

Implementasi Model Logistic Regression dalam Aplikasi Mobile untuk Deteksi Phishing Secara Real-Time

Taufik Irfan¹, Rifa Hanifatunnisa², Irnanda Fidela³

^{1,2,3}Jurusan Teknik Elektro, Politeknik Negeri Bandung
Jl. Gegerkalong Hilir, Ciwaruga, Kec. Parongpong, Kab. Bandung Barat, Indonesia
taufik.irfan@polban.ac.id

Abstrak

Kemajuan teknologi informasi dan komunikasi semakin berkembang, memberikan dampak signifikan pada penggunaan sosial media seluruh masyarakat di dunia. Namun, dibalik kemajuan teknologi yang setiap harinya memberikan kemudahan pada pekerjaan manusia, ancaman dan kejahatan yang dilakukan di sosial media pun kian meningkat, salah satunya adalah *phising*. *Phising* merupakan sebuah kejahatan siber dimana korban akan diarahkan menuju suatu situs palsu dimana penyerang akan mencuri data-data sensitif untuk kepentingannya. Serangan siber ini sudah marak di dunia media sosial dan memakan banyak korban dari usia muda hingga lansia. Untuk mengatasi hal tersebut, sudah banyak *platform* seperti *website* yang dapat mendeteksi *phising*, namun masih terdapat kekurangan dalam hal akurasi terhadap serangan yang semakin kompleks serta kemudahan pengguna untuk memakainya. Maka dari itu, "Implementasi Model *Logistic Regression* dalam Aplikasi Mobile untuk Deteksi Phising Secara *Real-time*" menjadi sebuah solusi yang tepat dalam menangani kejahatan *phising*. Dengan mengadaptasi model *logistic regression* sebagai alat untuk mengklasifikasikan sebuah URL menjadi dua kategori utama yakni *phishing* dan *non phishing* diterapkan ke dalam *mobile* aplikasi, model algoritma ini memiliki tingkat keakurasian yang cukup tinggi. Dengan menggunakan dataset yang terdiri dari 11.025 sampel dan 30 parameter, model algoritma yang diterapkan mampu mengenali dan membaca pola sebuah URL yang cukup kompleks berdasarkan dataset yang dipelajarinya. Setelah dilakukan pengujian dengan 150 URL berbeda, model ini menghasilkan kinerja evaluasi yang cukup memuaskan dengan penghitungan akurasi, presisi, *recall* dan *f1 score* sebesar 96%.

Kata kunci: Machine Learning, Mobile Aplikasi, Logistic Regression, Phising

Abstract

The rapid growth of information and communication technology has greatly impacted social media use worldwide. However, alongside these advances that make our daily lives easier, there's also a rise in social media crimes, such as phishing. Phishing is a cybercrime where victims are tricked into visiting a fake website so that attackers can steal their sensitive information for personal use. This type of attack is becoming increasingly common on social media, affecting people of all ages. To address this problem, many platforms, like websites that detect phishing, have been developed. However, these websites often struggle with accurately identifying more complex phishing attacks and can be difficult for users to navigate. This is why a "Mobile App for Detecting Phishing URLs Using a Logistic Regression Algorithm" is an effective solution for fighting phishing crimes. This mobile app uses a logistic regression model to classify URLs as either phishing or non phishing, achieving high accuracy. The model is trained on a dataset of 11,025 samples and 30 parameters, allowing it to recognize and understand complex URL patterns. After testing with 150 different URLs, the model showed impressive results, with an accuracy, precision, recall, and f1 score of 96%.

Keywords: Machine Learning, Mobile App, Logistic Regression, Phising

I. PENDAHULUAN

Di era digitalisasi, TIK (Teknologi Informasi dan Komunikasi) mengalami perkembangan yang begitu pesat. Hal ini ditunjukkan dengan banyaknya masyarakat di dunia yang telah menggunakan alat komunikasi seperti laptop, *smartphone*, dan

komputer dalam berbagai hal, serta hampir seluruh bidang saat ini telah menggunakan pemanfaatan dari TIK. Di Indonesia sendiri, menurut data BPS dari hasil pendataan Survei Susenas 2020, 53,73% populasi di Indonesia telah mengakses internet di tahun 2020, serta tingginya jumlah pengguna internet di Indonesia tidak terlepas dari pesatnya

perkembangan telepon seluler [1]. Pada tahun 2020 tercatat 90,75% rumah tangga di Indonesia telah memiliki minimal satu nomor telepon seluler. Angka ini meningkat dibandingkan dengan kondisi tahun 2017 (88,13%) [1]. Kondisi ini menunjukkan tingginya penggunaan internet serta pesatnya perkembangan seluler mencerminkan adanya penerimaan masyarakat Indonesia terhadap perkembangan TIK serta memiliki dampak ketergantungan dalam keberadaan internet, sehingga adanya perubahan pola masyarakat di bidang sosial, ekonomi, budaya, pertahanan, keamanan, dan sebagainya.

Banyaknya aktivitas yang dipengaruhi oleh teknologi dan internet ini, tanpa disadari juga membuka peluang bagi pihak yang tidak bertanggung jawab untuk melakukan tindak kejahatan di internet (*cybercrime*). Menurut Andi Hamzah dalam bukunya *Aspek - aspek Pidana di Bidang Komputer* (2013) menyatakan bahwa *cybercrime* adalah kejahatan di bidang komputer secara umum dapat diartikan sebagai penggunaan komputer secara illegal [2]. Sementara di Indonesia, dilansir dari data e-MP Robinopsnal Bareskrim Polri menunjukkan bahwa kepolisian menindak 8.831 kasus kejahatan siber sejak Januari hingga Desember 2022 [3]. Dari data tersebut, terdapat hal yang menyebabkan tingginya angka kejahatan siber di Indonesia. Pertama, tidak dapat dipungkiri bahwa adanya kemajuan TIK menjadikan dunia tanpa batas dimana dampaknya juga dapat memudahkan pihak yang tidak bertanggung jawab untuk melakukan tindak kejahatannya. Kedua, kurangnya edukasi dan *awareness* terhadap masyarakat Indonesia dalam menghadapi perkembangan TIK serta *Cyber Security* sehingga banyak oknum yang tidak bertanggung jawab memanfaatkan pengetahuan yang minim dari banyaknya masyarakat Indonesia.

Salah satu contoh dari *cybercrime* adalah *phishing*, yang merupakan sebuah kejahatan siber dalam bentuk penipuan untuk memperoleh informasi sensitif dari korban seperti, *username*, *password*, nomor identitas, nomor kartu kredit dan sebagainya. Pada umumnya, bentuk *phishing* yang sering kali beredar di masyarakat Indonesia adalah sebuah pemberitahuan atau notifikasi seperti berasal dari instansi berwenang yang mengharuskan korban mengakses situs *phishing* yang berakibat data sensitif dari korban terakses oleh pelaku. Selain itu, modus *phising* banyak disebar di platform *e-mail*, *sms* dan platform *online* lainnya. Salah satu contoh korban *phising* adalah artis terkenal Baim Wong pada tahun 2023,

yang dimana Ia menerima pesan *whatsapp* dari nomor yang tidak dikenal. Pengirim tersebut mengaku sebagai kurir paket dan mengirimkan foto berupa file. Setelah Ia mengunduh file tersebut, maka hanya ada tampilan loading yang berakibat adanya transaksi di *m-banking* korban [4]. Dari kejadian tersebut, dapat disimpulkan bahwa pelaku tidak pandang bulu dalam melancarkan aksi kejahatannya. Menurut Ketua Pengelola Nama Domain Internet Indonesia, Yudho menyatakan bahwa “Jumlah *phising* dalam kurun waktu 5 tahun terakhir terhitung sejak 2022 mencapai 34.622” [5], ini menunjukkan bahwa *cybercrime* dalam bentuk *phising* sering terjadi dan banyak merugikan masyarakat Indonesia.

