Secret Handshake Pada Tuas Pintu Dengan *Limit switch* Menggunakan Metode Klasifikasi Naïve Bayes

Yahya Ermaya[1], Aji Gautama Putrada[2], Siti Amatullah Karimah[3]

[1,2,3]Fakultas Informatika, Universitas Telkom, Bandung

[1]yahyaermaya@student.telkomuniversity.ac.id, [2]ajigps@telkomuniversity.ac.id,

[3] karimahsiti@telkomuniversity.ac.id

Abstrak

Seiring berkembangnya ilmu pengetahuan, sudah banyak cara untuk membobol kunci pintu rumah tradisional. Dengan ini, dibutuhkan kunci pintu baru untuk mengamankan rumah. Dalam penelitian ini dirancang kunci pintu baru berupa *smart lock* yang bernama *secret handshake* dengan menggunakan pergerakan tuas pintu sebagai *password* untuk membuka kunci pintu. *Secret handshake* menggunakan sensor *limit switch* dan metode klasifikasi naïve bayes untuk mengklasifikasi data yang dihasilkan oleh *limit switch*. Metode *Naïve Bayes* dipilih karena hanya membutuhkan data *training* dengan jumlah yang kecil. Sistem bekerja dengan membaca sensor *limit switch* lalu dikirimkan ke matlab melalui thingspeak untuk diproses menggunakan metode klasifikasi *naïve bayes* untuk menghasilkan nilai prediksi kebenaran *password* yang dimasukkan. Hasil penelitian yang dilakukan diketahui metode klasifikasi *naïve bayes* dapat membedakan *password* yang benar dan *password* yang salah dengan tingkat akurasi sebesar 93.33%.

Kata Kunci: ***smart lock, limit switch, secret handshake*.**

Abstract

Along with the development of knowledge, there are a lot of ways to break traditional house-key. Therefore, new house-key is needed to protect the house. In this research, new house-key with smart lock system named secret handshake is designed, using door handle movements as a password to open the house-key. Limit switch and naive bayes classification method are used by secret handshake system to classify the data from limit switch movements. Naive bayes method is choosen because it is only use small amounts of training data. Secret handshake system works when the limit switch read the movements and then the data sent to matlab through thingspeak to be processed using naive bayes classifier method and get the true password prediction values that has been input. The result showed that naive bayes cassifier method can differentiate between the true and wrong password with 93.33% accuracy.

Keywords: ***smart lock, limit switch, secret handshake*.**

1. Pendahuluan

Latar Belakang

Pada masa kini sistem keamanan sudah menjadi hal yang biasa dan menjadi suatu kewajiban untuk diterapkan pada suatu rumah. Hal ini sangat penting karena bertujuan untuk menjaga keamanan rumah. Seiring berkembangnya pengetahuan, sudah banyak tersebar cara untuk membobol kunci pintu rumah tradisional. Hal ini perlu mendapatkan perhatian karena keamanan rumah sudah tidak terjaga seperti sedia kala. Untuk mencegah hal tersebut terjadi, maka diperlukan perbaikan sistem keamanan untuk rumah. [1] Sistem keamanan rumah yang sudah banyak dipakai adalah alarm rumah. Namun walaupun adanya sistem keamanan alarm rumah, masih banyak orang yang dapat mengelabui sistem keamanan tersebut.

Untuk mengatasi masalah tersebut, diperlukan sistem keamanan tambahan baru yang belum dikenal oleh banyak orang untuk mengamankan rumah pintu berupa *smart lock* yang bernama *secret handshake* dengan menggunakan pergerakan tuas pintu sebagai *password* untuk membuka kunci pintu. *Secret handshake* menggunakan sensor *limit switch* dan *microcontroller* NodeMCU yang juga bekerja sebagai perantara komunikasi dengan internet. Untuk membedakan *password* yang benar dan *password* yang salah digunakan metode klasifikasi *naïve bayes* karena [2] hanya menggunakan jumlah data *training* yang sedikit.

