

# Klasifikasi SMS Center RSUD SMART Berdasarkan Jenis Keluhan Pelayanan Menggunakan TS Smoothing

Sholeh Rachmatullah, *Teknik Informatika Universitas Madura*, Esther Irawati Setiawan, *Teknologi Informasi Institut Sains dan Teknologi Terpadu Surabaya*, Reddy Alexandro Harianto, *Informatika Institut Sains dan Teknologi Terpadu Surabaya*

**Abstrak** – RSUD SMART memiliki layanan *SMS Center* yang digunakan untuk berinteraksi dengan masyarakat dalam menerima pertanyaan, masukan, saran dan kritik maupun pengaduan. Informasi yang diterima dalam bentuk SMS hanya disimpan dan tidak dikelompokkan berdasarkan unit atau layanan yang dituju sehingga pihak manajemen rumah sakit tidak bisa mengukur tingkat pelayanan di tiap unit. Penelitian ini melakukan klasifikasi terhadap data SMS dan saran responden dari masyarakat yang ditujukan kepada Direktur Rumah Sakit. Data SMS yang diklasifikasi berasal dari *database* aplikasi SMS Center RSUD SMART Pamekasan. Data SMS dan saran responden diklasifikasikan dalam 10 kelas yaitu Rawat Inap, Rawat jalan, Laboratorium, Farmasi, BPJS, Humas, Loket Pembayaran, Sarana dan Prasarana, Profesi dan tidak Terklasifikasi, serta melakukan *scoring SMS*. Sebelum melakukan proses klasifikasi terlebih dahulu dilakukan *pre-processing* seperti penyamaan karakter, penghapusan tanda baca, mengembalikan singkatan, terjemah bahasa daerah (Bahasa Madura), penghapusan angka, penghapusan kata yang tidak penting dalam SMS, dan *stemming* untuk mengubah kata menjadi kata dasar. Penelitian ini menggunakan algoritma *Naive Bayes* dengan *Two Stage (TS) Smoothing*. Dalam beberapa uji coba yang telah dilakukan terhadap 2292 data dengan presentase data training sebesar 20%, 30%, 40% dan 50% mendapatkan rata-rata akurasi sebesar 82,97% dengan nilai  $\lambda=0.2$ ,  $\mu=2000$  dan  $\text{threshold}=3$ . Bahkan dalam salah satu uji coba klasifikasi dengan *threshold* statis mencapai akurasi 86,73% sedangkan akurasi terendah dengan *threshold* dinamis mencapai 74,28%. Pengaturan *threshold* statis terbukti meningkatkan akurasi klasifikasi sebesar 6,14%.

**Kata Kunci:** *SMS Center*, klasifikasi, *Naive Bayes*, *TS Smoothing*

## I. PENDAHULUAN

**S**MMS Center merupakan layanan SMS *gateway* yang digunakan oleh RSUD dr. H. Slamet Martodirdjo (SMART) kabupaten Pamekasan untuk berinteraksi dengan

masyarakat menggunakan fasilitas SMS berupa pertanyaan, masukan, saran, kritik maupun pengaduan dari masyarakat, dan digunakan sebagai media untuk menyampaikan SMS di lingkup penanggung jawab unit. Informasi berupa SMS yang diterima dalam dua tahun hanya disimpan dalam data base dalam jumlah besar dan tidak terklasifikasi secara khusus.

Hal ini menyebabkan petugas kesulitan menghitung statistik keluhan masyarakat berdasarkan isi pesan singkat di setiap tahunnya. Penulis berusaha untuk mengklasifikasikan data SMS tersebut ke dalam *multi class* [1] berdasarkan ruang lingkungannya [2] dan hasilnya dapat dipergunakan oleh pihak manajemen untuk melakukan evaluasi di setiap unit serta proyeksi pengembangan Sumber Daya Manusia. Salah satu keluaran dari hasil klasifikasi misalnya dengan memberikan surat rekomendasi berupa surat Evaluasi Internal RSUD SMART terhadap unit pelayanan yang mendapatkan skor tertinggi dari laporan yang diterima melalui SMS Center [3].

Terdapat 4 jenis teknik *smoothing* [4], [5] di dalam penggunaan metode *Naive Bayes* untuk proses klasifikasi teks. Teknik tersebut adalah Jelinek-Mercer (JM) *smoothing*, Dirichlet (Dir) *smoothing*, *Absolute Discounting (AD) smoothing*, dan *Two-Stage (TS) smoothing*. Dari hasil penelitian tersebut diperlihatkan bahwa metode *smoothing* mampu meningkatkan tingkat akurasi dari *Naive Bayes* untuk proses klasifikasi teks.

Untuk memaksimalkan performa dari *Naive Bayes Classification* [6] dalam penelitian ini penulis menggunakan *Two-Stage Smoothing*. *Two-Stage Smoothing* digunakan karena dikenal dengan akurasi dan kecepatan dalam proses klasifikasi teks dibandingkan dengan tanpa menggunakan *smoothing* atau hanya mengandalkan *Naive Bayes* untuk mengklasifikasikan SMS [3]. Metode ini digunakan untuk menyelesaikan permasalahan *multi labeling* [7], [8], [9] dan multi klasifikasi yang disesuaikan dengan pola atau model data input yang ada dalam sistem sebelumnya.

## II. TINJAUAN PUSTAKA

*Text Classification* harus melalui beberapa tahapan yang harus dilakukan secara berurutan mulai dari awal atau input sampai dengan outputnya. Hal tersebut disebabkan karena masing-masing tahapan memiliki keterkaitan antara satu

13 Juni 2019, RSUD SMART

Sholeh Rachmatullah, Teknik Informatika, Universitas Madura, Pamekasan, Jawa Timur, Indonesia (e-mail: sholeh@unira.ac.id)

Esther Irawati Setiawan, Teknologi Informasi, Institut Sains dan Teknologi Terpadu Surabaya, Surabaya, Jawa Timur, Indonesia (e-mail: esther@stts.edu)

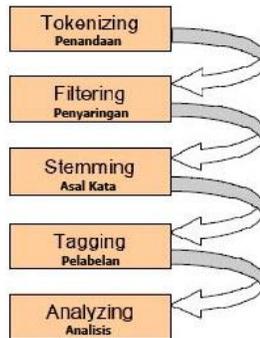
Reddy Alexandro Harianto, Informatika, Institut Sains dan Teknologi Terpadu Surabaya, Surabaya, Jawa Timur, Indonesia (e-mail: reddy@stts.edu)

dengan yang lainnya. Tahapan-tahapan dalam *Klasifikasi teks* tersebut memiliki tujuan untuk mengetahui pola dokumen yang diolah.