Berdasarkan uraian diatas, diperlukan tindak lanjut mengenai kasus *phising* di Indonesia. Dengan melihat permasalahan kasus *phising* yang umumnya terjadi pada pengguna *mobile*, perlu adanya sebuah solusi, salah satunya penerapan *machine learning* pada *mobile* aplikasi yang dapat mendeteksi link *phishing* dengan menggunakan algoritma *logistic regression*. Penerapan *machine learning* dalam *mobile* aplikasi pendeteksi *phising* ini memiliki fungsi untuk melakukan *proccessing data* melalui dataset yang dicantumkan. Sehingga, apabila pengguna memasukkan url *phishing* ke dalam *mobile* aplikasi ini, *machine learning* yang akan mempelajari pola dan karakteristik dari input data yang disesuaikan dengan dataset yang ada.

Algoritma *logistic regression* adalah salah satu model klasifikasi dari *machine learning* untuk mencari hubungan antara fitur input (diskrit/kontinu) dengan probabilitas hasil output diskrit tertentu [6]. Dalam kasus ini, algoritma *logistic regression* akan menganalisis sebuah input URL yang dimasukan oleh user dan mengkategorikan url tersebut sebagai url *phishing* dan *non phishing*. Cara kerjanya memodelkan probabilitas dari suatu kejadian atau parameter menggunakan fungsi logit (logistik). Harapannya, dengan menerapkan model ini sebagai pendeteksi *url phishing* dan *non phishing* dapat menjadi salah satu jalan keluar yang tepat untuk mencegah terjadinya lebih banyak lagi kasus *phishing*.

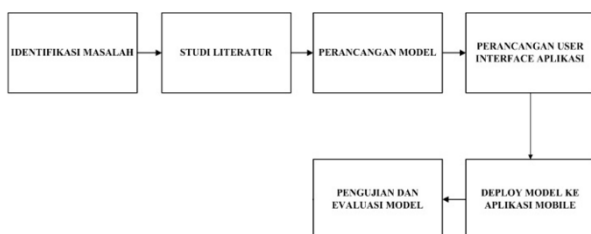
Perlu adanya sejumlah penelitian yang dilakukan mengenai analisis implementasi model *logistic regression* dalam aplikasi *mobile* pendeteksi URL *phising*. Penelitian ini bertujuan untuk melihat seberapa efektif penerapan algoritma *logistic regression* dalam mendeteksi URL *phising*. Berikut adalah beberapa penelitian sebelumnya sebagai landasan untuk memperkuat penelitian ini.

Dalam jurnal oleh Dwi dkk., berjudul *Comparison of Neural Network Algorithm, Naïve Bayes and Logistic Regression to Find the Highest Accuracy in Diabetes*, penulis membandingkan tiga algoritma dalam memprediksi penyakit diabetes. Hasilnya menunjukkan bahwa logistic regression memiliki akurasi tertinggi, yaitu 75,78%, disusul naïve bayes sebesar 74,87% dan neural network sebesar 69,27% [7]. Jurnal oleh Greessheilla dan Rido membahas penggunaan logistic regression untuk mengklasifikasikan persetujuan pinjaman di koperasi simpan pinjam. Algoritma ini digunakan dalam bentuk regresi logistik biner dan menunjukkan hasil akurasi yang baik berdasarkan penerapan istilah penting dalam model [8]. Brury Tangkere membandingkan logistic regression dan support vector classification dalam mendeteksi e-mail phishing. Kedua metode sama-sama menunjukkan akurasi sebesar 97%, sehingga keduanya dinilai efektif untuk klasifikasi e-mail phishing [9]. Penelitian oleh Fuzan dan Kiky menerapkan logistic regression, random forest, dan support vector machine untuk mendeteksi gejala awal COVID-19. Dari ketiga model, logistic regression mencatat akurasi terbaik sebesar 87% dalam memprediksi gejala awal penyakit tersebut [10]. Farida dan Ali membandingkan metode logistic regression dan random forest dengan pendekatan Correlation-based Feature Selection (CFS) dalam mendeteksi situs phishing. Hasilnya, random forest memberikan akurasi lebih tinggi dengan 17 prediksi benar dari 20 URL yang diuji [11]. Becti M. Susanto juga menggunakan teknik CFS dalam model binary logistic regression untuk mendeteksi phishing. Meskipun terjadi penurunan akurasi dari 93,99% menjadi 93,20% akibat meningkatnya false positive dan false negative, CFS

tetap membantu dalam menyaring fitur yang tidak relevan [12]. Dalam jurnal *Comparison of Logistic Regression, MultinomialNB, SVM, and K-NN Methods on Sentiment Analysis of Gojek App Review on the Google Play Store*, peneliti membandingkan empat algoritma machine learning. Hasilnya menunjukkan bahwa logistic regression memiliki akurasi terbaik, sekitar 82,45%, sedangkan metode K-NN menjadi yang paling tidak direkomendasikan [13]. Sari Ryani dkk. menerapkan regresi logistik biner menggunakan Python untuk menganalisis potensi pembukaan rekening oleh pengguna media sosial pada bank X. Model ini berhasil memberikan 15 probabilitas berbeda dengan hasil yang cukup signifikan [14]. Yuliana, Paradise, dan Mudawil menggunakan logistic regression untuk mendeteksi status gizi anak berdasarkan 657 data. Model ini mampu memberikan prediksi dengan baik, namun mengalami overfitting karena komposisi data yang kurang tepat dan tumpang tindih antara data latih dan uji [15]. Hendriyana, dkk melakukan penelitian dalam membandingkan algoritma *support vector machine*, *naïve bayes*, dan *regresi logistik* untuk memprediksi donor darah. Dalam implementasinya memanfaatkan perangkat lunak *RapidMiner*. Didapatkan hasil bahwa algoritma SVM memiliki akurasi tertinggi yakni 94,79% disusul dengan regresi logistik sebesar 82,59% dan naïve bayes sebesar 81,89% [16]. Dari tinjauan pustaka diatas, dapat disimpulkan bahwa performa dari model *logistic regression* cukup memberikan akurasi yang baik yakni diatas 80%. Pada penelitian kali ini, penulis perlu memperhatikan dataset, komposisi *data train* dan data uji serta program *python* yang benar agar model *logistic regression* dapat memiliki akurasi lebih dari 80%.

