

Analisis Pemodelan & Rekomendasi Berdasarkan Topik Ulasan Aplikasi *Access by KAI* Menggunakan *Latent Dirichlet Allocation* & Metode *How Might We*

Amanda Febrianti¹, Intan Purnamasari², Iqbal Maulana³

^{1,2,3}Program Studi Informatika, Universitas Singaperbangsa Karawang
Jl. H.S. Ronggowaluyo Telukjambe Timur Karawang 41361

Email: ¹ 2010631170039@student.unsika.ac.id, ² intan.purnamasari@staff.unsika.ac.id,
³ iqbal.maulana@staff.unsika.ac.id

Abstrak. PT Kereta Api Indonesia (KAI) merilis aplikasi pemesanan tiket bernama *Access by KAI* yang telah diunduh lebih dari 10 juta kali dan telah mendapat ulasan sebanyak lebih dari 187.000 kali di *Google Playstore*. Namun, dengan banyaknya data ulasan yang masuk untuk aplikasi *Access by KAI*, masih sulit untuk memahami aspek yang perlu ditingkatkan. Dalam hal ini diperlukan *topic modelling* yang berperan untuk mengklasifikasikan ulasan. Tujuan penelitian ini adalah untuk menerapkan metode *Latent Dirichlet Allocation* (LDA) pada pemodelan topik ulasan pengguna aplikasi *Access by KAI* di *Google Playstore* serta menyajikan rekomendasi yang diperoleh dari hasil kamus data atau *bag-of-word* melalui *fishbone diagram*. Penelitian ini menggunakan metodologi *lifecycle of data mining* yang terdiri dari tahap *problem definition, selecting text data mining approach, data collecting, text standardization, text processing, feature extraction, analysis, dan discovery*. Hasil penelitian ini diperoleh jumlah topik sebanyak 7 dengan coherence score sebesar 0,40279302. Kesimpulan dari setiap topik yaitu topik ke-1 membahas pembaruan aplikasi, versi yang tersedia, tampilan, dan keterkaitan dengan stasiun dan kota. Topik ke-2 pengguna mengeluhkan penurunan performa aplikasi setelah pembaruan. Topik ke-3 tentang penggunaan aplikasi *Access by KAI* untuk memesan tiket kereta, dengan sorotan pada versi aplikasi, pengalaman pengguna, dan kualitas aplikasi yang bervariasi dari baik hingga ribet. Topik ke-4 pengguna melaporkan kesulitan akses, terutama dalam masalah login dan pembayaran setelah update aplikasi. Topik ke-5 mengenai kesulitan login, performa aplikasi yang lambat, dan masalah dalam proses pemesanan tiket dan pembayaran yang menjadi sorotan utama. Topik ke-6 pengguna menyampaikan kekecewaan terhadap penurunan performa dalam hal kecepatan dan kesulitan login setelah update. Topik ke-7 mengenai keluhan pengguna terkait kesulitan membeli tiket kereta melalui aplikasi KAI setelah pembaruan atau *upgrade*. Batasan penelitian ini hanya menggunakan data ulasan aplikasi *Access by KAI* di *Google Play Store* dari bulan Juli sampai November 2023 dengan menggunakan metode *Latent Dirichlet Allocation* tanpa dibandingkan dengan metode lain.

Kata kunci: *aplikasi access by kai, topic modelling, Latent Dirichlet Allocation (lda)*

1 Pendahuluan

Transportasi *online* merupakan transportasi yang memanfaatkan aplikasi sebagai media pemesanan untuk memudahkan konsumen dalam memenuhi kebutuhan transportasi [1]. Kemajuan teknologi pada perangkat mobile saat ini telah berkembang dengan cepat, sehingga dapat mendukung eksistensi teknologi transportasi *online*. Berdasarkan data statistik [2] pengguna perangkat Android hingga akhir 2024 mencapai 88,35% dari total pengguna Android di Indonesia.

PT Kereta Api Indonesia (KAI) adalah perusahaan Badan Usaha Milik Negara (BUMN) yang bergerak di bidang transportasi kereta api di Indonesia. PT KAI memiliki berbagai layanan, termasuk aplikasi pemesanan tiket bernama *Access by KAI*. *Access by KAI* adalah aplikasi resmi dari PT Kereta Api Indonesia yang memudahkan pemesanan tiket kereta api, mulai dari tiket kereta antar kota, lokal, LRT Jabodebek, KCI, Bandara, hingga Kereta Cepat, semua dapat diakses melalui satu aplikasi [3]. Aplikasi ini memungkinkan pengguna untuk melakukan pemesanan tiket kereta secara *online* serta menerima informasi terkait kereta api yang dapat diakses dari *Google Playstore* dan *App Store* [4].

Walaupun Aplikasi *Access by KAI* menyediakan banyak kenyamanan, tetapi kemudahan yang ditawarkan tidak selalu menghasilkan kepuasan pengguna. Terkadang, suatu sistem juga memiliki kelemahan, sehingga diperlukan perbaikan-perbaikan pada aplikasi *Access by KAI*. Aplikasi *Access by KAI* telah diunduh lebih dari 10 juta kali dan telah mendapat ulasan sebanyak lebih dari 187.000 kali di *Google Playstore* [5]. Dari ulasan tersebut dapat diambil informasi mengenai kualitas aplikasi yang telah digunakan [6].

