IMPLEMENTASI *LATENT DIRICHLET ALLOCATION* (LDA) DALAM PEMODELAN TOPIK ULASAN PENGGUNA BSI *MOBILE* DI *GOOGLE PLAY STORE*

TUGAS AKHIR

Diajukan Sebagai Salah Satu Syarat Untuk Memperoleh Gelar Sarjana Program Studi Statistika

Disusun Oleh:

Rika Yulianti

19611094

PROGRAM STUDI STATISTIKA

FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM

UNIVERSITAS ISLAM INDONESIA

YOGYAKARTA

2023

HALAMAN PERSETUJUAN PEMBIMBING

TUGAS AKHIR

Judul	:	Implementasi *Latent Dirichlet Allocation* (LDA) *dalam* Pemodelan Topik Ulasan Pengguna BSI Mobile di *Google Play Store*
Nama Mahasiswa	:	Rika Yulianti
NIM	:	19611094

TUGAS AKHIR INI TELAH DIPERIKSA DAN DISETUJUI UNTUK DIUJIKAN

Yogyakarta, 9 Mei 2023

Pembimbing

(Dr. Edy Widodo, S.Si., M.Si.)

HALAMAN PENGESAHAN

TUGAS AKHIR

IMPLEMENTASI *LATENT DIRICHLET ALLOCATION* (LDA) DALAM PEMODELAN TOPIK ULASAN PENGGUNA BSI *MOBILE* DI *GOOGLE PLAY STORE*

Nama Mahasiswa : Rika Yulianti

NIM : 19611094

TUGAS AKHIR INI TELAH DIUJIKAN

PADA TANGGAL : 26 Mei 2023

Nama Penguji	**Tanda Tangan**
1. Ayundyah Kesumawati, S.Si., M.Si.	.....
2. Dina Tri Utari, S.Si., M.Sc.	.....
3. Dr. Edy Widodo, S.Si., M.Si.	.....

Mengetahui,

Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam

(Prof. Riyanto, S.Pd., M.Si., Ph.D.)

KATA PENGANTAR

بِسْمِ اللَّهِ الرَّحْمَٰنِ الرَّحِيمِ

Assalamu'alaikum Wr.Wb

Alhamdulillahirobil'alamin, puji syukur penulis ucapkan kepada Allah SWT yang telah memberikan rahmat serta hidayah-Nya sehingga penulis dapat menyelesaikan Tugas Akhir yang berjudul "Implementasi *Latent Dirichlet Allocation* (LDA) pada Pemodelan Topik Ulasan Pengguna BSI *Mobile* di *Google Play Store*". Tugas Akhir ini menjadi salah satu syarat yang wajib terpenuhi dalam penyelesaian studi S1 Statistika Universitas Islam Indonesia.

Penulis sadar bahwa penyusunan Tugas Akhir ini banyak mendapat dukungan dari berbagai pihak. Maka dari itu, penulis menyampaikan terima kasih kepada:

1. Bapak Prof. Riyanto, S.Pd., M.Si., Ph.D. selaku Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Islam Indonesia.
2. Bapak Dr. Edy Widodo, S.Si., M.Si. selaku Ketua Jurusan Statistika Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Islam Indonesia, sekaligus dosen pembimbing tugas akhir yang telah membimbing dari awal hingga penyelesaian penyusunan tugas akhir.
3. Ibu Dr. Atina Ahdika, S.Si., M.Si. selaku Ketua Program Studi Statistika Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Islam Indonesia.
4. Dosen pendamping dari Direktorat Pembinaan Kemahasiswaan (DPK) UII yang telah mendampingi mahasiswa penerima beasiswa agar fokus dalam penyelesaian studi di Universitas Islam Indonesia.
5. Ibu Wahyuniati selaku Ibu penulis dan Bapak Dwi Heru Purnomo selaku Ayah penulis yang telah memberikan dukungan moril maupun materi, serta Almarhum Bapak Risanta selaku Ayah kandung penulis yang menjadi penyemangat penulis untuk menyelesaikan studi S1 Statistika di Universitas Islam Indonesia.
6. Aditya Dwi Aprianta dan Adelia Putri Purnama Sari selaku adik penulis yang telah memberikan dukungan moril.

7. Keluarga besar penulis yang telah memberikan dukungan moril.
8. Seluruh teman-teman Statistika UII angkatan 2019, khususnya Sekar Salma Putri dan Salma Fitria Dewi selaku teman seperjuangan penulis yang selalu memberikan dukungan moril dan mau menjadi tempat keluh kesah penulis selama mengerjakan Tugas Akhir ini.
9. Salma Fitria Dewi, Diana Ayu Nur Halimah, Yesi Indriani, dan Iin Fadila Ramadhani yang memberikan tempat singgah selama bimbingan Tugas Akhir.