II. METODE PENELITIAN

A. Perancangan Sistem



Gambar 1. Diagram Blok Alur Pengerjaan

Gambar 1 merupakan diagram blok alur pengerjaan dalam perancangan dan realisasi mobile aplikasi pendeteksi url phishing dengan model

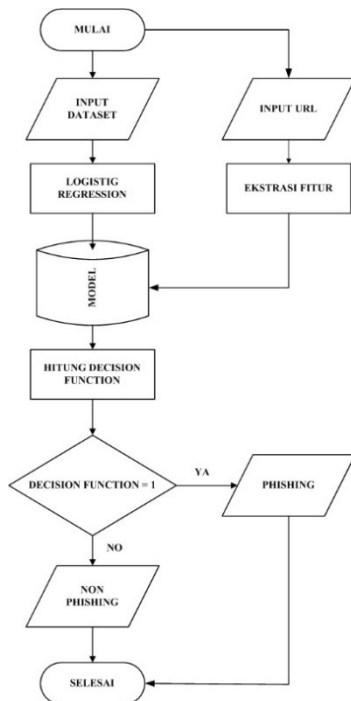
logistic regression. Dalam alur pengerjaan ini dibagi menjadi beberapa bagian utama sebagai berikut. Pertama adalah identifikasi masalah, dimana penulis mengenali dan mendefinisikan masalah yaitu maraknya phishing di Indonesia yang diselingi dengan penentuan solusi yang tepat untuk mengatasi masalah tersebut. Tahap kedua yakni studi literatur, dimana setelah mendidentifikasi masalah dan menemukan solusi, penulis melakukan pencarian lebih lanjut bagaimana solusi tersebut bisa dicapai dengan hasil yang maksimal, pemilihan metode yang sesuai hingga mencari referensi lebih lanjut terkait

desain dan model yang akan diterapkan untuk masalah yang telah diidentifikasi.

Tahap ketiga yang bisa dilakukan secara paralel, yakni perancangan desain dari *mobile* aplikasi atau *user interface* dengan perancangan model dari algoritma yang ditetapkan. Pada tahap ini, penulis terlebih dahulu membuat desain dari *mobile* aplikasi dan dituangkan kedalam android studio sebagai salah satu lingkungan pengembangan android. Setelahnya, penulis mulai untuk mengolah dataset dengan melakukan *pre-processing* dan menerapkan model algoritma *logistic regression* ke dalam dataset yang telah diolah. Langkah selanjutnya adalah *deployment* model yang telah dilatih ke dalam *mobile* aplikasi pada android studio, dilanjutkan dengan meng-*compile* file dalam bentuk .apk.

Tahap terakhir dari alur pengerjaan adalah pengujian dan evaluasi model. Pengujian dilakukan dengan beberapa tahapan yakni uji dataset, uji deep learning dan uji aplikasi. Pengujian tersebut dilakukan dengan memasukkan url random dan melihat output yang dihasilkan dari model yang telah dilatih. Setelah melakukan tahap pengujian, data yang telah dikumpulkan akan dikategorikan menjadi empat istilah yang ada di *confusion matrix* yang selanjutnya akan dihitung nilai evaluasi performa dari model logistic regression.

B. Perancangan Sistem Kerja Model Algoritma



Gambar 2. FlowChart Sistem Kerja Model

Pada Gambar 2 menunjukkan flowchat dari sistem kerja model logistic regression. Dalam pengimplementasian model logistic regression hingga dapat mengklasifikasikan url sebagai phishing atau

non phishing, diperlukan setidaknya 5 tahap utama. Tahap pertama yang merupakan inti dari keberhasilannya model yang dilatih adalah input dataset. Dataset yang di-*input*-kan sebagai bahan ajar untuk model yang akan diterapkan, telah melalui beberapa tahapan sebelumnya yaitu *pre-processing*, yang memiliki tujuan untuk membersihkan dan mempersiapkan data yang telah ada agar lebih akurat dan efisien dalam penggunaannya.

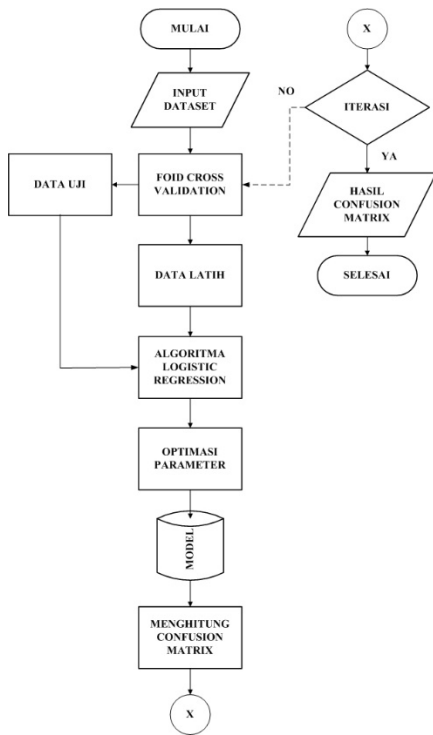
Setelah dipastikan bahwa dataset telah melewati tahap *pre-processing*, maka tahap selanjutnya adalah mengimplementasikan model *logistic regression* berdasarkan dataset yang telah diinput. Perlu diingat bahwa dalam pengimplementasian model, harus ditentukan pula komposisi antara data latih dengan data uji. Hal ini sangat mempengaruhi tingkat performa dari model yang berjalan agar mencegah terjadinya *overfitting* atau *underfitting*. Setelah mengimplementasikan model berdasarkan dataset yang diberikan, maka tahap selanjutnya yang menjadi tahap terakhir yaitu melakukan pengujian dan juga menghitung evaluasi dari model yang telah diterapkan. Untuk melakukan pengujian, pada penelitian ini dilakukan dengan memasukkan url satu persatu dan melihat keluaran dari hasil model yang telah bekerja. Hasil dari seluruh data yang telah melewati tahap pengujian akan disalurkan untuk diberi label sesuai dengan aturannya yang kemudian dihitung peforma dari model yang telah dilatih.

Tahap pengujian dibagi menjadi tiga bagian utama yang meliputi; pengujian model pyhton, pengujian model tensorflow dan pengujian model pada aplikasi. Harapannya, dengan melakukan tiga pengujian ini, dapat mengetahui peforma dari model *logistic regression* yang telah dilatih dari ketiga sisi dengan harapan melihat konsistensi dari seluruh kinerja model pada media yang berbeda.

