan			
	KK ₁	KK ₂	KK ₃	KK ₄
c ₁	1	0.113	0.869	0.224
c ₂	0.954	0.119	1	0
c ₃	0.064	1	0.040	0.290
c ₄	0.078	0.612	0	1

5. Pengelompokan data pelatihan ke dalam beberapa kluster

Berdasarkan data yang didapatkan pada Tabel 3.15 kemudian akan dilakukan pengelompokan ke dalam beberapa kluster berdasarkan nilai *threshold* () tertentu berdasarkan persamaan 2.16 dengan proses pada *pseudo-code* yang telah dijelaskan pada Gambar 2.7. Pada ilustrasi ini akan digunakan nilai = 0.7, sehingga kluster yang terbentuk adalah sebagai berikut :

$$S_1 = K_1, K_3$$

$$S_2 = K_1, K_3$$

$$S_3 = K_2$$

$$S_4 = K_4$$

6. Pencarian *search set* untuk setiap data pelatihan dari setiap kluster

Proses pencarian *search set* untuk setiap data pelatihan dari setiap kluster juga berdasarkan pada *pseudo-code* yang dijelaskan pada Gambar 2.7. Hasil pencarian *search set* tersebut adalah sebagai berikut :

$$G_1 = S_1 \cup S_2$$

$$G_2 = S_3$$

$$G_3 = S_1 \cup S_2$$

$$G_4 = S_4$$

3.2.3 Penghitungan *Prior Probability* dan Nilai *likelihoods*

1. Penentuan nilai *prior probability*

Tahapan selanjutnya dalam proses pelatihan adalah menentukan nilai *prior probability* berdasarkan pada persamaan 2.17 dan 2.18. Dalam proses penentuan nilai *prior probability* menggunakan nilai *smoothing constant* (*s*), seperti pada penelitian sebelumnya dalam ilustrasi ini akan digunakan nilai *s* = 0.1.

Berdasarkan data pada Tabel 3.9, dimisalkan contoh penentuan nilai *prior probability* untuk aspek kualitas $c_1, c_2, c_3,$ dan c_4 sebagai berikut :

c_1

$$P(H_1 = 1) = \frac{0.1+1+1+1+1}{2 \cdot 0.1+4} = \frac{4.1}{4.2} = \mathbf{0.9}$$

$$P(H_1 = 0) = 1 - 0.976 = \mathbf{0.0}$$

c_2

$$P(H_1 = 1) = \frac{0.1+0+0+0+0}{2 \cdot 0.1+4} = \frac{0.1}{4.2} = \mathbf{0.0}$$

$$P(H_1 = 0) = 1 - 0.024 = \mathbf{0.9}$$

c_3

$$P(H_1 = 1) = \frac{0.1+0+1+1+1}{2 \cdot 0.1+4} = \frac{3.1}{4.2} = \mathbf{0.7}$$

$$P(H_1 = 0) = 1 - 0.738 = \mathbf{0.2}$$

c_4

$$P(H_1 = 1) = \frac{0.1+1+0+0+0}{2 \cdot 0.1+4} = \frac{1.1}{4.2} = \mathbf{0.2}$$

$$P(H_1 = 0) = 1 - 0.262 = \mathbf{0.7}$$

Dengan langkah yang sama maka didapatkan nilai *prior probability* untuk seluruh aspek kualitas sebagai berikut :

Tabel 3.16 Hasil Penentuan Nilai Prior Probability

$P(H_j = 1)$				$P(H_j = 0)$			
c_1	c_2	c_3	c_4	c_1	c_2	c_3	c_4
0.976	0.024	0.738	0.262	0.024	0.976	0.262	0.738

2. Penentuan jumlah tetangga terdekat (k), penentuan tetangga terdekat (N^i) dan vektor jumlah label (n^i)

Tahapan selanjutnya adalah menentukan jumlah tetangga terdekat (k), jumlah tetangga terdekat yang digunakan dalam ilustrasi ini adalah 2. Setelah jumlah tetangga terdekat didefinisikan maka selanjutnya adalah mencari atau menentukan tetangga terdekat untuk setiap data pelatihan (N^i).

Penentuan N^i berdasarkan pada kluster dan *search set* yang telah didapatkan pada proses sebelumnya, dimisalkan untuk melakukan penentuan tetangga terdekat untuk kalimat kebutuhan KK_1 berarti berdasarkan pada *search set* G_1 , sehingga didapatkan tetangga terdekat untuk kalimat kebutuhan KK_1 adalah $\{KK_3\}$.

Tahapan selanjutnya adalah penentuan nilai vektor jumlah label (n^i) berdasarkan pada persamaan 2.19 dengan menggunakan data yang didapatkan ketika menentukan N^i dan data pada Tabel 3.9, misal penentuan nilai n^i untuk kalimat kebutuhan KK_1 , karena hanya ada satu kalimat kebutuhan terdekat yaitu KK_3 maka nilai n^i untuk KK_1 sama seperti vektor label dari KK_3 , sebagai berikut:

$$n^1 = l_{K3}$$

$$n^1 = 1,0,1,0$$

Dengan langkah yang sama, maka dihasilkan nilai N^i dan n^i untuk semua kalimat kebutuhan sebagai berikut :

Tabel 3.17 Hasil Penentuan nilai N^i dan n^i

Kalimat	N^i	n^i
KK_1	{K3}	1,0,1,0
KK_2	{ \emptyset }	0,0,0,0
KK_3	{K1}	1,0,0,1
KK_4	{ \emptyset }	0,0,0,0

Hasil penentuan tetangga terdekat pada K_2 dan K_4 sama dengan \emptyset karena pada tahap pembentukan kluster terlihat bahwa kluster yang memiliki anggota kalimat kebutuhan K_2 atau K_4 tidak memiliki anggota lain selain kalimat kebutuhan K_2 atau K_4 itu sendiri, sehingga kalimat K_2 maupun K_4 tidak memiliki tetangga terdekat satupun. Untuk mengatasi hal ini seharusnya jumlah kalimat kebutuhan yang akan digunakan sebagai data latih harus dalam jumlah yang besar.

3. Penentuan nilai $Z(e, j)$ dan $\bar{Z}(e, j)$

Tahapan selanjutnya adalah penentuan nilai $Z(e, j)$ dan $\bar{Z}(e, j)$, kedua nilai ini nantinya digunakan untuk penentuan nilai *likelihood* pada proses selanjutnya. Penentuan nilai $Z(e, j)$ dan $\bar{Z}(e, j)$ menggunakan persamaan 2.20, 2.21 dan 2.22 dengan memanfaatkan data yang didapatkan pada Tabel 1 dan Tabel 3.17. Misal dilakukan penentuan nilai $Z(e, j)$ dan $\bar{Z}(e, j)$ untuk $e = 0$, dimana $0 \leq e < k$ dan $j = 1$, dimana $0 \leq j < \text{jumlah label}$, sebagai berikut :

$$\begin{aligned} Z(0,1) &= y_1 \delta_0(1) + y_1 \delta_0(1) + y_1 \delta_0(1) + y_1 \delta_0(1) \\ &= (1 \times 0) + (1 \times 1) + (1 \times 0) + (1 \times 1) = 2 \end{aligned}$$

$$\begin{aligned} \bar{Z}(0,1) &= \bar{y}_1 \delta_0(1) + \bar{y}_1 \delta_0(1) + \bar{y}_1 \delta_0(1) + \bar{y}_1 \delta_0(1) \\ &= (0 \times 0) + (0 \times 1) + (0 \times 0) + (0 \times 1) = 0 \end{aligned}$$

Berdasarkan tahapan yang sama maka didapatkan keseluruhan nilai $Z(e, j)$ dan $\bar{Z}(e, j)$ yang ditunjukkan pada Tabel 3.18.

Tabel 3 18 Hasil Penentuan Nilai $Z(e, j)$ dan $\bar{Z}(e, j)$

	$Z(e, j)$			$\bar{Z}(e, j)$		
	$e = 0$	$e = 1$	$e = 2$	$e = 0$	$e = 1$	$e = 2$
$j = 1$	2	0	0	0	0	0
$j = 2$	0	0	0	4	0	0
$j = 3$	3	2	0	0	0	0
$j = 4$	1	2	0	2	0	0

4. Penentuan nilai *likelihoods* $P(E|H_j)$

Tahapan selanjutnya adalah menentukan nilai *likelihood* berdasarkan pada persamaan 2.23 dan 2.24 dengan menggunakan data yang telah didapatkan pada Tabel Tabel 3.18, dimisalkan dilakukan penentuan nilai *likelihood* dengan $e = 0$, dimana $0 \leq e \leq k$ dan $j = 1, 2, \dots, j$ jumlah label, sebagai berikut :

$$P(E = 0|H_1 = 1) = \frac{s + Z(0.1)}{(k + 1)s + Z(0.1) + Z(1.1) + Z(2.1)}$$

$$= \frac{0.1 + 2}{0.3 + 2 + 0 + 0} = \mathbf{0.9}$$

$$P(E = 0|H_1 = 0) = \frac{s + \bar{Z}(0.1)}{(k + 1)s + \bar{Z}(0.1) + \bar{Z}(1.1) + \bar{Z}(2.1)}$$

$$= \frac{0.1 + 0}{0.3 + 0 + 0 + 0} = \mathbf{0.3}$$

Berdasarkan tahapan yang sama maka didapatkan seluruh nilai *likelihood* seperti yang ditunjukkan pada Tabel 3.19 :

Tabel 3.19 Hasil Penentuan Nilai Likelihood $P(E|H_j)$

	$P(E = e H_j = 1)$			$P(E = e H_j = 0)$		
	$e = 0$	$e = 1$	$e = 2$	$e = 0$	$e = 1$	$e = 2$
$j = 1$	0.969	0.023	0.023	0.333	0.333	0.333
$j = 2$	0.846	0.077	0.077	0.939	0.030	0.030
$j = 3$	0.043	0.913	0.043	0.913	0.043	0.043
$j = 4$	0.488	0.488	0.023	0.333	0.333	0.333

Tabel 3.22 Hasil Penentuan Nilai $\text{sim}(d, c_j)$ dan $\mu_{c_j}(d^t)$ untuk data uji

Aspek Kualitas	$\text{si}(d, c_j)$			$\mu_{c_j}(d^t)$		
	KU ₁	KU ₂	KU ₃	KU ₁	KU ₂	KU ₃
c ₁	0	0	0	0	0	0
c ₂	0	0	0	0	0	0
c ₃	0.167	0	0	1	0	0
c ₄	0.096	0.009	0	0.575	1	0

2. Pengecekan $\mu_{c_j}(d^t)$ berdasarkan *threshold* dan menemukan *search set*

Dari hasil $\mu_{c_j}(d^t)$ yang didapatkan pada Tabel 3.22 maka akan dilakukan pengecekan dengan nilai *threshold*. Nilai disesuaikan seperti ketika ilustrasi pada tahap pelatihan yaitu 0.7, sehingga diketahui bahwa kalimat - kalimat uji diatas memiliki *search set* seperti yang ditunjukkan pada Tabel 3.23.

Tabel 3.23 Data Search Set untuk Kalimat Uji

Kalimat Uji	Search Set (G_i)
KU1	S3
KU2	S4
KU3	\emptyset

3. Penentuan tetangga terdekat dari dokumen uji (N^i) dan nilai vektor jumlah label (n^i)

Dari hasil search set yang didapatkan pada proses sebelumnya maka dapat ditentukan tetangga terdekat N^i dari kalimat uji dan nilai n^i seperti pada Tabel 3.24.

Tabel 3.24 Data tetangga terdekat dari dokumen uji (N^i) dan nilai vektor jumlah label (n^i) untuk Kalimat Uji

Kalimat Uji	Tetangga Terdekat (N^i)	Vektor Jumlah Label (n^i)
KU1	KK2	1,0,1,0
KU2	KK4	1,0,1,0
KU3	\emptyset	0,0,0,0

4. Penentuan nilai MAP untuk menentukan aspek kualitas pada dokumen uji.

Penentuan nilai MAP dilakukan berdasarkan persamaan 2.27 dengan memanfaatkan data klasifier yang ditunjukkan pada Tabel 3.16 dan Tabel 3.19.

Berikut hasil yang didapatkan dari proses penentuan nilai MAP untuk menentukan aspek kualitas dokumen uji :

$$P(H_1 = 1)P(E = 2|H_1 = 1) = 0.976 \times 0.969 = \mathbf{0.96}$$

$$P(H_1 = 0)P(E = 2|H_1 = 0) = 0.024 \times 0.333 = \mathbf{0.008}$$

Berdasarkan persamaan 2.23 karena $0,0946 > 0,008$ maka nilai $y_1^t = \mathbf{1}$ yang menandakan bahwa dokumen uji d_t termasuk kedalam aspek kualitas c_1 , dengan cara yang sama didapatkan pula nilai MAP lainnya sebagai berikut :

$$P(H_2 = 1)P(E = 2|H_2 = 1) = 0.024 \times 0.846 = \mathbf{0.0203}$$

$$P(H_2 = 0)P(E = 2|H_2 = 0) = 0.976 \times 0.939 = \mathbf{0.916}$$

$$0.021 < 0.916 \text{ maka nilai } y_2^t = \mathbf{0}$$

$$P(H_3 = 1)P(E = 0|H_3 = 1) = 0.738 \times 0.043 = \mathbf{0.032}$$

$$P(H_3 = 0)P(E = 0|H_3 = 0) = 0.262 \times 0.913 = \mathbf{0.239}$$

$$0.032 < 0.239 \text{ maka nilai } y_3^t = \mathbf{0}$$

$$P(H_4 = 1)P(E = 0|H_4 = 1) = 0.262 \times 0.488 = \mathbf{0.128}$$

$$P(H_4 = 0)P(E = 0|H_4 = 0) = 0.738 \times 0.333 = \mathbf{0.246}$$

$$0,128 < 0,246 \text{ maka nilai } y_4^t = \mathbf{0}$$

Dari hasil diatas dapat disimpulkan bahwa dokumen uji d_t termasuk kedalam aspek kualitas c_1 yaitu *functional correctness*.

