KATEGORISASI DOKUMEN TEKS SECARA MULTI LABEL MENGGUNAKAN FUZZY C-MEANS DAN K-NEAREST NEIGHBORS PADA ARTIKEL BERBAHASA INDONESIA

Rio Bayu Afrianto, Lisa Yuli Kurniawati
Jurusan Teknik Informatika, Fakultas Teknologi Informasi, Institut Teknologi Sepuluh Nopember
Kampus ITSSukolilo, Surabaya 60111
Email: rio11@mhs.if.its.ac.id

ABSTRAK

Permasalahan kategorisasi dokumen berperan penting dalam sistem temu kembali informasi. Kategorisasi dokumen teks yang telah ada biasanya hanya dapat melakukan klasifikasi dengan satu label saja untuk satu dokumen. Padahal dalam kenyataannya, sebuah artikel dapat memuat lebih dari satu kategorisehingga label dokumen yang diberikan dapat berjumlah lebih dari satu.Untuk itulah, penelitian ini mengusulkan sebuah metode baru untuk kategorisasi dokumen teks secara multi label dengan menggunakan fuzzy c-means dan k-nearest neighbors. Fuzzy c-means melakukan pengelompokan dokumen yang serupa terlebih dahulu sebelum proses pemberian label. Kemudian, penentuan label dokumen ditentukan oleh k-dokumen terdekat pada kelompok dokumen yang serupa. Uji coba dilakukan terhadap dokumen berita online sejumlah 175 dokumen yang terdiri atas tiga kategori label. Hasil uji coba menunjukkan bahwa metode yang diusulkan memberikan performa lebih baik dibanding metode lain. Hal ini ditunjukkandengan nilaiF1 sebesar 73,39% dan BEP sebesar 75,22%.

***Kata Kunci**: Fuzzy c-means*, Kategorisasi multi-label, *K-nearest neighbors*.

1. PENDAHULUAN

Kategorisasi dokumen teks secara *multi*-label adalah masalah yang berperan penting dalam sistem temu kembali informasi. Klasifikasi *multi* label berbeda dengan klasifikasi *single* label. Klasifikasi *single* label akan mengklasifikasikan suatu dokumen ke dalam satu kategori dokumen saja. Sedangkan klasifikasi *multi* label dapat mengelompokkan suatu dokumen ke lebih dari satu kategori. Sebagai contoh, sebuah artikel ilmiah yang didalamnya membahas secara rinci mengenai deteksi dini kanker ovarium dengan pendekatan bioinformatika dapat memiliki lebih dari satu label dokumen yaitu: kesehatan, bioinformatika dan wanita.

Telah banyak penelitian sebelumnya yang membahas mengenai klasifikasi dokumen teks seperti [1] yang menggunakan algoritma *k-nearest neighbors*, menggunakan informasi ketetanggaan untuk menentukan label dokumen. Metode lain menggunakan konsep probabilitas diusulkan oleh Joachim pada [2] yakni dengan menggunakan pendekatan probabilistik *bayesian*. Selain itu, ada juga penelitian yang diajukan oleh [3] dengan menggunakan algoritma *decision rule* dan [4] dengan menggunakan metode *support vector machine* (SVM). Namun penelitian diatas berfokus pada kategorisasi dokumen teks yang *single* label.

Untuk mengklasifikasikan artikel ke dalam *multi* label, Tsoumakas & Katakis melakukan pendekatan dengan mentransformasikan klasifikasi *multi* label menjadi beberapa kasus klasifikasi *single* label [5]. Dengan melakukan transformasi *single* label maka tiap-tiap data dilakukan pengecekan terhadap semua label yang ada. Pengecekan yang dilakukan akan menentukan apakah data tersebut "termasuk" atau "tidak termasuk" pada suatu label. Hal ini kurang efisien karena memerlukan waktu komputasi yang cukup lama untuk mengecek suatu dokumen ke semua label. Selain itu, metode ini tidak memperhitungkan korelasi antara label yang satu dengan label yang lain.

Beberapa pendekatan klasifikasi multi label yang lain telah diusulkan. Zhang dan Zhou menggunakan *back propagation neural network* (BP-MLL) yang digunakan untuk klasifikasi *single* label dengan memperhitungkan kriteria tiap-tiap label [6]. Penelitian lain yang dilakukan oleh Zhang dengan menggunakan algoritma MLKNN untuk klasifikasi dokumen teks berbahasa Inggris secara *multi* label dengan menggunakan metode *k-nearest neighbors*. Sebuah dokumen dalam koleksi dokumen akan dicari *k*-tetangga terdekatnya, kemudian digunakan metode *maximum a posteriori* (MAP) untuk menentukan label dokumen tersebut [7].