**A. Tahapan Klasifikasi**

Tahapan-tahapan dalam klasifikasi teks tersebut adalah sebagai berikut:

1. Tahapan *Tokenizing*. Pada tahap ini *string input* dipotong menurut setiap kata yang menyusunnya.
2. Tahapan *Filtering*. Tahapan ini merupakan kelanjutan tahapan *tokenizing* dimana kata-kata penting yang dihasilkan pada tahapan *tokenizing* diambil untuk disaring menggunakan dua metode yaitu *stop list* yang artinya membuang kata yang kurang penting, ataupun dengan metode *word list* yang artinya menyimpan kata penting. Kedua metode ini berfungsi untuk menyimpan kata yang dianggap penting. Tahapan ini sangat penting mengingat sebuah dokumen sering menggunakan kata penghubung.
3. Tahapan *Stemming*. Pada tahapan ini kata dasar atau akar kata yang berasal dari hasil *filtering* dicari, adalah tahap dimana mencari akar kata dari tiap kata hasil *filtering*, misalkan dengan membuang imbuhan yang ada pada kata tersebut. Proses ini menggunakan algoritma ECS [2], pengembangan dari *Confix Stemmer* [10].
4. Tahapan *Tagging*. Tahapan ini berfungsi untuk mencari bentuk awal dari tiap kata lampau dari hasil *stemming*. Bedanya tahapan ini tentu saja tidak dilakukan untuk teks Bahasa Indonesia yang tidak mengenal istilah kata lampau.
5. Tahap *Analyzing*. Tahapan ini merupakan tahap penentuan seberapa jauh keterhubungan antar kata dengan dokumen yang ada. Tahapan klasifikasi lihat Gambar 1.



Gambar 1. Tahapan *Text Classification*.

**B. Algoritma Naive Bayes dan Smoothing**

Algoritma *Naive Bayes* [6] merupakan pengklasifikasian dengan metode probabilitas dan statistik yang dikemukakan oleh ilmuwan Inggris Thomas Bayes, yaitu memprediksi peluang di masa depan berdasarkan pengalaman di masa sebelumnya sehingga dikenal sebagai teorema Bayes. *Naive Bayes* sendiri menggunakan kemiripan fitur antara data training dan data testing dimana nantinya akan diambil class yang paling mirip dari data training tersebut. Berdasarkan beberapa penilaian, algoritma ini dikenal sebagai algoritma yang sederhana, cepat dan berakurasi tinggi. *Naive Bayes* diasumsikan setiap atribut saling bebas, maka persamaan yang didapatkan adalah sebagai berikut:

$$P(c|d) = \frac{P(d|c)}{P(c)+P(d)} \quad (1)$$

Dimana *c* adalah sebuah kelas, *d* adalah sebuah dokumen. Setiap peluang pada masing-masing kelas akan dikalikan dan akan menghasilkan nilai *Naive Bayes* pada masing-masing rumus tersebut. Nilai tertinggi pada klasifikasi ini akan menjadi hasil klasifikasi dari *Naive Bayes* tersebut. Berikut adalah rumusnya:

$$C_{map} = \max_{c \in C} \hat{P}(c|d) = \max_{c \in C} \hat{P}(c) \prod_{1 \leq k \leq nd} P(t_k|c) \quad (2)$$

$$\hat{P}(c) = \frac{N(c)}{N} \quad (3)$$

Dimana  $\hat{P}(c|d)$  adalah estimasi probabilitas Kategori *c* terhadap dokumen *d*,  $\hat{P}(c)$  adalah estimasi probabilitas *prior* dari dokumen muncul di kategori *c*,  $P(t_k|c)$  adalah probabilitas bersyarat dari term *t<sub>k</sub>* yang muncul di kategori *c*.

*Smoothing* untuk NBC ini mengacu pada penyesuaian estimator maksimum *likelihood* untuk model bahasa sehingga akan lebih akurat. *Smoothing* sendiri tidak diperlukan untuk menetapkan nilai nol untuk kata yang tak terlihat. *Smoothing* memainkan dua peran penting yaitu meningkatkan akurasi model bahasa, serta mengakomodasi generasi pada umumnya dan kata noninformatif. Generator kemungkinan maksimum umumnya di bawah perkiraan probabilitas dari kata yang tak terlihat. Jadi tujuan utama *smoothing* adalah untuk memberikan probabilitas tidak nol untuk kata-kata yang tak terlihat dan meningkatkan akurasi probabilitas estimator.

Untuk menghindari nilai 0 pada hasil perhitungan probabilitas bersyarat dari *term* tertentu yang muncul di sebuah kategori diperlukan metode *smoothing* [5]. Beberapa metode *smoothing* yang dapat dilakukan, tetapi dari penelitian sebelumnya terdapat beberapa metode *smoothing* yang sudah diujicobakan pada *Naive Bayes* meliputi [4] :

1. Laplace Smoothing.

$$p(w|c_i) = \frac{1+c(w,c_i)}{|V|+\sum_{w' \in V} c(w',c_i)} \quad (4)$$

2. Two-Stage (TS) Smoothing, merupakan kombinasi antara Dirichlet Smoothing dengan metode interpolation.

$$p_{\lambda,\mu}(w|c_i) = (1 - \lambda) \frac{c(w,c_i)+\mu p(w|C)}{\sum_{w' \in V} c(w',c_i)+\mu} + \lambda p(w|C) \quad (5)$$

Modifikasi ini didasarkan pada penggunaan persamaan rumus laplace dan Two Stage Smoothing menjadi seperti dibawah ini:

$$p_{\lambda,\mu}(w|c_i) = (1 - \lambda) \frac{1+c(w,c_i)+\mu p(w|C)}{\sum_{w' \in V} c(w',c_i)+\mu} + \lambda p(w|C) \quad (6)$$

Modifikasi dilakukan jika pada saat perhitungan nilai probabilitas dengan menggunakan persamaan rumus Two-Stage Smoothing sudah ditentukan dan kata unik hasil *stemming* tidak ada di dalam semua basis data training maupun di dalam basis data training kelas itu sendiri.

