IMPLEMENTASI ROCCHIO'S CLASSIFICATION DALAM MENGKATEGORIKAN KOMENTAR SPAM DI BLOG

Andre Prilly Kurniawan[1], Ednawati Rainarli[2]

Program Studi Teknik Informatika
Fakultas Teknik dan Ilmu Komputer. Universitas Komputer Indonesia
Jl. Dipati Ukur No. 112-116 Bandung
E-mail : dreprilly@gmail.com[1], irene_edna@yahoo.com[2]

ABSTRAK

Spamming mengacu pada suatu informasi yang tidak diinginkan dan tidak relevan bagi pengguna. Fenomena ini sudah tersebar luas dan sering terlihat pada email, pesan singkat, blog dan forum. Pada penelitian ini kami mempertimbangkan masalah *spam* di blog. Di blog, sistem komentar yang disediakan penulis untuk memfasilitasi interaksi dengan pembaca menjadi target *spammer*. Sebenarnya pemilik blog sudah mencoba menanggulangi masalah ini dengan melakukan *monitoring* dan mengelola komentar secara *manual* dan menggunakan CAPTCHA. Memanfaatkan metode klasifikasi untuk meminimalisir terjadinya serangan komentar spam. Salah satu metode untuk mengklasifikasi adalah Rocchio Classification. Proses pengklasifikasian menggunakan beberapa fitur seperti penggunaan anchor text, selisih waktu antar artikel posting dan komentar, mereferensikan nama pengguna dalam komentar, penghitungan ratio kata dalam komentar dan mengukur tingkat kemiripan antara artikel dengan komentar. Hasil yang diperoleh dari pengujian menunjukkan bahwa metode Rocchio's Classification mampu digunakan untuk mengklasifikasi komentar spam atau komentar organik dengan rata-rata akurasi 95% dari berbagai skenario pengujian.

Kata Kunci: *Spam*, Blog, Komentar Spam, *Rocchio Classification*, Klasifikasi

1. PENDAHULUAN

Selama dekade terakhir, blog menjadi sangat populer di internet. Menurut WordPress, salah satu penyedia layanan *blog publishing* menyatakan bahwa penggunanya rata-rata membuat 69,8 juta postingan baru dan 42 juta komentar baru setiap bulannya [1]. Sayangnya, dengan jumlah traffic yang besar terdapat celah pengelolaan yang kurang baik sehingga blog dapat menjadi sasaran untuk spammers. Sekarang ini, sebenarnya para pemilik blog sudah menggunakan beberapa teknik untuk mengurangi komentar spam. Beberapa pemilik blog memilih melakukan monitoring dan mengelola komentar secara *manual*. Teknik lain yang digunakan pemilik blog untuk membedakan komentar yang dilakukan secara otomatis oleh *bot* dengan komentar asli yang dilakukan oleh *user* adalah dengan menggunakan CAPTCHA [2]. CAPTCHA biasanya berbentuk gambar yang berisi huruf dan angka yang mana sulit untuk dikenali secara otomatis oleh *bot*. Akan tetapi, riset telah membuktikan bahwa metode ini sangat mudah untuk dirusak [3].

Di tahun 2005 Mishne et al. [4] menggunakan pendekatan pemodelan bahasa untuk mendeteksi komentar spam dengan metode Kullback-Leibler divergence mendapatkan tingkat akurasi 83%. Di tahun 2011 Bhattarai et al. [5] menggunakan analisis konten untuk mengidentifikasi spam dengan fitur *words duplications, stopwords ratio etc.*, dengan hasil terbaik menggunakan metode *Support Vector Machine* (SVM). Dari pendekatan tersebut didapatkan tingkat akurasi tertinggi 86%. Di tahun 2012 Ashwin et al. [6] menggunakan analisis komentar dan hubungan antara postingan blog dengan komentar dengan menggunakan beberapa metode klasifikasi didapatkan tingkat akurasi tertinggi menggunakan metode *decision tree* dengan akurasi 92%.

Pada penelitian lain di tahun 2013 Pausta dkk. [7] membandingkan SVM dengan *Rocchio* untuk melakukan penelusuran katalog perpustakaan hasilnya *Rocchio* memiliki waktu pemrosesan lebih kecil 57,2% dan tingkat presisi 37,8% lebih besar dari SVM. *Rocchio Classification* yang diambil dari konsep *Rocchio Relevance Feedback* memiliki konsep desain hanya untuk mengklasifikasi dua kelas yaitu relevan dan tidak relevan [8]. Berdasarkan konsep tersebut pada penelitian ini akan sangat cocok dikarenakan pada penelitian ini akan mengklasifikasikan komentar ke dalam kategori spam atau bukan spam. Oleh karena itu, pada penelitian ini akan dilakukan implementasi *Rocchio Classification* dalam mengidentifikasi

komentar spam dengan harapan mendapatkan tingkat presisi yang lebih baik.

