

# Implementasi Vector Space Model Pada Sistem Pencarian MP3 Player

I Ketut Teguh Wibawa Lessmana Putra. T<sup>1</sup>, I Gusti Agung Gede Arya  
Kadyanan, S.Kom., M.Kom.<sup>2</sup>

Program Studi Informatika  
Fakultas Matematika dan Ilmu Pengetahuan Alam  
Universitas Udayana

<sup>1</sup>teguhlessmana01@gmail.com, <sup>2</sup>gungde@unud.ac.id

## Abstract

Perkembangan industri musik menghasilkan karya musik yang beragam, sehingga banyak sekali judul, genre dan penyanyi yang berbeda-beda. Untuk memudahkan pencarian musik yang menarik, perlu adanya sistem layanan informasi dan penyanyi. Untuk dapat mempermudah pencarian musik yang diminati dibutuhkan pembangunan sistem *information retrieval* (IR) atau sistem temu kembali informasi. Salah satu sistem *information retrieval* (IR) atau sistem temu kembali informasi yang sangat populer yaitu *vector space model*. *Vector space model* (VSM) adalah salah satu metode atau algoritma yang sering digunakan untuk sebuah sistem temu kembali informasi. Tujuan dari penelitian ini adalah untuk mengetahui penerapan *vector space model* dalam sistem pencarian MP3 *player*. Hasil penelitian menunjukkan bahwa implementasi *vector space model* pada sistem pencarian MP3 *player* dapat mempermudah pencarian musik atau lagu sesuai dengan keinginan dari pengguna. Melalui sistem ini pengguna dapat melihat nilai, dan juga rekomendasi terbaik dari berbagai *file* yang tersedia tergantung dari kata kunci yang dimasukkan pengguna berikan ke dalam sistem..

**Keywords:** Informasi, MP3, Musik, *Retrieval*, Sistem

## 1. Pendahuluan

Menurut Prey (2018), musik adalah bentuk yang terkait erat dengan ingatan, aspirasi, kehidupan sehari-hari, dan kelompok sosial tertentu, dan karenanya merupakan bentuk suara yang dipandang sebagai alat yang berguna yang memungkinkan individu untuk 'melihat'. Musik sebagai sarana hiburan yang dinikmati di seluruh dunia mencakup banyak genre. Musik ada di mana-mana, baik itu di film, iklan, video game, bahkan nada dering, menawarkan musik sebagai pengganti lonceng tradisional (Dewatara dan Agustin, 2019). Industri musik dunia berkembang pada tahun 70-an dan 80-an dan memuncak pada tahun 90-an berkat munculnya talenta-talenta muda. Antusiasme penonton untuk menyaksikan musisi idola mereka di setiap panggung cukup tinggi. Saat itu, industri musik berbasis pita magnetik menjadi pilihan utama konsumen, dimana toko kaset mudah ditemukan, penjualan kaset sangat tinggi di pasaran (Noviani et al., 2020).

Pada masa perkembangan industri musik sudah memasuki era digital. Industri musik digital dapat dibagi menjadi tiga tahap, yaitu: Tahap pertama adalah transisi dari teknologi kaset dan vinil ke CD. Fase ini memungkinkan produk digital (disk, kaset audio digital, kaset kompak digital, DVD) dijual dengan kualitas suara yang sama persis dengan rekaman digital aslinya. Fase kedua terjadi pada awal abad ke-21 dengan munculnya berbagi unduhan digital melalui teknologi kompresi MP3 yang dikembangkan oleh Motion Picture Experts Group (MPEG). Komersialisasi dan popularitas Internet publik dan munculnya jaringan peer-to-peer seperti Napster menyebabkan pembajakan musik. Unduhan digital tampak legal pada tahap ini, dan iTunes adalah yang pertama melakukannya; Fase ketiga adalah transisi saat ini, yaitu transisi ke streaming musik alih-alih menjual produk eksklusif secara

digital (baik gratis maupun berbasis langganan). Pertumbuhan model streaming ini telah mengurangi dampak ekonomi dari pembajakan musik digital (Devantara dan Agustin, 2019).

