



Sistem Rekomendasi Produk *Fitness* dan Olahraga Berbasis *Content-Based Filtering* Menggunakan Doc2vec dan Cosine Similarity

Airlangga Zabaniyah Putro Irdianto, Ilham Saifudin*, Taufiq Timur W.

Program Studi Teknik Informatika, Fakultas Teknik, Universitas Muhammadiyah Jember
airzaba16@gmail.com, ilham.saifudin@unmuhjember.ac.id*, taufiqtimur@unmuhjember.ac.id

Abstract

Recommendation systems help users identify products that match their needs amid the growing number of choices on digital platforms. The increasing variety of fitness and sports products can cause information overload and make relevant products difficult to find. This study develops and evaluates a content-based filtering recommendation system using Doc2Vec and cosine similarity. The study applies a quantitative experimental method to the Amazon Reviews'23 Dataset. *Product_name*, *category*, and *clean_text* are used as content attributes, while *Product_id* is used as the product identity. The data undergo selection, text cleaning, case folding, tokenization, and stopword removal, followed by an 80% training and 20% testing split. The Doc2Vec model uses the Paragraph Vector Distributed Bag-of-Words architecture with 100-dimensional vectors. Query vectors are produced through *infer_vector* and compared with product vectors using cosine similarity to create Top-N recommendations. Evaluation on four queries shows average Precision of 0.9500, 0.9000, and 0.8750; average Recall of 0.1100, 0.2078, and 0.6082; and average NDCG of 0.9732, 0.9584, and 0.9361 for Top-5, Top-10, and Top-30, respectively. The results show that smaller Top-N values provide higher precision and ranking quality, while larger Top-N values retrieve a broader set of relevant products. The system is implemented as a Flask-based website named Bolang.

Keywords: Recommendation System, Content-Based Filtering, Doc2Vec, Cosine Similarity, Fitness and Sports Products.

1. Pendahuluan

Sistem rekomendasi merupakan salah satu pendekatan untuk mengatasi permasalahan tersebut melalui proses penyaringan dan pemeringkatan informasi berdasarkan karakteristik tertentu (Saifudin & Widiyaningtyas, 2024). Salah satu pendekatan yang banyak digunakan adalah *content-based filtering*, yang menghasilkan rekomendasi berdasarkan tingkat kemiripan karakteristik *item* dengan preferensi pengguna tanpa bergantung pada riwayat interaksi pengguna lain (Khadka & Sthapit, 2025), sehingga cocok diterapkan ketika data interaksi pengguna belum tersedia secara optimal. Pada domain produk *fitness* dan olahraga, pendekatan ini dapat memanfaatkan informasi tekstual produk seperti nama produk, kategori, dan ulasan pengguna sebagai representasi karakteristik produk.

Sebagian penelitian *content-based filtering* masih menggunakan representasi teks berbasis frekuensi kata, seperti *bag-of-words* dan TF-IDF, yang terbatas dalam menangkap hubungan semantik antar kata, serta masih memanfaatkan data interaksi eksplisit seperti *rating* dan riwayat transaksi yang dapat menjadi kendala ketika data interaksi rendah atau tidak tersedia. Pendekatan *ranking* berbasis *rating*, misalnya kombinasi Singular Value Decomposition dengan pemeringkatan Gower pada *collaborative filtering*, juga menunjukkan bahwa strategi pemeringkatan turut menentukan kualitas rekomendasi (Saifudin dkk., 2025), sehingga aspek pemeringkatan perlu diperhatikan meskipun penelitian ini berfokus pada pendekatan berbasis konten. Pendekatan *semantic embedding* seperti Doc2Vec mampu menghasilkan representasi dokumen dalam bentuk vektor dengan mempertimbangkan hubungan semantik teks dan tetap mampu mengenali kemiripan makna meskipun terdapat perbedaan struktur kata (Iskandar & Kurniawati, 2025; Li dkk., 2025). Tingkat kesesuaian antara representasi vektor *query* dan dokumen produk selanjutnya diukur menggunakan *cosine similarity*, yang efektif mengidentifikasi hubungan semantik antar dokumen berbasis *dense embedding* (Karpukhin dkk., 2020).

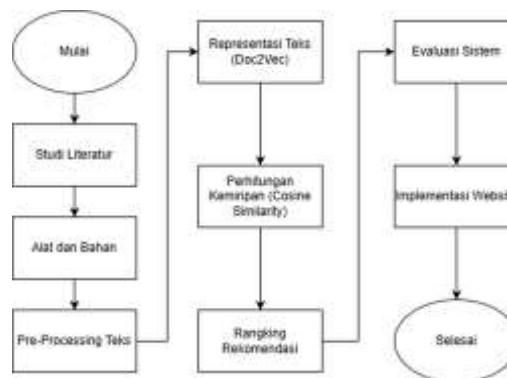
Berdasarkan uraian tersebut, penelitian ini mengembangkan sistem rekomendasi produk *fitness* dan olahraga berbasis *content-based filtering* menggunakan Doc2Vec dan *cosine similarity*. Kontribusi penelitian terletak pada penerapan representasi semantik terhadap nama produk, kategori, dan ulasan pengguna sebagai dasar rekomendasi,

sehingga pencarian kemiripan produk dilakukan berdasarkan konteks informasi dan bukan hanya kesamaan kata secara langsung. Kinerja sistem yang dibangun dievaluasi menggunakan metrik *Precision*, *Recall*, dan *Normalized Discounted Cumulative Gain* (NDCG) pada skenario Top-5, Top-10, dan Top-30 untuk mengukur ketepatan dan kualitas pemeringkatan rekomendasi yang dihasilkan.