Topik dan Batasannya

Berdasarkan latar belakang, dapat diidentifikasi masalah yang diantaranya bagaimana sistem *secret handshake* dapat bekerja sebagai sistem keamanan tambahan serta performansi metode klasifikasi *naïve bayes* dalam mebedakan *password* yang benar dan *password* yang salah.

Dari identifikasi masalah yang sudah dijelaskan, terdapat batasan-batasan masalah dalam penelitian yaitu pintu merupakan pintu bertuas dan tertutup, bukan seperti pintu pagar dan data yang dikirim dan diterima oleh ThingSpeak hanya 20 detik sekali karena tidak menggunakan akun ThingSpeak berbayar..

Tujuan

Tujuan berdasarkan identifikasi masalah dari penelitian ini yaitu merancang dan membuat perangkat sistem *secret handshake* dengan menerapkan metode klasifikasi naive bayes untuk menambahkan sistem keamanan. Tujuan pada penelitian ini dapat dilihat pada penjelasan pada **Tabel 1** yaitu:

Tabel 1. Keterkaitan antara tujuan, pengujian dan kesimpulan.

No	Tujuan	Pengujian	Kesimpulan
1	Merancang dan membuat perangkat sistem *secret handshake*.	Menjalankan fungsi perangkat dalam mendeteksi gerakan tuas pintu dan dapat mengunggah datanya ke platform *ThingSpeak* dan mengunduh kembali hasil klasifikasi.	Perangkat berfungsi sesuai yang diharapkan dan dapat mengunggah data ke platform ThingSpeak dan mengunduh kembali hasil klasifikasi.
2	Menerapkan metode klasifikasi *naive bayes* menggunakan data yang tersimpan pada *platform ThingSpeak*.	Menguji metode klasifikasi naive bayes dengan data baru untuk mengetahui kondisi penggunaan listrik	Hasil prediksi kebenaran *password* muncul.
3	Menganalisis performansi metode klasifikasi *naive bayes*.	Menggunakan evaluasi *confusion matrix* untuk mendapatkan tingkat akurasi antara hasil prediksi dengan penggunaan listrik.	Tingkat akurasi yang didapatkan cukup tinggi.

Organisasi Tulisan

Penulisan bab pertama membahas mengenai masalah dan batasan serta tujuan dari penelitian ini. Pada bab dua dbahas studi literatur yang digunakan sebagai bahan informasi dan referensi untuk perancangan sistem *secret handshake*. Pada bab tiga dijelaskan rancangan sistem secara umum dan analisis kebutuhan pada sistem *secret handshake*. Penulisan pada bab empat dijelaskan hasil pengujian dan analisis sistem *secret handshake*. Untuk penulisan bab lima dijelaskan hasil kesimpulan selama penelitian berlangsung dan saran yang diberikan.

2. Studi Terkait

[3] Penelitian ini terinspirasi oleh penelitian yang dilakukan oleh Michael A. Mahler, Qinghua Li, dan Ang Li dengan judul "*SecureHouse: A Home Security System Based on Smartphone Sensors*". Dalam penelitian tersebut, telah berhasil mengimplementasikan sensor *accelerometer* dan *magnetometer* yang terdapat pada *smartphone* untuk mendeteksi getaran dan pergerakan pintu. Mereka menggunakan metode prediksi *FeedForward* Neural Network (FFNN) untuk memprediksi getaran *smartphone*.

[4] Dalam penelitian Rahmat Ibrahim dengan judul "Rancang Bangun *Smart Lock System* untuk *Tanker*", telah berhasil membuat sebuah pintu otomatis dengan menggunakan suatu kode yang telah ditentukan sebagai *input*. Jika benar maka pintu akan terbuka dan sebaliknya apabila kode yang dimasukan salah maka *buzzer* aktif. Mikrokontroller yang digunakan adalah Arduino Uno. Bagian mekanik yang digunakan yaitu berupa solenoida. *Output* dari sistem tersebut berupa tampilan pada LCD. Sistem ini digunakan sebagai sistem keamanan rumah agar tidak ada lagi pembuatan kunci ganda oleh pihak yang tidak diinginkan.

