**JURNAL**

# PENGELOMPOKAN SKRIPSI MENGGUNAKAN *SELF ORGANIZING MAPS CLUSTERING* (STUDI KASUS : PRODI TEKNIK INFORMATIKA UNIVERSITAS NUSANTARA PGRI KEDIRI)

## *CLASSIFICATION OF THESIS USING SELF ORGANIZING MAPS CLUSTERING (CASE STUDY: INFORMATICS ENGINEERING STUDY PROGRAM OF UNIVERSITY NUSANTARA PGRI KEDIRI)*


**Oleh:**

**IKA ZULAIKAH**

**12.1.03.02.0095**

**Dibimbing oleh :**

1. **M. Rizal Arief, S.T., M.Kom**
2. **Daniel Swanjaya, M.Kom**

**PROGRAM STUDI TEKNIK INFORMATIKA**

**FAKULTAS TEKNIK**

**UNIVERSITAS NUSANTARA PGRI KEDIRI**


**2017**

Artikel Skripsi
Universitas Nusantara PGRI Kediri

# SURAT PERNYATAAN
# ARTIKEL SKRIPSI TAHUN 2017

**Yang bertanda tangan di bawah ini:**

| | |
|---|---|
| Nama Lengkap | : Ika Zulaikah |
| NPM | : 12.1.03.02.0095 |
| Telepun/HP | : 085706852365 |
| Alamat Surel (Email) | : ikazula93@gmail.com |
| Judul Artikel | : Pengelompokan Skripsi Menggunakan *Self Organizing Maps Clustering* (Studi Kasus : Prodi Teknik Informatika Universitas Nusantara PGRI Kediri) |
| Fakultas – Program Studi | : Teknik – Teknik Informatika |
| Nama Perguruan Tinggi | : Universitas Nusantara PGRI Kediri |
| Alamat Perguruan Tinggi | : Jl. K.H. Ahmad Dahlan No. 76, Mojoroto, Kediri, Jawa Timur 64112 |

Dengan ini menyatakan bahwa :

a. artikel yang saya tulis merupakan karya saya pribadi (bersama tim penulis) dan bebas plagiarisme;
b. artikel telah diteliti dan disetujui untuk diterbitkan oleh Dosen Pembimbing I dan II.

Demikian surat pernyataan ini saya buat dengan sesungguhnya. Apabila di kemudian hari ditemukan ketidaksesuaian data dengan pernyataan ini dan atau ada tuntutan dari pihak lain, saya bersedia bertanggungjawab dan diproses sesuai dengan ketentuan yang berlaku.

| Mengetahui | | Kediri, 30 Januari 2017 |
|---|---|---|
| Pembimbing I | Pembimbing II | Penulis, |
| M. Rizal Arief, S.T., M.Kom<br>NIDN 0716027504 | Daniel Swanjaya, M.Kom<br>NIDN 0723098303 | Ika Zulaikah<br>NPM 12.1.03.02.0095 |



Ika Zulaikah | 12.1.03.02.0095
Fakultas Teknik – Teknik Informatika
simki.unpkediri.ac.id
|| 1||

# PENGELOMPOKAN SKRIPSI MENGGUNAKAN *SELF ORGANIZING MAPS CLUSTERING* (STUDI KASUS : PRODI TEKNIK INFORMATIKA UNIVERSITAS NUSANTARA PGRI KEDIRI)

Ika Zulaikah
12.1.03.02.0095
Fakultas Teknik – Program Studi Teknik Informatika
Email : ikazula93@gmail.com
M. Rizal Arief, S.T., M.Kom dan Daniel Swanjaya, M.Kom.
UNIVERSITAS NUSANTARA PGRI KEDIRI

## ABSTRAK

**Ika Zulaikah**: Pengelompokan Skripsi Menggunakan *Self Organizing Maps Clustering* (Studi Kasus : Prodi Teknik Informatika Universitas Nusantara PGRI Kediri, Skripsi, TI, FT UN PGRI Kediri, 2017.

*Text mining* berbasis pengelompokan skripsi menggunakan metode *clustering Self Organizing Maps* (SOM), dimana metode ini digunakan untuk meng-*cluster* skripsi pada Prodi Teknik Informatika Universitas Nusantara PGRI Kediri. Pada Prodi Teknik Informatika Universitas Nusantara PGRI Kediri data skripsi mahasiswa masih belum terinventaris dengan baik. Sehingga sering terjadi kesulitan dalam pencarian dan pengecekkan topik Skripsi.