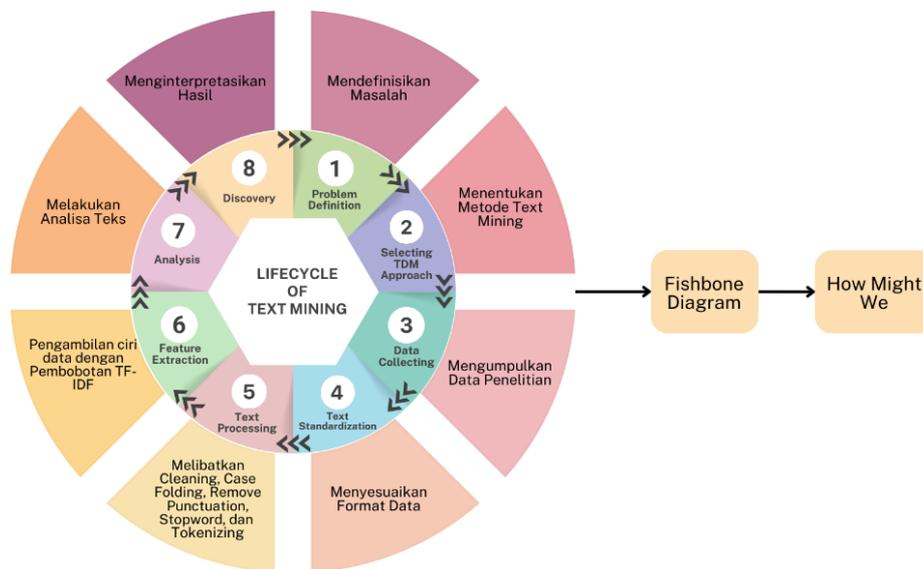
Melalui ulasan yang dituliskan oleh pengguna aplikasi, diharapkan ada *feedback* dari pengguna melalui ulasan yang mereka tulis pada aplikasi. Namun, dengan banyaknya data ulasan yang masuk untuk aplikasi *Access by KAI*, masih sulit untuk memahami aspek yang perlu ditingkatkan. Dalam hal ini diperlukan *topic modelling* [7] yang berperan untuk mengklasifikasikan ulasan, karena manajemen kategorisasi manual data ulasan memerlukan sumber daya keuangan dan manusia yang signifikan. Selain itu, ada sebanyak 50% pengguna internet mengandalkan rekomendasi *word-of-mouth* (opini) dari pengguna lain sebelum menggunakan produk tertentu, yang membentuk opini publik tentang aplikasi tersebut. Ini karena ulasan pengguna dapat menawarkan informasi terkini tentang suatu produk berdasarkan sudut pandang pengguna lain yang telah menggunakannya [8].

Berdasarkan permasalahan tersebut, maka penelitian ini berfokus pada analisis ulasan pengguna aplikasi *Access by KAI* pada *Google Playstore* dengan menggunakan metode *Latent Dirichlet Allocation* (LDA). Metode LDA

mengekstrak topik dari data melalui proses pemrosesan, penghubungan, dan klusterisasi data dengan tujuan menciptakan daftar topik yang memberikan bobot pada setiap dokumen [9]. Analisis ini bertujuan untuk menemukan kesimpulan yang merangkum informasi tersembunyi, seperti topik-topik yang sering diperbincangkan. Dari pengklasifikasian tersebut dapat dijadikan bahan referensi untuk evaluasi dalam kemajuan layanan aplikasi, memberi kemudahan bagi developer dalam memahami kendala pengguna, serta dapat memperbaiki citra perusahaan agar dapat bersaing di pasar dan menarik lebih banyak lagi pengguna.

2 Metode Penelitian

Metodologi yang digunakan pada penelitian ini yaitu *lifecycle of text mining*. Metode ini yang berperan dalam mendukung proses penemuan data. Data mining membantu dalam merencanakan ide dan mengimplementasikannya hingga tahap akhir [10]. Tahapan penelitian yang dilakukan pada proses penelitian ini dapat dilihat pada Gambar 1.



Gambar 1. Alur Penelitian

2.1 Problem Definition

Problem definition merupakan proses identifikasi yang dilakukan dengan tujuan menetapkan permasalahan sebagai dasar perumusan dari penelitian text mining. Proses ini melibatkan penguraian masalah secara terperinci untuk menemukan pendekatan yang sesuai dalam *text mining*. Agar mempermudah penemuan solusi

yang sesuai dengan masalah yang telah ditemukan, identifikasi masalah perlu dilakukan secara rinci.

2.2 *Selecting Text Data Mining Approach*

Selecting text mining approach merupakan proses yang dilakukan setelah *problem definition* yang tujuannya untuk mendapatkan metode terbaik yang nantinya digunakan pada penelitian *text mining*. Pada tahap ini penulis akan menentukan metode terbaik yang digunakan untuk penelitian *text mining* setelah ditetapkan masalah pada tahap *problem definition*.

2.3 *Data Collecting*

Pada penelitian ini data yang diperoleh sebagai data input yaitu ulasan atau komentar aplikasi *Access by KAI* di Google Play Store. Dimana proses pengambilan data ini dinamakan dengan teknik *scraping*. Pada pengumpulan data dilakukan teknik *web scraping* dengan menggunakan bahasa pemrograman python untuk pengambilannya. Periode pengambilan data dari bulan Juli sampai dengan November 2023.

2.4 *Text Standardization*

Text standardization merupakan tahap penyesuaian format data yang digunakan dengan bertujuan untuk mempermudah proses atau tahap *text mining* selanjutnya. Dokumen yang dikumpulkan dalam *data collection* bisa diperoleh dalam berbagai bentuk, oleh sebab itu pada tahap ini semua data diubah menjadi format yang sesuai dengan standar format data *text mining* yang telah ditetapkan.