Berbeda dengan kajian sistem rekomendasi berbasis *rating* atau interaksi eksplisit, misalnya melalui strategi pemeringkatan pada *collaborative filtering* (Saifudin dkk., 2025), penelitian ini secara khusus menasar kondisi ketika data interaksi pengguna terbatas, sehingga kemiripan konten tekstual menjadi sumber informasi utama dalam menyusun rekomendasi. Survei literatur sistem rekomendasi menunjukkan bahwa pemilihan metode representasi teks dan metrik evaluasi yang tepat sangat menentukan kualitas rekomendasi yang dihasilkan pada berbagai domain, termasuk domain produk konsumen (Saifudin & Widiyaningtyas, 2024). Oleh karena itu, penelitian ini juga memberikan gambaran empiris mengenai *trade-off* antara ketepatan dan cakupan rekomendasi ketika jumlah produk yang ditampilkan (Top-N) diperbesar, sebuah aspek yang masih jarang dibahas secara eksplisit pada penelitian sistem rekomendasi produk *fitness* dan olahraga berbahasa Indonesia.

2. Metode Penelitian

Penelitian menggunakan metode eksperimen kuantitatif dengan pendekatan *query-based recommendation*. Pengguna memasukkan kata kunci atau deskripsi kebutuhan produk melalui *website*. *Query* melalui praproses yang sama dengan dokumen produk, kemudian direpresentasikan menggunakan model *Doc2Vec* yang telah dilatih. *Cosine similarity* menghitung kemiripan antara vektor *query* dan seluruh vektor produk. Sistem mengurutkan nilai kemiripan secara menurun dan menampilkan produk dengan skor tertinggi sebagai rekomendasi *Top-N*. Alur penelitian disusun sesuai tahapan pada laporan tugas akhir dan ditunjukkan pada Gambar 1.



Gambar 1. Alur Penelitian

2.1. Dataset

Data berasal dari *Amazon Reviews'23 Dataset* kategori produk *fitness* dan olahraga sebanyak 19.613 produk. Atribut yang digunakan dan fungsinya ditunjukkan pada Tabel 1.

Tabel 1. Atribut *dataset* dan fungsi penelitian

Atribut	Status	Fungsi
<i>Product_id</i>	Dipakai	Identitas produk
<i>Product_name</i>	Dipakai	Nama dan ciri spesifik produk
<i>category</i>	Dipakai	Kategori produk
<i>clean_text</i>	Dipakai	Informasi tekstual dan ulasan
<i>average_rating</i> , <i>rating_count</i> , <i>price</i>	Tidak dipakai	Tidak digunakan dalam perhitungan kemiripan

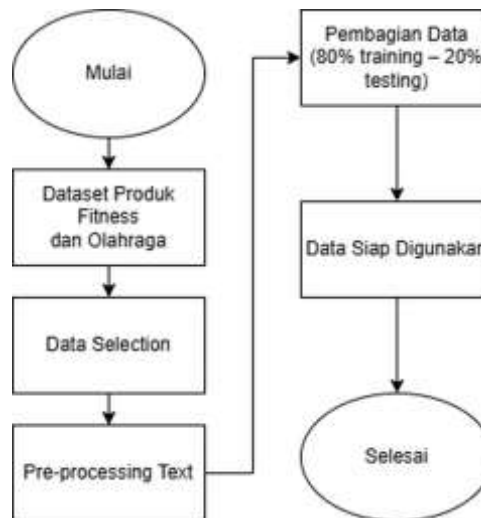
2.2. Preprocessing

Tahap *preprocessing* membersihkan dan menyeragamkan teks sebelum pemodelan. Prosesnya mencakup *data selection*, *cleaning text*, *case folding*, *tokenizing*, dan *stopword removal*. Nilai kategori *Other* dikosongkan, kemudian *Product_name*, *category*, dan *clean_text* digabungkan menjadi satu dokumen produk.

DOI: <https://doi.org/10.69693/ijmst.v4i3.13272>

Lisensi: Creative Commons Attribution 4.0 International (CC BY 4.0)

Data dibagi menggunakan *train-test split* dengan rasio 80% data latih dan 20% data uji serta *random_state* 42. Data latih digunakan untuk melatih *Doc2Vec*, sedangkan data uji digunakan dalam rekomendasi dan evaluasi. Tahapan tersebut ditunjukkan pada Gambar 2.



Gambar 2. Alur Preprocessing Data

2.3. Content-Based Filtering

Content-based filtering memberikan rekomendasi berdasarkan kesamaan konten *item* dengan kebutuhan pengguna. Pada penelitian ini, profil kebutuhan tidak dibentuk dari riwayat klik atau pembelian, tetapi dari *query* yang dimasukkan pengguna. *Query* dan dokumen produk direpresentasikan pada ruang vektor yang sama. Skor rekomendasi dihitung melalui fungsi kemiripan seperti pada Persamaan (1).

$$Score(q, d) = (V_q, V_d) \quad (1)$$

Pada Persamaan (1), q menunjukkan *query* pengguna dan d menunjukkan dokumen produk. V_q merupakan vektor *query* hasil inferensi *Doc2Vec*, sedangkan V_d merupakan vektor dokumen produk. Fungsi $sim(V_q, V_d)$ menggunakan *cosine similarity*. Produk dengan nilai *Score* tertinggi ditempatkan pada posisi teratas daftar rekomendasi. Mekanisme ini sesuai dengan prinsip *content-based filtering* yang memanfaatkan karakteristik *item* tanpa bergantung pada pengguna lain (Ko dkk., 2022; Nugroho dkk., 2024).

2.4. Implementasi Doc2Vec

Setiap dokumen hasil praproses diubah menjadi *TaggedDocument* dan digunakan untuk melatih *Doc2Vec* dengan arsitektur *Paragraph Vector Distributed Bag-of-Words (PV-DBOW)*. Arsitektur ini mempelajari hubungan dokumen dan kata tanpa bergantung pada urutan kata sehingga sesuai untuk teks ulasan yang memiliki panjang dan struktur beragam. Secara umum, proses representasi dokumen dinyatakan pada Persamaan (2).