3.3 Pembuatan Perangkat Lunak

Pada tahapan ini dilakukan implementasi dari desain sistem ke dalam kode program yang dapat dimengerti oleh komputer. Sistem yang dikembangkan dalam bentuk *desktop base* dengan bahasa pemrograman Java dan Python. Sedangkan untuk penyimpanan data pelatihan dan data uji menggunakan Basis Data MySQL dan Excell sehingga data yang digunakan untuk semua proses dapat diakses.

3.4 Uji Coba Sistem

Pada penelitian ini dilakukan pengujian perbandingan persentase *accuracy*, *precision*, *recall*, dan F1 antara data latih yang diperoleh dengan mempertimbangkan faktor semantik dengan data latih yang diperoleh tanpa mempertimbangkan faktor semantik. Tujuan dari pengujian ini adalah untuk mengetahui bahwa penambahan pengukuran semantik pada data latih yang diperoleh dapat meningkatkan kinerja klasifikasi. Disamping itu, penggunaan

hipernim dibatasi hanya 1 level. Hal ini dikarenakan jika penggunaan hipernim terlalu banyak dikhawatirkan akan menghilangkan makna asli dari *term* yang dikembangkan (Ramadhani, Rochimah and Yuhana, 2015).

Permasalahan lain apabila tidak ada batasan dalam penggunaan hipernim maka akan berdampak pada penurunan kinerja klasifikasi. Menurut Rozi dkk (2015) (Rozi, Faticah and Purwitasari, 2015) penggunaan hipernim mampu meningkatkan akurasi. Namun, penggunaan hipernim dalam beberapa dataset hanya akan menambah jumlah term yang akan dianggap sebagai noise sehingga mengurangi akurasi. Untuk membuktikan hipotesa ini maka pengembangan *term* gabungan dilakukan menggunakan dua skenario. Skenario pertama menggunakan 2 level hipernim dan 9 sinonim pada masing-masing hipernim seperti penelitian sebelumnya (Ramadhani, Rochimah and Yuhana, 2015). Sedangkan skenario kedua menggunakan 1 level hipernim dan 20 sinonim.

Uji coba ini, dilakukan terhadap hasil klasifikasi kalimat kebutuhan berdasarkan ISO/IEC 25010:2011. Hasil uji coba selanjutnya, akan diperiksa oleh tiga orang pakar di bidang rekayasa kebutuhan dan kualitas perangkat lunak, yang berprofesi sebagai dosen pengampu mata kuliah Rekayasa Perangkat Lunak dari Jurusan Teknik Informatika. Sedangkan untuk mengetahui kehandalan sistem yang dibangun, dapat dilakukan dengan cara membandingkan pengetahuan yang dimiliki oleh tiga pakar dengan keluaran yang dihasilkan oleh metode yang diusulkan. Pakar tersebut bertindak sebagai penguji pertama, sedangkan metode yang diusulkan bertindak sebagai penguji kedua. Analisis terhadap hasil pengujian dilakukan dengan menghitung nilai indeks statistik *Gwet's AC1*.

Adapun skenario uji coba yang dilakukan pada penelitian ini dapat diuraikan melalui beberapa tahapan berikut:

- 1) Menyediakan definisi standar kualitas ISO/IEC yang sudah ada. Definisi tersebut akan menjadi dasar bagi ahli untuk melakukan penilaian,
- 2) Memetakan hubungan kalimat kebutuhan dengan standar kualitas ISO/IEC 25010:2011, dengan menghitung kemiripan teks pada kalimat kebutuhan dan standar kualitas,
- 3) Membandingkan hasil nilai keluaran yang diperoleh dari tiga orang pakar dan metode yang diusulkan. Perbandingan nilai keluaran ini, digunakan

untuk melihat seberapa *reliable* hasil yang diperoleh dari metode yang diusulkan.

Sesuai pada penjelasan dalam sub-bab 2.8, dan 2.9, pada penelitian ini dilakukan dua model pengujian. Pengujian pertama pengukuran kesepakatan antara tiga pakar menggunakan gabungan dari Cohen Kappa dan Gwet's AC1. Kesepakatan ketiga pakar tersebut akan menjadi rujukan dari pelabelan data latih yang peroleh oleh metode yang diusulkan. Pengujian kedua pengukuran menggunakan *accuracy*, *precision*, *recall*, dan F1 antara pelabelan yang didapatkan dari kesepakatan ketiga pakar dengan data latih yang dihasikalkan oleh metode yang diusulkan.

BAB 4

HASIL DAN PEMBAHASAN

Pada pembahasan ini diberikan pemaparan mengenai implementasi sistem serta pengujian dari sistem berdasarkan skenario yang telah dirancang pada Bab Tiga. Proses implementasi dilakukan berdasarkan tahapan yang telah diberikan pada pembahasan sebelumnya. Selanjutnya pengujian sistem dilakukan dengan beberapa kondisi yang disesuaikan dengan skenario pengujian. Dari hasil pengujian yang telah didapatkan, selanjutnya diberikan pembahasan dan analisa dari setiap pengujian yang dilakukan dengan tujuan untuk mendapatkan hasil dari penelitian, sehingga mendapatkan kesimpulan yang diberikan pada pembahasan selanjutnya.

4.1 Lingkungan Pengujian

Lingkungan pengujian pada penelitian ini meliputi spesifikasi perangkat keras ditunjukkan dalam Tabel 4.1 dan spesifikasi perangkat lunak ditunjukkan dalam Tabel 4.2.

Tabel 4.1 Spesifikasi perangkat keras

No	Nama	Spesifikasi
1	Processor	Intel(R) Core(TM) 2 Duo CPU P7570 @2.26 GHz
2	Memori	4.00 GB
3	Harddisk	500 GB
4	VGA	Mobile Intel(R) 4 Series Express Chipset Family

Tabel 4.2 Spesifikasi perangkat lunak

No	Nama	Spesifikasi
1	Sistem Operasi	Windows 7 Ultimate 64-bit (6.1, Build 7600)
2	Development	Java NetBeans 8.2, JDK 8
3	Library	Apache Lucene 4.8.3, ritaWordnet, Apache POI, MIT JWI
4	Database	MySQL
5	Web Server	XAMPP

4.2 Skenario Pengujian

Pengujian dilakukan menggunakan dataset *voip* dan *geolocation* dengan jumlah 1366 kalimat kebutuhan. Kedua dataset tersebut terdiri dari 1141 kebutuhan fungsional dan 225 kebutuhan non-fungsional (Cleland-Huang *et al.*, 2006). Dimana 1366 kalimat kebutuhan tersebut akan dilabeli secara otomatis dengan tahapan yang telah dijelaskan pada bab sebelumnya, dari 1366 kalimat akan diambil 60% (820 kalimat) untuk digunakan sebagai data latih dan 40% (546 kalimat) digunakan untuk data uji. Untuk mengetahui kinerja dari masing - masing metode akan dilakukan proses evaluasi dengan pengukuran *accuracy*, *precision*, *recall*, dan F1. Berdasarkan hasil evaluasi tersebut nantinya akan diketahui kinerja klasifikasi terbaik dari ketiga skenario.

Skenario pengujian dilakukan dengan cara menjalankan sistem menggunakan dua data latih berbeda yaitu data latih yang digunakan menggunakan 2 level hipernim dan 9 sinonim pada masing-masing hipernim disebut skenario 1, dan data latih yang digunakan menggunakan 1 level hipernim dan 20 sinonim disebut skenario 2. Kedua data latih tersebut akan dievaluasi berdasarkan *ground truth* yang diperoleh dari kesepakatan ketiga pakar. Tujuan dari evaluasi ini untuk mengukur seberapa baik metode *unsupervised* yang digunakan untuk melakukan pelabelan secara otomatis. Kedua data latih tersebut akan dievaluasi menggunakan sejumlah rentang nilai yang dihasilkan sistem berupa *threshold* yaitu 0.1 hingga 0.7. Hasil evaluasi ini akan digunakan untuk menentukan nilai *threshold* yang seimbang antara keempat metode evaluasi. Nilai *threshold* terpilih akan digunakan pada proses klasifikasi.

Hasil pengukuran semantik antara *term-term* relevan setiap aspek kualitas dengan *term-term* pada dokumen ditambahkan pada data latih yang diperoleh dari skenario 2. Data latih dengan penambahan semantik disebut skenario 3. Selanjutnya, ketiga data latih tersebut akan dilakukan perbandingan kinerja klasifikasi menggunakan metode FSKNN. Metode klasifikasi yang digunakan menggunakan parameter jumlah tetangga terdekat (k) 10, 20, dan 30 dengan nilai *threshold* 0.5. Pengujian ini dilakukan untuk mengetahui pengaruh penambahan faktor semantik pada data latih yang diperoleh pada tahap otomatisasi data

pelatihan dan juga untuk mengetahui pengaruh pengembangan term gabungan antara 1 level hipernim dan 2 level hipernim.

Hasil klasifikasi dari ketiga skenario akan diukur menggunakan empat evaluasi metrik yaitu *accuracy*, *precision*, *recall*, dan F1. Pengukuran dilakukan menggunakan dua model. Model pertama menggunakan *ground truth*. Sedangkan model kedua menggunakan hasil pelabelan dari metode *unsupervised*.

4.3 Evaluasi Hasil Pengujian

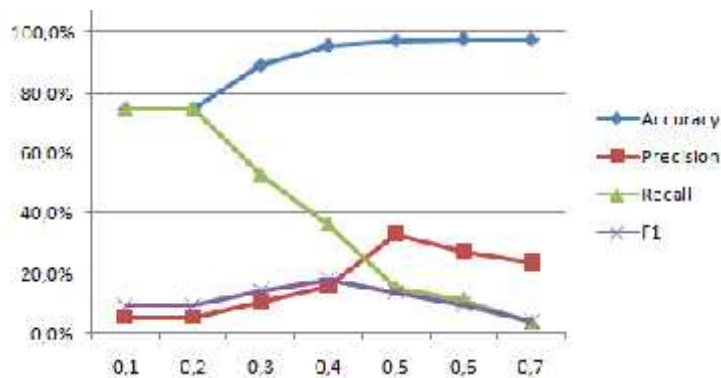
Pada subbab ini akan dibahas hasil pengujian dan evaluasi hasil pengujian dari setiap skenario yang telah dibahas sebelumnya. Data pengujian untuk keperluan evaluasi sistem diambil dari data latih yang terlabeli sebanyak 546 kalimat kebutuhan yang akan diklasifikasi ke dalam 31 aspek kualitas. Hasil pengujian yang peroleh dari rentang threshold 0.1 hingga 0.7 pada data latih yang dihasilkan secara otomatis dievaluasi menggunakan *accuracy*, *precision*, *recall*, dan F1. Begitu juga dengan hasil pengujian menggunakan metode klasifikasi FSKNN dengan jumlah tetangga terdekat (k) 10, 20, dan 30.

4.3.1 Evaluasi Hasil Pengujian Skenario 1

Nilai tertinggi evaluasi metrik precision pada skenario 1 terdapat pada nilai *threshold* 0.5. Akan tetapi berdasarkan keempat evaluasi metrik disimpulkan bahwa *threshold* 0.5 merupakan nilai yang memiliki tingkat keseimbangan tertinggi. Evaluasi hasil pengujian evaluasi pada skenario 1 berdasarkan *ground truth* pakar ditunjukkan pada Tabel 4.3 dan Gambar 4.1. Sedangkan hasil evaluasi menggunakan metode klasifikasi FSKNN dengan nilai (k) 10, 20, dan 30 ditunjukkan pada Tabel 4.4 dan Gambar 4.2.

Tabel 4.3 Tabulasi rata-rata kinerja *accuracy*, *precision*, *recall*, dan F1 berdasarkan *ground truth* pada skenario 1

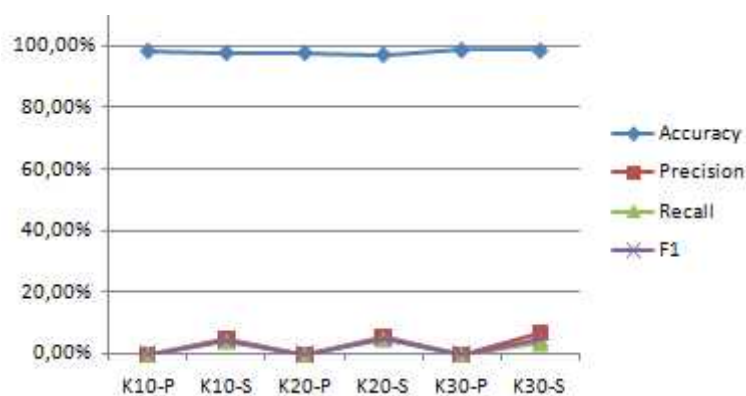
Threshold	Skenario 1 2 Level Hipernim @9 Sinonim			
	<i>Accuracy</i>	<i>Precision</i>	<i>Recall</i>	<i>F1</i>
0.1	74.5%	5.4%	74.7%	9.2%
0.2	74.5%	5.4%	74.7%	9.2%
0.3	89.1%	10.5%	52.7%	14.4%
0.4	95.6%	15.8%	36.4%	18.0%
0.5	97.3%	33.1%	15.3%	14.0%
0.6	97.7%	27.1%	11.1%	9.7%
0.7	97.6%	23.5%	3.9%	4.2%



Gambar 4.1 Grafik rata-rata kinerja *accuracy*, *precision*, *recall*, dan *F1* berdasarkan *ground truth* pada skenario 1

Tabel 4.4 Tabulasi rata-rata kinerja *accuracy*, *precision*, *recall*, dan *F1* menggunakan metode FSKNN pada skenario 1

K	Threshold	Skenario 1 FSKNN			
		<i>Accuracy</i>	<i>Precision</i>	<i>Recall</i>	<i>F1</i>
10	Pakar	98.32%	0.00%	0.00%	0.00%
	<i>Unsupervised</i>	97.72%	5.06%	4.34%	4.62%
20	Pakar	97.73%	0.00%	0.00%	0.00%
	<i>Unsupervised</i>	97.16%	5.81%	5.03%	5.11%
30	Pakar	98.58%	0.00%	0.00%	0.00%
	<i>Unsupervised</i>	98.41%	6.94%	3.72%	4.83%



Gambar 4.2 Grafik tabulasi rata-rata kinerja *accuracy*, *precision*, *recall*, dan *F1* menggunakan metode FSKNN pada skenario 1

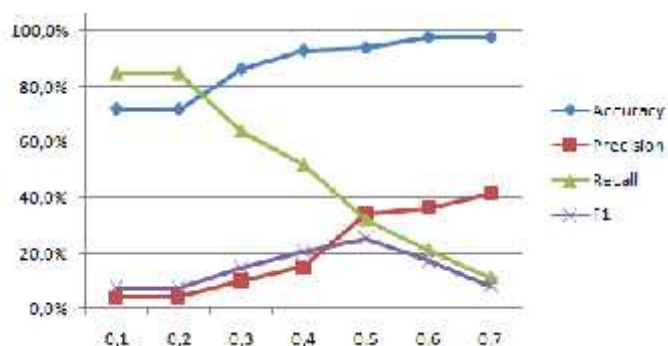
4.3.2 Evaluasi Hasil Pengujian Skenario 2

Nilai tertinggi evaluasi metrik *precision* pada skenario 2 terdapat pada nilai *threshold* 0.7. Akan tetapi berdasarkan keempat evaluasi metrik disimpulkan bahwa *threshold* 0.5 merupakan nilai yang memiliki tingkat keseimbangan

tertinggi. Evaluasi hasil pengujian evaluasi pada skenario 2 berdasarkan *ground truth* pakar ditunjukkan pada Tabel 4.5 dan Gambar 4.3. Sedangkan hasil evaluasi menggunakan metode klasifikasi FSKNN dengan nilai (k) 10, 20, dan 30 ditunjukkan pada Tabel 4.6 dan Gambar 4.4.