Berdasarkan penjelasan yang telah dipaparkan di atas, maka diharapkan metode yang digunakan merupakan solusi untuk meminimalisir terjadi tindakan *spamming* pada komentar di blog.

Tujuan yang diharapkan akan dicapai dalam penelitian ini adalah:

1. Mengkategorikan komentar spam dengan metode *Rocchio Classification*.
2. Menguji tingkat akurasi *Rocchio Classification* dalam mengkategorikan komentar spam.

2. LANDASAN TEORI

2.1. Spam di Blog

Spam di blog (juga disebut hanya spam blog, spam komentar, atau spam sosial) adalah bentuk spamdexing. Hal ini dilakukan dengan posting (biasanya secara otomatis) komentar acak, menyalin materi dari tempat lain yang tidak asli, atau mempromosikan layanan komersial ke blog, wiki, guestbook, atau forum diskusi online lainnya yang dapat diakses publik. Setiap aplikasi web yang

menerima dan menampilkan hyperlink yang dikirimkan oleh pengunjung mungkin menjadi target. Menambahkan tautan yang mengarah ke situs web spammer secara artifisial meningkatkan peringka situs di mesin pencari dimana popularitas URL berkontribusi terhadap nilai tersiratnya, contohnya adalah algoritme PageRank seperti yang digunakan oleh Google penelusuran. Hal tersebut akan meningkatkan situs komersial spammer yang terdaftar di depan situs lain untuk penelusuran tertentu, meningkatkan jumlah calon pengunjung dan pelanggan yang membayar [9].

2.2. Klasifikasi

Klasifikasi merupakan suatu pekerjaan menilai objek data untuk memasukannya ke dalam kelas tertentu dari sejumlah kelas yang tersedia. Dalam klasifikasi ada dua pekerjaan utama yang dilakukan. Pertama, pembangunan model sebagai prototype untuk disimpan sebagai memori. Kedua, penggunaan model untuk melakukan pengenalan/klasifikasi/prediksi pada suatu objek lain, agar diketahui di kelas mana objek data tersebut dalam model yang sudah disimpannya [10].

2.3. Metode *Rocchio Classification*

Rocchio classifiers merupakan salah satu metode pembelajaran supervised document classification. Metode klasifikasi rocchio membandingkan kesamaan isi antara data training dan data test dengan merepresentsaikan semua data ke dalam sebuah vector. Dalam menggunakan vector space model diperlukan batas-batas antar kelas untuk mengetahui klasifikasi yang sesuai. Teknik Rocchio menerapkan batas-batas tersebut dalam bentuk centroid untuk memberi batasan tersebut. Centroid sebuah kelas c adalah rata-rata semua vektor yang berada pada kelas c. Untuk menghitung nilai centroid dapat dilihat pada persamaan (1).

$$\vec{\mu}(c) = \frac{1}{|D_c|} \sum_{d \in D_c} \vec{v}(d) \quad (1)$$

Dengan:

$\vec{\mu}(c)$ = centroid kelas c

$|D_c|$ = total dokumen kelas c

$\vec{v}(d)$ = vektor dokumen yang telah dinormalisasi

Untuk menentukan kemiripan dua vektor space model yaitu dengan mengukur jarak. Dalam menentukan jarak antara dua vektor space model digunakan *euclidean distance* yang dapat dilihat pada persamaan (2).

$$\mathbf{d(q,p)} = \sqrt{(q_1 - p_1)^2 + (q_2 - p_2)^2 + \cdots + (q_n - p_n)^2} \quad (2)$$

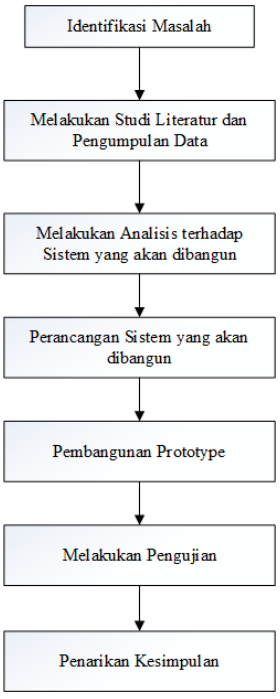
$$= \sqrt{\sum_{i=1}^{n} (q_i - p_i)^2}.$$

