E-ISSN : 2540 - 8984

JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)
Volume 07, Nomor 04, Desember 2022 : 1281 - 1293

KLASIFIKASI MULTI-LABEL MENGGUNAKAN METODE MULTI-LABEL K-NEAREST NEIGHBOR (ML-KNN) PADA PENYAKIT KANKER SERVIKS

Erisa Rizkyani[1], Iin Ernawati[2], Nurul Chamidah[3]
Program Studi Informatika / Fakultas Ilmu Komputer, Universitas Pembangunan Nasional Veteran Jakarta
Jl. RS. Fatmawati , Pondok Labu, Jakarta Selatan, DKI Jakarta, 12450, Indonesia
Email: erisar@upnvj.ac.id[1], iin_ernawati@yahoo.com[2], nurul.chamidah@upnvj.ac.id[3]

ABSTRAK

Berdasarkan data statistik GLOBOCAN 2020, kanker serviks menempati urutan ke-8 penyakit kanker yang banyak diderita perempuan di seluruh dunia dengan jumlah kasus sebanyak 604.127 kasus dengan angka kematian mencapai 341.831 jiwa. Sedangkan di Indonesia tercatat penderita penyakit kanker serviks berada di urutan ke-2 dengan jumlah kasus sebanyak 36.633 kasus dengan angka kematian mencapai 21.003 jiwa. Multi-Label K-Nearest Neighbor (ML-KNN) merupakan salah satu adaptive algorithm yang dapat digunakan untuk menyelesaikan kasus klasifikasi multi-label. Pada penelitain ini menggunakan dataset yang diperoleh dari website UCI Machine Learning. Pada dataset tersebut akan dilakukan pra-proses data dengan menghapus missing value, mengecek duplicate data, mengecek tipe data, dan melakukan resample data berupa oversampling pada label Biopsy karena data kelas 1 dan 0 yang tidak seimbang. Selanjutnya data dibagi menjadi data latih dan data uji dengan perbandingan 80:20. Pada data latih, dicari kedekatannya dengan nilai k yang sudah ditentukan yaitu K=1, K=3, K=5, K=7, dan K=9. Diperoleh hasil evaluasi performa terbaik yaitu saat nilai K=5 yang memperoleh nilai hamming loss sebesar 3,59%, akurasi sebesar 93%, precision weighted sebesar 93%, recall weighted sebesar 96%, dan f1-score weighted sebesar 94%.

***Kata Kunci**: Klasifikasi, Kanker Serviks, Multi-Label K-Nearest Neighbor (ML-KNN).*

ABSTRACT

Based on GLOBOCAN 2020 statistical data, cervical cancer ranks 8th most cancers suffered by women worldwide with 604,127 cases and 341,831 deaths. Meanwhile, in Indonesia, cervical cancer sufferers are in 2nd place with 36,633 cases with a death rate of 21,003 people. Multi-Label K-Nearest Neighbor (ML-KNN) is an adaptive algorithm that can be used to solve multi-label classification cases. This research uses a dataset obtained from the UCI Machine Learning website. The dataset will be pre-processed by deleting missing values, checking for duplicate data, checking data types, and resample data in the form of oversampling on the Biopsy label due to unbalanced class 1 and 0 data. Furthermore, the data is divided into training data and test data with a ratio of 80:20. In the training data, look for its proximity to the predetermined k value, namely K=1, K=3, K=5, K=7, and K=9. The best performance evaluation results were obtained when the value of K = 5 which obtained a hamming loss value of 3.59%, accuracy of 93%, precision weighted of 93%, recall weighted of 96%, and f1-score weighted of 94%.

***Keywords**: Classification, Cervical Cancer, Multi-Label K-Nearest Neighbor (ML-KNN).*

I. PENDAHULUAN

KANKER masih menjadi penyakit yang paling banyak menyebabkan kematian di seluruh dunia. Kanker serviks termasuk dalam jenis kanker yang paling mematikan dan banyak diderita oleh perempuan [1]. Berdasarkan data statistik GLOBOCAN 2020, kanker serviks menempati urutan ke-8 penyakit kanker paling banyak diderita perempuan di seluruh dunia dengan jumlah 604.127 kasus dengan angka kematian mencapai 341.831 jiwa. Sedangkan di Indonesia tercatat penderita penyakit kanker serviks berada di urutan ke-2 dengan jumlah 36.633 kasus dengan angka kematian mencapai sebesar 21.003 jiwa [2]. Dapat terjadi peningkatan pada angka tersebut apabila tidak adanya deteksi dini terhadap penyakit kanker serviks. Oleh karena itu perlu adanya informasi terkait pendeteksian penyakit kanker serviks yang dapat dilakukan sesegera mungkin.

Kanker serviks merupakan kanker yang menginfeksi bagian organ reproduksi wanita yaitu leher rahim/serviks yang terletak di antara rahim/uterus dengan liang senggama atau vagina [3]. Kanker serviks dapat dideteksi dengan menggunakan beberapa test seperti test *Hinselmann*, *Schiller*, *Citology*, dan *Biopsy*. *Hinselmann* atau biasa diketahui sebagai *Colcoscopy*, adalah prosedur diagnosis secara medis untuk memeriksa bagian serviks termasuk vagina dan vulva yang dilihat menggunakan alat khusus bernama kolposkop. Tes *schiller* merupakan tes medis yang memberikan larutan yodium dengan cara dioleskan pada serviks untuk mendiagnosis kanker

E-ISSN : 2540 - 8984

JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)
Volume 07, Nomor 04, Desember 2022 : 1281 - 1293

serviks. Setelah diolesi, jaringan akan berubah menjadi coklat apabila jaringan tersebut normal dan akan berwarna putih atau kuning apabila tidak normal. *Citology* atau biasa diketahui sebagai *Pap Smear* dilakukan dengan cara mengambil sampel sel dari serviks atau leher rahim. Setelah itu sel yang telah diambil dilihat menggunakan mikroskop untuk menentukan apakah sel tersebut, memiliki sifat normal, pra-kanker (calon kanker), atau bahkan sudah bersifat kanker. *Biopsy* merupakan prosedur pembedahan beberapa jaringan kecil yang dikeluarkan dari serviks [4].

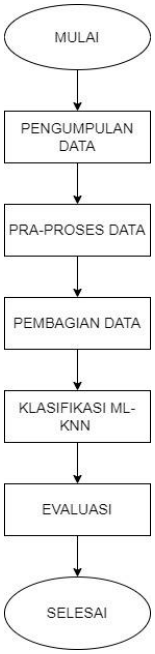
Multi-Label K-Nearest Neighbor (ML-KNN) termasuk dalam salah satu *adaptive algorithm* yang digunakan dalam melakukan klasifikasi multi-label. ML-KNN merupakan adaptasi dari algoritma K-NN, ML-KNN lebih fleksibel serta fungsional dibandingkan dengan algoritma tersebut. Keunggulan algoritma ML-KNN dengan algoritma klasifikasi multi-label lainnya yaitu pemodelannya yang lebih sederhana, efisien, dan memiliki performa yang lebih baik [5].