Salah satu pembawa audio musik yang cukup terkenal adalah MPEG (Moving Picture Expert Group)-1 Audio Layer 3 atau lebih dikenal dengan MP3. Pemutar MPEG (Moving Picture Expert Group)-1 Audio Layer 3 itu sendiri adalah pemutar MP3. Pemutar MP3 memungkinkan setiap orang untuk mendengarkan lagu secara digital tanpa kaset atau CD seperti sebelumnya, dan jutaan orang di seluruh dunia dapat berbagi rekaman musik dengan komputer yang terhubung ke Internet. Format MP3, sering dimainkan oleh pemutar MP3, adalah salah satu format audio paling populer untuk menyimpan data audio. Awalnya, MPEG Audio Layer-3 banyak digunakan oleh pengguna komputer. File MPEG Audio Layer-3 disimpan dengan ekstensi file MP3. Kemudian MPEG Audio Layer-3, biasanya disebut MP3 mulai sekarang. File MPEG terdiri dari bingkai kecil. Secara umum, setiap frame dapat berdiri sendiri. Setiap frame memiliki header yang berisi informasi tentang frame tersebut. File MPEG tidak memiliki file header, sehingga file MPEG dapat dipotong di mana saja selama masih dalam bingkai. Tidak seperti MP3, beberapa frame mungkin berisi bagian terkait. Header frame pertama dapat dibaca untuk membaca informasi yang terdapat dalam file MPEG. File MPEG yang menggunakan bit rate yang berbeda memiliki informasi frame yang berbeda (Sari et al., 2018).

Perkembangan industri musik Indonesia telah menghasilkan karya dalam genre musik yang sangat berbeda dan bahkan lebih beragam. Mengingat sifat masyarakat yang lebih menyukai apa yang tersedia, maka diperlukan sebuah aplikasi yang memudahkan pemilihan musik sesuai dengan kriteria pendengar. Oleh karena itu, diperlukan sistem layanan informasi dan temu kembali informasi dan information retrieval (IR). *Information Retrieval* (IR) atau sistem pencarian informasi merupakan salah satu cara untuk dengan mudah mencari informasi dari dokumen-dokumen yang terdapat dalam sistem. Search engine atau mesin pencari adalah suatu perangkat yang dapat memudahkan pengguna dalam mengakses informasi dengan menampilkan hasil pencarian berdasarkan informasi yang dicari pengguna. Search engine atau mesin pencari biasanya berupa website yang dibuat agar pengguna dapat menggunakannya untuk mencari dan menemukan informasi. Beberapa metode yang dapat digunakan untuk membangun sistem IR, yaitu model berbasis gravitasi, model ruang vektor, dan model ruang vektor tergeneralisasi (Salmon et al., 2020).

Sistem layanan informasi atau information service adalah suatu sistem yang proses utamanya secara otomatis mencari informasi yang berkaitan dengan kebutuhan pengguna dari sejumlah besar sumber informasi. Pengaplikasian dari sistem temu kembali informasi adalah search-engine atau mesin pencari. Sistem pencarian menggunakan dokumen sebagai objek data, sumber informasi. Dokumen diindeks oleh sistem tertentu seperti TF-IDF. Sistem temu kembali informasi yang baik dapat menampilkan dokumen relevan yang cocok dengan query yang dimasukkan, mengurutkan dokumen, dan membuang dokumen yang tidak relevan. Ada beberapa metode yang dapat Anda gunakan untuk mengindeks atau mencari dokumen terkait. Sistem temu kembali juga menggunakan presisi dan skor recall sebagai ukuran efektivitas sistem temu kembali informasi. Suatu sistem temu kembali informasi dikatakan baik jika nilai presisi dan recall sama (1:1) (Putra et al., 2019).

Sebuah sistem temu kembali informasi memiliki dua tugas, yaitu mengolah basis data terlebih dahulu dan kemudian menggunakan metode tertentu untuk menghitung tingkat kepentingan atau kemiripan dokumen dalam basis data dengan dokumen yang telah diproses sebelumnya dengan pertanyaan pengguna. Kueri yang dimasukkan pengguna diubah menurut aturan tertentu untuk mengekstrak istilah signifikan yang cocok dengan istilah yang diekstrak sebelumnya dari dokumen, dan berdasarkan istilah tersebut, relevansi antara kueri dan dokumen dihitung. Hasilnya, sistem mengembalikan daftar dokumen yang diurutkan berdasarkan nilai kemiripannya seperti yang diminta oleh pengguna (Anna dan Hendini, 2018). Dengan latar belakang ini, ada kebutuhan besar untuk pengembangan sistem berbasis temu kembali informasi untuk memfasilitasi pencarian dokumen tertentu yang berisi media pemutar MP3. Oleh karena itu, peneliti akhirnya melakukan penelitian yang disebut "Implementasi model ruang vektor dalam sistem pencarian pemutar MP3 Player".

