Penanganan Fitur Kontinyu dengan *Feature Discretization* Berbasis *Expectation Maximization Clustering* untuk Klasifikasi *Spam Email* Menggunakan Algoritma ID3

Safuan, Romi Satria Wahono, dan Catur Supriyanto
Fakultas Ilmu Komputer, Universitas Dian Nuswantoro
soft_sfn@yahoo.com, romi@romisatriawahono.net,caturs@research.dinus.ac.id

Abstrak: Pemanfaatan jaringan internet saat ini berkembang begitu pesatnya, salah satunya adalah pengiriman surat elektronik atau *email.* Akhir-akhir ini ramai diperbincangkan adanya *spam email*. *Spam email* adalah email yang tidak diminta dan tidak diinginkan dari orang asing yang dikirim dalam jumlah besar ke *mailing list*, biasanya beberapa dengan sifat komersial. Adanya *spam* ini mengurangi produktivitas karyawan karena harus meluangkan waktu untuk menghapus pesan *spam*. Untuk mengatasi permasalahan tersebut dibutuhkan sebuah filter email yang akan mendeteksi keberadaan spam sehingga tidak dimunculkan pada *inbox mail.* Banyak peneliti yang mencoba untuk membuat filter email dengan berbagai macam metode, tetapi belum ada yang menghasilkan akurasi maksimal. Pada penelitian ini akan dilakukan klasifikasi dengan menggunakan algoritma *Decision Tree Iterative Dicotomizer 3* (ID3) karena ID3 merupakan algoritma yang paling banyak digunakan di pohon keputusan, terkenal dengan kecepatan tinggi dalam klasifikasi, kemampuan belajar yang kuat dan konstruksi mudah. Tetapi ID3 tidak dapat menangani fitur kontinyu sehingga proses klasifikasi tidak bisa dilakukan. Pada penelitian ini, *feature discretization* berbasis *Expectation Maximization* (EM) *Clustering* digunakan untuk merubah fitur kontinyu menjadi fitur diskrit, sehingga proses klasifikasi *spam email* bisa dilakukan. Hasil eksperimen menunjukkan ID3 dapat melakukan klasifikasi *spam email* dengan akurasi 91,96% jika menggunakan data training 90%. Terjadi peningkatan sebesar 28,05% dibandingkan dengan klasifikasi ID3 menggunakan *binning*.

Kata kunci: Klasifikasi, *Spam email*, ID3, *Feature Discretization, Expectation Maximization Clustering*

1 PENDAHULUAN

Pemanfaatan jaringan internet saat ini berkembang begitu pesatnya, salah satunya adalah pengiriman surat atau pesan. Jalur internet sudah menggantikan pengiriman surat konvensional menjadi surat elektronik atau *email.* Dengan menggunakan *email,* pengiriman pesan dapat dilakukan dengan cepat antar negara di seluruh dunia. Akhir-akhir ini ramai diperbincangkan adanya *spam email*. *Spam* adalah email yang tidak diminta dan tidak diinginkan dari orang asing yang dikirim dalam jumlah besar ke *mailing list*, biasanya dengan beberapa sifat komersial dan dikirim dalam jumlah besar (Saad, Darwish, & Faraj, 2012). Beberapa berpendapat bahwa definisi ini harus dibatasi untuk situasi di mana penerima memilih untuk menerima email ini misalnya mencari pekerjaan atau mahasiswa penelitian yang sedang melakukan penelitian.

Menurut sebuah laporan yang diterbitkan oleh McAfee Maret 2009 (Allias, Megat, Noor, & Ismail, 2014), biaya kehilangan produktivitas per hari untuk pengguna kira-kira sama dengan $0,50. Hitungan ini berdasarkan dari aktivitas pengguna yang menghabiskan 30 detik untuk menangani dua pesan *spam* setiap hari. Oleh karena itu, produktivitas per karyawan yang hilang per tahun karena *spam* kira-kira sama dengan $182,50.

Dengan adanya masalah *spam* tersebut, ada banyak literatur yang diusulkan untuk menyaring *spam email*. Saadat Nazirova *et al.* (Nazirova, 2011) membagi filter *spam* berdasarkan teknik filternya menjadi 2 kategori yaitu metode untuk mencegah penyebaran spam dan metode untuk mencegah penerimaan spam. Metode pencegahan penyebaran spam diantaranya adalah peraturan pemerintah yang membatasi distribusi spam, pengembangan protokol email menggunakan otentikasi pengirim dan pemblokiran server email yang mendistribusikan spam. Metode pencegahan penerimaan spam dibagi menjadi 2 kategori yaitu penyaringan dengan menggunakan pendekatan teori dan berdasarkan area filtrasi (dari sisi server dan pengguna).

Salah satu cara penyaringan *spam* adalah menggunakan pendekatan teori berbasis pembelajaran. Beberapa penelitian yang pernah dilakukan dengan menggunakan algoritma klasifikasi berbasis pembelajaran adalah *Naïve Bayes* (NB) (Çiltik & Güngör, 2008) (Marsono, El-Kharashi, & Gebali, 2008), *Support Vector Machine* (SVM) (Sculley & Wachman, 2007), *Artifial Neural Networking* (ANN) (Wu, 2009), *Logistic Regression* (LR) (Jorgensen, Zhou, & Inge, 2008), *K-Nearest Neigbor* (KNN) (Méndez, Glez-Peña, Fdez-Riverola, Díaz, & Corchado, 2009) dan *Decission Tree* (Sheu, 2009).

Klasifikasi adalah proses menemukan model (atau fungsi) yang menggambarkan dan membedakan kelas data atau konsep, dengan tujuan menggunakan model tersebut supaya mampu memprediksi kelas objek dimana label kelasnya tidak diketahui (Han & Kamber, 2006). Di bidang klasifikasi, ada banyak cabang yang berkembang yaitu pohon keputusan, klasifikasi Bayesian , jaringan saraf dan algoritma genetika (Tsai, Lee, & Yang, 2008). Di antara cabang tersebut, pohon keputusan telah menjadi alat yang popular untuk beberapa alasan: (a) dibandingkan dengan jaringan saraf atau pendekatan berbasis bayesian, pohon keputusan lebih mudah diinterpretasikan oleh manusia; (b) lebih efisien untuk data pelatihan yang besar dibanding dari jaringan saraf yang akan memerlukan banyak waktu pada ribuan iterasi; (c) algoritma pohon keputusan tidak memerlukan pengetahuan domain atau pengetahuan sebelumnya; dan, (d) akan menampilkan akurasi klasifikasi lebih baik dibandingkan dengan teknik lain.