Penelitian kategorisasi dokumen teks secara *multi* label pada artikel berbahasa Indonesia cukup jarang dijumpai.Untuk itu, penelitian ini mengusulkan sebuah metode baru untuk kategorisasi dokumen teks berbahasa Indonesia dengan menggunakan FCM-KNN. Prinsip dari metode ini adalah dengan melakukan pengelompokan dokumen yang serupa terlebih

dahulu sebelum proses pemberian label. Pengelompokkan dokumen dilakukan dengan menggunakan algoritma *fuzzy c-means*.

Fuzzy C-Means (FCM) merupakan salah satu metode*clustering* yang merupakan bagian dari *HardK-Means*[8]. FCM menggunakan konsep pendekatan *fuzzy* sehingga sebuah data dapat menjadi anggota dari semua *cluster* yang ada. Matriks keanggotaan terbentuk dengan tingkat keanggotaan masing-masing data pada setiap *cluster* yang ada yang bernilai antara 0 hingga 1. Tingkat keberadaan data pada suatu *cluster* ditentukan oleh derajat keanggotaannya.

Dengan adanya pengelompokkan dokumen serupa terlebih dahulu diharapkan nantinya dapat meningkatkan performa kategorisasi dokumen teks serta lebih menghemat waktu komputasi untuk memilih *k*-tetangga terdekat dokumen.Hal ini dikarenakan pencarian *k*-dokumen terdekat hanya cukup dicari pada lingkup anggota kelompok dokumen yang serupa saja dan tidak perlu dibandingkan terhadap keseluruhan koleksi dokumen. Setelah itu, *maximum a posteriori*(MAP) digunakan sebagai penentu label suatu dokumen.

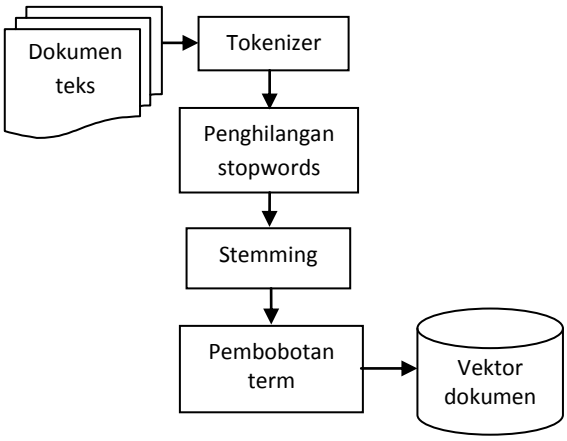
Tulisan ini dibagi menjadi 5 bagian. Latar belakang permasalahan dikemukakan pada bagian 1. Pada bagian 2 dipaparkan penelitian terkait mengenai kategorisasi dokumen teks secara multi-label. Kategorisasi dokumen teks dengan*FCM-KNN* dan uji coba dijelaskan pada bagian 3 dan 4. Pada akhir tulisan diuraikan kesimpulan yang diambil dari hasil penelitian.

2. PENELITIAN TERKAIT

2.1 Representasi Dokumen Teks

Dokumen teks termasuk kedalam jenis data yang tidak terstruktur. Untuk itu, sebelum dilakukan proses kategorisasi teks perlu dilakukan proses transformasi yang dapat mengubah teks-teks menjadi bentuk yang lebih efisien dan lebih siap untuk diproses ke proses selanjutnya. Prinsip dari proses transformasi ini adalah mengubah bentuk teks dan merepresentasikannya ke dalam konsep Vector Space Model (VSM).

Pada tahap transformasi ini dilakukan proses *tokenizer*, yakni pemecahan dokumen teks menjadi beberapa *token* atau kata berdasarkan pembatas berupa spasi atau tanda baca. Selanjutnya dilakukan proses penghapusan kata-kata yang bersesuaian dengan kata pada daftar *stopword*. *Stopword* adalah kata-kata yang dianggap tidak dapat merepresentasikan konten dari suatu dokumen teks. Kemudian, kata-kata yang tersisa setelah penghapusan *stopword* dilakukan *stemming*.



