

## PENGGABUNGAN KEPUTUSAN PADA KLASIFIKASI MULTI-LABEL

Agus Budi Raharjo<sup>1)</sup> dan Mohamed Quafafou<sup>2)</sup>

<sup>1)</sup>Jurusan Teknik Informatika, Fakultas Teknologi Informasi

Institut Teknologi Sepuluh Nopember, Surabaya

<sup>2)</sup>Jurusan Sains Komputer, Fakultas Sains,

Universitas Aix-Marseille

e-mail: [agus09@mhs.if.its.ac.id](mailto:agus09@mhs.if.its.ac.id)<sup>1)</sup>, [mohamed.quafafou@univ-amu.fr](mailto:mohamed.quafafou@univ-amu.fr)<sup>2)</sup>

### ABSTRAK

*Klasifikasi adalah bagian dari sistem pembelajar yang fokus pada pemahaman pola melalui representasi dan generalisasi data. Penentuan prediksi hasil klasifikasi terbaik menjadi masalah jika terdapat beberapa masukan dari metode yang berbeda-beda pada lingkungan data yang heterogen. Penggabungan keputusan dapat digunakan untuk menentukan rekomendasi keluaran beberapa metode klasifikasi. Kami memilih pendekatan voting dan meta-learning sebagai metode penggabungan keputusan. Ada dua fase yang dilakukan pada penelitian ini, yaitu fase pembangunan prediksi oleh metode klasifikasi yang heterogen dan fase penggabungan rekomendasi metode-metode tersebut menjadi satu kesimpulan jawaban. Karakteristik klasifikasi yang menjadi fokus adalah klasifikasi multi-label. Binary Relevance (BR), Classifier Chains (CC), Hierarchical of Multi-label Classifier (HOMER), dan Multi-label k Nearest Neighbors (MLkNN) adalah metode klasifikasi yang digunakan sebagai penyedia rekomendasi prediksi melalui pendekatan yang berbeda-beda. Pada fase penggabungan keputusan, metode Ignore diajukan sebagai pendekatan meta-learning. Ignore menggabungkan keputusan dengan cara mempelajari pola masukan dari sistem pembelajar. Untuk membandingkan kinerja Ignore, metode konsensus digunakan sebagai pendekatan voting. Hasil akhir menunjukkan bahwa Ignore memberikan hasil terbaik untuk parameter recall. Ignore memprediksi nilai false negative lebih sedikit dibandingkan dengan metode konsensus 0,5 dan 0,75. Hasil studi ini menunjukkan bahwa Ignore dapat digunakan sebagai meta-learning, meskipun kinerja Ignore harus diperbaiki agar dapat beradaptasi dengan data yang heterogen.*

**Kata Kunci :** klasifikasi multi-label, meta-learning, sistem Ignore.

### ABSTRACT

*Classification is the part of the learning system that focuses on understanding the pattern through the representation and generalization of data. How to determine the best prediction of classification could create a new problem if there are several inputs from the different methods with the heterogeneous data. Decision fusion can be used to define the recommendation output of several classification methods. We chose voting and meta-learning approaches as a decision fusion method. There are two phases proposed in this study, prediction construction phase using multi-strategy classification method and phase of decision fusion into a recommended conclusion. Classification characteristic used in this study is the multi-label classification. Binary Relevance (BR), Classifier Chains (CC), Hierarchical of Multi-label Classifier (HOMER), and Multi-label k Nearest Neighbors (MLkNN) are the selected classification methods used as the prediction provider through the different approaches. In decision fusion phase, Ignore method proposed as a meta-learning approach. Ignore combines the decisions by learning the patterns of input from the previous prediction systems. To compare the performance of Ignore, the consensus method is implemented as a comparator by adopting the voting approach. The final results show that Ignore brings the best performance in term of recall parameter. Ignore predicts the number of false negative less than consensus 0,5 and 0,75. This study shows that Ignore can be used as meta-learning, although its performance must be improved in order to adapt to the heterogeneous data.*

**Keywords:** Ignore system, meta-learning, multi-label classification.

## I. PENDAHULUAN

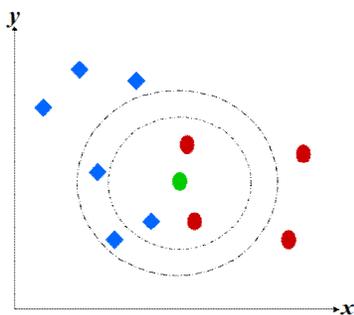
MESIN pembelajaran merupakan bagian dari bidang kecerdasan buatan yang fokus pada pola pemahaman. Ide utama dari mesin pembelajaran adalah membangun representasi data dan generalisasi. Representasi digunakan untuk evaluasi data pelatihan untuk mengetahui pola tertentu. Pola tertentu direpresentasikan ke dalam beberapa kelas. Generalisasi digunakan untuk memprediksi kelas dari data baru yang tidak terlihat. Salah satu metode yang digunakan dalam mesin pembelajaran adalah klasifikasi. Klasifikasi mengamati dari sekumpulan data dan mengklasifikasikan ke dalam beberapa kategori berdasarkan sifat data yang diketahui. Data masukan dari mesin pembelajaran terdiri dari kumpulan contoh. Masing-masing dari contoh terdiri dari kumpulan atribut dan label. Berdasarkan karakteristik label, metode klasifikasi dibagi menjadi empat bagian. Jika ada contoh yang hanya memiliki satu label dan hanya terdiri dari nilai ada dan tidak (probabilitas dua kelas), maka contoh ini akan diidentifikasi sebagai klasifikasi biner. Namun, jika contoh tersebut memiliki lebih dari dua probabilitas kelas, maka akan diidentifikasi sebagai klasifikasi multi-kelas. Klasifikasi multi-label merupakan salah satu dari bagian klasifikasi yang memiliki label lebih dari satu dan probabilitas kelas dari masing-masing label dikategorikan sebagai label biner. Jika probabilitas kelas adalah multi-kelas, maka akan diidentifikasi sebagai *multiple annotators* atau klasifikasi hierarki.

Ada dua permasalahan yang diajukan pada penelitian ini. Permasalahan pertama adalah bagaimana menciptakan penggabungan keputusan untuk data yang tidak terlihat jika terdapat beberapa prediksi dari metode klasifikasi. Untuk mendapatkan prediksi terbaik, kita dapat membandingkan parameter evaluasi dari masing-masing algoritma. Meskipun, kelemahan dari metode ini adalah tidak dapat mengambil keuntungan dari pengalaman masing-masing algoritma. Permasalahan kedua adalah menciptakan metode yang mudah menyesuaikan untuk karakteristik data yang heterogen karena tidak semua metode memiliki performa yang bagus dan handal untuk semua data. Pada permasalahan ini kami menganggap klasifikasi multi-label sebagai tujuan permasalahan. Permasalahan multi-label diajukan karena hal ini dapat mengatasi klasifikasi biner, konversi multi-kelas ke multi-label yang hanya memiliki satu label kardinal, dan *multiple annotators* dapat diubah ke dalam bentuk multi-label meskipun jumlah label baru merupakan perkalian dari jumlah kelas dengan jumlah label sebenarnya.