[5] Dalam penelitian Muhammad Izzuddin Mahali dengan judul "*Smart Door Locks Based on Internet of Things Concept with Mobile Backend As A Service*",telah berhasil menggabungkan teknologi dengan menggunakan ESP8266, *Firebase* dan aplikasi android. ESP8266 sebagai modul Wi-Fi juga berperan sebagai peralatan kontrol yang dapat memutar motor yang terdapat pada pintu dan berfungsi sebagai kunci. Pada penelitian tersebut, proses *programming* ke ESP8266 menggunakan bantuan board Arduino UNO. Modul ESP8266 mampu membaca keadaan database dan melaksanakan perintah sesuai dengan data yang ada. Penelitian tersebut sudah termasuk dalam *Internet of Things*.

[2] Pada penelitian ini dibahas mengenai metode klasifikasi *naive bayes* dalam memprediksi besarnya penggunaah listrik rumah tangga. Pada penelitian tersebut dijelaskan mengenai mekanisme pengimplementasian metode naive bayes dengan memanfaatkan parameter yang menjadi pendukung dalam memprediksi. Hasil dari penelitian tersebut didapat metode klasifikasi naive bayes dapat memprediksi dengan tingkat akurasi lebih dari 75%.

2.1 Metode Naive Bayes

Metode Naive Bayes adalah metode klasifikasi probabilistik yang sederhana dengan menghitung sekumpulan probabilistik dengan menjumlahkan nilai frekuensi dan kombinasi nilai dataset. Keuntungan menggunakan metode naive bayes yaitu hanya memerlukan data training skala kecil untuk mengestimasi parameter yang diperlukan pada proses pengklasifikasian[2]. Berikut persamaan teorema naive bayes menurut Bustami (Saleh, 2015:209) [2]:

$$P(H|X)= \frac{P(X|H)\cdot P(H)}{P(X)} \quad (1)$$

H : Hipotesis berupa kelas tertentu
X : Data prediksi yang belum masuk kedalam kelas
$P(H|X)$: Probabilitas terjadinya H berdasarkan kondisi X diketahui.
$P(X|H)$: Probabilitas terjadinya X berdasarkan kondisi pada H
P(H) : Probabilitas hipotesis H.
P(X) : Probabilitas X

Penjelasan lanjut terkait persamaan klasifikasi naive bayes terdapat proses analisis dengan menentukan kelas yang sesuai terhadap sampel. Proses klasifikasi dapat disederhanakan dalam persamaan seperti berikut [2]:

$$Posterior = \frac{Prior \times Likelihood}{Evidence}. \quad (2)$$

2.2 Evaluasi Performansi

Pengukuran akurasi menggunakan *confusion matrix* dengan ukuran evaluasi *True Positive*, *True Negative*, *False Positive* dan *False Negative*. Untuk mengukur akurasi menggunakan rumus berikut:

$$Accuracy = \frac{TP+TN}{P+N} \quad (3)$$

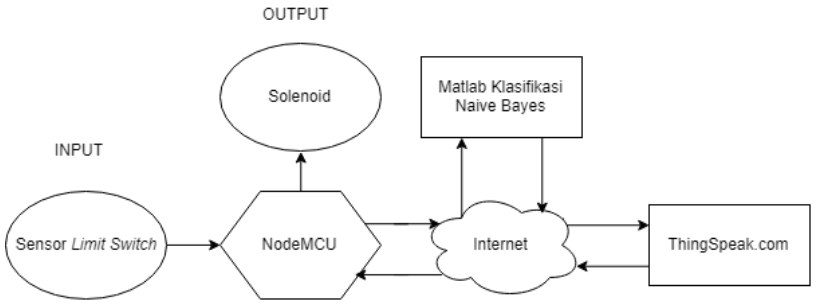
Pengukuran akurasi untuk klasifikasi diterapkan pada *machine learning*[6].*True Positive* adalah label positif yang dilabeli benar sedangkan *True Negative* adalah label negatif yang dilabeli benar, P adalah jumlah sampel positif sedangkan N adalah jumlah sampel negatif.