Gambar 1 . Transformasi Dokumen Teks

Stemming adalah proses pengubahan kata menjadi bentuk dasar[9]. Selanjutnya, setiap kata tersebut disebut sebagai *term*.

Untuk setiap *term* yang berbeda satu sama lain didaftar dan diberi bobot *term*. Pembobotan tiap *term* dihitung dengan menggunakan TF-IDF dengan rumusan berikut [10]:

$$w_{t,d} = \left(1 + \log_{10} tf_{t,d}\right) \times \log_{10} n/df_t, \qquad (1)$$

dimana, *n* adalah jumlah dokumen dalam keseluruhan koleksi dokumen, $tf_{t,d}$ merupakan jumlah kemunculan *termt* pada dokumen *d* dan df_t adalah jumlah dokumen yang memuat *termt* pada keseluruhan dokumen.

Sebuahdokumen direpresentasikan sebagai sebuah vektor yang berisi *term-term* pada dokumen tersebut disertai dengan bobot TF-IDF-nya. Bagan proses dari tahap transformasi ini dapat dilihat pada

Gambar 1.

2.2 Kategorisasi Teks Secara Multi-Label

Permasalahan kategorisasi dokumen secara multi-label secara matematis terdiriatas tiga vektor yaitu (D,T,C).D adalah kumpulan vektor dokumen yang dapat ditulis sebagai$D = \{(d_1, y_1), (d_2, y_2) \ldots (d_n, y_n)\}$. Sebuah vektor dokumen terdiri atas vektor d_i dan y_i yakni vektor bobot tiap kata dan label dokumen tersebut.

C merupakan kumpulan kategori label yang ada dan dapat dinotasikan kedalam$C = \{c_1, c_2 \ldots c_r\}$ dimana *r* menyatakan jumlah kategori. Sedangkan, T adalah *term-term* dari seluruh dokumen yang dinyatakan oleh $T = \{t_1, t_2 \ldots t_m\}$, dimana *m* merupakan jumlah *term*.

Perbedaan mendasar antara kategorisasi dokumen secara *single* label dan *multi*-label terletak pada vektor y_i. Jika pada *single* label vektor y_i

berukuran 1. Sedangkan pada *multi*-label vektor $y_i$berukuran p, dimana p>1. Hal tersebut dapat dilihat pada vektor label dokumen y_{ji} berikut:

$$y_{ji} = \begin{cases} 1, jika\ d_i\ termasuk\ pada\ c_j \\ 0, jika\ d_i\ bukan\ termasuk\ pada\ c_j \end{cases} \quad (2)$$

3. FCM-KNN

Secara garis besar studi ini terdiri atas dua tahapan. Tahap pertama adalah tahap pengelompokkan dokumen dengan menggunakan *fuzzy c-means*. Kemudian tahap selanjutnya adalah tahap kategorisasi dokumen teks dengan menggunakan *k-nearest neighbors*.

3.1*Fuzzy C-Means Clustering*

*Fuzzy C-Means*mengelompokkan dokumen teks berdasarkan prinsip dari *fuzzy*.Metode ini pertama kali diperkenalkan oleh Dunn pada tahun 1973 dan dikembangkan lagi oleh Bezdek di tahun 1981[8].Pengelompokkan ke dalam *cluster* ditentukan melalui nilai keanggotaan*fuzzy*. Jika nilai keangggotaan dokumen d_i pada *cluster*$c_j$melebihi suatu batas ambang tertentu, maka dokumen tersebut dinyatakan masuk menjadi anggota *cluster* tersebut.

Sebuah dokumen teks direpresentasikan sebagai kumpulan dari *term*. Setiap dokumen mempunyai sebuah vektor label berupa bobot *term* dan labelnya yang dinotasikan sebagai berikut: $D = \{(d_1, y_1), (d_2, y_2) \dots (d_n, y_n)\}$, dimana d_i menyatakan vektor bobot *term* dokumen ke *i*, *n* menyatakan jumlah dokumen dan *y* menyatakan label dari dokumen. Karena fokus dari studi ini berada pada konteks pengkategorian dokumen secara *multi*-label maka label dokumen *y* dalam hal ini berupa vektor berukuran $(1 \times p)$ dimana $p$menyatakan jumlah kategori. Adapun nilai dari vektor label tersebut bernilai 1 ketika dokumen tersebut berada pada kategori yang sesuai. Sebagai contoh, jika terdapat tiga kategori dokumen yaitu nasional, ekonomi dan pendidikan maka ketika d_1 memiliki label $y_1 = \{0,1,1\}$, maka dokumen d_1 merupakan dokumen yang memuat artikel berkenaan dengan ekonomi dan pendidikan.