Dengan:

p dan q = vektor yang telah dinormalisasi

3. METODE PENELITIAN

Pada penelitian ini metode penelitian yang digunakan yaitu metode penelitian eksperimental. Metode eksperimental adalah metode yang mempunyai tujuan untuk menjelaskan hubungan sebab-akibat antara satu variabel dengan lainnya [14]. Alur penelitian yang akan dilakukan pada penelitian ini dapat dilihat pada gambar 1 sebagai berikut:



Gambar 1. Alur Penelitian

3.1. Metode Pengumpulan Data

Metode pengumpulan data dengan menggunakan data pada penelitian yang dilakukan oleh Mishne et al. disisi lain yang digunakan dalam penelitian ini adalah sebagai berikut:

Studi Pustaka

Studi pustaka dilakukan dengan cara mempelajari, meneliti dan menelaah berbagai literatur dari perpustakaan yang bersumber dari buku-buku, jurnal ilmiah, situs-situs internet, dan bacaan-bacaan yang ada kaitannya dengan topik penelitian.

3.2. Metode Pembangunan Perangkat Lunak

Metode pembangunan perangkat lunak yang digunakan pada penelitian ini adalah model Prototype. Berikut tahapan-tahapan yang dilakukan dalam penelitian ini:

1. Analisis

Analisis masalah dilakukan untuk memahami masalah yang timbul dan mencari solusi untuk memecahkan masalah dalam menghasilkan klasifikasi komentar spam.

2. Kebutuhan Data

Pada tahap ini peneliti akan mengumpulkan data komentar untuk data masukan sistem.

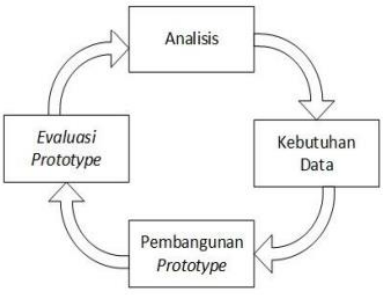
3. Pembangunan *Prototype*

Pada tahap ini akan diimplementasikan dari proses analisis dan kebutuhan sistem yang sudah didapatkan dan peneliti mencoba mengimplementasikan metode *Rocchio Classification* ke dalam logika-logika program.

4. Evaluasi *Prototype*

Program akan diuji dimana uji coba dilakukan untuk mengetahui kekurangan pada program. Jika masih ada kekurangan, maka prototype direvisi dengan tahapan-tahapan yang sebelumnya telah dilakukan.

Tahapan prototype yang dilakukan pada penelitian ini akan dijelaskan pada gambar 2.



Gambar 2. Model *Prototype*

4. HASIL DAN PEMBAHASAN

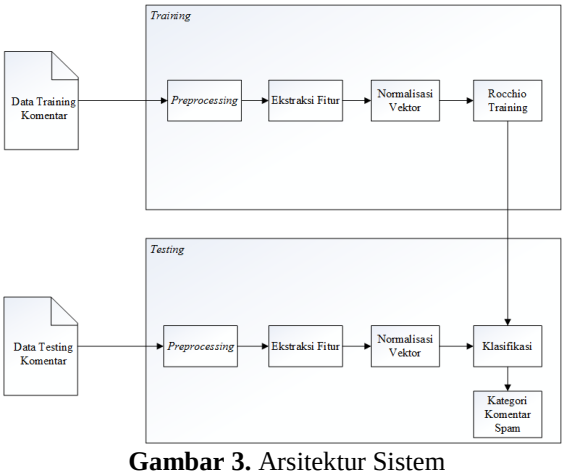
4.1. Analisis Sistem

Klasifikasi komentar spam menggunakan *Rocchio Classification* menjadi permasalahan yang akan dipecahkan dan dibahas pada penelitian ini, implementasi *Rocchio Classification* digunakan dalam pengklasifikasian komentar spam, sistem pengklasifikasian komentar spam berdasarkan fitur-fitur yang akan menjadi acuan dalam proses klasifikasi. Sistem yang dirancang dapat dijalankan pada perangkat komputer (PC) dengan bahasa pemrograman Javascript, yang dapat digunakan untuk melihat hasil klasifikasi komentar spam.