Tabel 4.5 Tabulasi rata-rata kinerja *accuracy*, *precision*, *recall*, dan F1 berdasarkan *ground truth* pada skenario 2

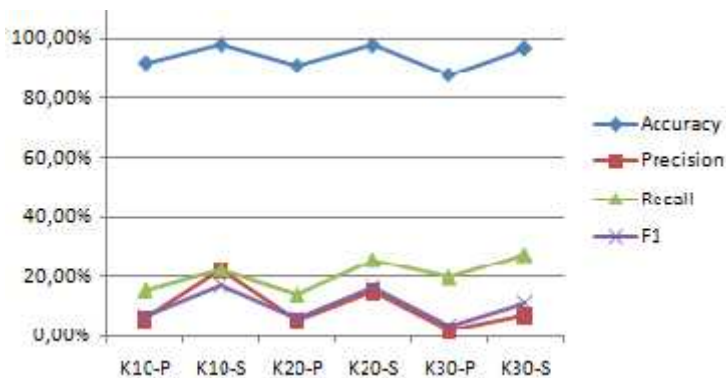
Threshold	Skenario 2 1 Level Hipernim @20 Sinonim			
	<i>Accuracy</i>	<i>Precision</i>	<i>Recall</i>	<i>F1</i>
0.1	71.8%	4.1%	84.9%	7.6%
0.2	71.8%	4.1%	84.9%	7.6%
0.3	86.2%	10.0%	64.1%	14.79%
0.4	93.0%	15.2%	51.8%	20.3%
0.5	94.0%	34.3%	32.1%	25.2%
0.6	97.8%	36.3%	21.3%	17.2%
0.7	97.8%	41.6%	11.3%	8.4%



Gambar 4.3 Grafik rata-rata kinerja *accuracy*, *precision*, *recall*, dan F1 berdasarkan *ground truth* pada skenario 2

Tabel 4.6 Tabulasi rata-rata kinerja *accuracy*, *precision*, *recall*, dan F1 menggunakan metode FSKNN pada skenario 2

K	Threshold 0.5	Skenario 2 FSKNN			
		<i>Accuracy</i>	<i>Precision</i>	<i>Recall</i>	<i>F1</i>
10	Pakar	91.97%	6.15%	15.63%	6.79%
	Unsupervised	98.00%	22.55%	22.65%	17.46%
20	Pakar	91.20%	5.65%	14.23%	5.98%
	Unsupervised	97.84%	15.10%	25.90%	17.09%
30	Pakar	87.92%	2.12%	19.86%	3.64%
	Unsupervised	96.89%	7.55%	27.64%	11.77%



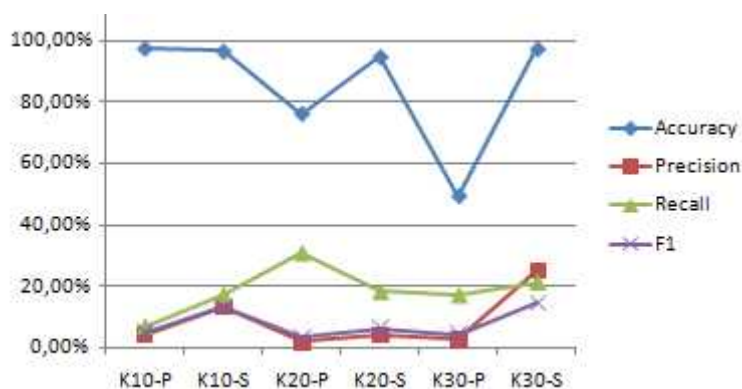
Gambar 4.4 Grafik tabulasi rata-rata kinerja *accuracy*, *precision*, *recall*, dan F1 menggunakan metode FSKNN pada skenario 2

4.3.3 Evaluasi Hasil Pengujian Skenario 3

Hasil evaluasi pada data latih dengan penambahan pengukuran semantik menggunakan metode klasifikasi FSKNN dengan nilai (k) 10, 20, dan 30 ditunjukkan pada Tabel 4.7 dan Gambar 4.5.

Tabel 4.7 Tabulasi rata-rata kinerja *accuracy*, *precision*, *recall*, dan F1 menggunakan metode FSKNN pada skenario 3

K	Threshold	Skenario 3 FSKNN			
		<i>Accuracy</i>	<i>Precision</i>	<i>Recall</i>	<i>F1</i>
10	Pakar	97.25%	4.17%	6.67%	5.13%
	Unsupervised	96.33%	13.42%	17.09%	13.45%
20	Pakar	76.13%	1.88%	30.86%	3.48%
	Unsupervised	94.55%	4.18%	18.15%	6.35%
30	Pakar	49.45%	2.74%	16.91%	4.35%
	Unsupervised	97.10%	25.05%	21.35%	14.73%



Gambar 4.5 Grafik tabulasi rata-rata kinerja *accuracy*, *precision*, *recall*, dan F1 menggunakan metode FSKNN pada skenario 3

4.4 Implementasi Sistem

Penelitian ini menggunakan standar ISO/IEC 25010:2011. Standar ISO/IEC 25010:2011 digunakan sebagai acuan untuk melakukan pelabelan pada 1366 kalimat kebutuhan. Tidak adanya standar baku untuk menentukan pelabelan menjadi permasalahan tersendiri. Oleh karena itu, proses pelabelan ditentukan menggunakan metode *unsupervised*. Sehingga untuk mengevaluasi performa dari metode yang dikembangkan, maka penelitian ini dibantu oleh beberapa pakar untuk melabeli kalimat kebutuhan yang diproses. Adapun proses anotasi yang dilakukan adalah dengan melakukan pelabelan sejumlah 225 kalimat kebutuhan yang sudah diidentifikasi non-fungsional oleh peneliti sebelumnya (Cleland-Huang *et al.*, 2006). Pelabelan pada 225 kalimat kebutuhan tersebut dilakukan oleh tiga pakar. Setiap kalimat kebutuhan bisa dilabeli lebih dari satu aspek kualitas.

Tabel 4.8 Komposisi Kelas dari 225 Kalimat Kebutuhan

Keputusan	No.	Aspek Kualitas	SubTotal NFRs
3 Pakar Sepakat	1	<i>Functional completeness</i>	13
	2	<i>Functional correctness</i>	31
	3	<i>Functional appropriateness</i>	13
	4	<i>Time behaviour</i>	31
	5	<i>Resource utilization</i>	28
	6	<i>Capacity</i>	3
	7	<i>Co-existence</i>	2
	8	<i>Interoperability</i>	2
	9	<i>Appropriateness recognizability</i>	26
	10	<i>Learnability</i>	2
	11	<i>Operability</i>	15
	12	<i>User error protection</i>	5
	13	<i>User interface aesthetics</i>	4
	14	<i>Accessibility</i>	7
	15	<i>Maturity</i>	3
	16	<i>Availability</i>	2
	17	<i>Fault tolerance</i>	7
	18	<i>Recoverability</i>	6
	19	<i>Confidentiality</i>	7
	20	<i>Integrity</i>	15
	21	<i>Non-repudiation</i>	8

Keputusan	No.	Aspek Kualitas	SubTotal NFRs
	22	<i>Accountability</i>	6
	23	<i>Authenticity</i>	11
	24	<i>Modularity</i>	4
	25	<i>Reusability</i>	19
	26	<i>Analysability</i>	5
	27	<i>Modifiability</i>	8
	28	<i>Testability</i>	16
	29	<i>Adaptability</i>	19
	30	<i>Installability</i>	2
	31	<i>Replaceability</i>	3
		Total	323
2 Pakar Sepakat	1	<i>Functional completeness</i>	1
	2	<i>Functional correctness</i>	5
	3	<i>Functional appropriateness</i>	2
	4	<i>Time behaviour</i>	14
	5	<i>Resource utilization</i>	12
	6	<i>Capacity</i>	6
	7	<i>Co-existence</i>	2
	8	<i>Interoperability</i>	1
	9	<i>Appropriateness recognizability</i>	1
	10	<i>Learnability</i>	5
	11	<i>Operability</i>	6
	12	<i>User error protection</i>	1
	13	<i>User interface aesthetics</i>	5
	14	<i>Accessibility</i>	0
	15	<i>Maturity</i>	5
	16	<i>Availability</i>	6
	17	<i>Fault tolerance</i>	5
	18	<i>Recoverability</i>	2
	19	<i>Confidentiality</i>	10
	20	<i>Integrity</i>	3
	21	<i>Non-repudiation</i>	1
	22	<i>Accountability</i>	1
	23	<i>Authenticity</i>	5
	24	<i>Modularity</i>	4
	25	<i>Reusability</i>	1
	26	<i>Analysability</i>	2
	27	<i>Modifiability</i>	3
	28	<i>Testability</i>	7

Keputusan	No.	Aspek Kualitas	SubTotal NFRs
	29	<i>Adaptability</i>	4
	30	<i>Installability</i>	3
	31	<i>Replaceability</i>	5
Total			128

Berdasarkan Tabel 4.8, penulis hendak membandingkan apakah pelabelan otomatis yang dilakukan oleh sistem sudah *reliable*. Pelabelan otomatis dilakukan menggunakan metode *unsupervised* dengan dua skenario yaitu 2 level hipernim, dan 1 level hipernim. Dalam proses pelabelan, kalimat kebutuhan dicocokkan dengan hasil anotasi oleh pakar yang bertugas untuk membuat *ground truth* setiap kalimat kebutuhan.

Adapun pada implementasinya, kalimat kebutuhan dilakukan praproses terlebih dulu. Praproses terdiri dari tokenisasi, stopword, dan stemming. Kontribusi utama pada penelitian ini yaitu menambahkan pengukuran keterkaitan semantik pada data pelatihan yang diperoleh pada proses otomatisasi pelabelan.

Berikut tahapan - tahapan prosesnya sesuai dengan desain sistem pada subbab 3.2:

1. Fase Otomatisasi Pelabelan Data Latih

a. Praproses

Pada fase ini 1366 kalimat kebutuhan akan dilakukan praproses untuk mendapatkan *term* atau kata dari setiap kalimat kebutuhan yang akan digunakan sebagai data latih. Hasil dari tahap praproses melalui sistem dapat ditunjukkan pada Gambar 4.1 dibawah ini.

```
[grant] [administr] [access] [through] [system] [consol]
[administr] [can] [find] [avail] [phone] [number]
[administr] [can] [find] [avail] [ip] [address]
[administr] [can] [associ] [ip] [address] [phone] [number]
[administr] [can] [map] [ip] [address] [phone] [number] [custom] [account]
[grant] [account] [abil] [origin] [receiv] [call]
[administr] [can] [block] [incom] [call] [account]
[administr] [can] [exclus] [allow] [incom] [call] [account]
```

Gambar 4.6 Potongan Praproses dalam sistem

b. Pengembangan *Term* Gabungan antara Hipernim dan Sinonim

Pada fase ini seluruh *term* relevan pada Tabel 2.2 dikembangkan dengan menggunakan *term* gabungan antara hipernim dan sinonim. Keseluruhan *term* relevan berjumlah 210. Hasil pengembangan *term* tersebut dapat dilihat pada Gambar 4.2. Secara keseluruhan pengembangan *term* relevan dapat dilihat pada Lampiran 1.

hipernim level 1 : [commodity, good, creation, quantity, chemical,

postagging untuk kata commodity adalah n

sinonim level 1 : [instrumentality, extra, instrumentation, playth:

instrumentality extra instrumentation plaything sheet toy inset in:

hipernim level 2 : [basic, staple, drygoods, entrant, export, expo:

postagging untuk kata basic adalah a

sinonim level 2 : [base, fundamental, underlying, standard, alkali:

base fundamental underlying standard alkalic canonic primary grass:

Gambar 4.7 Pengembangan *term* pada sistem

c. Pembobotan TF-IDF

Menggunakan hasil yang diperoleh dari tahap praproses dari kalimat kebutuhan dan tahap pengembangan *term* relevan maka dapat dilakukan pembobotan TF-IDF melalui sistem dengan persamaan 2.24, 2.25, 2.26. Hasil pembobotan TF-IDF terdiri dari bobot TF-IDF untuk tiap-tiap *term* relevan dalam setiap aspek kualitas dan dalam setiap kalimat kebutuhan yang akan dilabeli sebagai data latih. Hasil pembobotan TF-IDF dapat dilihat pada Gambar 4.3.

