Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer e-ISSN: 2548-964X
Vol. 1, No. 5, Mei 2017, hlm. 405-414 http://j-ptiik.ub.ac.id

Klasifikasi *Tweets* Pada Twitter Dengan Menggunakan Metode *Fuzzy K-Nearest Neighbour* (*Fuzzy K-NN*) dan *Query Expansion* Berbasis Apriori

Joda Pahlawan Romadhona Tanjung[1], Mochammad Ali Fauzi[2], Indriati[3]

Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Brawijaya
Email: [1]jodapahlawan@gmail.com, [2]moch.ali.fauzi@gmail.com, [3]indriati.tif@ub.ac.id

Abstrak

Twitter adalah alat percakapan unik yang memungkinkan kita untuk mengirim dan menerima pesan singkat yang disebut *tweet* dalam komunitas Twitter. *Tweets* adalah pesan singkat yang memiliki panjang yang terdiri dari 140 karakter. *Tweets* yang muncul di beranda Twitter semuanya bercampur aduk menjadi satu mulai dari kategori ekonomi, olahraga, teknologi, otomotif, kesehatan dan lain sebagainya. Ketika seorang pengguna mencari sebuah berita atau informasi yang diinginkan, permasalahan yang muncul adalah pengguna menjadi kesulitan untuk memilahnya. Proses klasifikasi dapat dilakukan untuk mengkategorikan sebuah *tweets* dengan menggunakan algoritme *Fuzzy K-Nearest Neighbour*. Namun, proses pengklasifikasian sebuah *tweets* sukar dilakukan karena tweets berupa *short-text*. Oleh karena itu, sebelum dilakukan proses klasifikasi sebuah tweets dilakukan *preprocessing* dan proses ekspansi kata terlebih dahulu dengan algoritme *Query Expansion* agar memberikan hasil maksimal pada proses klasifikasi. Pada penelitian yang dilakukan menghasilkan akurasi terbaik sebesar 82%. Akurasi terbaik didapatkan saat menggunakan metode *Fuzzy K-NN* dengan *Query Expansion* tanpa *preprocessing* serta *threshold* untuk nilai *support* >= 0.15 dan nilai *confidence* >= 1.

Kata kunci: *Twitter, Tweets, Klasifikasi, Fuzzy K-Nearest Neighbour, Query Expansion, Preprocessing*

Abstract

Twitter is a unique conversation tool that allows us to send and receive short messages called tweets in the Twitter community. Tweets are short messages that have a length of 140 characters. Tweets that appear on the homepage are all jumbled into one, posted variety ranging from the economy, sports, technology, automotive, healthcare and others. When users search for a news or information desired, the problem that arises is Twitter user difficult to find tweets. The classification process can be performed to categorize a tweets using an algorithm Fuzzy K-Nearest Neighbour. However, the process of classifying a tweets it is difficult to do because the tweets in the form of short-text. Therefore, before doing the classification process a tweets done preprocessing and word expansion beforehand with Query Expansion algorithms in order to provide maximum results in the classification. In the study conducted to produce the best accuracy by 82%. Best accuracy is obtained when using the Fuzzy KNN method with Query Expansion without preprocessing and threshold for the support value> = 0.15 and the value of confidence> = 1.

Keywords: *Twitter, Tweets, Classification, Fuzzy K-Nearest Neighbour, Query Expansion, Preprocessing*

1. PENDAHULUAN

Twitter adalah alat percakapan unik yang memungkinkan kita untuk mengirim dan menerima pesan singkat yang disebut *tweet* dalam komunitas Twitter. *Tweets* adalah pesan singkat, memiliki panjang yang terdiri dari 140 karakter (Crow Communications, 2011). *Tweets* yang muncul di beranda Twitter semuanya bercampur aduk menjadi satu mulai dari kategori ekonomi, olahraga, teknologi, otomotif, kesehatan dan lain sebagainya. (Phuvipadawat and Murata, 2010). Ketika seorang pengguna mencari sebuah berita atau informasi yang diinginkan, permasalahan yang muncul adalah pengguna menjadi kesulitan untuk memilahnya. Oleh karena itu, untuk pengkategorian *tweets* diperlukan proses klasifikasi.

Klasifikasi merupakan suatu teknik pada *text mining* yang mengelompokkan suatu konten berdasarkan kemiripan teksnya. Dengan klasifikasi ini memungkinkan suatu *tweets* yang ada di Twitter dikelompokkan menjadi satu berdasarkan kategorinya. Misal, konten sepak bola, basket dan catur dikelompokkan ke dalam kategoti olahraga (Sriram et al., 2010).

Salah satu metode klasifikasi yang dapat mengelompokkan berita atau informasi pada Twitter adalah metode *Fuzzy K-Nearest Neighbour* (*Fuzzy K-NN*). *Fuzzy K-Nearest Neighbour* serupa dengan *crisp K-NN* dalam hal pencarian label set untuk *K-Nearest Neighbour*. Algoritme *Fuzzy K-Nearest Neighbour* lebih menempatkan class membership ke dalam suatu pola (*pattern*) daripada menempatkan pola (*pattern*) ke dalam kelas khusus (Keller, Gray and Givens, 1985). Kelebihan dari penelitian sebelumnya yang dilakukan dengan menggunakan metode *Fuzzy K-NN* ini adalah tingkat keakurasian yang dihasilkan lebih baik jika dibandingkan dengan metode *K-Nearest Neighbour* (*K-NN*) dan *Support Vector Machine* (*SVM*). Dari hasil penelitian yang dilakukan didapatkan nilai persentase akhir *Fuzzy K-Nearest Neighbour* (*Fuzzy K-NN*) 64,12%, *K-Nearest Neighbour* (*K-NN*) 58,23% dan *Support Vector Machine* (*SVM*) 58,45% (Zhang, Niu and Nie, 2009).