Berdasarkan permasalahan yang ada, kami mengajukan sistem umum yang dapat mengambil keuntungan dari masing-masing pengalaman algoritma dan membangun model baru berdasarkan beberapa pola model. Tujuan penelitian ini adalah untuk mengembangkan metode untuk penggabungan keputusan dengan *learning meta-classifiers* dan lebih khususnya menggunakan metode *Ignore* [1]. Pada penelitian ini, kami mengembangkan sistem ke dalam dua bagian. Bagian pertama adalah proses klasifikasi tradisional dan bagian kedua adalah proses klasifikasi tambahan. Pada proses pertama, kami menyiapkan beberapa sumber data, menjalankan beberapa metode klasifikasi, dan mengevaluasi hasilnya. Pada bagian kedua, kami mengajukan metode *Ignore* untuk menggabungkan hasil pertama. Kedua bagian akan dibandingkan untuk menemukan performa terbaik.

## II. DASAR TEORI

Studi dilakukan menggunakan sistem pembelajar sebagai objek eksplorasi dan metode penggabungan keputusan sebagai pendekatan solusi.



Gambar 1. Ilustrasi kNN

### A. Supervised Learning

*Supervised Learning* adalah salah satu metode sistem pembelajar yang fokus pada data yang telah memiliki label. Nilai kebenaran suatu data diberikan dan digunakan untuk proses klasifikasi. Ada dua jenis karakteristik label yang dikembangkan pada studi ini, yaitu label tunggal dan multi-label. Label tunggal digambarkan dengan jumlah label kebenaran data yang tunggal. Metode penyelesaian kasus ini dijadikan sebagai metode klasifikasi dasar untuk kasus multi-label. Metode klasifikasi yang digunakan pada untuk menyelesaikan masalah label tunggal adalah *Support Vector Machine* (SVM) dan *k Nearest Neighbors* (kNN) [2].

SVM menemukan pola dengan cara mengukur jarak maksimal antar titik pada bidang dua dimensi. Bidang dua dimensi tersebut terdiri dari banyak titik yang merepresentasikan posisi data latih. Pengaturan parameter dilakukan untuk memberikan hasil terbaik. Parameter yang dapat dimodifikasi yaitu parameter kernel, parameter penalti yang dinotasikan dengan  $C$ , dan jenis fungsi kernel. Parameter kernel yang dimodifikasi pada penelitian ini adalah kernel *Gaussian* yang dinotasikan dengan  $\gamma$ . SVM memiliki empat fungsi kernel berdasarkan klasifikasi data pada bidang vektor, yaitu penggunaan persamaan linier, persamaan polinomial, persamaan eksponensial yang direpresentasikan dengan *Radial Basis Function* (RBF), dan persamaan *hyperbolic tangent* (*sigmoid*).

Kelemahan metode SVM adalah besarnya waktu komputasinya. SVM termasuk ke dalam problem kuadrat dengan kompleksitas antara  $O(n.m^2)$  sampai dengan  $O(n.m^3)$  dimana  $n$  adalah jumlah atributnya dan  $m$  adalah jumlah data yang diolah [3]. *Average Rank* (AR) digunakan untuk mengidentifikasi parameter SVM terbaik dengan memperhatikan rata-rata hasil yang diberikan setiap kombinasi parameter. AR adalah metode pemeringkatan sederhana yang mengukur peringkat berdasarkan nilai rata-rata data yang memiliki hasil  $d$ -dimensi [4]. Dimisalkan  $r_j^i$  adalah peringkat metode SVM, dimana  $j$  adalah kombinasi parameter tertentu dan  $i$  adalah hasil perhitungan  $j$ . Peringkat terbaik didefinisikan dengan nilai  $r$  yang terkecil dengan interval nilai antara 1 sampai dengan  $d$ . Perhitungan AR ditulis sesuai (1).

$$bestRank = \min((\sum_i r_j^i) / n) \tag{1}$$

kNN adalah pengembangan metode *Nearest Neighbors* yang digunakan untuk kasus pengenalan pola. Cara kerja metode ini adalah menemukan tetangga terdekat suatu titik pada bidang vektor 2 dimensi hingga mendapatkan  $k$  tetangga. Gambar 1 menjelaskan cara kerja sederhana dari kNN. Dimisalkan ada dua kelas, yaitu kelas berwarna biru dan merah. Titik hijau adalah data baru yang belum diketahui kelasnya. Jika didefinisikan  $k=3$ , maka titik hijau akan dikenali sebagai kelompok warna merah karena  $|merah| = 2$  dan  $|biru|=1$ . Namun jika  $k=5$ , maka objek hijau akan diklasifikasikan ke dalam grup biru karena  $|merah| < |biru|$ . Besarnya nilai  $k$  akan mengurangi kesalahan prediksi, namun akan mengurangi tingkat perbedaan kedua grup. Dengan  $m$  data dan  $k$  tetangga, kompleksitas kNN didefinisikan dengan  $O(m.k)$  [5].

Kami menggunakan dua pendekatan multi-label pada tulisan ini, yaitu metode transformasi dan adaptasi algoritma [2]. Metode transformasi adalah metode yang mengubah bentuk multi-label ke dalam label tunggal dan mempelajari keterkaitan antar label tersebut. Metode adaptasi algoritma adalah modifikasi mesin pembelajar yang telah tersedia agar dapat digunakan pada kasus multi-label. Penelitian [2] telah melakukan perbandingan antar metode multi-label dan menunjukkan bahwa *Hierarchy of Multi-Label Classifier* (HOMER) memberi hasil evaluasi terbaik, diikuti dengan *Binary Relevance* (BR) dan *Classifier Chains* (CC). Multi Label kNN (ML kNN) dipilih sebagai representasi metode adaptasi algoritma.

BR adalah metode transformasi sederhana dengan cara membagi tiap label secara terpisah dan mengelompokkan sesuai metode klasifikasi dasar yang digunakan. Penghitungan prediksi label secara terpisah membuat BR tidak bisa merepresentasikan korelasi antar label. Kompleksitas BR adalah  $O(L.f(|X|,|D|))$ , dimana  $L$  adalah kumpulan label,  $D$  adalah data latih,  $X$  adalah atribut, and  $f(|X|,|D|)$  adalah kompleksitas metode klasifikasi dasar. Oleh karena itu, BR

TABEL I

CONFUSION MATRICES

		Label kebenaran	
		<i>true</i>	<i>false</i>
Label prediksi	<i>true</i>	TP	FP
	<i>false</i>	FN	TN

memiliki waktu transformasi yang ringan dan linier.

