

Text Mining Classification Sebagai Rekomendasi Dosen Pembimbing Tugas Akhir Program Studi Sistem Informasi

Zuliar Efendi¹, Mustakim²

^{1,2}Laboratorium Data Mining Program Studi Sistem Informasi, Fakultas Sains dan Teknologi
Universitas Islam Negeri Sultan Syarif Kasim Riau
Jl. HR. Soebrantas No. 155 Simpang Baru, Tampan, Pekanbaru, Riau – Indonesia 28293
e-mail: ¹zuliar.efendi@students.uin-suska.ac.id, ²mustakim@uin-suska.ac.id

Abstrak

Penelitian merupakan karya ilmiah dari kegiatan mahasiswa berupa penelitian yang membahas suatu masalah sesuai dengan ilmu pada program studi yang ditempuh mahasiswa dengan menggunakan aturan yang berlaku serta bimbingan oleh dosen pembimbing. Dengan memanfaatkan data proposal tugas akhir, bisa ditemukan suatu pola menarik dengan menggunakan text mining. Didalam text mining terdapat klasifikasi yang salah satu algoritmanya yaitu K-Nearest Neighbour (KNN). KNN digunakan karena terdapat nilai k yang bisa menjadi alternatif pilihan. Sebelum menggunakan KNN, dilakukan pembobotan dengan menggunakan Term Frequency dan Inverse Document Frequency (TF-IDF). Setelah pembobotan, dilakukan perhitungan similaritas antar dokumen menggunakan Cosine Similarity. Kemudian digunakan nilai k pada KNN dari data yang telah diurutkan pada Cosine Similarity. Nilai k yang digunakan yaitu 15, karena kelas yang dihasilkan terdapat hasil tunggal atau satu yang dominan. Akurasi yang didapat dalam klasifikasi dengan k = 15 yaitu sebesar 60%.

Kata kunci: Klasifikasi, Cosine Similarity, KNN, Text Mining, TF-IDF

Abstract

research is the scientific work of the student activities in the form of research that addresses an issue in accordance with the knowledge on the courses taken by students using the applicable rules and guidance by the supervisor. Using data final project proposal, can be found an interesting pattern by using text mining. In text mining are one of the classification algorithm is K-Nearest Neighbor (KNN). KNN is used because there are values of k can be an alternative option. Before using KNN, is weighted by using Term Frequency and Inverse Document Frequency (TF-IDF). Having to weight, calculate the similarity between documents using the Cosine Similarity. Then used the k on KNN of data that has been sorted on Cosine Similarity. K value used is 15, because the generated classes are single result or one that is dominant. Accuracy obtained in the classification with k = 15 is equal to 60%.

Keywords: Classification, Cosine Similarity, KNN, Text Mining, TF-IDF

1. Pendahuluan

Program sarjana adalah pendidikan akademik bagi lulusan menengah atau sederajat sehingga mampu mengamalkan Ilmu Pengetahuan dan Teknologi melalui penalaran ilmiah [2]. Untuk mendapatkan gelar sarjana salah satu syaratnya yaitu skripsi, skripsi adalah karya ilmiah dari hasil penelitian mahasiswa program sarjana (S1) [2].

Tugas Akhir merupakan karya ilmiah dari kegiatan mahasiswa berupa penelitian yang membahas suatu masalah sesuai dengan ilmu pada program studi yang ditempuh mahasiswa dengan menggunakan aturan yang berlaku serta bimbingan oleh dosen pembimbing [1]. Dosen pembimbing sebagai bagian dari proses dalam penyelesaian penulisan skripsi mempunyai peran yang sangat penting, karena memiliki tanggung jawab agar mahasiswa mampu menyusun skripsi dengan baik dan berkualitas [3].

