IMPLEMENTASI ARTIFICIAL NEURAL NETWORK MULTI LAYER PERCEPTRON PADA DIAGNOSIS PENYAKIT KUCING BERBASIS ANDROID

Mohamad Nurkamal Fauzan[1)]
Prodi/Jurusan D4 Teknik Informatika Politeknik Pos Indonesia
Jln. Sari Asih No. 54 Kode Pos 40151 Bandung, Jawa Barat
[1]m.nurkamal.f@poltekpos.ac.id

ABSTRAK

Kucing merupakan salah satu hewan yang menggemaskan sehingga populer untuk dijadikan hewan peliharaan kesayangan. Pemelihara kucing tentunya sudah memahami apa saja yang harus dilakukan ketika melihat gejala yang tidak normal. Kucing yang tidak terawat dapat mengakibatkan kesehatan yang buruk dan berdampak pada kematian. Untuk melakukan diagnosis awal terhadap penyakit kucing, *artificial neural network* dengan metode *Multi layer Perceptron (MLP)* menghasilkan suatu model dan diimplementasikan pada perangkat android. Hasil pengujian terhadap model menunjukan nilai true positive yang tinggi.

Kata Kunci : *Diagnosis, Artificial Neural Network, Multi Layer Perceptron, android.*

1. PENDAHULUAN

1.1 Latar Belakang Masalah

Kucing merupakan salah satu hewan yang menggemaskan sehingga populer untuk dijadikan hewan peliharaan kesayangan. Pemelihara kucing tentunya sudah memahami apa saja yang harus dilakukan ketika melihat gejala yang tidak normal. Kucing yang tidak terawat dapat mengakibatkan kesehatan yang buruk dan berdampak pada kematian. Virus, bakteri dan parasit dapat berada pada tubuh kucing tanpa sepengetahuan pemiliknya [2] dan mungkin saja dapat menularkan pada pemiliknya [3]. Ada baiknya sebelum akhirnya kucing dibawa ke dokter pemilik sudah memiliki pengetahuan akan gejala penyakit kucing untuk langkah preventif. Dokter akan mendiagnosa penyakit dari gejala-gejala yang ditemukan hingga akhirnya memberikan solusi. Pengetahuan dokter terhadap hal ini akan dimodelkan dan diimplementasikan menjadi suatu aplikasi berbasis android. Sehingga pemilik kucing dapat memiliki pengetahuan akan penyakit kucing hanya lewat aplikasi yang mendiagnosis gejala-gejala yang tampak.

1.2 Identifikasi Masalah

Berdasarkan latar belakang yang telah dikemukakan di atas, masalah dapat diidentifikasi sebagai berikut :

1. Bagaimana membuat model pengetahuan untuk mendiagnosis penyakit kucing.
2. Bagaimana menerapkan solusi dalam bentuk aplikasi android tentang diagnosis penyakit kucing.

1.3 Tujuan

Adapun tujuan dari penelitian ini, yaitu:

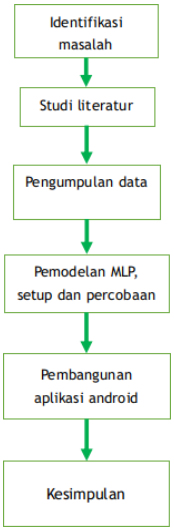
1. Terciptanya model pengetahuan yang dapat memprediksi penyakit kucing.
2. Terciptanya aplikasi android sebagai solusi model tersebut.

1.4 Ruang lingkup

Penelitian ini menitikberatkan pada pengukuran metode dengan dataset yang berasal dari penelitian sebelumnya oleh peneliti yang lain.

2. METODE PENELITIAN

Strategi untuk memecahkan masalah dan tujuan dibuat menjadi beberapa tahapan ditunjukan pada gambar 1.



Gambar 1. Metode penelitian

2. 1. Studi literatur

Studi literatur untuk mendapatkan data set dan klasifikasi [3] berupa gejala dan penyakit Berikut atribut pada gejala pada tabel 1.