Aman Kumar Sharma *et al.* (Sharma, 2011) membandingkan akurasi empat algoritma *Decision Tree* yaitu *Iterative Dichotomiser 3* (ID3), J48, *Simple Classification And Regression Tree* (CART) and *Alternating Decision Tree* (ADTree). CART menunjukkan hasil yang hampir sama dengan J48. ADTree dan ID3 menunjukkan akurasi kecil dibandingkan dengan CART dan J48. Hal ini menunjukkan bahwa algoritma J48 lebih disukai dibanding CART, ADTree dan ID3 dalam klasifikasi email spam yang mana ketepatan klasifikasi menjadi sangat penting.

Chakraborty *et al.*(Chakraborty & Mondal, 2012) melakukan penelitian dengan membandingkan dan

menganalisa tiga jenis teknikklasifikasi pohon keputusan yaitu *Naïve Bayes Tree* (NBT), C 4.5 (atau J48) dan *Logistik Model Tree* (LMT) untuk filtrasi spam. Hasil eksperimen menunjukkan bahwa LMT mempunyai akurasi sekitar 86% dan tingkat *false positif* jauh lebih rendah dari NBT dan J48. NBT membutuhkan waktu pelatihan tertinggi di antara semua klasifikasi pohon keputusan yang diteliti tapi memiliki *false positive rate* diantara J48 dan LMT. J48 memerlukan waktu pelatihan dan jumlah waktu berjalan paling sedikit di antara NBT dan LMT pada dataset yang sama.

Pada penelitian ini akan digunakan algoritma *decision tree* ID3 karena algoritma ini lebih baik dibanding algoritma *Decision Tree* yang lain seperti C4.5, CHAID dan CART (Sheu, 2009). Algoritma ID3 paling banyak digunakan di pohon keputusan (Jin & De-lin, 2009), terkenal dengan kecepatan tinggi dalam klasifikasi, kemampuan belajar yang kuat dan konstruksi mudah (Liu Yuxun & Xie Niuniu, 2010). Tetapi ID3 mempunyai kelemahan yaitu tidak dapat mengklasifikasikan fitur kontinyu dalam dataset (Jearanaitanakij, 2005) (Al-Ibrahim, 2011) sehingga proses klasifikasi tidak dapat dilakukan.

ID3 dirancang untuk menangani data pelatihan dengan nilai atribut diskrit dan simbolik (Al-Ibrahim, 2011). Untuk mengatasi masalah fitur kontinyu, *feature discretization* (FD) telah diusulkan dengan tujuan memperoleh representasi dari dataset yang lebih memadai untuk pembelajaran. Penggunaan teknik ini telah terbukti baik untuk meningkatkan akurasi klasifikasi dan menurunkan penggunaan memori (Ferreira & Figueiredo, 2012). FD berfungsi untuk merubah fitur kontinyu (real) menjadi fitur diskret (Dash, Paramguru, & Dash, 2011) (Madhu, Rajinikanth, & Govardhan, 2014), membagi nilai menjadi interval yang lebih kecil (Senthilkumar, Karthikeyan, Manjula, & Krishnamoorthy, 2012) (Al-Ibrahim, 2011) dan meningkatkan performa algoritma (Wijaya, 2008) sehingga lebih cocok digunakan untuk menangani masalah atribut / fitur kontinyu pada ID3.

Ferreira *et al.* (Ferreira & Figueiredo, 2014) (Ferreira & Figueiredo, 2012) membagi FD menjadi 2 kategori yaitu *supervised* dan *unsupervised.* Kategori *supervised* terdiri dari beberapa teknik misalnya *information entropy minimization* (IEM), ChiSquare, *bayesian belief networks* (BBN) dan *class-attribute interdependence maximization* (CAIM). Sedangkan kategori *unsupervised* terdiri dari *equal-interval binning* (EIB), *equal-frequency binning* (EFB) dan *proportional k-interval discretization* (PkID).

Gennady Agre *et al.* (Agre & Peev, 2002) dalam penelitiannya membandingkan 2 metode diskritisasi yaitu *entropy based discretization* Fayyad dan Irani *(supervised)* dengan *equal width binning* dan *equal frequency binning (unsupervised)* menggunakan 2 algoritma mesin pembelajaran *Simple Bayesian Classifier* (SBC) dan *Symbolic Nearest Mean Classifier* (SNMC). Hasil eksperimen menunjukkan bahwa dua metode diskritisasi *unsupervised* mempunyai performa lebih baik (terutama *equal frequency binning*) daripada metode *supervised entropy based discretization* yang diusulkan oleh Fayyad dan Irani.

Ankit Gupta *et al.*(Gupta, Mehrotra, & Mohan, 2010) membandingkan 2 metode diskritisasi berbasis *clustering* yaitu *Shared Nearest Neighbor* (SNN) dan K-Means yang digabung dengan *minimum entropy–maximum description length* (ME–MDL) menggunakan 3 algoritma klasifikasi *supervised* yaitu NB, SVM dan *Maximum Entropy*. Berdasarkan percobaan pada 11 dataset yang diamati, jika jumlah cluster yang diinginkan adalah sama dengan jumlah kelas atau jumlah kelas + 1, maka kinerja klasifikasi lebih baik dengan menggunakan ME-MDL. K-Means memberikan kinerja yang lebih baik dari SNN.

Penelitian yang dilakukan oleh Yong Gyu Junga et al (Jung, Kang, & Heo, 2014) membandingkan kinerja dari algorithma K-means and *Expectation Maximization* (EM). Dari percobaan yang telah dilakukan menunjukkan bahwa kecepatan pemrosesan K-means lebih lambat dibanding EM, tapi akurasi klasifikasi data adalah 94,7467% yang merupakan 7,3171% lebih baik dari yang didapat oleh EM. Tentu, ketidaktelitian dari K-means lebih rendah dibandingkan dengan yang ada pada algoritma EM. Secara keseluruhan, optimasi lebih lanjut harus diperkenalkan untuk mengurangi waktu.

Pada penelitian ini akan dilakukan proses klasifikasi *spam email* menggunakan algoritma ID3 dengan metode *feature discretization* berbasis EM *clustering*, karena EM dapat melakukan pengelompokan pada data yang mempunyai banyak rentang nilai yang berbeda secara signifikan (Ladysz, 2004) dan secara umum dapat diterapkan untuk fitur kontinyu dan kategori (I. Witten, 2011).

2 PENELITIAN TERKAIT

Salah satu masalah pada klasifikasi spam yaitu banyaknya atribut yang dihasilkan dari kata yang ada pada email. Banyak metode yang diusulkan untuk mengatasi masalah klasifikasi spam tersebut. Seperti penelitian yang dilakukan (Kumar, Poonkuzhali, & Sudhakar, 2012) yaitu dengan melakukan perbandingan pada beberapa algoritma data mining untuk klasifikasi spam. Algoritma klasifikasi yang dibandingkan adalah C4.5, C-PLS, C-RT, CS-CRT, CS-MC4, CS-SVC,ID3, K-NN, LDA, Log Reg TRILLS, Multi Layer Perceptron, Multilogical Logistic Regression, Naïve Bayes Continuous, PLS-DA, PLS-LDA, Random Tree dan SVM.Eksperimen dengan menggunakan fitur seleksi *fisher filtering*, *Relief filtering*, *Runs filtering* dan *Stepwise discriminant analysis.* Klasifikasi *Random Tree* dianggap sebagai pengklasifikasi terbaik, karena menghasilkan akurasi 99% melalui seleksi fitur *fisher filtering*.