Penulis menyadari bahwa penulisan Tugas Akhir ini masih terdapat banyak kekurangan karena keterbatasan pengetahuan dan kemampuan yang dimiliki oleh penulis. Maka dari itu, penulis mengharapkan adanya kritik dan saran yang bersifat membangun agar kedepannya karya tulis lain dapat menjadi lebih baik.

Wassalamualaikum Wr.Wb

Yogyakarta, 9 Mei 2023

Rika Yulianti

DAFTAR ISI

DAFTAR TABEL

DAFTAR GAMBAR

DAFTAR LAMPIRAN

PERNYATAAN

Dengan ini saya menyatakan bahwa dalam Tugas Akhir ini tidak terdapat karya karya yang sebelumnya pernah diajukan untuk memperoleh gelar kesarjanaan di suatu Perguruan Tinggi dan sepanjang pengetahuan saya juga tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan orang lain, kecuali yang diacu dalam naskah ini dan disebutkan dalam daftar pustaka.

Yogyakarta, 9 Mei 2023

Rika Yulianti

INTISARI

IMPLEMENTASI *LATENT DIRICHLET ALLOCATION* (LDA) DALAM PEMODELAN TOPIK ULASAN PENGGUNA BSI *MOBILE* DI *GOOGLE PLAY STORE*

Rika Yulianti
Program Studi Statistika, Fakultas Matematika dan Ilmu Pengetahuan Alam
Universitas Islam Indonesia

Penyelenggaraan KTT G20 di Indonesia mengangkat topik tentang transformasi ekonomi dan digital yang bertujuan untuk mempercepat pemulihan ekonomi global secara inklusif. BSI Mobile merupakan salah satu bentuk upaya pengembangan digitalisasi ekonomi di Indonesia. Bank Syariah mempunyai ciri khas dalam pengelolaannya karena didasarkan pada prinsip syariah islam. Penelitian ini bertujuan untuk mengetahui apa saja topik yang dibahas dalam ulasan pengguna BSI *Mobile* di *Google Play Store*. Data diambil dengan cara *scrapping* dari tanggal 11 Maret 2022 hingga 21 November 2022 dan diperoleh data sebanyak 20.000 ulasan. Setelah dilakukan *preprocessing* tersisa sebanyak 17.757 ulasan yang digunakan untuk analisis LDA. Data dianalisis menggunakan metode *Latent Dirichlet Allocation* (LDA). LDA merupakan salah satu metode pemodelan topik yang banyak digunakan karena dapat digunakan untuk data berjumlah besar, metode ini didasarkan pada konsep probabilitas untuk menemukan kemiripan suatu dokumen dan mengelompokkannya ke dalam beberapa topik. Hasil penelitian ini menunjukkan bahwa terbentuk 7 topik dengan *coherence score* 0,4350406246. Terdapat beberapa topik yang menunjukkan kesulitan pengguna saat aktivasi.

Kata Kunci : pemodelan topik, LDA, ulasan, BSI *Mobile*, *Google Play Store*.

ABSTRACT

LATENT DIRICHLET ALLOCATION (LDA) IMPLEMENTATION IN BSI MOBILE USER REVIEWS TOPIC MODELING ON GOOGLE PLAY STORE

Rika Yulianti
Department of Statistics, Faculty of Matematics and Natural Sciences
Universitas Islam Indonesia

The holding of the G20 Summit in Indonesia raised the topic of economic and digital transformation aimed at inclusively accelerating global economic recovery. BSI Mobile is a form of effort to develop the digitalization of the economy in Indonesia. Islamic banks have distinctive characteristics in their management because they are based on Islamic Sharia principles. This study aims to find out what topics are discussed in BSI Mobile user reviews on the Google Play Store. Data was taken by scrapping from March 11, 2022, to November 21, 2022, and obtained data from 20,000 reviews. After preprocessing, 17,757 reviews were used for LDA analysis. Data were analyzed using the Latent Dirichlet Allocation (LDA) method. LDA is a widely used topic modeling method because it can be used for large amounts of data, it is based on the concept of visibility to find the structure of a document and group it into several topics. The results of this study indicate that 7 topics are formed with a coherence score of 0,4350406246. Several topics indicate user difficulties during activation.