Tabel 1 Gejala

Gejala	Deskripsi
G1	Keratinisasi
G2	Gatal gatal
G3	Keropeng
G4	Ketombe
G5	Kutuan
G6	Kurus
G7	Bulu rontok
G8	Anoreksia
G9	Abdomen keras
G10	Muntah
G11	Diare
G12	Perut buncit
G13	Hilang nafsu makan
G14	Ada cacing di fases
G15	Pilek
G16	Bersin bersin
G17	Hidung tersumbat
G18	Badan lemas
G19	Mata berair Anemia
G20	Hidung berair
G21	Ringworm pada kulit
G22	Kulit kemerahan sampai lecet
G23	Jamuran
G24	Lesi berminyak pengganti
G25	Guratan parah pada telinga
G26	Adanya cairan hitam keluar
G27	Telinga terdapat lilin dan bau
G28	Diare campur darah
G29	Feses lembek
G30	Minum banyak
G31	Abdomen sakit
G32	Vaksinasi

Berikut tabel kelas penyakit pada tabel 2.

Tabel 2 Penyakit

Penyakit	Deskripsi
P1	Scabies
P2	Gastritis
P3	Helminthiasis
P4	Rhintis
P5	Dermatophytosis
P6	Dermatitis
P7	Enteritis
P8	Otitis
P9	Sehat

2. 2. Pemilihan Metode

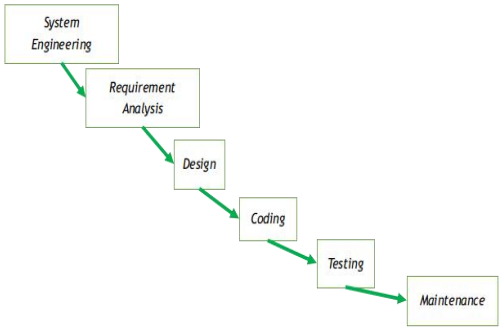
Pemilihan metode berupa *artificial neural network MLP* karena dapat menghasilkan pemodelan yang baik [4]. Data latih menggunakan keseluruhan data dari penelitian sebelumnya [3]. Hidden layer sebanyak 10 yang didapat dari jumlah kelas + 1.

Setelah model terbentuk, langkah berikutnya adalah mengukur *confusion matrix*, akurasi, *latency* pada sistem yang dimiliki penulis dan menyimpan model latih dalam file serial.

Model akan disimpan pada suatu server sehingga memiliki *web service* [5] untuk melayani input dari pengguna atau pemilik kucing. *Web service* pada server ditulis menggunakan Python.

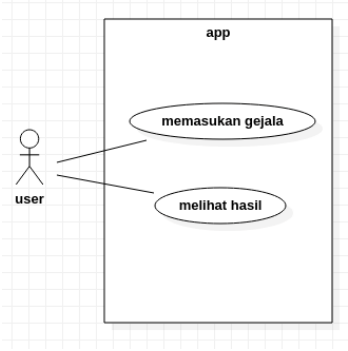
Suatu agensi *marketing social* melakukan survey dan melaporkan data jumlah pengguna mobile dari seluruh dunia. Indonesia termasuk pengguna perangkat mobile terbesar di dunia [1]. Perangkat mobile android dapat menjalankan *supervised machine learning* [8]. Berdasarkan hal tersebut maka aplikasi akan diimplementasikan pada android dengan penyesuaian kebutuhan.

Membuat prototype aplikasi android dengan metode waterfall [9][10] ditunjukan pada gambar 2. Kode program android ditulis meggunakan Java.



Gambar 2. Metode waterfall

Sebelum melakukan pengkodean langkah yang dilakukan adalah perancangan untuk itu digunakan UML. Berikut *usecase* diagram yang menunjukan relasi antara aktor dan aplikasi pada gambar 3.

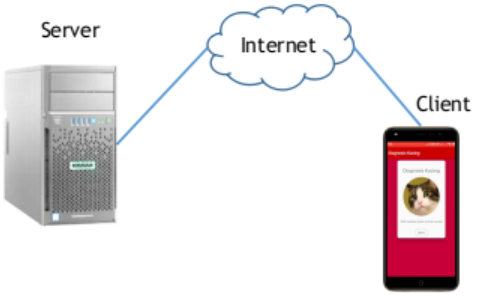


Gambar 3. *Use case*

Berikut class diagram aplikasi pada *web service* pada gambar 4.

