Sistem Temu Kembali Informasi Diagnosis Primordial Penyakit

Aszani, Hayyu Ilham Wicaksono, Uffi Nadzima
Magister Ilmu Komputer
Universitas Gadjah Mada

***Abstrak*—Pada penelitian ini, peneliti membahas masalah perkembangan data rekam medis yang terus meningkat. Pertumbuhan rekam medis perlu dimanfaatkan dengan baik untuk meningkatkan kinerja dokter dalam mendiagnosis suatu penyakit. peneliti mengeksplorasi teknologi temu kembali informasi dengan melakukan perluasan dataset gejala rekam medis dengan dataset yang lebih baik untuk memprediksi diagnosis. Peneliti kemudian memanfaatkan augmentasi dataset untuk meningkatkan akurasi sistem. Tantangan besar dari penelitian ini adalah dataset rekam medis yang tidak terstruktur, sehingga menyulitkan dalam menyesuaikan gejala dengan diagnosis tertentu. Maka dari itu, peneliti mengusulkan sebuah metode temu kembali informasi untuk memberikan rekomendasi diagnosis berdasarkan gejala dari dataset rekam medis, dengan melakukan augmentasi dataset. Hasil percobaan menunjukan bahwa metode yang kami usulkan mampu meningkatkan akurasi dalam memprediksi diagnosis penyakit.**

***Index Terms*—Augmentasi Data, *Cosine Similarity*, Diagnosis, Prediksi, TF-IDF**

I. Pendahuluan

Teknologi pencarian diagnosis penyakit digunakan untuk menemukan informasi penyakit yang sesuai dan relevan secara cepat. Pencarian tersebut diperoleh dari sekumpulan data rekam medis dan klinis yang sangat besar [1]. Rekam medis merupakan berkas berisi catatan dan dokumen tentang identitas pasien, pemeriksaan, pengobatan, tindakan dan pelayanan lain kepada pasien pada fasilitas kesehatan yang dilakukan secara manual maupun elektronik [2]. Pertumbuhan jumlah data rekam medis yang terus bertambah dapat dijadikan sebagai bahan informasi dan alat pendukung keputusan dalam melakukan diagnosis.

Diagnosis tahap awal yang tepat pada suatu penyakit akan memberikan pengaruh baik untuk pencegahan pada resiko yang lebih berbahaya. Klasifikasi suatu gejala penyakit ke dalam kode ICD, untuk membantu memberikan rekomendasi diagnosis penyakit sangat melelahkan dan memakan waktu. Seorang profesional yang telah memiliki banyak pengalaman membutuhkan waktu sekitar 20 menit per kasus [3]. Oleh karena itu perlu adanya suatu sistem yang dapat membantu untuk meningkatkan akurasi diagnosis dan mengurangi waktu yang diperlukan dalam menentukan diagnosis pasien [4]. Tentunya dengan eksistensi sistem tersebut diharapkan dapat memudahkan dan mempercepat proses pelayanan dan penanganan kesehatan kepada pasien.

Pada penelitian ini dengan memanfaatkan teknologi temu kembali informasi, diusulkan suatu sistem untuk mendiagnosis primordial atau tahap awal suatu penyakit yang tertera pada ICD 10 berdasarkan gejala yang dialami. meningkatkan akurasi prediksi suatu diagnosis peyakit, peneliti melakukan augmentasi dataset agar kemungkinan variasi gejala yang mirip dapat dikenali lebih optimal. Peneliti memanfaatkan metode *Top k-accurancy* untuk melakukan evaluasi terhadap sistem prediksi ini, dengan masukkan seluruh gejala penyakit yang tersedia di dataset.