Dalam membangun sistem klasifikasi komentar spam dilakukan beberapa tahapan analisis. Tahapan dari sistem yang dibuat ditunjukkan pada gambar 3. Tahapan yang dilakukan oleh sistem dibagi menjadi dua tahap. Tahap pertama yaitu pelatihan dan tahap kedua yaitu pengujian.

Pada tahap pertama dimulai dengan melakukan ekstraksi pada data latih untuk mendapatkan fitur seperti jumlah *anchor text*, selisih waktu tanggal *posting blog* dengan komentar dan kemunculan nama pengguna dalam kolom komentar. Kemudian dilanjutkan ke tahap *preprocessing*. Hasil dari *preprocessing* diekstraksi kembali untuk mendapatkan fitur yang lain seperti *ratio* pengulangan kata dan *post-commeny simmilarity*.

Pada alur proses sistem yang dijelaskan di atas, terdapat dua tahap dalam pengumpulan fitur yang digunakan, hal ini disebabkan fitur seperti jumlah *anchor text*, selisih waktu tanggal *posting blog* dengan komentar dan kemunculan nama pengguna dalam kolom komentar hanya bisa didapatkan sebelum tahap *preprocessing*. Kemudian setelah kelima fitur dalam bentuk vektor didapatkan, dilakukan tahap pelatihan dengan menggunakan metode *Rocchio Classification*. Pada tahap kedua yaitu pengujian, tahapan proses tidak jauh berbeda dengan tahapan pelatihan, yang membedakan pada tahapan pengujian dilakukan proses klasifikasi untuk menentukan suatu komentar dianggap spam atau bukan spam.



Gambar 3. Arsitektur Sistem

4.2. Analisis Data Masukan

Analisis data masukan yang dibutuhkan pada penelitian ini yaitu memiliki variabel posting blog, tanggal posting blog, *author* komentar, komentar, tanggal komentar dan kategori kelas komentar yang diformat dalam bentuk json yang diambil dari

penelitian Mishne dan Carmel yang berjumlah 400 data dari perkiraan 20 posting blog.

4.3. Ekstraksi Fitur

Komentar spam biasanya memiliki ciri tertentu yang mana akan digunakan sebagai fitur untuk membedakan dengan komentar bukan spam. Pada penelitian ini akan menggunakan lima fitur yaitu: jumlah anchor text, ratio pengulangan kata, post-comment similarity, kemunculan nama pengguna dalam kolom komentar dan selisih waktu tanggal postingan dengan komentar.

4.3.1. Jumlah Anchor Text

Teks yagn muncul di HTML antara tag <a ...> dan disebut sebagai *anchor text*. Teks-teks ini pada dasarnya membuat tautan ke halaman lain yang dapat terhubung dengan mengklik *hyperlink*. *Web crawlers* biasanya mengikuti tautan ini secara iteratif untuk menjelajahi laman web di internet. Komentar spam mencoba memasukkan banyak teks tautan yang mengarah ke situs *spammer* untuk meningkatkan peringkat halaman mereka di *search engine*. Berikut contoh komentar *spam* dengan beberapa *anchor text* ditunjukkan pada gambar 4 di bawah ini.

Spy Software http://spy-software.seo.se.com
Internet Spy Software http://internet-spy-software.seo.se.com

Gambar 4. Contoh Komentar *Spam* yang berisi banyak URL

4.3.2. Selisih Waktu Tanggal Postingan dengan Komentar

Sebuah artikel biasanya ditulis karena topik tersebut lagi hangat dibicarakan banyak orang contohnya seperti topik pilkada ketika memasuki masa pilkada, atau topik piala dunia sepakbola ketika menjelang piala dunia bergulir. Komentar yang bukan spam biasanya dilakukan ketika topik tersebut masih dalam lingkup yang hangat dibicarakan sedangkan komentar spam tidak tergantung terhadap waktu. Berikut rumus untuk menghitung selisih waktu tanggal postingan dengan komentar ditunjukkan pada gambar 5 di bawah ini.

$$\text{Selisih Waktu} = \text{Tanggal Posting Blog} - \text{Tanggal Komentar}$$

Gambar 5. Rumus Selisih Waktu Tanggal Postingan dengan Komentar

4.3.3. Kemunculan Nama Pengguna dalam Kolom Komentar

Sistem komentar selalu menyediakan kolom nama yang mana digunakan untuk memasukkan nama pengkomentar. Komentar yang bukan spam secara umum tidak akan memasukkan namanya di dalam komentarnya sedangkan spammer menggunakan *keywords* sebagai namanya dan memasukkannya ke dalam komentar, hal tersebut bertujuan untuk meningkatkan *keywords* pada *search engine*. Berikut contoh kemunculan nama pengguna dalam kolom komentar ditunjukkan pada gambar x di bawah ini.