Ada beberapa penelitian terdahulu yang dijadikan rujukan dari penelitian ini yaitu, penelitian [6] yang menghasilkan kesimpulan bahwa penggunaan metode ML-KNN dengan target multi-label menghasilkan nilai evaluasi performa yang lebih baik jika dibandingkan dengan penggunaan metode *decision tree*, *extra tree*, dan K-NN. Penelitian [4] dengan menggunakan *dataset* yang sama memperoleh kesimpulan bahwa algoritma K-NN lebih baik jika dibandingkan dengan algoritma *extra trees* dengan akurasi yang diperoleh dengan menggunakan algoritma K-NN sebesar 89%.

Berdasarkan penjelasan di atas maka akan dilakukan penelitian guna mengklasifikasi data pasien penyakit kanker serviks menggunakan metode *Multi-Label K-Nearest Neighbor* (ML-KNN) dan menganalisis bagaimana performa dari metode tersebut. Pada penelitian ini *dataset* yang digunakan berasal dari *website UCI Machine Learning*. Hal ini berdasarkan dari penelitian [4] yang digunakan sebagai rujukan yang mendeteksi data pasien yang menderita penyakit kanker serviks menggunakan metode K-NN. Pada penelitian [4] *dataset* yang digunakan memiliki data kelas 1 dan 0 yang tidak seimbang pada target labelnya, sehingga pada penelitian ini dilakukan *resample data* berupa *oversampling* kelas 1 pada kolom *Biopsy* lalu dilakukan klasifikasi menggunakan metode ML-KNN karena berdasarkan penelitian [6] diperoleh bahwa performa metode ML-KNN lebih baik daripada metode K-NN. Tujuan dari penelitian ini yaitu untuk melakukan klasifikasi menggunakan metode ML-KNN penyakit kanker serviks dan untuk mengetahui performa dari klasifikasi menggunakan metode ML-KNN pada penyakit kanker serviks.

II. METODOLOGI PENELITIAN

Tahapan kerja dari penelitian ini adalah seperti berikut ini:



Gambar. 1. Tahapan penelitian ini dimulai dari pengumpulan data, pra-proses data, pembagian data, klasifikasi ML-KNN dan evaluasi.

E-ISSN : 2540 - 8984

JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)
Volume 07, Nomor 04, Desember 2022 : 1281 - 1293

A. Pengumpulan Data

Dataset yang digunakan diperoleh dari *website University California Irvine (UCI) Machine Leaning Repository* yaitu *Cervical Cancer (Risk Factor)* dataset [7] yang berisikan data pasien '*Hospital Universitario de Caracas*' yang bertempat di Caracas, Venezuela. *Dataset* terdiri dari informasi demografi, kebiasaan dan rekam medis dari 858 pasien dengan 36 atribut. *Dataset* ini juga memiliki multi-label yang setiap pasiennya dapat terlibat dalam beberapa label di waktu yang bersamaan. Tipe data pada tiap kolom yang terdapat pada *dataset* yang digunakan dapat dilihat pada table I berikut ini.

TABEL I
TIPE DATA PADA TIAP KOLOM PADA *DATASET*

Kolom	Tipe Data	Kolom	Tipe Data	Kolom	Tipe Data
Age	*Integer*	*STDs (number)*	*Boolean*	*STDs: HPV*	*Boolean*
Number of sexual partners	*Integer*	*STDs: condylomatosis*	*Boolean*	*STDs: Number of diagnosis*	*Integer*
First sexual intercourse	*Integer*	*STDs: cervical condylomatosis*	*Boolean*	*STDs: Time since first diagnosis*	*Integer*
Number of pregnacies	*Integer*	*STDs: vaginal condylomatosis*	*Boolean*	*STDs: Time since last diagnosis*	*Integer*
Smokes	*Boolean*	*STDs: vulvo-perineal condylomatosis*	*Boolean*	*Dx: Cancer*	*Boolean*
Smokes (years)	*Integer*	*STDs: syphilis*	*Boolean*	*Dx: CIN*	*Boolean*
Smokes (packs/year)	*Boolean*	*STDs: pelvic inflammatory disease*	*Boolean*	*Dx: HPV*	*Boolean*
Hormonal Contraceptives	*Boolean*	*STDs: genital herpes*	*Boolean*	*Dx*	*Boolean*
Hormonal Contraceptives (years)	*Integer*	*STDs: molluscum contagiosum*	*Boolean*	*Hinselmann (Label)*	*Boolean*
IUD	*Boolean*	*STDs: AIDS*	*Boolean*	*Schiller (Label)*	*Boolean*
IUD (years)	*Integer*	*STDs: HIV*	*Boolean*	*Citology (Label)*	*Boolean*
STDs	*Boolean*	*STDs: Hepatitis B*	*Boolean*	*Biopsy (Label)*	*Boolean*

B. Pra-proses data

Setelah dilakukan pengambilan data, dilakukan tahapan pra-proses data. Pada tahap ini dilakukan beberapa proses yaitu pemeriksaan *missing value*, pemeriksaan *duplicate data*, pemeriksaan tipe data, dan *resample data*.

Pemeriksaan *missing value* dilakukan agar mengetahui ada atau tidaknya data yang hilang pada *dataset* yang digunakan. *Missing value* dapat mempengaruhi hasil penelitian sehingga akurasi yang dihasilkan tidak optimal. Seperti penelitian yang telah dilakukan sebelumnya [8] untuk kolom yang memiliki *missing value* lebih dari 50% dihilangkan dan untuk tiap baris yang memiliki *missing value* lebih dari 2 atribut juga dihilangkan.

Pemeriksaan *duplicate data* dilakukan untuk mengetahui pada *dataset* yang digunakan terdapat data yang duplikat atau tidak. Hal ini karena data yang duplikat dapat mempengaruhi hasil akhir dari penelitian yang dilakukan. Adanya data yang duplikat juga menyebabkan penyimpanan yang digunakan menjadi lebih besar dan pemrosesan yang berjalan akan menjadi lebih lama dibandingkan dengan tidak adanya data yang duplikat. Selanjutnya dilakukan pemeriksaan tipe data, pada tahap ini dilakukan pemeriksaan tipe data pada tiap kolom apakah tipe data pada tiap kolom tersebut sudah sesuai atau belum dengan deskripsi yang ada di *website UCI Machine Learning*.