Bagaimana mengimplementasikan metode *Self Organizing Maps* (SOM) pada aplikasi pengelompokan skripsi?

Pada sistem ini menggunakan abstrak sebagai *input*-an. Data-data abstrak tersebut telah tersimpan dalam *database MySQL*. Selanjutnya setiap kata dalam abstrak akan dipecah dengan menggunakan *Tokenizing* dan kata-kata yang telah dipecah akan di*steming*kan dan dikembalikan ke bentuk kata dasar. Proses selanjutnya yaitu pembobotan menggunakan metode TF-IDF. Dari proses TF-IDF tersebut akan menghasilkan bobot awal. Bobot awal dari proses TF-IDF selanjutnya akan dijadikan *input*-an untuk proses pembentukan jaringan SOM (*clustering*). Hasil dari proses tersebut berupa pelabelan pada masing-masing dokumen dan juga akan didapatkan bobot akhir.

Metode *Self Organizing Maps* bisa digunakan untuk mengelompokan skripsi dalam beberapa *cluster* dan hasil yang diperoleh dapat langsung ditampilkan.

**KATA KUNCI :** Metode *Self-Organizing Maps (SOM), Clustering,* Jaringan Syaraf Tiruan (JST), Kohonen.

## I. LATAR BELAKANG

Mahasiswa menyelesaikan Skripsi untuk mendapatkan gelar Strata 1 (S1). Skripsi tersebut nantinya akan dijadikan bahan pustaka penelitian dan sebagai koleksi untuk sebuah Program Studi (Prodi). Dalam sebuah Program Studi (Prodi) seiring dengan kelulusan Mahasiswa pada setiap tahunnya mengakibatkaan pertambahan banyaknya Skripsi. Tetapi Skripsi yang disimpan tersebut hanya diinventarisir di perpustakaan saja, dan hanya disediakan fitur pencarian Skripsi berdasarkan judul

saja, belum termasuk konten dari Skripsi tersebut.

Pada Prodi Teknik Informatika Universitas Nusantara PGRI Kediri data skripsi mahasiswa masih belum terinventaris dengan baik. Sehingga sering terjadi kesulitan dalam pencarian dan pengecekkan topik Skripsi. Untuk mengatasi masalah tersebut maka Penulis mengajukan Penelitian untuk mambantu Program Studi Teknik Informatika Universitas Nusantara PGRI Kediri untuk menginventaris dan mengelompokan Skripsi yang ada.

Aplikasi *text mining* merupakan aplikasi yang dapat mengolah data berupa tulisan. Penggunaan aplikasi *text mining* dapat menghemat waktu dan biaya, mengingat banyaknya jumlah dokumen yang ada. Aplikasi ini dapat mengelompokkan dan membagi dokumen ke dalam suatu kategori yang telah ditentukan sebelumnya. Setiap kategori dokumen memiliki sisi kemiripan yang nantinya dapat divisualisasikan dengan *cluster*.

*Clustering* dapat digunakan untuk membantu menganalisis dokumen dengan mengelompokkan secara otomatis dokumen yang memiliki kemiripan isi. *Clustering* adalah teknik yang digunakan untuk mengidentifikasi objek atau individu yang serupa dengan memperhatikan beberapa kriteria.

Pada penelitian sebelumnya pengelompokan dokumen sudah sering dilakukan. Seperti yang dilakukan oleh Ahmad Yusuf dan Tirta Priambadha pada tahun 2013 yang diberi judul "*Support Vector Machines* yang Didukung *K-Means Clustering* Dalam Klasifikasi Dokumen". Pada penelitian tersebut digunakan pengelompokan dokumen dengan *K-Means Clustering* sebelum melakukan klasifikasi mampu meningkatkan akurasi sebesar 0,5 % dan *recall* sebesar 0,4% pada data artikel yang digunakan. Klasifikasi akan mencapai akurasi terbaik pada parameter jumlah kelompok tertentu. Hal ini dipengaruhi dengan variasi data artikel yang digunakan. Jika menggunakan uji coba dengan data berbeda besar kemungkinan parameter jumlah kelompok dengan akurasi terbaik berbeda pula.