Twitter sudah menerapkan metode klasifikasi *tweets* pada situsnya yang memudahkan pengguna Twitter mendapatkan berita atau informasi berdasarkan kategorinya. Namun, ada kelemahan pada penerapan yang dilakukan oleh Twitter yaitu *tweets* yang diklasifikasi hanya berdasarkan profil akun. Akun yang diambil hanya akun yang resmi saja. Sehingga muncul dua permasalahan dari kasus tersebut, yaitu pertama *tweets* pengguna biasa atau tidak resmi tidak ditampilkan ke dalam *tweets* yang telah dikelompokkan berdasarkan kategori. Kemudian yang kedua, tidak diperhatikannya *tweets* dalam proses pengkategorian berita karena pengkategorian hanya berdasarkan akun saja (Phuvipadawat and Murata, 2010).

Pada penelitian yang pernah ada sebelumnya mengenai pengklasifikasian Twitter berdasarkan kategori, didapatkan hasil akurasi yang kurang maksimal dengan menggunakan metode Naïve Bayes (Perdana, 2013). Hal itu terjadi karena *tweets* yang ada di Twitter berupa *short-text*. *Short-text* memang agak susah untuk diklasifikasikan karena sebuah *short-text* mempunyai beberapa karakteristik yang melekat di setiap teksnya (Zelikovitz and Marquez, 2005). Setiap kata yang akan dijadikan karakterstik dari sebuah *short-text* harus dimasukkan ke dalam kelompok kata yang sesuai dengan kategorinya. Salah satu teknik dalam perbaikan klasifikasi *tweets* berupa *short-text* yaitu dengan menambahkan *query* baru atau *query expansion*.

Penelitian sebelumnya yang dilakukan oleh Bandyopadhyay, dkk, mencari *tweet* relevan yang cocok dengan query dari user dengan menggunakan teknik *query expansion* didapatkan hasil yang cukup bagus dibandingkan tanpa menggunakan *query expansion* (Bandyopadhyay et al., 2012). *Query expansion* merupakan suatu teknik untuk menambahkan *query* tambahan yang belum ada di *query* awal. Salah satu teknik *query expansion* yang dapat digunakan adalah algoritme apriori (Rungsawang et al., 1999). Algoritme apriori ini akan digunakan sebagai metode untuk membuat kamus kedekatan kata.

Berdasarkan permasalahan di atas, maka penulis ingin mencoba menambahkan metode *query expansion* pada klasifikasi *tweets* dan melakukan klasifikasi berdasarkan isi/konten dari *tweets*, bukan berdasarkan akunnya, agar hasil dari klasifikasi lebih baik dan lebih akurat lagi. Sehingga, diharapkan dapat memberikan dampak positif dalam perkembangan media sosial Twitter khususnya di Indonesia agar pengguna lebih mudah dalam membaca suatu konten berita di Twitter berdasarkan kebutuhan informasinya.

2. KAJIAN PUSTAKA

2.1. Twitter

Twitter adalah alat percakapan unik yang memungkinkan kita untuk mengirim dan menerima pesan singkat yang disebut *tweet* dalam komunitas Twitter. Komunitas Twitter terdiri dari :

1. *Following*

Komunitas didefinisikan dengan mengikuti pengguna Twitter lainnya. Kita dapat melihat *tweets* yang dibuat oleh semua orang yang kita ikuti. Mengikuti seseorang dapat diartikan kita berlangganan dengan *tweets* mereka.

2. *Followers*

Orang lain yang membaca *tweets* kita dan memutuskan untuk mengikuti kita. *Followers* atau pengikut dapat melihat *tweets* yang kita

kirimkan.

Tweets adalah pesan singkat, memiliki panjang yang terdiri dari 140 karakter, sehingga gampang untuk disaring (Crow Communications, 2011).

2.2. Klasifikasi Teks

Klasifikasi teks adalah sebuah teknik teks *mining* yang bertujuan untuk menempatkan teks pada kategori yang sesuai dengan karakteristik dari teks tersebut dengan menggunakan aturan – aturan tertentu. Terdapat dua metode dasar klasifikasi teks, yaitu *Unsupervised Text Classification* dan *Supervised Text Classification* (Suharso, 2008). *Unsupervised Text Classification* merupakan metode klasifikasi teks yang sebelumnya tidak memiliki pola atau aturan. Sedangkan *Supervised Document Classification* merupakan metode klasifikasi dokumen ke dalam pola – pola atau aturan yang sudah ditentukan sebelumnya melalui proses pembelajaran. Dokumen yang telah diklasifikasi sebelumnya ini disebut dengan dokumen latih, dokumen *training* atau *training sets* (Kumalasari, 2011).

2.3. *Text Preprocessing*

Pada tahap *text preprocessing* dilakukan beberapa proses untuk menyiapkan *tweets* untuk menjadi dokumen teks yang siap diolah pada tahap selanjutnya. Pada tahap ini pada umumnya terdapat beberapa proses, antara lain *tokenizing*, *filtering*, *stemming*, dan *term weighting* (Garcia, 2005).

2.3.1. *Tokenizing*

Proses *tokenizing* adalah proses pemotongan *string* masukan berdasarkan tiap kata yang terdapat pada kalimat. Setiap huruf *input* akan diubah menjadi huruf kecil. Semua tanda baca dan tanda hubung akan dihapuskan, termasuk semua karakter selain huruf alfabet (Garcia, 2005).

2.3.2. *Filtering*

Pada tahap *filtering* adalah tahap mengambil kata – kata penting dari hasil *token* (Garcia, 2005). Dalam proses ini dilakukan pembuangan kata yang tidak penting (*stoplist*) atau menyimpan kata yang penting (*wordlist*). *Stoplist* adalah daftar kata yang sering digunakan dan tidak menjelaskan isi dari dokumen, atau dapat disebut dengan istilah *stopword*. Contoh *stopwords* adalah ”yang”, “dan”, “di”, “dari” dan seterusnya.

