APLIKASI PENDETEKSI *WEBSITE PHISHING* MENGGUNAKAN *MACHINE LEARNING*

TUGAS AKHIR

Disusun dalam rangka memenuhi salah satu persyaratan

Untuk menyelesaikan program Strata-1 Departemen Teknik Informatika

Fakultas Teknik Universitas Hasanuddin

Makassar

Disusun Oleh:

DIKI WAHYUDI

D421 15 518

DEPARTEMEN TEKNIK INFORMATIKA

FAKULTAS TEKNIK

UNIVERSITAS HASANUDDIN

MAKASSAR

2020

LEMBAR PENGESAHAN

"APLIKASI PENDETEKSI WEBSITE PHISHING MENGGUNAKAN MACHINE LEARNING"

Disusun Oleh:

DIKI WAHYUDI **D421 15 518**

Skripsi ini telah dipertahankan pada Ujian Akhir Sarjana tanggal 04 Januari 2020. Diterima dan disahkan sebagai salah satu syarat memperoleh gelar Sarjana Teknik (S.T) pada Departemen Teknik Informatika Fakultas Teknik Universitas Hasanuddin.

Gowa, 07 Januari 2020

Disetujui Oleh:

Pembimbing I,

Dr. Eng. Muhammad Niswar, ST., M.IT.

NIP. 19730922 199903 1 001

Pembimbing II,

A. Ais Prayogi Alimuddin, ST., M.Eng.

NIP. 19830510 201404 1 001

Diterima dan disahkan oleh:

Ketua Departemen S1 Teknik Informatika

Dr. Amil Ahmad Ilham, S.T., M.IT

NIP. 19731010 199802 1 001

KATA PENGANTAR

Puji dan syukur penulis panjatkan kepada Tuhan yang Maha Esa karena berkat rahmat dan karunia-Nya sehingga tugas akhir yang berjudul "APLIKASI PENDETEKSI WEBSITE PHISHING MENGGUNAKAN MACHINE LEARNING" ini dapat diselesaikan sebagai salah satu syarat dalam menyelesaikan jenjang Strata-1 pada Departemen Teknik Informatika Fakultas Teknik Universitas Hasanuddin.

Penulis menyadari bahwa dalam penyusunan dan penulisan laporan tugas akhir ini tidak lepas dari bantuan, bimbingan serta dukungan dari berbagai pihak, dari masa perkuliahan sampai dengan masa penyusunan tugas akhir. Oleh karena itu, penulis dengan senang hati menyampaikan terima kasih kepada:

1. Kedua Orang tua penulis, Bapak Alimuddin dan Ibu Suriani yang selalu memberikan dukungan, doa, dan semangat serta selalu sabar dalam mendidik penulis sejak kecil;
2. Bapak Dr. Eng. Muhammad Niswar, ST., M.IT selaku pembimbing I dan Bapak A. Ais Prayogi Alimuddin, ST., M.Eng. selaku pembimbing II yang selalu menyediakan waktu, tenaga, pikiran dan perhatian yang luar biasa untuk mengarahkan penulis dalam penyusunan tugas akhir;
3. Bapak Dr. Eng. Ady Wahyudi Paundu, S.T., M.T. yang senantiasa memberikan nasehat, masukan, serta perhatian yang luar biasa kepada penulis dalam penyusunan tugas akhir;

4. Bapak Dr. Amil Ahmad Ilham, ST., M.IT., selaku Ketua Departemen Teknik Informatika Fakultas Teknik Universitas Hasanuddin atas bimbingannya selama masa perkuliahan penulis;
5. Para sahabat dan teman-teman yang telah memberikan begitu banyak bantuan selama penelitian, pengambilan data dan diskusi progress penyusunan tugas akhir;
6. Teman-teman Hypervisor FT-UH atas dukungan dan semangat yang diberikan selama ini;
7. Segenap Staf dan Dosen Departemen Teknik Informatika Fakultas Teknik Universitas Hasanuddin yang telah membantu penulis.
8. Orang-orang berpengaruh lainnya yang tanpa sadar telah menjadi inspirasi penulis.

Akhir kata, penulis berharap semoga Tuhan berkenan membalas segala kebaikan dari semua pihak yang telah banyak membantu. Semoga Tugas Akhir ini dapat memberikan manfaat bagi pengembangan ilmu.