Kualitas dari skripsi memiliki beberapa indikator yaitu mahasiswa mengalami ujian skripsi ulang/ tidak, waktu penyelesaian skripsi, nilai ujian dari skripsi, serta kesesuaian topik skripsi dengan keahlian/ kompetensi dosen pembimbing [3]. Sehingga mahasiswa perlu menemukan dosen pembimbing yang tepat sesuai dengan topik yang akan diangkat mahasiswa dalam penulisan tugas akhir.

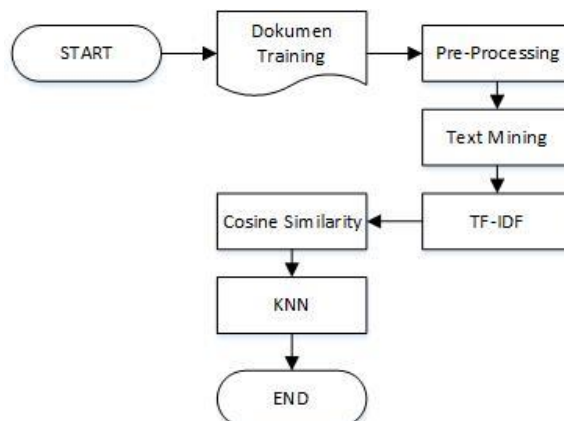
Program studi Sistem Informasi Universitas Islam Negeri Sultan Syarif Kasim Riau memiliki 2 konsentrasi matakuliah yaitu Manajemen Sistem Informasi dan Rekayasa Sistem Informasi. Setiap konsentrasi terdapat matakuliah pilihan yang diajarkan oleh dosen sesuai dengan keahlian/ kompetensinya, Begitu juga dalam penulisan Tugas Akhir. Sebelum menyusun Tugas Akhir, mahasiswa harus memasukkan proposal yang akan diseleksi apakah proposalnya diterima atau ditolak. Untuk itu, mahasiswa perlu untuk menemui dosen sesuai dengan topik yang diangkat supaya mahasiswa bisa melakukan bimbingan Pra-Proposal. Dengan memanfaatkan hasil dari penggalian *text*, dapat dilakukan otomatisasi dalam penentuan dosen pembimbing. Beberapa Penelitian-penelitian dalam penentuan dosen pembimbing tugas akhir sudah pernah dilakukan yaitu Sistem Rekomendasi Penentuan Dosen Pembimbing Tugas Akhir Dengan Menggunakan Algoritma Rabin-Karp [1]. Penelitian lain menggunakan *K-Nearest Neighbour* (KNN) yaitu Klasifikasi Teks dengan Menggunakan Algoritma *K-Nearest Neighbour Classifier* pada Dokumen Tugas Akhir [7]. Untuk menganalisa pola yang menarik dari sekumpulan data teks yang berjumlah besar digunakan *text mining* yang merupakan variasi dari *data mining* [4].

Klasifikasi adalah suatu proses kedalam kelas berdasarkan kesamaan sifat yang didalamnya terdapat *data training* (data latih) sebagai acuan klasifikasi serta *data testing* (data uji) untuk pengujian [5]. Tujuannya yaitu untuk memprediksi objek dari kelas yang belum mempunyai label atau kecenderungan data yang belum terklasifikasi [5]. *Nearest neighbour* adalah suatu pendekatan dalam mencari kasus dengan menghitung kedekatan berdasarkan pencocokan bobot dari fitur yang ada, pada algoritma ini jarak antar data dapat dihitung yang terdapat nilai K didalamnya. Nilai K pada *Nearest Neighbour* berarti k-data terdekat dari data uji [6].

Oleh karena itu dipilih algoritma K-NN untuk melakukan perekomendasi dosen pembimbing pada program studi sistem informasi Universitas Islam Negeri Sultan Syarif Kasim Riau. Penelitian yang dilakukan adalah dengan memanfaatkan data dari hasil seleksi proposal Tugas Akhir, yang kemudian bisa dimanfaatkan untuk menentukan dosen pembimbing.