CC adalah metode pengembangan dari BR. CC menyempurnakan kekurangan BR dengan memperhatikan keterkaitan antar label. Waktu komputasi CC adalah  $O(|L|.f(|X| + |L|,|D|))$ . CC memiliki kompleksitas yang sama dengan BR jika  $|L| < |X|$  dan menjadi lebih lama dibandingkan BR jika  $|L| > |X|$  [6].

HOMER mengimplementasikan paradigma *divide and conquer*. Cara kerja HOMER adalah membagi sekumpulan label pada satu contoh menjadi kelompok yang lebih kecil untuk menciptakan distribusi kelompok label yang lebih seimbang. Kompleksitas HOMER adalah  $O(f(|L|)+|L|)$  dengan  $f(|L|)$  adalah kompleksitas dari algoritma *Balanced Clustering*. Algoritma *Balanced Clustering* yang digunakan adalah *Balance k Means* dengan kompleksitas  $O(|L_n||D_n| + |Ln|^2)$  while  $D_n$  adalah jumlah label dari contoh  $X_n$  [7].

ML-kNN adalah metode adaptasi algoritma dari *k-Nearest Neighbors* untuk kasus multi-label. Fase generalisasi data latih pada ML-kNN ditunda hingga adanya permintaan oleh sistem. Fase pertama metode ini adalah proses perhitungan *prior probabilities* dan *posterior probabilities*. Proses ini dihitung hanya dengan menggunakan data latih. kNN diimplementasikan pada tahap *posterior probabilities distribution*. Prediksi probabilitas dilakukan dengan memperhatikan aturan *Bayesian* [8].

*Ignore* adalah metode pendekatan pada klasifikasi dengan menggunakan aspek ketidaktahuan sebagai dasar prediksi [1]. *Ignore* digunakan pada kasus *multiple annotators*, dimana ada sekumpulan praktisi melakukan prediksi dan *Ignore* bertugas untuk mencari kesimpulan prediksi para praktisi. *Ignore* menggunakan metode *Bayesian* untuk mempelajari data baru. Jika tingkat prediksi data di atas ambang batas, maka *Ignore* akan memberikan prediksi sesuai prediksi praktisi. Namun jika di bawah ambang batas, *Ignore* akan mengubah label prediksi menjadi "?" dan menunda prediksinya. Kompleksitas metode ini sama dengan algoritma *Bayesian* dalam lingkup mesin pembelajar. Cara kerja *Ignore* direpresentasikan oleh (2).

$$h_i^t = \begin{cases} 1 & \text{if } y_i^t = "?" \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

### B. Penggabungan Keputusan

Penggabungan keputusan adalah sekumpulan proses integrasi dari beberapa metode untuk menghasilkan kesimpulan yang memiliki standar sama. Metode penggabungan keputusan ada dua, yaitu metode suara terbanyak dan *meta-learning*. Metode suara terbanyak dilakukan dengan pengambilan suara mayoritas sebagai keputusan akhir. Metode ini mengasumsikan semua suara memiliki bobot yang sama. Dimisalkan ada sekumpulan pilihan  $S$  dimana  $S = \{s_1, s_2, \dots, s_k\}$  dan satu grup pemilih  $V$  dimana  $V = \{v_1, v_2, v_3, \dots, v_m\}$ . Keputusan terbaik adalah kandidat solusi yang dipilih paling banyak atau memenuhi ambang batas dengan kondisi  $\text{vote}(s_i) \leq |V|$ , dimana  $\text{vote}(s)$  adalah fungsi untuk memberikan suara. Dalam beberapa kasus, keputusan harus memenuhi  $\text{vote}(s_i) \geq (m/2) + 1$ .

*Meta-learning* adalah penggabungan keputusan dengan mempelajari data-data yang telah diberikan sebelumnya. Metode ini menerapkan pendekatan mesin pembelajar untuk memberi keputusan. Kelebihan sistem ini adalah

fleksibilitasnya terhadap data uji baru tanpa mengulangi proses prediksi oleh sistem sebelumnya. Informasi yang diambil untuk dipertimbangkan meliputi hasil prediksi sistem sebelumnya, karakteristik masalah, dan pola sebelumnya [9].

*C. Metode Evaluasi*

Pengukuran evaluasi disesuaikan dengan prediksi berbasis data uji. Tabel I. menunjukkan kondisi berdasarkan perbandingan antara hasil prediksi dengan tabel kebenaran. Dimisalkan ada fungsi prediksi  $H(x)$  dan tabel kebenaran  $Y$ , kemungkinan dari domain hasil tersebut adalah *true positive* (TP), *true negative* (TN), *false positive* (FP), atau *false negative* (FN). TP adalah kondisi dimana  $H(x_i)=Y_i$  dan  $Y_i = true$ . Ketika  $H(x_i)=Y_i$  dan  $Y_i = false$ , hal ini disebut TN. FP adalah kondisi ketika  $H(x_i)=true$  dan  $Y_i = false$ . FN adalah kondisi dimana  $H(x_i)=false$  dan  $Y_i = true$ .

Ada enam parameter yang digunakan untuk mengukur performa, seperti *hamming loss*, *accuracy*, *precision*, *recall*, *F-measure*, dan *subset accuracy* [10]. Semua nilai parameter direpresentasikan ke dalam bilangan asli non-negatif dengan nilai interval antara 0 dan 1. *Hamming loss* merupakan tingkat kesalahan antara label prediksi dengan label kebenaran. Nilai *hamming loss* yang lebih kecil menunjukkan performa yang lebih baik. Evaluasi ini direpresentasikan oleh (3).

$$HLoss = \frac{1}{m} \sum_{i=1}^m \frac{H(X_i)\Delta Y_i}{|L|} \tag{3}$$

*Accuracy* adalah tingkat prediksi yang paling dekat dengan label sebenarnya. Nilai yang lebih tinggi dari *accuracy* menunjukkan performa yang lebih baik. Nilai kebenaran parameter ini dihitung dari diagram irisan prediksi dengan label kebenaran dibagi dengan diagram gabungan prediksi dengan label kebenaran dan jumlah data uji  $m$ . Evaluasi ini direpresentasikan oleh (4).

$$Accuracy = \frac{1}{m} \sum_{i=1}^m \frac{|H(X_i) \cap Y_i|}{|H(X_i) \cup Y_i|} \tag{4}$$

*Precision* adalah jumlah prediksi yang relevan dengan label kebenaran dibandingkan dengan semua label bernilai 1 pada hasil prediksi. Performa yang bagus ditunjukkan dengan jumlah label relevan yang lebih tinggi atau nilai FP yang lebih rendah. Nilai parameter ini dihitung dari diagram irisan hasil prediksi dengan label kebenaran bernilai 1 dibagi dengan jumlah prediksi bernilai 1 dan jumlah data uji  $m$ . Evaluasi ini ditunjukkan dalam (5).