Pada penelitian ini digunakan algoritma *Self Organizing Maps* (SOM). Algoritma *Self Organizing Maps* (SOM) atau yang sering disebut dengan Jaringan Syaraf Tiruan

Kohonen, yang merupakan sebuah metode yang dikenalkan oleh Teuvo Kohonen pada tahun 1981. Jaringan Kohonen/SOM digunakan untuk mengelompokkan (*clustering*) data berdasarkan karakteristik/fitur-fitur data.

*Clustering* dengan menggunakan algoritma SOM sudah sering digunakan sebelumnya. Salah satunya adalah penelitian yang dilakukan oleh Ambarwati dan Edi Winarko pada tahun 2014 yang diberi judul "Pengelompokan Berita Indonesia Berdasarkan Histogram Kata Menggunakan *Self Organizing Maps*". Pada penelitian tersebut didapatkan hasil bahwa pada tahun 2007 sampai dengan tahun 2009 hasil pada *clustering* memiliki kecenderungan yang berbeda.

Berdasarkan penelitian yang dilakukan sebelumnya, penulis mengajukan judul Penelitian **"Pengelompokan Skripsi Menggunakan *Self Organizing Maps Clustering* (Studi Kasus : Prodi Teknik Informatika Universitas Nusantara PGRI Kediri)"** untuk membantu inventaris dokumen Skripsi yang ada pada Prodi Teknik Informatika Universitas Nusantara PGRI Kediri.

## II. METODE

*Self* Organizing *Map* (SOM) atau sering disebut *topology-preserving map* pertama kali diperkenalkan oleh Profesor Teuvo Kohonen pada tahun 1996. SOM merupakan salah satu teknik dalam *Neural Network* yang bertujuan untuk melakukan visualisasi data dengan cara mengurangi dimensi data melalui penggunaan *self-organizing neural network* sehingga manusia dapat mengerti *high-dimensional* data yang dipetakan dalam bentuk *low-dimensional*, data. Kohonen (Kohonen, 1989) menyatakan bahwa "Metode pembelajaran yang digunakan SOM adalah tanpa bimbingan dari suatu data input-target atau *unsupersived learning* yang mengasumsikan sebuah topologi yang terstruktur menjadikan unit unit kelas/*cluster*".

Han dan Kamber (Han dan Kamber, 2001) mengatakan bahwa "SOM merupakan metode berdasarkan model dari pendekatan jaringan syaraf tiruan". Dalam jaringan *Self Organizing Maps* (SOM), *neuron* target tidak diletakkan dalam sebuah baris seperti layaknya model jaringan syaraf tiruan (JST) yang lain. *Neuron* target diletakkan

dalam dua dimensi uang bentuk/topologinya dapat diatur. Topologi yang berbeda akan menghasilkan *neuron* sekitar *neuron* pemenang yang berbeda sehingga bobot yang dihasilkan juga akan berbeda. Pada *Self Organizing Maps* (SOM), perubahan bobot tidak hanya dilakukan pada bobot garis yang terhubung ke *neuron* pemenang saja, tetapi juga pada bobot garis ke *neuron-neuron* di sekitarnya. *Neuron* di sekitar neuron pemenang ditentukan berdasarkan jaraknya dari *neuron* pemenang.

Berikut ini adalah tahapan dalam algoritma pembelajaran SOM :

Step 0 : Inisialisasi bobot ($W_{ij}$)
Tentukan parameter jarak (R)
Tentukan *learning rate* ($\propto$ )

Step 1 : Jika kondisi *stop* salah, lakukan step 2 – 8

Step 2 : untuk semua *input vector* x lakukan step 3 – 5

Step 3 : untuk setiap j

$D(j) = \sum_i (W_{ij} - X_i)^2$ .....(2.1)

Step 4 : dapatkan j sehingga D(j) minimal

Step 5 : untuk semua j yang berada dalam jarak dan untuk semua i

$W_{ij}\ (baru) = W_{ij}\ (lama) + \propto (X_i - W_{ij}) lama$ (2.2)