TermRelevan	TFKualitas	TFReq	IDFReq	TFIDFKualitas	TFIDFReq
function	1	0	1.301	0.390	0
suitabl	1	0	1.301	0.390	0
complet	1	0	1.301	0.390	0
cover	1	0	1.301	0.390	0
specifi	1	0	1.301	0.390	0
task	1	1	1	0.299	1
object	1	0	1.301	0.390	0

Gambar 4.8 Hasil tahap pembobotan TF-IDF

d. Pengukuran Nilai Derajat Kemiripan

Setelah dilakukan tahapan pembobotan TF-IDF maka dapat dilakukan pengukuran nilai kemiripan antara suatu kalimat kebutuhan dengan suatu aspek kualitas dengan persamaan 2.27. Pengukuran nilai derajat kemiripan ini dilakukan terhadap 1366 kalimat kebutuhan dimana setiap kalimat kebutuhan dicari nilai derajat kemiripannya dengan 31 aspek kualitas berdasarkan pada Tabel 2.1. Dengan menerapkan nilai batas ambang tertentu terhadap nilai derajat kemiripan yang didapatkan maka akan dihasilkan 1366 kalimat kebutuhan yang terlabeli ke dalam 31 aspek kualitas. Hasil yang didapatkan dari pengukuran nilai derajat kemiripan melalui sistem dengan penerapan nilai ambang batas 0.1 adalah seperti yang ditunjukkan pada Gambar 4.4.

idKualitas	idReq	cosine Similarity
1	1	0
1	2	0
1	3	0
1	4	0
1	5	0.299
1	6	0.437
1	7	0.573
1	8	0
1	9	0
1	10	0.312

Gambar 4.9 Hasil pengukuran derajat kemiripan dalam database

2. Fase Pengukuran Keterkaitan Semantik

Pengukuran keterkaitan semantik dilakukan antara *term-term* relevan pada masing-masing aspek kualitas sejumlah 210 dengan *term-term* pada dokumen yang akan dilabeli. Proses pengukuran semantik dilakukan dengan menggunakan metode HSO (Hirst and St-onge, 1995). Hasil pengukuran keterkaitan semantik tersebut akan diterapkan pada proses otomatisasi data pelatihan. Contoh pengukuran keterkaitan semantik dapat dilihat pada Gambar 4.5.

```

+++++
Kalimat Kebutuhan Ke : 405
+++++
-----
TermRel ID :1 Term Relevan : function
-----
Term Kalimat 405: caller
Term Kalimat 405: can
Term Kalimat 405: place
Pos : n
HSO value : 0,4444
Term Kalimat 405: phone
Pos : n
HSO value : 0,2222
Term Kalimat 405: handset
Term Kalimat 405: hook
Term Kalimat 405: anytim
Term Kalimat 405: dure
Term Kalimat 405: comun
Term Kalimat 405: exchang
Term Kalimat 405: termin
Term Kalimat 405: audio

```

Gambar 4.10 Hasil Pengukuran Keterkaitan Semantik

3. Fase Pengelompokan Pola Pelatihan Ke Dalam Kluster

Mengacu pada penjelasan di subbab 3.2.3 maka fase ini terdiri dari beberapa tahap sebagai berikut :

- a. Penentuan distribusi $term_t$ pada aspek kualitas c_j

Dalam proses ini merupakan penentuan distribusi $term$ dan distribusi kalimat kebutuhan dari 210 $term$ relevan dengan 31 aspek kualitas.

Hasil penentuan tersebut dapat dilihat pada Gambar 4.6 dan Gambar 4.7.

```

PELATIHAN
Derajat dt
0.088 0.0 0.0 0.011 0.011 0.022 0.0 0.0 0.066 0.0 0.0 0.033
0.136 0.0 0.0 0.0 0.0 0.023 0.0 0,0 0.114 0.0 0.0 0.0
*****
0.075 0.0 0.0 0.0 0.0 0.0 0.0 0.015 0.0149 0.0 0.0 0.149

```

Gambar 4.11 Hasil Pada Tahap Penentuan Distribusi $Term d (t_i, c_j)$ Pada Apek

Kualitas c_j


```

Derajat dd
0.0 1.0 0.0 1.0 1.0 1.0 0.0 0.0 0.5 0.0 0.0 0.1 0.428
0.75 0.0 0.0 0.0 0.0 0.5 0.0 0.0 0.417 0.0 0.0 0.0 0.0
*****
0,625 0.0 0.0 0.0 0.0 0.0 0.0 1.0 0.083 0.0 0.0 0.333 0.143

```

Gambar 4.12 Hasil Pada Tahap Penentuan Distribusi $Term d (t_i, c_j)$ Pada Apek Kualitas c_j

b. Penentuan derajat keanggotaan $\mu_R(t_i, c_j)$ dan $\mu_d(t_i)$

Hasil dari penentuan distribusi $term t_i$ pada kategori c_j dapat digunakan untuk menentukan derajat keanggotaan $\mu_R(t_i, c_j)$ dan penentuan derajat keanggotaan $\mu_d(t_i)$ dilakukan terhadap 210 $term$ relevan dengan 31 kategori kualitas sedangkan penentuan derajat keanggotaan $\mu_d(t_i)$ dilakukan terhadap setiap data latih yaitu 820 kalimat kebutuhan dengan 210 $term$ relevan. Potongan hasil penentuan tersebut pada sistem dapat dilihat pada Gambar 4.8 dan Gambar 4.9.

```

PELATIHAN
Derajat ur
0.421 0.0 0.0 0.344 0.344 0.355 0.0 0.0 0.232 0.0 0.0 0.066 0.176 0.0
0.386 0.0 0.0 0.0 0.0 0.189 0.0 0.0 0.252 0.0 0.0 0.0 0.0 0.0
*****
0.421 0.0 0.0 0.344 0.344 0.355 0.0 0.0 0.232 0.0 0.0 0.088 0.176 0.0

```

Gambar 4.13 Hasil Pada Tahap Penentuan derajat keanggotaan $\mu_R(t_i, c_j)$

```

Derajat ud
0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.3 0.3 0.3 0.0 0.0 0.3 0.0 0.0 0.0
0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.3 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.5 0.0 0.0
*****
0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.3 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.5 0.0 0.0

```

Gambar 4.14 Hasil Pada Tahap Penentuan derajat keanggotaan $\mu_d(t_i)$

- c. Penentuan nilai similaritas $si(d, c_j)$ dengan *fuzzy similarity measure*
 Derajat keanggotaan $\mu_R(t_i, c_j)$ dan $\mu_d(t_i)$ yang telah didapatkan pada tahap sebelumnya dapat digunakan untuk penentuan nilai similaritas dengan *fuzzy similarity measure*. Penentuan nilai *fuzzy similarity measure* ini dilakukan antara setiap kalimat kebutuhan yang digunakan untuk data latih sebanyak 820 dengan tiap-tiap kategori kualitas sebanyak 31 kategori. Hasil penentuan *fuzzy similarity measure* dapat dilihat pada Gambar 4.10.

Derajat Fuzzy

0.021	0.004	0.004	0.001	0.010	0.015	0.010	0.010
0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
0.007	0.000	0.000	0.000	0.004	0.004	0.012	0.004
0.000	0.000	0.000	0.000	0.008	0.007	0.017	0.019
0.019	0.003	0.003	0.006	0.008	0.008	0.010	0.008
0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
0.004	0.000	0.000	0.000	0.000	0.000	0.004	0.000
0.014	0.002	0.002	0.001	0.003	0.005	0.006	0.003
0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000

Gambar 4.15 Hasil Pada Tahap Penentuan nilai similaritas $si(d, c_j)$

- d. Penentuan nilai derajat keanggotaan $\mu_{c_j}(d)$
 Setelah proses penentuan similaritas antara kalimat kebutuhan dengan aspek kualitas, tahapan selanjutnya adalah menentukan derajat keanggotaan antara kalimat kebutuhan dengan aspek kualitas $\mu_{c_j}(d)$. Penentuan nilai derajat keanggotaan $\mu_{c_j}(d)$ dilakukan terhadap setiap kalimat kebutuhan yang digunakan sebagai data latih sebanyak 820 kalimat kebutuhan dengan tiap – tiap aspek kualitas sebanyak 31 kategori. Hasil dari penentuan nilai derajat keanggotaan $\mu_{c_j}(d)$ dapat dilihat pada Gambar 4.11

Derajat UC												
0.681	0.167		0.149	0.089	0.602	1.0	0.527	0.535	0.795	0.087	0.608	0.868
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	0.0	0.0	0.0	0.0		0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.216	0.0	0.0	0.0	0.243	0.256		0.607	0.214	0.371	0.0	0.247	0.486
0.0	0.0	0.0	0.0	0.476	0.492	0.875	1.0	0.329	0.0	0.465	1	
0.595	0.105		0.094	0.744	0.471	0.511	0.515	0.419	1.0	0.338	0.651	0.972

Gambar 4.16 Hasil Pada Tahap Penentuan nilai derajat keanggotaan $\mu_{c_j}(d)$

- e. Pengelompokan data pelatihan ke dalam beberapa kluster

Hasil derajat keanggotaan $\mu_{c_j}(d)$ dapat digunakan untuk membentuk kluster atau kelompok kategori kualitas untuk setiap kalimat kebutuhan yang digunakan sebagai data latih dengan menggunakan nilai ambang batas tertentu. Pengelompokan ke dalam beberapa kluster berdasarkan nilai *threshold* () tertentu dengan proses pada *pseudo-code* yang telah dijelaskan pada Gambar 2.7. Pada pengujian khusus ini akan digunakan nilai = 0,04, sehingga kluster yang terbentuk adalah seperti yang ditunjukkan pada Gambar 4.12.

```

run:
KLUSTER
S1      KK367  KK397  KK398  KK409  KK410  KK421  KK437
S2
S3
S4      KK414  KK436  KK437  KK444  KK453  KK456  KK467
S5      KK183  KK381  KK413  KK573
S6      KK17   KK159  KK183  KK185  KK685
S7
S8      KK384  KK445  KK448  KK524  KK526  KK546  KK549
S9      KK440  KK444  KK445  KK448  KK524  KK658  KK707
S10
S11     KK524  KK526  KK546  KK549
S12     KK444  KK445  KK707
S13
S14     KK384  KK445
S15     KK524  KK658  KK707
S16
S17
S18
S19
S20
S21
S22     KK421  KK437  KK511  KK516  KK521

```

Gambar 4.17 Hasil Pada Tahap Pengelompokan data pelatihan ke dalam beberapa kluster

- f. Pencarian *search set* untuk setiap data pelatih dari setiap kluster

Proses pencarian *search set* untuk setiap data pelatihan dari setiap kluster juga berdasarkan pada *pseudo-code* yang dijelaskan pada Gambar 2.7. Hasil pencarian *search set* tersebut dapat dilihat pada Gambar 4.13.

```

SEARCH SET
G405 : S17 --> KK393  KK406      KK426
G406 : S17 --> KK393  KK405      KK426
G407 : ∅
G408 : ∅
G409 : S1US16 --> KK367  KK397  KK398  K
G410 : S1US16 --> KK367  KK397  KK398  K
G411 : S29 --> KK17      KK348  KK361
G412 : ∅
G413 : ∅
G414 : ∅
G415 : ∅
G416 : S29 --> KK17  KK348  KK361  KK380
G417 : S29 --> KK17  KK348  KK361  KK380
G418 : ∅
G419 : ∅
G420 : ∅
G421 : ∅
G422 : ∅

```

Gambar 4.18 Hasil Pada Tahap Pencarian *search set* untuk setiap data Pelatihan

4. Fase Penghitungan Prior Probability dan Nilai Likelihoods

- a. Penentuan nilai *prior probability*

Tahapan selanjutnya dalam proses pelatihan adalah menentukan nilai *prior probability*. Dalam proses penentuan nilai *prior probability* menggunakan nilai *smoothing constant* (s), seperti pada penelitian sebelumnya dalam pengujian khusus ini akan digunakan nilai $s = 0.1$. Hasil dari penentuan nilai *prior probability* dapat dilihat pada Gambar 4.14.

```

PRIOR PROBABILITY
H1 -> C1: 0.109753719   H0 -> C1: 0.890246281
H1 -> C2: 0.1         H0 -> C2: 0.9
H1 -> C3: 0.1         H0 -> C3: 0.9
H1 -> C4: 0.101219215 H0 -> C4: 0.898780785
H1 -> C5: 0.101219215 H0 -> C5: 0.898780785
H1 -> C6: 0.10243843  H0 -> C6: 0.89756157
H1 -> C7: 0.1         H0 -> C7: 0.9
H1 -> C8: 0.101219215 H0 -> C8: 0.898780785
H1 -> C9: 0.114630578 H0 -> C9: 0.885369422

```

Gambar 4.19 Hasil Pada Tahap Penentuan Nilai Prior Probability

- b. Penentuan jumlah tetangga terdekat (k), penentuan tetangga terdekat (N^i) dan vektor jumlah label (n^i)

Tahapan selanjutnya adalah menentukan jumlah tetangga terdekat (k), jumlah tetangga terdekat yang digunakan dalam pengujian khusus ini adalah 30. Setelah jumlah tetangga terdekat didefinisikan maka selanjutnya adalah mencari atau menentukan tetangga terdekat untuk setiap data pelatihan (N^i) dan menentukan nilai vektor jumlah label (n^i) seperti yang telah dicontohkan pada subbab 3.3.4. Potongan hasil dari penentuan jumlah tetangga terdekat (k) dapat dilihat pada Gambar 4.15.

```

Fun:
TETANGGA TERDEKAT
KK367 KK397 KK398 KK410 KK421 KK437 KK511 KK516 KK521 KK522 KK571 KK573

```

Gambar 4.20 Hasil Pada Tahap Penentuan jumlah k , (N^i), (n^i)

- c. Penentuan nilai $Z(e, j)$ dan $\bar{Z}(e, j)$

Tahapan selanjutnya adalah penentuan nilai $Z(e, j)$ dan $\bar{Z}(e, j)$, kedua nilai ini nantinya digunakan untuk penentuan nilai *likelihood* pada proses selanjutnya. Hasil dari penentuan nilai $Z(e, j)$ dan $\bar{Z}(e, j)$ dapat dilihat pada Gambar 4.16.