Selain itu, penelitian (Chakraborty & Mondal, 2012) dilakukan dengan membandingkan dan menganalisa tiga jenis teknik klasifikasi pohon keputusan yang pada dasarnya pengklasifikasi data mining yaitu *Naïve Bayes Tree* (NBT), C 4.5 (atau J48) dan *Logistik Model Tree* (LMT) untuk filtrasi spam. Sebelum dataset diterapkan pada algoritma yang diuji, dilakukan *preprocessing* (pemrosesan awal) dengan menggunakan seleksi fitur. Hasil eksperimen menunjukkan bahwa LMT mempunyai akurasi sekitar 86% dan tingkat *false positif* jauh lebih rendah dari NBT dan J48.

Analisis komparatif dilakukan pada penelitian (Hamsapriya, T., 2012) dengan beberapa algoritma klasifikasi yaitu *Multilayer Perceptron* (MLP), J48 dan *Naïve Bayes* (NB). Hasil penelitian menunjukkan bahwa algoritma klasifikasi yang sama menghasilkan performa yang berbeda ketika dijalankan pada dataset yang sama tetapi menggunakan perangkat lunak yang berbeda. Selanjutnya teramati bahwa pada dataset ini untuk MLP menghasilkan tingkat kesalahan yang sangat baik dibandingkan dengan algoritma lain.

Penelitian yang dilakukan oleh (Gupta et al., 2010) dilakukan untuk mengatasi masalah diskritisasi dari variabel kontinyu untuk algoritma klasifikasi mesin pembelajaran. Teknik yang digunakan yaitu K-means *clustering* and *shared nearest neighbor* (SNN) *clustering* digabung dengan *minimum entropy–maximum description length* (ME–MDL) menggunakan 3 algoritma klasifikasi *supervised* yaitu NB, SVM dan *Maximum Entropy*. Hasil penelitian menunjukkan,

jika SVM digabungkan dengan SNN atau K-means pada jumlah cluster sama yang sama dengan jumlah kelas atau jumlah kelas + 1, hasilnya tidak lebih baik dari ME-MDL. Sulit untuk menilai algoritma *clustering* yang lebih baik karena di 7 dari 11 kasus K-means lebih baik daripada SNN.

Penelitian yang dilakukan oleh Chharia *et.al* untuk klasifikasi *spam email* dengan mengkombinasikan beberapa algoritma klasifikasi, menggunakan diversifikasi dengan mengatur fitur dan pengklasifikasi berbeda yaitu *Multinomial Naïve Bayes with Modified Absolute Discount Smoothing Method* (MNB-MAD), *Naïve Bayes* (NB), *Bagging, CART, C4.5, ADTree, Random Forest* (Rnd), *Functional Trees* (FT) *dan SimpleLogistis* (SL). Data yang digunakan adalah *SpamAssassin corpus* dan *Enron corpus*. Hasil penelitian menunjukkan, akurasi yang dicapai oleh metode ensamble yang diusulkan pada *Spamassasin corpus* sebesar 96,4% sedangkan pada pada *Enron corpus* sebesar 98,6%.

Penelitian yang dilakukan oleh Ali Al Ibrahim *et.al* dilakukan untuk mengatasi masalah fitur kontinyu untuk algoritma ID3. Teknik yang digunakan adalah *Continuous Inductive Learning Algorithm* (CILA), yaitu sebuah algoritma baru yang mengadopsi algoritma ID3 untuk mendiskrit fitur kontinyu yang dibuat oleh Ali Al Ibrahim. Hasil penelitian menunjukkan, CILA dapat secara otomatis memilih interval angka yang berbeda dengan teknik diskritisasi yang lain. Waktu yang dibutuhkan juga lebih singkat dibanding dengan metode diskrit *unsupervised* yang lain.

3 METODE YANG DIUSULKAN

Data yang digunakan pada penelitian ini bersumber pada database spam email yang bersumber dari UCI *repository of machine learning database. Spambase* terdiri dari terdiri dari total 4601 *e-mail*, dimana 1813 (39.4%) adalah *spam* dan 2788 (60.6%) adalah *non-spam*. Koleksi *spam email* berasal dari HP *email* dan *spam email* individu. Koleksi *non-spam email* berasal dari *email* kantor dan *email* perseorangan. Setiap *email* telah dianalisa dan terdapat 58 atribut (57 atribut input dan 1 atribut target atau kelas) yang menjelaskan tentang *spam email.* Rincian dari atribut tersebut adalah:

1. 48 atribut bertipe *continuous* dengan *range* 0-100 yang beranggotakan kata. Kata yang dimaksud antara lain:

Make	*address*	*all*	*3d*	*Our*	*Over*
Remove	*Internet*	*Order*	*mail*	*Receive*	*Will*
People	*Report*	*Addresses*	*Free*	*Business*	*Email*
You	*Credit*	*Your*	*Font*	*000*	*Money*
Hp	*Hpl*	*George*	*650*	*Lab*	*Labs*
telnet	*857*	*Data*	*415*	*85*	*Technology*
1999	*Parts*	*Pm*	*Direct*	*Cs*	*Meeting*
Original	*Project*	*Re*	*Edu*	*Table*	*Conference*

   Dengan prosentase:

$$\frac{\text{Jumlah kata yang muncul pada email}}{\text{Total kesuluruhan kata pada email}} \text{ x } 100\ \% \qquad (1)$$

2. 6 atribut bertipe *continuous* dengan *range* 0-100 yang beranggotakan karakter:
   ";" "(" "["
   "!" "$" "#"

   Dengan prosentase seperti pada persamaan (1).

3. 1 atribut *bertipe continous real* dengan nilai minimal 1, yang berisi rata-rata deret huruf kapital yang tidak bisa dipecahkan.
4. 1 atribut bertipe *continous real* dengan nilai minimal 1, yang berisi nilai terpanjang deret huruf kapital yang tidak bisa dipecahkan.
5. 1 atribut bertipe *continous real* dengan nilai minimal 1, yang berisi nilai jumlah deret huruf kapital yang tidak bisa dipecahkan.
6. 1 atribut bertipe *nomimal* dengan nilai 0 atau 1, yang berisi data target / kelas.