Step 6 : *update learning rate*

Step 7 : *update* jarak

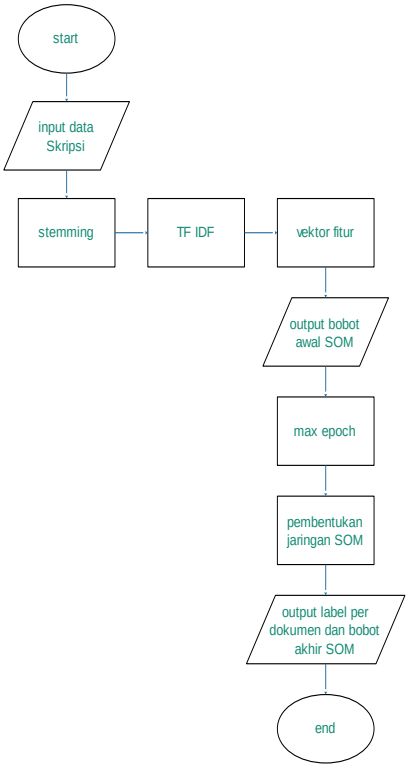
Step 8 : uji kondisi *stop*

$(Stop\ jika |\ W_{ij}\ (baru) - W_{ij}\ (lama) \leq \varepsilon)$...(2.3)

## III. HASIL DAN KESIMPULAN

### A. Hasil

Dari penjelasan sebelumnya, berikut ini merupakan gambaran dari alur sistem yang telah dibuat :




Gambar 3.1. *Flowchart* Sistem

Dalam *flowchart* Gambar 3.1 menunjukkan bahwa *user* melakukan *input* data Skripsi terlebih dahulu, kemudian data tersebut masuk ke proses *stemming*. *Stemming* dilakukan dengan menggunakan *Algoritma Tala*.

Proses selanjutnya yaitu pembobotan menggunakan metode TF-IDF. Dari proses TF-IDF tersebut akan menghasilkan bobot awal. Bobot awal dari proses TF-IDF selanjutnya akan dijadikan *input*-an untuk proses pembentukan jaringan SOM (*clustering*). Hasil dari proses tersebut berupa pelabelan pada masing-masing dokumen dan juga akan didapatkan bobot akhir.

Dari gambaran *flowchart* di atas, berikut ini simulasi perhitungan dari aplikasi yang telah dibuat :

Misalkan terdapat *Feature Vektor* yang diperoleh dari *Vektor* Dokumen Skripsi dan *Vektor term* sebagai berikut :

Tabel 3.1. Tabel vektor judul dokumen Skripsi dan vektor *term*:

| Vektor Judul Dokumen | Judul Dokumen Skripsi |
|---|---|
| D1 | Dokumen 1 |
| D2 | Dokumen 2 |
| D3 | Dokumen 3 |
| D4 | Dokumen 4 |
| D5 | Dst |

| Vektor *Terms* | *Terms* |
|---|---|
| t1 | Term1 |
| t2 | Term2 |
| t3 | Term3 |
| t4 | Term4 |
| t5 | dst |

Tabel 3.2. Tabel *feature vektor*

| **Vektor *term*** | **Vektor Judul Dokumen** | | |
|---|---|---|---|
| | **D1** | **D2** | **D3** |
| t1 | 1 | 1 | 0 |
| t2 | 0 | 1 | 0 |
| t3 | 1 | 1 | 1 |
| t4 | 0 | 0 | 1 |
| t5 | 1 | 1 | 0 |

Dari data *Feature Vektor* maka dapat dibuat *feature Matrik* (FM) sebagai berikut :

$$FM = \begin{bmatrix} X_{11} & X_{12} & X_{13} & X_{14} & X_{15} \\ X_{21} & X_{22} & X_{23} & X_{24} & X_{25} \\ X_{31} & X_{32} & X_{33} & X_{34} & X_{35} \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 \end{bmatrix}$$

*Training Feature Matrik* menggunakan bobot secara nilai random antara 0-1 untuk proses pelatihan :

$$W = \begin{bmatrix} W_{11} & W_{12} & W_{13} & W_{14} & W_{15} & W_{16} \\ W_{21} & W_{22} & W_{23} & W_{24} & W_{25} & W_{26} \\ W_{31} & W_{32} & W_{33} & W_{34} & W_{35} & W_{36} \\ W_{41} & W_{42} & W_{43} & W_{44} & W_{45} & W_{46} \\ W_{51} & W_{52} & W_{53} & W_{54} & W_{55} & W_{56} \end{bmatrix}$$

$$= \begin{bmatrix} 0{,}31725 & 0{,}29173 & 0{,}67431 & 0{,}42315 & 0{,}66743 & 0{,}15432 \\ 0{,}79437 & 0{,}22674 & 0{,}38742 & 0{,}81129 & 0{,}25437 & 0{,}87321 \\ 0{,}84216 & 0{,}84654 & 0{,}97123 & 0{,}34665 & 0{,}69214 & 0{,}54276 \\ 0{,}43272 & 0{,}79431 & 0{,}86942 & 0{,}91273 & 0{,}46511 & 0{,}59637 \\ 0{,}94325 & 0{,}24569 & 0{,}47316 & 0{,}64329 & 0{,}11773 & 0{,}11426 \end{bmatrix}$$