2.5 *Text Processing*

Pada tahap *text processing*, penulis akan mengolah data input untuk mempersiapkan teks menjadi data yang lebih terstruktur sebelum diolah lebih lanjut. *Preprocessing text* dilakukan dengan tujuan untuk standarisasi dan memudahkan pembacaan dalam proses *text mining* berikutnya.

2.6 *Feature Extraction*

Feature extraction merupakan proses pengambilan informasi penting dari data yang diproses menggunakan pembobotan TF-IDF. Hasil proses pembobotan ini merupakan proses pembobotan data teks dengan perhitungan TF-IDF yang diperoleh dengan cara mengubah data teks menjadi nilai numerik.

2.7 *Analysis*

Pada tahap *analysis*, penulis melakukan *topic modelling* menggunakan metode LDA. *Analysis* dilakukan dengan menentukan hasil pemodelan melalui visualisasi dari grafik *coherence score*. Dimana jika *coherence score* topik yang tinggi maka model yang dihasilkan tersebut yang baik. Setelah mendapatkan jumlah topik dari hasil grafik *coherence score*, kemudian akan diperoleh model LDA berdasarkan jumlah topiknya, dan model LDA tersebut ditampilkan dalam bentuk *wordcloud*.

2.8 *Discovery*

Pada tahap *discovery*, penulis akan membuat kesimpulan dari pola atau pengetahuan (*knowledge*) yang ditemukan dari keseluruhan teks sebagai hasil interpretasi atau evaluasi. Lalu, pada tahap ini hasil yang mana merupakan tahapan terakhir dari metode *Life Cycle of Text Mining*, penulis akan melakukan analisis terhadap output-output berdasarkan pengujian yang telah dilakukan.

3 Hasil dan Pembahasan

3.1 Hasil Penelitian

Hasil penelitian ini bertujuan untuk mendapatkan pemahaman tentang topik-topik apa saja yang sering diperbincangkan ulasan pengguna aplikasi *Access by KAI* pada *Google Playstore*. Dalam penelitian ini, digunakan metode analisis pemodelan topik dengan menggunakan *Latent Dirichlet Allocation* (LDA).

3.1.1 *Problem Definition*

Penulis menguraikan masalah berdasarkan rumusan masalah yaitu untuk menerapkan metode LDA pada pemodelan topik ulasan pengguna aplikasi *Access by KAI* di *Google Playstore*, menyajikan rekomendasi dari hasil kamus data/*bag-of-word* melalui fishbone diagram, serta menyusun ide-ide solusi dari hasil fishbone diagram menggunakan *How Might We*.

3.1.2 *Selecting Text Data Mining Approach*

Pada *selecting text data mining approach*, penulis menentukan pendekatan data mining yang digunakan pada penelitian ini mendapat referensi dari hasil penelitian sebelumnya. Penelitian sebelumnya sama-sama menggunakan metode LDA. Adapun masalah pada penelitian ini adalah menentukan *Topic Modeling* dengan LDA dalam satu studi kasus tentang ulasan pengguna aplikasi. Dimana dalam penelitian sebelumnya, hasil kesimpulannya menunjukkan bahwa model

yang diperoleh melalui metode LDA dapat dengan jelas diinterpretasikan oleh manusia.

3.1.3 Data Collecting

Pengambilan data dilakukan melalui halaman web <https://play.google.com/store/apps/details?id=com.kai.kaiticketing>. Dimana pengambilan data ulasan aplikasi *Access by KAI* ini dilakukan dengan cara *web scraping* dengan menggunakan google colab dengan bahasa pemrograman Python. Data yang berhasil dikumpulkan sebanyak 6150 ulasan.

3.1.4 Text Standardization

Langkah selanjutnya adalah penyesuaian format yang digunakan ini bertujuan untuk mempermudah tahapan text mining selanjutnya. Format data yang diperoleh yaitu berupa file dan diekstraksi menjadi format CSV. Informasi yang terdapat di data ini terdiri dari username pengguna, rating (peringkat), waktu (date), dan ulasan yang pengguna tulis.

3.1.5 Text Processing

Pada tahap *text processing* terdapat beberapa langkah yang dilakukan agar *dataset* berisi ulasan pengguna yang digunakan untuk *topic modelling* memudahkan dalam pemodelan. Tahap ini terdiri dari *cleaning*, *case folding*, *remove punctuation*, *stopword*, dan *tokenizing*. Berikut ini hasil tahapan *text processing* yang dapat dilihat pada Tabel 1.

Tabel 1. Tahap *Text Processing*

Tahapan	Sebelum	Sesudah
<i>Cleaning</i>	Terdapat kolom <i>userName</i> , <i>score</i> , <i>at</i> , dan <i>Content</i> dari 6150 data.	Hanya tersisa kolom <i>Content</i> dan 5526 data
<i>Case Folding</i>	Kenapa tiket harga 74ribu pasar Senen => Kutoarjo selalu penuh meski beda tanggal dan jauh" hari	kenapa tiket harga 74ribu pasar senen => kutoarjo selalu penuh meski beda tanggal dan jauh" hari
<i>Remove Punctuation</i>	kenapa tiket harga 74ribu pasar senen => kutoarjo selalu penuh meski beda tanggal dan jauh" hari	kenapa tiket harga ribu pasar senen kutoarjo selalu penuh meski beda tanggal dan jauh hari
<i>Stopword</i>	kenapa tiket harga ribu pasar senen kutoarjo selalu penuh meski beda tanggal dan jauh hari	kenapa tiket harga ribu pasar senen kutoarjo selalu penuh meski beda tanggal jauh hari
<i>Tokenizing</i>	kenapa tiket harga ribu pasar senen kutoarjo selalu penuh meski beda tanggal jauh hari	'kenapa', 'tiket', 'harga', 'ribu', 'pasar', 'senen', 'kutoarjo', 'selalu', 'penuh', 'meski', 'beda', 'tanggal', 'jauh', 'hari'

3.1.6 Feature Extraction

Pada tahap ini digunakan pembobotan TF-IDF untuk mengonversi data teks menjadi data numerik membantu menghasilkan data yang lebih spesifik, sehingga data yang tidak relevan dapat disaring dan tidak digunakan dalam proses selanjutnya. Hasil pembobotan TF-IDF dapat dilihat pada Tabel 2 berikut.

