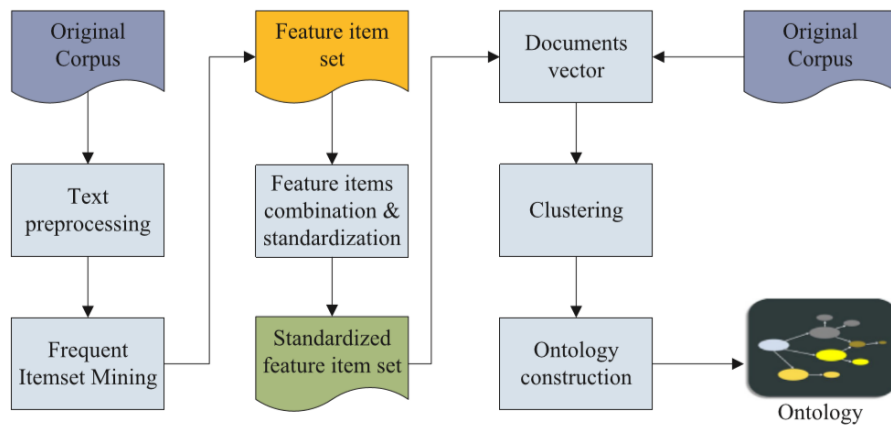


LAPORAN TUGAS

Nama : Rakha Asyrofi / Taufik Hidayat Mata Kuliah : Kecerdasan Komputasional
NRP : 05111950010038 / 05111950015004 Dosen : Dr. Eng. Chastine Fatichah, S.kom, M.Kom.



Gambar 1. Ontology Construction

Pertama, pra-proses corpus asli dan kemudian lakukan bagian dari Part of Speech (POS) tag. Selanjutnya, gunakan metode TF-IDF untuk menambang item fitur dalam teks dan melakukan pemangkasan buatan untuk menghasilkan fitur item set. Oleh karena itu, berdasarkan pendapat pakar domain, Kombinasi dan standarisasi item fitur yang sama dalam set item fitur. Melakukan proses Standarisasi berdasarkan fitur item set, model vektorisasi dari dokumen asli yang dibuat sebelumnya. Kemudian menggunakan algoritma pengelompokan k-means untuk mengelompokkan hasil vektorisasi tersebut beberapa kali. Menurut metode Langkah ke-7, paper ini membangun ontologi masalah kualitas berdasarkan hasil Clustering (Xu, Dang, & Munro, 2018).

1. Original Corpus

Corpus yang dibuat ini berdasarkan dari dataset dokumen Spesifikasi Kebutuhan Perangkat Lunak (SKPL) pada Gambar 2.. Dimana dari hasil pengambilan corpus tersebut dimuat dalam bentuk Comma Separated Value (CSV) untuk diambil kolom fitur Statement yang berupa Kebutuhan fungsional dan kebutuhan non fungsional. Copus ini berguna untuk salah satu proses yang dilakukan untuk mendapatkan konstruksi ontologi.

Loading Dataset...
done in 0.011s.

ID	Requirement Statement
0	F01 Users can create a new diary.
1	F02 Users can add titles to the diary.
2	F03 Users can add weather to the diary.
3	F04 Users can add dates to the diary.
4	F06 Users can add hours to the diary.
5	F07 Users can add seconds to the diary.
6	F08 Users can save diaries.
7	F09 Users can add photos to the diary.
8	F10 Users can read the diary that has been created.
9	F11 Users can share diaries in the form of postcards.
10	F12 Users can save postcards.
11	F13 Users can change the postcard background color.
12	F14 Users can change the color of posts on the pos...
13	F15 Users can delete the diary.
14	F16 Users can edit the diary that has been created.
15	F17 Users can open a calendar that contains a diary.
16	F18 The system can open a diary editor.
17	NF01 The application has a high level of availabili...
18	NF02 Applications must have a high degree of flexib...
19	NF03 This application must have a high level of int...
20	NF04 This application has a high usability aspect a...
21	NF05 This application must have a response time val...
22	NF06 Applications must have a high level of perform...