Metode klasifikasi yang diusulkan adalah menggunakan algoritma ID3 dengan diskrit fitur berbasis EM *clustering* untuk menangani fitur kontinyu pada dataset spam email. Evaluasi dilakukan dengan mengukur tingkat akurasi dan efisiensi.

Algoritma ID3 berusaha membangun *decision tree* (pohon keputusan) secara *top-down* (dari atas ke bawah) dengan mengevaluasi semua atribut yang ada menggunakan suatu ukuran statistik (yang banyak digunakan adalah *information gain*) untuk mengukur efektifitas suatu atribut dalam mengklasifikasikan kumpulan sampel data.

Untuk menghitung *information gain*, terlebih dahulu harus memahami suatu aturan lain yang disebut *entropy*. Di dalam bidang *Information Theory*, kita sering menggunakan *entropy* sebagai suatu parameter untuk mengukur heterogenitas (keberagaman) dari suatu kumpulan sampel data. Jika kumpulan sampel data semakin heterogen, maka nilai *entropy*-nya semakin besar. Secara matematis, *entropy* dirumuskan sebagai berikut:

$$Entropy = \sum_{i}^{c} -p_i \log_2 p_i \qquad (2)$$

dimana c adalah jumlah nilai yang ada pada atribut target (jumlah kelas klasifikasi). Sedangkan p_i menyatakan jumlah sampel untuk kelas *i*.

Setelah mendapatkan nilai entropy untuk suatu kumpulan sampel data, maka kita dapat mengukur efektifitas suatu atribut dalam mengklasifikasikan data. Ukuran efektifitas ini disebut sebagai *information gain*. Secara matematis, information gain dari suatu atribut A, dituliskan sebagai berikut:

$$Gain(S, A) = Entropy(S) - \sum_{v \in Value(A)} \frac{|S_v|}{|S|} Entropy(S_v) \qquad (3)$$

dimana:

A	: atribut
V	: menyatakan suatu nilai yang mungkin untuk atribut *A*
Values(A)	: himpunan nilai-nilai yang mungkin untuk atribut *A*
$\|Sv\|$	: jumlah sampel untuk nilai *v*
$\|S\|$	: jumlah seluruh data
Entropy (Sv)	: entropy untuk sampel-sampel yang memiliki nilai *v*.

Secara ringkas, langkah kerja Algoritma ID3 dapat digambarkan sebagai berikut:

1. Penghitungan IG dari setiap atribut
2. Pemilihan atribut yang memiliki nilai IG terbesar
3. Pembentukan simpul yang berisi atribut tersebut
4. Ulangi proses perhitungan *information gain* akan terus dilaksanakan sampai semua data telah termasuk dalam kelas yang sama. Atribut yang telah dipilih tidak diikutkan lagi dalam perhitungan nilai IG.

Expectation Maximization (EM) adalah algoritma perbaikan iteratif populer yang dapat digunakan untuk menemukan perkiraan parameter (Han & Kamber, 2006). EM merupakan salah satu metode untuk menemukan estimasi *maximum likelihood* dari sebuah dataset dengan distribusi tertentu. EM termasuk algoritma *partitional* yang berbasiskan model yang menggunakan perhitungan probabilitas, bukan jarak seperti umumnya algoritma *clustering* yang lainnya (Chharia & Gupta, 2013). Jika pada algoritma K-means, parameter utamanya adalah *centroid,* maka untuk EM parameter utamanya adalah q_{mk} dan α_k untuk mendapatkan nilai r_{nk} yaitu probabilitas dokumen n masuk ke klaster k atau probabilitas klaster k beranggotakan dokumen n.
Langkah-langkah algoritma EM adalah sebagai berikut:

1. *Guess Model Parameter*
   Proses ini adalah melakukan penebakan nilai probabilitas data terhadap sebuah klaster. Langkah *guess* pertama adalah *guess probability* data klaster sebagai *model parame*ter. Inisialisasi nilai probabilitas pada data kata dilakukan secara random/ acak. Untuk probabilitas klaster, totalnya harus selalu bernilai 1.

Tabel 1*Guess Model Parameter*

Y (klaster)	X1	X2	X3	X4	P(Y)
0	0,1	0,3	0,8	0,8	0,7
1	0,2	0,3	0,1	0,1	0,2
2	0,7	0,4	0,1	0,1	0,1

Dimana pada tahap ini akan ditebak nilai parameter q_{mk} dan α_k.

2. *Expectation Step*

$$r_{nk} = \frac{\alpha_k(\prod_{t_m \in d_n} q_{mk})(\prod_{t_m \ni d_n}(1-q_{mk}) \ )}{\sum_{k=1}^{K} \alpha_k(\prod_{t_m \in d_n} q_{mk})(\prod_{t_m \ni d_n}(1-q_{mk}) \ )} \quad (4)$$

dimana:
- r_{nk} adalah nilai probabilitas setiap dokumen n terhadap masing-masing *cluster* atau nilai probabilitas *cluster* k terhadap sebuah dokumen
- $\alpha_k(\prod_{t_m \in d_n} q_{mk})(\prod_{t_m \ni d_n}(1-q_{mk}) \ )$adalah probabilitas total term terhadap sebuah klaster
- $\sum_{k=1}^{K} \alpha_k(\prod_{t_m \in d_n} q_{mk})(\prod_{t_m \ni d_n}(1-q_{mk}) \ )$adalah nilai total probabilitas semua term terhadap semua klaster.

Setelah r_{nk}didapat, maka akan dihitung *Frequency Counts*

$$\sum_{n=1}^{N} r_{nk} \, \mathrm{I}(t_m \in d_n) \quad (5)$$

3. *Maximization Step*

$$q_{mk} = \frac{\sum_{n=1}^{N} r_{nk} \mathrm{I}(t_m \in d_n)}{\sum_{n=1}^{N} r_{nk}} \quad (6)$$

dimana:
- q_{mk} adalah nilai probabilitas term *m* terhadap sebuah klaster dimana term *m* tersebut merupakan anggota dari suatu dokumen *n*.
- $\sum_{n=1}^{N} r_{nk} \, \mathrm{I}(t_m \in d_n)$ adalah *frequency Counts*, probabilitas klaster k terhadap semua dokumen yang mempunyai term m sebagai anggotanya (nilai *term* m = 1).
- $\sum_{n=1}^{N} r_{nk}$ adalah probabilitas sebuah *cluster* k terhadap semua dokumen.