## 2. Metode Penelitian

*Vector Spatial Model* (VSM) adalah teknik atau algoritma yang banyak digunakan dalam sistem data mining. Algoritma ini merupakan model yang menggunakan pembobotan term untuk mengukur kemiripan atau derajat kemiripan (similar terms) antara sebuah dokumen dengan sebuah query

pencarian. Sistem temu kembali informasi menentukan kesamaan dokumen berdasarkan bagan representasi kata dan mengubahnya menjadi model ruang vektor.

Pencocokan dokumen dengan kueri didasarkan pada kesamaan antara vektor dokumen dan vektor kueri. Ide dasar dari model ruang vektor adalah menghitung jarak antar dokumen dan kemudian mengurutkannya sesuai dengan tingkat kedekatannya. Pekerjaan model ruang vektor dimulai dengan pelipatan huruf, pembersihan data, pengindeksan, penyaringan, derivasi dan tokenisasi, yaitu. mengiris string input berdasarkan setiap subword dan membagi dokumen menjadi tabel kepadatan kata.

Semua kata dalam dokumen digabungkan menjadi satu kata yang disebut istilah. Setiap dokumen direpresentasikan sebagai vektor yang dibandingkan dengan istilah yang dihasilkan. Analisis kesamaan, yang mengukur kesamaan dokumen, dilakukan dengan menghitung kosinus jarak antar dokumen. Kita dapat menggunakan persamaan berikut untuk menggambarkan kinerja sistem model ruang vektor:

2.1 Proses perhitungan *Vector Space Model* melalui tahapan *perhitungan term frequency* (tf) menggunakan persamaan (1):

$$Tf = tf_{ij} \quad (1)$$

Dimana  $tf$  merupakan istilah frekuensi sedangkan  $tf_{ij}$  merupakan jumlah banyaknya suatu kemunculan di dalam dokumen tersebut yang diistilahkan dengan  $t_i$  dalam dokumen  $d_j$ , frekuensi istilah ( $tf$ ) dihitung dengan menghitung jumlah kemunculan istilah  $t_i$  dalam dokumen  $d_j$ . Kemudian.

Dengan  $tf$  adalah *term frequency*, dan  $tf_{ij}$  adalah banyaknya kemunculan term  $t_i$  dalam dokumen  $d_j$ , *Term frequency* ( $tf$ ) dihitung dengan menghitung banyaknya kemunculan term  $t_i$  dalam dokumen  $d_j$ .

2.2 Perhitungan *Inverse Document Frequency* ( $idf$ ), menggunakan persamaan (2)

$$idf_i = \frac{\log N}{df_i} \quad (2)$$

Dengan  $idf_i$  adalah *inverse document frequency*,  $N$  adalah jumlah dokumen yang terambil oleh sistem, dan  $df_i$  adalah banyaknya dokumen dalam koleksi dimana term  $t_i$  muncul di dalamnya, maka Perhitungan  $idf_i$  digunakan untuk mengetahui banyaknya term yang dicari ( $df_i$ ) yang muncul dalam dokumen lain yang ada pada database.

2.3 Perhitungan *Term Frequency Inverse Document Frequency* ( $tfidf$ ), menggunakan persamaan (3)

$$W_{ij} = tf_{ij} \cdot \log N / df_i \quad (3)$$

Dimana  $W_{ij}$  adalah bobot dokumen,  $N$  adalah jumlah dokumen yang diambil oleh sistem,  $tf_{ij}$  adalah jumlah kemunculan term  $t_i$  dalam dokumen  $d_j$ , dan  $df_i$  adalah jumlah dokumen dalam kumpulan dimana term  $t_i$  muncul di dalamnya. Bobot dokumen ( $W_{ij}$ ) dihitung untuk

mendapatkan produk berbobot atau kombinasi dari frekuensi ekspresi (tfij) dan frekuensi dokumen terbalik (idf).