Gambar 2. Original Copus

2. Text preprocessing

Pada corpus yang berupa kumpulan kalimat kebutuhan dari dokumen spesifikasi kebutuhan perangkat lunak (SKPL) dilakukan text preprocessing pada Gambar 3 seperti text lowering untuk mengubah kalimat ke dalam bentuk lowercase, punctuation removal untuk menghilangkan tanda baca, tokenisasi untuk memperoleh tiap tiap kata dari suatu kalimat, stopword removal untuk menghilangkan kata kata yang berupa stopwords dalam bahasa inggris, stemming untuk merubah kata ke dalam bentuk kata dasar. Sehingga dari proses text preprocessing didapatlah kata kata yang benar benar bersih dari noise.

Loading Original & Cleaned Text...
done in 0.014s.

ID	F01	F02	F03	F04	F06	F07	F08	F09	F10	F11	F12	F13	F14	F15	F16	F17	F18	NF01	NF02
ORIGINAL	Users can create a new diary.	Users can add titles to the diary.	Users can add weather to the diary.	Users can add dates to the diary.	Users can add hours to the diary.	Users can add seconds to the diary.	Users can save diaries.	Users can add photos to the diary.	Users can read the diary that has been created.	Users can share diaries in the form of postcards.	Users can save postcards.	Users can change the postcard background color.	Users can change the color of posts on the pos...	Users can delete the diary.	Users can edit the diary that has been created.	Users can open a calendar that contains a diary.	The system can open a diary editor.	The application has a high level of availabili...	Applications must have a high degree of flexib...
CLEANED	user creat new diary	user add titl diary	user add weather diary	user add date diary	user add hour diary	user add second diary	user save diary	user add photo diary	user read diary ha creat	user share diary form postcard	user save postcard	user chang postcard background color	user chang color post postcard	user delet diary	user edit diary ha creat	user open calendar contain diary	system open diary editor	applic ha high level avail oper continu seven ...	applic must high degree flexibility which must a...

Gambar 3. Hasil Text Preprocessing

3. Frequent Itemset Mining

Selanjutnya dalam Gambar 4. yang menyebutkan seputar Frequent Itemset Mining dimana, kita mengambil nilai frekuensi yang muncul dari dokumen tersebut. dengan proses metode Bag of Words (BOW). Dimana dalam proses tersebut kita membuat fungsi vektorisasinya. lalu dilanjutkan dengan fitur nama mana saja yang terkandung dalam corpus tersebut. Kemudian kita transformasikan dalam bentuk array untuk melihat sebuah angka biner antara 0 dan 1 dalam sebuah korpus tersebut. Lalu membuatnya menjadi sebuah matrix frequency.

Loading frequency matrix...
done in 0.001s.

	abl	abov	access	add	addit	age	android	applic	aspect	avail	background	calendar	chang	code	color	connect	contain	contin	creat	data
user creat new diari	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
user add titl diari	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
user add weather diari	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
user add date diari	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
user add hour diari	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
user add second diari	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
user save diari	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
user add photo diari	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
user read diari ha creat	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
user share diari form postcard	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
user save postcard	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
user chang postcard background color	0	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	0	0	0	0
user chang color post postcard	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0
user delet diari	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
user edit diari ha creat	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
user open calendar contain diari	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0
system open diari editor	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Gambar 4. Frequent Itemset Mining

4. Feature Item Set

Pada Gambar 5 terlihat nilai pembobotan dari setiap kata yang dilakukan pada proses Feature Item Set. Pembobotan ini dilakukan menggunakan pendekatan Term Frequency-Inverse Document Frequency (TF-IDF). TF menghitung jumlah kemunculan kata pada suatu dokumen, artinya semakin, kata itu sering muncul, maka pembobotannya akan semakin meningkat secara linear, misal kata 'user' muncul 2 kali pada dokumen, maka pembobotan TF akan memberikan nilai pembobotan 2, sedangkan IDF menghitung invers fungsi logaritma dari jumlah dokumen yang mengandung kata tersebut, artinya semakin banyak dokumen yang mengandung kata tersebut, bobot dari kata tersebut akan semakin turun menurut fungsi logaritmik, sehingga nilai bobot totalnya adalah hasil perkalian dari pembobotan TF dan pembobotan IDF.