Tabel 1. *Pseudocode* Fuzzy C-Means

No	Langkah
1	Inisialisasi matriks membership $\boldsymbol{U} = [\boldsymbol{u_{ij}}]$, $\boldsymbol{U^0}$ secara acak, k=0
2	Pada iterasi k, hitung pusat cluster $\boldsymbol{c_i}$ dengan (2) menggunakan matriks membership $\boldsymbol{U^k}$.
3	Hitung nilai $\boldsymbol{U^{k+1}}$ dengan rumusan (3)
4	Jika kondisi $\left\|\boldsymbol{U^{k+1}} - \boldsymbol{U^k}\right\| < \boldsymbol{\xi}$ maka berhenti, dan jika tidak memenuhi kondisi tersebut maka ulangi langkah 2.

Tabel 2. *Pseudocode Search Set*

No	Langkah
1	for each doc$\boldsymbol{di}, \boldsymbol{1 \le i \le n}$
2	for each cluster $\boldsymbol{Sv}$, $\boldsymbol{1 \le i \le p}$
3	if($\boldsymbol{di} \in \boldsymbol{Sv}$)
4	then $\boldsymbol{Gu} = \boldsymbol{Gu} \cup \boldsymbol{Sv}$

Data masukan untuk proses pengelompokkan n buah dokumen adalah sebuah matriksX berukuran $n \times t_R$, dimana t_R adalah jumlah keseluruhan *term* pada koleksi dokumen. Sehingga X_{it} merupakan representasi bobot *term* ke-$t(t = 1,2, \dots, t_R)$ pada dokumen ke-i $(i = 1,2, \dots, n)$. Adapun *pseudocode* dari algoritma ini seperti yang tertera pada Tabel 1**.**

Untuk mengelompokkan dokumen teks dengan *fuzzy c-means*, parameter yang dibutuhkan adalah. langkah pertama yang perlu dilakukan adalah inisialisasi nilai matriks keanggotaan $u_{i,j}$ setiap dokumen d_i pada setiap *cluster* yang ada. Proses inisialisasi ini dilakukan secara acak. Langkah selanjutnya adalah menghitung pusat cluster c_j sesuai dengan rumusan berikut:

$$c_j = \frac{\sum_{i=1}^{n} u_{i,j}^{m} \times x_i}{\sum_{i=1}^{n} u_{i,j}^{m}}, \quad (3)$$

dengan nilai m>1 yang disebut sebagai *fuzzifier* atau bobot fuzzy. Sedangkan j adalah indeks kategori $(j = 1,2,3..p)$, p menyatakan jumlah kategori label.

Selanjutnya, dilakukan perhitungan untuk mencari nilai membership (u_{ij}) dokumen d_i terhadap cluster c_j, yang baru dengan rumusan berikut:

$$u_{ij} = \frac{1}{\sum_{k=1}^{p} \left(\frac{d_{ij}}{d_{ik}}\right)^{\frac{2}{m-1}}}, \quad (4)$$

dimana, d_{ij}menyatakan jarak antara dokumen d_i pada pusat cluster c_j.

Setelah itu, dilakukan perhitungan pusat cluster cj yang baru, kemudian nilai membership uij yang baru hingga kondisi $\|U^{k+1} - U^k\| < \xi$terpenuhi. ξadalah kondisi kriteria stop yang merupakan bilangan bernilai sangat kecil mendekati nilai 0.

Setelah dilakukan pengelompokan dokumen proses selanjutnya adalah menghitung *priorprobability* dan *likelihood* menggunakan aturan *Bayes*[11]. Perhitungan *prior probability* dilakukan untuk menghitung peluang suatu kejadian diyakini benar. Dalam konteks ini digunakan untuk menghitung kemungkinan suatu dokumen merupakan anggota dari kategori tertentu. Perhitungan ini didasarkan pada rumusan (5) dan (6).

$$P(H_j = 1) = \frac{s+\sum_{i=1}^{l} y_{ji}}{2s+l} \quad (5)$$

$$P(H_j = 0) = 1 - P(H_j = 1), \quad (6)$$

dimana *s* adalah konstanta bernilai positif, biasanya bilangan positif yang bernilai kecil (misalnya 0.1).