Keywords: *topic modeling, LDA, review, BSI Mobile, Google Play Store.*

BAB I
PENDAHULUAN

1.1. Latar Belakang Masalah

Diselenggarakannya KTT G20 di Indonesia pertama kali selama bergabung pada forum G20 sejak 1999 menjadi momentum baik bagi Indonesia (Mutiarasari, 2022). G20 merupakan forum utama kerja sama ekonomi internasional yang beranggotakan negara-negara dengan perekonomian besar didunia (Indah, 2022). Masuknya Indonesia dalam forum tersebut karena dinilai mempunyai ukuran dan potensi ekonomi sangat besar di Kawasan Asia (Mutiarasari, 2022). Salah satu topik utama yang akan diangkat dalam presidensi G20 Indonesia adalah transformasi ekonomi dan digital yang bertujuan untuk mempercepat pemulihan ekonomi global secara inklusif (Kementerian Koordinator Bidang Perekonomian Republik Indonesia, 2021).

Saat ini, salah satu bentuk upaya digitalisasi ekonomi di Indonesia adalah pengembangan *mobile banking* atau *m-banking* (Saputro, 2022). Salah satu *m-banking* di Indonesia adalah BSI *Mobile* yang dimiliki oleh Bank Syariah Indonesia. Bank Syariah Indonesia (BSI) adalah bank hasil *merger* antara Bank Mandiri Syariah, BNI Syariah, dan BRI Syariah pada tahun 2021 (Lestari & Keumala, 2022). Bank Syariah mempunyai ciri khas sendiri dalam pengelolaannya jika dibandingkan dengan bank konvensional yang mana pengelolaannya disesuaikan dengan prinsip syariah islam (Hasanah, Fitriani, & Hana, 2022). BSI *Mobile* sendiri mempunyai fitur yang memudahkan nasabah dalam meningkatkan nilai spiritual diri melalui fitur berbagi berupa penyaluran zakat, infak, dan wakaf secara mudah dan efisien karena dapat dilakukan dimanapun dan kapanpun (Febrianti dkk, 2021).

Hasil penelitian dari (Widyaningsih, S, & R, 2022) menyatakan bahwa persepsi kegunaan, kemudahan, dan informasi pengetahuan mempengaruhi keputusan penggunaan BSI *Mobile*. *Google Play Store* mempunyai halaman untuk menampung ulasan atau *reviews* dan *ratings* dari berbagai aplikasi, salah satunya adalah BSI *Mobile* (Google Play, 2023). Ulasan tersebut berisi persepsi atau pandangan pengguna BSI *Mobile* terhadap BSI *Mobile* itu sendiri yang kurang lebih dapat menggambarkan kegunaan, kemudahan, dan kekurangan dari BSI *Mobile*

(Widyaningsih, S, & R, 2022). Ulasan tersebut juga dapat dilihat oleh publik bukan hanya pengguna aplikasi BSI *Mobile* saja.

Pada *Google Play Store* telah terdapat sebanyak 129.000 ulasan terkait BSI *Mobile* (Google Play, 2023). Ulasan tersebut dapat dilakukan pengelompokkan apa saja topik yang dibahas didalamnya sehingga dapat membantu memberikan masukan kepada *developer* BSI *Mobile* untuk evaluasi dan perbaikan aplikasi kedepannya agar lebih baik, disamping dapat memberikan informasi kepada publik terkait topik yang paling banyak dibicarakan pada ulasan pengguna BSI *Mobile* di *Google Play Store*. Pengelompokan topik tersebut dapat dilakukan salah satunya menggunakan analisis pemodelan topik. Metode pemodelan topik yang sering digunakan adalah *Latent Dirichlet Allocation* (LDA) (Yaman, Sartono, & M. Soleh, 2021). Pemodelan topik menggunakan LDA didasarkan pada konsep probabilitas untuk mencari kemiripan suatu dokumen dan mengelompokkannya menjadi beberapa topik atau kelompok, metode tersebut masuk kedalam *unsupervised learning* karena data yang digunakan tidak ada label atau target (Fernanda, 2021). LDA juga mampu melakukan pemodelan topik pada data teks yang berjumlah sangat besar (Yaman, Sartono, & M. Soleh, 2021). Berdasarkan penjelasan tersebut, peneliti melakukan penelitian terkait implementasi LDA dalam pemodelan topik ulasan pengguna BSI *Mobile* di *Google Play Store*.

1.2. Rumusan Masalah

Bagaimana hasil pemodelan topik ulasan pengguna BSI *Mobile* di *Google Play Store*?