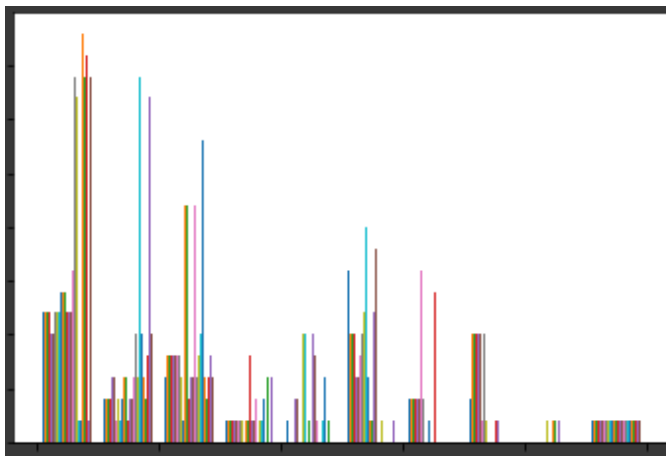
Loading frequency matrix...
done in 0.002s.

	abl	abov	access	add	addit	age	android	applic	aspect	avail	background	calendar	chang	code	color
user creat new diari	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
user add titl diari	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
user add weather diari	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
user add date diari	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
user add hour diari	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
user add second diari	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
user save diari	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
user add photo diari	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
user read diari ha creat	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
user share diari form postcard	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
user save postcard	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
user chang postcard background color	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.432485	0.000000
user chang color post postcard	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.432485	0.427765
user delet diari	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
user edit diari ha creat	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
user open calendar contain diari	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
system open diari editor	0.000000	0.494410	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000

Gambar 5. Pembobotan untuk setiap kata

5. Feature Items Combination & Standardization

Lalu selanjutnya kita menggunakan Feature Item Combination & Standardization sesuai dengan Gambar 6. menggunakan metode similaritas cosine yang digabung berdasarkan nilai hasil dari Term Frequency - Inverse Document Frequency. Sehingga didapatkan nilai kesamaan antara F01 sampai NF07. Hal ini kita dapatkan nilai Histogram sebagai berikut.



Gambar 6. Histogram Feature Item Combination & Standarization

6. Standardized Feature Item Set

Pada Gambar 7 terlihat matriks tingkat kemiripan dari masing masing kalimat kebutuhan yang ada pada dokumen. Nilai kemiripannya ini mempunyai rentang dari 0 sampai 1, 1 artinya kedua kebutuhan yang dibandingkan sangat mirip sedangkan 0 menandakan tidak ada kemiripan sama sekali

Loading frequency matrix...
done in 0.002s.