2.3.3. *Stemming*

Stemming merupakan tahap mecari root kata dari tiap kata hasil filtering. Pada tahap ini dilakukan proses pengembalian berbagai bentukan kata kedalam suatu representasi yang sama (Langgeni et al., 2010).

2.3.4. *Weighting*

Setelah serangkaian *tokenizing*, *filtering*, serta *stemming* langkah utama selanjutnya dari proses klasifikasi dokumen adalah pembobotan (Zhang, Niu and Nie, 2009). Metode yang akan digunakan dalam penelitian ini adalah *TF-IDF* (*term frequency* / *inverse document frequency*).

2.3.5. *Fuzzy K-Nearest Neighbour*

Inti dari metode ini adalah menentukan nilai keanggotaan sebagai jarak vektor dari *K-Nearest Neighbor* dan nilai keanggotaan pada suatu kelas tertentu. Dengan demikian sebuah dokumen D akan memiliki nilai keanggotaan pada kelas tertentu pula. Pengklasifikasian dengan menggunakan metode Fuzzy K-NN nantinya juga akan memilih nilai keanggotaan kelas pada dokumen D (μi(x)) yang paling tinggi (Zhang, Niu and Nie, 2009).

Menurut James M. Keller, Michael R. Gray dan Givens James A (1985) untuk memberikan nilai keanggotaan dokumen *testing* x dapat dihitung dengan Persamaan (1) dibawah ini:

$$\mu_i(\boldsymbol{x}) = \frac{\sum_{j=1}^{k} \mu_{ij} \left(sim\,(x,xj)^{\frac{2}{m-1}}\right)}{\sum_{j=1}^{k} \left(sim\,(x,xj)^{\frac{2}{m-1}}\right)} \quad (1)$$

Keterangan:

$sim(x, xj)$: *similarity* antara dokumen uji x dengan dokumen latih xj

x_j : dokumen *training* yang telah dihitung dengan persamaan 2.8

m : bobot pangkat (*weight exponent*) yang besarnya m > 1

μ_{ij} : nilai keanggotaan kelas ke-i pada tetangga ke-j

2.3.5. *Query Expansion*

Query expansion adalah memanjangkan *query* yang dimasukkan *user* dengan menambahkan beberapa *term* kedalamnya. Ekspansi *query* merupakan salah satu teknik yang dapat digunakan dalam membantu pengguna dalam memberikan *query* yang baik.

Ekspansi *query* dapat berperan sebagai penghubung karena adanya *vocabulary gaps* antara *query* dan dokumen. *Query* yang dimasukkan oleh *user* pada umumnya pendek dan *query expansion* dapat melengkapkan informasi yang ingin dicari *user* (Poernomo dan Gunawan, 2015).

2.3.6. Apriori

Algoritme apriori yang bertujuan untuk menemukan *frequent itemsets* dijalankan pada sekumpulan data. Analisis apriori didefinisikan suatu proses untuk menemukan semua aturan apriori yang memenuhi syarat minimum untuk *support* dan syarat minimum untuk *confidence*. *Support* adalah nilai penunjang, atau presentase kombinasi sebuah *item* dalam *database*. Sedangkan *confidence* adalah nilai kepastian yaitu kuatnya hubungan antar *item* dalam sebuah apriori (Syaifullah, 2010).

Support dan *confidence* dari asosiasi *term* $t_i \rightarrow t_j$ didefinisikan sebagai berikut:

$$D(t_i, t_j) = D(t_i) \cap D(t_j) \quad (2)$$

Keterangan:
$D(t_i)$: jumlah dokumen yang memiliki kata (*term*) t_i
$D(t_j)$: jumlah dokumen yang memiliki kata (*term*) t_j
$D(t_i) \cap D(t_j)$: jumlah dokumen yang memiliki kedua kata (*term*) t_i dan t_j

$$Sup_{t_i \rightarrow t_j} = \frac{\|D(t_i, t_j)\|}{\|D\|} \quad (3)$$

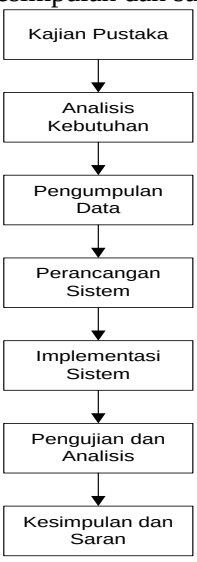
Keterangan:
$\|D(t_i, t_j)\|$: jumlah total dokumen yang memiliki kedua kata (*term*) t_i dan t_j
$\|D\|$: jumlah dokumen di dalam dataset

$$Conf_{t_i \rightarrow t_j} = \frac{\|D(t_i, t_j)\|}{\|D(t_i)\|} \quad (4)$$

Keterangan:
$\|D(t_i, t_j)\|$: jumlah total dokumen yang memiliki kedua kata (*term*) t_i dan t_j
$\|D(t_i)\|$: jumlah total dokumen yang memiliki kata (*term*) t_i

3. METODOLOGI

Metodologi penelitian ini dilakukan dalam beberapa tahap, yaitu: kajian pustaka, analisis kebutuhan, pengumpulan data, perancangan sistem, implementasi sistem, pengujian dan analisis, serta kesimpulan dan saran.