**C. Multi Klasifikasi Teks**

Kategorisasi dokumen teks secara multi-label adalah masalah yang berperan penting dalam sistem temu kembali informasi. Klasifikasi multi-label berbeda dengan klasifikasi single label. Klasifikasi *single label* akan mengklasifikasikan suatu dokumen ke dalam satu kategori dokumen saja. Sedangkan klasifikasi multi-label dapat mengelompokkan suatu dokumen ke lebih dari satu kategori. Sebagai contoh, sebuah artikel ilmiah yang di dalamnya membahas secara rinci

mengenai deteksi dini kanker ovarium dengan pendekatan bioinformatika dapat memiliki lebih dari satu label dokumen yaitu: kesehatan, bioinformatika dan wanita.

Untuk mengklasifikasikan artikel ke dalam *multi label* [7], dilakukan pendekatan dengan mentransformasikan klasifikasi multi label menjadi beberapa kasus klasifikasi *single label* [6]. Dengan melakukan transformasi *single label* maka tiap-tiap data dilakukan pengecekan terhadap semua label yang ada. Pengecekan yang dilakukan akan menentukan apakah data tersebut “termasuk” atau “tidak termasuk” pada suatu label. Hal ini kurang efisien karena memerlukan waktu komputasi yang cukup lama untuk mengecek suatu dokumen ke semua label. Metode ini tidak memperhitungkan korelasi antara label yang satu dengan label yang lain.

Penentuan multi klasifikasi dilakukan dengan dua cara yaitu dengan memberikan nilai threshold statis dan nilai threshold dinamis. Cara kerja multi klasifikasi mengambil 2 nilai probabilitas tertinggi dari kelas tertentu, hal ini dilakukan untuk menentukan apakah dokumen tersebut hanya tergolong *single classification* atau *multi classification*. Pada penentuan multi klasifikasi dengan nilai threshold statis, sistem akan menetapkan nilai threshold yang diinput secara statis, kemudian melakukan perbandingan selisih antara nilai threshold yang sudah ditentukan dengan nilai probabilitas terbesar kedua. Jika nilai threshold lebih besar dari nilai probabilitas terbesar kedua maka program akan menjadikan probabilitas kelas tersebut sebagai dokumen multi kelas. Ketika nilai threshold lebih kecil atau sama dengan nilai probabilitas terbesar kedua, maka dokumen tersebut hanya memiliki *single classification*.

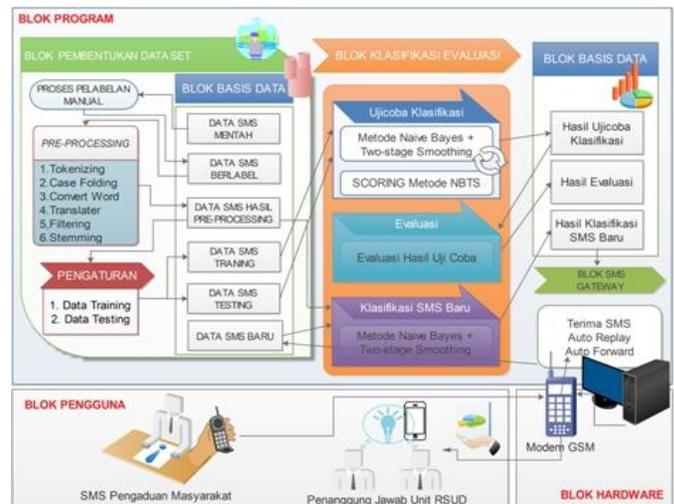
Pada penentuan *multi classification* [9] dengan nilai threshold dinamis, maka setiap dokumen yang diuji coba akan memiliki nilai threshold yang berubah-ubah tergantung pada jarak antar probabilitas kelas yang satu dengan probabilitas kelas lainnya. Cara kerja dari konsep ini adalah jika jarak nilai probabilitas kedua terbesar dengan nilai probabilitas terbesar pertama lebih dominan dari pada nilai probabilitas kelas lainnya terhadap nilai probabilitas pertama, maka probabilitas kelas tersebut merupakan bagian dari kelas probabilitas terbesar pertama yang berarti dokumen tersebut terdeteksi sebagai multi klasifikasi. Namun jika jaraknya tidak lebih dominan dari pada kelas lainnya maka dokumen diklasifikasikan ke dalam satu kelas saja [11].

### III. METODOLOGI PENELITIAN

Penelitian ini memiliki input utama yaitu data SMS yang masuk atau diterima pada database aplikasi SMS Center RSUD dr. H. Slamet Martodirdjo Kabupaten Pamekasan dan form isian komentar responden hasil survei kepuasan masyarakat selama 4 bulan yang dilakukan dalam kurun waktu bulan Mei dan Juni tahun 2015 untuk tahap pertama, kemudian pengambilan data responden tahap ke dua pada bulan Oktober dan Nopember tahun 2015 oleh LP3M Akademi Keperawatan Pemerintah Kabupaten Pamekasan dengan total keseluruhan yaitu 2292 data dan 415 data yang termasuk multi label klasifikasi. Sistem ini bersifat *multi-class* dimana setiap SMS yang diterima akan masuk ke salah satu atau lebih kelas yang selanjutnya akan diteruskan ke

penanggungjawab unit terkait yang disesuaikan dengan hasil klasifikasi oleh aplikasi.

Beberapa data SMS yang tersimpan dalam database Aplikasi SMS RSUD Pamekasan terdiri dari nomor urut, nomor pengirim, waktu diterimanya SMS (jam, tanggal, bulan dan tahun) teks SMS, isi pesan SMS dan jenis teks. Dalam proses klasifikasi data yang digunakan hanya teks SMS sehingga diperlukan dilakukan proses cleaning dengan mengambil data teks SMS saja, sedangkan nomor pengirim, waktu diterimanya SMS boleh diabaikan hal ini karena tidak berpengaruh terhadap proses selanjutnya. Sedangkan form isian komentar responden hasil survei kepuasan masyarakat selama 4 bulan pada bulan Mei, Juni, Oktober dan Nopember tahun 2015 oleh LP3M Akademi Keperawatan Pemerintah Kabupaten Pamekasan, ini merupakan hasil kerjasama dengan pihak RSUD yang dilakukan hanya dalam 1 tahun saja atau tidak rutin. Formulir yang berasal dari masyarakat berisikan form isian no responden, lokasi ruangan, nama, alamat asal, jenis kelamin, pendidikan, pekerjaan, usia, tanggal, status pemeriksaan, hari rawat ke, jawaban pilihan ganda dan saran responden. Data responden yang akan di masukkan ke dalam sistem aplikasi ini adalah hanya data saran responden dan proses pelabelan setiap isian data saran dari responden.