Makassar, 06 November 2019

Penulis

ABSTRAK

Phishing merupakan tindakan untuk mendapatkan informasi penting seseorang berupa username, password, dan informasi sensitif lainnya dengan memberikan *website* palsu yang mirip dengan aslinya. *Phishing* (memancing informasi penting) adalah suatu bentuk tindakan kriminal yang bermaksud untuk mendapatkan informasi rahasia dari seseorang, seperti username, password dan kartu kredit, dengan menyamar sebagai orang atau bisnis yang tepercaya dalam sebuah komunikasi elektronik resmi, seperti surat elektronik atau pesan instan. Seiring dengan perkembangan penggunaan media elektronik, yang diikuti dengan meningkatnya pula *cyber crime* seperti salah satunya serangan *phishing* ini. Oleh karena itu, untuk meminimalisir serangan *phishing* dibutuhkan sebuah sistem yang dapat mendeteksi serangan tersebut. *Machine Learning* merupakan salah satu metode yang dapat digunakan untuk membuat sistem yang dapat mendeteksi *phishing.* Data yang digunakan dalam penelitian ini sebanyak 11055 data website, yang terbagi atas dua *class* yaitu "*legitimate*" dan "*phishing*". Data ini kemudian dibagi dengan menggunakan *10-fold cross validation*. Sedangkan algoritma yang digunakan adalah algoritma *Support Vector Machine* (SVM) yang dibandingkan dengan algoritma *decision tree* dan *k-nearest neighbor* dengan melakukan optimasi parameter pada setiap algoritma. Dari hasil pengujian pada penelitian ini diperoleh akurasi sistem terbaik 85.71% menggunakan SVM *kernel polynomial* dengan nilai *degree* 9 dan *C 2.5*.

Kata Kunci: *phishing*, *machine learning*, *support vector machine* (SVM), ekstraksi fitur, optimasi parameter.

DAFTAR ISI

DAFTAR GAMBAR

DAFTAR TABEL

BAB I

PENDAHULUAN

1.1. Latar Belakang

Seiring perkembangan teknologi khususnya teknologi informasi seperti Internet yang telah membawa manusia menuju zaman yang berbeda, yaitu zaman yang terasa lebih mudah, praktis, simpel, dan dinamis. Dengan berkembangnya teknologi informasi orang dapat dengan mudah serta cepat dalam mendapatkan informasi apa yang diinginkan bahkan dalam mendapatkan informasi tersebut ada beberapa pihak yang melakukannya dengan cara ilegal serta merugikan orang lain. Internet menjadi bagian penting dalam kehidupan masyarakat. Internet telah mempengaruhi seluruh bagian kehiduapan masyarakat khususnya kehidupan social dan finansial. Sebagai contoh media sosial dan *website* yang digunakan untuk komunikasi dan bisnis. Namun seiring dengan perkembangan teknologi informasi tersebut telah membawa kekhawatiran tersendiri terhadap masyarakat, dimana adanya beberapa pihak yang tidak bertanggung jawab yang dapat merugikan orang lain.

Phishing merupakan tindakan untuk mendapatkan informasi penting seseorang berupa *username, password,* dan informasi sensitif lainnya dengan memberikan *website* palsu yang mirip dengan aslinya. *Phishing* (memancing informasi penting) adalah suatu bentuk tindakan kriminal yang bermaksud untuk tkan informasi rahasia dari seseorang, seperti *username, password* dan dit, dengan menyamar sebagai orang atau bisnis yang tepercaya dalam omunikasi elektronik resmi, seperti surat elektronik atau pesan instan.

Istilah *phishing* dalam bahasa Inggris berasal dari kata fishing ('memancing'), dalam hal ini berarti memancing informasi keuangan dan kata sandi pengguna. Dengan banyaknya kasus penipuan yang dilaporkan, metode tambahan atau perlindungan sangat dibutuhkan. Upaya-upaya itu termasuk pembuatan undang-undang, pelatihan pengguna, dan langkah-langkah teknis. *Phishing* biasanya susah dideteksi khususnya bagi masyarakat awam yang tidak bergerak di bagian teknikal. Hal ini tambah diperparah dengan meningkatnya penggunaan smartphone yang biasanya tidak memperlihatkan URL dari sebuah *website* secara keseluruhan. (Halim Z. 2017:72)

Berbicara mengenai *phishing* maka akan dikaitkan juga dengan sosial media yang penggunaannya telah meningkat setiap tahunnya. Dan apabila diperhatikan sosial media merupakan media yang paling penting bagi seorang hacker dalam melancarkan serangannya. Hal ini dikarenakan penggunaan sosial media yang begitu besar serta mudahnya mendapatkan informasi seseorang melalui akun sosial medianya. Serangan *phishing* seperti asal katanya "fishing" yaitu memancing dengan memberikan umpan kepada korban dan menunggu korban untuk mengambil umpan tersebut.

Dari beberapa penelitian yang telah dilakukan maka pada penelitian ini, penulis mencoba untuk membangaun sebuah aplikasi pendeteksi *phishing* menggunakan metode machine learning dengan algoritma klasifikasi *Support achine* (SVM), *Decision Tree*, dan *K-Nearest Neighbors* dengan Bahasa man python.

1.2. Rumusan Masalah

Berdasarkan latar belakang yang telah dipaparkan, maka rumusan masalah pada penelitian ini antara lain:

1. Bagaimana mengekstrak fitur dari sebuah halaman web berbasis URL untuk digunakan dalam proses deteksi *phishing*?
2. Bagaimana membangun sebuah sistem pendeteksi *phishing* menggunakan algoritma SVM?