*Set Parameter Learning rate* $(a)$ secara manual (random antara 0 – 1) yaitu 0,6; dengan tiap kenaikan *epoch* akan diset 0,5 x $(a)$:

Cari jarak terdekat *neuron* dengan *Euclidean Distance* :

$$d^2 = (euclidean\ distance)^2$$

$$= \sum_{k=l}^{n} \left(w_{j,k} - X_{i,k}\right)^2$$

**Contoh**

**Training data ke 1 Iterasi (*epoch*) 1 :**

$d^2 1 = (W_{11} - X_{11})^2 + (W_{21} - X_{12})^2 + (W_{31} - X_{13})^2 + (W_{41} - X_{14})^2 + (W_{51} - X_{15})^2$

$=(0{,}31725 - 1)^2 + (0{,}79437 - 0)^2 + (0{,}84216 - 1)^2 + (0{,}43272 - 0)^2 + (0{,}94325 - 1)^2$

= 1,31255

$d^2 2 = (W_{12} - X_{11})^2 + (W_{22} - X_{12})^2 + (W_{32} - X_{13})^2 + (W_{42} - X_{14})^2 + (W_{52} - X_{15})^2$

$=(0{,}29173 - 1)^2 + (0{,}22674 - 0)^2 + (0{,}84654 - 1)^2 + (0{,}79431 - 0)^2 + (0{,}24569 - 1)^2$

= 1,77652

$d^2 3 = (W_{13} - X_{11})^2 + (W_{23} - X_{12})^2 + (W_{33} - X_{13})^2 + (W_{43} - X_{14})^2 + (W_{53} - X_{15})^2$

$=(0{,}67431 - 1)^2 + (0{,}38742 - 0)^2 + (0{,}97123 - 1)^2 + (0{,}86942 - 0)^2 + (0{,}47316 - 1)^2$

= 1,29045

$d^2 4 = (W_{14} - X_{11})^2 + (W_{24} - X_{12})^2 + (W_{34} - X_{13})^2 + (W_{44} - X_{14})^2 + (W_{54} - X_{15})^2$

$=(0{,}42315 - 1)^2 + (0{,}81129 - 0)^2 + (0{,}34665 - 1)^2 + (0{,}91273 - 0)^2 + (0{,}64329 - 1)^2$

= 2,37813

$d^2 5 = (W_{15} - X_{11})^2 + (W_{25} - X_{12})^2 + (W_{35} - X_{13})^2 + (W_{45} - X_{14})^2 + (W_{55} - X_{15})^2$

$=(0{,}66743 - 1)^2 + (0{,}25437 - 0)^2 + (0{,}69214 - 1)^2 + (0{,}46511 - 0)^2 + (0{,}11773 - 1)^2$

= **<u>1,26481 (Pemenang)</u>**

$d^2 6 = (W_{16} - X_{11})^2 + (W_{26} - X_{12})^2 + (W_{36} - X_{13})^2 + (W_{46} - X_{14})^2 + (W_{56} - X_{15})^2$

$=(0{,}15432 - 1)^2 + (0{,}87321 - 0)^2 + (0{,}54276 - 1)^2 + (0{,}59637 - 0)^2 + (0{,}11426 - 1)^2$

= 2,82693

Maka bobot / *weight* minimum pada $d^2 5$, *update* dengan rumus

$$W_{ij}\ (baru) = W_{ij}(lama) + \propto (X_i - W_{ij})lama$$

Dan diperoleh hasil sebagai berikut :

$W_{15}$ (baru) = 0,86697

$W_{25}$ (baru) = 0,10175

$W_{45}$ (baru) = 0,18604

$W_{35}$ (baru) = 0,87686

$W_{55}$ (baru) = 0,64709

Jadi, bobot baru iterasi (*epoch*) ke 1 data ke 1 :

$$W = \begin{bmatrix} W_{11} & W_{12} & W_{13} & W_{14} & W_{15} & W_{16} \\ W_{21} & W_{22} & W_{23} & W_{24} & W_{25} & W_{26} \\ W_{31} & W_{32} & W_{33} & W_{34} & W_{35} & W_{36} \\ W_{41} & W_{42} & W_{43} & W_{44} & W_{45} & W_{46} \\ W_{51} & W_{52} & W_{53} & W_{54} & W_{55} & W_{56} \end{bmatrix}$$