Tabel 2. Hasil TF-IDF

Dokumen	aaa	aaccses	aapaann	...	pemerintah	pemesanan	...
1	0	0	0	...	0	0	...
2	0	0	0	...	0	0	...
3	0	0	0	...	0	0	...
4	0	0	0	...	0	4,8436352776813	...
5	0	0	0	...	0	0	...

3.1.7 Analysis

Pada tahap analysis bertujuan untuk meyakinkan hasil dari pemodelan topik menggunakan *Latent Dirichlet Allocation (LDA)* sesuai dengan yang telah penulis konsepskan sebelumnya. Untuk mengetahui hasil pemodelan, dapat digunakan *coherence score*. Dalam eksperimen pemodelan topik, penulis menentukan jumlah topik dengan skor mulai dari 1 hingga 10. Hasil eksperimen tersebut menunjukkan peningkatan dan penurunan dalam *coherence score*, yang merupakan ukuran yang digunakan untuk mengevaluasi *topic modelling*, di mana skor yang tinggi menunjukkan bahwa topik tersebut menghasilkan model yang baik. Hasil *coherence score* dapat dilihat pada Tabel 3.

Tabel 3. *Coherence Score*

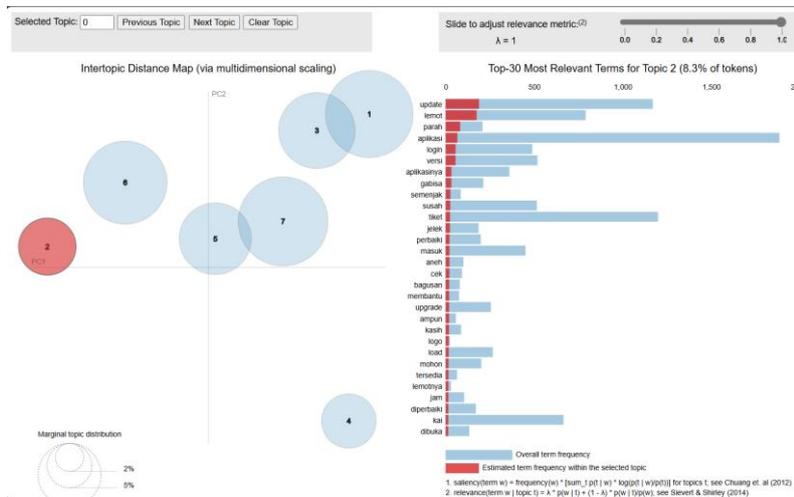
Number of topics	Coherence score
1	0,397273327
2	0,380767444
3	0,382743782
4	0,377366957
5	0,380010514
6	0,396092109
7	0,40279302
8	0,378164195
9	0,388621418

Setelah menghitung jumlah topiknya dari *coherence score*, model LDA akan dihasilkan berdasarkan banyak topiknya yaitu 7, dengan jumlah kata yang ditampilkan dalam model yaitu 10 kata dan masing-masing memiliki bobot.

Tabel 4. Hasil Pemodelan Topik LDA

Model LDA Topik 1	
Topic: 0	Words: 0.038*"aplikasi" + 0.031*"kai" + 0.031*"tiket" + 0.024*"stasiun" + 0.024*"update" + 0.020*"versi" + 0.016*"tampilan" + 0.012*"kota" + 0.010*"kereta" + 0.010*"access"
Model LDA Topik 2	
Topic: 1	Words: 0.046*"update" + 0.043*"lemot" + 0.020*"parah" + 0.016*"aplikasi" + 0.014*"login" + 0.014*"versi" + 0.008*"aplikasinya" + 0.008*"gabisa" + 0.007*"semenjak" + 0.007*"susah"
Model LDA Topik 3	
Topic: 2	Words: 0.050*"aplikasi" + 0.026*"tiket" + 0.018*"kereta" + 0.015*"versi" + 0.011*"stasiun" + 0.010*"pesan" + 0.010*"bagus" + 0.009*"masuk" + 0.009*"ribet" + 0.008*"jadwal"
Model LDA Topik 4	
Topic: 3	Words: 0.022*"susah" + 0.020*"log" + 0.017*"tiket" + 0.017*"upgrade" + 0.016*"aplikasi" + 0.011*"update" + 0.010*"gabisa" + 0.010*"sulit" + 0.010*"pembayaran" + 0.009*"akun"
Model LDA Topik 5	
Topic: 4	Words: 0.034*"aplikasi" + 0.030*"login" + 0.025*"tiket" + 0.020*"update" + 0.013*"aplikasinya" + 0.013*"load" + 0.012*"pembayaran" + 0.010*"susah" + 0.010*"kursi" + 0.009*"pesan"
Model LDA Topik 6	
Topic: 5	Words: 0.062*"aplikasi" + 0.046*"lemot" + 0.029*"update" + 0.026*"masuk" + 0.016*"susah" + 0.016*"bagus" + 0.016*"aplikasinya" + 0.013*"login" + 0.012*"tiket" + 0.011*"bgt"
Model LDA Topik 7	
Topic: 6	Words: 0.037*"tiket" + 0.028*"aplikasi" + 0.028*"update" + 0.022*"kai" + 0.015*"ribet" + 0.012*"susah" + 0.012*"beli" + 0.011*"pesen" + 0.011*"kereta" + 0.010*"upgrade"

Setelah model LDA diperoleh, model tersebut dapat divisualisasikan menggunakan PyLDAvis untuk melihat keterkaitan antar kata yang dihasilkan.