C. Perancangan Sistem Evaluasi Model

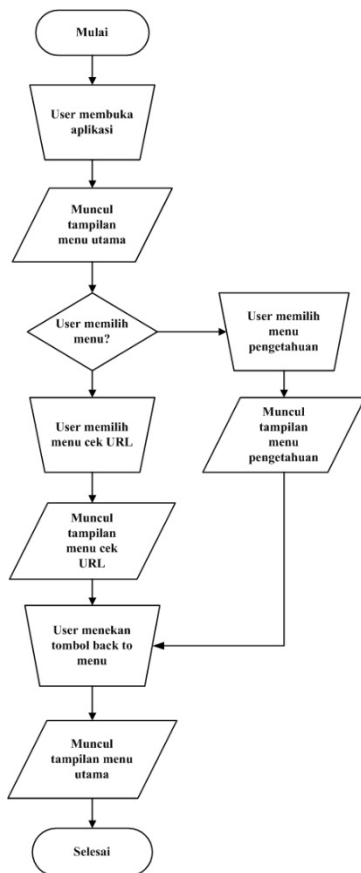
Pada *flowchart* Gambar 3 jelaskan bahwa ketika menginputkan dataset kedalam program, langkah selanjutnya adalah pembagian dataset menjadi data uji dan data latih. Setelah penulis membagi dataset menjadi data uji dan data latih, lalu selanjutnya adalah implementasi model *logistic regression* sesuai dengan komposisi dataset. Setelah implementasi model algoritma, maka perlu adanya *feature extraction* yang berisikan validasi dari seluruh fitur yang terdapat pada dataset. Hal ini berfungsi agar model dapat mengetahui batasan dari keseluruhan model dan mengintegrasikannya dengan dataset yang telah dimasukan. Setelahnya, dengan perintah print untuk perhitungan evaluasi yang berupa confusion matrix, maka hasil dari model yang telah dilatih dan diterapkan ke dalam model logistic regression, akan

memperlihatkan hasil evaluasi model berdasarkan data uji dan data latih yang telah dipelajarinya.



Gambar 3. Flowchart Evaluasi Model

D. Flow Chart Sistem Kerja Aplikasi



Gambar 4. Flowchart Sistem Kerja Aplikasi

Aplikasi yang nantinya akan digunakan oleh end user memiliki dua fitur utama, yakni Cek URL dan Knowledge seperti yang tertera pada Gambar 4. Pada menu cek url, pengguna dapat menginputkan URL yang nantinya akan dicek oleh model logistic regression yang telah di compile untuk menghasilkan dua kemungkinan keluaran, yakni URL termasuk phishing atau non phishing. Selain itu, pengguna juga dapat mengakses menu knowledge yang berisikan empat menu tambahan yakni what is phishing, example of phishing, preventing phishing dan yang terakhir overcoming phishing. Ketika pengguna mengakses menu knowledge, pengguna setidaknya dapat mendapatkan pengetahuan tambahan seputar phishing. Harapannya, ketika pengguna membuka aplikasi ini, pengguna dapat lebih waspada terhadap segala bentuk phishing dan lebih berhati-hati dalam menerima pesan masuk yang bisa membahayakan keselamatan data dan privasi pengguna.

III. HASIL DAN PEMBAHASAN

A. Implementasi Dataset

Setelah berhasil mengimplementasikan dataset ke dalam model algoritma, perlu adanya analisis lebih lanjut terkait dengan komposisi yang mengandung di dalam dataset yang penulis import. Berikut merupakan penjelasan secara rinci terkait komposisi yang terkandung dalam dataset.

```
Index(['Index', 'UsingIP', 'LongURL', 'ShortURL', 'Symbol@', 'Redirecting//',
      'PrefixSuffix-', 'SubDomains', 'HTTPS', 'DomainRegLen', 'Favicon',
      'NonStdPort', 'HTTPSDomainURL', 'RequestURL', 'AnchorURL',
      'LinksInScriptTags', 'ServerFormHandler', 'InfoEmail', 'AbnormalURL',
      'WebsiteForwarding', 'StatusBarCust', 'DisableRightClick',
      'UsingPopupWindow', 'IframeRedirection', 'AgeofDomain', 'DNSRecording',
      'WebsiteTraffic', 'PageRank', 'GoogleIndex', 'LinksPointingToPage',
      'StatsReport', 'class'],
      dtype='object')
```

Gambar 5. Komposisi Data Kolom

Gambar 5 di atas menunjukkan index setiap kolom yang ada di dataset sendiri. Di dalam dataset, terdapat 32 kolom dengan 31 sebagai fitur atau parameter yang menentukan suatu url *phishing* atau *non phishing* dan 1 kolom sebagai label atau kelas itu sendiri. Berikut tabel 1 merupakan penjelasan dari setiap parameter tersebut.

Tabel 1. Penjelasan Parameter

No	Parameter	Penjelasan
1	Index	Urutan baris dalam dataset
2	UsingIP	Menandakan bahwa dalam suatu URL menggunakan IP atau tidak
3	LongURL	Mengidentifikasi Panjang dari sebuah URL
4	ShortURL	Mengidentifikasi apakah URL terlalu pendek
5	Symbol@	Mengidentifikasi apakah URL mengandung symbol (@)
6	Redirecting//	Mengidentifikasi apakah URL menggunakan symbol //

7	PrefixSuffix	Menandakan penggunaan tanda hubung dalam domain
8	SubDomains	Jumlah subdomain yang digunakan dalam suatu URL
9	HTTPS	Menunjukkan apakah situs menggunakan HTTPS atau tidak
10	DomainRegLen	Panjang pendaftaran domain
11	Favicon	Memeriksa apakah favicon situs sesuai dengan yang resminya
12	NonStdPort	Apakah suatu URL menggunakan port non standar
13	HTTPSDomainURL	Memeriksa apakah URL dan domain menggunakan HTTPS
14	Request URL	Proporsi permintaan URL eksternal yang tidak sah dalam hal URL
15	AnchorURL	Persentase anchor tags (link) mengarah ke domain yang tidak terkait
16	LinkInScriptTags	Persentase link dalam script tag mengarah ke domain yang tidak sah
17	ServerFormHandler	Memeriksa apakah form handler pada server sesuai dengan domain yang sah
18	InfoEmail	Apakah situs memiliki alamat email yang mencurigakan
19	AbnormalURL	Apakah URL tidak normal jika dibandingkan dengan URL asli dari domain
20	WebsiteForwarding	Menandakan apakah situs menggunakan Teknik forwarding yang berlebihan
21	StatusBarCust	Memeriksa apakah status bar browser disesuaikan untuk menipu pengguna
22	DisableRightClick	Apakah situs phishing menonaktifkan klik kanan, yang sering mencegah pengguna melihat source code
23	UsisPopupWindow	Apakah situs menggunakan iframe untuk mengalihkan pengguna
24	IframeRedirection	Apakah situs menggunakan iframe untuk mengalihkan pengguna
25	AgeofDomain	Usia dari domain
26	DNSRecording	Keberadaan rekaman DNS untuk domain.
27	WebsiteTraffic	Tingkat lalu lintas ke situs web
28	PageRank	Peringkat halaman berdasarkan Google PageRank
29	GoogleIndex	Apakah URL diindeks oleh google
30	LinksPointingToPage	Jumlah link yang mengarah ke halaman
31	StatsReport	Laporan statistic mengenai situs
32	Class	Kategori label dari data atau keluaran hasil

Pada penelitian ini, proporsi sampel dengan label 1 atau *phishing* lebih banyak yakni sebesar 55.70% dibandingkan dengan proporsi sampel dengan label *non phishing* yaitu sebesar 44.30%. Namun dalam implementasinya, tidak ada pengaruh yang signifikan terkait dengan komposisi dataset ini.

Lalu terdapat tabel 2 menunjukkan informasi yang memuat komposisi setiap fitur. Di dalam tabel, terdapat 9 kolom yang berisikan parameter, count,

mean, std, min, 25%, 50% 75% dan max yang memiliki ketentuan dan fungsinya masing- masing.