II. Data dan Metode

A. Dataset

Dataset yang digunakan pada penelitian ini diambil dari beberapa sumber berbeda:

1) Dataset yang diperoleh dari repositori GitHub *Disease Detection based on Symptoms with treatment recommendation*. Dataset terdiri dari 261 penyakit dengan 489 fitur gejala dalam Bahasa Inggris, yang diberi nilai 0 dan 1. Peneliti melakukan proses *scrapping* melalui situs web National Health Portal of India. Dataset diterjemahkan ke dalam bahasa Indonesia untuk diimplementasikan dalam rancangan sistem pada penelitian ini.
2) Dataset ICD 10 dikombinasikan dengan data pada 1 sebagai detail kode penyakit sesuai standar internasional.
3) Dataset dari situs web Rekmed berupa rekam medis pasien yang berisi gejala yang dialami pasien dan diagnosis yang diberikan. Data tersebut digunakan sebagai data tes untuk evaluasi sistem yaitu dengan gejala yang dialami pasien sebagai masukkan sistem dan diagnosis penyakit yang diberikan sebagai keluaran.
4) Informasi tambahan terkait keluaran dari detail diagnosis penyakit yang diprediksi diambil dan diterjemahkan dari situs web Infobox Wikipedia.

B. Landasan Teori

Sistem temu kembali informasi adalah pencarian materi (umumnya dokumen) dari sesuatu yang tidak terstruktur (umumnya teks) dan memenuhi informasi dari dalam koleksi besar yang tersimpan di komputer [5].

Sistem temu kembali informasi memiliki 2 tahapan utama yaitu konstruksi indeks(*index construction/indexing*) dan pengolahan *query* untuk me-*retrieve* dokumen/informasi yang sesuai.

1) **TF-IDF**
   Teknik untuk mengukur sebuah kata dalam dokumen. Pada tahap ini, melakukan perhitungan bobot untuk setiap kata untuk menandakan tingkat kepentingan kata

dalam dokumen dan *corpus*. Dalam penelitian ini, gejala merupakan *term* dan penyakit sebagai dokumen.

Metode *TF-IDF* (*Term Frequency-Inverse Document Frequency*) digunakan untuk pembobotan term pada suatu dokumen. Metode ini digunakan untuk pengambilan informasi (*information retrieval*) yang merupakan teknik pembobotan statistik pada teks.

Bobot pada *TF-IDF* merupakan frekuensi kemunculaan sebuah kata/term i di dalam setiap dokumen j yang diindikasi sebagai *Term Frequency* yang dinotasikan dengan $tf_{i,j}$ dan total kemunculan suatu kata/term pada semua dokumen dinotasikan dengan df_i. Diperoleh persamaan *Inverse Document Frequency* (*IDF*) pada Persamaan (1) [6].

$$idf_i = \log\left(\frac{N}{df_i}\right) \tag{1}$$

Notasi N merupakan jumlah seluruh dokumen yang dilakukan pembobotan. Selanjutnya, nilai $tf_{i,j}$ dan idf_i yang diperoleh digunakan untuk menghitung bobot kata atau *term* pada masing-masing dokumen seperti yang ditunjukkan pada Persamaan (2).

$$w_{i,j} = tf_{i,j} \times idf_i \tag{2}$$

2) **Cosine Similarity**

Menurut Kocher dan Savoy dalam [7], *cosine similarity* merupakan suatu ukuran dari derajat kemiripan antara dua vektor dan dalam *inner product* pengukuran ini paling populer digunakan. Dalam penggunaannya pada penelitian ini, pengukuran *cosine similarity* digunakan untuk mengindikasi derajat kemiripan antara dua kalimat. Hasil pengukuran menghasilkan nilai antara 0 dan 1 dimana jika nilainya 0 maka tidak ada kemiripan antara dua kalimat tersebut dan sebaliknya jika nilai yang dihasilkan 1 maka kalimat tersebut diidentifikasi sebagai kalimat yang identik. Pengukuran *cosine similarity* ditunjukkan pada Persamaan (3) untuk membandingkan dua kalimat K_1 dan K_2.

$$\begin{aligned} sim(K_1, K_2) &= \frac{K_1 \cdot K_2}{\|K_1\| \, \|K_2\|} \\ &= \frac{\Sigma_{i=1}^{n} K_{1i} K_{2i}}{\sqrt{\Sigma_{i=1}^{n}(K_{1i})^2}\sqrt{\Sigma_{i=1}^{n}(K_{1i})^2}} \end{aligned} \tag{3}$$

Dengan K_{1i} dan K_{2i} merupakan komponen vektor pada kalimat K_1 dan K_2.

