Kluster *Bag-of-Word* Menggunakan Weka

Tari Mardiana[1], Rudy Dwi Nyoto[2]
[1]Jurusan Teknik Elektro dan Teknologi Informasi, UGM – Yogyakarta
[2]Program Studi Teknik Informatika, Universitas Tanjungpura - Pontianak
e-mail: tari.mardiana@gmail.com, rudy_dn@yahoo.com

***Abstrak*— Dalam bidang pengolahan bahasa alami dan sistem temu balik informasi, representasi sebuah data teks sangat penting untuk mendukung proses analisis data statistik di dalamnya. Data teks dengan bentuk tidak terstruktur dapat direpresentasikan secara sederhana menggunakan sekumpulan set kata yang disebut *bag-of-words* dan belum memiliki label atau kelas tertentu. Data *unsupervised* atau objek-objek yang belum memiliki label dapat dikelompokan menggunakan klustering berdasarkan kemiripan satu objek dengan objek lain. Artikel ini membahas perbandingan hasil pengelompokan *unsupervised* data menggunakan algoritma kluster yang tersedia pada *tools* Weka, yaitu *SimpleKMeans, X-Means, dan Farthest First. SimpleKMeans dan XMeans* digunakan untuk mengolah dataset dan mengelompokan berdasarkan jumlah kluster tetap yang digunakan, sedangkan *Farthest First* akan meletakan semua pusat kluster pada titik terjauh dari pusat kluster yang sudah ada untuk mengelompokan data. Dataset berasal dari UCI *machine learning* dengan menggunakan 3 koleksi data, yaitu Enron Email, NIPS Proceedings, dan Daily Kos Blog entries. Performa dataset diuji dengan berbagai masukan parameter yang berbeda meliputi jumlah kluster hingga evaluasi *sum squared error* (SSE), serta iterasi selama proses pengolahan data. Hasil penelitian diharapkan dapat dijadikan acuan untuk menentukan algoritma dan parameter yang sesuai untuk melakukan pengelompokan data yang tidak memiliki label.**

Kata Kunci—bag of words*, kluster, label, teks, *unsupervised

I. PENDAHULUAN

Klustering merupakan salah satu tugas utama dalam data mining selain regresi dan klasifikasi. Klustering merupakan *unsupervised learning* yang melakukan pembelajaran secara tidak terbimbing. Klustering adalah proses pengelompokan sekumpulan objek (disebut kluster) yang tidak memiliki label apapun, sehingga pengelompokan dilakukan hanya dengan melihat kemiripan satu objek dengan objek lainnya. Klustering sering digunakan dalam analisis data statistik yang melibatkan banyak bidang, meliputi *machine learning*, *image analysis*, pengenalan pola, sistem temu balik informasi (IR), dan bioinformatika [1].

Pemilihan pendekatan dan algoritma kluster harus disesuaikan dengan data dan tujuan dari klustering itu sendiri. Ada dua pendekatan umum dalam klustering yaitu *Partitional clustering* dan *Hierarchical clustering*. Selain dua pendekatan tersebut, terdapat pendekatan lain yaitu *Density-based*, *Grid-based*, dan *Model-based clustering* [2]. Berikut penjelasan beberapa pendekatan dan algoritma yang digunakan dalam klustering.

1. *Partitioning-based*
   Klustering dengan partisi akan membagi objek ke dalam subset (kluster) sehingga tiap satu data objek akan dikelompokan tepat dalam satu subset saja. Contoh: *k*-means, Fuzzy *c*-means (FCM), *k*-medoids, CLARA, CLARANS, PAM
2. *Hierarchical-based*
   Pendekatan hirarki akan mengelompokkan objek yang mirip dengan meletakannya pada hirarki yang berdekatan dan sebaliknya objek yang tidak mirip akan berada pada hirarki yang berjauhan. Contoh: BIRCH, AGNES, DIANA
3. *Density-based*
   Pendekatan ini cocok digunakan untuk klustering objek yang memiliki *noise* dan *outlier*, namun sulit menemukan kluster yang memiliki bentuk yang berubah-ubah. Contoh: DBSCAN, OPTICS, DENCLUE, CLIQUE
4. *Grid-based*
   Pendekatan yang mebagi objek ke dalam sejumlah sel yang membentuk struktur grid. Contoh: STING, WaveKluster
5. *Model-based clustering*
   Pendekatan didasarkan pada dugaan-dugaan mengenai sebuah model untuk tiap kluster, kemudian mencari data objek yang sesuai untuk model tersebut. Contoh: SOM, COBWEB

Dalam artikel ini akan dibahas bagaimana melakukan pengelompokan data kluster menggunakan Weka dan membandingkan performa algoritma klustering yang tersedia berdasarkan beberapa pendekatan yang sudah dijelaskan sebelumnya. Weka sebagai *data mining tools* memiliki keuntungan yaitu lebih baik dalam tampilan serta mendukung banyak algoritma klustering [3].