$$Precision = \frac{TP}{TP+FP} \quad (4)$$

Selain akurasi, dilakukan pengukuran untuk nilai *precision* untuk menentukan tingkat ketepatan antara hasil keluaran sistem dengan informasi yang benar dan *recall* untuk mengetahui tingkat keberhasilan dalam mengembalikan informasi.

$$Recall = \frac{TP}{P}. \quad (5)$$

3. Sistem *Secret Handshake*



Gambar 1. Gambaran Umum Sistem.

Pada **Gambar 1.** dijelaskan bahwa sistem *secret handshake* memiliki keluaran berupa pergerakan solenoid yang berfungsi sebagai kunci pintu. Sensor yang digunakan adalah sensor *limit switch* yang berfungsi untuk mendapatkan data dari gerakan tuas pintu. NodeMCU digunakan sebagai *microcontroller* dan juga perantara komunikasi dengan internet. Data yang didapatkan dari *limit switch* akan disimpan di ThingSpeak kemudian akan dikirimkan ke MATLAB untuk diproses menggunakan metode kalsifikasi *naïve bayes*. Hasil klasifikasi *naïve bayes* akan dikirimkan kembali ke ThingSpeak kemudian dikirimkan ke NodeMCU. Jika hasil klasifikasi *password* benar, maka solenoid akan terbuka. Sebaliknya, jika hasil klasifikasi *password* salah maka solenoid tidak akan terbuka.

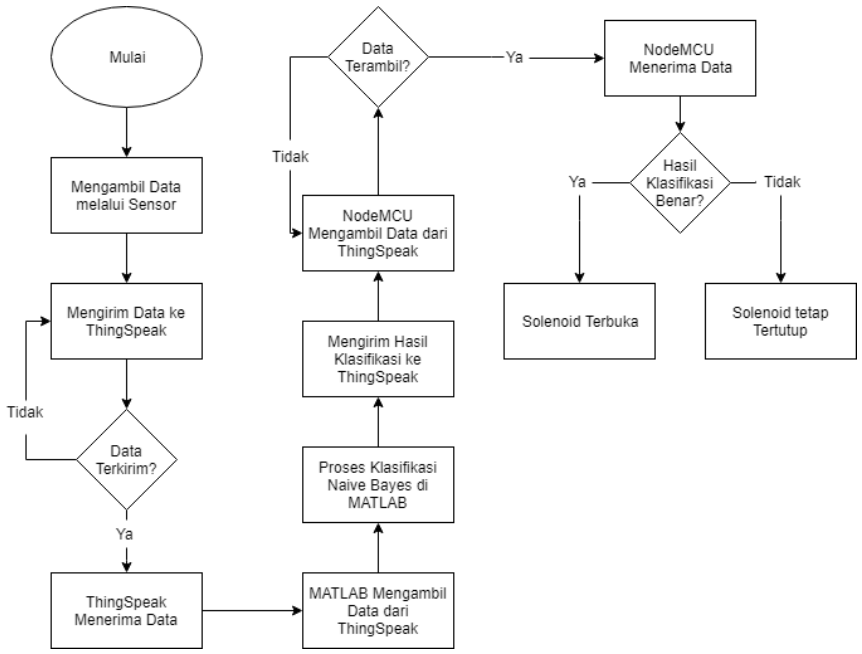
3.1 Kebutuhan Sistem

Sistem *Secret Handshake* memiliki fungsionalitas sebagai berikut:

- Sistem mampu mendeteksi pergerakan tuas pintu.
- Sistem dapat terintegrasi dengan ThingSpeak dan MATLAB untuk mengolah nilai *input* dari sensor dengan metode klasifikasi *naive bayes*.
- Sistem dapat mengeluarkan *state solenoid* berdasarkan hasil klasifikasi.