$$Precision = \frac{1}{m} \sum_{i=1}^m \frac{|H(X_i) \cap Y_i|}{|H(X_i)|} \tag{5}$$

*Recall* merupakan jumlah prediksi yang relevan dengan label kebenaran dibandingkan dengan semua label bernilai 1 pada label kebenaran. Performa terbaik ditunjukkan dengan jumlah label yang relevan lebih tinggi atau nilai FN yang lebih rendah. Nilai parameter ini dihitung dari diagram irisan hasil prediksi dengan label kebenaran bernilai 1 dibagi dengan jumlah label bernilai 1 dan jumlah data uji  $m$ . Evaluasi ini didefinisikan dalam (6).

$$Recall = \frac{1}{m} \sum_{i=1}^m \frac{|H(X_i) \cap Y_i|}{|Y_i|} \tag{6}$$

*F-measure* merupakan rata-rata yang selaras antara *precision* dan *recall*. Performa yang bagus ditunjukkan oleh nilai rata-rata selaras yang lebih tinggi. Nilai parameter ini dihitung dari dua kali nilai irisan hasil prediksi dengan label kebenaran bernilai 1 dibagi dengan jumlah total prediksi dan label kebenaran yang bernilai 1 dan jumlah data uji  $m$ . Evaluasi ini dirumuskan ke dalam (7).

$$F - measure = \frac{1}{m} \sum_{i=1}^m \frac{2 |H(X_i) \cap Y_i|}{|H(X_i) + Y_i|} \quad (7)$$

*Subset accuracy* adalah pengukuran yang ketat dari kumpulan label yang diprediksi dengan kumpulan label kebenaran dari masing-masing contoh. Performa yang lebih baik ditunjukkan dengan nilai dari prediksi benar yang lebih tinggi.  $I(H(X)=Y)$  merupakan implementasi tabel kebenaran XOR dimana  $I(true)=1$  dan  $I(false)=0$ . Nilai parameter ini didapatkan dari jumlah total fungsi  $I$  dibagi jumlah data uji  $m$ . Evaluasi ini didefinisikan ke dalam (8).

$$SubsetAccuracy = \frac{1}{m} \sum_{i=1}^m I(H(X_i) = Y_i) \quad (8)$$

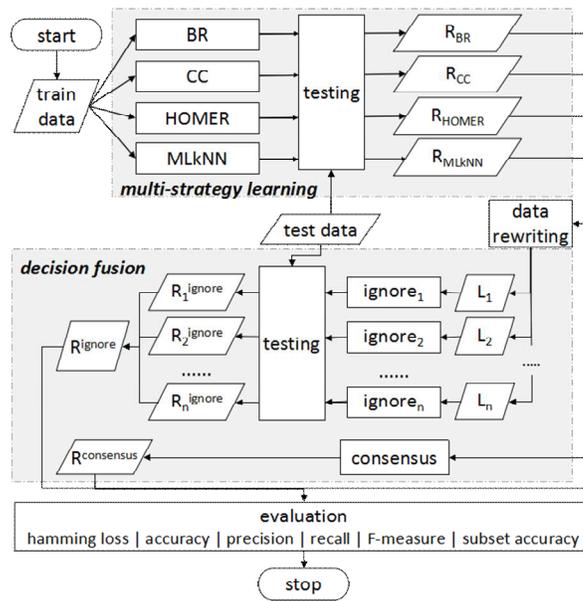
*Canberra distance* merupakan jenis pengukuran jarak pada ruang vektor multi-dimensi. Metode ini sering digunakan sebagai matriks untuk membandingkan peringkat [11]. Keuntungan dari *Canberra distance* dalam kasus pemeringkatan adalah mampu untuk menentukan sensitivitas antar vektor. Manfaat ini berguna untuk memahami korelasi antara dua atau lebih sistem sejenis yang diukur dari nilai idealnya. Misalkan ada dua titik vektor  $A$  dan  $B$  dimana  $A = a_1, a_2, a_3, \dots, a_d$ ,  $B = b_1, b_2, b_3, \dots, b_d$ , dan  $d$  adalah jumlah dimensi. Masing-masing bagian harus memiliki interval dan nilai non-negatif yang sama. Hal ini biasanya ditulis dalam nilai rasional antara 0 dan 1. Rumus dari metode ini dideskripsikan ke dalam (9).

$$d(A, B) = \sum_{i=1}^n \frac{|a_i - b_i|}{|a_i| + |b_i|} \quad (9)$$

Pada pengukuran evaluasi, *Canberra distance* digunakan untuk membuat peringkat baru berdasarkan parameter yang diajukan. *Canberra* mengukur jarak dari vektor 6-dimensi dari titik vektor ideal ke titik yang diprediksi. Titik vektor ideal terdiri dari  $min(hamming\ loss)$ ,  $max(accuracy)$ ,  $max(precision)$ ,  $max(recall)$ ,  $max(F-measure)$ , dan  $max(subset\ accuracy)$ . Hal ini dapat direpresentasikan dengan  $V_{ideal} = \{0, 1, 1, 1, 1, 1\}$ .

### III. ANALISIS DAN PERANCANGAN SISTEM

Terdapat dua tahap utama dalam penelitian ini. Tahap pertama adalah proses klasifikasi multi-label. Tahap kedua adalah pendekatan penggabungan keputusan dimana hasil pertama ditransformasi dan digunakan sebagai data masukan untuk sistem kedua. Data pra proses diimplementasikan pada tahap pertama dan kedua. Data pra proses pertama dilakukan dengan mengubah data ke dalam bentuk standar *Attribute-Relation File Format* (ARFF) [12]. Pada tahap kedua, data ditulis ulang dengan cara memisahkan masing-masing label. *Ignore* dipasang pada setiap label untuk membangun model baru dan menguji data yang tidak terlihat. Setiap model digabungkan kembali setelah *Ignore* memberikan keluaran berupa prediksi tiap label. Konsensus diimplementasikan untuk menunjukkan perbandingan



Gambar 2. Gagasan Penelitian

penggabungan keputusan. Teknik umum dari proses penelitian ini direpresentasikan pada Gambar 2.

#### A. Klasifikasi Multi-Label

Langkah pertama adalah normalisasi data. Proses ini dilakukan dengan mengubah data masukan ke dalam ARFF dan memisahkan data latih dan data uji menjadi 80% sebagai data latih/kalibrasi dan 20% sebagai data uji/validasi. Komposisi ini disesuaikan dengan masukan standar dari *Ignore*. Langkah selanjutnya adalah klasifikasi data. Pada langkah ini, model baru dibangun hanya dengan menggunakan data kalibrasi dan diuji dengan data validasi. Keluaran dari proses ini adalah prediksi kalibrasi, tingkat kepercayaan dari setiap label dalam bilangan rasional antara 0 sampai dengan 1, prediksi validasi, dan hasil evaluasi. Prediksi data dan tingkat kepercayaan digunakan sebagai masukan *Ignore*. Hasil evaluasi digunakan untuk membandingkan hasil akhir.