Gambar 2. Arsitektur Sistem

Dalam penelitian ini data SMS dan saran responden diklasifikasikan ke dalam dalam sepuluh kelas antara lain Humas, Sarana dan Prasarana, Laboratorium, Farmasi, Rawat Inap, Rawat Jalan, Loker Pembayaran, BPJS, Profesi dan Tidak Terklasifikasi. Kelas Tidak Terklasifikasikan diperuntukkan untuk SMS yang sifatnya tidak spesifik mengarah ke sembilan kelas yang sudah disebutkan di atas, misalnya SMS Spam yang bersal dari orang yang tidak bertanggungjawab, [12] SMS Iklan yang berisikan berbagai promosi layanan atau penjualan dari orang lain, dan SMS Operator merupakan informasi yang dikirimkan oleh operator penyedia jaringan kartu selular, biasanya SMS yang diterima dari operator memiliki nomor khusus empat sampai enam digit nomor pengirim.

Klasifikasi tersebut dimaksudkan untuk mendistribusikan SMS ke masing-masing unit agar ditangani dan dijadikan bahan monitoring dan evaluasi unit tersebut. Setiap SMS yang masuk diproses untuk mengetahui kecocokan data dengan salah satu atau lebih kelas yang tersedia. SMS baru yang diterima oleh SMS Gateway akan proses sistem, sistem mengolah data SMS dengan melakukan *pre-processing* dan melakukan proses klasifikasi sehingga menghasilkan multi kelas, kemudian aplikasi akan meneruskan pesan SMS tersebut kepada unit yang dituju berdasarkan hasil klasifikasi. Selain unit, pihak manajemen rumah sakit juga dapat memonitoring dan mengevaluasi kinerja pelayanan dengan lebih mudah. Dari hasil klasifikasi setiap periode tertentu, sistem akan memberikan rekomendasi berupa surat pemberitahuan terhadap unit yang memiliki skoring tertinggi berdasarkan laporan dari data *SMS Center*. Model klasifikasi yang dibangun akan dijabarkan secara menyeluruh dalam suatu sistem yang terstruktur yang disajikan pada Gambar 2.

Membangun arsitektur sistem untuk mempermudah kerangka kerja sistem yang akan dikerjakan, dalam hal ini dilakukan pemisahan blok untuk program, hardware dan pengguna. Pada Gambar 2 arsitektur sistem bagian BLOK PROGRAM merupakan bagian dari sistem perangkat lunak yang berfungsi untuk melakukan pengolahan dan pemrosesan data. BLOK PROGRAM memiliki 4 sub blok diantaranya Blok Pembentukan Data Set, Blok Basis Data, Blok Klasifikasi Evaluasi, dan Blok *SMS Gateway*.

Proses awal ditunjukkan dalam Blok Pembentukan Data Set, ada 3 proses yang dilakukan yaitu proses pelabelan secara manual pada data SMS mentah, *Pre-Processing*, dan pengaturan data *testing* serta *training*. Pada proses *Pre-Processing* ada tahapan *stemming* yang menggunakan algoritma *Enhanced Confix Stripping (ECS) Stemmer* [13]. Pada Blok Basis Data mempunyai 9 jenis data yaitu data SMS mentah, SMS berlabel, SMS hasil *pre-processing*, SMS *trainig*, SMS *testing*, hasil uji coba klasifikasi, hasil evaluasi, SMS baru, dan hasil klasifikasi SMS baru.

Blok Klasifikasi Evaluasi terdiri dari 3 proses yaitu proses pertama yaitu Uji Coba Klasifikasi berfungsi untuk melakukan perhitungan menggunakan metode *Naive Bayes* dengan *Two-Stage Smoothing (NBTS)* [4] dan *Scoring*. *Scoring* menggunakan metode NBTS untuk menghitung skor masing masing dokumen agar diketahui tingkat keluhan yang ada. Tingkat keluhan dibagi menjadi tiga yaitu ringan dengan skor 1, sedang dengan skor 3, dan tinggi dengan skor 5. Nilai rata rata skor diperoleh dari jumlah dokumen dengan total skor dibagi 2. Selanjutnya proses kedua Evaluasi berfungsi menganalisa hasil uji coba klasifikasi agar sesuai harapan penelitian, dan proses ketiga Klasifikasi SMS Baru berfungsi untuk mengklasifikasikan data SMS baru yang diterima dari blok *SMS gateway*. BLOK HARDWARE berfungsi sebagai SMS gateway yang bertugas untuk menerima SMS, dan *auto respon SMS*.

$$Score\ Total = \frac{\sum Dokumen + \sum Score\ dokumen}{2} \quad (8)$$

Fungsi dari proses perhitungan NBTS dalam melakukan proses klasifikasi teks yang bersifat multi kelas dibedakan berdasarkan penentuan nilai *threshold* statis dan nilai *threshold* dinamis. Proses perhitungan fungsi *Naive Bayes*

dengan *Two Stage Smoothing* terhadap setiap kelasnya akan menghasilkan nilai probabilitas NBTS setaiap kelas. Setelah itu carilah 2 angka nilai maksimum dari keseluruhan kelas, kemudian ambillah 2 nilai terbesar dari nilai probabilitas NBTS, langkah selanjutnya adalah menentukan nilai *threshold* dengan nilai statis atau dinamis.

Nilai maksimum terhadap suatu fungsi akan dipilih sebagai hasil keputusan. Hasil keputusan bisa menunjukkan data SMS tersebut tergolong dalam satu kelas atau maksimal dua kelas. Dalam menentukan hasil klasifikasi bersifat multi kelas atau tidak maka dibuatlah pendekatan penyelesaian dengan menggunakan batasan atau *threshold*. *Threshold* berfungsi untuk mengukur apakah nilai *Naive Bayes Classification Two Stage (NBCTS)* maksimum ke 2 merupakan bagian dari nilai *Naive Bayes* tertinggi ke 1. Jika nilai *Naive Bayes* lebih besar dari nilai *threshold* yang ditentukan maka hasil klasifikasinya hanya memiliki satu kelas saja, jika nilai *Naive Bayes* terbesar ke 2 lebih kecil atau sama dengan nilai *threshold* maka hasil klasifikasinya memiliki dua kelas atau multi kelas.