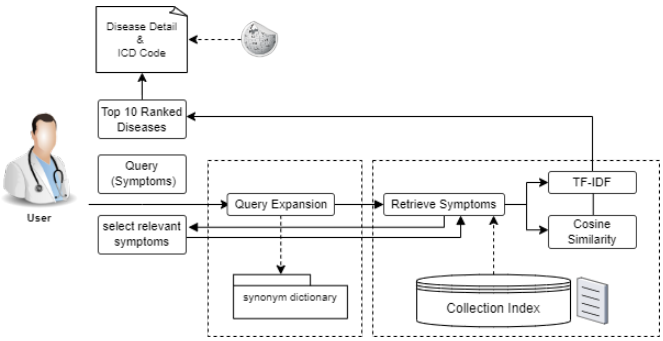
3) **Evaluasi**

Beberapa teknik evaluasi cocok untuk mengukur prediksi skor suatu sistem rekomendasi tanpa proses training data. Penggunaan *Top k-accuracy-score* merupakan salah satu model evaluasi untuk menggeneralisasi suatu prediksi dari akurasi skor [8]. Prediksi suatu informasi menggunakan *Information Retrieval* umumnya kalkulasi dari banyak item yang relevan dan tidak relevan, yang akan mengarah ke rekomendasi informasi tertentu. Evaluasi dengan *Top k-accuracy-score* dapat digunakan dalam kasus klasfikasi biner dan multiclass. Untuk mendapatkan *Top k-accuracy-score* dapat didefinsikan seperti ditunjukkan pada Persamaan (4). Jika fij adalah kelas prediksi untuk i sampel hingga j prediksi skor tertinggi, dengan y merupakan nilai yang benar.

$$\text{top-k accuracy}(y, \hat{f}) = \frac{\Sigma_{i=0}^{n_{\text{samples}}-1} \cdot \Sigma_{j=1}^{k}(\hat{f}_{i,j} = y_i)}{n_{\text{samples}}} \tag{4}$$

C. Rancangan Sistem

Sistem yang diusulkan pada penelitian ini memiliki desain yang ditunjukkan pada Gambar 1. Berdasarkan Gambar 1



Gambar. 1. *Overview* arsitektur sistem

implementasi sistem diawali dengan pengguna memasukkan gejala. Sistem akan melakukan tahap *preprocessing* terhadap gejala tersebut, meliputi *case folding*, tokenisasi, dan *stemming*. Sistem melakukan perluasan kueri dengan sinonim dari masukkan gejala. Sinonim gejala tersebut akan disesuaikan dengan gejala-gejala yang terdapat pada basis data. Sistem memberikan gejala yang terkait kepada dokter sehingga dokter dapat memilih gejala-gejala yang relevan dengan keluhan penderita menghasilkan daftar gejala akhir.

Gejala yang telah relevan tersebut diubah ke dalam bentuk vektor. Vektor gejala digunakan untuk pengukuran tingkat kemiripan dengan data menggunakan dua model, yaitu pembobotan TF-IDF dan *cosine similarity*.

Hasil dari pengukuran akan tertampil daftar prediksi penyakit yang gejalanya memiliki tingkat kemiripan tinggi, terhadap daftar gejala yang ada di basis data. Sistem akan menampilkan sebanyak k prediksi penyakit dengan persentase tingkat kemiripannya. Berdasarkan diagnosis dari sistem, dokter dapat menentukan atau memvalidasi penyakit yang paling sesuai dengan masukkan gejala. Untuk menduk ung pengetahuan dari penyakit yang diprediksi sistem akan menampilkan informasi detail penyakit dan kode ICD-10 dari penyakit yang ingin diketahui detailnya.