Tabel 2. Parameter Dataset

Feature	Count	Mean	Std	Min	25%	50%	75%	Max
UsingIP	11054.0	0.31	0.95	-1.0	-1.0	1.0	1.0	1.0
LongURL	11054.0	-0.63	0.77	-1.0	-1.0	-1.0	-1.0	1.0
ShortURL	11054.0	0.74	0.67	-1.0	1.0	1.0	1.0	1.0
Symbol@	11054.0	0.70	0.71	-1.0	1.0	1.0	1.0	1.0
Redirecting//	11054.0	0.74	0.67	-1.0	1.0	1.0	1.0	1.0
PrefixSuffix-	11054.0	-0.73	0.68	-1.0	-1.0	-1.0	-1.0	1.0
SubDomains	11054.0	0.06	0.82	-1.0	-1.0	0.0	1.0	1.0
HTTPS	11054.0	0.25	0.91	-1.0	-1.0	1.0	1.0	1.0
DomainRegLen	11054.0	-0.34	0.94	-1.0	-1.0	-1.0	1.0	1.0
Favicon	11054.0	0.63	0.78	-1.0	1.0	1.0	1.0	1.0
NonStdPort	11054.0	0.73	0.69	-1.0	1.0	1.0	1.0	1.0
HTTPSDomainURL	11054.0	0.68	0.74	-1.0	1.0	1.0	1.0	1.0
RequestURL	11054.0	0.19	0.98	-1.0	-1.0	1.0	1.0	1.0
AnchorURL	11054.0	-0.08	0.72	-1.0	-1.0	0.0	0.0	1.0
LinksInScriptTags	11054.0	-0.12	0.76	-1.0	-1.0	0.0	0.0	1.0
ServerFormHandler	11054.0	-0.60	0.76	-1.0	-1.0	-1.0	-1.0	1.0
InfoEmail	11054.0	0.64	0.77	-1.0	1.0	1.0	1.0	1.0
AbnormalURL	11054.0	0.71	0.71	-1.0	1.0	1.0	1.0	1.0
WebsiteForwarding	11054.0	0.12	0.32	0.0	0.0	0.0	0.0	1.0
StatusBarCust	11054.0	0.76	0.65	-1.0	1.0	1.0	1.0	1.0
DisableRightClick	11054.0	0.91	0.41	-1.0	1.0	1.0	1.0	1.0
UsingPopupWindow	11054.0	0.61	0.79	-1.0	1.0	1.0	1.0	1.0
IframeRedirection	11054.0	0.82	0.58	-1.0	1.0	1.0	1.0	1.0
AgeofDomain	11054.0	0.06	1.00	-1.0	-1.0	1.0	1.0	1.0
DNSRecording	11054.0	0.38	0.93	-1.0	-1.0	1.0	1.0	1.0
WebsiteTraffic	11054.0	0.29	0.83	-1.0	0.0	1.0	1.0	1.0
PageRank	11054.0	-0.48	0.88	-1.0	-1.0	-1.0	1.0	1.0
GoogleIndex	11054.0	0.72	0.69	-1.0	1.0	1.0	1.0	1.0
LinksPointingToPage	11054.0	0.34	0.57	-1.0	0.0	0.0	1.0	1.0
StatsReport	11054.0	0.72	0.69	-1.0	1.0	1.0	1.0	1.0
class	11054.0	0.11	0.99	-1.0	-1.0	1.0	1.0	1.0

Sebagai contoh ketika akan menganalisis tabel komposisi dari parameter dengan fitur UsingIP. Pada parameter ini, *count* menghasilkan nilai 11.054 yang berarti observasi untuk parameter ini sebesar 11.054. Selanjutnya, mean 0,313914 yang berarti rata-rata penggunaan IP dalam URL adalah 0,313914. Std atau variasi dalam parameter ini sebesar 0,949495 dan nilai minimum adalah -1 yang mana menunjukkan bahwa URL yang tidak menggunakan IP mendapatkan nilai -1. 25% dari jumlah data memiliki nilai -1 dan median atau nilai tengahnya -1. Sedangkan 75% dari total data memiliki nilai 1 dan nilai maksimum dalam parameter ini adalah 1. Terdapat *Feature Extraction* yang ditetapkan dalam penelitian ini, dimana memiliki fungsi untuk memproses identifikasi dan mengumpulkan berbagai karakteristik atau atribut dari sebuah url untuk digunakan dalam analisis lebih lanjutnya. Dalam implementasi yang telah diterapkan dalam mengklasifikasikan sebuah URL ke dalam *phishing* dan *non phishing feature extraction* dengan output nilai -1 0 dan 1 berfungsi untuk menunjukkan seberapa besar kemungkinan URL yang akan diklasifikasikan sebagai *phishing* atau tidak. Ketika URL yang dimasukan memiliki nilai -1 pada *feature* tertentu, kemungkinan pada *feature* tersebut URL termasuk mencurigakan. Sedangkan ketika URL yang dimasukan memiliki nilai 1 maka kemungkinan pada *feature* tersebut URL termasuk ke dalam kategori aman. Nantinya, perhitungan dari setiap *feature* akan diakumulasikan dan dibuat probabilitas yang

akhirnya dilabeli sebagai URL *phishing* dan *non phishing*. Untuk lebih jelasnya, berikut tabel 3 merupakan nilai *feature extraction* dari program yang telah diterapkan dan ketentuan nilai probabilitasnya.