Pada *dataset* yang digunakan terdapat 668 baris data, dengan empat label target yang setiap labelnya memiliki jumlah kelas yang tidak seimbang (*Imbalanced data*), hal ini diketahui dari rasio ketidakseimbangan pada masing-masing label. Rasio ketidakseimbangan dihitung dan menghasilkan rasio kelas 0 dan kelas 1 pada label *Hinselmann* sebesar 21,27; label *Schiller* sebesar 9,6; label *Citology* sebesar 16,13; dan label *Biopsy* sebesar 13,84. Sehingga dilakukan *resample data* berupa *oversampling* kelas 1 pada kolom *Biopsy*.

C. Pembagian Data

Selanjutnya *dataset* yang telah di pra-proses sebelumnya akan dibagi menjadi 2 yaitu data latih dan data uji. Pada penelitian ini digunakan rasio 80:20 sehingga data latih sebanyak 80% dan data uji sebanyak 20%.

D. Klasifikasi ML-KNN

Selanjutnya dilakukan klasifikasi dengan menggunakan metode *Multi-Label K-Nearest Neighbor* (ML-KNN). ML-KNN yang merupakan penurunan dari algoritma *K-Nearest Neighbor*, ML-KNN pertama kali diusulkan oleh Zhi-hua Zhou dan Minling Zhang. Algoritma ML-KNN adalah algoritma klasifikasi multi-label yang mewarisi

E-ISSN : 2540 - 8984

JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)
Volume 07, Nomor 04, Desember 2022 : 1281 - 1293

stabilitas dan efisiensi yang tinggi dari algoritma K-NN, dan cocok untuk masalah klasifikasi multi-label dengan distribusi data yang kacau atau data yang memiliki lebih dari satu kelas atau label secara bersamaan [9]. ML-KNN masih perlu mencari k tetangga terdekat dari sampel uji dalam sampel pelatihan. Berbeda dengan K-NN, digunakan metode *bayesian* untuk menghitung *probabilitas* 0 atau 1 untuk setiap label data dalam proses penentuan label sampel uji dan menggunakan prinsip *Maximum A Posteriori* (MAP) untuk menentukan nilai akhir label. Langkah-langkah algoritma ML-KNN adalah sebagai berikut:

1. Hitung *prior probability* $\mathrm{P}(H_1^l)$ dari tiap label dari data latih yang nilainya 1. Setelah itu hitung *prior probability* $\mathrm{P}(H_0^l)$ dari tiap label dari data latih yang nilainya 0. Dengan persamaan sebagai berikut.

$$P(H_1^l) = \frac{s+\sum_{j=1}^{n} \vec{y}_{x_i}}{s\times 2+n} \quad (1)$$

$$P(H_0^l) = 1 - P(H_1^l) \quad (2)$$

   Dengan $\vec{y}_{x_i}$ adalah set label data ke-I, $P(H_1^l)$ adalah *prior probability* dari data latih yang berlabel l, $P(H_0^l)$ adalah *prior probability* dari data latih yang tidak berlabel l, s adalah parameter *smoothing* (bernilai 1), dan n merupakan banyaknya data latih.
2. Tentukan nilai K sebagai banyaknya jumlah tetangga terdekat lalu hitung jarak antara data uji dengan data latih menggunakan rumus *Euclidean distance*. Jika x adalah data uji, y adalah data latih, dan m adalah jumlah atribut, maka rumus *Euclidean distance* sebagai berikut.

$$E(x,y) = \sqrt{\sum_{i=1}^{m}(x_i - y_i)^2} \quad (3)$$

3. Hitung vektor keanggotaan untuk mengetahui banyaknya data tetangga dari data uji yang berlabel l dengan persamaan berikut ini.

$$\vec{C}_x(l) = \sum_{\alpha\in N(x)} \vec{y}_\alpha(l) \quad (4)$$

   Dengan $\vec{C}_x(l)$ sebagai banyaknya data tetangga dari x yang berlabel l dan $N(x)$ sebagai himpunan k tetangga terdekat dari x.
4. Hitung posterior probabilitas dari tiap label pada data tetangga yang bernilai 1 dan 0. Dengan persamaan berikut ini.

$$P(E_j^l|H_1^l) = \frac{s+c[j]}{s\times(k+1)+\sum_{j=1}^{k} c[j]} \quad (5)$$

$$P(E_j^l|H_0^l) = \frac{s+c\prime[j]}{s\times(k+1)+\sum_{j=1}^{k} c\prime[j]} \quad (6)$$

   Dengan E_j^l sebagai jumlah data j tetangga yang berlabel l, c[j] sebagai jumlah data tetangga yang bernilai 1 di label j, c'[j] sebagai jumlah data tetangga yang bernilai 0 di label j, dan k sebagai nilai k yang sudah ditentukan.
5. Setelah itu untuk mendapatkan label pada data uji $\vec{y}_t(l)$, dicari menggunakan *Maximum a Posterior* (MAP) berdasarkan *posterior probability* yang telah diperoleh sebelumnya. Label untuk data uji diputuskan berdasarkan nilai MAP yang lebih besar. $\vec{y}_t(l)$ dapat dicari dengan persamaan sebagai berikut.

$$\vec{y}_t(l) = \arg max_{b\in\{1,0\}} P(H_b^l|E^l_{\vec{C}_x(l)}) \quad (7)$$

   Menggunakan *bayesian rule*, persamaan 2.8 dapat dituliskan menjadi.

$$\vec{y}_t(l) = \arg max_{b\in\{1,0\}} \frac{P(H_b^l).P(E^l_{\vec{C}_x(l)}|H_b^l)}{P(E^l_{\vec{C}_x})}$$

$$= \arg max_{b\in\{1,0\}} P(H_b^l).P(E^l_{\vec{C}_x(l)}|H_b^l) \quad (8)$$

   Dengan $E^l_{\vec{C}_x(l)}$ sebagai jumlah tetangga terdekat yang berlabel l dan $P(H_b^l)$ sebagai *prior probability* dari data yang bernilai 0 atau 1 pada label l.

E-ISSN : 2540 - 8984

JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)
Volume 07, Nomor 04, Desember 2022 : 1281 - 1293

E. Evaluasi

Pada klasifikasi multi-label, dibutuhkan pengukuran performa yang berbeda dengan pengukuran performa pada klasifikasi label tunggal. Dalam klasifikasi multi-label, kinerja tiap label target harus diperhitungkan [10]. Terdapat beberapa perhitungan evaluasi untuk klasifikasi multi-label, yaitu dengan melihat nilai dari *hamming loss* dan akurasinya. Selain itu juga menggunakan *confusion matrix* untuk dicari nilai *precision*, *recall*, dan *F-Measure/F1-Score*. Untuk nilai *precision*, *recall*, dan *f1-score* terdapat nilai rata-rata yang berbeda yaitu *micro*, *macro*, dan *weighted*. Rata-rata *micro* digunakan ketika ada kebutuhan untuk menimbang setiap kejadian atau prediksi secara merata. Rata-rata *macro* digunakan ketika semua label perlu diperlakukan sama untuk mengevaluasi kinerja keseluruhan pengklasifikasi dan rata-rata *weighted* digunakan ketika terjadi ketidakseimbangan kelas [11].