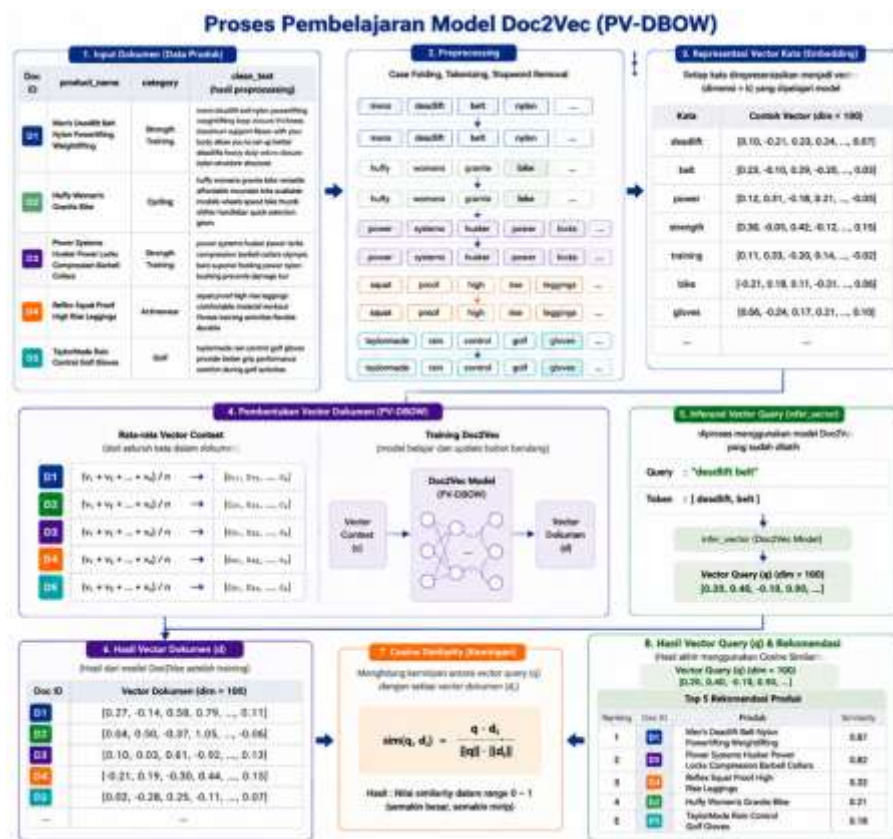
$$d = f(D) \quad (2)$$

Pada Persamaan (2), D adalah dokumen gabungan *Product_name*, *category*, dan *clean_text*, $f(\cdot)$ adalah fungsi pembelajaran *Doc2Vec*, dan d adalah vektor dokumen yang dihasilkan. Vektor tersebut menyimpan karakteristik semantik dokumen dan menjadi masukan bagi proses pengukuran kemiripan. *Doc2Vec* mampu mempertahankan informasi makna pada tingkat dokumen sehingga sesuai untuk analisis kesamaan teks (Iskandar & Kurniawati, 2025; Li dkk., 2025).

Tabel 1. Parameter Pelatihan Model *Doc2Vec*

Parameter	Nilai	Keterangan
<i>vector_size</i>	100	Jumlah dimensi vektor dokumen
<i>window</i>	5	Jumlah kata dalam konteks pembelajaran
<i>min_count</i>	2	Frekuensi minimum kata agar diproses
<i>epochs</i>	20	Jumlah iterasi pelatihan
<i>dm</i>	0	Arsitektur <i>PV-DBOW</i>
<i>workers</i>	1	Jumlah <i>thread</i> pelatihan
<i>seed</i>	42	Menjaga konsistensi hasil

Model menghasilkan vektor dokumen berdimensi 100. *Query* pengguna menjalani praproses yang sama, lalu fungsi *infer_vector* menghasilkan vektor *query* dengan dimensi yang sama. Model hanya dilatih menggunakan data latih. Vektor dokumen pada data uji juga dibentuk melalui *infer_vector* sehingga evaluasi mencerminkan kemampuan model menghadapi dokumen yang tidak digunakan saat pelatihan. Diagram proses pembelajaran model *Doc2Vec* dengan arsitektur *PV-DBOW* ditunjukkan pada Gambar 3.



Gambar 3. Proses Pembelajaran Model *Doc2Vec* (*PV-DBOW*)

2.5. Implementasi *Cosine Similarity*

Cosine similarity mengukur kesamaan arah antara vektor *query* dan vektor dokumen. Metode ini digunakan karena perhitungan menitikberatkan pada orientasi vektor sehingga perbedaan panjang teks tidak langsung mendominasi skor kemiripan. Perhitungan *cosine similarity* ditunjukkan pada Persamaan (3).

$$\text{CosineSimilarity}(q, d) = \frac{q \cdot d}{\|q\| \|d\|} \quad (3)$$

Pada Persamaan (3), $q \cdot d$ merupakan *dot product* antara vektor *query* dan dokumen. $\|q\|$ dan $\|d\|$ merupakan norma masing-masing vektor. Nilai *cosine similarity* berada pada rentang -1 sampai 1. Nilai yang semakin mendekati 1 menunjukkan arah vektor yang semakin serupa. Sistem mengurutkan produk berdasarkan nilai tertinggi untuk menghasilkan daftar *Top-N*. Penggunaan *Doc2Vec* dan *cosine similarity* mendukung pencarian dokumen berdasarkan hubungan semantik, bukan hanya kesamaan kata secara langsung (Iskandar & Kurniawati, 2025).

2.6 Evaluasi Model

Evaluasi menggunakan *Precision@N*, *Recall@N*, dan *NDCG@N*. *Precision@N* mengukur proporsi produk relevan di antara N produk pada daftar rekomendasi, sedangkan *Recall@N* mengukur proporsi produk relevan yang berhasil ditemukan pada Top-N dibandingkan dengan seluruh produk relevan untuk *query* pada data evaluasi. *NDCG@N* menilai kualitas urutan dengan memberi bobot lebih besar pada produk relevan yang berada di posisi atas. Penggunaan ketiga metrik pada evaluasi Top-N sejalan dengan kajian evaluasi sistem rekomendasi mutakhir (Bauer dkk., 2024; Akhadam dkk., 2025), sedangkan formulasi Top-N pada penelitian ini mengikuti evaluasi yang digunakan oleh Saifudin dkk. (2025).