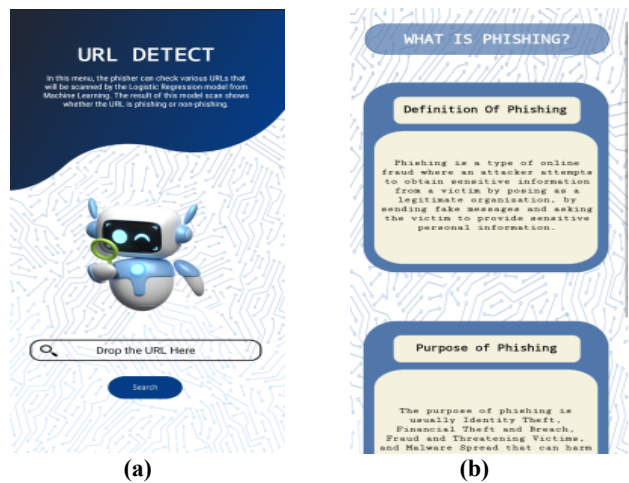
Tabel 3. Feature Extraction

No.	Feature Extraction	Ketentuan
1	UsingIp	Return -1 jika URL menggunakan alamat IP
2	longUrl	Return 1 jika panjang URL < 54, return 0 jika panjang URL antara 54 dan 75
3	shortUrl	Return -1 jika URL menggunakan layanan pemendek URL
4	symbol	Return -1 jika URL mengandung simbol @
5	redirecting	Return -1 jika terdapat lebih dari satu "/" dalam URL
6	prefixSuffix	Return -1 jika domain URL mengandung karakter -
7	SubDomains	Return 1 jika terdapat satu '.', return 0 jika terdapat dua '.', return -1 jika terdapat lebih dari dua '.'
8	Https	Return 1 jika URL menggunakan HTTPS
9	DomainRegLen	Return 1 jika usia domain lebih dari 12 bulan
10	Favicon	Return 1 jika favicon berasal dari domain yang sama
11	NonStdPort	Return -1 jika domain menggunakan port non-standar, selain itu return 1
12	HTTPSDomainURL	Return -1 jika domain menggunakan HTTPS
13	RequestURL	Return 1 jika < 22% permintaan URL berasal dari domain yang berbeda, return 0 jika antara 22%-61%, return -1 jika lebih dari 61%
14	AnchorURL	Return 1 jika < 31% anchor URL adalah tidak aman, return 0 jika antara 31%-67%, return -1 jika lebih dari 67%
15	LinksInScriptTags	Return 1 jika < 17% link dalam tag skrip adalah tidak aman, return 0 jika antara 17%-81%, return -1 jika lebih dari 81%
16	ServerFormHandler	Return -1 jika form handler kosong atau "about", return 0 jika form handler berasal dari domain berbeda, return 1 jika berasal dari domain yang sama
17	InfoEmail	Return -1 jika URL mengandung mailto
18	AbnormalURL	Return -1 jika response URL berbeda dengan data WHOIS
19	WebsiteForwarding	Return 1 jika history URL <= 1, return 0 jika <= 4
20	StatusBarCust	Return 1 jika tidak ditemukan script onmouseover
21	DisableRightClick	Return 1 jika tidak ditemukan script yang mendisable klik kanan, selain itu return -1
22	UsingPopupWindow	Return 1 jika tidak ditemukan alert(), selain itu return -1
23	IframeRedirection	Return 1 jika tidak ditemukan iframe, selain itu return -1
24	AgeofDomain	Return 1 jika usia domain lebih dari 6 bulan
25	DNSRecording	Return 1 jika usia domain lebih dari 6 bulan
26	WebsiteTraffic	Return 1 jika Alexa rank < 100000, return 0 jika >= 100000, selain itu return -1

27	PageRank	Return 1 jika global rank < 100000, selain itu return -1
28	GoogleIndex	Return 1 jika URL terindeks oleh Google
29	LinksPointingToPage	Return 1 jika jumlah link < 1, return 0 jika <= 2
30	StatsReport	Return -1 jika URL atau IP address mencurigakan

B. Implementasi Aplikasi

Dalam pengimplementasian aplikasi, terdapat beberapa perubahan desain UI. Perubahan tersebut dilakukan karena menyesuaikan kebutuhan penulis yang memiliki tujuan bahwa aplikasi yang sampai ke *end user* atau pengguna dapat dipahami dan mudah dimengerti. Namun, dengan adanya perubahan desain dari aplikasi ini, tidak merubah fungsionalitas dari tujuan pengimplementasian model algoritma *machine learning* terhadap pendeteksian URL *phishing* dan *non phishing*. Di dalam aplikasi terdapat navigasi yang berupa tombol yang mengarahkan ke halaman selanjutnya. Tombol navigasi ini memiliki fungsi untuk memudahkan pengguna dalam menjalankan aplikasinya. Perubahan yang dilakukan oleh pengguna ke dalam hasil implementasi berada pada halaman URL detect dan seluruh tampilan knowledge. Pada halaman URL detect sebagaimana mengacu pada Gambar 6, penulis mengubah susunan antara maskot, tulisan dengan tampilannya agar dapat dipahami dan indah dilihat. Perubahan user interface ini dilakukan karena pada *layout* url detect akan dimasukan model dari tensorflow yang akan mendeteksi inputan URL yang dimasukkan pengguna dan mengklasifikasikannya sebagai *phishing* dan *non phishing*. Tombol *search* pada menu url detect akan menghubungkan dan mengekstrasi sebuah URL masuk yang akan disambungkan dengan model yang telah dimasukan.



Gambar 6. (a) Tampilan menu URL Detect (b) Tampilan menu What is Phising

C. Hasil Pengujian

Hasil perhitungan dari kinerja model ini dibagi menjadi dua kategori. Kategori pertama adalah hasil perhitungan evaluasi model *logistic regression* dalam lingkup data uji dan data latih. Hasil perhitungan ini memuat empat kategori yaitu akurasi, presisi, recall dan f1 score. Sedangkan kategori kedua adalah hasil perhitungan evaluasi model *logistic regression* dalam lingkup pengujian, yang dimana penulis mengumpulkan sejumlah data yang nantinya dihitung melalui rumus matematis yang memuat empat kategori yaitu akurasi, presisi, recall dan f1 score.

1) Penghitungan Kinerja Model Data Uji dan Data Latih

```

Logistic Regression : Accuracy on Training Data: 92.865
Logistic Regression : Accuracy on test Data: 92.734

Logistic Regression : f1_score on Training Data: 93.627
Logistic Regression : f1_score on test Data: 93.616

Logistic Regression : Recall on Training Data: 94.412
Logistic Regression : Recall on test Data: 94.898

Logistic Regression : Precision on Training Data: 92.856
Logistic Regression : Precision on test Data: 92.368
    
```

Gambar 7. Evaluasi Model Data

Gambar 7 di atas menunjukkan hasil dari evaluasi kinerja data latih dan data uji pada model algoritma *logistic regression* yang memuat empat kategori seperti diatas. jika dilihat dari keseluruhan data yang kita dapatkan diketahui bahwa model algoritma telah melalui proses pembelajaran dengan cukup baik. Jika dilihat dari hasil akurasi, model memiliki nilai yang cukup tinggi baik itu pada data *train* ataupun data uji. Hal ini menunjukkan bahwa model memiliki kinerja yang baik dan mampu mengklasifikasi data dengan benar dalam Sebagian data kasus. Walaupun terjadi perbedaan yang kecil antara kedua akurasi (92,865% vs 92, 734%) namun model tidak mengalami *overfitting* yang berlebihan, yang artinya model cukup mengeneralisasi dan bekerja baik pada data yang belum pernah dilihat sebelumnya.

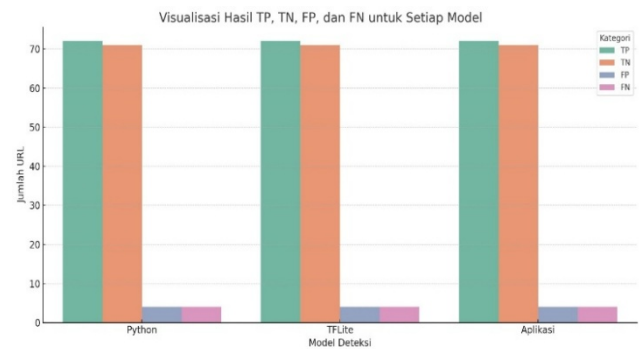
Selanjutnya adalah nilai presisi dalam kinerja evaluasi model *logistic regression*. Nilai presisi tinggi yang dihasilkan oleh model yakni 92% pada kedua data, menunjukkan bahwa ketika model memprediksi kelas positif (*phishing*), prediksi tersebut hamper selalu benar. Walaupun terdapat sedikit penurunan 0.488% antara data latih dan data uji, namun angka tersebut tidak mempengaruhi kinerja model dalam memprediksi. Maka dari itu, dengan nilai presisi yang cukup tinggi, jumlah *false positif* dalam evaluasi tersebut cukup sedikit.