Hamming loss merupakan perhitungan yang dilakukan untuk mengetahui banyaknya kesalahan pada proses klasifikasi terhadap data yang diuji. *Hamming loss* direpresentasikan dengan *Hloss(h)* yang bila nilainya semakin kecil maka performa dari klasifikasi yang telah dilakukan semakin baik [4].

$$hloss(h) = \frac{1}{NL}\sum_{i=1}^{N}\sum_{i=1}^{L}\left|I(y_j^{(i)} \neq \hat{y}_j^{(i)})\right| \quad (10)$$

Dengan N sebagai jumlah banyaknya data, L sebagai jumlah banyaknya label, $y_j^{(i)}$ sebagai label yang benar untuk data ke i dan kelas ke j, dan $\hat{y}_j^{(i)}$sebagai label yang diprediksi untuk data ke i dan kelas ke j.

Akurasi merupakan perhitungan untuk mengetahui persentase label yang diklasifikasikan dengan benar terhadap jumlah label untuk setiap data. Akurasi dengan nilai yang lebih tinggi menunjukkan bahwa performa model klasifikasi yang digunakan lebih baik [12].

$$Accuracy = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{Y_i \cap \hat{Y}}{Y_i \cup \hat{Y}_i}\right| \quad (11)$$

Dengan n sebagai banyaknya sampel, Y_i sebagai label yang benar untuk data ke i, dan $\hat{Y}_i$ sebagai label yang diprediksi untuk data ke i.

Confusion matrix memberikan ringkasan dari semua prediksi yang dibuat dan dibandingkan dengan nilai aktual yang diharapkan. Dengan menggunakan *confusion matrix*, dapat terlihat dengan jelas prediksi mana yang salah dan jenis kesalahan apa yang dibuat [13]. *Confusion matrix* dapat digunakan untuk menampilkan hasil evaluasi dari pemodelan algoritma. Karena pada *dataset* yang digunakan memiliki ketidakseimbangan kelas pada ke-4 kolom target label, maka evaluasinya menggunakan rata-rata *weighted* yang persamaan *precision*, *recall*, dan *f1-score*-nya seperti berikut ini.

$$Precision\ Weighted = \frac{\sum_{i=1}^{n}|y_i|\frac{TP_i}{TP_i+FP_i}}{\sum_{i=1}^{n}|y_i|} \quad (12)$$

$$Recall\ Weighted = \frac{\sum_{i=1}^{n}|y_i|\frac{TP_i}{TP_i+FN_i}}{\sum_{i=1}^{n}|y_i|} \quad (13)$$

$$F1-Score\ Weighted = \frac{\sum_{i=1}^{n}|y_i|\frac{2TP_i}{2TP_i+FP_i+FN_i}}{\sum_{i=1}^{n}|y_i|} \quad (14)$$

Dengan TP adalah *True Positive*, FP adalah *False Positive*, FN adalah *False Negative*, n sebagai banyaknya label, dan yi sebagai jumlah bobot (TP + FN) pada label ke-i.

III. HASIL DAN PEMBAHASAN

A. Pra-proses Data

Pada *dataset* yang digunakan terdapat beberapa baris yang memiliki *missing value* lebih dari 2 atribut sehingga baris-baris tersebut dihilangkan atau dihapus menggunakan *tools Microsoft excel*. Saat dilakukan pemeriksaan kembali terhadap *dataset* yang digunakan, ternyata masih terdapat *missing value* pada beberapa kolom, sehingga dilakukan penghilangan/penghapusan terhadap kolom yang memiliki *missing value* lebih dari 50% dari total jumlah baris *dataset*.

E-ISSN : 2540 - 8984

JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)
Volume 07, Nomor 04, Desember 2022 : 1281 - 1293

Setelah menghilangkan *missing value* dari *dataset*, tahap selanjutnya yaitu melakukan pemeriksaan *duplicate data* pada tiap baris *dataset*. Setelah dilakukan pemeriksaan *duplicate data*, tidak ditemukan adanya *duplicate row data* pada *dataset* yang digunakan.

Selanjutnya dilakukan pemeriksaan tipe data pada *dataset* yang digunakan, terdapat beberapa kolom yang datanya memiliki tipe data yang tidak sesuai dengan deskripsi yang ada di *UCI Machine Learning* karena *google collaboration* kurang tepat saat membaca tipe data pada tiap kolom dari *dataset* yang di-*import* dari *google sheet*. Sehingga pada kolom-kolom tersebut diubah tipe datanya agar menyesuaikan dengan yang ada di *UCI Machine Learning*. Sehingga *dataset* yang digunakan ada sebanyak 668 data pasien dengan 34 atribut yang 4 di antaranya merupakan target label. Berikut ini data yang digunakan seperti pada table II.

TABEL II
DATASET

Age	*Number of Sexual Partners*	*First Sexual Intercourse*	*Number of Pregnacies*	*Smokes*	*Smokes (years)*	…	*Hinselmann*	*Schiller*	*Citology*	*Biopsy*
18	4	15	1	0	0	…	0	0	0	0
15	1	14	1	0	0	…	0	0	0	0
52	5	16	4	1	37	…	0	0	0	0
46	3	21	4	0	0	…	0	0	0	0
42	3	23	2	0	0	…	0	0	0	0
…	…	…	…	…	…	…	…	…	…	…
34	3	18	0	0	0	…	0	0	0	0
32	2	19	1	0	0	…	0	0	0	0
25	2	17	0	0	0	…	0	0	1	0
33	2	24	2	0	0	…	0	0	0	0
29	2	20	1	0	0	…	0	0	0	0

Setelah dilakukan pemeriksaan *missing value*, *duplicate data*, dan tipe data, tahap selanjutnya dilakukan *resample data*. *Resample data* dilakukan karena dataset yang digunakan memiliki rasio data kelas 0 dan kelas 1 pada tiap labelnya tidak seimbang dan pada penelitian ini dilakukan *oversampling* kelas 1 pada kolom *Biopsy*.

B. Klasifikasi ML-KNN

Dataset yang telah di pra-proses sebelumnya akan dibagi menjadi 2 yaitu data latih dan data uji. Pada penelitian ini digunakan rasio 80:20 sehingga data latih sebesar 80% yaitu sebanyak 996 data dan data uji sebesar 20% yaitu sebanyak 250 data. Selanjutnya pada data latih dilakukan pelatihan metode klasifikasi multi-label dengan menggunakan algoritma *Multi Label K-Nearest Neighbor* (ML-KNN). Data latih tersebut dicari kedekatan setiap data yang dihitung menggunakan rumus *euclidean distance*. Kedekatan setiap data dicari dengan nilai k yang sudah ditentukan yaitu K= 1, K = 3, K = 5, K = 7, dan K = 9.