Penentuan Nilai Z dan Z absen						
	Z			Z absen		
	e=0	e=1	e=2	e=0	e=1	e=2
1	2	0	0	773	0	14
2	0	0	0	820	0	0
3	0	0	0	820	0	0
4	1	0	0	770	0	0
5	1	0	0	817	0	0
6	0	1	1	801	0	3
7	0	0	0	820	0	0
8	1	0	0	793	0	0
9	4	1	2	697	0	25
10	0	0	0	820	0	0

Gambar 4.21 Hasil Pada Tahap Penentuan nilai $Z(e, j)$ dan $\bar{Z}(e, j)$

d. Penentuan nilai *likelihoods* $P(E|H_j)$

Tahapan selanjutnya adalah menentukan nilai *likelihood* dengan menggunakan data yang telah didapatkan pada tahap penentuan prior probability dan penentuan vektor jumlah label maka dapat dihitung nilai *likelihoods* seperti pada Gambar 4.17.

Penentuan LIKELYHOODS						
	Z			Z absen		
	e=0	e=1	e=2	e=0	e=1	e=2
1	0.039	0	0	0.098	0	0.002
2	0	0	0	0.099	0	0
3	0	0	0	0.099	0	0
4	0.024	0	0	0.099	0	0
5	0.024	0	0	0.099	0	0
6	0	0.019	0.019	0.099	0	0.001
7	0	0	0	0.099	0	0
8	0.024	0	0	0.099	0	0
9	0.039	0.010	0.019	0.096	0	0.003
10	0	0	0	0.099	0	0

Gambar 4.22 Pada Tahap Penentuan nilai *likelihoods* $P(E|H_j)$

5. Fase Klasifikasi FSKNN

Dalam fase klasifikasi ini akan diuji dengan 3 kalimat yang memiliki tingkat keterhubungan dengan kualitas perangkat lunak yang berbeda – beda, yaitu dengan kalimat yang memiliki hubungan yang sangat dekat, dekat, dan jauh. Berikut kalimat yang akan digunakan sebagai data uji :

KU1. Shall update Decommuted_eng_mem_list with the values extracted from Current_HK_telem, Prior_HK_telem, and Sci_eng_data. [Kalimat Kebutuhan ke 1343 dari 1366]

KU2. The geolocation software shall follow the coding standards established by the MODIS project in MODIS Software Development Standards and Guidelines. [1344]

KU3. The geolocation software shall use all applicable mandatory SDPTK routines and any additional SDPTK routines that are useful. [1345]

a. Hasil Kalimat Uji 1

Dari langkah – langkah seperti yang telah dijelaskan pada subbab 3.2.5 dan eksperimen awal pada subbab 3.3.5 maka dapat ditentukan hasil klasifikasi dari sistem untuk kalimat uji KU1 seperti yang ditunjukkan pada Gambar 4.18.

```
MAPVALUE kalimat uji 1 dari hasil proses pengujian sebagai berikut
000000000000000000000000000000000000
Sehingga diketahui kalimat uji 1 masuk dalam aspek kualitas sebagai berikut :
x x x x x x x x x x x x x x x x Integrity x x x x x x Testability x x x
```

Gambar 4.23 Hasil Klasifikasi Kalimat Uji 1

b. Hasil Kalimat Uji 2

Dari langkah – langkah seperti yang telah dijelaskan pada subbab 3.2.5 dan eksperimen awal pada subbab 3.3.5 maka dapat ditentukan hasil klasifikasi dari sistem untuk kalimat uji KU2 seperti yang ditunjukkan pada Gambar 4.19.

```
MAPVALUE kalimat uji 2 dari hasil proses pengujian sebagai berikut
000000000000010011000000000000000000
Sehingga diketahui kalimat uji 2 masuk dalam aspek kualitas sebagai berikut :
x x x x x x x x x x UserErrorProtection x x Maturity Availability x x x x x x
```

Gambar 4. 24 Hasil Klasifikasi Kalimat Uji 2

c. Hasil Kalimat Uji 3

Dari langkah – langkah seperti yang telah dijelaskan pada subbab 3.2.5 dan eksperimen awal pada subbab 3.3.5 maka dapat ditentukan hasil

BAB 5

KESIMPULAN DAN SARAN

Pada bab ini akan membahas kesimpulan dari penelitian yang telah dilakukan dan memberikan saran untuk penelitian selanjutnya berdasarkan hasil pengujian dan evaluasi pada bab sebelumnya.

5.1 Kesimpulan

Dari hasil penelitian yang telah dilakukan maka dapat disimpulkan beberapa hal sebagai berikut :

1. Data latih dapat dihasilkan secara otomatis dengan menggunakan pendekatan *unsupervised* berdasarkan aspek kualitas ISO/IEC 25010:2011. Data latih didapatkan dengan mempertimbangkan dua pengembangan *term* yaitu menggunakan 2 level hipernim dengan 9 sinonim dan pengembangan *term* menggunakan 1 level hipernim dengan 20 sinonim.
2. Penggunaan 1 level hipernim dan 20 sinonim pada fase otomatisasi data pelatihan terbukti dapat meningkatkan akurasi berdasarkan *ground truth* dari pakar.
3. Penambahan semantik dilakukan dengan menggunakan metode HSO. Perbandingan antara *term-term* relevan aspek kualitas dengan term-term pada dokumen akan menghasilkan jumlah *term* yang memiliki keterkaitan semantik. Jumlah tersebut akan ditambahkan pada data latih yang diperoleh pada skenario 2.
4. Penambahan pengukuran semantik pada data latih yang diperoleh pada skenario 2 terbukti dapat meningkatkan kinerja klasifikasi.
5. Pendekatan yang disulkan yaitu skenario 2 dan skenario 3 terbukti lebih baik dibandingkan skenario 1 untuk keempat evaluasi metrik baik menggunakan *ground truth* pakar maupun sistem.
6. Penggunaan dataset dengan jumlah kelas yang tidak imbang membuat nilai evaluasi untuk *precision* dan *recall* rendah pada semua pengujian.
7. Karena jumlah kebutuhan non-fungsional lebih sedikit dibandingkan kebutuhan fungsional membuat evaluasi metrik *precision* menjadi tolak

ukur untuk menentukan kualitas sistem. Semakin tinggi nilai *precision* maka semakin baik pendekatan yang diusulkan.

5.2 Saran

Sistem klasifikasi kebutuhan non-fungsional ini menggunakan dataset dengan jumlah kelas tidak seimbang yaitu 1141 fungsional dan 225 non-fungsional. Ketidakseimbangan tersebut berdampak pada kinerja klasifikasi yang sangat rendah baik menggunakan pendekatan *unsupervised* maupun *supervised*. Penelitian dapat dikembangkan dengan menambahkan metode untuk mengatasi ketidakseimbangan data.

DAFTAR PUSTAKA

- Al-Kilidar, H., Cox, K. and Kitchenham, B. (2005) 'The use and usefulness of the ISO/IEC 9126 quality standard', *2005 International Symposium on Empirical Software Engineering, ISESE 2005*, pp. 126–132. doi: 10.1109/ISESE.2005.1541821.
- Al-Qutaish, R. E. (2010) 'Quality models in software engineering literature: an analytical and comparative study', *Journal of American Science*, 6(3), pp. 166–175.
- Behkamal, B., Kahani, M. and Akbari, M. K. (2009) 'Customizing ISO 9126 quality model for evaluation of B2B applications', *Information and Software Technology*. Elsevier B.V., 51(3), pp. 599–609. doi: 10.1016/j.infsof.2008.08.001.
- Budanitsky, A. and Hirst, G. (1998) 'Semantic distance in WordNet: An experimental , application-oriented evaluation of five measures'.
- Casamayor, A., Godoy, D. and Campo, M. (2010) 'Identification of non-functional requirements in textual specifications: A semi-supervised learning approach', *Information and Software Technology*. Elsevier B.V., 52(4), pp. 436–445. doi: 10.1016/j.infsof.2009.10.010.
- Çelik, K. (2013) 'A Comprehensive Analysis of using Semantic Information in Text Categorization'.
- Chen, C., Tseng, F. S. C. and Liang, T. (2010) 'Data & Knowledge Engineering An integration of WordNet and fuzzy association rule mining for multi-label document clustering', *DATAK*. Elsevier B.V., 69(11), pp. 1208–1226. doi: 10.1016/j.datak.2010.08.003.
- Christiane, F. (1998) 'Applications of Wordnet', in *WordNet: An Electronic Lexical Database*. MITP. Available at: <https://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=6285383>.
- Chung, L. and Leite, J. D. P. (2009) 'On Non-Functional Requirements in Software Engineering', *Conceptual modeling: Foundations and ...*, pp. 363–379. doi: 10.1007/978-3-642-02463-4_19.

- Cleland-Huang, J. *et al.* (2006) 'The detection and classification of non-functional requirements with application to early aspects', *Proceedings of the IEEE International Conference on Requirements Engineering*, pp. 36–45. doi: 10.1109/RE.2006.65.
- Febrero, F., Calero, C. and Moraga, M. Á. (2016) 'Software reliability modeling based on ISO/IEC SQuaRE', *Information and Software Technology*, 70, pp. 18–29. doi: 10.1016/j.infsof.2015.09.006.
- Gomaa, W. H. (2013) 'A Survey of Text Similarity Approaches', 68(13), pp. 13–18.
- Gwet, K. (2013) 'Self-care behaviors of Italian patients with heart failure', *European Journal of Cardiovascular Nursing*, 12(1), p. S11. doi: <http://dx.doi.org/10.1177/1474515113477019>.
- Hirst, G. and St-onge, D. (1995) 'Lexical chains as representations of context for the detection and correction of malapropisms', (April).
- Jensen, L. S. and Martinez, T. (2000) 'Improving Text Classification by Using Conceptual and Contextual Features'.
- Jiang, J., Tsai, S. and Lee, S. (2012) 'Expert Systems with Applications FSKNN : Multi-label text categorization based on fuzzy similarity and k nearest neighbors q', *Expert Systems With Applications*. Elsevier Ltd, 39(3), pp. 2813–2821. doi: 10.1016/j.eswa.2011.08.141.
- Kadi, I., Idri, A. and Ouhbi, S. (2016) 'Quality evaluation of cardiac decision support systems using ISO 25010 standard', *2016 IEEE/ACS 13th International Conference of Computer Systems and Applications (AICCSA)*, (1), pp. 1–8. doi: 10.1109/AICCSA.2016.7945657.
- Khatter, K. and Kalia, A. (2013) 'Impact of non-functional requirements on requirements evolution', *International Conference on Emerging Trends in Engineering and Technology, ICETET*, pp. 61–68. doi: 10.1109/ICETET.2013.15.
- Miller, G. A. (1995) 'WordNet : A Lexical Database for English', 38(11), pp. 39–41.
- Pan, F. (2006) 'Multi-Dimensional Fragment Classification in Biomedical Text By', (September).

- Pavel Kruliš (2012) '[Master Thesis]ICT support for NFRs analysis, NFRs effort estimation and NFRs traceability'.
- Prajapati, P., Thakkar, A. and Ganatra, A. (2012) 'A Survey and Current Research Challenges in Multi-Label Classification Methods', (1), pp. 248–252.
- Rahimi, M., Mirakhorli, M. and Cleland-Huang, J. (2014) 'Automated extraction and visualization of quality concerns from requirements specifications', *2014 IEEE 22nd International Requirements Engineering Conference, RE 2014 - Proceedings*, pp. 253–262. doi: 10.1109/RE.2014.6912267.
- Ramadhani, D. A., Rochimah, S. and Yuhana, U. L. (2015) 'Classification of Non-Functional Requirements Using Semantic-FSKNN Based ISO/IEC 9126', *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, 13(4), p. 1456. doi: 10.12928/telkomnika.v13i4.2300.
- Rashwan, A., Ormandjieva, O. and Witte, R. (2013) 'Ontology-based classification of non-functional requirements in software specifications: A new corpus and SVM-based classifier', *Proceedings - International Computer Software and Applications Conference*, (ii), pp. 381–386. doi: 10.1109/COMPSAC.2013.64.
- Rozi, F., Fatichah, C. and Purwitasari, D. (2015) 'Ekstraksi Kata Kunci Berdasarkan Hipernim Menggunakan Fuzzy Association Rule Mining untuk Pengelompokan Dokumen', *Jurnal Ilmiah Teknologi Informasi (JUTI)*, 13(2), pp. 190–197.
- Saptarini, I., Rochimah, S. and Yuhana, U. L. (2016) 'Security Quality Measurement Framework for Academic Information System (AIS) Based on ISO / IEC 25010 Quality Model', pp. 128–135.
- Sommerville, I. (2010) *Software Engineering, Software Engineering*. doi: 10.1111/j.1365-2362.2005.01463.x.
- Suharso, W. and Rochimah, S. (2012) 'Sistem Penambang'.
- Ullah, S., Iqbal, M. and Khan, A. M. (2011) 'A survey on issues in non-functional requirements elicitation', *Proceedings - International Conference on Computer Networks and Information Technology*, pp. 333–340. doi: 10.1109/ICCNIT.2011.6020890.
- Widyantoro, D. H. and Yen, J. (2000) 'in Text Classification Task'.

Xiang, H. *et al.* (2015) 'Semantic modelling and automated reasoning of non-functional requirement conflicts in the context of softgoal interdependencies', *IET Software*, 9(6), pp. 145–156. doi: 10.1049/iet-sen.2014.0153.