1.3. Tujuan Penelitian

Tujuan dalam penelitian ini adalah:

1. Mengekstrak fitur dari sebuah halaman website dengan berbasis URL untuk digunakan dalam proses deteksi *phishing*.
2. Membangun aplikasi pendeteksi *phishing* menggunakan algoritma SVM dengan bahasa pemrograman python.

1.4. Manfaat Penelitian

Manfaat yang diharapkan dalam penelitian ini adalah:

1. Bagi masyarakat, dapat menggunakan aplikasi pendeteksi *phishing* ini untuk mengecek apakah *website* yang akan diakses adalah *phishing* atau bukan.
2. Bagi peneliti, dapat digunakan untuk menambah pengetahuan dan sebagai referensi mengenai deteksi *phishing* dengan menggunakan machine learning.

3. Bagi institusi pendidikan, dapat digunakan sebagai referensi dalam pengembangan penelitian topik terkait untuk mempelajari deteksi *phishing* dengan menggunakan machine learning.

1.5. Batasan Masalah

Ruang lingkup pembahasan tugas akhir ini dibatasi hanya mencakup hal-hal berikut:

1. Aplikasi dibangun dalam bentuk *Command Line Interface* (CLI).
2. Bahasa yang digunakan dalam membangun aplikasi pendeteksi *phishing* yaitu bahasa pemrograman python.
3. Output dari aplikasi pendeteksi *phishing* berupa persentase *phishing* dan hasil prediksi *phishing*.

1.6. Sistematika Penulisan

Untuk memberikan gambaran singkat mengenai isi tulisan ini, maka akan diuraikan beberapa tahapan dari penulisan secara sistematis, yaitu:

BAB I PENDAHULUAN

Bab ini menguraikan secara umum mengenai hal yang menyangkut latar belakang, perumusan masalah, batasan masalah, tujuan, dan manfaat penelitian.

BAB II TINJAUAN PUSTAKA

Bab ini berisi teori-teori terkait hal-hal yang mendasari dan berhubungan dengan penelitian, termasuk di dalamnya iridologi, visi komputer, dan metode-metode yang digunakan dalam penelitian.

BAB III METODOLOGI PENELITIAN

Bab ini berisi tentang perencanaan dan proses penerapan algoritma dan metode-metode dalam pengolahan data, mulai dari preprocessing hingga menghasilkan prediksi.

BAB IV HASIL DAN PEMBAHASAN

Bab ini berisi tentang hasil penelitian dan pembahasan terkait pengolahan data yang telah dilakukan yang disertai dengan tabel hasil penelitian.

BAB V PENUTUP

Bab ini berisi tentang kesimpulan yang didapatkan berdasarkan hasil penelitian yang telah dilakukan serta saran untuk pengembangan sistem yang lebih lanjut.

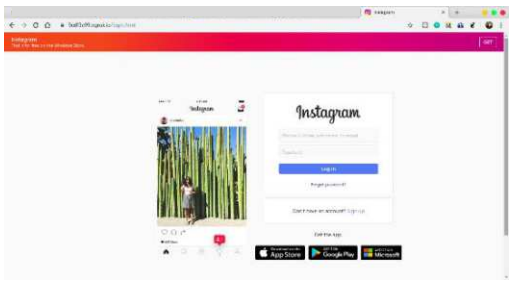
BAB II

TINJAUAN PUSTAKA

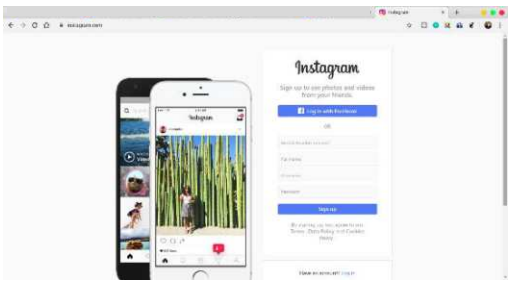
2.1. Phishing

Phishing merupakan tindakan untuk mendapatkan informasi penting seseorang berupa username, password, dan informasi sensitif lainnya dengan memberikan *website* palsu yang mirip dengan aslinya.

Phishing (memancing informasi penting) adalah suatu bentuk tindakan kriminal yang bermaksud untuk mendapatkan informasi rahasia dari seseorang, seperti username, password dan kartu kredit, dengan menyamar sebagai orang atau bisnis yang tepercaya dalam sebuah komunikasi elektronik resmi, seperti surat elektronik atau pesan instan. Dengan banyaknya kasus penipuan yang dilaporkan, metode tambahan atau perlindungan sangat dibutuhkan. Upaya-upaya perlindungan itu termasuk pembuatan undang-undang, pelatihan pengguna, dan langkah-langkah teknis. *Phishing* biasanya susah dideteksi khususnya bagi masyarakat awam yang tidak bergerak di bagian teknikal.[1]