II. URAIAN PENELITIAN

A. *Bag of words* (BoW)

Semua dokumen dapat direpresentasikan secara sederhana menggunakan *Bag-of-words* (BoW). BoW adalah sebuah model yang merepresentasikan objek secara global misalnya kalimat teks atau dokumen sebagai bag (*multiset*) kata tanpa memperdulikan tata bahasa bahkan urutan kata untuk menjaga keanekaragamannya [4]. Dengan kata lain, BoW merupakan kumpulan kata-kata unik dalam dokumen. Contoh sederhana

pembentukan *bag-of-words* untuk teks dokumen sebagai berikut

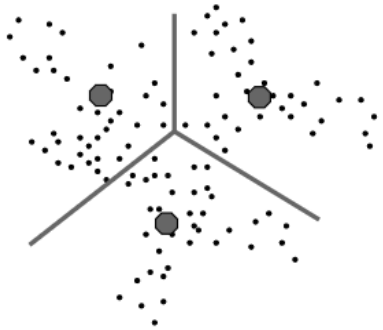
```
Teks: Sari senang membaca novel, Ina juga
penggemar novel  remaja.
```

Teks diatas dapat disusun menjadi BoW, dengan menggunakan kata unik yang direpresentasikan sekali saja sehingga membentuk urutan yang berbeda kemudian dihitung frekuensi kemunculannya.

Tabel 1.
Contoh pembentukan *bag-of-words*

No	Kata	Distribusi frekuensi
1	Sari	1
2	Senang	1
3	Membaca	1
4	Novel	2
5	Ina	1
6	Juga	1
7	Penggemar	1
8	Remaja	1

Distribusi frekuensi kata dapat dibandingkan dan digunakan untuk menilai kemiripan antara dua atau lebih dokumen dengan cara menghitung jarak keduanya. Teknik umum yang digunakan antara lain *Euclidean*, *Manhattan*, dan *Cosine distances* [5]. Dalam klustering BoW model, tiap pusat kluster (*centroid*) didefinisikan sebagai bentuk visual dari kata yang dapat dikelompokan berdasarkan kemiripannya.



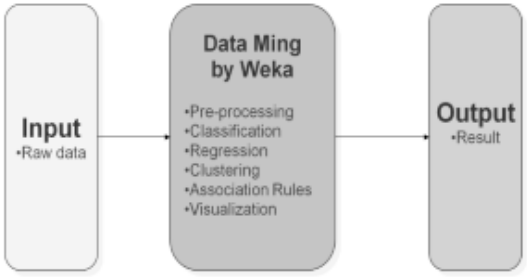
Gambar 1. *Contoh klustering model BoW*

B. WEKA

WEKA (*Waikato Environment for Knowledge Analysis*) [6] merupakan perangkat lunak data mining yang dikembangkan oleh Universitas Waikato, New Zealand. Diimplementasikan pertama kali pada tahun 1997 dan mulai menjadi *open source* pada tahun 1999. Hingga saat ini Weka sudah mencapai versi 3.6.11 dengan berbagai pengembangan dari versi pertama 3.3. Ditulis dalam bahasa pemrograman Java, Weka juga didukung oleh GUI yang sangat baik dan *user friendly*, dapat mengolah berbagai file data seperti *.csv dan .*arff serta memiliki fitur utama seperti *data pre-processing tools*, *learning algorithms* dan berbagai metode evaluasi [7]. Selain itu, Weka juga dapat memberikan hasil dalam bentuk visual, seperti tabel dan kurva.

Gambar 2. *Tampilan awal Weka Tools*

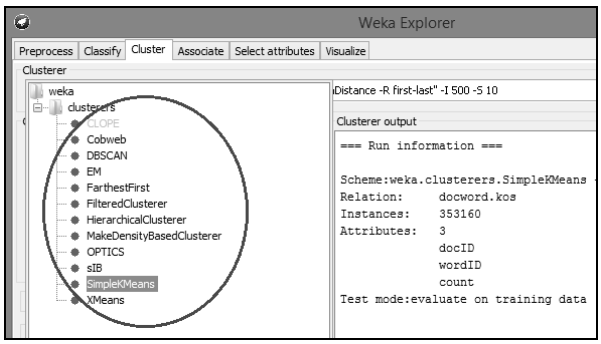
Weka terdiri dari beberapa tools yang dapat digunakan untuk melakukan tugas *pre-processing data*, *classification*, *regression*, klustering, *association rules*, dan visualisasi [8].