D. Implementasi

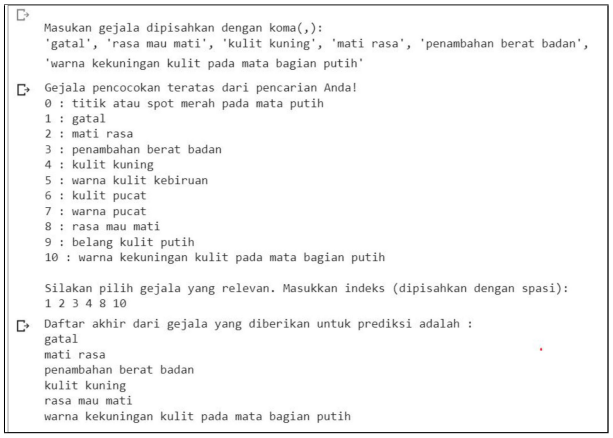
Rancangan pada Gambar 1 diimplementasikan ke dalam bahasa pemrograman Python.

1) **Pemrosesan Kueri**

Kueri dimasukkan oleh pengguna yang berisi gejala-gejala penyakit dalam satu baris kueri dan setiap gejalanya dipisahkan dengan koma(,). Kueri yang masuk

dilakukan *preprocessing* yaitu tokenisasi, *case folding* dan *stemming*.

2) **Perluasan Kueri**
Proses ini menggunakan masukkan dari hasil *preprocessing* kueri. Gejala yang sesuai diperluas dengan kombinasi dari daftar sinonimnya sehingga menghasilkan daftar gejala yang cocok yang telah dibubuhkan indeks sehingga pengguna dapat memilih gejala-gejala yang sesuai dengan indeksnya.

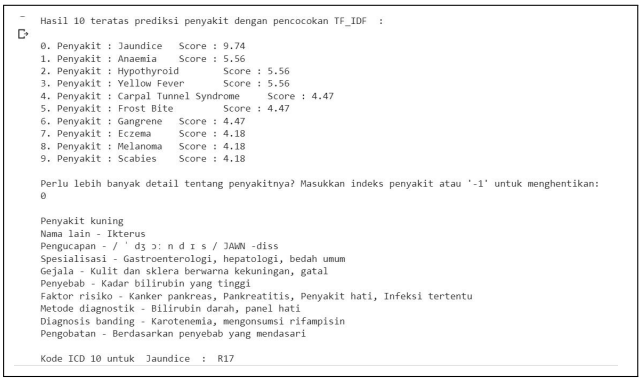


Gambar. 2. Implementasi seleksi pemasukkan gejala

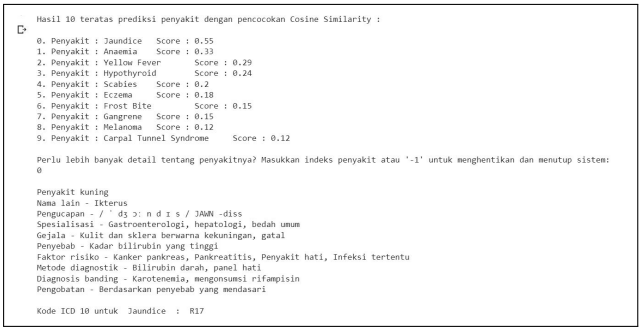
3) **Seleksi Gejala**
Selain perluasan kueri gejala yang cocok, sistem juga akan menampilkan gejala-gejala yang mungkin berkaitan dengan gejala yang dimasukkan pengguna. Dari gejala-gejala yang di-*retrieve* tersebut pengguna dapat menyeleksi gejala yang diberikan, memilih lanjut ke daftar gejala berkaitan lainnya atau berhenti memberikan masukkan gejala. Gambar 2 merupakan contoh implementasi pemasukan gejala, kombinasi perluasan gejala yang telah di proses oleh sistem, dan seleksi gejala berdasarkan kemiripan gejala pada dataset.