Gambar 2. Visualisasi PyLDAvis

Pada Gambar 2 menunjukkan visualisasi pyldavis jika salah satu topik dipilih, dalam hal ini topik 2 yang dipilih sehingga lingkaran akan berubah warna menjadi merah. Selanjutnya, *bar chart* di sisi kanan panel akan berubah warna merah, menunjukkan estimasi *term frequency* topik yang dipilih. Pada panel sebelah kiri, terdapat pemetaan jarak antar topik (*intertopic distance map*) melalui *multidimensional scaling*, yang juga menampilkan *cluster* topik dalam bentuk lingkaran dengan nomor tertentu pada setiap cluster. Sementara itu, pada panel sebelah kanan, terdapat 30 terminologi yang paling relevan untuk topik tertentu. Selanjutnya model LDA seluruh topik tersebut dapat ditampilkan melalui bentuk wordcloud yang dapat dilihat pada Gambar 3 berikut.



Gambar 3. Visualisasi Wordcloud

3.1.8 Discovery

Dalam tahap *discovery*, penulis melakukan analisis terhadap hasil pemodelan topik ulasan aplikasi *Access by KAI* menggunakan *Latent Dirichlet Allocation* (LDA). Berdasarkan *coherence score* tertinggi, diperoleh 7 topik utama. *Output* penelitian ini diperoleh dari aplikasi yang dibangun oleh penulis dengan

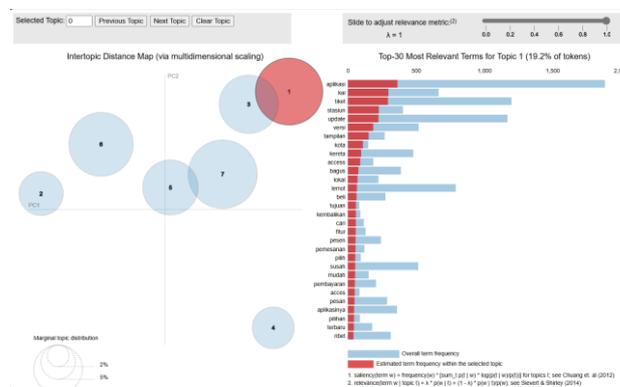
menggunakan bahasa pemrograman Python. Selanjutnya, penulis menginterpretasikan data *output* yang telah diperoleh. Berikut adalah penjelasan analisis *output* dari penelitian ini.

3.1.8.1 Model LDA Topik ke-1

Pada topik ke-1 didapatkan model LDA yaitu sebagai berikut:

Tabel 5. Model LDA Topik ke-1
Model LDA Topik 1

Topic: 0
 Words: 0.038*"aplikasi" + 0.031*"kai" + 0.031*"tiket" + 0.024*"stasiun" + 0.024*"update" + 0.020*"versi" + 0.016*"tampilan" + 0.012*"kota" + 0.010*"kereta" + 0.010*"access"



Gambar 4. Visualisasi PyLDAvis Topik ke-1

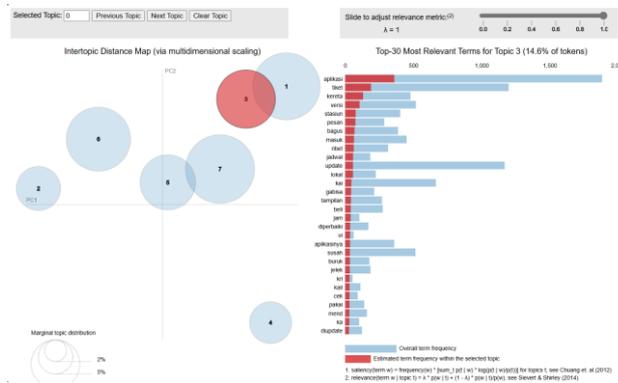
Berdasarkan Tabel 5 dan Gambar 4 menunjukkan hasil yang ditampilkan 10 kata kunci pada topik ke-1. Hasil pada topik ke-1 membahas mengenai aplikasi *Access by KAI* yang digunakan untuk pemesanan tiket kereta. Fokus utamanya meliputi pembaruan (*update*) aplikasi, versi aplikasi yang tersedia, dan tampilan aplikasi. Pengguna juga membahas tentang stasiun-stasiun dan kota-kota yang terkait dengan penggunaan aplikasi ini, serta kemudahan akses ke aplikasi tersebut.