ID	F01	F02	F03	F04	F06	F07	F08	F09	F10	F11	F12	F13	F14	F15	F16	F17	F18
ID																	
F01	1.000000	0.553949	0.553949	0.553949	0.552648	0.552648	0.631320	0.553949	0.707396	0.503924	0.333006	0.263477	0.263477	0.633261	0.707396	0.506825	0.280837
F02	0.553949	1.000000	0.787655	0.787655	0.785806	0.785806	0.630071	0.787655	0.501049	0.502928	0.332347	0.262956	0.262956	0.632008	0.501049	0.505823	0.280281
F03	0.553949	0.787655	1.000000	0.787655	0.785806	0.785806	0.630071	0.787655	0.501049	0.502928	0.332347	0.262956	0.262956	0.632008	0.501049	0.505823	0.280281
F04	0.553949	0.787655	0.787655	1.000000	0.785806	0.785806	0.630071	0.787655	0.501049	0.502928	0.332347	0.262956	0.262956	0.632008	0.501049	0.505823	0.280281
F06	0.552648	0.785806	0.785806	0.785806	1.000000	0.783961	0.628592	0.785806	0.499872	0.501747	0.331567	0.262339	0.262339	0.630525	0.499872	0.504635	0.279623
F07	0.552648	0.785806	0.785806	0.785806	0.783961	1.000000	0.628592	0.785806	0.499872	0.501747	0.331567	0.262339	0.262339	0.630525	0.499872	0.504635	0.279623
F08	0.631320	0.630071	0.630071	0.630071	0.628592	0.628592	1.000000	0.630071	0.571031	0.573173	0.669000	0.299684	0.299684	0.720282	0.571031	0.576472	0.319429
F09	0.553949	0.787655	0.787655	0.787655	0.785806	0.785806	0.630071	1.000000	0.501049	0.502928	0.332347	0.262956	0.262956	0.632008	0.501049	0.505823	0.280281
F10	0.707396	0.501049	0.501049	0.501049	0.499872	0.499872	0.571031	0.501049	1.000000	0.455801	0.301205	0.238316	0.238316	0.572787	0.825586	0.458425	0.254018
F11	0.503924	0.502928	0.502928	0.502928	0.501747	0.501747	0.573173	0.502928	0.455801	1.000000	0.543693	0.430175	0.430175	0.574935	0.455801	0.460145	0.254971
F12	0.333006	0.332347	0.332347	0.332347	0.331567	0.331567	0.669000	0.332347	0.301205	0.543693	1.000000	0.554808	0.554808	0.379931	0.301205	0.304075	0.000000
F13	0.263477	0.262956	0.262956	0.262956	0.262339	0.262339	0.299684	0.262956	0.238316	0.430175	0.554808	1.000000	0.817017	0.300605	0.238316	0.240587	0.000000
F14	0.263477	0.262956	0.262956	0.262956	0.262339	0.262339	0.299684	0.262956	0.238316	0.430175	0.554808	0.817017	1.000000	0.300605	0.238316	0.240587	0.000000
F15	0.633261	0.632008	0.632008	0.632008	0.630525	0.630525	0.720282	0.632008	0.572787	0.574935	0.379931	0.300605	0.300605	1.000000	0.572787	0.578245	0.320411
F16	0.707396	0.501049	0.501049	0.501049	0.499872	0.499872	0.571031	0.501049	0.825586	0.455801	0.301205	0.238316	0.238316	0.572787	1.000000	0.458425	0.254018
F17	0.506825	0.505823	0.505823	0.505823	0.504635	0.504635	0.576472	0.505823	0.458425	0.460145	0.304075	0.240587	0.240587	0.578245	0.458425	1.000000	0.462911
F18	0.280837	0.280281	0.280281	0.280281	0.279623	0.279623	0.319429	0.280281	0.254018	0.254971	0.000000	0.000000	0.000000	0.320411	0.254018	0.462911	1.000000
NF01	0.000000	0.000000	0.000000	0.000000	0.103766	0.000000	0.000000	0.000000	0.098208	0.000000	0.000000	0.000000	0.000000	0.000000	0.098208	0.000000	0.000000
NF02	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.117392
NF03	0.154444	0.154139	0.154139	0.154139	0.153777	0.153777	0.175668	0.154139	0.139695	0.247663	0.180844	0.143086	0.143086	0.176208	0.139695	0.141027	0.000000