#### B. Pendekatan Ignore

Tahap kedua adalah *Ignore approach*. Proses ini terdiri dari metode transformasi, konstruksi *Ignore*, dan perbandingan hasil akhir menggunakan *Canberra distance*. Metode transformasi diimplementasikan dengan memisahkan satu set label  $L$  secara independen. Hal ini memungkinkan implementasi metode klasifikasi label tunggal untuk memproses data secara parsial. Metode ini memiliki waktu perhitungan linier untuk pendekatan multi-label. Namun, karena satu set label ditransformasikan ke dalam label-label independen, maka pendekatan ini tidak dapat memberikan korelasi antar label dalam satu contoh.

	Dataset	Domain	Data kasus			Atribut		Label			
			$ X $	Data latih	Data uji	Biner	Numerik	$ L $	$L_c$	$L_D$	<i>distinct</i>
1	Flags	Gambar	104	155	39	9	10	7	3,392	0,485	54
2	Emotions	Musik	593	474	119	0	72	6	1,869	0,311	27
3	Medical	Teks	978	782	196	1449	0	45	1,245	0,028	94
4	CAL500	Musik	502	402	100	0	68	174	26,044	0,15	502
5	Scene	Gambar	2407	1926	481	0	294	6	1,074	0,179	15
6	Yeast	Biologi	2417	1934	483	0	103	14	4,237	0,303	198

Proses selanjutnya adalah konstruksi model *Ignore*. Pada langkah ini, *Ignore* mempelajari pola dari setiap metode klasifikasi. Data input terdiri dari atribut, label kebenaran, dan prediksi kalibrasi. *Ignore* membentuk model baru berdasarkan hubungan antara atribut dengan prediksi yang telah dibuat oleh model-model sebelumnya. Model *ignore* diuji menggunakan data uji yang sama pada tahap pertama. Keluaran dari proses ini adalah hasil evaluasi, *confusion matrices*, dan prediksi baru. Proses selanjutnya adalah perbandingan evaluasi dari hasil tahap pertama dan kedua. Perbandingan ini dilakukan dengan mengukur jarak antara hasil ideal dengan hasil saat ini menggunakan *Canberra distance*.

### C. Pengaturan Sumber Data

Parameter data yang diukur meliputi komposisi data latih dan data uji mulai dari 194 sampai 2417 contoh, jumlah atribut mulai dari 19 sampai 1449, jumlah label mulai dari 6 sampai 174, nilai *Label Cardinality* ( $L_C$ ) mulai dari 1074 sampai 26044, nilai *Label Density* ( $L_D$ ) mulai dari 0,028 sampai 0,485, dan pola label yang berbeda mulai dari 15 sampai 502 pola. Domain dataset terdiri dari gambar, musik, teks, dan biologi. Semua informasi *meta-data* ditulis dalam Tabel II. Terdapat enam jenis data dari domain yang berbeda, yaitu *emotions* [13], *flags* [14], *medical* [15], *CAL500* [16], *scene* [17], dan *yeast* [18]. Data *flags* berasal dari beberapa representasi gambar dengan 19 atribut dan 7 label. Ini adalah data yang ringan karena jumlah contoh, atribut, dan labelnya yang sedikit. Di antara dataset lain untuk percobaan ini, dataset *flags* memiliki nilai  $L_D$ , yaitu 0,485. Dataset *emotions* adalah dataset yang berbasis musik. Dataset *medical* memiliki jumlah atribut terbesar di antara dataset lainnya, yaitu 1449 atribut dengan jumlah label 45. Meskipun demikian, atribut *medical* hanya bernilai biner. *CAL500* memiliki nilai  $L_C$  terbesar karena jumlah labelnya yang terbanyak, yaitu 174 label. Hal ini menjadikan *CAL500* memiliki pola label tertinggi di antara dataset lain. Karakteristik spesial dari *CAL500* adalah jumlah atribut yang kurang dari jumlah label. *Scene* adalah dataset berbasis gambar. Meskipun jumlah dari contohnya besar, tapi *scene* hanya memiliki 15 label berbeda dengan nilai  $L_D$  yang kecil. *Yeast* adalah data yang berbasis biologi. Data ini memiliki nilai terbesar dari contoh dengan 103 atribut dan 14 label.

TABEL III.

KOMBINASI PARAMETER  $\gamma$  DAN  $C$ .

Dataset	SVM	
	$\gamma$	$C$
Flags	0,001953125	2048
Emotions	3,05176E-05	32768
Medical	0,0078125	2048
CAL500	2	2048
Scene	8	8192
Yeast	2	2

### D. Pengaturan Teknis Eksperimen

Penelitian ini dilakukan dengan menggunakan pustaka mesin pembelajar *Mulan*. Pustaka *Mulan* digunakan sebagai sumber metode BR, CC, HOMER, dan MLkNN [19]. *Mulan* adalah pustaka multi-label yang dibangun di bawah kerangka kerja Weka. Kolaborasi LibSVM dan Weka diimplementasikan pada kode sumber SVM [18]. kNN adalah metode klasifikasi dasar yang tertanam secara otomatis pada MLkNN, sehingga tidak perlu dilakukan modifikasi atau penambahan pustaka baru. Weka dan *Mulan* adalah pustaka yang diimplementasikan pada bahasa Java. Metode *Ignore* diimplementasikan dalam bahasa R. Percobaan dilakukan pada server dengan spesifikasi *processor* Intel Core i5 2.50 GHz dengan RAM 4 GB dan sistem operasi Windows 7 Home Premium. Sumber utama kode pada studi ini dibangun

dalam bahasa Java dan R, sementara data pra proses dibangun pada bahasa C.