Precision@N dihitung sebagai jumlah produk relevan yang muncul pada Top-N dibagi dengan N, yaitu jumlah produk yang direkomendasikan. Dengan demikian, penyebut *Precision@N* pada skenario Top-5, Top-10, dan Top-30 masing-masing adalah 5, 10, dan 30. Nilai *Precision@N* yang tinggi menunjukkan bahwa sebagian besar produk dalam daftar Top-N relevan terhadap *query* (Saifudin dkk., 2025).

$$\text{Precision@N} = \frac{|\text{Relevant items in Top} - N|}{N} \quad (4)$$

Recall@N dihitung sebagai jumlah produk relevan yang muncul pada Top-N dibagi dengan jumlah seluruh produk relevan untuk *query* pada data evaluasi, bukan dibagi dengan N. Penyebutnya karena itu dapat berbeda untuk setiap *query*. Nilai *Recall@N* yang tinggi menunjukkan bahwa sistem berhasil menemukan bagian yang lebih besar dari keseluruhan produk relevan (Saifudin dkk., 2025).

$$\text{Recall@N} = \frac{|\text{Relevant items in Top} - N|}{|\text{Total relevant items for query}|} \quad (5)$$

NDCG@K diperoleh dengan membagi *DCG@K* dengan *IDCG@K*. Nilainya berada pada rentang 0 sampai 1. Nilai yang semakin mendekati 1 menunjukkan bahwa urutan rekomendasi semakin mendekati kondisi ideal, sedangkan nilai yang mendekati 0 menunjukkan kualitas pemeringkatan yang rendah (Stergiopoulos dkk., 2024; Jadon & Patil, 2025).

$$\text{NDCG@K} = \frac{\text{DCG@K}}{\text{IDCG@K}} \quad (6)$$

Evaluasi dilakukan menggunakan empat *query*, yaitu *resistance band for beginners*, *adjustable dumbbell for home workout*, *yoga mat for exercise*, dan *exercise bike*. Setiap *query* diuji pada skenario Top-5, Top-10, dan Top-30 untuk menganalisis perubahan ketepatan, cakupan, dan kualitas urutan rekomendasi ketika jumlah produk yang ditampilkan bertambah.

3. Hasil dan Pembahasan

Pengujian dilakukan menggunakan data produk *fitness* dan olahraga dari *Amazon Reviews'23 Dataset*. Hasil penelitian disajikan sesuai urutan pada laporan, yaitu implementasi *preprocessing* data, pelatihan model *Doc2Vec*, implementasi sistem rekomendasi, hasil evaluasi sistem, dan implementasi *website* Bolang. Hasil setiap tahap

disajikan melalui uraian, tabel, grafik, dan tampilan antarmuka agar pembahasan berfokus pada keluaran penelitian.

3.1. Hasil

Hasil penelitian mencakup implementasi *preprocessing* data, pelatihan model Doc2Vec, implementasi sistem rekomendasi, evaluasi sistem, dan implementasi *website* Bolang. Urutan subbab mengikuti tahapan hasil pada laporan penelitian agar hubungan antara pengolahan data, pembentukan vektor, pemeringkatan, dan penyajian rekomendasi dapat ditelusuri secara sistematis.

3.1.1 Implementasi *Preprocessing* Data

Tahap *preprocessing* menggunakan *dataset* yang memuat *Product_id*, *Product_name*, *category*, *average_rating*, *rating_count*, *price*, dan *clean_text*. Proses rekomendasi menggunakan *Product_name*, *category*, dan *clean_text* sebagai sumber informasi tekstual, sedangkan *Product_id* berfungsi sebagai identitas produk. Atribut *average_rating*, *rating_count*, dan *price* tidak digunakan karena penelitian berfokus pada kemiripan konten.

Atribut *Product_name*, *category*, dan *clean_text* digabungkan menjadi satu dokumen untuk setiap produk. Kategori *Other* dihilangkan karena tidak memberikan informasi kategori yang spesifik. *Dataset* kemudian dibagi menggunakan *hold-out validation* dengan komposisi 80% data latih dan 20% data uji pada *random_state* 42. Data latih digunakan untuk membangun model *Doc2Vec*, sedangkan data uji digunakan untuk menguji dan mengevaluasi sistem rekomendasi.

3.1.2 Pelatihan Model *Doc2Vec*

Dokumen pada data latih diubah menjadi *TaggedDocument*. Setiap dokumen dipecah menjadi token huruf kecil dan diberi tag berupa indeks unik agar model dapat membedakan setiap dokumen selama proses pembelajaran. Tahap ini menghasilkan data masukan yang terstruktur untuk pelatihan model *Doc2Vec*.

Model *Doc2Vec* dilatih menggunakan arsitektur *Paragraph Vector Distributed Bag of Words (PV-DBOW)* dengan *dm* bernilai 0. Konfigurasi parameter pelatihan telah dijelaskan pada Tabel 1 di bagian metode penelitian sehingga tidak ditampilkan kembali pada bagian hasil.

Setelah proses pelatihan selesai, model menghasilkan representasi vektor dokumen berdimensi 100. Model *Doc2Vec* mempelajari hubungan semantik antara dokumen dan kata pada data latih. Dokumen pada data uji direpresentasikan menggunakan fungsi *infer_vector* sehingga setiap produk memiliki vektor pada ruang representasi yang sama dengan *query* pengguna.

3.1.3 Implementasi Sistem Rekomendasi

Sistem rekomendasi menerima *query* pengguna, menjalankan *preprocessing*, membentuk vektor *query* menggunakan *infer_vector*, lalu menghitung *cosine similarity* terhadap seluruh vektor produk pada data uji. Nilai *similarity* diurutkan dari nilai tertinggi sehingga produk dengan kedekatan konten terbesar ditempatkan pada peringkat awal.

Sistem menyimpan representasi vektor *query* yang telah terbentuk agar *query* yang sama menghasilkan keluaran yang konsisten. Pengujian menggunakan empat *query* yang mewakili kebutuhan produk pada beberapa kategori, yaitu *resistance band for beginners*, *adjustable dumbbell for home workout*, *yoga mat for exercise*, dan *exercise bike*.