$$= \begin{bmatrix} 0{,}31725 & 0{,}29173 & 0{,}67431 & 0{,}42315 & 0{,}86697 & 0{,}15432 \\ 0{,}79437 & 0{,}22674 & 0{,}38742 & 0{,}81129 & 0{,}10175 & 0{,}87321 \\ 0{,}84216 & 0{,}84654 & 0{,}97123 & 0{,}34665 & 0{,}87686 & 0{,}54276 \\ 0{,}43272 & 0{,}79431 & 0{,}86942 & 0{,}91273 & 0{,}18604 & 0{,}59637 \\ 0{,}94325 & 0{,}24569 & 0{,}47316 & 0{,}64329 & 0{,}64709 & 0{,}11426 \end{bmatrix}$$

Catatan :

*Bobot baru iterasi data ke 1 ini digunakan untuk bobot awal data ke 2*

**Misalkan ini adalah hasil update bobot :**

Bobot baru iterasi (*epoch*) ke1 data ke $i_1$ :

$$W\ (bobot) = \begin{bmatrix} 0{,}31725 & 0{,}29173 & 0{,}67431 & 0{,}42315 & 0{,}86697 & 0{,}15432 \\ 0{,}79437 & 0{,}22674 & 0{,}38742 & 0{,}81129 & 0{,}10175 & 0{,}87321 \\ 0{,}84216 & 0{,}84654 & 0{,}97123 & 0{,}34665 & 0{,}87686 & 0{,}54276 \\ 0{,}43272 & 0{,}79431 & 0{,}86942 & 0{,}91273 & 0{,}18604 & 0{,}59637 \\ 0{,}94325 & 0{,}24569 & 0{,}47316 & 0{,}64329 & 0{,}64709 & 0{,}11426 \end{bmatrix}$$

Bobot baru iterasi (*epoch*) ke 1 data ke $i_2$ :

$$W\ (bobot) = \begin{bmatrix} 0{,}72690 & 0{,}29173 & 0{,}67431 & 0{,}42315 & 0{,}86697 & 0{,}15432 \\ 0{,}91775 & 0{,}22674 & 0{,}38742 & 0{,}81129 & 0{,}10175 & 0{,}87321 \\ 0{,}93686 & 0{,}84654 & 0{,}97123 & 0{,}34665 & 0{,}87686 & 0{,}54276 \\ 0{,}17309 & 0{,}79431 & 0{,}86942 & 0{,}91273 & 0{,}18604 & 0{,}59637 \\ 0{,}97730 & 0{,}24569 & 0{,}47316 & 0{,}64329 & 0{,}64709 & 0{,}11426 \end{bmatrix}$$

Bobot baru iterasi (*epoch*) ke 1 data ke $i_3$ :

$$W\ (bobot) = \begin{bmatrix} 0{,}72690 & 0{,}11669 & 0{,}67431 & 0{,}42315 & 0{,}86697 & 0{,}15432 \\ 0{,}91775 & 0{,}09070 & 0{,}38742 & 0{,}81129 & 0{,}10175 & 0{,}87321 \\ 0{,}93686 & 0{,}93862 & 0{,}97123 & 0{,}34665 & 0{,}87686 & 0{,}54276 \\ 0{,}17309 & 0{,}91772 & 0{,}86942 & 0{,}91273 & 0{,}18604 & 0{,}59637 \\ 0{,}97730 & 0{,}09828 & 0{,}47316 & 0{,}64329 & 0{,}64709 & 0{,}11426 \end{bmatrix}$$

Catatan :

*Bobot epoch 1 terakhir ini digunakan untuk bobot awal epoch 2*

Sebelum pindah ke *epoch* 2 lakukan pengecekan bobot sebagai berikut :

Bobot awal *epoch* 1 :

$$W\ (bobot) = \begin{bmatrix} 0{,}31725 & 0{,}29173 & 0{,}67431 & 0{,}42315 & 0{,}66743 & 0{,}15432 \\ 0{,}79437 & 0{,}22674 & 0{,}38742 & 0{,}81129 & 0{,}25437 & 0{,}87321 \\ 0{,}84216 & 0{,}84654 & 0{,}97123 & 0{,}34665 & 0{,}69214 & 0{,}54276 \\ 0{,}43272 & 0{,}79431 & 0{,}86942 & 0{,}91273 & 0{,}46511 & 0{,}59637 \\ 0{,}94325 & 0{,}24569 & 0{,}47316 & 0{,}64329 & 0{,}11773 & 0{,}11426 \end{bmatrix}$$