Website Phishing



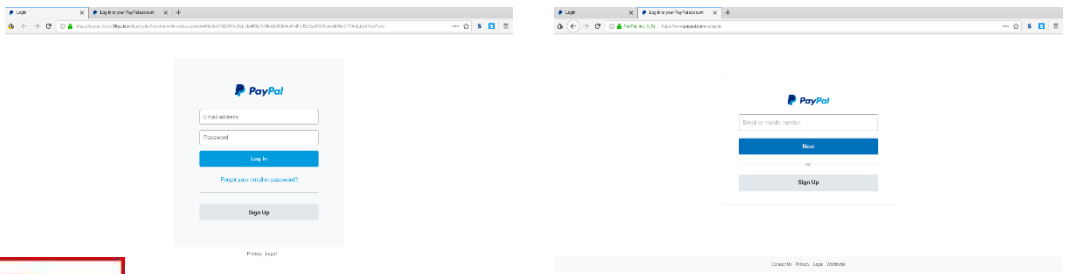
Website Asli

r 2.1. Perbandingan *website phishing* Instagram dengan website aslinya

"Prediksi Website Pemancing Informasi Penting Phising Menggunakan Support Vector VM)", Information System for Educators and Professionals, 2017, hlm. 72.

Serangan *phishing* biasanya berupa sebuah email yang seolah-olah berasal perusahaan resmi, misalnya dari *website-website* yang biasa digunakan oleh *user*. Tujuan serangan ini adalah untuk mendapatkan data-data pribadi *user*, misalnya *password*, nomor rekening, nomor kartu kredit, dan data-data sensitif lainnya.

Seorang penyerang *phishing* biasanya disebut *phisher*. Dalam melakukan serangan, seoarang *phisher* akan menyamar menjadi orang lain yang berasal dari pihak resmi dari sebuah perusahaan dengan cara mengirimkan alamat web yang seolah-olah mengarah pada web resmi dari perusahaan tersebut, padahal alamat web tersebut akan diarahkan ke alamat web yang telah dibuat oleh *phisher* yang menyerupai dengan web resmi perusahaan tersebut. Dengan diarahkannya target ke alamat web yang palsu tersebut maka target akan diminta untuk memasukkan data-data sensitive yang diperlukan oleh *phisher*, seperti diminta untuk memasukkan *username*, *password*, nomor rekening, nomor kartu kredit, dan lain-lain. Seorang *phisher* dalam melakukan serangan *phishing* ini tentunya tidak hanya melakukan serangan terhadap satu target melainkan banyak target, dengan alasan bahwa dari banyak target tersebut akan terdapat satu atau dua target berhasil terpancing dengan serangan *phishing* tersebut.



Website Phishing Website Asli

ar 2.2. Perbandingan *website phishing* PayPal dengan website aslinya

Phishing biasanya memanfaatkan email, *website* palsu, spyware dan berbagai media lainnya untuk melakukan aksinya. Beberapa hal yang menyebabkan aksi *phishing* ini terus terjadi dan memakan banyak korban adalah:[2]

1. Ketidaktahuan atau kurangnya pengetahuan

   Kurangnya pengetahuan akan teknologi komputer membuat pelaku *phishing* mudah mendapatkan mangsanya. Dengan memberikan email yang menakutkan, seperti ancaman hilangnya nama domain akan membuat korbannya segera melakukan apa yang diminta.

2. Tampilan palsu yang menyesatkan

   Pemalsuan *website* dan gambar-gambar sangat mudah dilakukan melalui internet dan pengguna awam biasanya tidak menyadari hal tersebut. Hanya dengan melakukan *copy* dan *paste*, sebuah *website* yang mirip dengan asli akan langsung tercipta. *Phisher* juga bisa membuat *website* yang tampak sangat bagus dengan berbagai komentar pengguna yang semuanya fiktif untuk meyakinkan calon korbannya.

3. Kurangnya perhatian pada indikator keamanan

   Sangat sering, pesan-pesan yang muncul tidak dibaca oleh *user*. Biasanya, pesan-pesan ini terlalu teknis untuk *user* sehingga mereka selalu mengklik tombol "OK" untuk melanjutkan. Kebiasaan semacam ini membawa keuntungan tersendiri untuk *phisher* sehingga mereka

ainal Arifin Al, "Cyber Crime Dalam Bentuk Phising Dalam Undang-Undang Nomor 008 Tentang Informasi Dan Transaksi Elektronik Perspektif Hukum Pidana Islam", Islam Negeri Sunan Ampel, 2016, hlm. 61.

bisa memalsukan *website* dan mendapatkan informasi berharga yang dimasukkan oleh korbannya.

4. Meningkatnya penggunaan platform mobile

   Dengan meningkatnya penggunaan platform mobile ikut meningkatkan pula tindak serangan *phishing*. Hal ini dikarenakan dengan platform mobile seorang *user* kurang memperhatikan alamat web yang kunjungi karena layarnya yang cukup kecil sehingga URL dari web tersebut tidak ditampilkan secara keseluruhan.