Lampiran 1

Aspek Kualitas	Term Relevan	Hasil Pengembangan Term Gabungan Hipernim Sinonim
<i>Functional completeness</i>	Functional	operative, structural, usable, useable, serviceable, functional, working, practical, operable, useful, functioning, utilitarian, operational, running.
	Suitability	silence, unusefulness, rhythmicity, neediness, sweetness, caliber, unregularity, powerfulness, uncheerfulness, boundedness, urbanity, ending, irruption, unoriginality, worldliness, ugliness, worthlessness, ordinariness.
	Completeness	wholeness, agency, point, being, homozygosity, separation, status, dystopia, annulment, comprehensiveness, entirety, totality, fullness, entireness, rawness, integrality, rationality, rationalness.
	Covers	rest, feminize, degauss, expose, sensitize, tinct, intimate, exteriorize, read, rivet, slop, consume, subcontract, anticipate, dislodge, saltate, wrestle, herd.
	Specified	void, simplify, westernize, decimalise, full, undo, iodinate, conjugate, effeminise, hire, secure, provide, charter, qualify, rent, sign, specify, lease.
	Tasks	thin, retard, awake, weave, account, blast, loaf, destalinize, direct, funk, flux, wrestle, bend, worm, volunteer, course, thrust, sashay.
	Objectives	meat, disbelief, quoit, subject, metaknowledge, unbelief, piece, idea, bourn, square, place, hold, purpose, point, get, orient, design, overshoot.
<i>Functional correctness</i>	Functional	operational, functional, structural, functioning, operative, usable, running, useable, operable, serviceable, utilitarian, useful, working, practical.
	Suitability	register, directiveness, powerfulness, utility, inconsequence, incredibility, romanticism, nature, aura, disappearance, emersion, solubility, abstractionism, credibility, ornateness, image, characteristic, certainty.
	Correctness	incredibility, brachycephaly, complexness, positiveness, divisibility, combustibility, publicity, clearness, shape, immobility, change, discoloration, instance, modification, hardness, show, decorativeness, certainty
	Results	defining, processing, event, metempsychosis, sensitization, increase, levitation, activity, pulsation, zoom, stampede, progression, luck, pulsation, progress, consequence, hazard, gravitation
	Needed	iridesce, swing, adorn, broil, personify, bedevil, wash, continue, hold, slant, survey, squint, osculate, agree, iridesce, require, throw, object
	Degree	oddment, makeup, discard, cheerfulness, treasure, fullness, magnitude, colony, extension, estate, spirituality, letting, salvage, spirituality, things, landholding, stockholding, realty
	Precision	precision, exactitude, trueness, preciseness, minuteness, fidelity, petiteness, trueness, stuntedness, delicacy, precision, grain, preciseness, runtiness, tininess
<i>Functional appropriateness</i>	Functional	practical, operative, working, utilitarian, functional, useful, running, serviceable, operational, useable, structural, operable, usable, functioning
	Suitability	negativeness, illogic, volume, timbre, ulteriority, logicalness, changelessness, appearance, moderation, incredibility,

Aspek Kualitas	Term Relevan	Hasil Pengembangan Term Gabungan Hipernim Sinonim
		marvel, wonder, flash, complexion, unrealism, naturalness, understandability, schema
	Appropriateness	grace, correctitude, decorousness, appropriateness, priggishness, reserve, swashbuckling, correctness, manner, becomingness, decorousness, grace, decency, reserve, rightness, priggishness, appropriateness, seemliness
	Facilitate	refine, regenerate, satisfy, cure, bushel, aid, indulge, succor, back, subscribe, determine, serve, help, care, motivate, assist, effectuate, succor
	Accomplishment	sally, hindrance, hardening, lifelessness, pickings, inactivation, retrieval, clotting, destruction, wobbler, break, trouble, burst, crash, reversal, facility, ornament, sensitivity
	Tasks	mound, gain, rush, overturn, layer, receive, scupper, generate, carbonize, bestride, start, take, flux, fly, swoop, clerk, tilt, flow
	Objectives	terminus, telomere, equalizer, bourn, backboard, intent, railhead, essence, design, train, square, deploy, drive, purpose, orient, calculate, represent, level
<i>Time behaviour</i>	Performance	learn, designate, inform, demonstrate, surcharge, stultify, narrate, uncover, campaign, vouch, impute, disabuse, place, confound, thrive, value, recount, enter
	Efficiency	speed, loading, turnover, magnification, rate, quotient, productivity, efficiency, proportionality, teemingness, plentitude, lushness, albedo, profuseness, proportionality, voluptuousness, plentitude, turnover
	Time	gentile, pericarp, explorer, mutilator, departed, referral, sheet, combatant, dessertspoonful, disconcertment, odium, sheepishness, discomposure, degradation, disconcertion, bastardization, animalisation, dehumanization
	Behavior	stoppage, stillness, introversion, lamentation, usage, uncleanliness, shaping, transcription, measuring, execution, host, move, supplying, worship, politics, wastefulness, measurement, lead
	response	dent, comeupance, backwash, brisance, corollary, hazard, aftereffect, response, comeuppance, issue, repercussion, spat, heir, impact, branch, ectotherm, sibling, kinswoman
	processing	hydrolise, bubble, objectify, deodorise, sadden, engrave, colorize, liberalize, industrialize, discommode, process, redound, color, excite, queer, colour, endanger, stimulate
	Throughput	footstep, containerful, sericulture, brainchild, dose, pick, ullage, picking, skinful, statuary, trinketry, microflora, pile, flinders, mixture, tenantry, parcel, biology
	Rates	score, order, speculate, reason, link, appraise, value, reassess, place, rad, building, fivesome, room, optics, solvability, meeting, pitch, fiftieth
	Functions	immobility, materiality, detergency, depositary, divisibility, uncludedness, arboretum, role, subjectivism, waterworks
<i>Resource utilization</i>	Performance	profile, formulate, teach, apprise, stress, contradict, model, cite, learn, terminate, decorate, cut, gape, flourish, retard, ramify, count, suggest
	Efficiency	quotient, albedo, eccentricity, productivity, frequency, mileage, odds, magnification, stoichiometry, luxuriance, albedo, teemingness, cornucopia, bountifulness, bounty, stoichiometry, overabundance, amplitude
	Resource	share, resource, property, credit, finances, liabilities, capital, percentage, overage, deal, bid, shot, protection, piece, pains,

Aspek Kualitas	Term Relevan	Hasil Pengembangan Term Gabungan Hipernim Sinonim
		end, strain, stake
	Utilization	overdrive, consumption, end, involution, phenomenon, digestion, sincerity, shedding, muliebrity, moderation, activation, sin, sublimation, control, pattern, amphidiploid, promotion, disturbance
	Amounts	capital, resource, security, holding, treasure, circumstances, share, finances, intangible, slice, receivables, reallocation, go, shim, batting, stab, protection, security
	Functions	deadness, excellence, inhumanity, sufficiency, water, voluptuousness, ulteriority, fruitlessness, positiveness, waterworks
<i>Capacity</i>	Performance	announce, evoke, cue, carnalize, smile, negate, rat, finger, remonstrate, value, set, curve, substantiate, enlarge, explicate, yaw, abound, boom
	Efficiency	magnification, turnover, efficiency, quotient, frequency, signal/noise, signal/noise, rate, prevalence, hematocrit, wealth, proportionality, prevalence, bountifulness, copiousness, voluptuousness, profusion, turnover, plenty
	Capacity	reach, capableness, posture, strength, executability, sensitiveness, suggestibility, grasp, competency, strength, defensibility, capacity, overkill, associableness, posture, executability
	Maximum	bound, cutoff, boundary, acme, limit, apex, vertex, end, extreme, apex, peak, metacenter, acme, metacentre, crown
	Limits	volume, coverage, area, mass, ambit, scope, top, length, compass, limit, deductible, figure, report, compass, advance, frontage, gain, length
	Parameter	incessant, stable, unswerving, faithful, staunch, steadfast, changes, invariable, uninterrupted, quantum, constant, binomial, parameter, amount, sum, total, product, proportional
<i>Co-existence</i>	Compatibility	tenderness, drift, thing, bravery, nonpartisanship, astonishment, ungratefulness, faintness, dislike, pathos, carrier, partnership, kindheartedness, agency, underperformer, burden, nervousness, solicitude
	Coexistence	kill, act, measure, tally, weigh, gibe, remain, straddle, preexist, suit, litter, make, intrude, face, hang, consist, contain
	Required	want, ask, order, claim, invite, encore, solicit, call, need, boil, browse, bedevil, bake, appear, confound, fuddle, impend
	Sharing	lap, dag, interface, rainfly, example, lappet, barndoor, memory, image, criticality, neighbourhood, interface, horde, house, village, slump, hamlet, settlement
	Environment	guiltiness, utilization, prepossession, scandalisation, moderatorship, deanery, improvement, catering, prognathism, ludicrous, foolish, derisory, nonsensical, ridiculous, laughable, idiotic, preposterous, cockeyed
	Detrimental	prejudicious, prejudicial, detrimental, damaging, harmful
	Impact	break, channel, lick, irruption, reciprocation, splicing, objective, engagement, burst, swab, brushing, spinney, panel, gunfight, fray, hairweaving, precipitator, groundcover
<i>Interoperability</i>	Compatibility	levity, passion, astonishment, nonpartisanship, soulfulness, pain, stab, desire, affect, tenderness, incumbrance, bother, worry, firm, jumpiness, empathy, discomposure, disconcertion
	Interoperability	wholesomeness, naivety, content, faithfulness, credibleness, divisibility, factuality, finiteness, infinitude, suppleness, form, competency, flexibility, immunocompetence, sensitiveness, penetration, capability, interoperability

Aspek Kualitas	Term Relevan	Hasil Pengembangan Term Gabungan Hipernim Sinonim
	Exchange	idiolect, schmoose, talking, nothings, spell, schmooze, dictation, colloquy, pronunciation, burst, cover, barrage, gossiping, talk, counterfire, confabulation, rap, talking
	Information	request, bespeak, quest, receipt, acknowledge, return, plant, implant, deliver, refusal, repertoire, humor, mythology, battery, signal, counseling, metaknowledge, library
<i>Appropriateness recognizability</i>	Usability	snootiness, navigability, brachycephalism, assembler, unresponsiveness, combustibleness, impenetrability, understandability, certainty, waterworks
	Appropriateness	correctitude, decency, grace, correctness, decorum, improperness, primness, manner, impropriety, priggishness, primness, decency, decorousness, grace, appropriateness, modesty, reserve, correctness
	Recognized	snaffle, confuse, belong, estimate, benefit, bristle, value, confound, honour, plant, bastardise, canonise, return, espouse, recognise, concede, label
	Users	heteroploid, survivor, bomber, nonworker, comforter, topper, saver, technologist, scrapper, nondrinker, creeper, sprawler, indigen, rusher, dieter, learner, advocate, archaist
	System	serviceability, household, house, duplicate, ceramic, restoration, detergency, inset, sheet, weaponry, arms, hardware, connexion, connector, brick, transport, connecter, equipment
	Needs	laxness, radiance, diploidy, homelessness, rustication, proclamation, multivalency, ascendancy, saturation, scandalization, tautness, resistance, danger, circumstance, absolutism, deification, exoneration, light
<i>Learnability</i>	Usability	debugger, fidelity, positiveness, unchangingness, repository, ineptness, navigability, popularity, uncertainty, waterworks
	Learn	keratoscopy, lavatory, barroom, octette, can, allegretto, direction, capriccio, tracing
	Achieve	work, reach, win, accomplish, run, follow, supersede, clear, enter, decapitate, rock, harm, croquet, get, lynch, bat, steal, push
	Goals	defense, welcome, deal, proposal, percentage, mind, witticism, signaling, batch, odor, dock, remains, wit, tissue, unbreakableness, church, buttocks, butt
	Freedom	ethos, power, hostility, point, situation, uncheerfulness, crystallisation, depth, unemployment, democracy, republic, constitutionalism, autarchy, lobby, amphictyony, union, suzerain, dyarchy
	Risk	force, reinstatement, ascendancy, innocence, stigmatism, shrubbery, eye, bedground, climate, exposure, peril, perilousness, vulnerability, riskiness, insecurity
	Satisfaction	rejoicing, bonheur, satisfaction, belonging, gladness, merriment, gaiety, blitheness, gladsomeness, commutation, entail, nationalization, complacency, royalty, gratification, overpayment, quittance, motility
<i>Operability</i>	Usability	complexness, wholesomeness, pathos, truth, compiler, unnaturalness, unusefulness, transportation, incorrectness, waterworks
	Operation	concretion, development, forbiddance, enterprise, variation, accomplishment, defence, labour, trial, decoration, process, affection, falsity, example, boom, change, sunshine, plaything
	Easy	elementary, unchaste, pleasing, abundant, clean, smooth, undemanding, gentle, easygoing, available, soft, waxy, relaxed, wanton, sluttish, comfortable, casual, promiscuous, painless, simple