Bobot awal *epoch* 2 :

$$W\ (bobot) = \begin{bmatrix} 0{,}72690 & 0{,}11669 & 0{,}67431 & 0{,}42315 & 0{,}86697 & 0{,}15432 \\ 0{,}91775 & 0{,}09070 & 0{,}38742 & 0{,}81129 & 0{,}10175 & 0{,}87321 \\ 0{,}93686 & 0{,}93862 & 0{,}97123 & 0{,}34665 & 0{,}87686 & 0{,}54276 \\ 0{,}17309 & 0{,}91772 & 0{,}86942 & 0{,}91273 & 0{,}18604 & 0{,}59637 \\ 0{,}97730 & 0{,}09828 & 0{,}47316 & 0{,}64329 & 0{,}64709 & 0{,}11426 \end{bmatrix}$$

$$\text{MSE} = \sqrt[2]{(Bobot\ awal\ epoch\ 1 - Bobot\ awal\ epoch\ 2)^2}$$

$$= \sqrt[2]{0{,}26057 + 0{,}09458 + 0{,}00000 + 0{,}00000 + 0{,}45533 + 0{,}00000}$$

$$\text{MSE} = \sqrt[2]{0{,}81049}$$

$$\text{MSE} = 0{,}90027$$

**Keterangan :**

*Epoch* akan berhenti jika nilai MSE = 0,001 (set manual program)

Jika MSE belum menunjukkan 0,001, lakukan perhitungan untuk iterasi (*epoch*) ke 2. Langkah perhitungan pada iterasi (*epoch*) ke 2 sama dengan langkah perhitungan pada iterasi (*epoch*) ke 1. Setelah melakukan perhitungan jarak terdekat pada iterasi (*epoch*) ke 2, dihasilkan bobot / weight minimum pada $d^2 5$ muncul sebagai pemenang. Tahap selanjutnya lakukan update pada tersebut. Dan selanjutnya akan menghasilkan bobot baru iterasi (*epoch*) ke 2 data ke 1.

Setelah beberapa iterasi (*epoch*) dengan MSE 0,001, maka didapat bobot akhir baru (disimpan) yang akan digunakan untuk pengenalan data selanjutnya (*Searching User*).

**<u>Berikut ini adalah hasil dari bobot akhir :</u>**

$$W\ (bobot) = \begin{bmatrix} 0{,}99955 & 0{,}00048 & 0{,}67431 & 0{,}42315 & 0{,}99978 & 0{,}15432 \\ 0{,}99986 & 0{,}00037 & 0{,}38742 & 0{,}81129 & 0{,}00017 & 0{,}87321 \\ 0{,}99990 & 0{,}99975 & 0{,}97123 & 0{,}34665 & 0{,}99980 & 0{,}54276 \\ 0{,}00028 & 0{,}99966 & 0{,}86942 & 0{,}91273 & 0{,}00030 & 0{,}59637 \\ 0{,}99963 & 0{,}00040 & 0{,}47316 & 0{,}64329 & 0{,}99942 & 0{,}11426 \end{bmatrix}$$

Dari data sebelumnya diketahui bahwa data yang ada sebagai berikut :

$i_1$ = (1, 0, 1, 0, 1)

$i_2$ = (1, 1, 1, 0, 1)

$i_3$ = (0, 0, 1, 1, 0)

dari ke 3 data tersebut dicari jarak terdekatnya dengan *Euclidean Distance* adalah sebagai berikut :

| (sample $i_1$) | (sample $i_2$) | (sample $i_3$) |
|---|---|---|
| $d^2 1$ = 0,99972 | $d^2 1$ = **0,00000 (win)** | $d^2 1$ = 3,99752 |
| $d^2 2$ = 2,99756 | $d^2 2$ = 3,99682 | $d^2 2$ = **0,00000 (win)** |
| $d^2 3$ = 1,29045 | $d^2 3$ = 1,51561 | $d^2 3$ = 0,84655 |
| $d^2 4$ = 2,37813 | $d^2 4$ = 1,75555 | $d^2 4$ = 1,68555 |
| $d^2 5$ = **0,00000 (win)** | $d^2 5$ = 0,99966 | $d^2 5$ = 2,99780 |
| $d^2 6$ = 2,82693 | $d^2 6$ = 2,08051 | $d^2 6$ = 1,17135 |

Maka dari hasil perhitungan *Euclidean Distance* di atas dapat disimpulkan data telah ter*cluster* :

$i_1$ masuk kelompok *cluster* 5, $i_2$ masuk kelompok *cluster* 1, dan $i_3$ masuk kelompok *cluster* 2.