Salah bentuk teknik serangan *phishing* adalah *URL Obfuscation.* URL (*Uniform Resource Locator* atau alamat web yang diketik dalam browser untuk membuka suatu *website*) *Obfuscation* adalah suatu teknik menyamarkan alamat URL sehingga tampak tidak mencurigakan untuk pengguna. Adapun macam-macamnya adalah sebagai berikut:[3]

1. String yang menyesatkan

   Memanfaatkan string yang tampak asli seperti adanya kata-kata "Microsoft" atau kata-kata yang umum dikenal. Untuk memalsukan *website* "Microsoft" misalnya, seorang *phisher* akan membuat direktori menggunakan kata-kata yang sama yaitu "Microsoft", seperti http://XX.com/Microsoft.com/freelogin.php pelaku kemudian akan

ainal Arifin Al, "Cyber Crime Dalam Bentuk Phising Dalam Undang-Undang Nomor 008 Tentang Informasi Dan Transaksi Elektronik Perspektif Hukum Pidana Islam", Islam Negeri Sunan Ampel, 2016, hlm. 66.

membuat halaman jebakan untuk mendapatkan *username* dan *password* atau informasi berharga lainnya.

2. Menggunakan simbol "@"

   Simbol "@" sebenarnya digunakan untuk *website* yang membutuhkan autentikasi di mana tanda sebelum simbol "@" menunjukkan *username*, sedangkan setelahnya menunjukkan domain. Contoh sederhana pada email sto@jasakom.com.

   Kata "sto" menunjukkan nama sedangkan "jasakom.com" menunjukkan domain. Teknik ini pernah memakan banyak korban dan pernah sangat populer, yaitu http://www.microsoft.com@www.hacker.com. Domain ini jika di klik oleh pengguna maka akan masuk ke situs www.hacker.com, bukan situs Microsoft yang sebenarnya.

3. Nama yang mirip

   Teknik yang pernah menimpa situs klikbca.com ini akan membuat nama yang mirip dan memanfaatkan kelemahan *user* yang suka salah ketik atau salah ingat. Sebagai contoh pada kasus klikbca.com, *phisher* bisa membuat *website* kilikbca.com, klickbca.com dan lain sebagainya. Tentu saja, alamat palsu ini juga dibuat dengan tampilan yang sama persis dengan situs aslinya. Memanfaatkan nama yang mirip tidak harus selalu memanfaatkan kesalahan ketikan atau kesalahan ingat. *Phisher* juga bisa membuat nama domain yang tampak asli seperti microsoft-online.com, microsoft-user.com, dan lain sebagainya.

4. *Shortener URL*

Layanan *shortener* URL menjadi terkenal dan sering digunakan, sebagai contoh perhatikan URL dari Amazon berikut ini:

"http://www.amazon.com/Kindle-Wireless-Reading-Display-Globally/dp/B003FSUDM4/ref=amb_link_353459562_2?pf_rd_m=ATVPDKIKX0DER&pf_rd_s=center-10&pf_rd_r=11EYKTN682A79=1i=B002Y27P3M"

Menghafal alamat yang sedemikian panjang tentu akan sulit dan bahkan hampir tidak mungkin untuk dilakukan kecuali untuk orang-orang jenius. Kini, dengan bantuan layanan *shortener* URL, alamat contoh di atas bisa berubah menjadi:

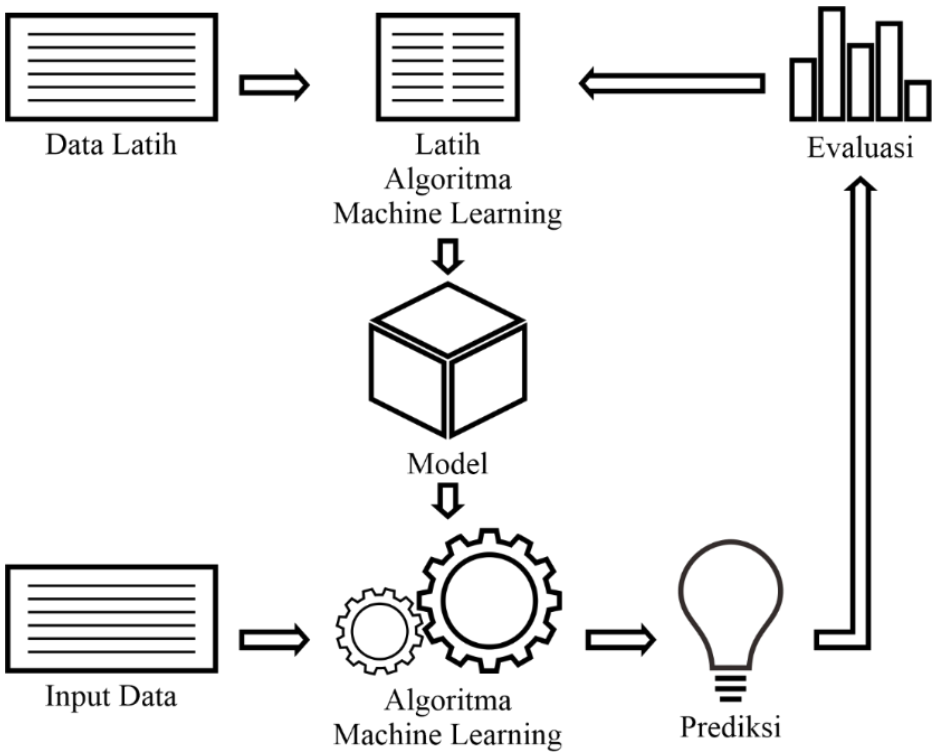
"http://tinyurl.com/KindleWireless"

Karena terbiasa untuk menggunakan layanan semacam ini, banyak orang yang tidak lagi memperhatikan alamat asli yang digunakan. Pelaku phishing bisa memanfaatkan ini untuk menutupi URL asli yang digunakan.