3.1.8.2 Model LDA Topik ke-2

Pada topik ke-2 didapatkan model LDA yaitu sebagai berikut:

Tabel 6. Model LDA Topik ke-2
Model LDA Topik 2

Topic: 1
 Words: 0.046*"update" + 0.043*"lemot" + 0.020*"parah" + 0.016*"aplikasi" + 0.014*"login" + 0.014*"versi" + 0.008*"aplikasinya" + 0.008*"gabisa" + 0.007*"semakin" + 0.007*"susah"



Gambar 6. Visualisasi PyLDAvis Topik ke-3

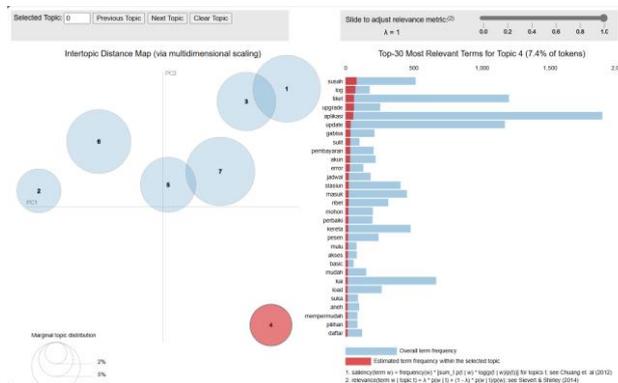
Berdasarkan Tabel 7 dan Gambar 6 menunjukkan hasil yang ditampilkan 10 kata kunci pada topik ke-3. Hasil pada topik ke-3 membahas mengenai penggunaan aplikasi *Access by KAI*. Pengguna membahas tentang versi aplikasi yang tersedia dan pengalaman mereka dalam menggunakan aplikasi tersebut untuk memesan tiket kereta. Ada perhatian terhadap kemudahan atau kerumitan dalam proses pemesanan dan masuk ke aplikasi, serta komentar tentang kualitas aplikasi, baik yang positif (bagus) maupun negatif (ribet). Selain itu, topik ini juga menyentuh informasi tentang jadwal kereta dan stasiun yang relevan bagi pengguna aplikasi.

3.1.8.4 Model LDA Topik ke-4

Pada topik ke-4 didapatkan model LDA yaitu sebagai berikut:

Tabel 8. Model LDA Topik ke-4

Model LDA Topik 4
Topic: 3
Words: 0.022*"susah" + 0.020*"log" + 0.017*"tiket" + 0.017*"upgrade" + 0.016*"aplikasi" + 0.011*"update" + 0.010*"gabisa" + 0.010*"sulit" + 0.010*"pembayaran" + 0.009*"akun"



Gambar 7. Visualisasi PyLDAvis Topik ke-4

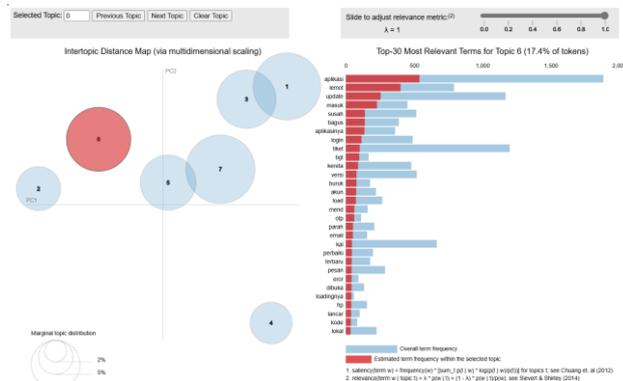
Berdasarkan Tabel 8 dan Gambar 7 menunjukkan hasil yang ditampilkan 10 kata kunci pada topik ke-4. Sama halnya seperti pembahasan topik ke-2, hasil pada topik ke-4 membahas mengenai kesulitan yang dialami pengguna *Access by KAI*, terutama setelah dilakukan *update*. Sedangkan pembahasan pada topik ke-4 ini pengguna melaporkan kesulitan dalam melakukan log masuk (log) dan proses pembayaran, serta masalah-masalah terkait akun mereka. Ada juga keluhan bahwa aplikasi menjadi sulit digunakan dan tidak berfungsi dengan baik (gabisa) setelah pembaruan.

3.1.8.5 Model LDA Topik ke-5

Pada topik ke-5 didapatkan model LDA yaitu sebagai berikut:

Tabel 9. Model LDA Topik ke-5

Model LDA Topik 5
Topic: 4
Words: 0.034*"aplikasi" + 0.030*"login" + 0.025*"tiket" + 0.020*"update" + 0.013*"aplikasinya" + 0.013*"load" + 0.012*"pembayaran" + 0.010*"susah" + 0.010*"kursi" + 0.009*"pesan"



Gambar 9. Visualisasi PyLDAvis Topik ke-6

Berdasarkan Tabel 10 dan Gambar 9 menunjukkan hasil yang ditampilkan 10 kata kunci pada topik ke-6. Sama halnya seperti pembahasan topik ke-2, ke-4 dan ke-5 hasil pada topik ke-6 membahas mengenai keluhan pengguna terhadap performa aplikasi tiket kereta yang menjadi lemot setelah dilakukan update. Pengguna mengalami kesulitan untuk masuk (login) ke aplikasi dan mengeluhkan bahwa aplikasi menjadi sangat lambat dan susah digunakan. Meskipun ada yang menyebut aplikasi ini bagus, masalah utama tetap terfokus pada penurunan performa setelah pembaruan.

3.1.8.7 Model LDA Topik ke-7

Pada topik ke-7 didapatkan model LDA yaitu sebagai berikut:

Tabel 11. Model LDA Topik ke-7

Model LDA Topik 7
Topic: 6
Words: 0.037*"tiket" + 0.028*"aplikasi" + 0.028*"update" + 0.022*"kai" + 0.015*"ribet" + 0.012*"susah" + 0.012*"beli" + 0.011*"pesen" + 0.011*"kereta" + 0.010*"upgrade"

Faktor dan permasalahan yang ada pada fishbone diagram tersebut selanjutnya dibagi menjadi poin-poin berdasarkan aspek permasalahan yang menjadi kendala pada aplikasi *Access by KAI*. Poin-poin permasalahan yang ditemukan tersebut selanjutnya disusun dengan teknik “How Might We” untuk mendefinisikan masalah. Kemudian dilakukan *brainstorming* ide-ide solusi berdasarkan teknik “How Might We” untuk menemukan solusi yang tepat untuk masalah yang telah diketahui. Hasil “How Might We” dapat dilihat pada Tabel 12 berikut.