Cara kerja dari algoritma ML-KNN yang digunakan dapat dilihat pada contoh yang menggunakan sebanyak 10 data latih yang telah dibagi. Selanjutnya akan dilakukan perhitungan jarak antara ke-10 data latih dengan data uji. Data latih yang digunakan seperti yang terlihat pada tabel III.

TABEL III
DATA LATIH

Age	*Number of Sexual Partners*	*First Sexual Intercourse*	*Number of Pregnacies*	*Smokes*	*Smokes (years)*	…	*Hinselmann*	*Schiller*	*Citology*	*Biopsy*
20	1	18	1	0	0	…	1	1	0	1
19	1	17	1	0	0	…	0	0	0	0
25	3	17	2	0	0	…	1	1	1	1
…	…	…	…	…	…	…	…	…	…	…
21	1	17	2	0	0	…	1	1	0	1
28	2	19	2	0	0	…	1	1	0	1
26	3	19	2	0	0	…	1	1	1	0

Selanjutnya dilakukan penentuan label dari 1 contoh data uji yang belum diketahui labelnya, seperti yang terlihat pada tabel IV.

TABEL IV
DATA UJI

Age	*Number of Sexual Partners*	*First Sexual Intercourse*	*Number of Pregnacies*	*Smokes*	*Smokes (years)*	…	*Hinselmann*	*Schiller*	*Citology*	*Biopsy*
28	3	15	3	0	0	…				

E-ISSN : 2540 - 8984

JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)
Volume 07, Nomor 04, Desember 2022 : 1281 - 1293

Pada tabel IV, belum diketahui label untuk contoh data uji tersebut, maka labelnya akan dicari menggunakan perhitungan *euclidean distance* dengan rumus (3). Langkah-langkah algoritma ML-KNN adalah sebagai berikut.

1. Hitung *prior* probabilitasnya menggunakan rumus (1) dan (2). Perhitungan dari rumus-rumus tersebut akan diterapkan pada kolom *Hinselmann* seperti berikut ini.

$$P(H_1^l) = \frac{s + \sum_{j=1}^{n} \vec{y}_{x_i}}{s \times 2 + n}$$
$$P(H_1^l) = \frac{1 + 8}{1 \times 2 + 10}$$
$$P(H_1^l) = \frac{9}{12}$$
$$P(H_1^l) = 0{,}75$$

$$P(H_0^l) = 1 - P(H_1^l)$$
$$P(H_0^l) = 1 - 0{,}75$$
$$P(H_0^l) = 0{,}25$$

Begitu seterusnya sampai ke-4 label sudah dihitung. Hasil perhitungan dapat dilihat pada tabel V di bawah ini.

TABEL V
HASIL PERHITUNGAN *PRIOR PROBABILITY*

Keterangan	*Hinselmann*	*Schiller*	*Citology*	*Biopsy*
Prior Probabilitas	$P(H_1^l) = 0{,}75$ $P(H_0^l) = 0{,}25$	$P(H_1^l) = 0{,}66$ $P(H_0^l) = 0{,}33$	$P(H_1^l) = 0{,}42$ $P(H_0^l) = 0{,}58$	$P(H_1^l) = 0{,}42$ $P(H_0^l) = 0{,}58$

2. Tentukan nilai K sebagai banyaknya jumlah tetangga terdekat dari data uji terhadap data latih. Pada contoh ini akan menggunakan nilai K=3. Hitung jaraknya menggunakan rumus (3).
3. Selanjutnya hitung vektor keanggotaan untuk mengetahui banyaknya data tetangga dari data uji yang berlabel l dengan persamaan (4).
4. Selanjutnya hitung *posterior probability* dari tiap label pada data tetangga yang bernilai 1 menggunakan rumus (5) dan 0 menggunakan rumus (6) seperti berikut.

$$P(E_j^l | H_1^l) = \frac{s + c[j]}{s \times (k+1) + \sum_{j=1}^{k} c[j]}$$
$$P(E_j^l | H_1^l) = \frac{1 + 3}{1 \times (3+1) + 3}$$
$$P(E_j^l | H_1^l) = \frac{4}{7}$$
$$P(E_j^l | H_1^l) = 0{,}57$$

$$P(E_j^l | H_0^l) = \frac{s + c'[j]}{s \times (k+1) + \sum_{j=1}^{k} c'[j]}$$
$$P(E_j^l | H_0^l) = \frac{1 + 0}{1 \times (3+1) + 0}$$
$$P(E_j^l | H_0^l) = \frac{1}{4}$$
$$P(E_j^l | H_0^l) = 0{,}25$$

Begitu seterusnya sampai ke-4 label sudah dihitung. Hasil perhitungan dapat dilihat pada tabel VI di bawah ini.

E-ISSN : 2540 - 8984

JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)
Volume 07, Nomor 04, Desember 2022 : 1281 - 1293

TABEL VI
HASIL PERHITUNGAN *POSTERIOR PROBABILITY*

Keterangan	*Hinselmann*	*Schiller*	*Citology*	*Biopsy*
Prior Probabilitas	$P(H_1^l)$ = 0,75 $P(H_0^l)$ = 0,25	$P(H_1^l)$ = 0,66 $P(H_0^l)$ = 0,33	$P(H_1^l)$ = 0,42 $P(H_0^l)$ = 0,58	$P(H_1^l)$ = 0,42 $P(H_0^l)$ = 0,58
Tetangga pertama	1	1	1	1
Tetangga kedua	1	1	0	1
Tetangga ketiga	1	1	0	1
Vektor keanggotaan	$\vec{C}_x(l)$ = 3	$\vec{C}_x(l)$ = 3	$\vec{C}_x(l)$ = 1	$\vec{C}_x(l)$ = 3
Posterior Probabilitas	$P(E_j^l \mid H_1^l)$ = 0,57 $P(E_j^l \mid H_0^l)$ = 0,25	$P(E_j^l \mid H_1^l)$ = 0,57 $P(E_j^l \mid H_0^l)$ = 0,25	$P(E_j^l \mid H_1^l)$= 0,4 $P(E_j^l \mid H_0^l)$ = 0,5	$P(E_j^l \mid H_1^l)$ = 0,57 $P(E_j^l \mid H_0^l)$ = 0,25

5. Selanjutnya untuk menentukan label dari data uji dengan mencari *Maximum a posteriori (MAP)* digunakan rumus (8) seperti berikut.

$$\vec{y}_t(Hinselmann) = \arg max_{b\in\{1,0\}} P(H_b^l).P(E^l_{\vec{C}_x(l)}|H_b^l)$$
$$\vec{y}_t(Hinselmann) = \arg \max P(H_1^l).P(E^l_{\vec{C}_x(l)}|H_1^l)$$
$$\vec{y}_t(Hinselmann) = 0{,}75 \times 0{,}57$$
$$\vec{y}_t(Hinselmann) = 0{,}42$$

$$\vec{y}_t(Hinselmann) = \arg max_{b\in\{1,0\}} P(H_b^l).P(E^l_{\vec{C}_x(l)}|H_b^l)$$
$$\vec{y}_t(Hinselmann) = \arg max\, P(H_0^l).P(E^l_{\vec{C}_x(l)}|H_0^l)$$
$$\vec{y}_t(Hinselmann) = 0{,}25 \times 0{,}25$$
$$\vec{y}_t(Hinselmann) = 0{,}06$$

Dari perhitungan di atas diperoleh hasil *maximum a posteriori (MAP)* dari data yang bernilai 0 lebih besar dari hasil *maximum a posteriori (MAP)* dari data yang bernilai 1 dan untuk penentuan label dari data uji menggunakan nilai MAP yang paling besar, maka data uji kolom *Hinselmann* bernilai 0. Seperti itu seterusnya sampai keseluruhan label dari data uji selesai ditentukan. Hasilnya seperti yang terlihat pada tabel VII berikut ini.