Setiap *query* menghasilkan 50 kandidat produk dengan nilai *cosine similarity* tertinggi. Daftar *Top-50* tersebut digunakan sebagai dasar evaluasi pada skenario *Top-5*, *Top-10*, dan *Top-30*. Contoh sepuluh rekomendasi teratas untuk *query resistance band for beginners* disajikan pada Tabel 2.

Tabel 2. Hasil Rekomendasi untuk Query “Resistance Band for Beginners”

Rank	Product Name	Category	Similarity
1	Resistance Bands	Fitness & Cardio	0,741259217
2	YJan Resistance Bands Set	Fitness & Cardio	0,677715063
3	Mohigher Resistance Loop Bands	Yoga & Pilates	0,665315092
4	Impex Marcy Kettlebell Weight Set	Strength Training	0,66449985
5	Pasnity Hand Grip Strengtheners	Fitness & Cardio	0,657420456
6	ZSOQ Pull Up Assist Resistance Bands	Yoga & Pilates	0,656843007
7	HEAD Liquidmetal 8 Tennis Racquet	Tennis	0,656267881
8	SunFeeling Resistance Bands Set	Fitness & Cardio	0,654103041
9	VSNOON Resistance Band Set	Fitness & Cardio	0,647708595
10	Senston Junior Tennis Racket	Tennis	0,647339225

Resistance Bands memperoleh nilai tertinggi sebesar 0,7413 dan menempati peringkat pertama. Produk lain yang memiliki hubungan langsung dengan resistance band juga berada pada posisi atas. Namun, *HEAD Liquidmetal 8 Tennis Racquet* dan *Senston Junior Tennis Racket* turut muncul karena model mempertimbangkan kedekatan vektor dari keseluruhan informasi teks. Hasil ini menunjukkan bahwa Doc2Vec menangkap hubungan semantik, tetapi kemiripan konteks olahraga dapat menyebabkan produk berbeda memperoleh nilai *similarity* yang cukup tinggi.

3.1.4 Hasil Evaluasi Sistem

Evaluasi menggunakan Precision@N, Recall@N, dan NDCG@N. Sebelum metrik dihitung, setiap produk diberi *relevance score* secara otomatis berdasarkan kesesuaian *query* dengan *Product_name*. Nilai 3 menunjukkan produk sangat relevan, nilai 2 relevan, nilai 1 kurang relevan, dan nilai 0 tidak relevan. Untuk Precision@N dan Recall@N, produk dengan skor lebih dari 0 diperlakukan sebagai relevan secara biner, sedangkan NDCG@N tetap menggunakan skor bertingkat 0–3. Skala penilaian relevansi disajikan pada Tabel 3.

Tabel 3. Skala Penilaian Relevansi

Nilai	Keterangan
3	Sangat Relevan
2	Relavan
1	Kurang Relevan
0	Tidak Relevan

Setelah nilai relevansi diperoleh, keempat *query* diuji pada *Top-5*, *Top-10*, dan *Top-30*. Hasil *Precision*, *Recall*, dan *NDCG* masing-masing ditunjukkan pada Tabel 4, Tabel 5, dan Tabel 6. Rata-rata hasil evaluasi disajikan pada Tabel 7.

Tabel 4. Hasil Evaluasi *Precision*

No	Query	Precision Top 5	Precision Top 10	Precision Top 30
1	Resistance band for beginners	0,8	0,7	0,8333
2	Adjustable dumbbell	1,0	0,9	0,8667
3	Yoga mat	1,0	1,0	0,9667
4	Exercise bike	1,0	1,0	0,8333

Tabel 4 menunjukkan bahwa *Precision* umumnya tinggi. Pada *Top-5*, tiga *query* memperoleh nilai 1,0000 dan *query resistance band for beginners* memperoleh 0,8000. Hasil tersebut menunjukkan bahwa produk pada peringkat awal memiliki tingkat kesesuaian yang baik terhadap kebutuhan pengguna.

Tabel 5. Hasil Evaluasi *Recall*

No	Query	Recall Top 5	Recall Top 10	Recall Top 30
1	<i>Resistance band for beginners</i>	0.1026	0.1795	0.641
2	<i>Adjustable dumbbell</i>	0,1163	0.2093	0.6047
3	<i>Yoga mat</i>	0.102	0.2041	0.5918
4	<i>Exercise bike</i>	0,119	0.2381	0.5952

Tabel 5 memperlihatkan peningkatan *Recall* ketika jumlah rekomendasi bertambah. Nilai tertinggi pada *Top-30* mencapai 0,6410 untuk *query resistance band for beginners*. Peningkatan tersebut menunjukkan bahwa daftar rekomendasi yang lebih panjang mampu menemukan lebih banyak produk relevan.

Tabel 6. Hasil Evaluasi *NDCG*

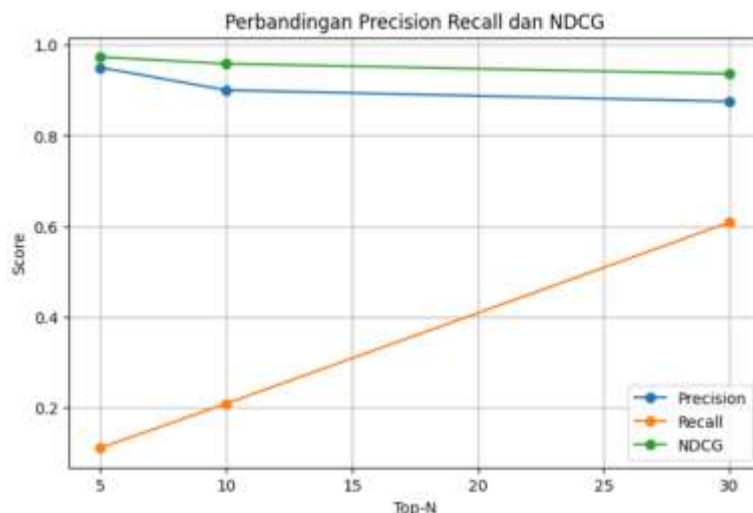
No	Query	NDCG Top 5	NDCG Top 10	NDCG Top 30
1	<i>Resistance band for beginners</i>	0,9879	0.9531	0.929
2	<i>Adjustable dumbbell</i>	0.9336	0.9357	0.9201
3	<i>Yoga mat</i>	0.9711	0.974	0.9512
4	<i>Exercise bike</i>	1.0	0.9706	0.9442