Nilai recall pada hasil evaluasi kinerja dari data train dan data uji model *logistic regression* menghasilkan nilai yang cukup tinggi yakni lebih dari

94%. Dengan memperoleh nilai yang cukup tinggi ini, menandakan bahwa model algoritma memiliki kemampuan yang sangat baik dalam mendeteksi dan mengidentifikasi kasus positif atau URL *phishing*. Dan perhitungan terakhir yakni f1 score. F1 score merupakan ukuran harmonis dari perhitungan presisi dan recall. Nilai yang diperoleh dari hasil evaluasi model pada kategori ini yakni lebih dari 93%. Dengan kata lain, model memiliki kemampuan yang baik dalam mendeteksi kelas positif dengan akurasi yang cukup tinggi dan juga mengurangi kesalahan positif palsu.

2) Hasil Data Pengujian Model

Pengujian dilakukan menggunakan 151 URL yang memuat 76 URL sebagai status *phishing* dan 75 URL sebagai status *non phishing*. URL tersebut didapat oleh penulis dari kumpulan link *phishing* di platform online phishtank. Satu per satu, setiap link tersebut dimasukan dan diuji oleh penulis ke dalam tiga kondisi.



Gambar 8. Hasil Pengujian Model

Gambar 8 menunjukkan grafik hasil pengujian pada ketiga kondisi. Ketiga kondisi adalah pengujian pada pemrograman *python*, pengujian pada TFLite dan pada aplikasi. Maksud dari pengujian diatas adalah untuk melihat hasil keluaran dari masing – masing kondisi apakah sama atau tidak. Jika ditinjau dari hasil data yang dimiliki, menunjukkan bahwa pada setiap kondisi pengujian menghasilkan output yang sama yang berarti model algoritma tetap bekerja dengan maksimal di berbagai kondisi manapun.

3) Penghitungan Kinerja Model Pengujian

a. **Penghitungan Akurasi** digunakan untuk mengukur proporsi prediksi yang dilakukan oleh model algoritma baik itu prediksi benar maupun negative dari keseluruhan prediksi yang dilakukan oleh model.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

$$Accuracy = \frac{73 + 72}{73 + 72 + 3 + 3}$$

$$Accuracy = 96,0\%$$

b. **Penghitungan Presisi** digunakan untuk mengukur proporsi prediksi positif benar yang dilakukan oleh model algoritma.

$$Precision = \frac{TP}{TP + FP}$$

$$Precision = \frac{73}{73 + 3}$$

$$Precision = \frac{73}{76}$$

$$Precision = 96,0\%$$

c. **Penghitungan Recall** digunakan untuk mengukur proporsi prediksi positif benar dari semua kasus positif actual yang dilakukan oleh model algoritma.

$$Recall = \frac{TP}{TP + FN}$$

$$Recall = \frac{73}{73 + 3}$$

$$Recall = 96,0\%$$

d. **F1 score** yang digunakan untuk melihat keseimbangan antara presisi dan recall.

$$F1\ Score = 2x \frac{Precision \times Recall}{Precision + Recall}$$

$$F1\ Score = 2x \frac{96,0 \times 96,0}{96,0 + 96,0}$$

$$F1\ Score = 96\%$$

4) Pengujian User Interface Aplikasi

Untuk pengujian pada user interface aplikasi dilakukan untuk melihat apakah aplikasi dapat *compatible* atau berjalan di berbagai macam tipe *smartphone* android dengan tipe android yang bermacam-macam. Untuk parameter yang dilihat dari pengujian ini merupakan *compatible* atau tidaknya aplikasi ke *smartphone* dan versi android yang berbeda serta apakah adanya dislokasi tulisan, gambar dan lain sebagainya pada keseluruhan aplikasi saat dijalankan. Untuk lebih jelasnya, berikut tabel 4 hasil dari pengujian user interface aplikasi

Tabel 4. Hasil Pengujian User Interface

No	Tipe HP	Versi Android	Comp-atible	Dislo-kasi
1	Samsung A02s	12	Ya	Tidak
2	Redmi Note 10s	13	Ya	Tidak
3	Samsung A35 5G	13	Ya	Tidak
4	Samsung A34 5G	13	Ya	Tidak
5	Samsung A20s	12	Ya	Tidak
6	Samsung J7 Prime	12	Ya	Tidak
7	Oppo Reno7	12	Ya	Tidak
8	Samsung A05s	13	Ya	Tidak

9	Oppo A57	13	Ya	Tidak
10	Samsung Z Fold3	13	Ya	Tidak
11	Redmi Note 12 Pro	13	Ya	Tidak
12	Samsung A54	13	Ya	Tidak
13	Samsung A02s	12	Ya	Tidak
14	Samsung S24	13	Ya	Tidak
15	Realme C67	12	Ya	Tidak
16	Redmi Note 13	13	Ya	Tidak
17	Samsung A33	13	Ya	Tidak
18	Realme C35	13	Ya	Tidak
19	Vivo Y36	13	Ya	Tidak
20	Oppo A96	13	Ya	Tidak
21	Samsung A54	12	Ya	Tidak
22	Samsung A11	12	Ya	Tidak
23	Vivo V27	12	Ya	Tidak
24	Redmi Note 10s	12	Ya	Tidak
25	Samsung A53	12	Ya	Tidak

Berdasarkan tabel 4 diatas, terdapat 25 macam *smartphone* berbeda dengan versi android 12 – 13. Jika ditinjau dari data yang telah didapat, diketahui bahwa dengan semua tipe *smartphone*, aplikasi mobile yang dibuat dapat berjalan dengan baik dan tidak ada dislokasi posisi ataupun fungsi.

Dari keseluruhan hasil yang diperoleh untuk pengimplementasian mobile aplikasi pendeteksi URL phishing berbasis algoritma *logistic regression* menghasilkan respon yang cukup baik dari segala aspeknya. Aspek yang sangat berpengaruh dalam penerapan model *logistic regression* adalah dataset. Dataset yang didapat dari platform online bernama kaggle ini, memuat sampel data sebanyak 11.054 data dengan fitur atau parameter sebanyak 30. Dari parameter tersebut, jika ditelaah lebih dalam memiliki komposisi yang lebih besar pada sampel yang berlabel 1 atau *phishing* dibandingkan dengan yang berlabel -1 atau *non phishing*, yakni sebesar 55.70% banding 44.30%. Kondisi ini, merupakan kondisi yang sangat menguntungkan dikarenakan ketika kita melakukan penerapan model algoritma terhadap dataset untuk menjadi bahan pembelajaran, maka model akan memiliki sedikit lebih banyak informasi mengenai pola apa saja dari sebuah URL hingga diklasifikasikan sebagai *phishing*. Selain itu, pengaruh dari dataset terhadap klasifikasi sebuah URL adalah penerapan *feature extraction*. Dalam kasus ini, penulis mendefinisikan setiap fitur yang ada di dalam dataset menjadi *Feature Extraction*. Dengan kondisi ini, penerapan *Feature Extraction* sangat membantu model algoritma dalam pengklasifikasian sebuah URL karena mengandung nilai- nilai tertentu (-1, 0 dan 1) yang nantinya akan diakumulasikan oleh model untuk diberikan satu label yakni *phishing* dan *non phishing* pada suatu URL.