#### 2.4 Perhitungan Jarak *query*, menggunakan persamaan (4)

$$|q| = \sqrt{\sum_{i=1}^t (W_{iq})^2} \quad (4)$$

Dengan  $|q|$  adalah jarak kueri dan  $W_{iq}$  merupakan bobot kueri dari dokumen ke-i, kemudian jarak kueri ( $|q|$ ) dihitung untuk mendapatkan jarak kueri bobot kueri ( $W_{iq}$ ) dari dokumen yang dicari oleh sistem. Jarak survei dapat dihitung menggunakan akar kuadrat dari persamaan survei.

#### 2.5 Perhitungan Jarak Dokumen, menggunakan persamaan (5)

$$|d_j| = \sqrt{\sum_{i=1}^t (W_{ij})^2} \quad (5)$$

Dengan  $|d_j|$  jarak dokumen dan  $W_{ij}$  adalah berat dokumen, kemudian jarak dokumen ( $|d_j|$ ) dihitung oleh sistem untuk mendapatkan jarak dokumen dari berat dokumen ( $W_{ij}$ ). Jarak dokumen dapat dihitung menggunakan persamaan akar kuadrat dari dokumen.

#### 2.6 Menghitung *index terms* dari dokumen dan *query* ( $q, d_j$ ). menggunakan persamaan (6)

$$q, d_j = \sum_{i=1}^t W_{iq} - W_{ij} \quad (6)$$

Dengan  $W_{ij}$  adalah bobot *term* dalam dokumen,  $W_{iq}$  adalah bobot *query*.

#### 2.7 Pengukuran *Cosine Similarity* menghitung nilai kosinus sudut antara dua *vector* menggunakan persamaan (7)

$$\text{sim}(q, d_j) = \frac{q \cdot d_j}{|q| \cdot |d_j|} \quad (7)$$

Kesamaan antara kueri dan dokumen atau  $\text{Sin}(q, d_j)$  berbanding lurus dengan jumlah bobot kueri ( $q$ ) di atas bobot dokumen ( $d_j$ ), akar kuadrat dari  $q$  ( $|q|$ ) dikalikan dengan kuadrat Berbanding terbalik dengan nilainya. Akar dokumen ( $|d_j|$ ). Perhitungan kemiripan/kesamaan ini menghasilkan bobot dokumen yang mendekati nilai 1 atau lebih tinggi dari nilai yang diperoleh dari perhitungan inner product.

### 3. Hasil dan Pembahasan

#### 3.1 Hasil

Dalam penelitian ini peneliti menggunakan beberapa sampel judul lagu dengan yang sedang populer dikalangan masyarakat untuk dapat digunakan sebagai bahan analisis data. Adapun judul lagu yang digunakan oleh peneliti adalah sebagai berikut:

D1= Dalam Hitungan

D2= Hati Terlatih

D3= Hitungan Cinta

D4= Rindu Dalam Hati

D5= Rindu Yang Terlarang

D6= Yang Terdalam

D7= Yang Terpilih

Jadi jumlah judul lagu ada 7 jika pengguna menggunakan kata kunci rindu (Q), maka:

- Dokumen MP3 yang memiliki bobot tertinggi =.....?
- Dokumen MP3 yang memiliki bobot terendah =.....?
- Urutan dokumen dengan kata kunci rindu (Q) =.....?
- 

**Tabel 3.1 Ilustrasi Perhitungan *Vector Space Model***

Terms	Q	D1	D2	D3	D4	D5	D6	D7
<b>Cinta</b>	0	0	0	1	0	0	0	0
<b>Dalam</b>	0	1	0	0	1	0	0	0
<b>Hati</b>	0	0	1	0	1	0	0	0
<b>Hitungan</b>	0	1	0	1	0	0	0	0
<b>Rindu</b>	1	0	0	0	1	1	0	0
<b>Terdalam</b>	0	0	0	0	0	0	1	0
<b>Terlarang</b>	0	0	0	0	0	1	0	0
<b>Terlatih</b>	0	0	1	0	0	0	0	0
<b>Terpilih</b>	0	0	0	0	0	0	0	1
<b>Yang</b>	0	0	0	0	0	1	1	1