4) **Prediksi Penyakit**
   a) **Model Prediksi Dengan TF-IDF dan *Cosine Similarity***
   Diagnosis diperoleh dari prediksi penyakit yang cocok dengan gejala yang diberikan. Model yang digunakan untuk memprediksi yaitu TF-IDF dan *cosine similarity*. Berdasarkan gambar 3 dan 4, dua model yang dikembangkan akan menampilkan 10 prediksi diagnosis teratas yang mirip dengan gejala pada dataset. Model TF-IDF cenderung memiliki skor prediksi yang lebih tinggi daripada model *cosine similarity*.
   b) **Detail Penyakit**
   Pada bagian ini, pengguna dapat memilih salah satu penyakit dari hasil prediksi untuk mengetahui informasi yang lebih detail yang diambil dari Infobox Wikipedia dan kode ICD-10 dari penyakit yang dipilih.

5) **Pengujian**



Gambar. 3. Hasil diagnosis menggunakan model TF-IDF



Gambar. 4. Hasil diagnosis menggunakan model *cosine similarity*

Dalam tahapan ini, sistem akan dijalankan dan diuji cobakan untuk mengetahui apakah sistem berjalan sesuai dengan hasil analisa dan tujuan yang diharapkan. Untuk mengetahui kemampuan model sistem temu balik informasi yang telah dibangun dalam penelitian ini, maka akan dilakukan pengujian dengan mengukur kualitas retrieval

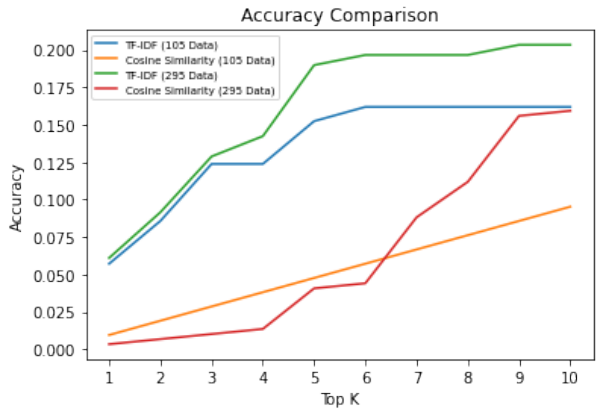
III. HASIL

Performa sistem dievaluasi dengan membandingkan penyakit yang diprediksi dengan diagnosis penyakit pada data rekam medis pasien.

TABEL I
PERBANDINGAN EVALUASI TOP *k* AKURASI
PADA TF-IDF DAN *Cosine Similarity*

k	**TF-IDF (105 Data)**	**Cosine Similarity (105 Data)**	**TF-IDF (295 Data)**	**Cosine Similarity (295 Data)**
1	0.057143	0.009524	0.061017	0.003390
2	0.085714	0.019048	0.091525	0.006780
3	0.123810	0.028571	0.128814	0.010169
4	0.123810	0.038095	0.142373	0.013559
5	0.152381	0.047619	0.189831	0.040678
6	0.161905	0.057143	0.196610	0.044068
7	0.161905	0.066667	0.196610	0.088136
8	0.161905	0.076190	0.196610	0.111864
9	0.161905	0.085714	0.203390	0.155932
10	0.161905	0.095238	0.203390	0.159322

Berdasarkan hasil evaluasi sistem menggunakan *Top k-accuracy-score* diperoleh perbandingan seperti pada Gambar 5. Grafik perbandingan akurasi dari 2 metode *retrieval* yang



Gambar. 5. Grafik evaluasi sistem prediksi penyakit

digunakan dan pengujian pada 2 data uji yaitu yang hanya memiliki 105 daftar gejala dan label penyakitnya dan data yang telah diaugmentasi dengan mengkombinasikan gejala yang dimasukkan sehingga diperoleh 190 data tambahan untuk diuji hasil prediksinya.