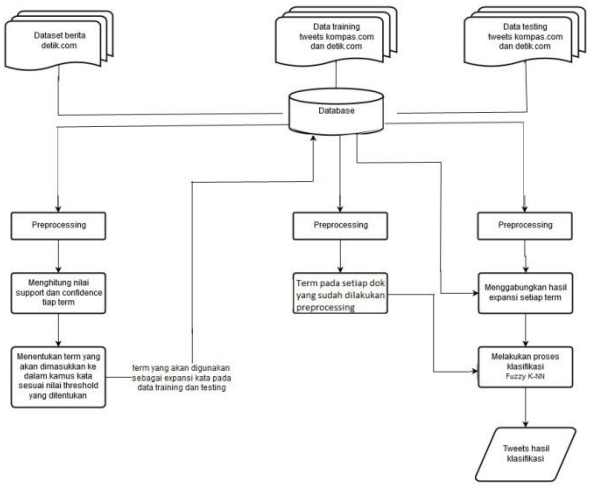


Gambar 1. Diagram Alir Penelitian

4. ANALISIS DAN PERANCANGAN

4.1. Deskripsi Umum Sistem

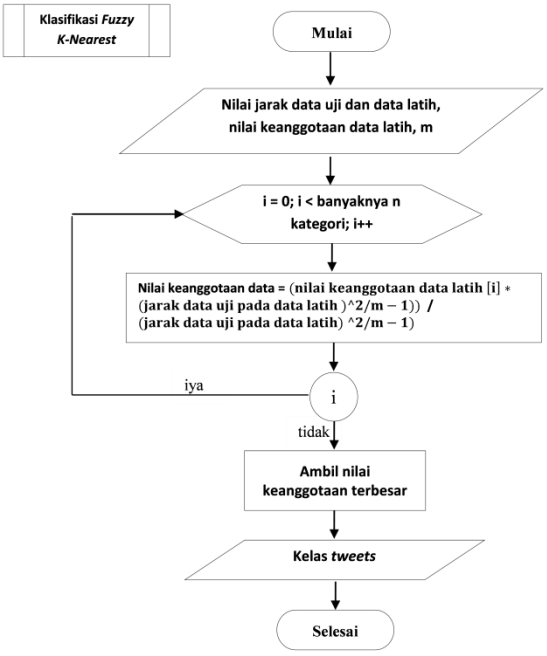
Cara kerja sistem dijelaskan oleh Gambar 2. Secara umum, prinsip kerja sistem akan menghasilkan hasil akhir berupa klasifikasi *tweets*. Proses klasifikasi dilakukan dengan menggunakan metode *Fuzzy K-Nearest Neighbour*. Sebelum dilakukan proses klasifikasi, terlebih dahulu sistem akan melakukan proses expansi kata dengan *query expansion* pada *dataset*. Proses expansi kata didapatkan dengan algoritme *support* dan *confidence*. Untuk pengambilan kata yang akan dimasukkan ke kamus kata untuk digunakan sebagai expansi kata, adalah kata yang mempunyai nilai *support* dan *confidence* lebih dari *threshold*. *Dataset*, data latih, dan data uji sebelumnya dilakukan *preprocessing* untuk menghilangkan kata-kata yang dianggap tidak penting oleh sistem. Proses ekspansi kata hanya dilakukan pada data uji saja. Untuk data yang digunakan, *dataset* berupa data berita yang diambil dari portal berita *detik.com*, serta data latih dan data uji menggunakan *tweets* dari *kompas.com* dan *detik.com*.



Gambar 2. Deskripsi Umum Sistem

4.2. Penyelesaian Metode *Fuzzy K-Nearest Neighbour*

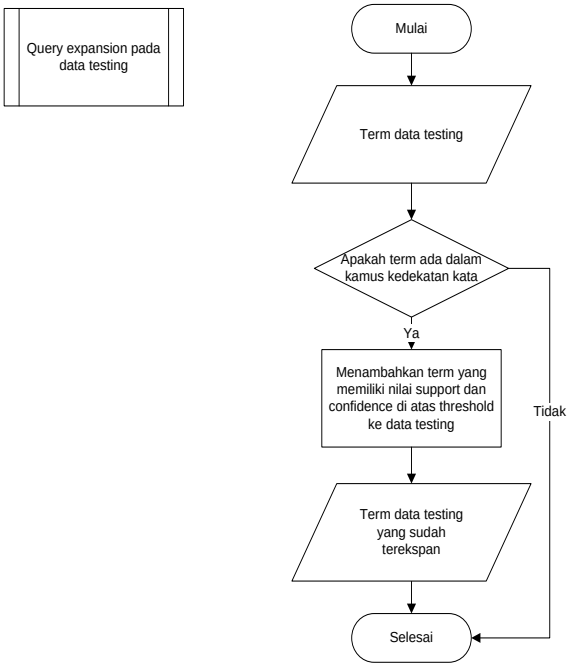
Fuzzy K-Nearest Neighbour merupakan pengembangan dari metode *K-Nearest Neighbour*. Metode ini digunakan dalam pengklasifikasian *tweets* pada Twitter. Pengklasifikasian dengan metode ini memiliki banyak kelebihan dibandingkan dengan metode *K-Nearest Neighbour*, yaitu terdapat perhitungan keanggotaan kelas pada setiap data latihnya. Penyelesaian metode *Fuzzy K-Nearest Neighbour* dijelaskan pada diagram alir dibawah ini (Gambar 3.).



Gambar 3. Alur Proses Metode *Fuzzy K-Nearest Neighbour*

4.3 Penyelesaian Metode *Query Expansion*

Query Expansion diperlukan dalam proses klasifikasi agar dihasilkan tingkat akurasi yang tinggi. Cara kerja dari metode ini adalah dengan menambahkan kata yang sering muncul bersamaan dari kamus kedekatan kata ke dalam data uji berupa *tweets*.