3.2 Pengukuran Jarak

Metode yang efektif digunakan untuk mengukur nilai kemiripanantara dua buah dokumenteks adalah dengan menghitung tingkat kemiripanantara kedua buah dokumen tersebut dengan *cosine similarity*. Semakin besarnilai cosine similarity antara dua buah dokumen, maka semakin tinggi nilai kemiripan antara kedua dokumen tersebut.

Sebuah dokumen teks dapat direpresentasikan sebagai suatu kumpulan *term* dengan ruang berdimensi t_R. Kemiripanantara dua buah dokumen $d_1$dan d_2 dapat didefinisikan sebagai

$$cosine(d_1, d_2) = \frac{d_1 . d_2}{\|d_1\|\|d_2\|}, \quad (7)$$

dimana $d_1.d_2$adalah hasil*product* dari d_1 dan d_2 dihitung dengan $\sum_{k=1}^{n} d_{1k} d_{2k}$, $\|d_1\|$merupakan panjang vektor dari d_1 yang dihitung dengan $\sqrt{\sum_k^n x_k^2}$.Karena yang dibutuhkan dalam *fuzzy c-means* adalah *distance measure* untuk mengukur jarak antara suatu dokumen terhadap pusat *cluster* maka dari itu digunakan rumusan berikut[13]:

$$dist(d_1, d_2) = 1 - cosine(d_1, d_2). \quad (8)$$

3.3 K-Nearest Neighbors dan MAP

Dari hasil pengelompokan dokumen, dilakukan proses klasifikasi menggunakan metode *k-nearest neighbor* untuk menentukan label dari setiap dokumen.

Metode *k-nearest neighbors* membutuhkan parameter *k* untuk memilih *k*-dokumen yang memiliki kemiripan paling dekat dengan dokumen asal[12]. Untuk menentukan *k*-tetangga terdekat digunakan informasi Search Set *G*. Pembentukan search set suatu dokumen mengikuti aturan Tabel 2.

Setelah menentukan *k*-dokumen terdekat. Langkah berikutnya menghitung *likelihood*.*Likelihood* digunakan untuk menghitung kemungkinan suatu dokumen merupakan anggota dari suatu ktegori dengan memperhitungkan informasi ketetanggan dokumen tersebut. *Likelihood* dapat dihitung dengan menggunakan formula sebagai berikut:

$$P(E = e|H_j = 1) = \frac{s+Z(e,j)}{(k+1)s+\sum_{v=0}^{k} Z(v,j)} \quad (9)$$

$$P(E = e|H_j = 0) = \frac{s+\tilde{Z}(e,j)}{(k+1)s+\sum_{v=0}^{k} \tilde{Z}(v,j)}, \quad (10)$$

dimana $e = 0,1,\dots,k$ dan$j = 1,2,\dots,p$, sedangkan $Z(e,j)$ dan $\tilde{Z}(e,j)$ dapat dihitung dengan menggunakan formula berikut:

$$Z(e,j) = \sum_{i=1}^{l} y_{ji}\, \delta_{ei}(j) \quad (11)$$

$$\tilde{Z}(e,j) = \sum_{i=1}^{l} \tilde{y}_{ji}\, \delta_{ei}(j) \quad (12)$$

$$\delta_{ei}(j) = \begin{cases} 1, if\ e = n_j^i \\ 0, if\ e \neq n_j^i \end{cases} \quad (13)$$

$$\tilde{y}_{ji} = 1 - y_{ji}, \quad (14)$$

n_j^i adalah label dokumen d_i pada kategori c_j.

Langkah terakhir adalah penentuan label. Penentuan label ini digunakan untuk menentukan suatu dokumen masuk dalam kategori mana. Untuk menetukan suatu dokumen ternasuk ke dalam kategori tertentu dapat dihitung mengggunakan formula berikut:

$$y_j^t = \begin{cases} 1, if\ A > B \\ 0, if\ B > A \\ R[0,1], otherwise \end{cases} \quad (15)$$

dimana,

$$A = P(Hj = 1)P\left(E = n_j^t \middle| Hj = 1\right) \quad (16)$$

$$B = P(Hj = 0)P\left(E = n_j^t \middle| Hj = 0\right) \quad (17)$$

Jika nilai y_j^t bernilai 1 mengindikasikan bahwa dokumen j masuk pada kategori t. Sebaliknya, jika nilai y_j^tbernilai 0 maka dokumen j bukan termasuk anggota pada kategori t.