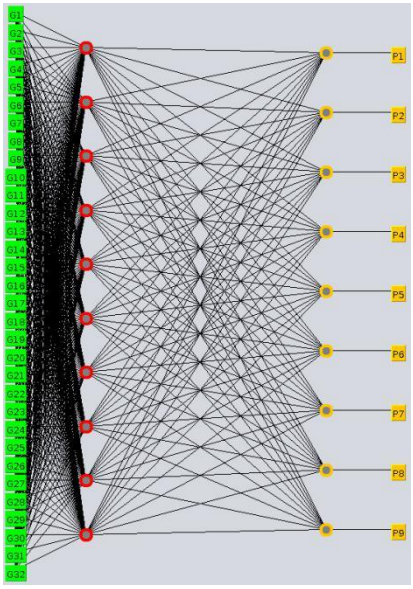
Gambar 4. Class diagram

Berikut arsitektur sistem yang dibangun seperti pada gambar 5, *web service* yang diletakan di server internet meyebabkan pengguna dapat berinteraksi dengan server.



Gambar 5. Arsitektur sistem

3. HASIL DAN PEMBAHASAN



Gambar 6. Neural Network

Pada gambar 6. merupakan visualisasi neural network dan pembangkitan fungsi Sigmoid [7].

Model didapatkan dari keseluruhan dataset dan menghasilkan nilai confidence sebesar 100% dan waktu latency latih selama 0.01398777961730957 detik. Gambar 7 menunjukan kode untuk mendapatkan pemodelan.

```
In [45]: model = MLPClassifier(solver='lbfgs', alpha=1e-5,
         ...                    hidden_layer_sizes=(10), random_state=1)
         start = time.time()
         model.fit(x, y)
         stop = time.time()
         print(f"Training time: {stop - start}s")

         Training time: 0.01398777961730957s
```

Gambar 7. Kode pemodelan MLP

Tabel 3. Confusion matrix

P1	P2	P3	P4	P5	P6	P7	P8	P9	
1	0	0	0	0	0	0	0	0	P1
0	1	0	0	0	0	0	0	0	P2
0	0	1	0	0	0	0	0	0	P3
0	0	0	1	0	0	0	0	0	P4
0	0	0	0	1	0	0	0	0	P5
0	0	0	0	0	1	0	0	0	P6
0	0	0	0	0	0	1	0	0	P7
0	0	0	0	0	0	0	1	0	P8
0	0	0	0	0	0	0	0	1	P9

Hasil *confusion matrix* pada tabel 3 menunjukan bahwa prediksi kelasifikasi dengan tepat terpetakan sesuai data latih.

Tabel 4. Akurasi

True Positive	False Positive
1	0
1	0
1	0
1	0
1	0
1	0
1	0
1	0
1	0

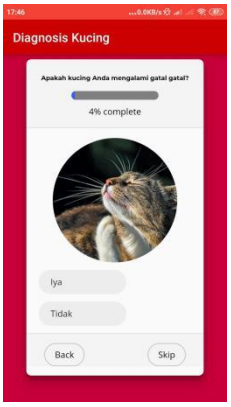
Seluruh nilai model berisi 1 dalam memprediksi kelas *True positive* sedangkan *false positive* seluruhnya bernilai 0 ditunjukan pada hasil tabel 4.

Lama latency waktu memprediksi dengan input [0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1] adalah 0.0006799697875976562 detik dengan output kelas P8.

Hasil *confusion matrix* menunjukan terjadinya *overfitting* [6]. Hal ini dapat disebabkan oleh dataset yang terlalu sedikit untuk dilakukan teknik *split* ataupun *cross validation*.