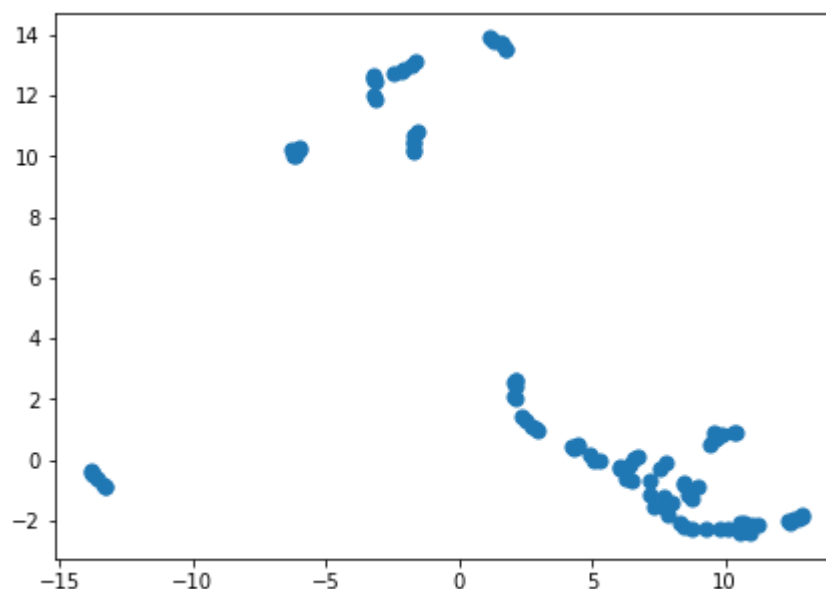
Gambar 7. Nilai kemiripan dari masing-masing kalimat kebutuhan

7. Document Vector

Pada nilai vektorisasi yangd dijelaskan pada Gambar 8. bahwa penggunaan word2vec berdasarkan perbandingan kata dasar yang berjumlah 104 dan nilai dari term setiap pernyataan kebutuhan berjumlah 26. Sehingga didapatkan nilai dari pendekatan untuk mendapatkan kata-kata yang muncul pada dokumen tersebut serta label pendekatan mana saja yang dekat dengan kata dasar tersebut serta buat nilai biner 0 dan 1 dari term tersebut.

Selanjutnya dilakukan proses training antara data X dan label Y menghasilkan nilai bentuk yang sama yaitu 666 bentuk dan panjang 104 kata.

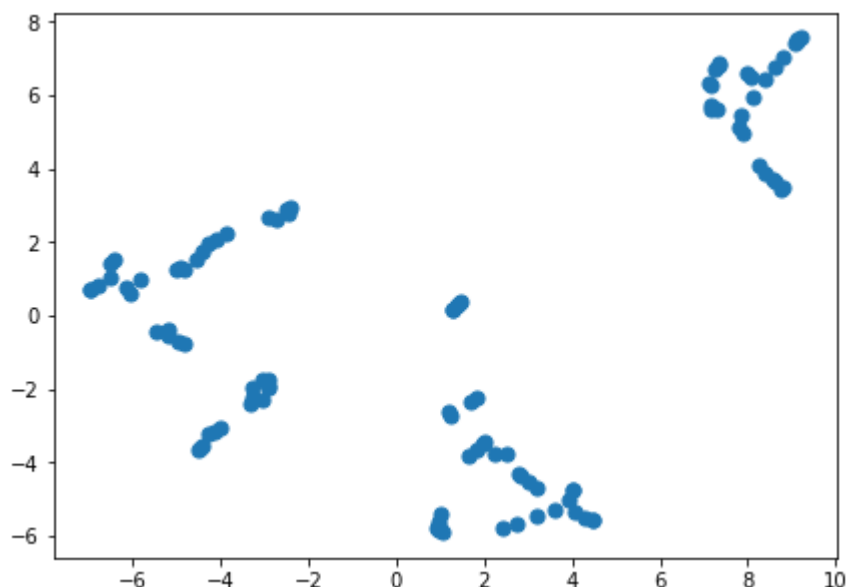
Kemudian dilanjutkan dengan pengujian pada weight dan backpropagation, untuk mendapatkan nilai prediksi vektor yang kita uji. Dari hasil beberapa training yang kita uji dari 10000 iterasi dan diambil nilai loses sekitar 2.26% dan kita ambil nilai vektorisasinya berdasarkan weight dan backpropagation tersebut. Lalu kita mengujinya bila kata coba cari fungsi pencarian kata yang terdekat dari 'abl' adalah 'run'. Setelah itu baru kita cari nilai jarak terdekat dari kedua kata tersebut, antara 'abl' dan 'run' sekitar 1.64 sedangkan bila kita bandingkan kata 'abl' dan 'abov' 4.57. sehingga bisa simpulkan dengan pendekatan word2vec ini kita bisa latih kata mana saja yang sama dengan berdekatan sesuai dengan pernyataan kebutuhan pada dokumen tersebut.