Kemudian dihitung probabilitas sebuah klaster k:

$$\alpha_k = \frac{\sum_{n=1}^{N} r_{nk}}{N} \quad (7)$$

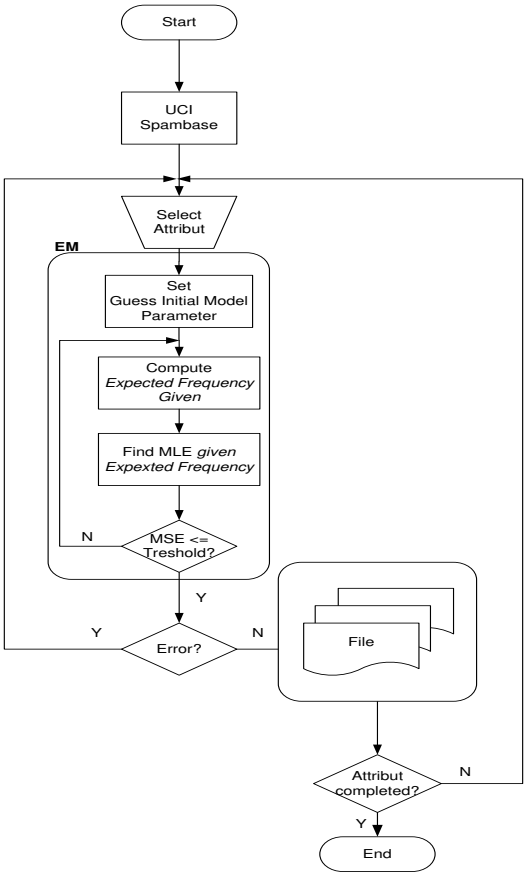
dimana *N* adalah probabilitas total klaster

4. Ulangi langkah 2 dan 3 sampai *Convergence*.
   Nilai probabilitas klaster data bersifat *Convergence* jika *update* probabilitas data terhadap klaster data tidak berubah-ubah lagi. Dengan kata lain nilai probabilitas dokumen terhadap sebuah klaster sudah bernilai 1.
   Langkah 1: Tentukan nilai *threshold*. Semakin kecil nilai *threshold* maka semakin dekat dengan *convergence*. Dalam hal ini nilai *threshold* nya adalah nol.
   Langkah 2: Hitung nilai *Means Square Error* dengan menggunakan rumus:

$$MSE(\vartheta) = E[(\vartheta - \theta)^2] \quad (8)$$

   Langkah 3: Bandingkan Nilai MSE dengan *threshold*
   Jika $MSE \leq$ *threshold* maka *convergence* dan iterasi berhenti.

Tahap *feature discretization* adalah memproses tiap fitur pada spambase dengan algoritma EM yang menghasilkan *output* sebuah file. Jumlah file yang terbentuk sesuai dengan jumlah fitur yang berhasil diproses oleh EM. Hasil proses ini digabung menjadi satu untuk diproses pada tahap klasifikasi. Proses *feature discretization* dapat dilihat pada Gambar 1.



Gambar 1 Proses *Feature Discretization*

Akurasi yang dihasilkan dihitung menggunakan *confusion matrix*. Perhitungan pada *confusion matrix* dihitung berdasarkan prediksi *positif* yang benar (*True Positif*), prediksi *positif* yang salah (*False Positif*), prediksi negatif yang benar (*True Negatif*) dan prediksi *negatif* yang salah (*False Negatif*).

$$Akurasi = \frac{TP + TN}{TP+FP+TN+FN} \quad (9)$$

Semakin tinggi nilai akurasinya, semakin baik pula metode yang dihasilkan

4 HASIL EKSPERIMEN

Eksperimen dilakukan dengan menggunakan komputer dengan spesifikasi processor Intel Celeron M560 2.13 GHz CPU, 2 GB RAM, dan sistem operasi Microsoft Windows 7 *Professional* 32-bit. *Software* yang digunakan adalah bahasa pemrograman PHP dan RapidMiner 5.2.

Tabel 2 Data UCI *Spambase*

word_freq_make	word_freq_address	word_freq_all	word_freq_3d	word_freq_our	word_freq_over	Class
0	0,64	0,64	0	0,32	0	1
0,21	0,28	0,5	0	0,14	0,28	1
0,06	0	0,71	0	1,23	0,19	1
0	0	0	0	0,63	0	1
0	0	0	0	0,63	0	1
0,3	0	0,3	0	0	0	0
0,96	0	0	0	0,32	0	0
0	0	0,65	0	0	0	0
0,31	0	0,62	0	0	0,31	0

Pada eksperimen ini, data yang digunakan adalah 4601 data email spambase dari UCI Machine *Learning repository* yang terdiri dari 57 atribut kontinyu dan 1 atribut nominal berisi data target/kelas. Nilai yang terdapat pada masing-masing fitur adalah prosentase munculnya kata dibandingkan dengan total keseluruhan data pada email. Sedangkan pada fitur kelas hanya ada 2 nilai yaitu 0 dan 1, 0 menunjukkan label *ham* (bukan *spam*) sedangkan label 1 menunjukkan label *spam* seperti terlihat pada Gambar1.

Pada tahap FDdata spambase diproses menggunakan algoritma EM *clustering* pada setiap fitur selain fitur kelas dan membentuk sebuah file. File yang dihasilkan kemudian digabungkan sehingga terbentuk sebuah file. Untuk lebih jelasnya dapt dilihat pada Tabel 3 dan Tabel 4.

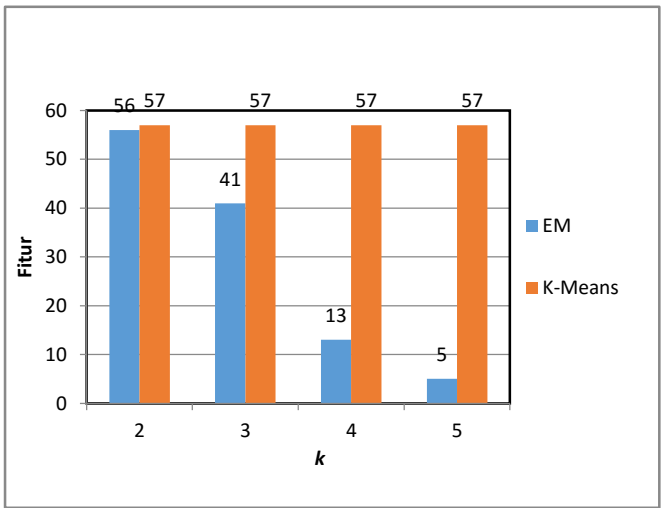
Tabel 3 Data Hasil FD

capital_run_length_average	cluster_0_probability	cluster_1_probability	cluster_2_probability	cluster_3_probability	cluster
3,8	,0	1,0	,0	,0	cluster_1
5,1	,0	,9	,0	,1	cluster_1
9,8	,0	,0	,0	1,0	cluster_3
3,5	,0	1,0	,0	,0	cluster_1
2,5	,7	,3	,0	,0	cluster_0
9,7	,0	,0	,0	1,0	cluster_3
1,7	,9	,1	,0	,0	cluster_0
4,7	,0	1,0	,0	,0	cluster_1