Pengaturan nilai *threshold* ada dua cara, yang pertama adalah *threshold* statis artinya penentuan nilai *threshold* ditentukan oleh user dan bersifat tetap atau tidak berubah ketika diuji cobakan terhadap setiap teks, sedangkan yang ke dua adalah *threshold* dinamis dengan penentuan nilai *threshold* bergantung pada rata-rata nilai probabilitas kelas di setiap proses klasifikasinya sehingga sifat *threshold* ini selalu berubah atau selalu berbeda disetiap teks yang di klasifikasikan.

Algoritma 1. Klasifikasi NBTS *Threshold* Statis

[Fungsi : menghitung nilai probabilitas teks]

[input : hasil *stemming* teks data *training* dan *testing*]

[output : klasifikasi kelas]

- 1: Lakukan proses *pre-processing* [*tokenizing, casefolding, convert word, translater, filtering, stemming*]
- 2: Tentukan koefisien kontrol  $\mu$ (miu),  $\lambda$ (lamda), *threshold* dan modifikasi rumus.
- 3: Hitung *prior probability naive bayes* semua kelas.
- 4: Menghitung nilai probabilitas *Two Stage Smoothing* setiap kata terhadap semua kelas.
- 5: Hitung nilai probabilitas setiap kelas.
- 6: Lakukan normalisasi.
- 7: *Sorting* nilai probabilitas kelas.
- 8: Ambil 2 nilai probabilitas tertinggi.
- 9: Periksa nilai probabilitas tertinggi ke 2. Jika nilainya lebih besar dari *threshold* maka output satu kelas, tetapi jika nilainya lebih kecil dari *threshold* maka outputnya multi kelas.

Contoh SMS dengan menggunakan algoritma 1, dengan teks awal “Kami keluarga pasien rsud pamekasan, kami merasa kurang puas dengan pelayanan pengambilan obat di depo farmasi 1 karena kurangnya pegawai, sehingga kami harus nunggu+- 1 jam lebih. Kami mohon pelayanan yg cepet dan memuaskan,ttd ach muchlis. ”. Kemudian lakukan proses *pre-processing*, teks berubah menjadi “keluarga pasien puas layan ambil obat depo farmasi kurang pegawai nunggu jam layan cepet puas muchlis ”. Menentukan koefisien kontrol  $\mu = 0,1$ ,  $\lambda = 0,3$ , *threshold* dan modifikasi rumus. Menghitung nilai

prior probability naive bayes rumus no 1 terhadap semua kelas, diperoleh hasil  $P(\text{Class}=\text{loket pembayaran}) = 68/1574 = 0.0432$ .

Menghitung nilai probabilitas *Two Stage Smoothing* setiap kata terhadap semua kelas, maka hasil yang diperoleh adalah  $P_{\lambda\mu}(\text{word}=\text{keluarga}|\text{class}=\text{tidak terklasifikasi}) = 0.0019$ . Kemudian menghitung nilai probabilitas setiap kelas, maka diperoleh hasil  $P(D|\text{rawat inap}) = 3,7107067445934E-50$ . Lakukan normalisasi hasilnya  $P(\text{Class}=\text{tidak terklasifikasi}) = 1.5870950299765E-14$ . Urutkan nilai probabilitas kelas dari tertinggi hingga terendah dan ambil 2 nilai probabilitas tertinggi, diperoleh kelas Farmasi= $54455863,6174$  dan Rawat Jalan= $1,6049$ . Periksalah nilai probabilitas tertinggi ke 2 untuk menentukan hasil klasifikasi.

Proses disinilah yang membedakan hasil output klasifikasinya. Hal ini dilakukan untuk mengetahui apakah outputnya memiliki satu atau dua kelas. Jika nilai probabilitas lebih besar dari *threshold* maka output single kelas, tetapi jika nilainya lebih kecil dari dari *threshold* maka outputnya multi kelas. Dari contoh diatas nilai probabilitas rawat jalan adalah  $1,6049045186192E-8$  sedangkan *threshold*nya  $0,6$  maka hasilnya satu kelas, artinya Rawat Jalan bukan bagian dari kelas pertama. Kesimpulan dari hasil klasifikasi teks diatas adalah Farmasi.

Aplikasi yang dirancang menunjukkan hasil perhitungan dengan menggunakan *threshold* statis dan *threshold* dinamis. *Threshold* statis bersifat tidak berubah untuk setiap teks yang diklasifikasikan berdasarkan metode NBTS. Sedangkan *threshold* dinamis [13], [14] bersifat selalu berubah secara otomatis untuk setiap teks yang diklasifikasikan berdasarkan nilai NBTS dengan menghitung jarak rata-rata yang paling dominan terhadap nilai NB tertinggi pertama.

Algoritma 2. Multi Klasifikasi *Threshold* Dinamis

[Fungsi : klasifikasi multi kelas dinamis]

[input : nilai probabilitas semua kelas ]

[output : hasil klasifikasi]

- 1: Hitung rata2 nilai probabilitas NB+Smoothing setiap teks.
- 2: Ambil nilai rata2 sebagai nilai *threshold* dinamis.
- 3: Ambil 2 nilai probabilitas tertinggi.
- 4: Bandingkan nilai probabilitas tertinggi ke 2 dengan nilai *threshold*.
- 5: Tampilkan hasil klasifikasi nilai probabilitas ke-1 dan ke-2 jika ada.

Contoh SMS dengan menggunakan algoritma 2, dengan teks awal “perawat cuek, kurang ramah pelayanan lebih diperbaiki lagi khususnya untuk perawat harus lebih perhatikan kondisi dan kebutuhan pasien.”. Kemudian lakukan proses *pre-processing*, teks berubah menjadi “perawat cuek ramah layan baik awat hati kondisi pasien”. Menentukan Koefisien control nilai  $\mu=0,2$  dan  $\lambda =2000$ . Mencari nilai probabilitas *naive bayes* setiap kelasnya, diperoleh  $P(\text{Class}=\text{profesi}) = 0.1910$ . Menghitung nilai probabilitas

setiap kata hasil *stemming* terhadap masing-masing kelas misalnya kata “ramah”, maka diperoleh  $P_{\lambda\mu}(\text{word}=\text{ramah} | \text{class}=\text{tidak terklasifikasi}) = 0.0042$ . Menghitung nilai probalitas seluruh kelas terhadap setiap kata misalnya klasifikasi Profesi,  $P(D|\text{profesi}) = 2,0220 E-20$ . Normalisasi menjadi  $P(\text{Class}=\text{profesi}) = 1.5870$ . Urutkan nilai probabilitas kelas dari tertinggi hingga terendah dan ambil 2 nilai probabilitas tertinggi, maka hasilnya kelas Rawat Inap= $1,4361$  dan profesi= $0,4647$ . Periksalah nilai probabilitas tertinggi ke 2 untuk menentukan hasil klasifikasi.