Gambar 3. *Tugas data mining menggunakan Weka [9]*

Proses kluster digunakan untuk melakukan identifikasi pengelompokan dari beberapa kejadian dalam dataset agar dapat menghasilkan informasi yang dapat dianalisis oleh pengguna. Ada beberapa pilihan dalam *sub-menu* kluster weka antara lain: *use training set*, *supplied test set percentage split*, dan *classes to cluster evaluation* yang digunakan untuk membandingkan seberapa baik data yang dibandingkan tanpa diberikan kelas antar data. Dalam proses pengelompokan di Weka, beberapa atribut juga dapat diabaikan dengan tujuan hanya menggunakan data yang memberikan hasil spesifik saja dan baik digunakan untuk dataset besar yang banyak atribut.

Untuk membantu proses klustering, terdapat beberapa algoritma pengelompokan (klusterer) yang dapat digunakan untuk pengujian. Tidak semua algoritma cocok diterapkan pada dataset, tergantung atribut yang dimiliki, ada tidaknya *noise* dan outliers serta tujuan yang ingin dicapai [10].



Gambar 4. *Berbagai klusterer untuk pengolahan dataset*

Dalam penelitian [11], *k*-means merupakan salah satu algoritma yang cocok digunakan untuk dataset dalam skala

besar karena kompleksitas algoritma yang tinggi. Namun jika dilihat dari sisi waktu proses yang lebih cepat, *Farthest First* lebih direkomendasikan untuk mengolah data yang besar. Sedangkan *XMeans* lebih memudahkan tugas klustering dibandingkan *k-M*eans, dapat mengatasi masalah pemilihan jumlah kluster *k* yang digunakan karena algoritma ini tidak membutuhkan *k* sebagai *input* namun lebih mengarah pada *distance function* yang digunakan [12].

1.1 *SimpleKMeans*

Teknik klustering dengan *k*-means merupakan teknik sederhana yang mengelompokan objek dengan meminimasi jarak *sum of squared* antara objek dan *centroid* yang saling berkaitan. *K*-means adalah teknik yang paling sederhana dan populer digunakan dalam penelitian yang berhubungan dengan klustering [3][10][11]. Dalam teknik ini terdapat *distance metric* yang umum digunakan yaitu *Euclidean* dan *Manhattan* [10][13]. Kelemahan teknik ini adalah cukup sensitif terhadap posisi awal pusat kluster. Untuk menangani masalah inisiasi posisi, maka dapat dilakukan dua pendekatan yaitu memilih nilai kluster *k* secara acak atau memilih sejumlah nilai inisiasi yang berbeda diluar dari titik objek [11].

1.2 *X-Means*

Kesulitan dalam penentuan jumlah kluster yang digunakan dan waktu proses yang lama untuk data yang besar dalam klustering mendorong penyempurnaan teknik *k*-means yaitu dengan XMeans [14]. Pengelompokan dilakukan berdasarkan *distance function* yang lebih spesifik, antara lain Euclidean Distance, Manhattan Distance, dan ChebyShev Distance. Teknik ini bekerja dengan cara mencari ruang diantara tempat kluster dan jumlah kluster untuk melakukan optimasi Bayesian Information Criterion (BIC) dan memberikan keputusan apakah centroid harus dibagi atau tidak. Hanya dapat digunakan untuk data Numeric.

1.3 *Farthest First*

Selain *XMeans*, muncul variasi lain dari *K*-Means yaitu *Farthest First y*ang secara acak memilih satu titik sebagai pusat pertama berdasarkan parameter masukan yang tersedia, kemudian menghitung pusat selanjutnya secara rekursif dengan melihat titik berdasarkan jarak maksimal dan minimal dari centroid sebelumnya hingga semua titik konvergen. Teknik ini lebih menangani masalah terkait waktu proses, algoritma ini belum sempurna namun hampir optimal [1].