Aspek Kualitas	Term Relevan	Hasil Pengembangan Term Gabungan Hipernim Sinonim
	Control	baron, soapiness, materiality, destructiveness, factualness, expressiveness, opaqueness, perfection, insolubility, gas, incompatibility, attainability, flatulency, interestingness, persuasiveness, effectuality, dominance, authorisation
<i>User error protection</i>	Usability	complexity, assembler, naturalness, ineptitude, responsiveness, unlawfulness, changelessness, divisibility, usability, waterworks
	User	demandeur, urinator, changer, achiever, hater, nondrinker, compulsive, closer, scrag, apprehender, measurer, pursuer, objector, ejector, image, untier, affiant, baldhead
	Error	error, retrieval, discovery, disposition, recovery, egress, abidance, equalisation, delivery, miscreation, flash, collapse, lapsing, loser, captive, misfortune, dysphagia, washout
	Protection	source, stiffening, sericulture, last, implantation, residency, dissimulation, overdrive, conditionality, mystification, positioning, vagary, chelation, libration, behaviour, measuring, pleasure, hell
	Making	response, reciprocity, product, dime, contingency, nightwear, decimalization, offspring, beachwear, deformation, tempo, easing, lessening, velocity, deceleration, separation, speedup, kc
<i>User interface aesthetics</i>	Usability	clarity, unsatisfactoriness, rightness, naivete, badness, impenetrability, occidentalism, spinnability, directivity, waterworks
	User	arrogator, surrenderer, polymorph, occultist, cripple, carrottop, acquaintance, abator, topper, technologist, chooser, anomaly, planner, individualist, longer, cripple, innocent, cashier
	Interface	thermosphere, squeaker, aerofoil, release, decker, comb, reflector, autocue, applicator, bunk, substrate, roadbed, stinker, superficies, strip, garden, antiquity, trundle
	Aesthetics	theosophism, revolutionism, prescriptivism, classicalism, dualism, contextualism, rationalism, theory, aesthetics, feminism, millennium, imitation, dualism, irridentism, gymnosophy, ethicism, rationalism, descriptivism
	Enables	raiment, marvel, influence, reverberation, reverse, decrease, cent, crown, destiny, wpm, frequency, lessening, pace, scintillation, surprise, shimmer, deathrate, mph
	Pleasing	nourish, suit, nurture, cover, allay, behoove, board, drench, serve, qualify, serve, do, function, answer, cover
	Satisfying	flock, suit, cross, march, rendezvous, fend, celebrate, rhyme, behave, cross
	Interaction	extractor, whirler, beatification, democratization, set, dependency, busywork, washing, vitrification, humour, qualification, execution, paving, variable, marker, fetch, pain, squeaker
<i>Accessibility</i>	Usability	depository, useableness, range, sameness, quantifiability, worldliness, instrumentality, amorality, ineptness, waterworks
	Access	reactionary, outside, precise, just, proper, straight, satisfactory, exact, veracious, prepossession, urbanisation, nomination, line, squeaker, erectness, form, danger, impropriety
	System	thing, menage, function, lemon, antiquity, troop, float, den, gang, means, system, device, container, connection, toiletry, furnishing, weaponry, conveyance
	People	quantity, man, multitude, actinide, circuit, otherworld, subgroup, verna, vinyl, covering, series, community, nub, collection, reliving, rock, cosmos, labor
	Wide	panoptic, fanlike, inaccurate, opened, ample, spacious, large, extensive, wide, deep, comfortable, encompassing, full, thick, broad, ope

Aspek Kualitas	Term Relevan	Hasil Pengembangan Term Gabungan Hipernim Sinonim
		n,beamy,sweeping,blanket,comprehensive
	Range	bound,proportion,ambit,meridian,resultant,strength,peak,acme,ultimateness,bound,frontage,depth,length,newscast,peanuts,payroll,range,paysheet
	Characteristics	abstract,consistence,concentration,section,quantity,smell,fullness,strength,grade,lease,rateables,stockholdings,trust,heirloom,commonage,realty,stockholding,immovable
	Capability	romanticism,unfaithfulness,impenetrability,difficulty,inaccuracy,physicalness,utility,pathos,infinitude,suppleness,competence,pliancy,penetration,flexibility,capability,form,flexibleness,contractility
<i>Maturity</i>	Reliability	direction,prerequisite,socialisation,creditworthiness,requirement,answerability,deprivation,land,province,turn,sport,accountancy,hell,survivor,wrongdoing,jinx,delectation,utilisation
	Maturity	wholeness,enlargement,action,conditionality,immatureness,point,condition,revocation,happiness,union,republic,diarchy,country,community,power,cell,cadre,state
	Under	below,under,sane,natural,regular,mean,modal,standard,normal,typical,average,perpendicular,median
	Normal	sane, natur, regular, mean, modal, standard, normal, typic, averag, perpenducular, median
	Operation	rigidifying,swing,destruction,rivalry,leveling,blockade,subbing,vampirism,communicating,sphere,exclusion,absurd,water,example,matter,contact,cause,vampirism
<i>Availability</i>	Reliability	detrribalization,incumbency,calming,privatisation,direction,penalization,preserve,normalization,detrribalisation,obfuscation,ironing,creamery,lamenter,shop,plight,locating,enjoyment,brokerage
	Availability	usable,easy,addressable,visible,ready,open,acquirable,disposable,accessible,purchasable,free,lendable,gettable,uncommitted,forthcoming,gettable,useable,available,procurable,obtainable
	Accessible	reachable,getatable,comprehensible,accessible,available,handy,convenient,approachable,comprehensible
	Required	necessitate,clamor,expect,govern,hold,postulate,appeal,exact,book,lend,throw,hurt,osculate,broil,like,confound,translate,suffer
<i>Fault tolerance</i>	Reliability	mobilisation,accountability,requirement,safekeeping,communication,distaff,corrections,land,detrribalisation,conduct,chore,trade,performing,stickup,assist,training,beehive,craft
	Fault	departure,egress,touching,hinderance,abidance,leveling,uncovering,derivation,assumption,malfunction,ruin,survivor,complaint,reversion,folie,nonconformity,avalanche,griever
	Tolerance	power,tolerance,dynamism,lustiness,stoutness,heartiness,huskiness,survival,skin,indulgence,tolerance,acceptance,toughness,leniency,stamina,overtolerance,leniece,toleration
	Intended	purpose,contemplate,speculate,evolve,make,center,concentrate,hatch,program,hope,overshoot,aim
	Despite	successful,fashionable,stylish,incoming
	Presence	delimit,compare,scintillate,inhabit,litter,kill,compact,weigh,range,nose,eyeball,contain,fox,cruise,hum,harmonize, confront
	Hardware	connection,system,means,connexion,arms,connecter,bomb,connector,furnishing,confirmation,thickener,emission,

Aspek Kualitas	Term Relevan	Hasil Pengembangan Term Gabungan Hipernim Sinonim
		rock,toner,fill,filling,vernix,secret
	Software	cypher,reading,omerta,statement,cryptograph,command,micro code,address,access
<i>Recoverability</i>	Reliability	province,dependableness,kingdom,privatisation,reliability,penalization,denationalisation,safekeeping,penalisation, support,stickup, writing,diversion,busywork,hiding,wash,pit,lumberyard
	Recover	wan,chasten,hasten,eroticize,waken,synthesize,twine,inseminate,rankle,express,preisolate,segregate,garner,earn, recover,glom,lease,count
	Event	standardisation,guiltiness,memorialization,brutalization,pageantry,diversity,ecology,impurity,condemnation, resolution,harvest,offshoot,electricity,backwash,luck,effect,pu lsation,force
	Interruption	bedim,incandesce,emulsify,sanitise,sentimentalise,eternize,stiffen,decelerate,detransitivise,filibuster,hang,catch, bottleneck,check,stall
	Failure	propulsion,touch,mitsvah,leaning,delivery,egression,communicating,disposal,residence,victim,outburst,luckiness, nonconformity,languisher,miscreation,attendant,wonder,down fall
	Data	block,packet,multitude,pantheon,set,procession,crop,phrasing, agglomeration,swinging,hurrying,collage,cause, botany,family,stream,advance,bunch
	Directly	straight,instantly,immediately,now,directly,forthwith,flat,direct,straightaway
	Affected	switch,accouterment,breakup,movement,filtration,impact,nationalisation,increase,twinkle,deceleration,megacycle, retardation,frequence,rpm,vagary,kc,avulsion,breakup
	Desired	felicitate,state,drop,outcry,itch,place,reorder,care,voice,indweller,winner,decedent,unfastener,advocator,showman, extravert,brunette,redheader
<i>Confidentiality</i>	Security	deflector,way,impaction,regularization,frizz,dampener,centre,t autness,vacuolization,impregnability,risklessness, security,invulnerability,salvation
	Confidentiality	insularity,covertness,solitariness,insulation,quarantine,hiding,s eclusion,hiddenness,solitude,confidentiality, covertness,hiding,bosom
	Ensures	tick,justify,condemn,uphold,back,ascertain,propose,corroborate,float,detain,straighten,commercialize,synchronize, arterialise,contract,bestialize,trace,professionalize
	Data	repertoire,solicitation,planting,motley,gathering,family,edition ,armamentarium,lanthanide,advancement,gathering, vegetation,wardrobe,slippage,disbandment,decolonisation,pla nting,miscellany
	Accessible	available,accessible,reachable,convenient,handy,comprehensible,getatable,comprehensible,approachable
	Authorized	permission,opinion,relegating,conge,title,toleration,passport,te stament,allowance,turn,clothe,defervesce,caravan, implant,sour,thin,vanish,fold
<i>Integrity</i>	Security	unsusceptibility,protuberance,scandalisation,mode,depressor,s ilence,whiteness,plectrum,serration,risklessness, invulnerability,salvation,impregnability,security
	Integrity	preparation,nationhood,damnation,conflict,maturity,pontificat e,revocation,illumination,raptus,autarchy,ochlocracy,plutocrac

Aspek Kualitas	Term Relevan	Hasil Pengembangan Term Gabungan Hipernim Sinonim
		y, cadre, superpower, autocracy, republic, technocracy, community
	Prevents	spoil, balk, cross, forefend, exclude, block, kibosh, avert, stymie, foil, hinder, thwart, obviate, baffle, impede, frustrate, forestall, save, foreclose, blockade
	Unauthorized	unaccredited, unofficial, unlicensed, illegitimate, unauthorised, unauthorized, unlicensed, wildcat
	Access	straight, exact, correct, rightmost, precise, proper, ripe, rightist, split, connectivity, serration, textile, horizontality, restoration, jaunty, mummification, material
	Modify	expiry, guinea, depolarisation, impact, playing, alienation, ninepence, accessory, blue, cps, pace, development, speedup, deathrate, spacing, shimmer, birthrate, kilohertz
	Data	sainthood, straggle, exposition, caboodle, batch, people, agglomerate, ingathering, findings, universe, content, cumulus, pursuit, flagging, expo, leapfrog, travel, coup
<i>Non-repudiation</i>	Security	place, bait, regularisation, uncomfortableness, spermicide, strike, block, prophylactic, encapsulation, security, salvation, impregnability, invulnerability, risklessness
	Repudiation	touching, disinterment, egress, delivery, felicitation, explanation, leveling, proposal, boast, desertion, forgoing, escape, proscription, shunning, renunciation, displacement, forswearing, banishment
	Verity	maintain, justify, claim, vitalize, back, document, reaffirm, reassert, buttress, whish, lurch, whine, guarantee, sanction, prance, spirt, change, sit
	Place	weight, short, readiness, footer, magnetization, nationhood, antagonism, li, emcee, centrepiece, performer, invariant, facet, magnet, host, pleaser, contour
	Events	home, whiteness, demand, frizz, normality, subservience, deshabile, exercise, ascendancy, nonevent, offshoot, conductivity, fortune, backwash, offspring, brisance, pulsation, materialization
	Actions	manoeuvre, disassemble, litter, seethe, evade, enact, underact, clean, snap, descend, force, manage, condescend, interrupt, manoeuvre, attempt, festinate, jump
	Repudiated	place, disregard, forecast, snub, resist, think, declare, spurn, prove, ignore, suspect, discount, discredit, recuse, mistrust, distrust, reprobate, repudiate
	Later	after, afterwards, afterward, subsequently, later
<i>Accountability</i>	Security	impaction, energizer, sounder, synchroflash, protuberance, susceptibility, normalcy, aspergill, shoehorn, risklessness, salvation, impregnability, invulnerability, security
	Accountability	prerequisite, socialization, calming, deprivation, accountability, privation, duty, detribalization, preserve, crier, nympholept, rolling, variance, behavior, flick, positioning, line, continuation
	Traced	dispense, dissect, review, investigate, remedy, consider, botanise, sieve, analyse, appraise, name, survey, inspect, trace, vivisect, assay, follow, investigate
	Uniquely	xenon, solder, barium, argonon, tellurium, ytterbium, dubnium, fluorine, radon, francium, cobalt, iron, iridium, lutecium, manganese, berkelium, bismuth, ruthenium
	Entity	abstraction, thing, entity
<i>Authenticity</i>	Security	melioration, toy, converter, demand, ignitor, goal, afterburner, sup

Aspek Kualitas	Term Relevan	Hasil Pengembangan Term Gabungan Hipernim Sinonim
		port, nomination, salvation, security, risklessness, invulnerability, impregnability
	Authenticity	changeability, utility, ulteriority, corporality, memorability, legitimacy, stuff, bad, relativity, rigor, cogency, plausibleness, validity, cred, rigour, plausibility, legitimacy, genuineness
	Identity	notable, eidos, lion, time, character, form, individuality, state, toast, oneness, equality, individuality, personableness, personhood, unity, similarity, selfsameness, indistinguishability
	Subject	acknowledge, bespeak, plant, implant, carry, quest, receipt, render, deliver, lot, info, idea, commitment, perception, parcel, procession, mind, bunch
	Resource	intangible, capital, funds, liabilities, circumstances, credit, property, belongings, sum, end, striving, stake, coulter, essay, run, shot, finances, quoin
	Verify	reenforce, uphold, document, maintain, sanction, substantiate, establish, support, steady, ramble, turn, motor, rove, hie, localise, further, rise, meander
	Claimed	bless, claim, plead, attest, animadvert, show, vouch, reassert, exclaim, pan, outflank, withdraw, intervene, circulate, crank, encourage, refinance, show
<i>Modularity</i>	Maintain	start, dispatch, exclude, suppress, approach, commit, counter, react, overprotect, smut, lift, colourize, supercharge, weaponize, demagnetize, discolor, glorify, redo
	Modular	modular, standard
	Composed	supplement, attempt, pack, add, appoint, activate, machinate, trigger, mount, bolshevize, fly, unfit, dynamize, brutalise, operate, port, begin, bake
	Discrete	distinct, separate, discrete
	Components	neck, stuff, batting, basis, position, chukker, unconnectedness, somewhere, rest, kernel, offer, vehicle, story, mixture, subject, signaling, residual, center
	Change	repeat, proceed, synchronize, coincide, befall, strike, backlash, develop, find, emulsify, compartmentalize, inactivate, vascularize, acetify, herald, denitrify, publicize, dynamize
	Minimal	nominal, tokenish, minimum, borderline, marginal, stripped, token, minimal, negligible
	Impact	avalanche, lens, outbreak, finish, anastigmat, thing, osculation, stroking, episode, bottlebrush, jack, undergrowth, swatter, paintbrush, dock, slugfest, gunplay, affray
<i>Reusability</i>	Maintain	surprise, stampede, volunteer, fence, contend, straiten, dare, charm, harbor, communise, end, concentrate, cut, slenderise, stabilize, coarsen, boil, clarify
	Reusable	reusable, recyclable, utile, reclaimable, useful
	Asset	amorality, badness, highness, spinnability, burnability, uncertainty, form, ulteriority, measurability, treat, beginning, opaqueness, foreignness, ending, sweetness, immateriality, divisibility, fecundity
	More	much, more, many
	System	practicability, block, furnishing, pavement, outfit, cloth, slip, household, mystification, connective, connector, connexion, means, brick, implement, arms, furnishing, munition
	Building	owe, oxygenize, merit, disqualify, purify, glow, radicalize, win, cantilever, exaggerate, overpraise, overleap, oversimplify
	Other	different, separate, another, early, opposite, unusual, new, otherwise