3.2 Alur Diagram pada Sistem

Terdapat alur diagram sistem *secret handshake* yang menjelaskan cara kerja sistem dengan proses secara bertahap. Proses dari alur diagram sistem dapat dilihat pada **Gambar 2.**:



Gambar 2. Alur Diagram Sistem *Secret Handshake.*

3.2 Spesifikasi Sistem

Pada penelitian Sistem *secret handshake*, peneliti menggunakan perangkat keras dan perangkat lunak dengan fungsionalitas yang dijelaskan pada **Tabel 2** berikut:

Tabel 2. Spesifikasi Sistem.

Jenis Perangkat	Fungsionalitas
NodeMCU	Menerima data sensor dan mengiriman data sensor ke server menggunakan WiFi.
Sensor *LimitSwitch*	Saklar yang tersambung pada perangkat sistem untuk pengambilan data.
Arduino IDE	Perangkat lunak yang digunakan untuk *coding* dalam pengambilan dan pengolahan data sensor.
ThingSpeak IoT Platform	*Platform* IoT yang digunakan untuk penyimpanan data sensor di internet.
MATLAB	Perangkat lunak untuk penerapan metode klasifikasi naive bayes.

3.3 Data pada Sistem

Data penelitian didapatkan dengan menggunakan sebuah sensor *limit switch*. Data tersebut berupa rangkaian pergerakan tuas pintu selama 3 detik. Data yang digunakan untuk data *training* sebanyak 135 rangkaian data. Untuk data *testing* diambil sebanyak 30 rangkaian data baru. Data-data tersebut akan diolah dengan metode klasifikasi *naive bayes*.

3.4 Skenario Pengujian

Pada penelitian ini, peneliti akan menguji fungsionalitas pada sistem dan menganalisis performansi dari metode naive bayes yang diterapkan. Peneliti melakukan pengujian dengan skenario sebagai berikut:

1. Pengujian rangkaian sistem *secret handshake*.
   Peneliti menggunakan sebuah sensor *limit switch* untuk dihubungkan ke mikrokontroller NodeMCU kemudian dilakukan pengaturan agar semua sensor dapat terbaca dan data yang dibaca dapat diunggah ke *ThingSpeak*.
2. Pengujian metode klasifikasi *naive bayes*.
   Peneliti menggunakan data training yang sudah didapatkan dan menggunakan data tersebut untuk proses *learning* dengan metode klasifikasi *naive bayes*. Kemudian data *testing* akan diujikan dengan hasil dari klasifikasi tersebut untuk menghasilkan nilai prediksi kebenaran *password* yang diunggah ke ThingSpeak.
3. Menganalisis performansi metode klasifikasi naive bayes.
   Untuk mengetahui performansi dari metode klasifikasi naive bayes, digunakan evaluasi *confusion matrix* sebagai alat ukur untuk mengetahui performansi berdasarkan tingkat akurasi,*precision*,dan *class recall*.