2.2. Machine Learning

Machine learning adalah disiplin ilmu dari *Artificial Intelligence* (Kecerdasan Buatan) yang menggunakan teknik statistika untuk menghasilkan suatu model otomatis dari sekumpulan data yang biasa disebut *dataset*, dengan tujuan memberikan komputer kemampuan untuk "belajar". Pembelajaran mesin hine learning memungkinkan komputer mempelajari sejumlah data (*learn a*) sehingga dapat menghasilkan suatu model untuk melakukan proses

input-output tanpa menggunakan kode program yang dibuat secara eksplisit. Proses belajar tersebut menggunakan algoritma khusus yang disebut *machine learning algorithms*. Terdapat banyak *algoritma machine learning* dengan efesiensi dan spesifikasi kasus yang berbeda-beda. Tidak hanya individu yang belajar meningkatkan kecerdasan nya tetapi mesin juga membutuhkan hal tersebut untuk meningkatkan kecerdasannya dan memiliki kemampuan yang cerdas dan tidak dimiliki oleh mesin lainnya.[4]



Gambar 2.3. Konsep *Machine Learning*

Secara fundamental cara kerja *machine learning* adalah belajar seperti manusia dengan menggunakan contoh-contoh dan setelah itu barulah dapat

[4] ia, "Apa itu Machine Learning dan Cara Kerjanya", diakses dari w.advernesia.com/blog/data-science/machine-learning-adalah/, pada tanggal 10 019 pukul 10.27.

menjawab suatu pertanyaan terkait. Proses belajar ini menggunakan data yang disebut *train dataset* (data latih). Berbeda dengan program statis, *machine learning* diciptakan untuk membentuk program yang dapat belajar sendiri. Gambaran konsep *machine learning* dapat dilihat pada Gambar 2.3.

Dari data tersebut, komputer akan melakukan proses belajar (training) untuk menghasilkan suatu model. Proses belajar ini menggunakan *algoritma machine learning* sebagai penerapan teknik statistika. Model inilah yang menghasilkan informasi, kemudian dapat dijadikan pengetahuan untuk memecahkan suatu permasalahan sebagai proses input-output. Model yang dihasilkan dapat melakukan klasifikasi ataupun prediksi kedepannya.

Untuk memastikan efesiensi model yang terbentuk, data akan dibagi menjadi data latih (*train dataset*) dan data uji (*test dataset*). Pembagian data yang digunakan bervariasi bergantung algoritma yang digunakan. Pada umumnya *train dataset* lebih banyak dari *test dataset*, misalnya dengan rasio 3:1. *Test dataset* digunakan untuk menghitung seberapa efesien model yang dihasilkan untuk melakukan klasifikasi atau prediksi kedepannya yang disebut *test score*. Semakin banyak data yang digunakan, *test score* yang dihasilkan semakin baik. Nilai *test score* bisa berada dalam rentang 0 sampai 1 atau -1 sampai 1.

2.3. *Support Vector Machine* (SVM)

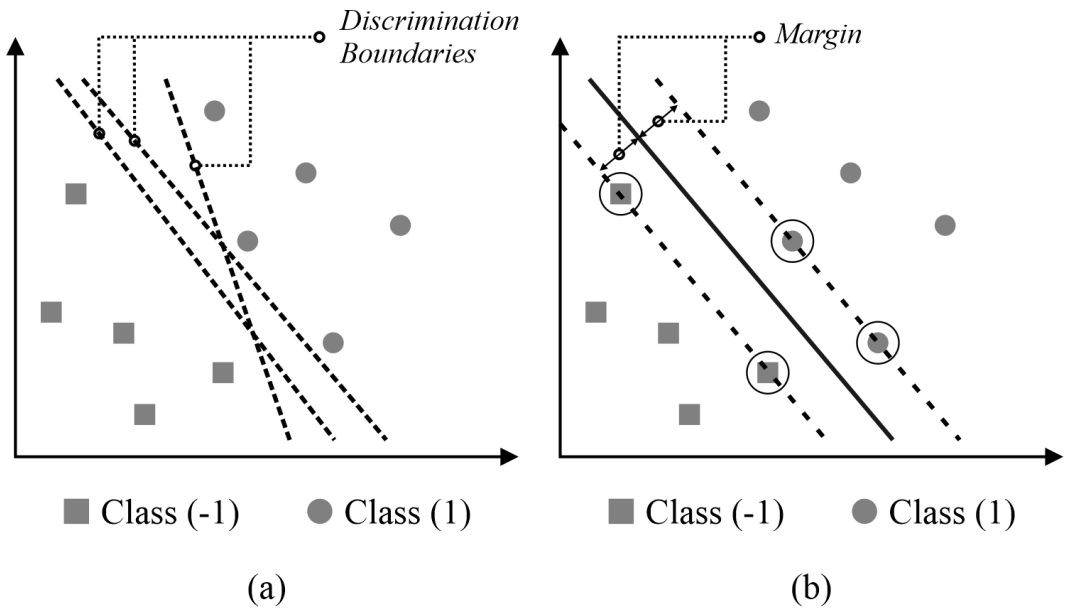
Support Vector Machine (SVM) adalah salah satu metode yang akhir- akhir ini mendapat perhatian. *Support Vector Machine* (SVM) dikembangkan oleh Boser, Guyon, Vapnik, dan pertama kali dipresentasikan pada tahun 1992 di