C. Dataset

Untuk melakukan klustering dengan *raw data*, maka diperlukan dataset untuk diuji coba. Dalam paper ini, data diambil dari repositori *UCI Machine Learning* dengan dataset *bag-of-words* [15] yang terdiri dari 5 koleksi teks tidak terstruktur, namun dalam penelitian ini hanya digunakan 3 koleksi teks berukuran kecil yaitu Enron Email Dataset [16], Daily Kos Blog entries [17], dan NIPS Proceedings [18]. Semua dataset tidak memiliki label dan akan dilakukan klustering menggunakan algoritma yang tersedia di Weka. Koleksi teks terdiri dari atribut nama dokumen *docID*, kata yang dihitung dalam distribusi frekuensi *wordID*, dan *count* yang merupakan banyaknya kata dengan *wordID* tertentu yang tampil dalam dokumen *docID* .

D. Langkah Penelitian

Dalam uji coba yang dilakukan, jumlah dokumen *D* yang akan diproses antara lain Enron = 15.934 buah; Daily Kos = 3.430 buah; dan NIPS = 1.500 buah dengan jumlah *instance* masing-masing sebanyak 1.048.573 *instance*, 353.161 *instance*, dan 746.319 *instance*.

Semua dataset harus dirubah formatnya terlebih dahulu dari *.txt ke *.csv, kemudia di simpan ulang dalam bentuk *.arff file melalui menu *Tools* pada Weka. Untuk mendapatkan perbandingan hasil klustering, digunakan 3 metode klustering, yaitu *SimpleKMeans, X-Means, dan Farthest First*. Tiap algoritma akan diujikan dengan merubah beberapa parameter dari input data, meliputi jumlah kluster dan *seed* (*initial centroid*). Untuk mendapatkan hasil evaluasi class yang terbentuk maka semua nilai numeric pada dataset harus diubah dalam bentuk nominal menggunakan filter *unsupervised NumericToNominal* sebelum data diolah.

Jumlah kluster *k* sebenarnya untuk *raw data* bersifat *unknown* sehingga diperlukan uji coba dengan nilai *k* yang berbeda untuk mendapat kandidat nilai *k* terbaik yang menghasilkan nilai evaluasi *Sum of Square Error* (SSE) kluster *instance* paling minimal. SSE merupakan cara validasi kluster dimana error tiap titik objek adalah jarak ke kluster terdekat.

III. HASIL PENGUJIAN DAN ANALISIS

Pengujian dibagi dalam 3 skenario untuk melihat performa dataset yang diolah dan algoritma yang digunakan. Pengujian pertama yaitu membandingkan persentase kluster yang terbentuk dari algoritma *SimpleKMeans* (SK), *Xmeans* (XM) dan *Farthest First* (FF) untuk 3 kategori teks menggunakan parameter awal kluster *k*=2 dan nilai *seed s*=10, sedangkan *distance function* yang digunakan yaitu *Euclidean*.

Tabel II.

Persentase Pembentukan Kluster *Bag-of-Words*

Klusterer	Enron		KOS		NIPS	
	Kluster 0	Kluster 1	Kluster 0	Kluster 1	Kluster 0	Kluster 1
Simple-KMeans	44%	56%	51%	49%	50%	50%
Xmeans	--	--	48%	52%	50%	50%
Farthest First	100%	0%	100%	0%	97%	3%

Pada hasil pengujian Tabel II dapat dilihat bahwa *SimpleKMeans* dan *XMeans* hampir memberikan jumlah rata-rata hasil klustering yang sama yaitu sekitar 50% untuk tiap kategori data. Pada prinsipnya, *XMeans* bekerja tanpa melihat *k* yang digunakan. Implementasi dengan dataset yang besar seperti Enron tidak dapat memberikan keluaran hasil klustering yang dihasilkan, sedangkan *Farthest First* mengelompokan data pada kluster 0 sebanyak 100% untuk Enron dan KOS. Hal ini dikarenakan FF hanya memilih satu titik sebagai pusat utama, sehingga hanya titik yang belum konvergen dimasukan dalam kluster 1.