Gambar 8. Representasi Word2Vec

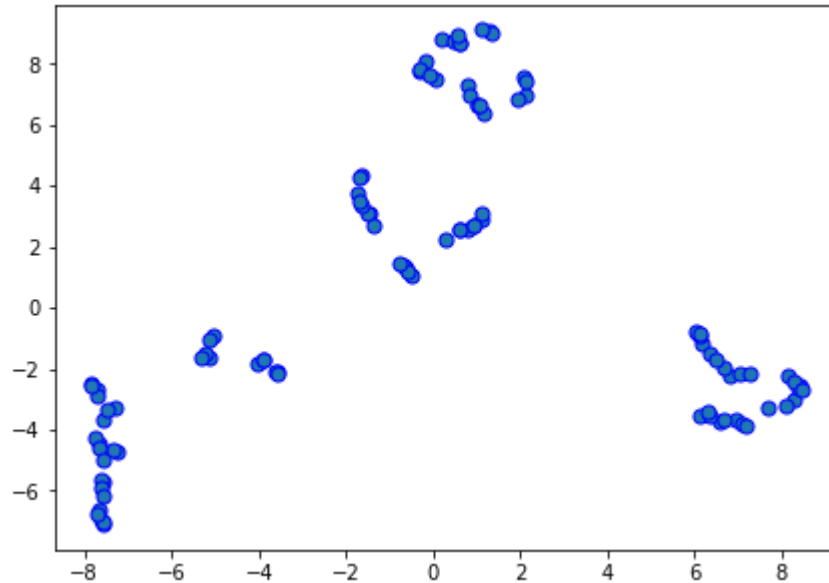
8. Clustering

Pada Gambar 9 terlihat visualisasi dari hasil cosine similarity yang didapat untuk setiap kalimat kebutuhan. pada gambar terlihat ada beberapa kalimat kebutuhan yang berdekatan dan ada juga yang berjauhan, dengan begini kita dapat melakukan clustering untuk mengelompokkan kalimat kalimat kebutuhan tersebut.



Gambar 9. Sebelum K-Means Clustering

Clustering yang dilakukan menggunakan algoritma K-Means Clustering, hasil dari proses clustering dapat dilihat pada Gambar 8. Representasi nilai kemiripan dari setiap kalimat kebutuhan terlihat semakin jelas.



Gambar 10. Setelah K-Means Clustering

9. Ontology Construction

Setelah dilakukan proses clustering yang telah dilakukan oleh K-Means clustering pada proses sebelumnya. Maka dilakukan proses Hirarki Clustering sesuai dengan Gambar 11. bisa dilihat kita membuat kluster sesuai dengan jarak diambil sesuai dengan linkage masing-masing simpul yaitu berupa beberapa kelas antara lain berdasarkan term dan masing-masing node yang terhubung.



Gambar 9. cluster.Hirarki

Referensi:

Xu, Z., Dang, Y., & Munro, P. (2018). Knowledge-driven intelligent quality problem-solving system in the automotive industry. *Advanced Engineering Informatics*. <https://doi.org/10.1016/j.aei.2018.08.013>