4. Evaluasi

4.1 Hasil Pengujian Rangkaian Sistem *Secret Handshake*

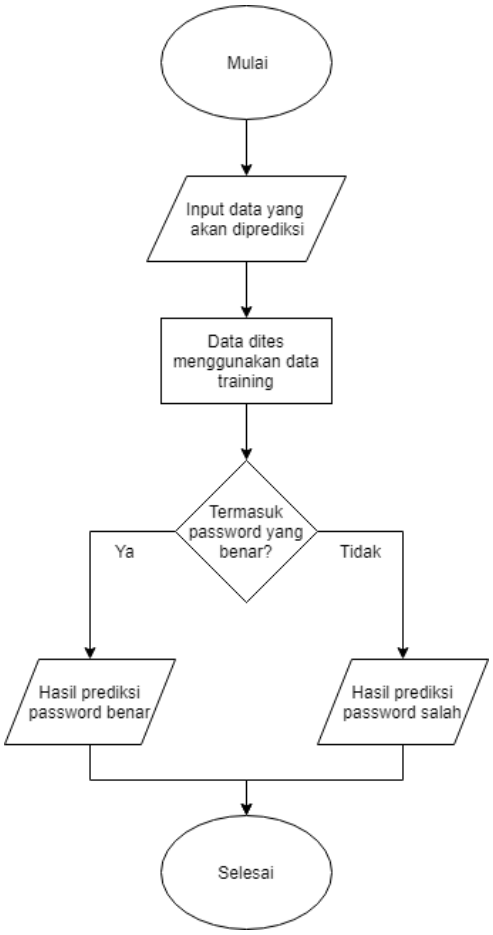
Pada pengujian ini digunakan perangkat lunak Arduino IDE untuk pengambilan data sensor dan mengirimkan datanya ke ThingSpeak. Sensor yang diuji adalah sensor *limit switch*. Pengujian berhasil dilakukan dengan data sensor terunggah pada *platform ThingSpeak*. Hasil pengujian dapat dilihat pada **Gambar 4**:

Gambar 4. Upload Data ThingSpeak.

Field 1 adalah gerakan pertama dari rangkaian *password* yang didapat dari sistem sedangkan Field 2 adalah gerakan kedua dan seterusnya sampai Field 8. Data bernilai 1 saat tuas pintu digerakkan ke bawah dan data bernilai 0 saat tuas pintu digerakkan ke atas.

4.2 Proses Pengujian Metode Klasifikasi Naive Bayes

Data rangkaian *password* diujikan dan diprediksi hasilnya menggunakan metode klasifikasi *naïve bayes*. Proses pengujiannya dapat dilihat pada **Gambar 5** berikut :

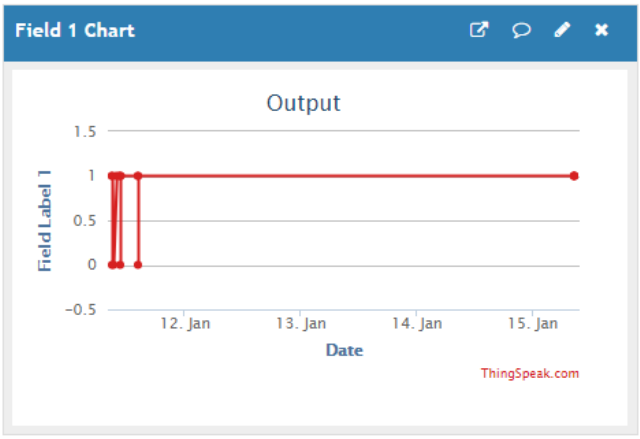


Gambar 5. Alur Klasifikasi *Naive Bayes*.

Hasil prediksi *password* yang benar akan bernilai 1 sedangkan hasil prediksi *password* yang salah akan bernilai 0. Setelah hasil prediksi keluar akan langsung diunggah ke ThingSpeak.