*E. Pengaturan Parameter*

Untuk mendapatkan kinerja terbaik, dilakukan modifikasi beberapa parameter pada metode klasifikasi label tunggal dan klasifikasi multi-label. Untuk HOMER, klaster  $Cl = \{2, 3, 4, 5, 6, 7, 8\}$  dipertimbangkan dalam rangka untuk mendapatkan ketepatan prediksi terbaik [7]. Pilihan parameter SVM yang dimodifikasi adalah  $\gamma$  dan  $C$ . Parameter  $\gamma$  dan  $C$  dipilih dari beberapa kombinasi dengan  $\gamma \in 2^{-15}, 2^{-13}, 2^{-11}, 2^{-9}, \dots, 2^1, 2^3$  dan  $C \in 2^{-5}, 2^{-3}, 2^{-1}, 2^1, \dots, 2^{13}, 2^{15}$  [3]. Selain itu, estimasi probabilitas diaktifkan agar tingkat kepercayaan direpresentasikan ke dalam bilangan rasional. Algoritma pencarian *Grid* diimplementasikan untuk menemukan hasil kombinasi  $\gamma$  dan  $C$  yang terbaik. Algoritma ini membandingkan kombinasi  $\gamma$  dan  $C$  menggunakan representasi jejaring berbentuk persegi panjang dimana  $\gamma$  adalah panjang dan  $C$  adalah lebarnya. Pendekatan ini merupakan metode pencarian keseluruhan yang lengkap untuk domain dua dimensi, tetapi memiliki kelemahan untuk kasus  $d$ -dimensi, dimana  $d > 2$ . Seluruh proses pencarian parameter SVM dibangun hanya menggunakan data latih dengan proses 10 *cross-validation*. AR digunakan untuk menentukan peringkat kombinasi parameter terbaik berdasarkan pada enam parameter evaluasi. Tabel III menunjukkan kombinasi hasil terbaik untuk setiap dataset. Pada *Ignore*, digunakan parameter standar dengan nilai ambang batas adalah 0,5. Kami menetapkan ambang batas konsensus sebesar 0,5 dan 0,75 sebagai pembanding perilaku pada metode penggabungan keputusan.

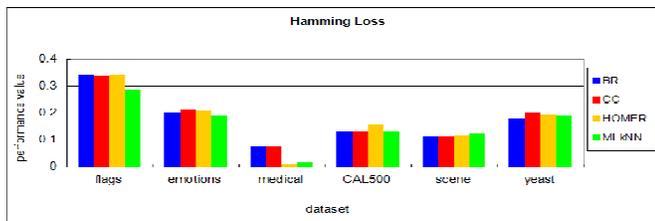
IV. HASIL DAN DISKUSI

Evaluasi dilakukan pada klasifikasi multi-label dan metode gabungan keputusan secara terpisah. Hasil akhir ditampilkan ke dalam diagram batang dengan sumbu  $X$  adalah urutan dataset dan sumbu  $Y$  adalah hasil pengukuran. Hasil pengukuran ditampilkan ke dalam interval antara 0 sampai dengan 1. Hasil terbaik pada *accuracy*, *subset accuracy*, *recall*, *precision*, dan *F-measure* ditunjukkan dengan nilai tertinggi, sedangkan hasil terbaik pada *hamming loss* ditunjukkan dengan nilai terendah. Pada *Canberra distance*, evaluasi ditampilkan ke dalam diagram batang dengan  $X$  adalah dataset dan  $Y$  adalah jarak sistem prediksi dengan nilai ideal. Sumbu  $Y$  ditampilkan ke dalam interval 1 sampai dengan  $k$ , dimana  $k$  adalah jumlah parameter evaluasi. Hasil terbaik ditunjukkan dengan jarak terendah.

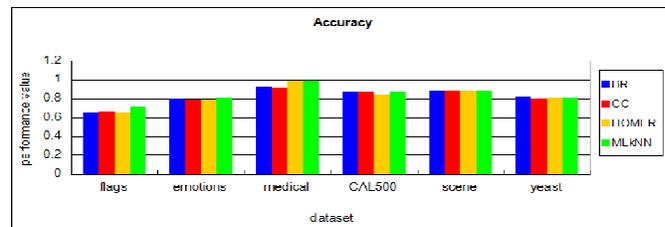
*A. Hasil pada Klasifikasi Multi-label*

Gambar 3 menjelaskan perbandingan antar metode klasifikasi. Berdasarkan diagram tersebut, HOMER menunjukkan hasil yang stabil pada semua data dan semua parameter. BR dan CC memberikan hasil yang mirip pada semua data dan parameter. Perbedaan signifikan antara BR dan CC dengan HOMER dan MLkNN terlihat pada dataset *medical*. Pada dataset tersebut, HOMER dan MLkNN memberikan rata-rata hasil evaluasi yang lebih baik.

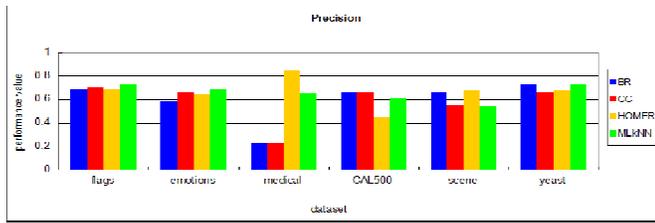
Perbandingan hasil akhir klasifikasi ditampilkan pada Gambar 3.g. MLkNN menampilkan hasil terbaik pada dataset *flags* dan *emotions*, sedangkan HOMER unggul pada dataset *medical*, *CAL500*, *scene*, dan *yeast*. BR dan CC memberikan hasil yang mirip. Berdasarkan perbandingan tersebut, metode transformasi memberikan hasil yang lebih baik dibandingkan dengan metode adaptasi algoritma. Simpangan baku terbesar ditunjukkan di dataset *medical*.



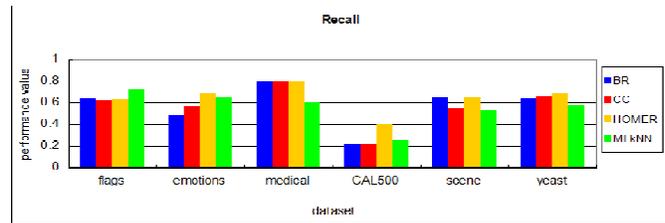
(a)



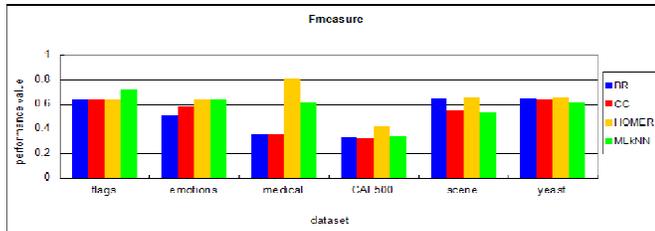
(b)



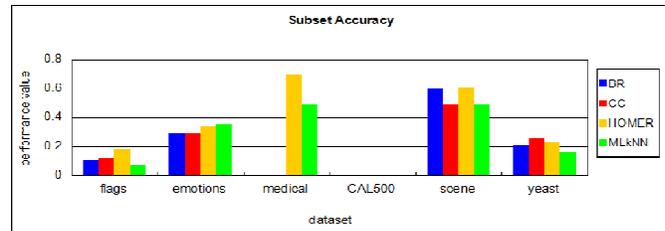
(c)