Proses disinilah yang membedakan hasil output klasifikasinya. Nilai *threshold* disetiap data uji nilainya akan selalu berubah tergantung pada nilai probabilitasnya, maka rata-rata nilai probabilitas= $0,4995$ . Jadikan nilai nilai rata-rata sebagai nilai *threshold* dinamis. Ambil 2 nilai probabilitas tertinggi, maka diperoleh klasifikasi 1 adalah Rawat Inap dengan nilai  $1,4361$  dan klasifikasi ke 2 adalah Profesi dengan nilai  $0,4647$ . Bandingkan nilai probabilitas tertinggi ke 2 dengan *threshold*. Karena nilai probabilitas kelas Profesi lebih kecil dari nilai *threshold* maka kelas tersebut bagian dari kelas pertama atau output dari hasil klasifikasi ini adalah multi kelas. Berdasarkan hasil perhitungan sebelumnya, disimpulkan hasil klasifikasinya adalah Rawat Inap dan Profesi.

Proses awal aplikasi ini dimulai dari melakukan pelabelan manual terhadap data SMS mentah sesuai dengan daftar kelas yang telah ditentukan dan dilanjutkan dengan proses *pre-processing* dari data SMS yang sudah berlabel. Hasil dari tahap *pre-processing* kemudian lakukan pengaturan data training dan data testingnya dipilah berdasarkan persentase 20-50% untuk dilakukan proses uji coba klasifikasi menggunakan NBTS. Hasilnya dievaluasi untuk mengetahui dan membandingkan tingkat akurasi pada proses klasifikasi. Setelah proses pengujian dan evaluasi selesai maka dihitung jumlah kelas terbesar akan mendapatkan rekomendasi berupa surat teguran mengenai pelayanan yang diberikan terhadap masyarakat. Setiap SMS baru yang dikirimkan oleh masyarakat akan dilakukan *pre-processing* dan dilakukan pengklasifikasian dengan menggunakan data training yang didapatkan dari seluruh data SMS hasil *pre-processing* dengan menggunakan *Naive Bayes* dan *TS Smoothing* [9]. Hasil klasifikasi SMS baru digukan untuk meneruskan pesan secara otomatis dari masyarakat kepada penanggung jawab setiap unit yang berkaitan berdasarkan isi SMS yang telah diklasifikasi.

IV. HASIL EKSPERIMEN

Sebelum melakukan uji coba klasifikasi, yang harus dilakukan adalah memilah data SMS dan responden menjadi data *training* dan data *testing*. Pemilihan ini dilakukan 4 variasi data *testing* berdasarkan presentase 20%, 30%, 40%, dan 50% dari keseluruhan data. Pemilihan data testing ini dilakukan secara acak berdasarkan indeknya (kode teks). Untuk setiap pemilihan tersebut dilakukan uji coba klasifikasi menggunakan NBTS. Pemilahan data testing dan data training dalam sistem ditunjukkan pada Tabel 1.



#	Isi Pesan	Hasil Preprocessing	Class	Hasil Class	Skor
46	perawat kalau berkomunikasi harus lebih halus., dokter saja halus kok perawatnya agak kasar	perawat komunikasi halus dokter halus perawat kasar	profesi	profesi	1
47	kalau malam perawat jangan suka tidur, apalgi yang punya tugas malam	malam perawat tidur punya tugas malam	rawat inap, profesi	rawat inap	1
48	perawat harus tanggap terhadap pasien yang lagi sakit di rawat inap	perawat tanggap pasien sakit rawat inap	rawat inap, profesi	profesi, rawat inap	1

Gambar 4. Hasil Klasifikasi Data Testing

TABEL I  
DATA TESTING DAN DATA TRAINING

No	Persentase Data Testing (%)	Jumlah Data	Data Testing	Data Training
1	20	2292	458	1834
2	30	2292	682	1610
3	40	2292	912	1380
4	50	2292	1143	1149

Pada Tabel 1 pemilahan data testing dan data training dilakukan secara dinamis dengan memilih kode teks dalam *database*. Hal ini dilakukan agar dimungkinkan untuk menambah variasi pemilihan data *testing* dan data *training*, dengan ketentuan lebih banyak data *training* dari pada *testing*.

TABEL II  
UJI COBA KLASIFIKASI THRESHOLD STATIS

No	Data uji	$\lambda$	$\mu$	Threshold	Akurasi	Rata2
1	458	0,2	2000	3	85,22%	
2	458	0,2	0,5	3	81,25%	
3	458	0,3	1000	2	84,56%	
4	458	0,2	0,5	2	80,14%	82,79%
5	682	0,2	2000	3	85,48%	
6	682	0,2	0,5	3	82,11%	
7	682	0,3	1000	2	83,28%	
8	682	0,2	0,5	2	81,23%	83,03%
9	912	0,2	2000	3	86,73%	
10	912	0,2	0,5	3	82,02%	
11	912	0,3	1000	2	85,09%	
12	912	0,2	0,5	2	80,70%	83,64%
13	1143	0,2	2000	3	85,91%	
14	1143	0,2	0,5	3	80,66%	
15	1143	0,3	1000	2	83,55%	
16	1143	0,2	0,5	2	79,70%	82,46%

Pada Tabel 2 menunjukkan hasil uji coba klasifikasi menggunakan metode NBTS dengan pengaturan *threshold* statis. Koefisien lamda, miu dan *threshold* disesuaikan nilainya untuk mendapatkan nilai akurasi yang tinggi, pada beberapa percobaan akan diambil akurasi yang tertinggi di setiap presentase data testing, dalam hal ini no uji coba ke 1, 5, 9 dan 13 adalah hasil akurasi tertinggi.

Pada Tabel 3 menunjukkan hasil uji coba klasifikasi menggunakan metode NBTS dengan pengaturan *threshold* dinamis. Pada Tabel 2 kolom *threshold* bernilai 0 artinya nilai

ini memiliki nilai *threshold* yang selalu berubah tergantung nilai probabilitas masing-masing teks yang akan diklasifikasikan.