2. Metode Penelitian

Metode pada penelitian ini bisa dilihat pada gambar dibawah ini :



Gambar 1. Proses Text Mining

Pada tahap pengolahan *text mining* dilakukan tahap *preprocessing* yang dijelaskan pada Gambar 2.



Gambar 2. Preprocessing Text Mining

2.1. Text Mining

Text mining adalah penggalian informasi dari teks oleh user menggunakan *tools* analisis. Secara umum *text mining* mengadopsi proses-proses didalam data *mining* dan didalam *text mining* juga menggunakan teknik data *mining* [5].

2.2. Text Preprocessing

Text preprocessing menjadi tahap awal dalam *text mining*. *Preprocessing* dilakukan untuk menghilangkan bagian atau teks yang tidak diperlukan sehingga mendapatkan data yang berkualitas untuk dieksekusi [5]. Pertama dalam tahap *preprocessing* yaitu *tokenizing* yang bertujuan untuk memecah kalimat menjadi perkata yang terpisah dikenal dengan nama *term* atau token [6]. Selanjutnya *filtering* dengan melakukan penghapusan tanda baca, merubah huruf capital menjadi huruf kecil dan penghapusan *stopword* yang bertujuan untuk menghapus kata-kata yang tidak bermanfaat atau tidak memiliki pengaruh dalam proses [6]. Terakhir yaitu *stemming* untuk mendapatkan kata dasar dari kata yang telah mendapatkan imbuhan atau keterangan lainnya. *Stemming* yang digunakan yaitu *stemming* Nazief dan Adriani karena Algoritma ini memiliki akurasi lebih besar dibandingkan dengan algoritma porter [11].

2.3. Pembobotan (Term Weighting)

Term Frequency dan *Inverse Document Frequency* (TF-IDF) merupakan pembobotan yang sering digunakan dalam penelusuran informasi dan *text mining* [12]. Persamaan untuk menghitung TF adalah:

$$TF(d, t) = f(d, t) \dots \dots \dots (1)$$

Ket:

f(d,t) : kemunculan kata t dalam dokumen d

Persamaan untuk menghitung nilai IDF dapat dilihat sebagai berikut:

$$IDF(t) = \log(N/df(t)) \dots \dots \dots (2)$$

Persamaan untuk menghitung TF-IDF yaitu:

$$TFIDF = TF(d, t). IDF(t) \dots \dots \dots (3)$$

2.4. Cosine Similarity

Cosine similarity merupakan perhitungan umum yang digunakan dalam menghitung similaritas antar dokumen [12].

2.5. K-Nearest Neighbour

Algoritma *K-Nearest Neighbour* (KNN) adalah metode untuk klasifikasi pada objek berdasarkan data latih yang jaraknya paling dekat dari objek tersebut [7]. Dekat atau jauhnya tetangga biasanya dihitung berdasarkan jarak Euclidean dengan rumus umum:

$$d = \sqrt{\sum_{i=1}^n (a_i - b_i)^2} \dots \dots \dots (4)$$

Keterangan:

- d = jarak
- a = data uji/testing
- b = sampel data
- i = variable data
- n = Dimensi data

2.6 k-Fold Cross Validation

k-Fold Cross Validation merupakan salah satu metode untuk mengetahui rata-rata keberhasilan dari suatu system [13]. Menghitung nilai akurasinya dapat dilakukan dengan persamaan:

$$Akurasi = \frac{\text{Jumlah Klasifikasi benar}}{\text{Jumlah data uji}} \times 100\% \dots \dots \dots (5)$$

3. Analisa dan Hasil

Pada tahap ini dilakukan analisa data proposal pada tahap 1 dan tahap 2 semester ganjil tahun 2016/ 2017 pada program studi sistem informasi yang sudah disebarakan atau diinformasikan. Pengolahan ini menggunakan sebanyak 107 data dengan pembagian 98 data *training* dan 10 data *testing* (data uji) dengan algoritma *K-Nearest Neighbour*. *Preprocessing* pada tahap *Tokenizing* dan *Filtering* menggunakan *tools* weka, serta menggunakan Algoritma Nazief dan Adriani untuk *stemmer*. Setelah di *stemmer* dilakukan perhitungan, berikut adalah penjabaran hasil dan analisisnya:

3.1. Persiapan Data

Pencarian pola menarik dari judul atau topik tugas akhir yang diangkat dengan menggunakan *text mining* untuk mendapatkan *output* atau keluaran yaitu nama dosen pembimbing yang bisa menjadi rekomendasi pembimbing pada Tugas Akhir. Berikut adalah data awal yang digunakan:

Tabel 1. Data *Training*

Dokumen	Judul	Dosen Pembimbing
D1	Rancang Bangun Sistem Informasi Bimbingan dan Konseling	AAN
D2	Rancang Bangun Sistem Otomasi Perpustakaan Berbasis SLIMS	AAN
D3	Rancang Bangun Sistem Informasi Manajemen Administrasi Star Gym Berbasis Sms Gateway	AAN
...
D97	Aplikasi <i>Mobile Tour System</i> Berbasis Android	ZNL

Tabel 2. Data *Testing*

Data Testing	Judul	Dosen Pembimbing
Test 1	Implementasi Algoritma Modified k-Nearest Neighbors Pada Klasifikasi Penerima Program Keluarga Harapan (PKH)	MTK
Test 2	Sistem Pakar Diagnosa Penyakit Pada Ayam Dengan Metode <i>Certainty Factor</i> Berbasis <i>Web</i>	TKA
Test 3	Penerapan <i>Multi Attribute Utility Theory</i> Untuk Pengambilan Keputusan Pengunjung Perpustakaan Berbasis <i>Cluster</i>	MTK
...
Test 10	Rancang Bangun Sistem Informasi Pemetaan Geografis Berbasis Android	MJM

3.2. Preprocessing

Preprocessing dilakukan dengan menggunakan *tools weka* yaitu pada tahap *Tokenizing* serta Tahap *Filtering*. *Tokenizing* dilakukan untuk memecah kalimat menjadi sebuah kata, dan *Filtering* untuk menghapus tanda baca, merubah menjadi huruf kecil serta menghilangkan *stopword*. *Stopword* yang digunakan adalah *stoplist* tala [10].

Setelah melakukan *Tokenizing* dan *Filtering*, kemudian dilakukan *stemming*. Pada tahap ini dilakukan menggunakan algoritma Nazief dan Adriani. Berikut adalah hasil dari *preprocessing*:

Tabel 3. *Preprocessing*

Term	TF						
	D1	D2	D3	D4	D5	...	D107
	AAN	AAN	AAN	AAN	ANI	...	MJM
administrasi	0	0	1	0	0	...	0
akademik	0	0	0	1	0	...	0
bangun	1	1	1	0	0	...	1
basis	0	1	1	0	0	...	1
bimbing	1	0	0	0	0	...	0
...
wisata	0	0	0	0	0	...	0

3.3. Pembobotan (Term Weighting)

Pembobotan dilakukan dengan menggunakan *Term Frequency* dan *Inverse Document Frequency* (TF-IDF). Berikut hasilnya:

Tabel 4. TF-IDF

Term	TF							IDF
	D1	D2	D3	D4	D5	...	D107	DF

administrasi	0	0	1	0	0	...	0	3	1.5522
akademik	0	0	0	1	0	...	0	4	1.4273
bangun	1	1	1	0	0	...	1	27	0.5980
basis	0	1	1	0	0	...	1	33	0.5108
bimbing	1	0	0	0	0	...	0	1	2.0293
...
wisata	0	0	0	0	0	...	0	1	2.0293

Setelah didapatkan nilai TF, DF dan IDF. Selanjutnya dilakukan perhitungan bobot dalam setiap dokumen. Hasil perhitungannya sebagai berikut:

Tabel 5. Bobot

Term	Wdt						
	D1	D2	D3	D4	D5	...	D107
Administrasi	0.0000	0.0000	1.5522	0.0000	0.0000	...	0.0000
Akademik	0.0000	0.0000	0.0000	1.4273	0.0000	...	0.0000
Bangun	0.5980	0.5980	0.5980	0.0000	0.0000	...	0.5980
Basis	0.0000	0.5108	0.5108	0.0000	0.0000	...	0.5108
Bombing	2.0293	0.0000	0.0000	0.0000	0.0000	...	0.0000
...
Wisata	0.0000	0.0000	0.0000	0.0000	0.0000	...	0.0000

3.4. Cosine Similarity

Setelah melakukan proses pembobotan, selanjutnya data akan melalui tahap klasifikasi. Menghitung similaritas antara dokumen dengan menggunakan *Cosine Similarity* dengan menghitung kemiripan vektor antara data *testing* dengan data *training* yang sudah terklasifikasi. Berikut adalah salah satu perhitungan pada dokumen 100 (data *testing* 3) dengan data yang sudah terklasifikasi :

Tabel 6. Perkalian bobot D100

Term	WD100						
	D100*D1	D100*D2	D100*D3	D100*D4	D100*D5	...	D100*D107
administrasi	0.0000	0.0000	0.0000	0.0000	0.0000	...	0.0000
akademik	0.0000	0.0000	0.0000	0.0000	0.0000	...	0.0000
Bangun	0.0000	0.0000	0.0000	0.0000	0.0000	...	0.0000
Basis	0.0000	0.2609	0.2609	0.0000	0.0000	...	0.2609
Bombing	0.0000	0.0000	0.0000	0.0000	0.0000	...	0.0000

Tabel 7. Perkalian bobot D100 (Lanjutan)

...
Wisata	0.0000	0.0000	0.0000	0.0000	0.0000	...	0.0000
Total	0.0000	2.6705	0.2609	0.0000	0.0000	...	0.2609

Berikut adalah total dari *cosine similarity* dari data *testing* yang telah dilakukan perhitungan:

Tabel 8. Penyelesaian *Cosine Similarity*

Dokumen	D1	D2	D3	D4	D5	Dn
Test 1	0.0000	0.0000	0.0000	0.0000	0.0000	...
Test 2	0.0631	0.3241	0.3241	0.0631	0.0000	...
Test 3	0.0000	2.6705	0.2609	0.0000	0.0000	...
...
Test 10	0.8934	1.0393	1.1544	0.1781	0.0000	...

Setelah dilakukan perkalian pembobotan antara data testing dengan data training yang telah terklasifikasi. Selanjutnya dilakukan perhitungan panjang vektor pada setiap dokumen termasuk data testing. Dari hasil perhitungan didapatkan panjang vektor dari setiap dokumen sebagai berikut:

Tabel 9. Panjang Vektor

Term	Panjang Vektor						
	D1	D2	D3	D4	D5	...	D107
administrasi	0.0000	0.0000	2.4095	0.0000	0.0000	...	0.0000
Akademik	0.0000	0.0000	0.0000	2.0372	0.0000	...	0.0000
Bangun	0.3576	0.3576	0.3576	0.0000	0.0000	...	0.3576
Basis	0.0000	0.2609	0.2609	0.0000	0.0000	...	0.2609
Bombing	4.1183	0.0000	0.0000	0.0000	0.0000	...	0.0000
...
Wisata	0.0000	0.0000	0.0000	0.0000	0.0000	...	0.0000
Total	9.1302	11.6856	22.4470	2.2154	17.0975	...	5.0986
P. Vektor	3.0216	3.4184	4.7378	1.4884	4.1349	...	2.2580