Gambar 4. Alur Proses Metode *Query Expansion*

5. IMPLEMENTASI

5.1. Batasan Implementasi

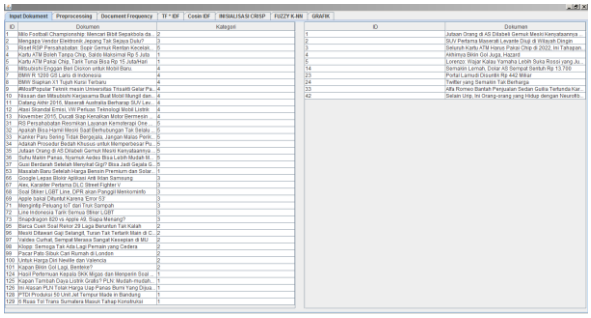
Batasan implementasi merupakan batasan proses yang dapat dilakukan oleh sistem berdasarkan perancangan yang telah diuraikan pada bab sebelumnya. Batasan implementasi bertujuan untuk membuat sistem sesuai dengan ruang lingkup yang jelas dan tidak keluar dari tujuan utama dari sistem. Batasan-batasan implementasi sistem tersebut adalah sebagai berikut:

1. Klasifikasi *Tweets* Pada Twitter Menggunakan Metode *Fuzzy k-Nearest Neighbours* (*Fuzzy K-NN*) Dan *Query Expansion* berbasis Apriori dirancang dan dijalankan menggunakan aplikasi *dekstop* berbahasa JAVA.
2. Metode penyelesaian masalah yang digunakan adalah *Fuzzy k-Nearest Neighbours* (*Fuzzy K-NN*) dan *Query Expansion.*

3. Data yang digunakan sebagai data latih dan data uji merupakan *tweets* yang didapatkan dari *kompas* dan *detik.*
4. Keluaran yang dikeluarkan berupa hasil klasifikasi pada kategori ekonomi, olahraga, teknologi, otomotif, dan kesehatan.

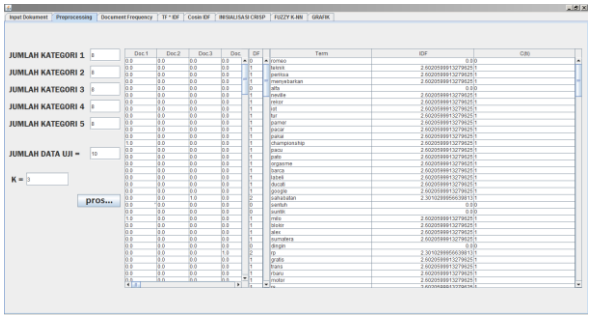
5.2. Implementasi Antarmuka

Halaman ini menampilkan data latih dan data uji yang digunakan dalam pengujian. Implementasi halaman data latih dan data uji ditunjukkan pada Gambar 5. berikut.



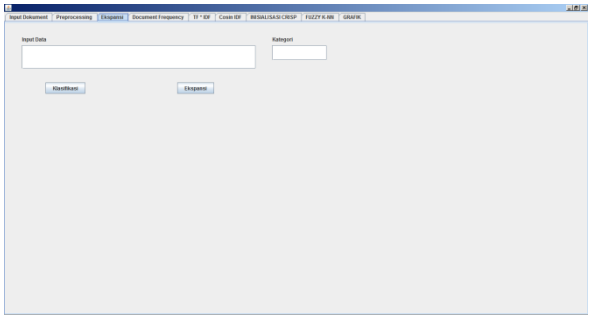
Gambar 5. Antarmuka Data Latih dan Data Uji

Halaman ini menampilkan *vector space model* hasil *preprocessing* dan perhitungan *Term Frequency* (TF_i), *Document Frequency* (DF_i), *Inverse Document Frequency* (IDF) yang digunakan dalam proses pembobotan TF.IDF. Implementasi halaman tersebut ditunjukkan pada pada Gambar 6. berikut.



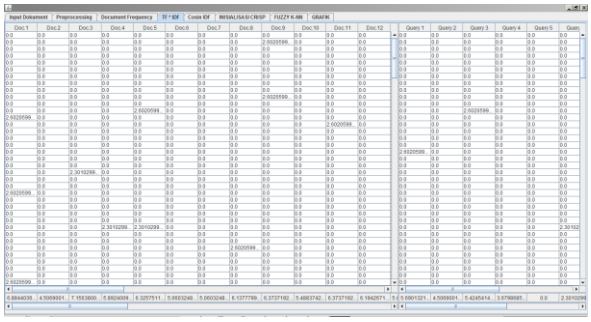
Gambar 6. Antarmuka *Preprocessing*

Halaman ini digunakan untuk memasukkan data secara manual, kemudian dilakukan klasifikasi dan ekspansi kata, serta ditampilkan hasil dari kategorinya. Implementasi halaman ekspansi ditunjukkan pada Gambar 7. berikut.



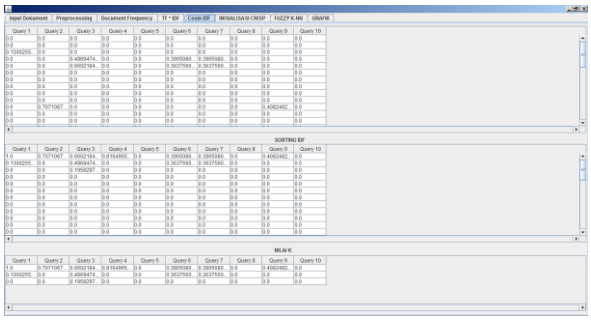
Gambar 7. Antarmuka Ekspansi

Halaman ini menampilkan *vector space model* hasil perhitungan pembobotan TF.IDF. Implementasi halaman tersebut ditunjukkan pada pada Gambar 8. berikut.



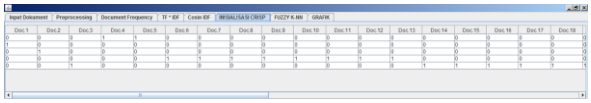
Gambar 8. Antarmuka Pembobotan TF.IDF

Halaman ini menampilkan *vector space model* hasil perhitungan *cosine similarity* antara data uji dan latih dari hasil pembobotan TF. IDF dan menentukan nilai variabel k (tetangga terdekat). Implementasi halaman tersebut ditunjukkan pada pada Gambar 9. berikut.



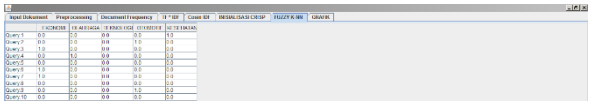
Gambar 9. Antarmuka *Cosine Similarity*

Halaman ini untuk menentukan nilai keanggotaan data latih dalam himpunan *Fuzzy* dengan menggunakan inisialisasi *crisp.* Implementasi halaman tersebut ditunjukkan pada pada Gambar 10. berikut.