Sumber: Perhitungan

**Tabel 3.2 Ilustrasi Perhitungan Vector Space Model (Lanjutan)**

Terms	N/df <sub>i</sub>	idf <sub>i</sub> = logN/df <sub>i</sub>
<b>Cinta</b>	7/1= 7	0,845
<b>Dalam</b>	7/2= 3,5	0,544
<b>Hati</b>	7/2= 3,5	0,544
<b>Hitungan</b>	7/2= 3,5	0,544
<b>Rindu</b>	7/3= 2,3	0,362
<b>Terdalam</b>	7/1= 7	0,845
<b>Terlarang</b>	7/1= 7	0,845
<b>Terlatih</b>	7/1= 7	0,845
<b>Terpilih</b>	7/1= 7	0,845
<b>Yang</b>	7/3= 2,3	0,362

Sumber: Perhitungan

**Tabel 3.3 Ilustrasi Perhitungan Vector Space Model (Lanjutan)**

Terms	Bobot W <sub>ij</sub> = tf <sub>ij</sub> .logN/df <sub>i</sub>							
	Q	D1	D2	D3	D4	D5	D6	D7
<b>Cinta</b>	0	0	0	0,845	0	0	0	0
<b>Dalam</b>	0	0,544	0	0	0,544	0	0	0
<b>Hati</b>	0	0	0,544	0	0,544	0	0	0
<b>Hitungan</b>	0	0,544	0	0,544	0	0	0	0
<b>Rindu</b>	0,362	0	0	0	0,362	0,362	0	0
<b>Terdalam</b>	0	0	0	0	0	0	0,845	0
<b>Terlarang</b>	0	0	0	0	0	0,845	0	0
<b>Terlatih</b>	0	0	0,845	0	0	0	0	0
<b>Terpilih</b>	0	0	0	0	0	0	0	0,845
<b>Yang</b>	0	0	0	0	0	0,362	0,362	0,362
<b>Jumlah</b>	0,362	1,088	1,389	1,389	1,45	1,569	1,207	1,207

Sumber: Perhitungan

Berdasarkan tabel 3.1 maka dapat diketahui bahwa bobot dokumen tertinggi adalah senilai 1,569 yaitu *file* D5 dengan judul lagu Rindu Yang Terlarang, kemudian bobot dokumen terendah adalah 1,088 yaitu *file* D1 dengan judul lagu Dalam Hitungan. Maka secara berurutan *file* yang akan muncul secara berurutan ketika pengguna menggunakan kata kunci rindu adalah sebagai berikut;

1. D5 (Rindu Yang Terlarang)
2. D4 (Rindu Dalam Hati)
3. D3 (Hitungan Cinta) dan D2 (Hati Terlatih)
4. D6 (Yang Terdalam) dan D7 (Yang Terpilih)
5. D1 (Dalam Hitungan)

### 3.2 Pembahasan

Menurut Anna dan Hendini (2018), skema desain untuk pengambilan data lagu saat menggunakan sistem ruang vektor menggambarkan bagaimana alur pemrosesan dari sistem pengambilan data lagu dibangun. Prosesnya dimulai dengan pengunjung memasukkan judul lagu atau query ke dalam sistem. Query kemudian dijalankan melalui beberapa proses seperti case folding, data cleaning, tokenization, filtering, stemming, indexing, dll, sehingga dapat dihasilkan bobot atau ranking. Judul lagu terkait.

Pengguna menerima informasi berupa kumpulan lagu terkait dan dapat melihat bobot setiap lagu yang direkomendasikan oleh sistem model ruang vektor. Dari hasil penyelidikan kami, kami dapat melihat bahwa itu cukup untuk sampai pada tahap bobot dokumen. Melalui pembobotan dokumen tersebut diketahui bahwa urutan *file* judul lagu yang akan muncul dari 7 sampel yang digunakan oleh peneliti dengan menggunakan kata kunci rindu secara berurutan adalah: D5 (Rindu Yang Terlarang);

D4 (Rindu Dalam Hati); D3 (Hitungan Cinta) dan D2 (Hati Terlatih); D6 (Yang Terdalam) dan D7 (Yang Terpilih); D1 (Dalam Hitungan).