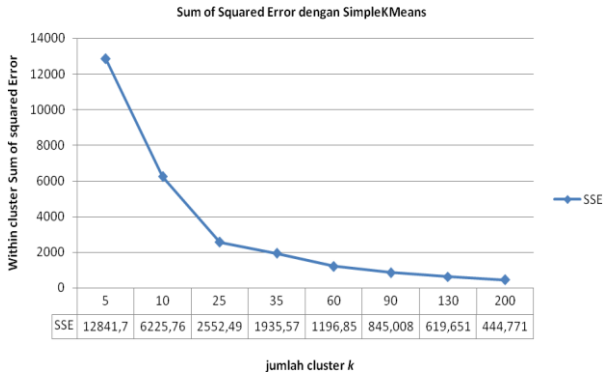
Pengujian kedua yaitu menggunakan fitur *classes to cluster evaluation* yang bertujuan untuk melihat seberapa baik dataset yang diuji dalam melakukan klustering tanpa diberikan *class* antar data. Untuk menggunakan fitur ini, semua dataset dalam bentuk Numeric harus diubah dalam bentuk nominal agar dapat diproses.

Tabel III.Hasil pengujian *Classes to clusterer evaluation*

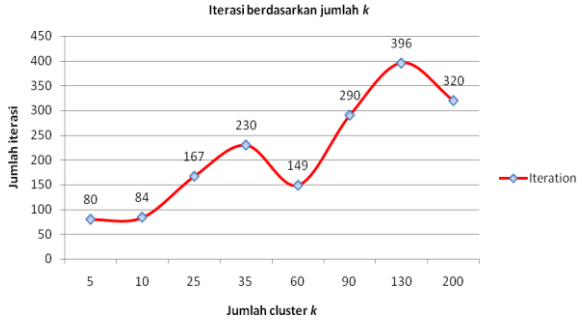
Klusterer	**Enron**		
	Jumlah Instance	Incorrectly clustered instances	Correctly clustered instances
Simple-KMeans	1.048.573	242.139	806.434
		23,0922%	76,9078%
Farthest First		242.323	806.250
		23,1098%	76,8902%
	KOS		
	Jumlah Instance	Incorrectly clustered instances	Correctly clustered instances
SimpleKMeans	353.161	62.181	290.980
		17,6073%	82,3927%
Farthest First		61.879	291.282
		17,5215%	82,4785%
	NIPS		
	Jumlah Instance	Incorrectly clustered instances	Correctly clustered instances
SimpleKMeans	746.319	298.309	448.010
		39,9709%	60,0291%
Farthest First		298.290	448.029
		39,9683%	60,0317%

Hasil pengujian pada Tabel III merupakan evaluasi dari klustering dataset pada tabel II sebelumnya dengan mengabaikan atribut *count* selama proses pengujian. Berdasarkan hasil pengujian, dapat dilihat bahwa kesalahan kluster yang menggunakan klusterer *Farthest First* lebih kecil dari *SimpleKMeans* walaupun nilai selisih tidak terlalu signifikan. Hal ini dikarenakan adanya irisan antar kluster yang tidak bisa diprediksi sehingga pengelompokan menjadi tidak tepat.

Dalam penelitian [12] melalui proses heuristik, mengusulkan kandidat *k* terbaik yaitu 10,25,35,60, dan 90. Karena data yang digunakan besar, maka jumlah *k* ditambahkan nilai 130 dan 200. Pengujian ketiga menggunakan nilai SSE dibandingkan dengan parameter *k* (Gambar 5) dan iterasi yang terjadi (Gambar 6). Dataset yang digunakan hanya kategori KOS tanpa memperhatikan waktu prosesnya.



Gambar 5. SSE untuk nilai *k* berbeda



Gambar 6. Iterasi Dataset KOS untuk nilai *k* berbeda

Berdasarkan Gambar 5, jumlah kluster yang digunakan berpengaruh terhadap nilai SSE. Semakin besar *k* yang digunakan maka SSE yang dihasilkan semakin kecil. Sebaliknya pada Gambar 6, jumlah iterasi tidak bergantung pada *k* yang digunakan. Iterasi dilakukan untuk melakukan perhitungan ulang hingga mendapatkan kondisi paling ideal dari sebuah dataset. Iterasi paling baik yaitu saat nilai *k*=130 sebanyak 396 kali.

IV. KESIMPULAN/RINGKASAN

Weka merupakan perangkat data mining yang dapat digunakan untuk melakukan tugas klustering dari dataset yang tersedia karena menyediakan banyak algoritma klusterer yang mendukung untuk pengujian. Dari hasil penelitian, 3 klusterer yaitu *SimpleKMeans, Xmeans dan Farthest First* dapat digunakan untuk melakukan proses pengelompokan dataset tanpa label dengan representasi *bag-of-words* dengan tingkat keberhasilan pembentukan kluster untuk *SimpleKMeans* dan *Xmeans* sebesar 50%. Dataset dapat diolah dengan baik menggunakan Weka tanpa diperlukan *pre-processing* karena data tidak mengandung *noise* ataupun *outlier*.