Get a Generic Viagra alternative at Cheap Generic ViagraGet a Cheap Generic Viagra alternative at Cheap Generic Viagra
Comments posted by: Cheap Generic Viagra at March 11, 2005 09:27 PM

Gambar 6. Contoh Komentar *Spam* yang Berisi Nama Pengguna di dalam Komentarnya

4.3.4. Ratio Pengulanan Kata

Komentar *spam* menggunakan pengulangan kata-kata untuk menarik *search engine* sedangkan komentar organik lebih sering mengalir mengikuti konteks artikel terkait. Karena sebagian besar komentar *blog* bersifat singkat, kata yang sama jarang terulang dalam komentar organik.Berikut rumus untuk menghitung *ratio* pengulangan kata ditunjukkan pada gambar x di bawah ini.

$$\text{Ratio Pengulangan Kata} = 1 - \frac{\textit{Jumlah kata unik di komentar}}{\textit{Jumlah total kata di komentar}}$$

Gambar 7. Rumus Ratio Pengulangan Kata

4.3.5. Post-Comment Similarity

Spammer menggunakan *script* yang dihasilkan komputer untuk menghasilkan jutaan komentar *spam* yang siap dikirim. Namun, dalam banyak kasus, komentar *spam* otomatis ini tidak terkait dengan konteks artikel *blog*. Berikut contoh komentar *spam* yang tidak terkait dengan konteks artikel *blog* ditunjukkan pada gambar x di bawah ini.

Hi, I just wanted to say thank you guys! I really like your site and I hope you'll continue to improving it.

Gambar 8. Contoh Komentar *Spam* yang Tidak Terkait Konetks Artikel *Blog*

4.4. Pengujian Skenario Pertama

Pengujian skenario pertama dilakukan dengan menguji komentar yang termasuk dalam data latih, pengujian ini bertujuan untuk mengetahui tingkat pengenalan terhadap data komentar yang sudah dilatih. Data komentar yang digunakan berjumlah 400 data yang terdiri dari 2 kelas dengan masing masing kelas terdapat 200 data.

Tabel 1. Confusion Matrix Pengujian Skenario Pertama

Kelas		Prediksi		Akurasi
		Spam	Bukan Spam	
Target	Spam	195	5	97,5%
	Bukan Spam	12	188	94%
Rata-rata				95,7%

4.5. Pengujian Skenario Kedua

Pengujian skenario kedua dilakukan dengan menguji komentar yang berbeda dengan data latih,

pengujian ini bertujuan untuk mengetahui tingkat pengenalan terhadap data komentar di luar data latih. Data latih yang digunakan berjumlah 300 data yang terdiri dari 2 kelas dengan masing masing kelas terdapat 150 data dan data uji yang digunakan berjumlah 100 data yang terdiri dari 2 kelas dengan masing-masing kelas terdapat 50 data.

Tabel 2. Confusion Matrix Pengujian Skenario Kedua

Kelas		Prediksi		Akurasi
		Spam	Bukan Spam	
Target	Spam	47	3	94%
	Bukan Spam	50	0	100%
Rata-rata				97%

4.6. Pengujian Skenario Ketiga

4.6.1. Pengujian dengan Nilai K-Fold yang digunakan adalah 5

Pengujian dengan nilai k-fold 5 adalah pengujian sebanyak 5 kali putaran, artinya dataset dibagi menjadi 5 sama banyak. Pada penelitian ini data yang digunakan sebanyak 400 data dan akan dibagi menjadi 5 yaitu data A1 = 80, data A2 = 80, data A3 = 80, data A4 = 80 dan data A5 = 80. Pada putaran pertama, data A1 digunakan sebagai data uji sedangkan A2 sampai dengan A5 digunakan sebagai data latih. Pada putaran kedua, data A2 digunakan sebagai data uji sedangkan data A1, A3, A4 dan A5 digunakan sebagai data latih. Begitu juga pada putaran ketiga dst sehingga setiap grup data akan mendapatkan giliran menjadi data uji dan data latih.