4. UJI COBA

Data yang digunakan untuk menguji sistem kategorisasi dokumen teks secara multi-label dengan menggunakan FCM-KNN didapatkan dari situs berita online*Kompas*[1]. Data yang digunakan ada dua jenis yaitu data *training* dan data *testing*. Data *training* digunakan untuk membangun model klasifikasi. Data *testing* digunakan untuk melihat performa sistem dari model klasifikasi yang telah didapatkan. Data *training* untuk uji coba ini berjumlah175 berita sedangkan untuk data testing

[1]http://www.kompas.com

Tabel 3. Nilai F1 dan BEP(%) dari Hasil Uji Coba

Metode	k=5		k=10		k=15		k=20		k=25	
	F1	BEP	F1	BEP	F1	BEP	F1	BEP	F1	BEP
FCM-KNN	69,7	69,9	73,39	75,22	38.02	41.67	54.5	56.6	57.4	57.6
MLKNN	-	-	68.22	70.04	67.81	69.78	67.25	69.45	66.94	69.20

digunakan sejumlah 50 berita. Untuk jumlah kategori yang digunakan sebanyak tiga macam yang terdiri atas kategori nasional, bisnis keuangan, dan pendidikan.

Sebelum melakukan klasifikasi, dokumen-dokummen tersebut perlu dilakukan pra-proses untuk menyaring kata-kata penting yang dapat merepresentasikan isi dokumen. Adapun proses dari tahapan pra-proses secara garis besar dapat dibagi menjadi lima operasi meliputi:

1. *Tokenizer*: proses penghilangan angka, tanda baca (*filtering*) dan konversi huruf kapital dan huruf kecil (*case folding*). Selanjutnya kata-kata tersebut disebut sebagai *term*.
2. *Stopwords Removal*: menghilangkan kata-kata yang dianggap tidak berkontribusi banyak untuk merepresentasikan konten dokumen. Setiap kata pada dokumen yang identik dengan kata yang terdapat didalam *stopword list* maka akan dihilangkan. Di dalam bahasa Indonesia banyak terdapat kata-kata yang sering muncul namun tidak merepresentasikan konten secara signifikan antara lain : “dan”, “dari”, “yang.”
3. *Stemming*: Mendapatkan bentuk dasar dari kata-kata yang tersisa pada dokumen teks untuk mendapatkan bentuk dasar dari kata-kata yang memiliki stem sama, misalnya kata ‘hubungan’, ‘menghubungkan’ dan ‘hubungi’ akan direduksi menjadi bentuk yang sama yaitu hubung.
4. Setiap*term* yang tersisa pada setiap dokumen diberi bobot sesuai dengan rumusan bobot TF-IDF.
5. Representasikan setiap dokumen berdasarkan kumpulan *terms* hasil proses dari *stemming* menggunakan nilai hasil pembobotannya.

Setelah pra-proses sudah dilakukan, maka dokumen pun siap untuk diklasifikasi. Uji coba dilakukan dengan menggunakan metode yang telah dijelaskan, yaitu FCM dan KNN. Sedangkan metode yang digunakan sebagai pembanding untuk klasifikasi *multi label* adalah MLKNN[6]. MLKNN adalah metode yang dapat secara langsung digunakan untuk klasifikasi multi label tanpa merubah tiap-tiap kategori menjadi nilai biner.

Untuk mengukur performa sistem ini digunakan dua jenis pengukuran yaitu F1 dan BEP.

$$F1 = \frac{2 \times MicroP \times MicroR}{MicroP + Micro\ R} \quad (18)$$

$$BEP = \frac{MicroP + MicroR}{2} \quad (19)$$

$$MicroP = \frac{\sum_{i=1}^{p} TPi}{\sum_{i=1}^{p} TPi + FPi} \quad (20)$$

$$MicroR = \frac{\sum_{i=1}^{p} TPi}{\sum_{i=1}^{p} TPi + FNi}, \quad (21)$$

dimana *p* adalah jumlah kategori. TP(*True Positive*) adalah jumlah dokumen *testing* yang diklasifikasikan secara benar. FN(*False Negative*) adalah banyaknya dokumen *testing* yang seharusnya menjadi anggota *c* tetapi tidak diklasifikasikan menjadi anggota *c*. FP (*False Positive*) adalah banyaknya dokumen *testing* yang seharusnya bukan anggota *c* tetapi dideteksi sebagai anggota *c*.