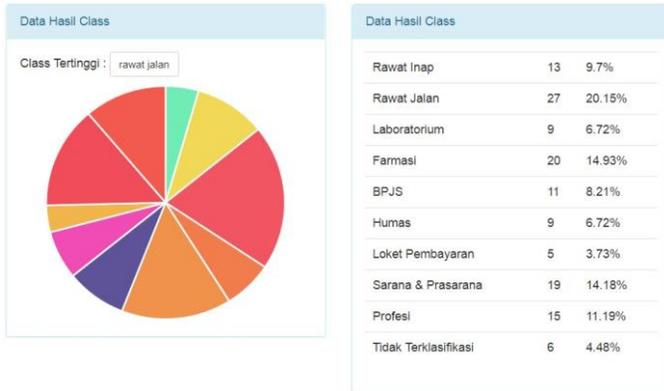
TABEL III  
UJI COBA KLASIFIKASI THRESHOLD DINAMIS

No	Data uji	$\lambda$	$\mu$	Threshold	Akurasi	Rata2
1	458	0,1	3000	0	77,49%	
2	458	0,2	900	0	76,61%	
3	458	0,1	0,3	0	74,83%	
4	458	0,2	0,3	0	75,07%	76,00%
5	682	0,1	3000	0	80,06%	
6	682	0,2	900	0	76,98%	
7	682	0,1	0,3	0	76,25%	
8	682	0,2	0,3	0	76,10%	77,35%
9	912	0,1	3000	0	80,59%	
10	912	0,2	900	0	78,95%	
11	912	0,1	0,3	0	76,21%	
12	912	0,2	0,3	0	76,21%	77,99%
13	1143	0,1	3000	0	77,87%	
14	1143	0,2	900	0	77,17%	
15	1143	0,1	0,3	0	74,28%	
16	1143	0,2	0,3	0	74,72%	76,01%

Pada Gambar 4 hasil klasifikasi data *testing* menunjukkan isi Pesan merupakan field untuk menampung isi pesan, hasil dari proses pre-processing di tampilkan pada kolom hasil preprocessing, proses pelabelan yang dilakukan oleh pakar hasilnya ditunjukkan pada kolom *Class* sedangkan hasil proses dari perhitungan klasifikasi multi kelas ditunjukkan pada kolom Hasil *Class*, kemudian hasil proses *scoring* ditunjukkan pada Skor. Pada pojok kanan atas juga disediakan filter yang berfungsi untuk menampilkan hasil klasifikasi berdasarkan seluruh data atau berdasarkan multi label-nya atau berdasarkan hasil multi klasifikasi-nya. Ada dua tombol proses yaitu tombol a untuk melihat detail perhitungan proses klasifikasi NBTS dan tombol b untuk melihat hasil detail perhitungan proses *SCORING NBTS*.

Pada Gambar 5 disajikan contoh SMS baru yang telah diklasifikasikan menggunakan *NBTS*. Fitur ini akan digunakan setelah dilakukan sinkronisasi antara aplikasi ini dengan Aplikasi SMS Center RSUD yang sudah ada sebelumnya ditampilkan sebelah kanan berupa tabel statistik hasil klasifikasi dari data SMS berupa presentase masing-masing kelas dan tampak sebelah kiri ada grafik *pie* yang

menggambarkan luasan persentase kelas yang ada. Terdapat skoring tertinggi yang mendapatkan SMS terbanyak dan skor berdasarkan tingkat keluhan, sehingga dalam hal ini bisa direkomendasikan mendapatkan surat peringatan terhadap penanggung jawab Rawat Jalan.



Gambar 5. Grafik hasil Klasifikasi SMS baru

Informasi kelas tertinggi yang terletak di atas gambar grafik, juga berfungsi untuk mencetak surat peringatan terhadap penanggung jawab unit berdasarkan hasil skoring terbesar yang diperoleh dari rata rata jumlah klasifikasi ditambah total skor masing-masing SMS.

TABEL IV  
ANALISA AKURASI UJI COBA KLASIFIKASI

No	Threshold	1	2	3	4	Rerata (%)
1	Manual	82,78%	83,03%	83,64%	82,46%	82,97%
2	Dinamis	75,99%	77,35%	77,99%	76,01%	76,84%
					Selisih	6,14%

Pada Tabel 4 dapat diketahui rata-rata akurasi uji coba klasifikasi menggunakan *Naïve Bayes* dengan *TS Smoothing* menggunakan *threshold* Statis(Manual) 82,97%, sedangkan rata-rata akurasi uji coba klasifikasi menggunakan *Naïve Bayes* dengan *TS-Smoothing* menggunakan *threshold* Dinamis sebesar 76,84%. Terbukti klasifikasi menggunakan *NBTS threshold* Statis dapat meningkatkan akurasi klasifikasi sebesar 6,14% apabila dibandingkan dengan klasifikasi *NBTS threshold* Dinamis. Besar kecilnya akurasi klasifikasi sangat dipengaruhi oleh terjadinya kesalahan klasifikasi.

TABEL V  
STATISTIK DATA UJI COBA KLASIFIKASI

No	Class	Jumlah Data Testing
1	Rawat Inap	122
2	Rawat Jalan	74
3	Laboratorium	8
4	Farmasi	35
5	BPJS	36
6	Humas	67
7	Loket Pembayaran	25
8	Sarana dan Prasarana	172
9	Profesi	116
10	Tidak Terklasifikasi	27

TABEL VI  
STATISTIK DATA UJI COBA KLASIFIKASI

No	Medode NBTS	Klasifikasi Benar	Klasifikasi Salah
1	Threshold statis	583	99
2	Threshold dinamis	546	136

Uji coba klasifikasi dilakukan dengan data latih 30% dari total data sms dan responden sebesar 682 data testing pada Tabel 1 nomor 2. Pemilahan data testing dilakukan secara dinamis oleh sistem dengan cara mengambil kode teks secara acak pada masing-masing kelas sebesar 30%. Statistik jumlah data testing bisa dilihat pada Tabel 5. Selisih Jumlah data testing antar kelas sangat besar atau tidak seimbang, sehingga berpengaruh pada hasil uji coba klasifikasi lihat Tabel 6, hasil klasifikasi salah semakin besar.

Pada Tabel 7 bagian ini akan dipaparkan hasil uji metode dari semua *smoothing* yang ada. Tahap pengujian ini menggunakan data testing baru sebesar 20% dari keseluruhan data.