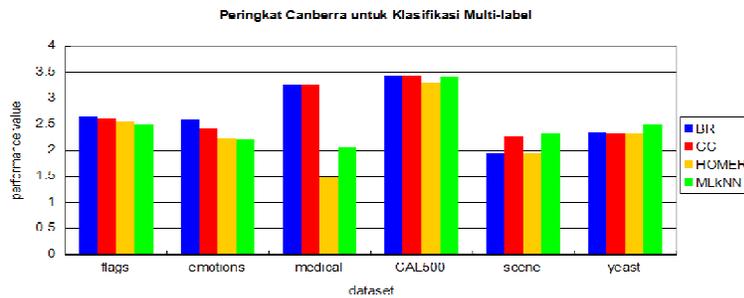
(d)



(e)



(f)

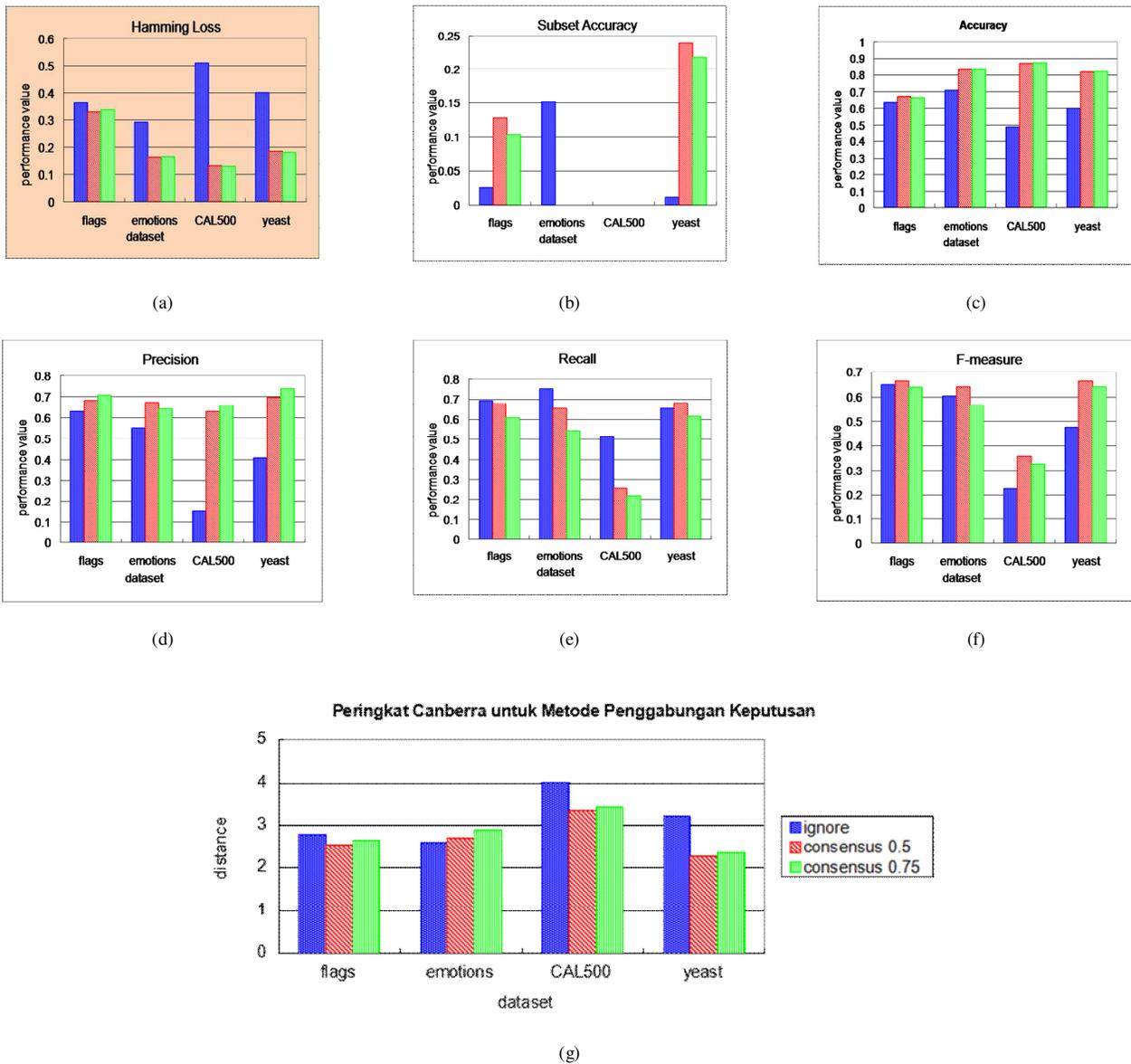


(g)

Gambar 3. Hasil perbandingan metode klasifikasi multi-label sesuai parameter evaluasi : (a) *hamming loss*, (b) *accuracy*, (c) *precision*, (d) *recall*, (e) *F-measure*, (f) *subset accuracy*, (g) peringkat berdasarkan *Canberra Distance*.

### B. Hasil pada Penggabungan Keputusan

Data yang diuji pada metode penggabungan keputusan terdiri dari *flags*, *emotions*, *CAL500* dan *yeast*. *Medical* dan *scene* tidak diujikan karena keluaran yang melebihi kemampuan hitung *Ignore*. Berdasarkan Gambar 4, *Ignore* dapat digunakan sebagai *meta-learning* meskipun memiliki hasil yang lebih rendah dibandingkan pendekatan konsensus. *Ignore* memberikan rata-rata hasil terbaik pada dataset *emotions* atau pada parameter evaluasi *recall*. Hasil *recall* yang bagus menunjukkan bahwa *Ignore* memberikan prediksi bernilai 1 lebih banyak dibandingkan label kebenaran. Hal ini menunjukkan bahwa ambang batas standar *Ignore* perlu dimodifikasi ke nilai yang lebih tinggi untuk memberikan prediksi bernilai 1 yang lebih ketat. Konsensus 0,5 menampilkan hasil yang lebih baik dibandingkan konsensus 0,75. Hal ini menunjukkan bahwa ambang batas 0,5 lebih mendekati hasil ideal dibandingkan 0,75. Meskipun demikian, hasil akhir yang diukur dengan *Canberra distance* pada Gambar 4.g menunjukkan kemiripan evaluasi antara Konsensus 0,5 dan 0,75. *Ignore* menampilkan simpangan yang besar untuk dataset *CAL500* dan *yeast*.



Gambar 4. Hasil perbandingan metode penggabungan keputusan sesuai parameter evaluasi : (a) *hamming loss*, (b) *subset accuracy*, (c) *accuracy*, (d) *precision*, (e) *recall*, (f) *F-measure*, (g) peringkat berdasarkan *Canberra Distance*.