Nilai *NDCG* pada Tabel 6 berada pada rentang 0,9201 sampai 1,0000. Nilai yang mendekati 1 menunjukkan bahwa produk relevan secara konsisten ditempatkan pada posisi atas. *Query exercise bike* memperoleh *NDCG* 1,0000 pada *Top-5*, sedangkan nilai terendah sebesar 0,9201 terdapat pada *query adjustable dumbbell* pada *Top-30*.

Tabel 7. Rata-Rata Hasil Evaluasi Sistem

Evaluasi	Top 5	Top 10	Top 30
<i>Precision</i>	0,95	0,9	0,875
<i>Recall</i>	0,11	0,2078	0,6082
<i>NDCG</i>	0.9732	0.9584	0.9361

Berdasarkan Tabel 7, rata-rata *Precision* turun dari 0,9500 pada *Top-5* menjadi 0,8750 pada *Top-30*. Sebaliknya, *Recall* meningkat dari 0,1100 menjadi 0,6082. Rata-rata *NDCG* tetap tinggi, yaitu 0,9732 pada *Top-5*, 0,9584 pada *Top-10*, dan 0,9361 pada *Top-30*. Hasil tersebut didasarkan pada relevansi otomatis sehingga belum sepenuhnya menggambarkan penilaian semantik pengguna atau pakar.

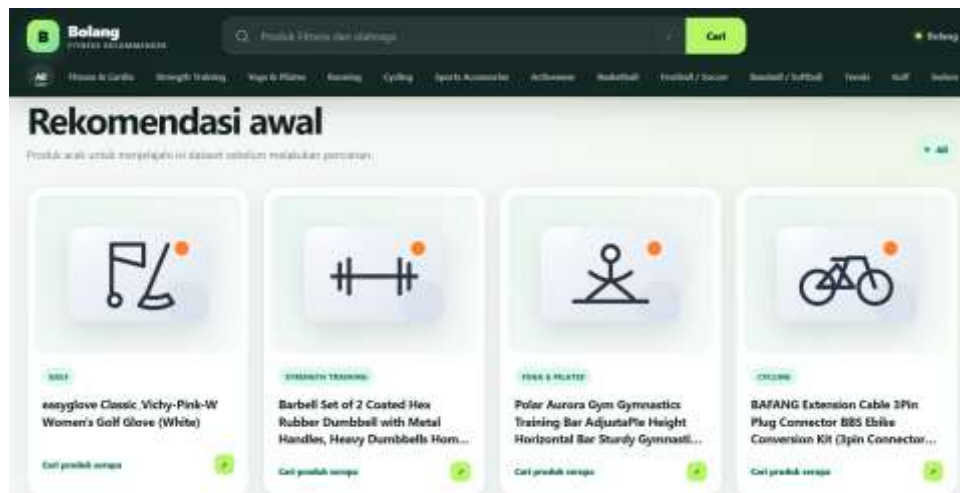


Gambar 4. Grafik Hasil Evaluasi Sistem

Gambar 4 memperlihatkan bahwa *Precision* dan *NDCG* menurun secara bertahap ketika jumlah rekomendasi bertambah, sedangkan *Recall* meningkat secara signifikan. Pola tersebut menunjukkan perubahan antara ketepatan, cakupan, dan kualitas urutan pada setiap skenario *Top-N*.

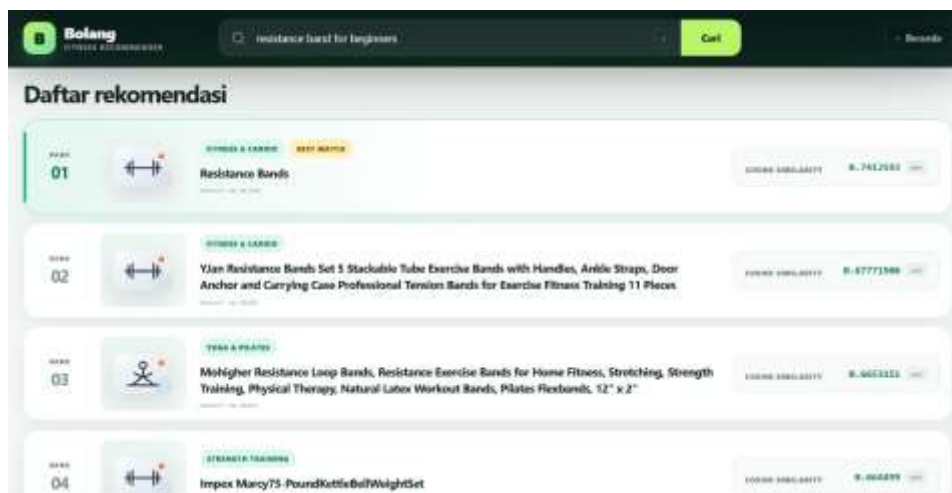
3.1.5 Implementasi Website

Model diimplementasikan pada *website* Bolang menggunakan *framework* Flask. Halaman beranda menyediakan logo, kolom pencarian, menu kategori, rekomendasi awal, dan kartu produk. Pengguna dapat memasukkan kata kunci atau memilih fitur pencarian produk serupa. Sistem menjalankan *preprocessing*, membentuk vektor *query* menggunakan Doc2Vec, menghitung *cosine similarity*, lalu menyusun produk berdasarkan nilai tertinggi.