TABEL VII  
AKURASI SEMUA SMOOTHING

No	Smoothing	Koefisien Kontrol	Threshold	Akurasi (%)
1	Laplace	-	2	85,65
2	Jelinek-Mercer(JM)	$\lambda=0.1$	2	86,75
3	Dirichlet (DIR)	$\mu=0.1$	2	88,30
4	Absolute Discounting (DS)	$\delta=0.1$	2	83,22
5	Two Stage (TS)	$\lambda=0.1, \mu=3000$	0	88,52

Berdasarkan hasil uji coba bisa disimpulkan Laplace smoothing, Dir Smoothing, JM Smoothing dan AD smoothing akurasinya masih dibawah TS-Smoothing.

## V. KESIMPULAN

Adapun beberapa kesimpulan yang didapatkan dalam penelitian ini, antara lain:

1. Penelitian ini telah berhasil mengklasifikasikan dokumen dalam bentuk teks dengan baik menggunakan metode *naive bayes* dengan *TS-Smoothing*.
2. Pada penelitian ini rata-rata akurasi klasifikasi dengan menggunakan *Naïve Bayes TS-Smoothing dan threshold Statis* sebesar 82,97%, sedangkan rata-rata akurasi klasifikasi dengan *threshold Dinamis* sebesar 76,84%. Maka *Naïve Bayes* dengan *TS-Smoothing* mampu untuk memecahkan permasalahan pada multi klasifikasi.
3. Metode *Two Stage Smoothing* dengan *threshold statis* terbukti meningkatkan akurasi klasifikasi sebesar 6,14%.
4. Pada uji coba klasifikasi *threshold statis* diperoleh pengujian akurasi terendah sebesar 80,13% dan akurasi tertinggi sebesar 86,73%.
5. Pada uji coba klasifikasi *threshold dinamis* diperoleh pengujian akurasi terendah sebesar 74,28% dan akurasi tertinggi sebesar 80,59%.
6. Rekomendasi surat evaluasi internal diberikan kepada unit dengan skoring terbesar, tidak hanya berdasarkan jumlah SMS yang diterima di setiap unit.

7. Kesalahan klasifikasi sering disebabkan oleh tidak seimbang jumlah SMS di setiap kelas pada data training.
  8. TS-Smoothing lebih baik dari segi akurasi dibandingkan dengan metode yang lainnya.
- Minat penelitiannya adalah bidang *Artificial Intelligence, Computer Vision, Data Mining*.

#### DAFTAR PUSTAKA

- [1] G. Malik, M. Tarique, "On Machine Learning Techniques For Multi-class Classification", *International Journal of Advancements in Research & Technology*, vol.3, no.2, Feb 2014.
- [2] R. Adipratama, R.A. Hendrawan, R.P. Kusumawardani, "Klasifikasi SMS Pada Sistem Support Ticket Pengaduan Pelanggan Dengan Menggunakan Metode Naïve Bayes", *Undergraduate Thesis of Information System: ITS Library*, 2011.
- [3] Badar Said, Yuliana Melita P, "Klasifikasi Data SMS Center Bupati Pamekasan Menggunakan Naïve Bayes Dengan Mad Smoothing", *In IdeaTech*, pp.92-99, 2015.
- [4] Q Yuan, G Cong, N.M. Thalmann, "Enhancing naive bayes with various smoothing methods for short text classification," in *WWW 12 Companion*, ed. New York: ACM, 2012, pp.645-646.
- [5] Shruti Aggarwal, Devinder Kaur, "Naïve Bayes Classifier with Various Smoothing Techniques for Text Documents", *IJCTT*, 2013.
- [6] D.D. Arifin, Shaufiah, M.A. Bijaksana, "SMS Filtering Menggunakan Naive Bayes Classifier dan FP-Growth Algorithm Frequent Itemset", *e-Proceeding of Engineering*, vol.2, no.2, Agustus 2015.
- [7] Afrianto, Kurniawati, "Kategorisasi Dokumen Teks secara Multi Label Menggunakan Fuzzy C-Means dan K-Nearest Neighbors pada Artikel Berbahasa Indonesia", *JUTI*, 2013.
- [8] Tsoumakas, G., & Katakis, I., "Multilabel classification: An overview", *International Journal of Data Warehousing and Mining*, vol.3, no.3, pp. 1-13, 2007.
- [9] Zhihua Wei, Hongyun Zhang, Zhifei Zhang, Wen Li, Duoqian Miao. "A Naive Bayesian Multi-label Classification Algorithm With Application to Visualize Text Search Results". *International Journal of Advanced Intelligence*, vol.3, No.2, pp. 173-188, July 2011.
- [10] R. K. Hapsari, Y. J. Santoso, "Stemming Artikel Berbahasa Indonesia dengan Pendekatan *Confix-stripping*", in *Prosiding Seminar Nasional Manajemen Teknologi XXII*, ed. Surabaya: Program Studi MMT-ITS, pp.251-258, Jan 2015.
- [11] M.K. Dalal, M.A. Zaveri, "Automatic Text Classification: A Technical Review", *International Journal of Computer Applications*, vol.28, pp.0975-8887, Agst. 2011.
- [12] Gilang Jalu Selo W.T, Budi Susanto, Rosa Delima, "Implementasi Naïve Bayesian Classifier Untuk Kasus Filtering SMS Spam", *Informatika*, 2013.
- [13] Arifin, A.Z., I.P.A.K. Mahendra dan H.T. Ciptaningtyas, "Enhanced Confix Stripping Stemmer and Ants Algorithm for Classifying News Document in Indonesian Language". *ICTS*, 2009.
- [14] Karl-Michael Schneider, *Techniques for Improving the Performance of Naive Bayes for Text Classification*, citeseerx, 2005.

**Sholeh Rachmatullah** lahir di kota Pamekasan, Jawa Timur, pada tahun 1985. Menyelesaikan studi S1 di program studi Teknik Informatika Universitas Trunojo pada tahun 2007. Sholeh menyelesaikan studinya pada jurusan Pasca Sarjana Teknologi Informasi STTS pada tahun 2017. Sekarang sebagai Dosen program studi Teknik Informatika Universitas Madura. Minat penelitian adalah data mining, rekayasa perangkat lunak dan sistem informasi.

**Reddy Alexandro Harianto** lahir di Surabaya, Jawa Timur, Indonesia, pada tahun 1989. Dia menyelesaikan studi S1 di program studi Teknik Informatika Sekolah Tinggi Teknik Surabaya pada tahun 2011, dan menyelesaikan studinya pada jurusan Teknologi Informatika di Institut Teknologi Sepuluh Nopember Surabaya pada tahun 2015.