Aspek lainnya adalah penghitungan *confusion matrix* yang terbagi menjadi empat kategori yaitu akurasi, presisi, recall dan f1 score pada beberapa kondisi. Jika dilihat perhitungan *confusion matrix* dengan empat kategori tersebut dalam lingkup data latih dan data uji, model memberikan angka yang relative baik dengan rata-rata lebih dari 92%. Hal ini menandakan bahwa model algoritma dapat berjalan dengan baik didalam proses pembelajaran melalui data training dan proses klasifikasi melalui data uji hingga menghasilkan angka yang baik. Selain itu, komposisi nilai keempat kategori antara *training* data dan *test* data tidak terlalu signifikan jauh. Walaupun pada perhitungan akurasi, f1 score dan presisi terjadi penurunan angka, namun perbedaan tersebut tidak sampai 1% yang menunjukkan bahwa tidak terjadinya *overfitting* yang berlebihan didalam pembelajaran model algoritma. Namun jika dibandingkan hasil perhitungan *confusion matrix* dengan data hasil pengujian terdapat lonjakan kenaikan nilai perhitungan evaluasi kinerja model. Dari hasil perhitungan evaluasi kinerja model algoritma *logistic regression* terhadap empat kategori yang sama yaitu, akurasi, presisi, recall dan f1 score menunjukkan hasil yang cukup memuaskan yakni 96%. Dengan melihat kondisi ini, diketahui bahwa hasil pembelajaran model membuahkan tingkat keakurasian yang lebih tinggi karena pada saat pengujian, menghasilkan nilai kenaikan yang cukup signifikan.

IV. KESIMPULAN

Implementasi model *Logistic Regression* dalam aplikasi mobile untuk mendeteksi *Phishing* secara *real-time* dapat direalisasikan dengan baik dan membuahkan hasil yang cukup memuaskan. Dengan tingkat keakurasian, presisi, recall dan f1 score sebesar 96% aplikasi ini mampu dengan benar mengklasifikasikan sebuah URL yang dimasukan oleh user sebagai *phishing* dan *non phishing*. Hal ini menunjukan model *logistic regression* berhasil digunakan sebagai salah satu model klasifikasi yang memiliki performa baik khususnya dalam pengklasifikasian URL *phishing* dan *non phishing*.

UCAPAN TERIMA KASIH

Terima kasih kepada pihak P3M Polban yang telah mendukung penulis dalam melakukan penelitian ini

REFERENSI

- [1] T. K. al Lestari, *STATISTIK TELEKOMUNIKASI INDONESIA 2020*. Badan Pusat Statistik, 2020.
- [2] "PENGERTIAN CYBERCRIME MENURUT PARA AHLI." 2013. [Online]. Available: https://ogapermana.blogspot.com/2013/04/pengertian-cyber-crime-menurut-para-ahli_11.html
- [3] "KEJAHATAN SIBER DI INDONESIA NAIK

- BERKALI LIPAT," Polri, Pusiknas Bareskrim. [Online]. Available: https://pusiknas.polri.go.id/detail_artikel/kejahatan_siber_di_indonesia_naik_berkali-kali_lipat
- [4] S. Hardianyah, "CONTOH KASUS PHISING PADA APLIKASI WHATSAPP - BAIM WONG," *Liputan 6*. [Online]. Available: <https://www.liputan6.com/showbiz/read/5437837/baim-wong-apes-jadi-korban-penipuan-dengan-modus-phising-malware-yang-dikirim-lewat-aplikasi-chat?page=4>
- [5] P. Rahman Fauzi, "KASUS PHISING DI INDONESIA SELAMA 5 TAHUN TERAKHIR." [Online]. Available: <https://www.detik.com/jatim/berita/d-6483650/ada-34-622-kasus-phising-di-indonesia-selama-5-tahun-terakhir>
- [6] "machine-learning-2-logistic-regression-96b3d4e7b603 @ vincentmichael089.medium.com." [Online]. Available: <https://vincentmichael089.medium.com/machine-learning-2-logistic-regression-96b3d4e7b603>
- [7] D. Y. Utami, E. Nurlelah, and F. N. Hasan, "Comparison of Neural Network Algorithms, Naive Bayes and Logistic Regression to predict diabetes," *J. Informatics Telecommun. Eng.*, vol. 5, no. 1, pp. 53–64, 2021, doi: 10.31289/jite.v5i1.5201.
- [8] Greessheilla Phylosta P.B and Rido Febryansyah, "Permohonan Pinjaman Pada Koperasi Simpan Pinjam," *Ilmudata.org*, vol. 2, no. 12, pp. 1–12, 2022.
- [9] B. B. Tangkere, "Analisis Performa Logistic Regression dan Support Vector Classification untuk Klasifikasi Email Phising," vol. 5, no. 4, pp. 442–450, 2024.
- [10] F. Azimah and K. Rizky Nova Wardani, "Sistem Pendeteksi Gejala Awal Covid-19 dengan Penggunaan Metode AI Project Cycle," *J. Locus Penelit. dan Pengabd.*, vol. 1, no. 6, pp. 405–418, 2022, doi: 10.36418/locus.v1i6.135.
- [11] Farida and A. Mustopa, "Perbandingan Logistic Regression dan Random Forest menggunakan Correlation-based Feature Selection untuk Deteksi Website Phishing," *Sist. J. Sist. Inf.*, vol. 12, no. 1, pp. 13–20, 2023, [Online]. Available: <http://sistemasi.ftik.unisi.ac.id>
- [12] B. M. Susanto, "Binary Logistic Regression Untuk Mendeteksi Website Phising Menggunakan Correlation-Based Feature Selection," *J. Teknol. Inf. dan Terap.*, vol. 2, no. 2, pp. 255–260, 2019.
- [13] A. Maulana, Inayah Khasnaputri Afifah, Asghafi Mubarrak, Kiagus Rachmat Fauzan, Ardhan Dwintara, and B. P. Zen, "Comparison of Logistic Regression, Multinomialnb, Svm, and K-Nn Methods on Sentiment Analysis of Gojek App Reviews on the Google Play Store," *J. Tek. Inform.*, vol. 4, no. 6, pp. 1487–1494, 2023, doi: 10.52436/1.jutif.2023.4.6.863.
- [14] M. Regresi *et al.*, "MEDIA SOSIAL TERHADAP PROBABILITAS PEMBUKAAN," vol. 7, no. 2, pp. 438–449, 2024.
- [15] Yuliana, Paradise, and Mudawil Qulub, "Detection of Children'S Nutritional Status Using Machine Learning With Logistic Regression Algorithm," *JURTEKSI (Jurnal Teknol. dan Sist. Informasi)*, vol. 10, no. 2, pp. 267–274, 2024.
- [16] H. Hendriyana, I. M. Karo Karo, and S. Dewi, "Analisis perbandingan Algoritma Support Vector Machine, Naive Bayes dan Regresi Logistik untuk Memprediksi Donor Darah," *J. Teknol. Terpadu*, vol. 8, no. 2, pp. 121–126, 2022, doi: 10.54914/jtt.v8i2.581.