Setelah di dapatkan hasil pada tabel ini dan panjang vektor pada tabl ini, kemudian dilakukan perhitungan *cosine similarity*. Berikut adalah *cosine similarity* pada dokumen 100 (data *testing* 3):

Tabel 10. Nilai *Cosine Similarity*

D1	D2	D3	D4	D5	...	D107
0.0000	0.1518	0.0107	0.0000	0.0000	...	0.0224

Setelah didapatkan nilai *cosine similarity*, selanjutnya urutkan tingkat kemiripan data. Diperoleh:

Tabel 11. Urutan Tingkat Kemiripan

Dokumen	D48	D59	D2	D101	D104	...	D106
Cosine Similarity	0.3240	0.1754	0.1518	0.1167	0.1133	...	0.0000

3.5. K-Nearest Neighbour

Setelah dilakukan perhitungan *cosine similarity*, selanjutnya menentukan *class* dari data testing, pada hal ini nilai k yang digunakan yaitu k = 15. Berikut adalah hasil klasifikasi pada Dokumen 100 (data *testing* 3) dengan KNN:

Tabel 12. Klasifikasi D100

Dokumen	D48	D59	D2	D101	D104	...	D107
Class	MTK	MTK	AAN	MTK	MTK	...	MJM
Cosine Similarity	0.3240	0.1754	0.1518	0.1167	0.1133	...	0.0224

Hasil klasifikasi pada dokumen 100 (data *testing* 3) berada pada class MTK. Yang artinya dosen pembimbing tugas akhir pada data testing 3 yaitu MTK. Berikut adalah hasil klasifikasi 10 data *testing* dengan nilai k= 15, jika menggunakan nilai k= 10 terdapat beberapa data *testing* yang tidak dapat kelas tunggal/ dominan. berikut hasil k= 15 menggunakan KNN, diperoleh:

Tabel 13. Hasil Klasifikasi

Dokumen	Data Testing	Class
D98	Test 1	MTK
D99	Test 2	EKI
D100	Test 3	MTK
D101	Test 4	MTK
D102	Test 5	TKA
D103	Test 6	MTK
D104	Test 7	MTK
D105	Test 8	STM
D106	Test 9	MJM
D107	Test 10	MJM

3.6. Akurasi

Dari perhitungan sebelumnya didapatkan kelas pada setiap data *testing*. Selanjutnya perlu dilakukan pengujian akurasi dari hasil KNN ini. Berikut adalah tabel hasilnya:

Tabel 14. Hasil Akurasi

Dokumen	Data Testing	Hasil
D98	Test 1	Sesuai
D99	Test 2	Tidak Sesuai
D100	Test 3	Sesuai
D101	Test 4	Sesuai
D102	Test 5	Tidak Sesuai
D103	Test 6	Sesuai
D104	Test 7	Sesuai
D105	Test 8	Tidak Sesuai
D106	Test 9	Tidak Sesuai
D107	Test 10	Sesuai

Selanjutnya, dihitung akurasi dari metode *K-Nearest Neighbor* dengan k= 15 dan diperoleh:

$$Akurasi = \frac{6}{10} = 0.6 * 100\% = 60\%$$

4. Kesimpulan

Berdasarkan hasil penelitian dan pembahasan yang telah dilakukan, diperoleh bahwa menggunakan *Text Mining* dengan algoritma klasifikasi yaitu *K-Nearest Neighbor* (KNN) untuk rekomendasi dosen pembimbing tugas akhir berhasil dilakukan dengan akurasi sebesar 60% dengan menggunakan 98 data training dan 10 data *testing*. Jumlah data *training* mempengaruhi akurasi dari klasifikasi terutama pada penelitian ini menggunakan algoritma *K-Nearest Neighbor*.

Beberapa saran untuk penelitian berikutnya yaitu bisa menggunakan *datasets* dengan jumlah lebih banyak, menambahkan abstrak dari tugas akhir untuk dilihat polanya untuk meningkatkan akurasi dari algoritma, serta menggunakan algoritma yang berbeda untuk dilakukan perbandingan.