4.3 Hasil Pengujian Metode Klasifikasi Naive Bayes

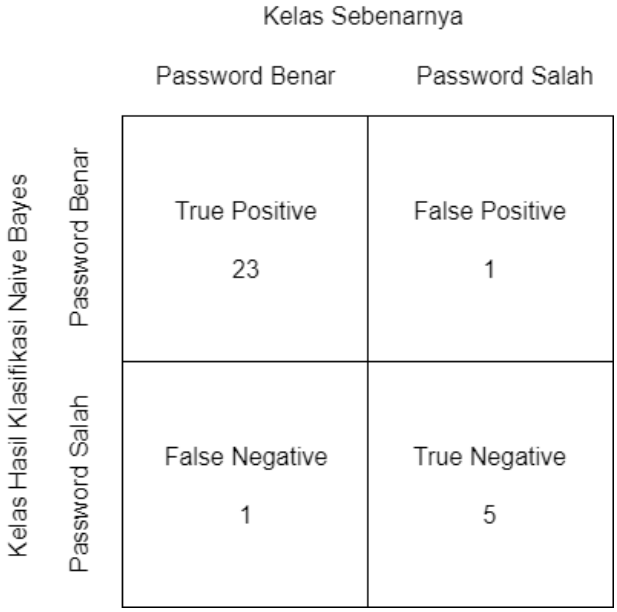
Hasil pengujian dari klasifikasi naive bayes menggunakan data testing menghasilkan nilai prediksi. Hasil prediksi akan ke ThingSpeak dalam bentuk grafik pada **Gambar 5** berikut :



Gambar 5. Prediksi Hasil Klasifikasi *Password.*

Data bernilai 1 adalah *password* yang tergolong benar oleh metode *naïve bayes*, sedangkan data bernilai 0 saat *password* tergolong salah.

4.4 Hasil Pengujian dan Analisis Performansi Metode Naive Bayes

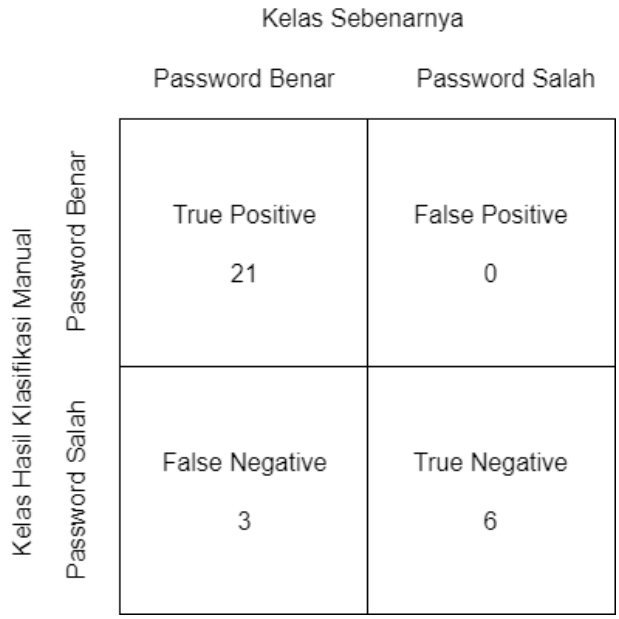


Gambar 6. *Confusion Matrix Naïve Bayes*.

Metode naive bayes berhasil mendeteksi *password* yang benar dan *password* yang salah dengan ketepatan sebesar 93.33%. Nilai Akurasi terhitung dimana nilai yang di prediksi mendekati dengan nilai sebenarnya. Dari 24 *password* yang benar sistem mendapatkan 23 *password* yang tergolong benar dan dari 6 *password* yang salah sistem mendapatkan 5 *password* yang tergolong salah sehingga nilai *accuracy*- nya adalah $\left(\frac{23+5}{24+6}\right) * 100\% = 93.33\%$.

Precision merupakan tingkat ketepatan antara informasi yang benar dengan hasil yang diberikan sistem. Terdapat 24 *password* yang benar, sistem mendapatkan 23 *password* benar dan dari 6 *password* yang tergolong salah, sistem mendapatkan 1 *password* yang tergolong benar, sehingga nilai *precision* yang didapatkan adalah $\left(\frac{23}{23+1}\right) * 100\% = 95.83\%$.

Class Recall merupakan tingkat keberhasilan sistem dalam mendeteksi jumlah benarnya kelas hasil keluaran sistem dengan kelas yang sebenarnya. Hasil dari sistem menunjukkan bahwa dari 24 data yang tergolong dalam *password* yang benar, sistem mendeteksi 23 data yang termasuk dalam *password* yang benar sehingga nilai *recall* adalah $\left(\frac{23}{24}\right) * 100\% = 95.83\%$.