Tabel 4 Data Hasil Penggabungan

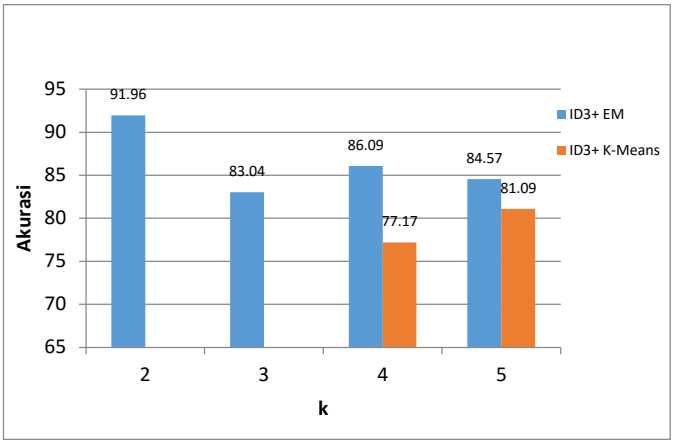
capital_ run_ length_ average	capital_ run_ length_ longest	capital_ run_ length_ total	Char _freq_ !	Class
cluster_1	cluster_1	cluster_2	cluster_2	Spam
cluster_1	cluster_1	cluster_2	cluster_0	Spam
cluster_3	cluster_2	cluster_3	cluster_0	Spam
cluster_1	cluster_1	cluster_1	cluster_0	Spam
cluster_1	cluster_1	cluster_1	cluster_0	Spam
cluster_1	cluster_0	cluster_0	cluster_1	Spam
cluster_0	cluster_0	cluster_1	cluster_0	Spam
cluster_0	cluster_0	cluster_0	cluster_1	Spam
cluster_0	cluster_0	cluster_0	cluster_0	Spam
cluster_0	cluster_1	cluster_1	cluster_0	Spam
cluster_0	cluster_0	cluster_0	cluster_1	Ham
cluster_0	cluster_0	cluster_0	cluster_2	Ham
cluster_0	cluster_0	cluster_0	cluster_1	Ham
cluster_0	cluster_0	cluster_0	cluster_1	Ham
cluster_0	cluster_0	cluster_1	cluster_1	Ham
cluster_0	cluster_0	cluster_0	cluster_0	Ham
cluster_0	cluster_0	cluster_1	cluster_1	Ham
cluster_0	cluster_0	cluster_1	cluster_1	Ham
cluster_0	cluster_0	cluster_1	cluster_0	Spam
cluster_0	cluster_0	cluster_0	cluster_1	Spam
cluster_0	cluster_0	cluster_0	cluster_0	Spam
cluster_0	cluster_1	cluster_1	cluster_0	Spam
cluster_0	cluster_0	cluster_1	cluster_1	Ham

Eksperimen pertama dilakukan untuk mengetahui seberapa besar pengaruh proses FD menggunakan EM terhadap klasifikasi *spam email* dibandingkan dengan algoritma K-Means yang merupakan metode *clustering* yang sering digunakan. Pada eksperimen dengan EM, jika dilakukan perubahan pada nilai *k* akan terjadi penurunan jumlah fitur. Semakin besar nilai *k* akan semakin kecil jumlah fitur diskrit yang dihasilkan. Hasil ekperimen dapat dilihat pada Gambar 2.



Gambar 2 Grafik Hubungan antara Nilai *k* dan Jumlah Fitur Diskrit yang Dihasilkan

Eksperimen selanjutnya dilakukan proses klasifikasi dengan algoritma ID3 menggunakan data training sebesar 70%. Dari Gambar 3 dapat diketahui bahwa akurasi klasifikasi ID3 menggunakan FD dan EM mampu mengungguli K-Means pada klasifikasi *spam email.*



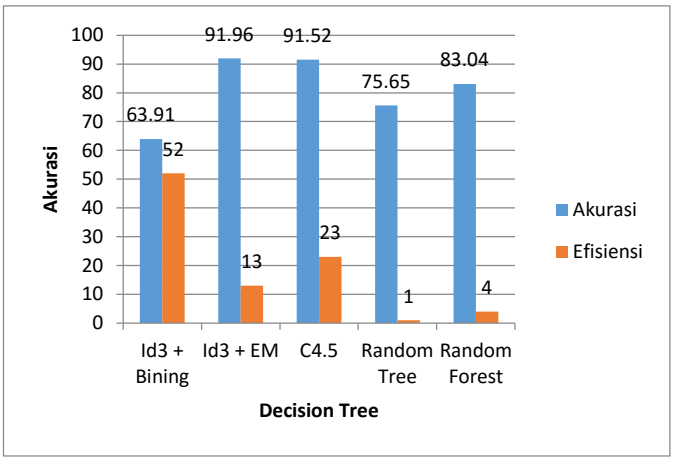
Gambar 3 Grafik Habungan antara Nilai Klaster *k* dengan Akurasi

Eksperimen tahap berikutnya yaitu dengan merubah rasio data yang digunakan pada proses training dengan menggunakan jumlah fitur sebanyak 56, dikarenakan pada eksperimen *feature discretization* akurasi tertinggi ID3+EM dicapai pada jumlah fitur tersebut. Rasio dirubah mulai dari 50%, 60%, 70%, 80% dan 90%. Hasil eksperimen menunjukkan ada kenaikan akurasi sesuai dengan peningkatan rasio data, seperti terlihat pada Tabel 5.

Tabel 5 Hasil Ekperimen ID3-EM dengan Merubah Rasio Data Training

Data training(%)	50	60	70	80	90
Efisiensi (sec)	16	12	12	13	13
Akurasi (%)	89,78	91,52	90,29	91,25	91,96

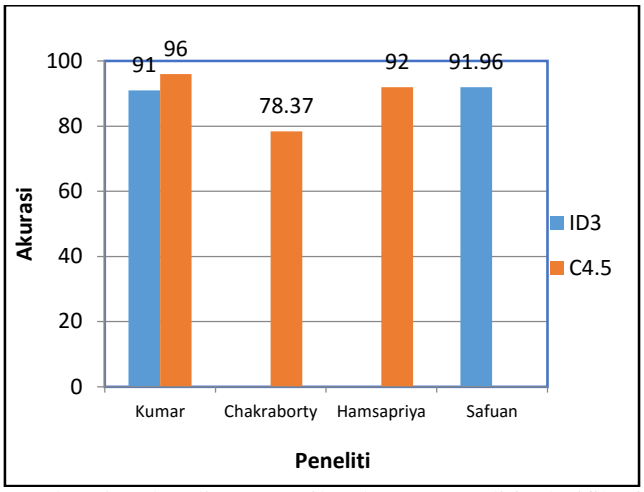
Eksperimen dilanjutkan dengan klasifikasi menggunakan algoritma *decision tree*(DT) yang lain yaitu algoritma C4.5, *Random Forest* dan *Random Tree*. Parameter yang digunakan adalah data training 90% karena pada ekperimen ini didapatkan akurasi ID3+EM yang maksimal. Hasil eksperimen dapat dilihat pada Tabel 4.15.