Ucapan Terima Kasih

Terima kasih yang sebesar-besarnya kepada Fakultas Sains dan Teknologi Universitas Islam Negeri Sultan Syarif Kasim Riau atas bantuan dana penelitian yang diberikan serta fasilitas dan dukungan dari pimpinan. Demikian juga ucapan terima kasih kepada Tim Puzzle Research Data Technology (Predatech) Fakultas Sains dan Teknologi UIN Sultan Syarif Kasim Riau atas segala masukan, koreksi dan bantuan pelaksanaan kegiatan sehingga penelitian ini dapat diselesaikan dengan baik.

Daftar Pustaka

- [1] Salam dkk. *Sistem Rekomendasi Penentuan Dosen Pembimbing Tugas Akhir Dengan Menggunakan Algoritma Rabin-Karp*. Techno.COM. 2015; Vol. 14, No. 3, Agustus.
- [2] Sugiyono. *Cara Mudah Menyusun Skripsi, Tesis dan Disertasi*. Yogyakarta : ALFABETA. 2013.
- [3] Hariyati, Rini Maryuni. 2012. *Survey Kinerja Dosen Pembimbing Skripsi Dan Kualitas Skripsi Mahasiswa Akuntansi STIE Malangkecewara*. JDA, Vol. 4, No. 2, September.
- [4] Kurniawan dkk. *Klasifikasi Konten Berita Dengan Metode Text Mining*. Jurnal Dunia Teknologi Informasi, 2012; Vol. 1, No.1.
- [5] Sanjaya dan Absar. *Pengelompokan Dokumen Menggunakan Winnowing Fingerprint dengan Metode K-Nearest Neighbour*. Jurnal CoreIT. 2015; Vol. 1, No. 2, Desember.
- [6] Winarko dan Jumadi. *Penggunaan KNN (K-Nearest Neighbor) untuk Klasifikasi Teks Berita yang Tak-Terkelompokkan pada saat Pengklasteran Oleh STC (Suffix Tree Clustering)*. 2015; Vol. 9, No. 1.
- [7] Larasati, Rosiana. *Klasifikasi Teks dengan Menggunakan Algoritma K-Nearest Neighbour Classifier Pada Dokumen Tugas Akhir*. Tugas Akhir: Universitas Widyatama.
- [8] Herdiawan. *Analisis Sentimen Terhadap Telkom Indihome Berdasarkan Opini Publik Menggunakan Metode Improved K-Nearest Neighbor*. Komputa
- [9] Jayanti dan Noeryanti. *Aplikasi Metode K-Nearest Neighbor dan Analisis Diskriminasi Untuk Analisis Resiko Kredit pada Koperasi Simpan Pinjam di Kopinkra Sumber Rejeki*. SNAST. 2014; 15 November.
- [10] Tala, Fadilla Z. *A Study of Stemming Effect on Information Retrieval in Bahasa Indonesia*. Master's Thesis, University of Amsterdam. 2003.
- [11] Augusta, Ledy. *Perbandingan Algoritma Stemming Porter dengan Algoritma Nazief & Adriani untuk Stemming Dokumen Teks Bahasa Indonesia*. KNS. 2009; November.
- [12] Purwanti, Endah. *Klasifikasi Dokumen Temu Kembali Informasi dengan K-Nearest Neighbour Information Retrieval Document Classified with K-Nearest Neighbor*. Record and Library Journal. 2015; Vol. 1, No. 2, Juli-Desember.
- [13] Banjarsari, dkk. *Penerapan K-Optimal Pada Algoritma Knn untuk Prediksi Kelulusan Tepat Waktu Mahasiswa Program Studi Ilmu Komputer Fmipa Unlam Berdasarkan IP Sampai Dengan Semester 4*. KLIK. 2015; Vol. 2, No. 02, September.