Gambar 7. *Confusion Matrix* tanpa *Naïve bayes*.

Tanpa menggunakan metode klasifikasi *naive bayes*, didapatkan 21 *password* yang tergolong benar dari 24 total *password* yang seharusnya benar dan didapat 6 *password* yang tergolong salah dari total 6 *password* yang seharusnya salah, sehingga nilai *accuracy*-nya adalah $\left(\frac{21+6}{24+6}\right) * 100\% = 90\%.$

Dari 24 *password* yang seharusnya benar, sistem mendapatkan 21 *password* yang tergolong benar dan tidak ada *password* yang tergolong benar tetapi seharusnya salah, sehingga nilai *precision* yang dihasilkan adalah $\left(\frac{21}{21+0}\right) * 100\% = 100\%.$

Dari 24 *password* yang seharusnya benar, sistem mendapatkan 21 *password* yang tergolong benar. Maka dapat disimpulkan bahwa nilai *recall* yang dihasilkan adalah $\left(\frac{21}{24}\right) * 100\% = 87.5\%.$

Dari perbandingan akurasi menggunakan metode klasifikasi *naïve bayes* dan akurasi tanpa klasifikasi *naïve bayes* dapat disimpulkan bahwa menggunakan metode klasifikasi *naïve bayes* lebih baik dalam segi kedekatan nilai hasil prediksi dengan nilai sebenarnya.

5. Kesimpulan

Kesimpulan yang didapat dari penelitian ini adalah metode *naive bayes* memiliki akurasi yang baik dan dapat diterapkan pada sistem *secret handshake* dengan tingkat akurasi sebesar 93.33%. Terdapat selisih 3.33% lebih besar dari akurasi jika tidak digunakan metode klasifikasi *naïve bayes* dengan tingkat akurasi 90%. Sistem dapat berfungsi sesuai yang diharapkan dan dapat terhubung dengan *platform ThingSpeak* dengan baik. Akan tetapi, akun yang digunakan untuk menyimpan data pada *server ThingSpeak* tidak berbayar sehingga terdapat batasan untuk mengunggah data setiap 20 detik sekali.

Saran yang dapat diberikan dari penelitian ini yaitu mengganti *ThingSpeak* dengan menggunakan platform lainnya atau membuat platform sendiri yang serupa karena jika menggunakan *ThingSpeak* maka harus menggunakan MATLAB *versi desktop* yang tidak dapat mengambil data dari *ThingSpeak* kurang dari satu menit yang lalu, maka dibutuhkan delay selama satu menit setelah mengambil data agar data yang sebelumnya telah diambil tidak terambil kembali. Selain itu, metode alogritma *machine learning* lainnya dapat digunakan sebagai pembanding dalam tingkat akurasi.

Daftar Pustaka

[1] Shandy Y.D., Rakhmatsyah A., dan Suwastika N.A. 2015. *Implementasi Sistem Kunci Pintu Otomatis Untuk Smart Home Menggunakan SMS Gateway*. Telkom University.

[2] A. Saleh. 2015. *Implementasi metode klasifikasi naive bayes dalam memprediksi besarnya penggunaan listrik rumah tangga*. *Creative Information Technology Journal*.

[3] Michael A. Mahler, Qinghua Li, Ang Li. 2017. *SecureHouse: A Home Security System Based on Smartphone Sensors*. Department of Computer Science and Computer Engineering, University of Arkansas.

[4] Ibrahim R. 2017. *Rancangan Bangun Smart Lock System untuk Tanker*. Politeknik Negeri Jakarta.

[5] Mahali. M.I. 2016. *Smart Door Locks Based On Internet of Things Concept With Mobile Backend as a Service*. Universitas Negeri Yogyakarta.

[6] Suyanto. 2014. *Data Mining* Untuk Klasifikasi dan Klasterisasi Data. Penerbit Informatika, Bandung.