Gambar 5. Halaman Beranda Website Bolang

Gambar 5 menunjukkan halaman beranda. Menu kategori mencakup All, *Fitness & Cardio*, *Strength Training*, *Yoga & Pilates*, *Running*, dan kategori olahraga lain. Rekomendasi awal ditampilkan secara acak untuk membantu pengguna menjelajahi produk sebelum menjalankan pencarian khusus. Setiap kartu memuat kategori, nama produk, dan tombol Cari produk serupa yang menggunakan nama produk sebagai *query* baru.



Gambar 6. Halaman Hasil Rekomendasi Website Bolang

Gambar 6 menampilkan hasil untuk *query* *resistance band for beginners*. Produk disusun berdasarkan peringkat dan nilai *cosine similarity*. *Resistance Bands* memperoleh nilai 0,7412592 dan diberi label *Best Match*. Halaman ini juga menampilkan kategori, nama produk, nomor baris *dataset*, tombol penyalinan nilai, serta kolom pencarian

baru. *Query* di luar domain *fitness* dan olahraga dapat menghasilkan rekomendasi yang kurang sesuai karena model hanya mempelajari kosakata dan informasi pada *dataset* penelitian.

3.2. Pembahasan.

Pembahasan menganalisis hubungan antara keluaran implementasi dan kualitas rekomendasi. Analisis difokuskan pada performa sistem, faktor pembentuk hasil, penggunaan pembagian data 80:20, karakteristik *Top-N*, serta keunggulan dan keterbatasan sistem rekomendasi berbasis *content-based filtering* yang dikembangkan.

3.2.1 Analisis Performa Sistem

Sistem mampu mengubah *query* menjadi vektor Doc2Vec dan membandingkannya dengan vektor produk menggunakan *cosine similarity*. Pada *query resistance band for beginners*, produk *Resistance Bands* berada pada peringkat pertama dengan nilai 0,7412592, diikuti *YJan Resistance Bands Set* sebesar 0,67771506 dan *Mohigher Resistance Loop Bands* sebesar 0,6653151. Urutan tersebut menunjukkan bahwa sistem dapat mengenali keterkaitan antara kebutuhan pengguna dan informasi produk, kemudian menyusun hasil dari nilai kemiripan tertinggi.

Performa evaluasi menunjukkan pola yang konsisten. *Precision* dan *NDCG* paling tinggi terdapat pada daftar yang lebih pendek, sedangkan *Recall* meningkat ketika jumlah rekomendasi diperbesar. *NDCG* yang tetap berada di atas 0,93 pada seluruh nilai rata-rata *Top-N* menunjukkan bahwa produk yang dinilai relevan tetap dominan pada posisi awal meskipun jumlah hasil bertambah.

3.2.2 Analisis Faktor yang Memengaruhi Hasil Rekomendasi

Hasil rekomendasi dipengaruhi oleh lima faktor utama. Pertama, kelengkapan *Product_name*, *category*, dan *clean_text* menentukan informasi yang dipelajari model. Kedua, *preprocessing* memengaruhi kebersihan dan konsistensi token. Ketiga, pembagian data menentukan jumlah dokumen yang digunakan untuk pelatihan dan pengujian. Keempat, parameter *vector_size*, *window*, *min_count*, *epochs*, *dm*, *workers*, dan *seed* memengaruhi pembentukan ruang vektor. Kelima, susunan kata dalam *query* menentukan kedekatannya dengan kosakata pada *dataset*.

Produk yang memiliki istilah berbeda tetap dapat muncul ketika konteks keseluruhan dokumennya berdekatan. Kondisi tersebut menjadi kelebihan representasi semantik, tetapi juga dapat menghasilkan rekomendasi yang kurang tepat. Kemunculan produk raket tenis pada *query resistance band for beginners* menunjukkan bahwa kedekatan konteks olahraga belum selalu identik dengan kesesuaian jenis produk.

3.2.3 Analisis Pembagian Data

Data dibagi menjadi 80% data latih dan 20% data uji. Porsi data latih yang lebih besar memberi kesempatan bagi Doc2Vec untuk mempelajari pola teks produk, sedangkan data uji digunakan untuk menilai kemampuan sistem pada dokumen yang tidak dilibatkan dalam pelatihan. Vektor data uji dibentuk melalui *infer_vector* sehingga evaluasi mendekati kondisi ketika sistem menerima dokumen baru.

Konfigurasi 80:20 digunakan sebagai dasar pelatihan dan evaluasi, bukan sebagai skenario perbandingan beberapa rasio. Oleh sebab itu, hasil penelitian tidak dapat digunakan untuk menyimpulkan bahwa rasio tersebut lebih baik daripada 70:30 atau 90:10. Perbandingan beberapa rasio dapat dilakukan pada penelitian berikutnya untuk menguji kestabilan representasi dan perubahan kualitas rekomendasi.

3.2.4 Analisis Top-N Rekomendasi

Top-N membatasi jumlah produk yang ditampilkan dari daftar yang telah diurutkan berdasarkan *cosine similarity*. *Top-5* menghasilkan rata-rata *Precision* 0,9500 dan *NDCG* 0,9732 sehingga sesuai ketika sistem memprioritaskan ketepatan dan kualitas peringkat awal. *Top-30* menurunkan *Precision* menjadi 0,8750 dan *NDCG* menjadi 0,9361, tetapi meningkatkan *Recall* menjadi 0,6082.

Temuan tersebut menunjukkan *trade-off* antara ketepatan dan cakupan. Daftar pendek mengurangi peluang masuknya produk yang kurang relevan, sedangkan daftar panjang meningkatkan jumlah produk relevan yang berhasil ditemukan. Pemilihan *Top-N* perlu disesuaikan dengan tujuan antarmuka. Pencarian cepat dapat menggunakan *Top-5* atau *Top-10*, sedangkan eksplorasi produk yang lebih luas dapat menggunakan *Top-30*.

3.2.5 Keunggulan dan Keterbatasan Sistem

Sistem memiliki keunggulan berupa penggunaan representasi semantik Doc2Vec, penggabungan tiga atribut tekstual, pemeringkatan menggunakan *cosine similarity*, evaluasi pada beberapa *Top-N*, dan implementasi dalam *website*. Pendekatan *content-based filtering* tidak membutuhkan riwayat klik, *rating* personal, atau transaksi sehingga dapat diterapkan ketika data interaksi pengguna belum tersedia.