Gambar 4 Grafik Perbandingan ID3-EM dengan DT

Hasil eksperimen menunjukkan bahwa akurasi klasifikasi menggunakan algoritma ID3 dengan proses FD menggunakan EM memiliki akurasi yang lebih tinggi dan waktu proses yang lebih kecil dibandingkan dengan algoritma C4.5. Tetapi *random forest* dan *random tree* mempunyai waktu proses yang lebih cepat dibanding ID3.

Untuk membuktikan bahwa penelitian yang dilakukan mempunyai kontribuasi terhadap penelitian, maka dilakukan perbandingan dengan peneliti terdahulu yang sudah melakukan penelitian pada klasifikasi spam email, yaitu Kumar *et.al*, Chakraborty *et. al* dan Hamsapriya *et. al.* Perbandingan hasil penelitian dapat dilihat pada data grafik pada Gambar 5.



Gambar 5 Perbandingan Hasil Beberapa Peneliti Klasifikasi *Spam Email*

Dari grafik pada Gambar 5 dapat dilihat bahwa ID3 dari penelitian yang telah dilakukan memiliki akurasi yang lebih tinggi dibanding dengan ID3 pada penelitian Kumar *et.al*. ID3 juga mengungguli akurasi algoritma C4.5 pada penelitian yang dilakukan oleh Chakraborty *et.al*, hampir seimbang (selisih 0,04) dengan hasil penelitian Hamsapriya *et.al* tetapi masih lebih rendah dibanding akurasi C4.5 pada penelitian yang dilakukan oleh Kumar *et.al.* Hal ini dimungkinkan karena penggunaan fitur seleksi dan fitur reduksi pada penelitian Kumar *et.al* dapat menemukan fitur yang benar-benar berpengaruh pada klasifikasi. Sedangkan *feature discretization* pada penelitian ini hanya merubah fitur kontinyu menjadi diskrit saja tanpa memilih fitur yang berpengaruh pada klasifikasi. Ini diperlihatkan dengan berkurangnya akurasi walaupun fiturnya lebih sedikit seperti terlihat pada Gambar 3. Jadi pada penelitian ini jumlah fitur yang kecil tidak menambah akurasi klasifikasi seperti yang dihasilkan oleh fitur seleksi pada penelitian Kumar *et.al* sehingga perlu dilakukan proses fitur seleksi dulu sebelum proses *feature discretization* dilakukan.

5 KESIMPULAN

Dari hasil pengujian diatas, dapat disimpulkan bahwa penerapan *feature discretization* berbasis EM *clustering* dapat mengubah fitur kontinyu menjadi diskrit sehingga klasifikasi *spam email* dengan algoritma ID3 dapat dilakukan dan akurasinya meningkat dibanding penggunaan FD selain EM.

Hasil percobaan menunjukkan bahwa dalam klasifikasi *spam email*, ID3 dapat menghasilkan akurasi 91,96% dengan menggunakan jumlah data training 90% dan jumlah fitur sebanyak 56 yang dihasilkan dari nilai $k = 2$ pada EM. Hasil eksperimen juga menunjukkan bahwa akurasi ID3+EM meningkat sebesar 28,05% dibandingkan dengan ID3+ *binning*. Metode *binning* adalah sebuah metode diskritisasi yang umum digunakan dengan memeriksa "nilai tetangga", yaitu dengan mengurutkan dari yang terkecil sampai dengan yang terbesar kemudian dipartisi ke dalam beberapa *bin*.

FD dengan EM *clustering* terbukti dapat meningkatkan akurasi pada algoritma ID3. Namun ada beberapa faktor yang dapat dicoba untuk penelitian selanjutnya, agar dapat menghasilkan metode yang lebih baik lagi, yaitu:

1. Pada penelitian selanjutnya mungkin bisa menggunakan dataset email selain *spambase* UCI seperti *SpamAssassin*

corpus dan *Enron corpus* untuk mencoba performa metode FD dengan EM ini.

2. Penerapan *pruning* pada ID3 untuk meningkatkan akurasi klasifikasi. *Pruning* adalah proses yang dilakukan untuk memotong atau menghilangkan beberapa cabang (*branches*) yang tidak diperlukan. Cabang atau node yang tidak diperlukan dapat menyebabkan ukuran *tree* menjadi sangat besar yang disebut *over-fitting*. *Over-fitting* akan menyebabkan terjadinya misklasifikasi, sehingga tingkat akurasi klasifikasi menjadi rendah

REFERENSI

Agre, G., & Peev, S. (2002). On Supervised and Unsupervised Discretization. *Methods*, *2*(2).

Al-Ibrahim, A. (2011). Discretization of Continuous Attributes in Supervised Learning algorithms. *The Research Bulletin of Jordan ACM - ISWSA*, *7952*(Iv).

Allias, N., Megat, M. N., Noor, M., & Ismail, M. N. (2014). A hybrid Gini PSO-SVM feature selection based on Taguchi method. In *Proceedings of the 8th International Conference on Ubiquitous Information Management and Communication - ICUIMC '14* (pp. 1–5). New York, New York, USA: ACM Press. http://doi.org/10.1145/2557977.2557999

Chakraborty, S., & Mondal, B. (2012). Spam Mail Filtering Technique using Different Decision Tree Classifiers through Data Mining Approach - A Comparative Performance Analysis. *International Journal of Computer Applications*, *47*(16), 26–31.

Chharia, A., & Gupta, R. K. (2013). Email classifier: An ensemble using probability and rules. In *2013 Sixth International Conference on Contemporary Computing (IC3)* (pp. 130–136). IEEE. http://doi.org/10.1109/IC3.2013.6612176

Çiltik, A., & Güngör, T. (2008). Time-efficient spam e-mail filtering using n-gram models. *Pattern Recognition Letters*, *29*(1), 19–33. http://doi.org/10.1016/j.patrec.2007.07.018

Dash, R., Paramguru, R. L., & Dash, R. (2011). Comparative Analysis of Supervised and Unsupervised Discretization Techniques. *International Journal of Advances in Science and Technology*, 29–37.