Gambar 10. Antarmuka Inisialisasi *Crisp*

Halaman ini untuk menentukan nilai keanggotaan data latih dalam himpunan *Fuzzy* dengan menggunakan inisialisasi *crisp*. Implementasi halaman tersebut ditunjukkan pada pada Gambar 11. berikut.



Gambar 11. Antarmuka Klasifikasi *Fuzzy K-Nearest Neighbor*

Halaman ini menampilkan grafik akurasi dari hasil perhitungan *Fuzzy K-Nearest Neighbor* beserta *K-Nearest Neighbor*. Implementasi halaman tersebut ditunjukkan pada pada Gambar 12. berikut.

Gambar 12. Antarmuka Akurasi

6. PENGUJIAN DAN ANALISIS

Pada pengujian ini digunakan data uji sebanyak 10 data dari masing-masing kategori atau sebanyak 50 data untuk semua kategori. Sedangkan data latih yang digunakan sebanyak 40 data dari masing-masing kategori atau sebanyak 200 data untuk semua kategori. Kemudian nilai *k* yang digunakan adalah 3, 5, 10, dan 17.

Pengujian ini terdiri dari tiga jenis pengujian yaitu sebagai berikut:

1. Pengujian Dengan *Preprocessing* dan Variasi *Query Expansion*
2. Pengujian Dengan *Query Expansion* dan Variasi *Preprocessing*
3. Pengujian *Preprocessing* dan *Query Expansion* Secara Keseluruhan

6.1. Pengujian Dengan *Preprocessing* dan Variasi *Query Expansion*

Pengujian ini menjelaskan tentang pengujian pada data latih dan data uji yang dilakukan *preprocessing* dan pada data uji yang dilakukan variasi *query expansion*. Pengujian ini mencakup dua lingkup yaitu, skenario pengujian dan analisis terhadap hasil pengujian tersebut.

6.1.1. Skenario Pengujian Dengan *Preprocessing* dan Variasi *Query Expansion*

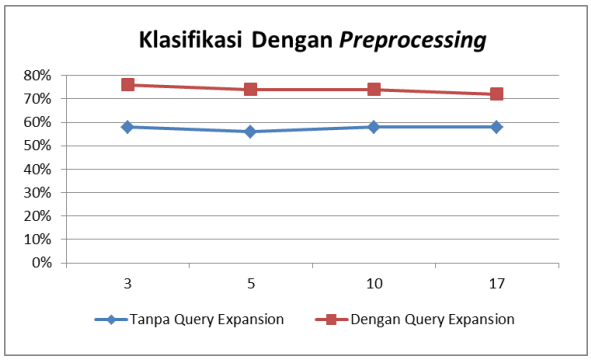
Pengujian ini dilakukan untuk mengetahui pengaruh *query expansion* pada data uji terhadap hasil klasifikasi pada data latih dan data uji yang dilakukan *preprocessing*. Pada pengujian ini, data latih dan data uji sudah dilakukan *preprocessing*. Sedangkan data uji dilakukan variasi, yaitu dengan menambahkan algoritme *query expansion* dan tanpa menambahkan algoritme *query expansion*. Hasil pengujian dapat dilihat pada Tabel 1. berikut.

Tabel 1. Hasil Pengujian Tanpa dan Dengan *Query Expansion*

Nilai *k*	Akurasi Tanpa *Query Expansion*	Akurasi Dengan *Query Expansion*
3	58%	76%
5	56%	74%
10	58%	74%
17	58%	72%

6.1.2. Analisis Pengujian Tanpa dan Dengan *Query Expansion*

Pengujian variasi *query expansion* pada data uji dengan *preprocessing* pada data latih dan data uji menghasilkan akurasi terbaik pada angka 76%. Hasil terbaik ini didapat pada pengujian dengan *query expansion*.



Gambar 13. Grafik Akurasi Hasil Pengujian Tanpa dan Dengan *Query Expansion*

Dari grafik pengujian di atas, hasil akurasi kedua pengujian tersebut memiliki nilai akurasi

yang stabil walaupun nilai *k*-nya bertambah. Hal ini menunjukkan bahwa besar nilai *k* tidak terlalu berpengaruh pada proses klasifikasi baik dengan *query expansion*, maupun tanpa *query expansion*. Kemudian klasifikasi dengan *query expansion* memiliki nilai yang lebih baik dibandingkan klasifikasi tanpa *query expansion* walaupun keduanya dilakukan *preprocessing*.

Penambahan *query expansion* pada data uji memberikan hasil klasifikasi yang lebih baik pada klasifikasi *short-text*, karena kata pada data uji terkadang tidak muncul pada data latih sehingga menyebabkan hasil klasifikasi yang kurang sempurna. Dengan *query expansion*, data uji ditambahkan kata baru dahulu sebelum dilakukan proses klasifikasi. Tujuannya agar pembendaharaan kata yang ada pada data uji menjadi semakin banyak, sehingga proses klasifikasi akan menghasilkan akurasi yang lebih baik jika dibandingkan tanpa menggunakan *query expansion*.

6.2. Pengujian Dengan *Query Expansion* dan Variasi *Preprocessing*

Pengujian ini menjelaskan tentang pengujian pada data latih dan data uji yang dilakukan *query expansion* dan pada data uji yang dilakukan variasi *preprocessing*. Pengujian ini mencakup dua lingkup yaitu, skenario pengujian dan analisis terhadap hasil pengujian tersebut.