TABEL VII
HASIL PERHITUNGAN *POSTERIOR PROBABILITY*

Keterangan	*Hinselmann*	*Schiller*	*Citology*	*Biopsy*
Prior Probabilitas	$P(H_1^l)$ = 0,75 $P(H_0^l)$ = 0,25	$P(H_1^l)$ = 0,66 $P(H_0^l)$ = 0,33	$P(H_1^l)$ = 0,42 $P(H_0^l)$ = 0,58	$P(H_1^l)$ = 0,42 $P(H_0^l)$ = 0,58
Tetangga pertama	1	1	1	1
Tetangga kedua	1	1	0	1
Tetangga ketiga	1	1	0	1
Vektor keanggotaan	$\vec{C}_x(l)$ = 3	$\vec{C}_x(l)$ = 3	$\vec{C}_x(l)$ = 1	$\vec{C}_x(l)$ = 3
Posterior Probabilitas	$P(E_j^l \mid H_1^l)$ = 0,57 $P(E_j^l \mid H_0^l)$ = 0,25	$P(E_j^l \mid H_1^l)$ = 0,57 $P(E_j^l \mid H_0^l)$ = 0,25	$P(E_j^l \mid H_1^l)$= 0,4 $P(E_j^l \mid H_0^l)$ = 0,5	$P(E_j^l \mid H_1^l)$ = 0,57 $P(E_j^l \mid H_0^l)$ = 0,25
Maximum A Posteriori (MAP) Probabilitas	$\vec{y}_t(l)1$ = 0,42 $\vec{y}_t(l)0$ = 0,06	$\vec{y}_t(l)1$ = 0,38 $\vec{y}_t(l)0$ = 0,08	$\vec{y}_t(l)1$ = 0,16 $\vec{y}_t(l)0$ = 0,29	$\vec{y}_t(l)1$ = 0,23 $\vec{y}_t(l)0$ = 0,14
Label untuk data uji	1	1	0	1

Berdasarkan tabel VII, diperoleh label untuk data uji dengan data latih sebanyak 10 baris yaitu *Hinselmann* adalah 1, *Schiller* adalah 1, *Citology* adalah 0, dan *Biopsy* adalah 1. Karena waktu perhitungan yang lama, maka klasifikasi Multi-Label K-Nearest Neighbor diterapkan menggunakan *Python* untuk mempersingkat waktu komputasi dengan menggunakan nilai K=1, K=3, K=5, K=7, dan K=9.

E-ISSN : 2540 - 8984

JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)
Volume 07, Nomor 04, Desember 2022 : 1281 - 1293

C. Evaluasi

Evaluasi dilakukan dengan melihat nilai *hamming loss* dan akurasi. Selain itu juga digunakan *confusion matrix* untuk dilihat nilai *precision*, *recall*, dan *f-measure/f1-score*. Karena *dataset* yang digunakan tidak seimbang untuk kelas 0 dan kelas 1-nya pada kolom *Hinselmann*, *Schiller*, *Citology*, dan *Biopsy* maka digunakan rata-rata *weighted* pada *precision*, *recall*, dan *f1-score* untuk evaluasinya. Evaluasi dilakukan dengan melihat perbandingan nilai K untuk mengetahui perbedaan performa dari setiap nilai K. Perbandingan nilai K bertujuan untuk mendapatkan nilai K yang optimal terhadap dataset.

1. Nilai K=1

   Berikut ini tabel yang berisikan hasil *confusion matrix* klasifikasi menggunakan ML-KNN saat nilai K=1 untuk masing-masing label.

TABEL VIII
CONFUSION MATRIX KLASIFIKASI ML-KNN SAAT NILAI K=1

	Hinselmann	*Schiller*	*Citology*	*Biopsy*
True Positive	53	101	36	117
True Negative	191	137	202	124
False Positive	3	8	6	9
False Negative	3	4	6	0

Tabel VIII di atas merupakan *Confusion Matrix* dari klasifikasi ML-KNN saat nilai K=1. Dengan menggunakan persamaan 10 sampai persamaan 14, diperoleh nilai *hamming loss*, akurasi, *precision weighted*, *recall weighted*, dan *f1-score weighted* seperti tabel IX berikut ini.

TABEL IX
HASIL NILAI *HAMMING LOSS*, AKURASI, *PRECISION WEIGHTED*, *RECALL WEIGHTED*, DAN *F1-SCORE WEIGHTED* SAAT NILAI K=1

Evaluasi	Hasil Nilai
Hamming Loss	3,9%
Akurasi	92%
Precision Weighted	92%
Recall Weighted	96%
F1-Score Weighted	94%

Tabel IX di atas merupakan hasil nilai *hamming loss*, akurasi, *precision weighted*, *recall weighted*, dan *f1-score weighted* saat nilai K=1 dengan nilai *hamming loss* sebesar 3,9%, akurasi sebesar 92%, *precision weighted* sebesar 92%, *recall weighted* sebesar 96%, dan *f1-score weighted* sebesar 94%.

2. Nilai K=3

   Berikut ini tabel yang berisikan hasil *confusion matrix* klasifikasi menggunakan ML-KNN saat nilai K=3 untuk masing-masing label.

TABEL X
CONFUSION MATRIX KLASIFIKASI ML-KNN SAAT NILAI K=3

	Hinselmann	*Schiller*	*Citology*	*Biopsy*
True Positive	50	98	36	114
True Negative	193	139	203	125
False Positive	1	6	5	8
False Negative	6	7	6	3

Tabel X di atas merupakan *Confusion Matrix* dari klasifikasi ML-KNN saat nilai K=3. Dengan menggunakan persamaan 10 sampai persamaan 14, diperoleh nilai *hamming loss*, akurasi, *precision weighted*, *recall weighted*, dan *f1-score weighted* seperti tabel XI berikut ini.