Ferreira, A. J., & Figueiredo, M. a T. (2012). An unsupervised approach to feature discretization and selection. *Pattern Recognition*, *45*(9), 3048–3060. http://doi.org/10.1016/j.patcog.2011.12.008

Ferreira, A. J., & Figueiredo, M. a T. (2014). Incremental filter and wrapper approaches for feature discretization. *Neurocomputing*, *123*, 60–74. http://doi.org/10.1016/j.neucom.2012.10.036

Gupta, A., Mehrotra, K. G., & Mohan, C. (2010). A clustering-based discretization for supervised learning. *Statistics & Probability Letters*, *80*(9-10), 816–824. http://doi.org/10.1016/j.spl.2010.01.015

Hamsapriya, T., D. K. R. and M. R. C. (2012). A Comparative Study of Supervised Machine Learning Techniques for Spam E-mail Filtering. In *2012 Fourth International Conference on Computational Intelligence and Communication Networks* (Vol. 6948, pp. 506–512). IEEE. http://doi.org/10.1109/CICN.2012.14

Han, J., & Kamber, M. (2006). *Data Mining: Concepts and Techniques*. *Morgan Kaufmann Publishers is an imprint of Elsevier* (Vol. 54). http://doi.org/10.1007/978-3-642-19721-5

I. Witten, E. F. (2011). Data Mining : Practical Machine Learning Tools and Techniques. *Morgan Kaufmann Publishers Is an Imprint of Elsevier*.

Jearanaitanakij, K. (2005). Classifying Continuous Data Set by ID3 Algorithm. In *2005 5th International Conference on Information Communications & Signal Processing* (pp. 1048–1051). IEEE. http://doi.org/10.1109/ICICS.2005.1689212

Jin, C., & De-lin, L. (2009). An Improved ID3 Decision Tree Algorithm. *Proceedings of 2009 4th International Conference on Computer Science & Education*, 127–130.

Jorgensen, Z., Zhou, Y., & Inge, M. (2008). A Multiple Instance Learning Strategy for Combating Good Word Attacks on Spam Filters. *Journal of Machine Learning Research*, *8*, 1115–1146. Retrieved from http://jmlr.csail.mit.edu/papers/volume9/jorgensen08a/jorgensen08a.pdf

Jung, Y. G., Kang, M. S., & Heo, J. (2014). Clustering performance comparison using *K* -means and expectation maximization algorithms. *Biotechnology & Biotechnological Equipment*, *28*(sup1), S44–S48. http://doi.org/10.1080/13102818.2014.949045

Kumar, R. K., Poonkuzhali, G., & Sudhakar, P. (2012). Comparative Study on Email Spam Classifier using Data Mining Techniques. *Proceedings of the International MultiConference of Engineers and Computer Scientists*, *I*.

Ladysz, R. (2004). Clustering of Envolving Time Series Data.

Liu Yuxun, & Xie Niuniu. (2010). Improved ID3 algorithm. In *2010 3rd International Conference on Computer Science and Information Technology*. http://doi.org/10.1109/ICCSIT.2010.5564765

Madhu, G., Rajinikanth, T. V, & Govardhan, A. (2014). Feature Selection Algorithm with Discretization and PSO Search Methods for Continuous Attributes. *International Journal of Computer Science and Information Technologies*, *5*(2), 1398–1402.

Marsono, M. N., El-Kharashi, M. W., & Gebali, F. (2008). Binary LNS-based naïve Bayes inference engine for spam control: noise analysis and FPGA implementation. *IET Computers & Digital Techniques*, *2*(1), 56. http://doi.org/10.1049/iet-cdt:20050180

Méndez, J. R., Glez-Peña, D., Fdez-Riverola, F., Díaz, F., & Corchado, J. M. (2009). Managing irrelevant knowledge in CBR models for unsolicited e-mail classification. *Expert Systems with Applications*, *36*(2), 1601–1614. http://doi.org/10.1016/j.eswa.2007.11.037

Nazirova, S. (2011). Survey on Spam Filtering Techniques. *Communications and Network*, *03*(03), 153–160. http://doi.org/10.4236/cn.2011.33019

Saad, O., Darwish, A., & Faraj, R. (2012). A survey of machine learning techniques for Spam filtering. *Journal of Computer Science*, *12*(2), 66–73.

Sculley, D., & Wachman, G. M. (2007). Relaxed online SVMs for spam filtering. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval - SIGIR '07* (Vol. 36, p. 415). New York, New York, USA: ACM Press. http://doi.org/10.1145/1277741.1277813

Senthilkumar, J., Karthikeyan, S., Manjula, D., & Krishnamoorthy, R. (2012). Web Service Based Feature Selection and Discretization with Efficiency. *2012 IEEE Sixth International Conference on Semantic Computing*, 269–276. http://doi.org/10.1109/ICSC.2012.51

Sharma, A. K., Sahni, S. (2011). A Comparative Study of Classification Algorithms for Spam Email Data Analysis. *International Journal on Computer Science and Engineering (IJCSE)*, (May), 1890–1895.

Sheu, J. J. (2009). An efficient two-phase spam filtering method based on e-mails categorization. *International Journal of Network Security*, *9*(1), 34–43.

Tsai, C.-J., Lee, C.-I., & Yang, W.-P. (2008). A discretization algorithm based on Class-Attribute Contingency Coefficient. *Information Sciences*, *178*(3), 714–731. http://doi.org/10.1016/j.ins.2007.09.004

Wijaya, A., Wahono, R.S. (2008). Two-Step Cluster based Feature Discretization of Naïve Bayes for Outlier Detection in Intrinsic Plagiarism Detection. *Journal of Intelligent Systems*, (February 2015), 2–9.

Wu, C.-H. (2009). Behavior-based spam detection using a hybrid method of rule-based techniques and neural networks. *Expert Systems with Applications*, *36*(3), 4321–4330. http://doi.org/10.1016/j.eswa.2008.03.002

BIOGRAFI PENULIS

Safuan. Lahir pada tanggal 28 Februari 1972 di Kota Semarang, Jawa Tengah. Memperoleh gelar Sarjana Komputer (S.Kom) dari Jurusan Sistem Komputer, STEKOM, Semarang pada tahun 2007. Serta memperoleh gelar M.Kom dari Fakultas Ilmu Komputer, Universitas Dian Nuswantoro pada tahun 2015.

Romi Satria Wahono. Memperoleh Gelar B.Eng dan M.Eng pada fakultas Computer Science, Saitama University, Japan, dan Ph.D pada fakultas Software Engineering, Universiti Teknikal Malaysia Melaka. Mengajar di fakultas Ilmu Komputer, Universitas Dian Nuswantoro, Indonesia. Merupakan pendiri dan CEO Brainmatics, sebuah perusahaan yang bergerak di bidang *software development*, Indonesia. Bidang minat penelitian adalah *Software Engineering* dan *Machine Learning*. Profesional member dari ACM dan asosiasi ilmiah IEEE.

Catur Supriyanto. Dosen di Fakultas Ilmu Komputer, Universitas Dian Nuswantoro, Semarang, Indonesia. Menerima gelar master dari Universiti Teknikal Malaysia Melaka (UTEM), Malaysia. Minat penelitiannya adalah *information retrieval, machine learning, soft computing* dan *intelligent system.*