6.2.1 Skenario Pengujian Dengan *Preprocessing* dan Variasi *Query Expansion*

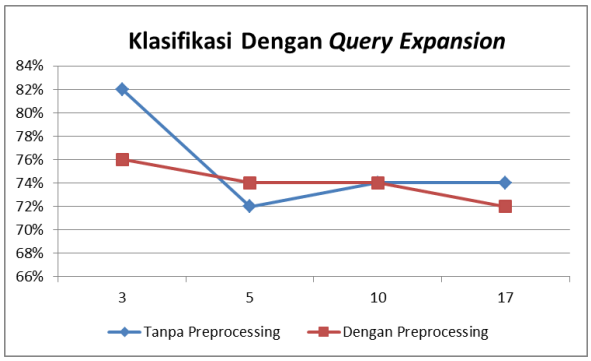
Pengujian ini dilakukan untuk mengetahui pengaruh *preprocessing* pada data latih dan data uji terhadap hasil klasifikasi pada data uji yang dilakukan *query expansion*. Pada pengujian ini, data uji sudah dilakukan *query expansion*. Sedangkan data latih dan data uji dilakukan variasi, yaitu dengan menggunakan *preprocessing* dan tanpa menggunakan *preprocessing*. Hasil pengujian dapat dilihat pada Tabel 2. berikut.

Tabel 2. Hasil Pengujian Tanpa dan Dengan *Preprocessing*

Nilai *k*	Akurasi Tanpa *Preprocessing*	Akurasi Dengan *Preprocessing*
3	82%	76%
5	72%	74%
10	74%	74%
17	74%	72%

6.2.2. Analisis Pengujian Tanpa dan Dengan *Preprocessing*

Pengujian variasi *preprocessing* pada data latih dan data uji dengan penambahan *query expansion* pada data uji menghasilkan akurasi terbaik pada angka 82%. Hasil terbaik ini didapat pada pengujian tanpa *preprocessing*.



Gambar 14. Grafik Akurasi Hasil Pengujian Tanpa dan Dengan *Preprocessing*

Dari grafik pengujian di atas, hasil akurasi pengujian tanpa *preprocessing* tersebut memiliki nilai akurasi yang bervariasi seiring dengan bertambahnya nilai *k*. Hal ini menunjukkan bahwa besar nilai *k* berpengaruh pada proses klasifikasi tanpa *preprocessing*. Kemudian klasifikasi tanpa *preprocessing* dengan nilai *k* = *3* memiliki nilai akurasi yang lebih baik dibandingkan klasifikasi menggunakan *preprocessing*. Hal ini disebabkan *stemming* yang digunakan pada *preprocessing* yaitu *Stemming* Nazief Andriani masih kurang sempurna dalam menghapus awalan atau akhiran pada kata-kata tertentu. Sehingga kata yang sudah dilakukan *preprocessing* malah menjadi tidak ada di data latih dan menyebabkan hasil kategori tidak tepat.

6.3. Pengujian *Preprocessing* dan *Query Expansion* Secara Keseluruhan

Pengujian ini menjelaskan tentang pengujian secara keseluruhan. Pada pengujian ini dilakukan perbandingan hasil pada kedua pengujian sebelumnya ditambah dengan pengujian tanpa *preprocessing* dan tanpa *query expansion*.

6.3.1. Skenario Pengujian *Preprocessing* dan *Query Expansion* Secara Keseluruhan

Pengujian ini dilakukan untuk menganalisis hasil pengujian terbaik dari semua pengujian yang telah dilakukan. Hasil pengujian dapat

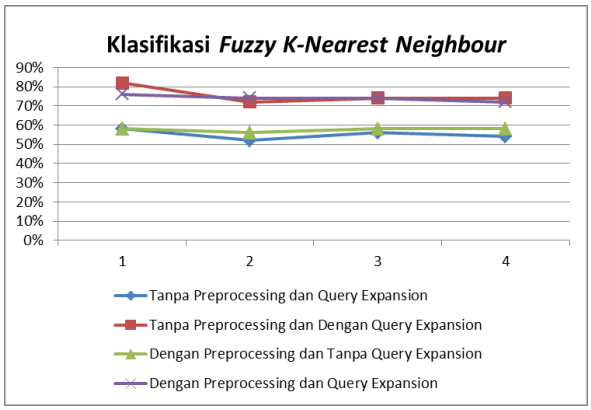
dilihat pada Tabel 3. berikut.

Tabel 3. Hasil Pengujian Secara Keseluruhan

Nilai *k*	Tanpa *Preprocessing* dan *Query Expansion*	Tanpa *Preprocessing* dan Dengan *Query Expansion*	Dengan *Preprocessing* dan Tanpa *Query Expansion*	Dengan *Preprocessing* dan *Query Expansion*
3	58%	82%	58%	76%
5	52%	72%	56%	74%
10	56%	74%	58%	74%
17	54%	74%	58%	72%

6.3.2. Analisis Pengujian Dengan Preprocessin dan Query Expansion Secara Keseluruhan

Berdasarkan pengujian yang telah dilakukan, didapatkan hasil akurasi terbaik pada pengujian tanpa menggunakan *preprocessing* dan dengan menggunakan *query expansion* yaitu sebesar 82% pada nilai *k* = 3.



Gambar 15. Grafik Akurasi Hasil Pengujian Secara Keseluruhan

Analisis dari masing-masing pengujian adalah sebagai berikut:

1. Pengujian tanpa *preprocessing* dan tanpa *query expansion* maka data uji tidak berubah atau tetap seperti asalnya. Karena tidak dilakukan *preprocessing* dan tidak ditambah *query expansion* maka hasil akurasinya pun tidak terlalu bagus sama seperti pengujian dengan *preprocessing* dan tanpa *query expansion*.
2. Pengujian tanpa *preprocessing* dan dengan *query expansion* maka data uji hanya ditambah kata baru dari ekspansi kata saja. Hasil akurasi terbaik ada pada pengujian ini disebabkan tidak dilakukan *preprocessing* terutama pada bagian *stemming* yang sudah dijelaskan sebelumnya.
3. Pengujian dengan *preprocessing* dan tanpa *query expansion* maka data uji hanya dilakukan *preprocessing* saja. Hasil akurasi tidak terlalu bagus sama seperti pengujian tanpa *preprocessing* dan tanpa *query expansion*.
4. Pengujian dengan *preprocessing* dan dengan *query expansion* maka data uji dilakukan *preprocessing* dan ditambahkan *query expansion*. Hasil akurasi lebih baik dibanding pengujian lain dan hasilnya lebih stabil seiring bertambahnya nilai *k*.