E-ISSN : 2540 - 8984

JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)
Volume 07, Nomor 04, Desember 2022 : 1281 - 1293

TABEL XI
HASIL NILAI *HAMMING LOSS*, AKURASI, *PRECISION WEIGHTED*, *RECALL WEIGHTED*, DAN *F1-SCORE WEIGHTED* SAAT NILAI K=3

Evaluasi	Hasil Nilai
Hamming Loss	4,2%
Akurasi	92%
Precision Weighted	94%
Recall Weighted	93%
F1-Score Weighted	93%

Tabel XI di atas merupakan hasil nilai *hamming loss*, akurasi, *precision weighted*, *recall weighted*, dan *f1-score weighted* saat nilai K=3 dengan nilai *hamming loss* sebesar 4,2%, akurasi sebesar 92%, *precision weighted* sebesar 94%, *recall weighted* sebesar 93%, dan *f1-score weighted* sebesar 93%.

3. Nilai K=5
Berikut ini tabel yang berisikan hasil *confusion matrix* klasifikasi menggunakan ML-KNN saat nilai K=5 untuk masing-masing label.

TABEL XII
CONFUSION MATRIX KLASIFIKASI ML-KNN SAAT NILAI K=5

	Hinselmann	*Schiller*	*Citology*	*Biopsy*
True Positive	53	101	36	117
True Negative	192	138	203	124
False Positive	2	7	5	9
False Negative	3	4	6	0

Tabel XII di atas merupakan *Confusion Matrix* dari klasifikasi ML-KNN saat nilai K=5. Dengan menggunakan persamaan 10 sampai persamaan 14, diperoleh nilai *hamming loss*, akurasi, *precision weighted*, *recall weighted*, dan *f1-score weighted* seperti tabel XIII berikut ini.

TABEL XIII
HASIL NILAI *HAMMING LOSS*, AKURASI, *PRECISION WEIGHTED*, *RECALL WEIGHTED*, DAN *F1-SCORE WEIGHTED* SAAT NILAI K=5

Evaluasi	Hasil Nilai
Hamming Loss	3,59%
Akurasi	93%
Precision Weighted	93%
Recall Weighted	96%
F1-Score Weighted	94%

Tabel XIII di atas merupakan hasil nilai *hamming loss*, akurasi, *precision weighted*, *recall weighted*, dan *f1-score weighted* saat nilai K=5 dengan nilai *hamming loss* sebesar 3,59%, akurasi sebesar 93%, *precision weighted* sebesar 93%, *recall weighted* sebesar 96%, dan *f1-score weighted* sebesar 94%.

4. Nilai K=7
Berikut ini tabel yang berisikan hasil *confusion matrix* klasifikasi menggunakan ML-KNN saat nilai K=7 untuk masing-masing label.

TABELX IV
CONFUSION MATRIX KLASIFIKASI ML-KNN SAAT NILAI K=7

	Hinselmann	*Schiller*	*Citology*	*Biopsy*
True Positive	43	88	34	104
True Negative	192	138	203	124
False Positive	2	7	5	9
False Negative	13	17	8	13

E-ISSN : 2540 - 8984

JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)
Volume 07, Nomor 04, Desember 2022 : 1281 - 1293

Tabel XIV di atas merupakan *Confusion Matrix* dari klasifikasi ML-KNN saat nilai K=7. Dengan menggunakan persamaan 10 sampai persamaan 14, diperoleh nilai *hamming loss*, akurasi, *precision weighted*, *recall weighted*, dan *f1-score weighted* seperti tabel XV berikut ini.

TABEL XV
HASIL NILAI *HAMMING LOSS*, AKURASI, *PRECISION WEIGHTED*, *RECALL WEIGHTED*, DAN *F1-SCORE WEIGHTED* SAAT NILAI K=7

Evaluasi	Hasil Nilai
Hamming Loss	7,39%
Akurasi	88%
Precision Weighted	92%
Recall Weighted	84%
F1-Score Weighted	88%

Tabel XV di atas merupakan hasil nilai *hamming loss*, akurasi, *precision weighted*, *recall weighted*, dan *f1-score weighted* saat nilai K=7 dengan nilai *hamming loss* sebesar 7,39%, akurasi sebesar 88%, *precision weighted* sebesar 92%, *recall weighted* sebesar 84%, dan *f1-score weighted* sebesar 88%.

5. Nilai K=9
Berikut ini tabel yang berisikan hasil *confusion matrix* klasifikasi menggunakan ML-KNN saat nilai K=9 untuk masing-masing label.

TABEL XVI
CONFUSION MATRIX KLASIFIKASI ML-KNN SAAT NILAI K=9

	Hinselmann	*Schiller*	*Citology*	*Biopsy*
True Positive	41	86	32	100
True Negative	190	134	203	121
False Positive	4	4	5	12
False Negative	15	19	10	17

Tabel XVI di atas merupakan *Confusion Matrix* dari klasifikasi ML-KNN saat nilai K=9. Dengan menggunakan persamaan 10 sampai persamaan 14, diperoleh nilai *hamming loss*, akurasi, *precision weighted*, *recall weighted*, dan *f1-score weighted* seperti tabel XVII berikut ini.

TABEL XVII
HASIL NILAI *HAMMING LOSS*, AKURASI, *PRECISION WEIGHTED*, *RECALL WEIGHTED*, DAN *F1-SCORE WEIGHTED* SAAT NILAI K=9

Evaluasi	Hasil Nilai
Hamming Loss	9,3%
Akurasi	84%
Precision Weighted	89%
Recall Weighted	81%
F1-Score Weighted	85%

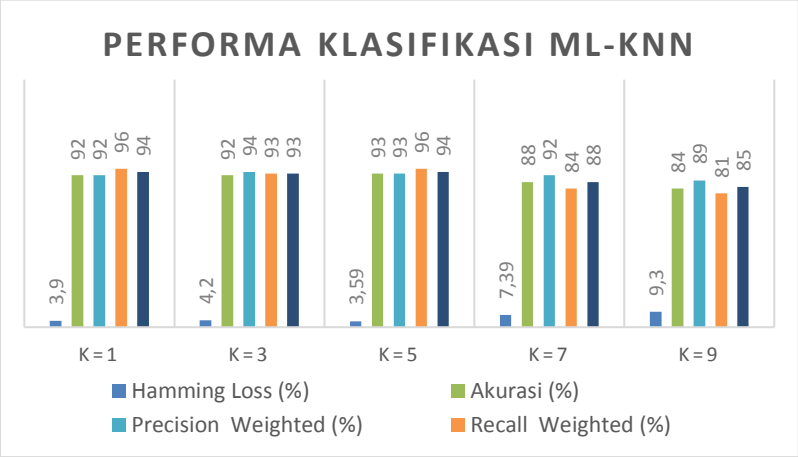
Tabel XVII di atas merupakan hasil nilai *hamming loss*, akurasi, *precision weighted*, *recall weighted*, dan *f1-score weighted* saat nilai K=9 dengan nilai *hamming loss* sebesar 9,3%, akurasi sebesar 84%, *precision weighted* sebesar 89%, *recall weighted* sebesar 81%, dan *f1-score weighted* sebesar 85%.