Tabel 12. *How Might We*

No	Aspek Permasalahan	Ide-Ide Solusi
1.	Login	<ul style="list-style-type: none"> a. Mengimplementasikan proses single sign-on untuk memungkinkan pengguna masuk ke akun mereka dengan satu kali autentikasi. b. Menyediakan pilihan untuk login melalui identitas digital, seperti pengenalan wajah atau sidik jari. c. Menggunakan teknologi otentikasi dua faktor (2FA) untuk meningkatkan keamanan tanpa mengorbankan kenyamanan.
2.	Pemesanan Tiket	<ul style="list-style-type: none"> a. Merancang antarmuka pengguna yang lebih intuitif dengan panduan langkah demi langkah yang jelas dalam proses pemesanan tiket. b. Menyediakan opsi pencarian yang lebih canggih, seperti filter khusus dan rekomendasi berdasarkan preferensi pengguna. c. Menerapkan fitur pembayaran satu kali klik untuk mempercepat proses pembelian tiket.
3.	Kinerja Aplikasi	<ul style="list-style-type: none"> a. Melakukan optimisasi kode dan kompresi gambar untuk mengurangi beban aplikasi dan mempercepat waktu respon. b. Menggunakan teknologi caching untuk mempercepat pengambilan data dari server dan meningkatkan responsifitas aplikasi. c. Mengembangkan mode offline yang memungkinkan pengguna untuk tetap menggunakan aplikasi bahkan tanpa koneksi internet.
4.	Proses Update	<ul style="list-style-type: none"> a. Menerapkan sistem pembaruan otomatis yang tidak mengganggu pengguna dan dilakukan di latar belakang. b. Menyediakan riwayat perubahan yang jelas dan panduan langkah demi langkah untuk membantu pengguna menavigasi perubahan setelah pembaruan. c. Mengintegrasikan mekanisme umpan balik pengguna yang memungkinkan pengguna memberikan masukan tentang pembaruan.
5.	Performa Aplikasi Setelah Update	<ul style="list-style-type: none"> a. Menerapkan tes kinerja menyeluruh sebelum merilis pembaruan untuk memastikan tidak ada penurunan kinerja. b. Menyediakan mekanisme pengguliran kembali ke versi sebelumnya jika pengguna mengalami masalah performa setelah pembaruan. c. Memperbarui aplikasi secara bertahap untuk memungkinkan pemantauan performa sebelum dan setelah implementasi.
6.	Waktu Load Aplikasi	<ul style="list-style-type: none"> a. Mengoptimalkan ukuran aplikasi dan asetnya untuk mempercepat waktu load. b. Menggunakan teknologi prefetching untuk memuat data yang diperlukan sebelum pengguna meminta akses ke fitur tertentu.

7.	Kompabilitas dan Manajemen Data	<ul style="list-style-type: none"> c. Memanfaatkan teknologi penyimpanan lokal untuk menyimpan data yang sering diakses dan mengurangi waktu load. a. Melakukan uji kompatibilitas menyeluruh dengan perangkat dan sistem operasi terbaru sebelum merilis pembaruan. b. Menerapkan strategi pengelolaan data yang efisien, termasuk pembaruan terjadwal dan penggunaan teknologi enkripsi data.
8.	Versi Aplikasi	<ul style="list-style-type: none"> a. Membangun tim dedicated untuk mengidentifikasi, memperbaiki, dan merilis patch untuk masalah yang ada pada versi aplikasi saat ini secara berkala. b. Menyediakan saluran komunikasi yang jelas bagi pengguna untuk melaporkan masalah dan memperoleh dukungan teknis. c. Menerapkan sistem umpan balik pengguna yang terstruktur untuk mengumpulkan masukan tentang masalah yang dialami pengguna.
9.	Pembayaran Tiket	<ul style="list-style-type: none"> a. Menyediakan opsi pembayaran yang beragam, termasuk kartu kredit, dompet digital, dan pembayaran melalui platform pihak ketiga. b. Menggunakan teknologi enkripsi dan deteksi penipuan untuk menjaga keamanan transaksi pembayaran. c. Menyediakan pengingat pembayaran dan konfirmasi instan setelah pembayaran berhasil.
10.	Koneksi Internet	<ul style="list-style-type: none"> a. Mengoptimalkan aplikasi untuk bekerja dalam mode offline dengan menyimpan data lokal dan menyediakan fungsi penting yang dapat diakses tanpa koneksi internet. b. Menggunakan teknologi load balancing dan caching untuk mengurangi dampak koneksi internet yang tidak stabil terhadap kinerja aplikasi. c. Memberikan saran dan panduan kepada pengguna untuk memperbaiki masalah koneksi internet saat mereka mengalami masalah.
11.	Masalah Server	<ul style="list-style-type: none"> a. Mengimplementasikan sistem pemantauan server yang terus-menerus untuk mendeteksi dan mengatasi masalah sebelum memengaruhi pengguna. b. Memiliki rencana pemulihan bencana yang solid untuk mengurangi downtime dan mempercepat pemulihan setelah terjadi masalah server. c. Menggunakan teknologi replikasi server untuk meningkatkan redundansi dan mengurangi risiko kegagalan server.
12.	Aksesibilitas Aplikasi	<ul style="list-style-type: none"> a. Melakukan penelitian pasar yang mendalam untuk memahami kebutuhan pengguna dengan berbagai kondisi lingkungan. b. Mengembangkan fitur aksesibilitas seperti pilihan kontras tinggi, dukungan layar sentuh, dan pengaturan font yang dapat disesuaikan. c. Bermitra dengan organisasi dan kelompok advokasi disabilitas untuk mendapatkan masukan dan memvalidasi solusi aksesibilitas yang diusulkan.