7. PENUTUP

7.1 Kesimpulan

Berdasarkan hasil pengujian dan analisis klasifikasi *tweets* pada Twitter dengan menggunakan metode *Fuzzy K-Nearest Neighbour* (*Fuzzy K-NN*) dan *Query Expansion* berbasis Apriori dapat diambil kesimpulan sebagai berikut:

1. Untuk membantu pengguna dalam memilih *tweets* yang dibutuhkan berdasarkan kebutuhannya, *tweets* yang ada di Twitter bisa diklasifikasikan menjadi beberapa kategori. Proses untuk melakukan klasifikasi yang dilakukan peneliti meliputi 3 tahap yaitu, *preprocessing*, penambahan kata baru atau ekspansi kata, serta klasifikasi dengan metode *Fuzzy K-Nearest Neighbour* (*Fuzzy K-NN*).
2. *Text preprocessing* tidak berpengaruh secara signifikan terhadap proses klasifikasi. Akurasi tertinggi jika menggunakan preprocessing adalah sebesar 76%. Sedangkan, tanpa menggunakan *text preprocessing*, hasil akurasi yang didapatkan adalah sebesar 82%. Oleh karena itu dapat disimpulkan bahwa hasil klasifikasi tidak dipengaruhi secara signifikan oleh *text preprocessing*.
3. Algoritme *query expansion* sangat berpengaruh terhadap proses klasifikasi *tweets* karena sebuah *tweets* hanya barupa *short-text* sehingga sangat sulit diklasifikasikan. Akurasi tertinggi yang didapatkan jika menggunakan *query expansion* adalah sebesar 76%. Kemudian akurasi tertinggi yang diperoleh jika tanpa menggunakan *query* expansion adalah sebesar 58%. Oleh karena itu untuk mengklasifikasikan sebuah *tweets* diperlukan algoritme *query expansion* untuk

mendapatkan hasil klasifikasi yang lebih baik.

7.2 Saran

Saran yang dapat diberikan untuk pengembangan penelitian selanjutnya, antara lain:

1. Proses expansi dengan algoritme *query expansion* membutuhkan waktu yang lama saat proses *running*. Oleh karena, itu perlu dilakukan optimasi algoritme agar proses *running* data tidak memakan waktu yang sangat lama.
2. Diperlukan pengujian klasifikasi dengan menggunakan algoritme klasifikasi selain *Fuzzy K-Nearest Neighbour* atau *Naïve Bayes* untuk mengetahui apakah algoritme *query expansion* juga dapat diterapkan untuk metode klasifikasi lainnya serta untuk mengetahui metode klasifikasi yang memiliki nilai akurasi terbaik jika digabungkan dengan *query expansion*.
3. *Text preprocessing* yang dilakukan peneliti belum sepenuhnya sempurna karena masih belum bisa menangani masalah seperti penghapusan awalan dan akhiran secara efektif pada tahap *stemming*. Oleh karena itu, pada penelitian selanjutnya diperlukan algoritme *stemming* yang lebih baik lagi untuk hasil klasifikasi yang lebih akurat.

8. DAFTAR PUSTAKA

Bandyopadhyay, A., Ghosh, K., Majumder, P. and Mitra, M. (2012). Query expansion for microblog retrieval. *International Journal of Web Science*, 1(4), p.368.

Crow Communications. (2011). Twitter For Beginners. *Social Media DIY Workshop for Small Business : United States*.

Garcia, E., Dr. (2005). Document Indexing Tutorial, http://www.miislita.com/information-retrieval-tutorial/indexing.html, diakses pada tanggal 4 Maret 2016.

Keller, J., Gray, M. and Givens, J. (1985). A fuzzy K-nearest neighbor algorithm. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-15(4), pp.580-585.

Langgeni, D. P., dkk. (2010). Aplikasi Logika Fuzzy untuk Pendukung Keputusan Edisi 2. *Graha Ilmu : Yogyakarta*.

Perdana, R.S. (2013). Pengkategorian Pesan Singkat Berbahasa Indonesia Pada Jejaring Sosial Twitter Dengan Metode Klasifikasi Naïve Bayes. *S1. Program Teknologi Informasi dan Ilmu Komputer, Universitas Brawijaya*.

Phuvipadawat, S. and Murata, T. (2010). Breaking News Detection and Tracking in Twitter.*2010 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology*.

Poernomo T.P., B. dan Gunawan, Ir. (2015). Sistem Information Retrieval Pencarian Kesamaan Ayat Terjemahan Al Quran Berbahasa Indonesia Dengan Query Expansion Dari Tafsirnya. *Teknik Informatika Sekolah Tinggi Manajemen Informatika dan Komputer Asia, Teknik Informatika Sekolah Tinggi Teknik Surabaya : Surabaya.*

Sriram, B., Fuhry, D., Demir, E., Ferhatosmanoglu, H. and Demirbas, M. (2010). Short text classification in twitter to improve information filtering. *Proceeding of the 33rd international ACM SIGIR conference on Research and development in information retrieval - SIGIR '10*.

Syaifullah, M. A. (2010). Implementasi Data Mining Algoritme Apriori Pada Sistem Penjualan. *Sekolah Tinggi Manajemen Informatika dan Ilmu Komputer : Yogyakarta*.

Zelikovitz, S. and Marquez, F. (2005). Transductive Learning For Short-Text Classification Problems Using Latent Semantic Indexing. *International Journal of Pattern Recognition and Artificial Intelligence*, 19(02), pp.143-163.

Zhang, J., Niu, Y. and Nie, H. (2009). Web Document Classification Based on Fuzzy k-NN Algorithm. *2009 International Conference on Computational Intelligence and Security*.