D. Hasil Evaluasi Klasifikasi Menggunakan ML-KNN

Berikut ini gambar visualisasi berisikan hasil evaluasi klasifikasi menggunakan metode ML-KNN.

E-ISSN : 2540 - 8984

JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)
Volume 07, Nomor 04, Desember 2022 : 1281 - 1293



Gambar. 2. Visualisasi Evaluasi Klasifikasi ML-KNN saat nilai K=1, K=3, K=5, K=7, dan K=9.

Dari gambar 2 diketahui bahwa nilai K=5 memperoleh nilai *hamming loss*, akurasi, *precision weighted*, *recall weighted*, dan *f1-score weighted* yang lebih baik daripada nilai k lainnya yaitu dengan nilai *hamming loss* sebesar 3,59%, akurasi sebesar 93%, *precision weighted* sebesar 93%, *recall weighted* sebesar 96%, dan *f1-score weighted* sebesar 94%. .

Dibandingkan dengan penelitian [4] yang juga menggunakan metode ML-KNN, nilai *hamming loss* yang peroleh tidak lebih baik daripada yang diperoleh pada penelitian tersebut. Hal ini dapat disebabkan oleh penggunakan dataset yang berbeda. Namun, apabila dibandingkan dengan penelitian [5] yang juga menggunakan *dataset* yang sama dengan penelitian ini dan menggunakan metode yang berbeda yaitu metode K-NN, diketahui bahwa pada penelitian ini memperoleh nilai akurasi yang lebih baik dibandingkan dengan penelitian tersebut, sehingga untuk *dataset* kanker serviks ini diperoleh hasil evaluasi yang lebih baik dengan klasifikasi menggunakan metode ML-KNN dibandingkan dengan klasifikasi menggunakan metode K-NN.

IV. KESIMPULAN

Berdasarkan penelitian yang dilakukan, diperoleh kesimpulan sebagai berikut.

1. Klasifikasi penyakit kanker serviks menggunakan metode ML-KNN dilakukan dengan melakukan perhitungan *prior probability* dari tiap label pada data latih yang nilainya 1 dan 0. Lalu menentukan nilai K sebagai banyaknya tetangga terdekat, yaitu K=1, K=3, K=5, K=7, dan K=9. Setelah itu melakukan perhitungan jarak antara data uji dengan data latih menggunakan rumus *euclidean distance*. Berikutnya menghitung vektor keanggotaan untuk mengetahui banyaknya data tetangga dari data uji yang berlabel l. Selanjutnya melakukan perhitungan *posterior probability* dari tiap label pada data tetangga yang bernilai 1 dan 0. Setelah itu untuk menentukan label dari data uji, dicari menggunakan *Maximum a posteriori* (MAP). Label pada data uji diputuskan berdasarkan nilai MAP yang paling besar.
2. Klasifikasi penyakit kanker serviks menggunakan ML-KNN dengan melakukan *oversampling* pada kolom *Biopsy* diperoleh performa terbaiknya yaitu saat nilai K=5 dengan memperoleh nilai *hamming loss* sebesar 3,59%, akurasi sebesar 93%, *precision weighted* sebesar 93%, *recall weighted* sebesar 96%, dan *f1-score weighted* sebesar 94%.

DAFTAR PUSTAKA

[1] Praningki, T., & Budi, I. (2018). Sistem Prediksi Penyakit Kanker Serviks Menggunakan CART, Naive Bayes, dan k-NN. Creative Information Technology Journal, 4(2), 83. https://doi.org/10.24076/citec.2017v4i2.100.

[2] Global Cancer Observatory. (2020). Global Cancer Observatory Indonesia. Diakses tanggal 4 Oktober 2021. Site: https://gco.iarc.fr/today/data/factsheets/populations/360-indonesia-fact-sheets.pdf.

[3] Purwoastuti&Walyani. (2015). Ilmu Obstetri & Ginekologi Sosial untuk Kebidanan. Yogyakarta:Pustaka Baru Press.

[4] Dharma, A., Manalu, P., Sinaga, G. S., Siringoringo, R., Palangai, I. S., & Setiawan, K. (2020). Deteksi Pola Pasien Kanker Serviks dengan Algoritma Extra Trees dan K-Nearest Neighbor. Jurnal Ilmu Komputer dan Sistem Informasi (JIKOMSI), 3(1), 32-36.

[5] Dan Zhu, Hui Zhu, Ximeng Liu, Hui Li, Fengwei Wang, Hao Li, Dengguo Feng, CREDO: Efficient and privacy-preserving multi-level medical prediagnosis based on ML-kNN, Information Sciences, Volume 514, 2020, Pages 244-262, ISSN 0020-0255, https://doi.org/10.1016/j.ins.2019.11.041.

[6] Li, S., & Ou, J. (2021). Multi-Label Classification of Research Papers Using Multi-Label K-Nearest Neighbour Algorithm. Journal of Physics: Conference Series, 1994(1). https://doi.org/10.1088/1742-6596/1994/1/012031.

[7] UCI Machine Learning. (2017). Site: https://archive.ics.uci.edu/ml/datasets/Cervical+cancer+%28Risk+Factors%29.

[8] Aminah, Siti Hawa. (2018). Prediksi Diagnosa Kanker Serviks Berdasarkan Informasi Demografi, Kebiasaan, dan Rekam Medis Menggunakan Algoritma Support Vector Machine. Institut Teknologi Sepuluh Nopember, Departemen Sistem Informasi, Fakultas Teknologi Informasi dan Komunikasi. Surabaya: Departemen Sistem Informasi.

[9] Yu, C., Wu, H., Liu, H. (2020). Smart Device Recognition: Ubiquitous Electric Internet of Things. Germany: Springer Singapore.

[10] Ceylan, Z., & Pekel, E. (2017). BAT algorithm for Cryptanalysis of Feistel cryptosystems. International Journal of Intelligent Systems and Applications in Engineering, 3(2), 82. https://doi.org/10.18201/ijisae.82426.

[11] Kumar, Ajitesh. (2020, September 4). Vitalflux. Diakses pada 2 Juni 2022. Site: https://vitalflux.com/micro-average-macro-average-scoring-metrics-multi-class-classification-python/.

[12] Tarekegn, A. N., Giacobini, M., & Michalak, K. (2021). A review of methods for imbalanced multi-label classification. Pattern Recognition, 118, 107965. https://doi.org/10.1016/j.patcog.2021.107965.).

[13] Brownlee, Jason. Master Machine Learning Algorithms: Discover How They Work and Implement Them From Scratch. (2016). United States: Machine Learning Mastery.