Annual Workshop on Computational Learning Theory. Konsep dasar SVM sebenarnya merupakan kombinasi harmonis dari teoriteori komputasi yang telah ada puluhan tahun sebelumnya, seperti margin hyperplane (Duda & Hart tahun 1973, Cover tahun 1965, Vapnik 1964, dsb.), kernel diperkenalkan oleh Aronszajn tahun 1950, dan demikian juga dengan konsep-konsep pendukung yang lain. Akan tetapi hingga tahun 1992, belum pernah ada upaya merangkaikan komponen-komponen tersebut. Prinsip dasar SVM adalah *linear classifier*, dan selanjutnya dikembangkan agar dapat bekerja pada problem *non-linear*. dengan memasukkan konsep *kernel trick* pada ruang kerja berdimensi tinggi.[5]

Support Vector Machine (SVM) juga dikenal sebagai teknik pembelajaran mesin (*machine learning*) paling mutakhir setelah pembelajaran mesin sebelumnya yang dikenal sebagai *Neural Network* (NN). Baik SVM maupun NN tersebut telah berhasil digunakan dalam pengenalan pola. Pembelajaran dilakukan dengan menggunakan pasangan data input dan data output berupa sasaran yang diinginkan. Pembelajaran dengan cara ini disebut dengan pembelajaran terarah (*supervised learning*). Dengan pembelajaran terarah ini akan diperoleh fungsi yang menggambarkan bentuk ketergantungan input dan outputnya. Selanjutnya, diharapkan fungsi yang diperoleh mempunyai kemampuan generalisasi yang baik, dalam arti bahwa fungsi tersebut dapat digunakan untuk data input di luar data pembelajaran. diperoleh mempunyai kemampuan generalisasi yang baik, dalam arti bahwa fungsi tersebut dapat digunakan untuk data input di luar data pembelajaran.

[5] Erton F., "Support Vector Machine", Universitas KH. A. Wahab Hasbullah, 2018, hlm.

Konsep SVM dapat dijelaskan secara sederhana sebagai usaha mencari *hyperplane* terbaik yang berfungsi sebagai pemisah dua buah *class* pada *input space*. Gambar 2.4. bagian (a) memperlihatkan beberapa *pattern* yang merupakan anggota dari dua buah *class*: positif (dinotasikan dengan 1) dan negatif (dinotasikan dengan -1). *Pattern* yang tergabung pada *class* negatif disimbolkan dengan kotak, sedangkan *pattern* pada *class* positif, disimbolkan dengan lingkaran. Proses pembelajaran dalam problem klasifikasi diterjemahkan sebagai upaya menemukan garis (*hyperplane*) yang memisahkan antara kedua kelompok tersebut. Berbagai alternatif garis pemisah (*discrimination boundaries*) ditunjukkan pada gambar 2.4. bagian (a).[6]



Gambar 2.4. SVM Berusaha Menemukan *Hyperplane* Terbaik

Hyperplane pemisah terbaik antara kedua *class* dapat ditemukan dengan r *margin hyperplane* dan mencari titik maksimalnya. *Margin* adalah jarak

[6] Erton F., "Support Vector Machine", Universitas KH. A. Wahab Hasbullah, 2018, hlm.

antara *hyperplane* tersebut dengan data terdekat dari masing-masing *class*. Subset data training set yang paling dekat ini disebut sebagai support vector. Garis solid pada Gambar 2.4. bagian (b) menunjukkan *hyperplane* yang terbaik, yaitu yang terletak tepat pada tengah-tengah kedua *class*, sedangkan titik kotak dan lingkaran yang berada dalam lingkaran hitam adalah *support vector*. Upaya mencari lokasi *hyperplane* optimal ini merupakan inti dari proses pembelajaran pada SVM.

2.3.1. Karakteristik SVM

Berikut ini karakteristik yang dimiliki oleh algoritma Support Vector Machine (SVM):[7]

1. Secara prinsip SVM adalah *linear classifier*
2. *Pattern recognition* dilakukan dengan mentransformasikan data pada *input space* ke ruang yang berdimensi lebih tinggi, dan optimisasi dilakukan pada ruang vector yang baru tersebut. Hal ini membedakan SVM dari solusi *pattern recognition* pada umumnya, yang melakukan optimisasi parameter pada ruang hasil transformasi yang berdimensi lebih rendah daripada dimensi *input space*.
3. Menerapkan strategi *Structural Risk Minimization* (SRM)
4. Prinsip kerja SVM pada dasarnya hanya mampu menangani klasifikasi dua class.