4 Kesimpulan

Hasil pemodelan topik dengan *Latent Dirichlet Allocation* (LDA) menggunakan data dari 6150 ulasan aplikasi *Access by KAI* menunjukkan adanya 7 topik dengan *coherence score* sebesar 0,40279302. Topik ke-1 membahas pembaruan aplikasi,

versi, tampilan, dan keterkaitan dengan stasiun dan kota. Topik ke-2 pengguna mengeluhkan penurunan performa aplikasi setelah pembaruan. Topik ke-3 penggunaan aplikasi untuk memesan tiket kereta dengan sorotan pada versi, pengalaman pengguna, dan kualitas aplikasi. Topik ke-4 pengguna kesulitan akses pada login dan pembayaran setelah update. Topik ke-5 mengenai kesulitan login, performa aplikasi yang lambat, dan masalah dalam proses pemesanan tiket dan pembayaran. Topik ke-6 kekecewaan pengguna terhadap penurunan performa dalam hal kecepatan dan kesulitan login setelah update. Topic ke-7 mengenai keluhan pengguna terkait kesulitan membeli tiket kereta melalui aplikasi setelah pembaruan atau upgrade.

Hasil visualisasi fishbone diagram menggambarkan secara jelas permasalahan berdasarkan topik-topik yang sering dibahas oleh pengguna dari hasil pemodelan topik dengan LDA dengan pembagian faktor berdasarkan *Man, Methods, Machine, Materials*, dan *Environment*.

Dengan menggunakan teknik *How Might We*, peneliti merumuskan berbagai pertanyaan yang bertujuan untuk menemukan solusi atas masalah yang diidentifikasi. Ide-ide solusi yang dihasilkan dari brainstorming meliputi aspek yang terdiri dari login, pemesanan tiket, kinerja aplikasi, proses update, performa aplikasi setelah update, waktu load aplikasi, komparabilitas dan manajemen data, versi aplikasi, pembayaran tiket, koneksi internet, masalah server, dan aksesibilitas aplikasi.

5 Referensi

- [1] R. Sari, "Pengaruh Adanya Transportasi Online Terhadap Kesejahteraan Driver di Kota Bandar Lampung Dalam Perspektif Ekonomi Islam (Studi Pada Komunitas Driver Go-Jek di Kota Bandar Lampung)," Skripsi. Universitas Islam Negeri Raden Intan Lampung, 2020.
- [2] StatCounter, "Mobile Operating System Market Share Indonesia." [Online]. Tersedia: <https://gs.statcounter.com/os-market-share/mobile/indonesia>. [Diakses 5 Desember 2023]
- [3] "Sejarah Perkeretaapian," PT KERETA API INDONESIA (PERSERO). [Online]. Tersedia: <https://www.kai.id/>. [Diakses 8 Desember 2023]
- [4] N. B. Sidauruk and N. Riza, "Sentimen Analisis Data Pengguna Terhadap KAI AccessSystematic Literature Review," *J. Mhs. Tek. Inform.*, vol. 7, no. 2, pp. 1297–1303, 2023.
- [5] PT. Kereta Api Indonesia, "Access by KAI." [Online]. Tersedia: <https://play.google.com/store/apps/details?id=com.kai.kaiticketing>. [Diakses: 21 Desember 2023]
- [6] A. E. Sari, S. Widowati, and K. M. Lhaksana, "Klasifikasi Ulasan Pengguna Aplikasi Mandiri Online di Google Play Store dengan Menggunakan Metode Information Gain dan Naive Bayes Classifier," *e-Proceeding Eng.*, vol. 6, no. 2, pp. 9143–9157, 2019.

- [7] M. Y. Hendrawan and N. W. K. Projo, "Topic Modelling in Knowledge Management Documents BPS Statistics Indonesia," *Proc. Int. Conf. Data Sci. Off. Stat.*, vol. 2021, no. 1, pp. 119–130, Jan. 2021, doi: 10.34123/ICDSOS.V2021I1.52.
- [8] S. A. Aaputra, Didi Rosiyadi, Windu Gata, and Syepri Maulana Husain, "Sentiment Analysis Analysis of E-Wallet Sentiments on Google Play Using the Naive Bayes Algorithm Based on Particle Swarm Optimization," *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 3, no. 3, pp. 377–382, 2019, doi: 10.29207/resti.v3i3.1118.
- [9] M. Luvian chisni chilmi, "*Latent Dirichlet Allocation (LDA) untuk Mengetahui Topik Pembicaraan Warganet Twitter tentang Omnibus Law*," Skripsi. *Univ. Islam Negeri Syarif Hidayatullah*, 2021.
- [10] A. Shaikh, N. A. Mahoto, and A. Unar, "Jamshoro under the CC by 4.0 International License," *901 Mehran Univ. Res. J. Eng. Technol.*, vol. 38, no. 4, pp. 2413–7219, 2019, doi: 10.3316/INFORMIT.729766023678328.