Kelemahan klustering terletak pada penggunaan jumlah kluster *k* untuk melakukan inisiasi *centroid*. Diperlukan lebih banyak informasi atribut dari dataset sehingga tingkat kesalahan dalam pengelompokan *instance* lebih kecil. Jumlah *k* mempengaruhi nilai *sum of squared error*, semakin besar *k* maka semakin kecil nilai error-nya. Sedangkan jumlah iterasi yang terjadi tidak dipengaruhi jumlah *k* yang digunakan. Namun untuk mendapatkan hasil iterasi kluster maksimal, pengelompokan dataset tanpa label dengan ukuran besar dapat menggunakan nilai *k=130*.

DAFTAR PUSTAKA

[1] N. Sharma, A. Bajpai, and R. Litoriya, "Comparison the various klustering algorithms of weka tools," Int. J. Emerg. Technol. Adv. Eng., vol. 2, no. 5, May 2012.

[2] Gina, "Klustering," LINTAS, 2012. [Online]. Available: https://ginageh.wordpress.com/2008/10/28/klustering/. [Diakses: 05-Dec-2014].

[3] Sapna Jain, M. A. Alam, and M. N. Doja, "K-Means Klustering Using Weka Interface," presented at the 4th National Conference INDIA Computing For Nation Development, Bharati Vidyapeeth's Institute of Computer Applications and Management, New Delhi, 2010.

[4] "Bag of words model." [Online]. Available: http://en.wikipedia.org/wiki/Bag-of-words_model. [Diakses: 05-Dec-2014].

[5] N. Grattan, "Bag of Words and Frequency Distributions in C#," Nick Grattan's Blog, 09-Jun-2014. [Online]. Available: http://nickgrattan.wordpress.com/2014/06/09/bag-of-words-and-frequency-distributions-in-c/. [Diakses: 05-Dec-2014].

[6] "Weka 3: Data Mining Software in Java." [Online]. Available: http://www.cs.waikato.ac.nz/~ml/weka/. [Diakses: 02-Dec-2014].

[7] F. Eibe, "Machine Learning with WEKA," Department of Computer Science, University of Waikato, New Zealand, 22-Feb-2011.

[8] S. Jiménez, "Text Classification and Klustering with WEKA: WEKA A guided example," Univercidad Nacional De Colombia.

[9] N. Sharma and S. Niranjan, "Optimization Of Word Sense Disambiguation Using Klustering In Weka," IntJComputer Technol. Appl., vol. 3, no. 4, pp. 1598–1604, Aug. 2012.

[10] D. Sinwar and R. Khausik, "Study of Euclidean and Manhattan Distance Metrics using Simple K-Means Klustering," Int. J. Res. Appl. Sci. Eng. Technol., vol. 2, no. V, May 2014.

[11] R. Sharma, M. A. Alam, and A. Rani, "K-Means Klustering in Spatial Data Mining using Weka Interface," presented at the International Conference on Advances in Communication and Computing Technologies (ICACACT), 2012.

[12] S. Surisetty and S. K. Dhamodaran, "Klustering Similar Animation Programs using Unsupervised learning." [Online]. Available: http://sharathkumardhamodaran.weebly.com/uploads/5/5/6/7/5567064/ml_-_project.pdf. [Diakses: 02-Dec-2014].

[13] A. Singla and Karambir, "Comparative Analysis & Evaluation of Euclidean Distance Function and Manhattan Distance Function Using K-means Algorithm," Int. J. Adv. Res. Comput. Sci. Softw. Eng., vol. 2, no. 7, Jul. 2012.

[14] D. Pelleg and A. Moore, "X-Means: Extending k-means with Efficient Estimation of the Number of Klusters," presented at the Seventeenth International Confrence on Machine Learning, 2000, pp. 727–734.

[15] "Bag of Words Data Set." [Online]. Available: https://archive.ics.uci.edu/ml/machine-learning-databases/bag-of-words/. [Diakses: 01-Dec-2014].

[16] "Enron Email Dataset." [Online]. Available: http://www.cs.cmu.edu/~enron/. [Diakses: 01-Dec-2014].

[17] "Daily Kos." [Online]. Available: http://www.dailykos.com/#. [Diakses: 01-Dec-2014].

[18] "NIPS Proceedingsβ." [Online]. Available: http://papers.nips.cc/. [Diakses: 01-Dec-2014].