Berdasarkan hasil penelitian ini, kita dapat menyimpulkan bahwa penerapan model ruang vektor dalam sistem pencarian pemutar MP3 memfasilitasi pencarian musik dan lagu yang diinginkan pengguna. Melalui sistem ini, pengguna dapat melihat nilai dan rekomendasi terbaik dari berbagai file yang tersedia, tergantung dari kata kunci yang telah pengguna berikan ke sistem. Hasil penelitian ini serupa dengan penelitian Anna dan Hendini (2018). Studi mereka menjelaskan bahwa menerapkan model ruang vektor ke sistem pencarian lagu berbasis judul dapat mempercepat proses menemukan lagu terkait. Kemudian lagu yang akan kita cari akan sesuai dengan permintaan si pengunjung tersebut. Dengan sistem ini, dimungkinkan untuk melihat nilai bobot dari setiap lagu yang sesuai, yang dipilih dalam urutan tertinggi atau sedang populer. Salman dan lainnya. (2020) juga menemukan bahwa mesin pencari perpustakaan menggunakan metode model ruang vektor dengan kemampuan pencarian dokumen memberikan informasi yang lengkap tentang suatu dokumen atau buku di perpustakaan. Selain itu, aplikasi diuji dengan Recall and Precision yang memberikan nilai 1 yang menunjukkan bahwa sistem dapat menemukan dokumen yang relevan sesuai dengan dokumen yang relevan dalam koleksi.

#### 4. Kesimpulan

Berdasarkan hasil penelitian ini dapat disimpulkan bahwa penerapan *vector space model* terhadap sistem pencarian MP3 *player* dapat mempermudah pencarian musisi atau lagu sesuai dengan keinginan dari pengguna. Melalui sistem ini pengguna dapat melihat nilai, dan juga rekomendasi terbaik dari berbagai *file* yang tersedia sesuai dengan kata kunci yang dimasukkan pengguna ke dalam sistem. Sistem *vector space model* sangat direkomendasikan untuk digunakan dalam *search engine* (mesin pencari) suatu dokumen, karena kemampuannya telah terbukti untuk menemukan dokumen yang relevan berdasarkan dokumen yang telah dikoleksi.

#### Referensi

- [1] Anna, A. Hendini, "Implementasi Vector Space Model Pada Sistem Pencarian Mesin Karaoke", *Jurnal Evolusi*, Volume 6, No. 1, ISSN: 2338 – 8161, (2018), 1-6.
- [2] D. Noviani, R. Pratiwi, S. Silvianadewi, M.B. Alexandri, M.A. Hakim, "Pengaruh Streaming Musik Terhadap Industri Musik Di Indonesia", *Jurnal Bisnis Strategi*, Vol. 29, No. 1, P-ISSN : 1410-1246, E-ISSN : 2580-1171, (2020), 14 – 25
- [3] G.W. Dewatara, S.A. Agustin, "Pemasaran Musik Pada Era Digital Digitalisasi Industri Musik Dalam Industri 4.0 Di Indonesia", *Wacana*, Volume 18, No. 1, pISSN:1412-7873, eISSN: 2598-7402, (2019), 1- 10.
- [4] M. S. Putra, N.P.A. Widiari, I. W. Gunaya, "Implementasi Generalized Vector Space Model (GVSM) dalam Pencarian Buku di Perpustakaan", *Merpati*, Vol. 7, No. 1, ISSN: 2252-3006, (2019), 86- 94.
- [5] R. Prey, "Nothing personal :algorithmic individuation on music streaming platforms", *Media, CultureandSociety*, 40(7), (2018), 1086–1100.
- [6] S. Salmon, D. Paseru, V. Kumenap, "Implementasi Metode Vector Space Model Pada Search Engine Perpustakaan," *Prosiding Seminar Nasional Sistem Informasi dan Teknologi (SISFOTEK) ke 4*, (2020), 84- 92.
- [7] T.P. Sari, S.D. Nasution, R.K. Hondro, "Penerapan Algoritma Levenstein Pada Aplikasi Kompresi File MP3", *KOMIK (Konferensi Nasional Teknologi Informasi Dan Komputer)*, Volume 2, Nomor 1, eISSN 2597-4645, pISSN 2597-4610, (2018), 334- 342.

*This page is intentionally left blank*