Keterbatasannya meliputi penggunaan satu domain produk, ketergantungan pada kualitas teks, satu konfigurasi parameter dan pembagian data, serta relevansi otomatis berbasis pencocokan kata. Penilaian tersebut belum sepenuhnya merepresentasikan penilaian semantik manusia. Sistem juga belum menggunakan riwayat pencarian, klik, *rating*, atau pembelian sehingga rekomendasi belum bersifat personal bagi setiap pengguna.

4. Kesimpulan

Penelitian berhasil membangun sistem rekomendasi produk *fitness* dan olahraga berbasis *content-based filtering* menggunakan Doc2Vec dan *cosine similarity*. *Product_name*, *category*, dan *clean_text* digabungkan menjadi dokumen produk. Model Doc2Vec PV-DBOW menghasilkan vektor berdimensi 100, sedangkan *query* direpresentasikan melalui *infer_vector*. *Cosine similarity* menghitung kemiripan *query* dengan dokumen produk dan menjadi dasar pemeringkatan *Top-N*. Hasil evaluasi menunjukkan rata-rata *Precision*, *Recall*, dan *NDCG* sebesar 0,9500, 0,1100, dan 0,9732 pada *Top-5*; 0,9000, 0,2078, dan 0,9584 pada *Top-10*; serta 0,8750, 0,6082, dan 0,9361 pada *Top-30*. *Top-N* yang lebih kecil menghasilkan ketepatan dan kualitas urutan lebih tinggi. *Top-N* yang lebih besar meningkatkan kemampuan sistem menemukan produk relevan. Kombinasi Doc2Vec dan *cosine similarity* dapat digunakan untuk menghasilkan rekomendasi berdasarkan kemiripan semantik konten produk. Sistem telah diimplementasikan sebagai *website* Bolang berbasis Flask.

Reference

1. Akhadam, A., Kbibchi, O., Mekouar, L., & Iraqi, Y. (2025). A comparative evaluation of recommender systems tools. IEEE Access, 13, 29493-29522. <https://doi.org/10.1109/ACCESS.2025.3541014>
2. Bauer, C., Zangerle, E. V. A., & Said, A. (2024). Exploring the landscape of recommender systems evaluation: Practices and perspectives. ACM Transactions on Recommender Systems, 2(1). <https://doi.org/10.1145/3629170>
3. Iskandar, D., & Kurniawati, A. (2025). Analisis perbandingan teknik Word2Vec dan Doc2Vec dalam mengukur kemiripan dokumen menggunakan cosine similarity. Jurnal Teknologi Informasi dan Ilmu Komputer, 12(1), 133-144. <https://doi.org/10.25126/jtiik.2025129143>
4. Jadon, A., & Patil, A. (2025). A comprehensive survey of evaluation techniques for recommendation systems. In Computation of Artificial Intelligence and Machine Learning (pp. 281-304). Springer. https://doi.org/10.1007/978-3-031-71484-9_25
5. Jannach, D. (2023). Evaluating conversational recommender systems. Artificial Intelligence Review, 56, 2365-2400. <https://doi.org/10.1007/s10462-022-10229-x>
6. Khadka, A., & Sthapit, S. (2025). A review on scholarly publication recommender systems: Features, approaches, evaluation, and open research directions. Informatics, 12(4), 108. <https://doi.org/10.3390/informatics12040108>
7. Ko, H., Lee, S., Park, Y., & Choi, A. (2022). A survey of recommendation systems: Recommendation models, techniques, and application fields. Electronics, 11(1), 141. <https://doi.org/10.3390/electronics11010141>
8. Li, B., Wang, W., Huang, Z., Liu, Z., Wang, H., Feng, Q., & Zhao, N. (2025). Doc2Vec-based similarity analysis of tobacco standard texts. International Journal of Modern Physics C, 37(6). <https://doi.org/10.1142/S0129183125420100>
9. Nugroho, R. G., Sumarlinda, S., & Srirahayu, A. (2024). Camping equipment recommendation system using content-based filtering method: A case study of Berkah Outdoor45. International Journal Software Engineering and Computer Science, 4(3), 1077-1088. <https://doi.org/10.35870/ijsecs.v4i3.3078>
10. Polatidis, N., Baimagambetov, A., Kehagias, D., Trovati, M., Sarigiannidis, P., et al. (2026). Datasets for recommender systems: A survey and comparative overview. Journal of Intelligent Information Systems. <https://doi.org/10.1007/s10844-026-01068-1>
11. Rainio, O., Teuho, J., & Klen, R. (2024). Evaluation metrics and statistical tests for machine learning. Scientific Reports, 14, 6086. <https://doi.org/10.1038/s41598-024-56706-x>
12. Roy, D., & Dutta, M. (2022). A systematic review and research perspective on recommender systems. Journal of Big Data, 9, 59. <https://doi.org/10.1186/s40537-022-00592-5>
13. Saifudin, I., & Widiyaningtyas, T. (2024). Systematic literature review on recommender system: Approach, problem, evaluation techniques, datasets. IEEE Access, 12, 19827-19847. <https://doi.org/10.1109/ACCESS.2024.3359274>
14. Saifudin, I., Widiyaningtyas, T., Zaeni, I. A. E., & Aminuddin, A. (2025). SVD-GoRank: Recommender system algorithm using SVD and Gower's ranking. IEEE Access, 13, 17931-17945. <https://doi.org/10.1109/ACCESS.2025.3533558>
15. Stergiopoulos, V., Vassilakopoulos, M., Tousidou, E., & Corral, A. (2024). An academic recommender system on large citation data based on clustering, graph modeling and deep learning. Knowledge and Information Systems, 66(8). <https://doi.org/10.1007/s10115-024-02094-7>

DOI: <https://doi.org/10.69693/ijmst.v4i3.13272>

Lisensi: Creative Commons Attribution 4.0 International (CC BY 4.0)