Aspek Kualitas	Term Relevan	Hasil Pengembangan Term Gabungan Hipernim Sinonim
<i>Analysability</i>	Maintain	e, strange, former, other, past maintain, wait, repeat, interact, forestall, attempt, anticipate, solemnise, wharf, detransitivize, waken, obliterate, enable, overcloud, match, mingle, attune, louden
	Analyze	mistreat, communicate, scupper, irrigate, chlorinate, chrome, subject, discommode, power, mollycoddle, parody, caricature, cosset, guy, step, tantalise, rib, baby
	Possible	expectancy, foretaste, obverse, conceptualisation, law, conceptuality, attainability, part, chance, potentiality, attainability, opportunity, attainableness, chance, conceivability, potency, potential, achievability
	Assess	critique, valuate, meditate, rationalise, figure, ascribe, score, rivet, reassess, odour, ninetieth, feel, foursome, compositeness, quintette, seventieth, quorum, archidiaconate
	Impact	finish, outburst, kiss, middleman, transmission, skimming, rub, delegate, splice, eraser, zapper, attenuator, placement, brushing, inductor, oar, coppice, ruffle
	Intended	pay, connect, hatch, devise, formulate, make, rationalise, philosophize, construct, aim, hope, overshoot
	Change	transpire, recur, bechance, backfire, encounter, backlash, fall, give, befall, puncture, deform, infract, orientalise, untie, riddle, regress, gnaw, crystallize
	Diagnose	operate, administer, sift, transfuse, dress, dispense, vet, cup, iodise, consider, check, survey, compare, vivisect, scrutinize, follow, anatomise, reexamine
	Deficiencies	mood, pureness, innocence, mendicancy, deshabelle, ascendance, hairlessness, atonia, impecuniousness, deficit, tightness, shortage, deficiency, necessity, absence, stringency, shortness, want
	Causes	overture, family, trichion, crossing, root, corner, prelude, issue, bloodline, filiation, egress, presumption, reason, descent, lexicalisation, infection, wear, justification
	Failures	disposition, interference, egress, effectuation, activity, motivating, proclamation, disinterment, derivation, pariah, downfall, nympholept, sorrower, contingency, flop, colpoxerosis, outbreak, contingence
	Identity	regularise, plump, pace, define, actuate, reset, incline, resolve, interpolate, capitalize, melt, conjugate, puree, rosin, factor, blur, worry
	Modified	error, civility, failure, godsend, impairment, conclusion, contingency, marvel, union, velocity, detachment, fluctuation, megacycle, increase, tempo, deathrate, flow, play
<i>Modifiability</i>	Maintain	safeguard, essay, perpetrate, use, straiten, garner, deny, control, jug, deprave, depersonalize, emulsify, masculinize, etherealize, renormalize, animate, transmute, affect
	Modified	tuppence, nightwear, transgression, damage, emphasizing, sparkling, burst, conversion, alienation, cps, getaway, nativity, rpm, sublimation, surprise, development, kHz, kilohertz
	Detects	imminence, plurality, illumination, agency, degree, omnipotence, inactiveness, medium, impendence, conceive, professionalise, hydrate, saponify, drift, acetylyze, consider, stabilize, check
	Degrading	disgrace, affront, smash, demean, crush, diss, degrade, sting, mortify, sunder, morph, collimate, glamorize, calcify, vulgarise, eliminate, contract, internationalize
	Existing Quality	alive, extant, existing, active, existent, present sunniness, set, trait, possibility, cheerfulness, law, group, dimensio

Aspek Kualitas	Term Relevan	Hasil Pengembangan Term Gabungan Hipernim Sinonim
<i>Testability</i>	Maintain	n, personality, unpleasantness, fruitlessness, occidentalism, uncloudedness, incomprehensibility, changelessness, inheritance, changeability, strangeness
	Tested	misbehave, take, lag, try, pursue, salt, offer, shoot, comport, symmetrize, corrupt, demonize, raise, normalise, vaporise, vulgarize, renormalise, amend
	Test	pore, place, measure, choose, approve, link, assign, order, grade, concentration, fivesome, consistency, anisotropy, form, compositeness, lowness, unsolvability, upgrade
	Criterion	praise, score, estimate, fail, mull, speculate, study, relate, reassess, downgrade, arcminute, spatiality, rating, gentleness, cast, constitution, overvaluation, optics
	Established	kt, operator, play, weight, chance, multinomial, utility, bit, point, bimetallic, norm, goalpost, stump, cord, banner, gauge, oriflamme, scale
	Determine	public, unresolved, staring, unrestricted, undefendable, unconcealed, coarse, vulnerable, unsealed, beat, confabulate, kindle, bandy, plant, style, deliberate, confer, bring
<i>Adaptability</i>	Portability	candidature, vindication, causa, procession, grounds, experiment, force, nature, antecedent, forbear, synchronizing, forebear, progenitor, family, kin, progeny, desynchronisation, synchrony
	Adaptability	manoeuvrability, locomotion, manipulability, play, movableness, unsteadiness, ricketiness, portability, maneuverability, wasting, sneeze, palsy, sickness, hurting, hypocalcemia, cicatrice, diarrhea, unconstraint
	Different	suitability, popularity, humanness, limitlessness, directivity, attraction, sanctitude, simplicity, manhood, sensitivity, capableness, interoperability, pliancy, pliability, totipotency, penetration, competence, capability
	Evolving	various, diametrical, antithetical, contrary, antithetic, contrastive, varied, polar, several, different, divers, assorted, distinct, variant, contrasting, unlike, diverse, divergent, other, incompatible
	Hardware	complexify, demist, distil, cure, verbify, suburbanise, intervene, tenderize, cook, organise, reproduce, erupt, expand, yield, embellish, sire, actualise, bring
	Software	ironware, bomb, implement, armament, means, connective, conveyance, connexion, toiletry, contaminant, nonconductor, skinny, mote, format, ballast, program, packing, elastomer
	Operational	codification, reference, prescription, transcription, cypher, postcode, correspondence, cipher, writing
	Environments	operative, operating, operational, usable, useable, effective, fighting, operable, functional, active, serviceable
<i>Installability</i>	Portability	chair, biz, place, despair, generalship, protuberance, stillness, studentship, chaplaincy, cockeyed, ridiculous, derisory, unlogical, illogical, laughable, idiotic, nonsensical, ludicrous
	Installed	motivity, maneuverability, looseness, locomotion, restlessness, motility, play, manoeuvrability, movableness, wasting, febrility, hemoglobinemia, ketonuria, haemoglobinemia, cicatrice, exophthalmos, anemia, ague
	Effectiveness	isomerise, check, demonise, splatter, excite, deaminate, cannibalise, colorise, neutralize, decoration, mystification, electroplate, instrumentality, plaything, creation, closing, structure, interpolation
		stranglehold, good, neediness, motionlessness, immaturity, naïveté

Aspek Kualitas	Term Relevan	Hasil Pengembangan Term Gabungan Hipernim Sinonim
		vete, conductivity, turbulence, memorability, oligospermia, chokehold, wakefulness, incompatibility, valence, puissance, sleep, stranglehold, authority
	Efficiency	prevalence, productivity, loading, magnitude, haematocrit, eccentricity, albedo, hematocrit, amplitude, eccentricity, prevalence, loading, speed, lushness, productivity, profuseness, st oichiometry
	Successfully	erbium, actinium, manganese, beryllium, tambac, pewter, thorium, chromium, astatine, zirconium, chromium, barium, potassium, caesium, nickel, osmium, palladium, cerium
	Remove	evacuate, dispel, set, place, lock, force, shunt, hustle, precess, refill, l atter, guard, trice, shipment, secondment, stock, feature, endorsement
<i>Replaceability</i>	Portability	portability, restlessness, ricketiness, manoeuvrability, manipulability, motility, motivity, maneuverability, play, hypercholesterolemia, festination, hemoglobinemia, spots, hypercholesteremia, lipidaemia, irregularity, sottishness, pyrosis
	Replaceability	inconstancy, fungibility, variance, transmutability, duality, liquidity, changefulness, transposability, substitutability, insolubility, connectivity, permutableness, hydrophobicity, weakness, edibility, style, consistency, scent
	Another	another, other
	Product	instrumentality, plaything, lemon, block, fungible, cushioning, salvage, ware, ornament, primary, alkalic, elemental, alkaline, elementary, canonical, basal, rudimentary, canonic
	Same	said, duplicate, same, comparable, aforementioned, unvarying, unchanged, like, equal, unvaried, synoptic, identical, synoptical, aforesaid, similar, indistinguishable, homophonic, one, very, selfsame
	Purpose	jackstones, terminus, thought, finish, core, culture, bourn, touchdo wn, experience, propose, typify, target, import, place, orient, charge, drive, aspire

Lampiran 2

Hasil Pengukuran Keterkaitan Semantik antara Term Relevan Aspek Kualias dengan Term pada Kalimat Kebutuhan.

++++
Kalimat Kebutuhan Ke : 401

++++

TermRel ID :1 Term Relevan : function

Term Kalimat 401 : calle
Term Kalimat 401 : disabl
Term Kalimat 401 : account
Term Kalimat 401 : can
Term Kalimat 401 : receiv
Term Kalimat 401 : ani
Term Kalimat 401 : phone
Pos : n
HSO value : 0.2222
Term Kalimat 401 : call
Pos : n
HSO value : 0.2222
Term Kalimat 401 : from
Term Kalimat 401 : ani
Term Kalimat 401 : caller

++++
Kalimat Kebutuhan Ke : 407

++++

TermRel ID :94 Term Relevan : under

Term Kalimat 407: caller
Term Kalimat 407: whose
Term Kalimat 407: number
Term Kalimat 407: identifi
Term Kalimat 407: be
Term Kalimat 407: restrict
Term Kalimat 407: under
Pos : a

HSO value : 1

Pos : r

HSO value : 1

Term Kalimat 407: calle

+++++

Kalimat Kebutuhan Ke : 484

+++++

TermRel ID :187 Term Relevan : test

Term Kalimat 484: system

Term Kalimat 484: need

Term Kalimat 484: construct

Term Kalimat 484: maintain

Term Kalimat 484: map

Term Kalimat 484: from

+++++

Kalimat Kebutuhan Ke : 526

+++++

TermRel ID :35 Term Relevan : amount

Term Kalimat 526: system

Pos : n

HSO value : 0.2778

Term Kalimat 526: softwar

Term Kalimat 526: must

Term Kalimat 526: abl

Term Kalimat 526: expand

Term Kalimat 526: maximum

Pos : n

HSO value : 0.3333

Term Kalimat 526: number

Pos : n

HSO value : 0.4444

Term Kalimat 526: user

Term Kalimat 526: account

Term Kalimat 526: system

Pos : n

HSO value : 0.2778

Lampiran 3

Kesepakatan ketiga Pakar dalam membuat Ground Truth menggunakan AC1

KK	Functional Completeness			Hasil Kesepakatan (Gwet's AC1)		
	Pakar 1	Pakar 2	Pakar 3	P1+P2	P2+P3	P1+P3
50				1	0,2	0,2
51				0,2	1	0,2
52						
53						
54						
55						
56						
57						
58						

Lampiran 3 merupakan kesepakatan dari ketiga pakar dalam menentukan apakah kalimat kebutuhan tersebut masuk ke dalam aspek kualitas tertentu. Kalimat Kebutuhan ke 50 dan 51 masuk ke dalam aspek kualitas *Functional Completeness*. Hal ini dikarenakan kesepakatan yang diperoleh dari dua pakar. Kesepakatan tertinggi dari keduanya menghasilkan nilai 1. Akan tetapi, salah satu dari kedua pakar tidak sepakat maka akan menghasilkan nilai 0.2.

BIOGRAFI PENULIS



Lukman Hakim adalah anak pertama dari dua bersaudara. Penulis dilahirkan dari keluarga sederhana dari pasangan Hamsus Sholihin dan Agustin. Penulis memulai pendidikan dasar pada tahun 1995 di SDN 8 Besuki, dan melanjutkan ke jenjang sekolah menengah pertama pada tahun 2001 di SMPN 1 Banyuglugur. Pada tahun 2004 penulis melanjutkan ke jenjang sekolah menengah atas di SMAN 1 Suboh dilanjutkan ke jenjang diploma 3 pada tahun 2007 di Universitas Muhammadiyah Jember dan pada tahun 2010 penulis melanjutkan ke jenjang S1 di Universitas Muhammadiyah Jember. Pada tahun 2017 penulis melanjutkan ke jenjang S2 di Jurusan Teknik Informatika Institut Teknologi Sepuluh Nopember. Penulis akan memulai karir sebagai dosen program studi D4 Teknik Informatika di Politeknik Negeri Jember.