## B. Kesimpulan

Melalui pembahasan yang telah diuraikan pada bab sebelumnya, maka dapat ditarik kesimpulan yaitu:

Metode *Self Organizing Maps* bisa digunakan dalam aplikasi *text mining* untuk mengelompokkan skripsi dalam beberapa *cluster* dan hasil yang diperoleh dapat langsung ditampilkan.

## IV. DAFTAR PUSTAKA

Adelheid, Andrea & Nst, Khairil. 2012. *Buku Pintar Menguasai PHP MySQL*. Jakarta: Mediakita

Al, Henderi Et. 2008. *Pengertian Use Case Diagram Deskripsi* (Online), tersedia: http://www.jelajahiinternet.com/2014/10/pengertian-use-case-diagram-deskripsi.html, 21 Januari 2016.

Ambarwati & Winarko, Edi. 2013. *Pengelompokan Berita Indonesia Berdasarkan Histogram Kata Menggunakan Self-Organizing Map*, 8 (1). (Online), tersedia: http://genome.tugraz.at/MedicalInformatics2/SOM.pdf, diunduh 5 Januari 2016.

Darmono & Hasan, Ani. M. 2002. *Menyelesaikan Skripsi Dalam Satu Semester*. Jakarta: PT Gramedia Widiasarana Indonesia.

Han, J., M. Kamber, A. Tung. 2001. *Spatial Clustering Methods In Data Mining*. Geographic Data Mining and Knowledge Discovery. H. Miller and J. Han. Londor, Taylor & Fancis:188-217

Irwansyah, Edy & Faisal, Muhammad. 2015. *Advanced Clustering Teori dan Aplikasi*. Yogyakarta: Deepublish.

Kohonen, T. 1989. *Self-organization and Associztive Memory*. Third Edition. Berlin: Springer-Verlag.

Kohonen, Teuvo. 1990. *The Self Organizing Map*. IEEE.

Nofriadi. 2015. *Java Fundamental dengan Netbeans 8.0.2*, (Online). Yoyakarta : Deepublish.

Rahayu, Minto. 2009. *Bahasa Indonesia Di Perguruan Tinggi*. Jakarta : PT Gramedia Widiasarana Indonesia.

Sugiyono. 2008. *Metode Penelitian Pendidikan*. Bandung: Alfabeta.

Supardi, Yuniar. 2007. *Pemrograman Database dengan Java dan MySQL*. Jakarta: PT Elex Media Komputindo.

Supriyatno. 2010. *Pemrograman Database menggunakan Java dan MySQL untuk Pemula*. Jakarta: Mediakita.

Tala F. Z. 2004. *A Study of Stemming Effects on Information Retrieval in Bahasa Indonesia*. Institute for Logic, Language and Computation Universiteit van Amsterdam The Netherlands.

Tan, P. et al. 2006. *Intruduction to Data Mining*. Boston: Pearson Education.

Utomo, Mardi Siswo. 2013. *Implementasi Stemmer Tala pada Aplikasi Berbasis Web*, 18 (1). (Online), tersedia: http://www.unisbank.ac.id/ojs/index.php/fti1/article/view/1673/607, diunduh 25 Januari 2016.

Wahyuningtyas, Ferida Kristin. 2014. *Rekomendasi Resep Masakan Menggunakan Metode Fuzzy dan Self Organizing Map (SOM)*. Skripsi. Kediri : Universitas Nusantara PGRI.

Westra, P.1991. *Pengertian dan Definisi Sekripsi* (Online), tersedia: http://www.kumpulandefinisi.com/2015/07/pengertian-dan-definisi-sekripsi_97.html, 13 Januari 2016.

Yusuf, Ahmad & Priambadha, Tirta. 2013, *Support Vector Machines yang Didukung K-Means Clustering Dalam Klasifikasi Dokumen*, 11 (1). (Online), tersedia: http://juti.if.its.ac.id/index.php/juti/article/view/15/14, diunduh 12 Januari 2016.