Pengujian	Spam	Bukan Spam	Prediksi Benar	Akurasi
A1	58	22	78	97,5%
A2	43	37	79	98,7%
A3	62	18	75	93,7%
A4	37	43	73	91,2%
A5	0	80	78	97,5%
Rata-rata				95,72%

4.6.2. Pengujian dengan Nilai K-Fold yang digunakan adalah 8

Pengujian dengan nilai k-fold 8 adalah pengujian sebanyak 8 kali putaran, artinya dataset dibagi menjadi 50 sama banyak. Pada penelitian ini data yang digunakan sebanyak 400 data dan akan dibagi menjadi 8 yaitu data A1 = 50, data A2 = 500, data A3 = 50, data A4 = 50, data A5 = 50, data A6 = 50, data A7 = 50 dan data A8 = 50. Pada putaran pertama, data A1 digunakan sebagai data uji sedangkan A2 sampai dengan A8 digunakan sebagai data latih. Pada putaran kedua, data A2 digunakan sebagai data uji sedangkan data A1, A3, A4, A5, A6, A7 dan A8 digunakan sebagai data latih. Begitu juga pada putaran ketiga dst sehingga setiap grup data akan mendapatkan giliran menjadi data uji dan data latih.

Pengujian	Spam	Bukan Spam	Prediksi Benar	Akurasi
A1	36	14	50	100%
A2	25	25	48	96%
A3	30	20	49	98%
A4	46	4	47	94%
A5	30	20	47	94%
A6	28	22	47	94%
A7	5	45	45	90%
A8	0	50	50	100%
Rata-rata				95,75%

4.7. Kesimpulan Pengujian

Berdasarkan hasil skenario pengujian pertama yaitu pengujian data latih sama dengan data uji dapat ditarik kesimpulan bahwa metode rocchio classification dapat mengklasifikasikan dengan akurasi sebesar 95,7%. Kemudian berdasarkan hasil skenario pengujian kedua yaitu pengujian data uji tidak terdapat dalam data latih, metode rocchio classification dapat mengklasifikasikan dengan akurasi sebesar 97%.

Berdasarkan hasil skenario pengujian ketiga yaitu pengujian dengan menggunakan metode k-fold cross validation, metode rocchio classification dapat mengklasifikasikan dengan rata-rata akurasi sebesar 95,72% dengan nilai k adalah 5 dan 95,75% dengan nilai k adalah 8.

Dari hasil pengujian, kategori komentar bukan spam lebih sulit dikenali daripada kategori komentar spam. Keakuratan metode rocchio classification memiliki tingkat yang cukup baik dalam mengkategorikan komentar.

5. KESIMPULAN

Berdasarkan pembahasan tahapan pengujian maka dapat ditarik kesimpulan bahwa metode *Rocchio Classification* memiliki nilai akurasi yang cukup baik dalam mengklasifikasikan komentar dari beberapa skenario pengujian dan dalam beberapa kasus komentar spam yang cenderung mirip dengan komentar organik, *Rocchio Classification* cukup kesulitan untuk memprediksi dengan benar.

Adapun saran yang dapat diberikan untuk pengembangan selanjutnya yaitu penanganan kata tidak baku dan penambahan fitur baru untuk mempelajari ciri-ciri spesifik komentar spam maupun organik.

DAFTAR PUSTAKA

[1] WordPress, “Stats – WordPress.com”, [Daring]. Tersedia pada: https://wordpress.com/activity/. [Diakses 27 Agustus 2016].

[2] Saini, B.S, Bala, A.: “Bot Protection using CAPTCHA: Gurmukhi Script”, Vol. 2, pp. 267, May 2013.

[3] Mori, G., Malik, J.: “Recognizing Objects in Adversarial Clutter: Breaking a Visual CAPTCHA” In IEEE, 2003.
[4] Mishne et al.: “Blocking Blog Spam with Language Model Disagreement”, 2005.
[5] Bhattarai et al.: “A Self-supervised Approach to Comment Spam Detection based on Content Analysis”, 2011.
[6] Ashwin et al.: “Comment Spam Classification in Blogs through Comment Analysis and Comment Blog Post Relationships”, 2012.
[7] Yugianus dkk.: “Pengembangan Sistem Penelusuran Katalog Perpustakaan Dengan Metode Rocchio Relevance Feedback”, 2013.
[8] Manning et al.: “Introduction to Information Retrieval”, Chapter 14, 2009, hal. 292-295.
[9] Saleh, Rachmad. Spam dan Hijacking Email. Jakarta : Andi Publisher, 2008, hal. 06 – 46.
[10] Prasetyo, Eko. 2014. Data Mining. Yogyakarta: Andi Offset.