Tabel 3 menunjukkan perbandingan nilai F1 dan BEP diantara dua metode, yaitu FCM-KNN dan MLKNN. Pada uji coba kali ini digunakan nilai α = 0,33 dan nilai *k* yang berbeda-beda. Adapun parameter *k* yang diuji coba untuk *k-nearest neighbors* pada FCM-KNN dan MLKNN antara lain 5, 10, 15, 20 dan 25. Dapat dilihat bahwa nilai F1 dan BEP terbaik untuk metode FCM-KNN dan MLKNN terjadi pada k=10. Tabel 3juga menunjukkan metode FCM-KNN memperoleh nilai F1 sebesar 73,39% dan BEP sebesar 75,22%. Nilai ini lebih tinggi dibandingkan dengan MLKNN yang memperoleh nilai F1 sebesar 68,22% dan BEP sebesar 70,04%. Nilai F1 dan BEP yang didapatkan dapat berbeda-beda tergantung *dataset* yang digunakan.

5. KESIMPULAN

Kategorisasi dokumen teks secara *multi*-label adalah masalah yang berperan penting dalam sistem temu kembali informasi. Klasifikasi *multi* label berbeda dengan klasifikasi *single* label. Klasifikasi *single* label akan mengklasifikasikan suatu dokumen ke dalam satu kategori dokumen saja. Sedangkan klasifikasi *multi* label dapat mengelompokkan suatu dokumen ke lebih dari satu kategori.

Pada makalah ini diusulkan sebuah metode baru, yaitu FCM-KNN, untuk melakukan klasifikasi multi label artikel berbahasa Indonesia. Metode ini menggabungkan metode *fuzzy c-*

*means*dan *k-nearest neighbors*. Uji coba dilakukan dengan membandingkan FCM-KNN dengan metode pengelompokkan multi-label yang lain yaitu MLKNN.Hasil uji coba menunjukkan bahwa metode yang diusulkan memberikan performa lebih baik dibanding MLKNN.

6. DAFTAR PUSTAKA

[1] Aha, D. W. (1997). "Lazy learning: Special issue editorial". Artificial Intelligence Review, 11(1–5), 7–10.

[2] Joachim T. (1997). "A probabilistic analysis of the rocchio algorithm with tfidf for text categorization". In: International Conference on Machine Learning, (pp. 143–151).

[3] Fuhr, N., & Buckley, C. (1991). "A probabilistic learning approach for document indexing". ACM Transactions on Information Systems, 9(3), 223–248.

[4] Joachims, T. (1998). "Text categorization with support vector machines: Learning with many relevant features". In: European conference on machine learning, (pp. 137–142).

[5] Tsoumakas, G., & Katakis, I. (2007). "Multi-label classification: An overview". International Journal of Data Warehousing and Mining, 3(3), 1–13.

[6] Zhang, M. L., & Zhou, Z. H. (2006). "Multilabel neural networks with applications to functional genomics and text categorization".IEEE Transactions on Knowledge and Data Engineering, 18(10), 1338–1351.

[7] Zhang, M. L., & Zhou, Z. H. (2007). "ML-kNN: A lazy learning approach to multi-label learning". Pattern Recognition, 40(7), 2038–2048

[8] J.C. Bezdek, Pattern Recognition with Fuzzy Objective FunctionAlgorithms, Plenum, New York, 1981.

[9] Mahendra, I Putu Adhi Kerta. (2008). "Enhanced Confix Stripping Stemmer And Ants Algorithm For Classifying News Document In Indonesian Language".The 5th International Conference on Information & Communication Technology and SystemsISSN 2085-1944

[10] Salton G. dan C.Buckley. 1988. Term-Weighting Approaches in Automatic Text Retrieval. Departement of Computer Science, Cornell University.

[11] Alpaydin, E. (2004). Introduction to machine learning. The MIT Press

[12] Soucy, P. & Mineau, G. W. (2001). "A simple KNN algorithm for textcategorization. In Data Mining", 2001. ICDM 2001 Proceedings IEEEinternational conference on 29 Nov.–2 Dec. 2001 (pp. 64–68).

[13] Ichino, M., & Yaguchi, H. (1994). Generalized Minkowski metria formixed feature-ljpe data analysis. IEEE Transactıons On Systems, Man,and Cybernetıcs, 24(4).