Hasil implementasi pada aplikasi android. Pengguna akan melihat tampilan awal (gambar 7). Setelah pengguna menekan tombol mulai maka pengguna akan mulai mengisi gejala yang ada pada kucing (gambar 8) dan setelah semua pertanyaan terisi maka sistem akan memprediksi dan tampil pada layar diagnosis (gambar 9).

Gambar 7. Tampilan awal



Gambar 8. Pengguna mengisi pertanyaan

Gambar 9. Hasil diagnosis

4. KESIMPULAN DAN SARAN

4.1 Kesimpulan

Dari analisis data dan pembahasan yang telah dilakukan, dapat ditarik kesimpulan yang sekaligus menjawab rumusan masalah penelitian sebagai berikut:

1. Model pengetahuan menggunakan *artificial neural network* MLP berhasil dibangun.
2. Aplikasi android diagnosis penyakit kucing berjalan dengan baik.

4.2 Saran

Berikut beberapa saran dari hasil laporan.

1. Untuk menghindari *overfitting* perlu ditambahkan dataset yang lebih banyak sehingga dapat dilakukan *cross validation* atau pada kondisi tertentu melakukan strategi yang lain [6].
2. Perlu dilakukan pembaharuan model pada kondisi tertentu seperti perubahan pada gejala dan penyakit.

DAFTAR PUSTAKA

[1] Wijaya K.K., 2015. Berapa jumlah pengguna website, mobile, dan media sosial di Indonesia?. id.techinasia.com. [Online] Tersedia di: https://id.techinasia.com /laporan-pengguna-website-mobile- media-sosial-indonesia. [Diakses Pada tanggal 10 Maret 2017]

[2] Purnomo, D., Irawan, B., & Brianorman, Y. (2017). Sistem Pakar Diagnosa Penyakit Pada Kucing Menggunakan Metode Dempster-Shafer Berbasis Android. Jurnal Coding Sistem Komputer Untan, 05(1), 23–32.

[3] Nugraha, A. A. S., Hidayat, N., & Fanani, L. (2018). Sistem Pakar Diagnosis Penyakit Kucing Menggunakan Metode Naive Bayes – Certainty Factor Berbasis Android. Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer (J-PTIIK) Universitas Brawijaya, 2(2), 650–658.

[4] Polezer, G., Tadano, Y. S., Siqueira, H. V., Godoi, A. F. L., Yamamoto, C. I., de André, P. A., … Godoi, R. H. M. (2018). Assessing the impact of PM2.5 on respiratory disease using artificial neural networks. Environmental Pollution, 235, 394–403. https://doi.org/10.1016/j.envpol.2017.12.111

[5] Ghani, I., Wan-Kadir, W. M. N., & Mustafa, A. (2019). Web service testing techniques: A systematic literature review. International Journal of Advanced Computer Science and Applications, 10(8), 443–458. https://doi.org/10.14569/ijacsa.2019.0100858

[6] Ying, X. (2019). An Overview of Overfitting and its Solutions. In Journal of Physics: Conference Series (Vol. 1168). Institute of Physics Publishing. https://doi.org/10.1088/1742-6596/1168/2/022022

[7] Lee, D. H., Kim, Y. T., & Lee, S. R. (2020). Shallow landslide susceptibility models based on artificial neural networks considering the factor selection method and various non-linear activation functions. Remote Sensing, 12(7). https://doi.org/10.3390/rs12071194

[8] Nigam, C., Narang, M., Chaurasia, N., & Nanda, A. (2020). M-Health Android Application Using Firebase, Google APIs and Supervised Machine Learning. In Advances in Intelligent Systems and Computing (Vol. 1070, pp. 670–681). Springer. https://doi.org/10.1007/978-3-030-32523-7_50

[9] Suryani, K., Khairudin, & Syahmaidi, E. (2017). Online tracer study of Bung Hatta University. International Journal of GEOMATE, 13(7), 20–27. https://doi.org/10.21660/2017.37.TVET011

[10] Setiawan, D., Suratno, T., & Lutfi, L. (2018). Analisis, Desain dan Rancang Bangun Sistem Informasi Akademik Mobile Berbasis Android. ELKHA, 10(2), 73. https://doi.org/10.26418/elkha.v10i2.28272