## V. KESIMPULAN

Pada studi ini kami mengenalkan pendekatan metode penggabungan keputusan untuk lingkungan klasifikasi multi-label. Metode klasifikasi yang diimplementasikan adalah BR, CC, HOMER, dan MLkNN. Metode penggabungan keputusan yang dibandingkan adalah pendekatan *meta-learning* menggunakan *Ignore* dan pendekatan dasar suara terbanyak menggunakan konsensus dengan ambang batas 0,5 dan 0,75. Pengukuran hasil akhir dilakukan dengan membandingkan enam parameter evaluasi, yaitu *hamming loss*, *accuracy*, *precision*, *recall*, *subset accuracy*, dan *F-measure*.

Berdasarkan perbandingan metode klasifikasi multi-label, HOMER memberikan hasil terbaik dibandingkan MLkNN, BR, dan CC. MLkNN unggul pada jumlah data yang kecil dan sederhana. Meskipun CC merupakan pengembangan dari BR, namun hasil akhir menunjukkan bahwa BR memberikan hasil yang lebih baik. Hasil perbandingan metode penggabungan keputusan menunjukkan bahwa konsensus 0,5 memberikan hasil terbaik dibandingkan konsensus 0,75 dan *Ignore*. Namun konsensus 0,5 dan konsensus 0,75 harus melakukan proses ulang jika

terdapat data uji baru, sedangkan *Ignore* dapat mempelajari dan memprediksi secara mandiri. Selain itu, *Ignore* dapat mempelajari prediksi metode klasifikasi yang mengandung kesalahan. *Ignore* menunjukkan hasil terbaik hanya pada data *emotions* atau pada parameter *recall*. Selain parameter dan data tersebut, konsensus 0,5 dan 0,75 memberikan hasil yang lebih baik.

Pada studi ini kami berkonsentrasi pada penggunaan *Ignore* biner sebagai metode *meta-learning* baru dan tidak mengubah parameter standarnya. Pada penelitian selanjutnya kami akan memodifikasi proses internal *Ignore* dan mengusulkan *Ignore* multi-kelas untuk menyelesaikan masalah pengambilan keputusan. Metode klasifikasi multi-label saat ini memiliki hasil evaluasi yang memiliki simpangan baku <50% dari rentang interval pengukuran. Kami mengajukan metode baru yang dapat memberikan nilai evaluasi yang lebih heterogen. Selain itu, karakteristik *label balancing* perlu dipelajari untuk mengetahui perilaku mesin pembelajar terhadap lingkungan multi-label.

#### UCAPAN TERIMA KASIH

Penelitian ini terlaksana atas dukungan oleh Kementerian Pendidikan dan Kebudayaan Republik Indonesia, Campus France Indonesia, Bourses du Gouvernement Français, Pasca Sarjana ITS, Teknik Informatika ITS, dan Universitas Aix-Marseille.

#### DAFTAR PUSTAKA

- [1] C. Wolley dan M. Quafafou, "Learning from multiple annotators: when data is hard and annotators are unreliable," Data Mining Workshops (ICDMW), 2012 IEEE 12th International Conference on, hal. 514-521, IEEE, 2012.
- [2] G. Madjarov, D. Kocev, D. Gjorgjevikj, dan S. Džeroski, "An extensive experimental comparison of methods for multi-label learning," *Pattern Recognition*, vol. 45, no. 9, hal. 3084-3104, 2012.
- [3] C. W. Hsu, C. C. Chang, dan C. J. Lin, "A practical guide to support vector classification," 2010.
- [4] P.B. Brazdil dan C. Soares, "A comparison of ranking methods for classification algorithm selection," *Machine Learning: ECML 2000*, hal. 63-67, Springer, 2000.
- [5] D. Bremner, E. Demainer, J. Erickson, J. Lacono, S. Langern, P. Morin, dan G. Toussaint, "Output sensitive algorithms for computing nearest-neighbour decision boundaries," *Discrete & Computational Geometry*, vol. 33, no.4, hal. 593-604, 2005.
- [6] J. Read, B. Pfahringer, G. Holmes, dan E. Frank, "Classifier chains for multi-label classification," *Machine Learning*, vol. 85, no. 3, hal. 333-359, 2011.
- [7] G. Tsoumakas, I. Katakis, dan I. Vlavas. "Effective and efficient multilabel classification in domains with large number of labels," *Proc. ECML/PKDD 2008 Workshop on Mining Multidimensional Data (MMD 2008)*, hal. 30-44, 2008.
- [8] M. Zhang dan Z. Zhou, "MI-knn: A lazy learning approach to multi-label learning," *Pattern recognition*, vol. 40, no. 7, hal. 2038-2048, 2007.
- [9] R. Vilalta dan Y. Drissi, "A perspective view and survey of meta-learning," *Artificial Intelligence Review*, vol. 18, no. 2, hal. 77-95, 2002.
- [10] G. Tsoumakas, I. Katakis, dan I. Vlavas. "Mining multi-label data," *Data mining and knowledge discovery handbook*, hal. 667-685, Springer, 2010.
- [11] G. Jurman, S. Riccardonna, R. Visintainer, dan C. Furlanello, "Canberra distance on ranked lists," *Proceedings, Advances in Ranking NIPS 09 Workshop*, hal. 22-27, 2009.
- [12] I. H. Witten, M. A. Hall, dan E. Frank, "Data Mining," 2011.
- [13] K. Trohidis, G. Tsoumakas, G. Kalliris, dan I. P. Vlavas, "Multi-label classification of music into emotions," *ISMIR*, vol. 8, hal. 325-330, 2008.
- [14] E. C. Gonçalves, A. Plastino, dan A. A. Freitas, "A genetic algorithm for optimizing the label ordering in multi-label classifier chains," *Tools with Artificial Intelligence (ICTAI), 2013 IEEE 25<sup>th</sup> International Conference on*, hal. 469-476, IEEE, 2013.
- [15] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, dan I. H. Witten, "The weka data mining software: an update," *ACM SIGKDD explorations newsletter*, vol. 11, no. 1, hal. 10-18, 2009.
- [16] D. Turnbull, L. Barrington, D. Torres, dan G. Lanckriet, "Semantic annotation and retrieval of music and sound effects," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 16, no. 2, hal. 467-476, 2008.
- [17] M.R. Boutell, J.Luo, X.Shen, dan B.C. Matthew, "Learning multi-label scene classification," *Pattern Recognition*, vol. 37, no. 9, hal. 593-604, 2005.
- [18] J. P. Pestian, C. Brew, P. Matykievicz, DJ. Hovermle, N. Johnson, K. N. Cohen, dan W. Duch, "A shared task involving multi-label classification of clinical free text," *Proceedings of the Workshop on BioNLP 2007: Biological, Translational, and Clinical Language Processing*, hal. 97-104, Association for Computational Linguistics, 2007.
- [19] G. Tsoumakas, E. Spyromitros-Xioufis, J. Vilcek, dan I. Vlavas, "Mulan: A java library for multi-label learning," *The Journal of Machine Learning Research*, vol. 12, hal. 2411-2414, 2011.