[7] Erton F., "Support Vector Machine", Universitas KH. A. Wahab Hasbullah, 2018, hlm.

2.3.2. Kelebihan dan Kelemahan SVM

Dalam memilih solusi untuk menyelesaikan suatu masalah, kelebihan dan kelemahan masing-masing metode harus diperhatikan. Selanjutnya metode yang tepat dipilih dengan memperhatikan karakteristik data yang diolah. Dalam hal SVM, walaupun berbagai studi telah menunjukkan kelebihan metode SVM dibandingkan metode konvensional lain, SVM juga memiliki berbagai kelemahan.[8]

a. Kelebihan SVM

1) Generalisasi

Generalisasi didefinisikan sebagai kemampuan suatu metode untuk mengklasifikasikan suatu *pattern*, yang tidak termasuk data yang dipakai dalam fase pembelajaran metode itu. Vapnik menjelaskan bahwa generalization error dipengaruhi oleh dua faktor: error terhadap training set, dan satu faktor lagi yang dipengaruhi oleh dimensi VC (Vapnik-Chervokinensis). Strategi pembelajaran pada *neural network* dan umumnya metode *machine learning* difokuskan pada usaha untuk meminimalkan error pada training-set. Strategi ini disebut *Empirical Risk Minimization* (ERM). Adapun SVM selain meminimalkan error pada training-set, juga meminimalkan faktor kedua. Strategi ini disebut *Structural*

[8] Erton F., "Support Vector Machine", Universitas KH. A. Wahab Hasbullah, 2018, hlm.

Risk Minimization (SRM), dan dalam SVM diwujudkan dengan memilih hyperplane dengan margin terbesar. Berbagai studi empiris menunjukkan bahwa pendekatan SRM pada SVM memberikan error generalisasi yang lebih kecil daripada yang diperoleh dari strategi ERM pada neural network maupun metode yang lain.

2) *Curse of dimensionality*

Curse of dimensionality didefinisikan sebagai masalah yang dihadapi suatu metode *pattern recognition* dalam mengestimasikan parameter (misalnya jumlah hidden neuron pada neural network, stopping criteria dalam proses pembelajaran dsb.) dikarenakan jumlah sampel data yang relatif sedikit dibandingkan dimensional ruang vektor data tersebut. Semakin tinggi dimensi dari ruang vektor informasi yang diolah, membawa konsekuensi dibutuhkannya jumlah data dalam proses pembelajaran. Pada kenyataannya seringkali terjadi, data yang diolah berjumlah terbatas, dan untuk mengumpulkan data yang lebih banyak tidak mungkin dilakukan karena kendala biaya dan kesulitan teknis. Dalam kondisi tersebut, jika metode itu "terpaksa" harus bekerja pada data yang berjumlah relatif sedikit dibandingkan dimensinya, akan membuat proses estimasi parameter metode menjadi sangat sulit. *Curse of dimensionality* sering dialami

dalam aplikasi di bidang biomedical engineering, karena biasanya data biologi yang tersedia sangat terbatas, dan penyediaannya memerlukan biaya tinggi. Vapnik membuktikan bahwa tingkat generalisasi yang diperoleh oleh SVM tidak dipengaruhi oleh dimensi dari input vector. Hal ini merupakan alasan mengapa SVM merupakan salah satu metode yang tepat dipakai untuk memecahkan masalah berdimensi tinggi, dalam keterbatasan sampel data yang ada.

3) Landasan teori

Sebagai metode yang berbasis statistik, SVM memiliki landasan teori yang dapat dianalisa dengan jelas, dan tidak bersifat kuliah umum.

4) Feasibility

SVM dapat diimplementasikan relative mudah, karena proses penentuan support vector dapat dirumuskan dalam QP problem. Dengan demikian jika kita memiliki library untuk menyelesaikan QP problem, dengan sendirinya SVM dapat diimplementasikan dengan mudah. Selain itu dapat diselesaikan dengan metode sekuensial sebagaimana penjelasan sebelumnya.

b. Kelemahan SVM

SVM memiliki kelemahan atau keterbatasan, antara lain:

1) Sulit dipakai dalam *problem* berskala besar. Skala besar dalam hal ini dimaksudkan dengan jumlah sampel yang diolah.
2) SVM secara teoritik dikembangkan untuk *problem* klasifikasi dengan dua class. Dewasa ini SVM telah dimodifikasi agar dapat menyelesaikan masalah dengan *class* lebih dari dua, antara lain strategi *One versus rest* dan strategi *Tree Structure*. Namun demikian, masing-masing strategi ini memiliki kelemahan, sehingga dapat dikatakan penelitian dan pengembangan SVM pada *multiclass-problem* masih merupakan tema penelitian yang masih terbuka.