Pada TF-IDF hasil akurasi prediksi penyakit dengan k prediksi dengan nilai tertinggi yaitu pada $k = 10$ akurasinya hanya 16% sedangkan setelah dilakukan augmentasi pada data ujinya, akurasi $k = 10$ prediksi meningkat dengan kenaikan pada $k = 1$ sekitar 1% dan $k = 10$ sekitar 4% menjadi 20%. Hasil akurasi pada metode *cosine similarity* secara keseluruhan lebih rendah dari akurasi dengan metode TF-IDF dengan tingkat prediksi tertinggi pada $k = 10$ yaitu $9,5\%$ dan dengan data uji yang diaugmentasi mengalami penurunan akurasi untuk $k = 1$ sampai $k = 6$. Nilai akurasi meningkat pada $k = 7$ hingga $k = 10$ dengan akurasi paling tinggi yaitu mendekati 16%.

Peningkatan hasil akurasi pada data uji yang telah diaugmentasi ini dianalisis bahwa masukkan gejala yang relevan tentunya akan meningkatkan akurasi prediksi penyakitnya. Pada data uji yang kami gunakan pada penelitian ini meskipun data telah diaugmentasi menjadi hampir 3 kali lebih besar dari data uji sebelumnya diperoleh peningkatan hasil akurasi yang tidak terlalu besar. Hal tersebut dianggap bahwa masukkan gejala yang tidak relevan terhadap penyakitnya tidak akan mempengaruhi hasil prediksinya.

IV. Kesimpulan

Dari penelitian ini diperoleh data uji yang telah diaugmentasi secara umum menghasilkan akurasi hasil prediksi yang lebih baik. Tingkat akurasi hasil prediksi menggunakan metode TF-IDF lebih besar dibanding dengan prediksi menggunakan metode *cosine similarity*.

V. Saran

Untuk penelitian selanjutnya dapat dilakukan metode augmentasi yang berbeda atau penambahan data diagnosa penyakit dengan gejala yang lebih banyak dan relevan sehingga dapat memberikan prediksi penyakit yang lebih akurat.

Referensi

[1] M. Mustakim and R. Wardoyo, "Survey model-model pencarian informasi rekam medik elektronik," *JISKA (Jurnal Informatika Sunan Kalijaga)*, vol. 3, no. 3, p. 132–144, Aug. 2019. [Online]. Available: https://ejournal.uin-suka.ac.id/saintek/JISKA/article/view/33-01

[2] R. Silalahi and E. Sinaga, "Perencanaan implementasi rekam medis elektronik dalam pengelolaan unit rekam medis klinik pratama romana," *Jurnal Manajemen Informasi Kesehatan Indonesia*, vol. 7, p. 22, 03 2019.

[3] Commonwealth of Australia, "MBS Telehealth Services from 1 July 2022," 2022. [Online]. Available: http://www.mbsonline.gov.au/internet/mbsonline/publishing.nsf/Content/Factsheet-telehealth-1July22

[4] V. K and S. Jyothi, "Decision support system for congenital heart disease diagnosis based on signs and symptoms using neural networks," *International Journal of Computer Applications*, vol. 19, 04 2011.

[5] C. D. Manning, P. Raghavan, and H. Schütze, *Introduction to Information Retrieval*. USA: Cambridge University Press, 2008.

[6] A. R. Lahitani, A. E. Permanasari, and N. A. Setiawan, "Cosine similarity to determine similarity measure: Study case in online essay assessment," in *2016 4th International Conference on Cyber and IT Service Management*, 2016, pp. 1–6.

[7] K. Park, J. Hong, and W. Kim, "A methodology combining cosine similarity with classifier for text classification," *Applied Artificial Intelligence*, vol. 34, pp. 1–16, 02 2020.

[8] scikit-learn developers, "Metrics and scoring: quantifying the quality of predictions," 2022. [Online]. Available: https://scikit-learn.org/stable/modules/model_evaluation.html