1.3. Batasan Masalah

1. Data yang digunakan merupakan data sampel berupa ulasan pengguna BSI *Mobile* di *Google Play Store* sebanyak 20.000 ulasan sejak tanggal 11 Maret 2022 hingga 21 November 2022.
2. Pengambilan data dilakukan dengan cara *scrapping* menggunakan *Google Colab*.
3. Pengolahan data menggunakan analisis pemodelan topik dengan metode LDA yang dilakukan di *Google Collab* dan *Microsoft Excel*.

1.4. Tujuan Penelitian

Mengetahui hasil pemodelan topik ulasan pengguna BSI *Mobile* di *Google Play Store* menggunakan LDA.

1.5. Manfaat Penelitian

Berdasarkan hasil pemodelan topik ulasan pengguna BSI *Mobile* di *Google Play Store* menggunakan LDA dapat digunakan oleh *developer* BSI *Mobile* sebagai bahan evaluasi dan perbaikan aplikasi BSI *Mobile* agar lebih baik terutama pada bagian aktivasi, serta dapat memberikan informasi kepada publik terkait topik yang paling banyak dibicarakan pada ulasan pengguna BSI *Mobile* di *Google Play Store* adalah terkait kesulitan saat aktivasi yang dibahas dalam 3 topik dari 7 topik yang terbentuk.

BAB II
TINJAUAN PUSTAKA

Beberapa penelitian terdahulu dapat digunakan sebagai acuan dalam penelitian ini yang terkait pemodelan topik menggunakan *Latent Dirichlet Allocation* (LDA) dan BSI *Mobile*. Hal tersebut penting diketahui untuk mencari pembaruan dari penelitian terdahulu dengan penelitian ini yang dapat dilihat dari perbedaan dan persamaannya.

Penelitian dari (Mahmud, 2023) terkait analisis sentimen aplikasi BSI *Mobile* pada ulasan *google play*. Variabel yang digunakan adalah ulasan pengguna BSI *Mobile* di *google play*. Metode yang digunakan adalah Naïve Bayes. Hasil dari penelitian ini, diketahuinya bahwa terdapat sebesar 57,3% ulasan positif dan 42,7% ulasan negatif dengan total ulasan sebanyak 4527 ulasan serta akurasi sebesar 98%.

Pemodelan topik digunakan oleh (Reni & Vanomy, 2023) untuk melakukan penelitian terkait *the role of e-wallet's actual consumer in Indonesia with policy perspective and consumer perception*. Variabel yang digunakan adalah artikel terkait persepsi konsumen terhadap *e-wallet* dari website resmi berita online dan populer di Indonesia. Metode yang digunakan adalah LDA. Hasil dari penelitian ini, diketahuinya bahwa konsumen *e-wallet* yang sebenarnya di Indonesia sangat dipengaruhi oleh Bank Indonesia dan kebijakan Otoritas Jasa Keuangan mengenai *e-wallet*. Selain itu, persepsi konsumen terhadap *e-wallet* memiliki peran penting dalam kepentingan aktual konsumen untuk menggunakan *e-wallet*.

Penelitian terkait aplikasi MyPertamina dilakukan oleh (Oktafiandi, 2023) menggunakan pemodelan topik. Variabel yang digunakan adalah *tweet* dengan tagar #mypertamina. Metode yang digunakan adalah LDA. Hasil penelitian diperoleh 7 topik dengan nilai *perplexity* -8,08844 dan *topic coherence* 0.49860. Kata yang berpengaruh pada pembentukan tidak mempunyai korelasi dengan aplikasi MyPertamina seperti war, game, gamenya, dan hero sehingga dibutuhkan cara penyaringan yang lebih baik untuk mendapatkan data yang bersih.

Analisis ulasan aplikasi MyPertamina dilakukan juga oleh (Abdurrazzaq, 2023) menggunakan pemodelan topik. Variabel yang digunakan adalah ulasan.

Metode yang digunakan adalah LDA. Hasil penelitian diperoleh 7 topik dengan nilai *perplexity* -8,08844 dan *topic coherence* 0.49860. Kata yang berpengaruh pada pembentukan tidak mempunyai korelasi dengan aplikasi MyPertamina seperti war, game, gamenya, dan hero sehingga dibutuhkan cara penyaringan yang lebih baik untuk mendapatkan data yang bersih.

Penelitian dari (Astuti & Cahyono, 2023) terkait analisis *topic modelling* persepsi pengguna internet. Variabel yang digunakan adalah *tweet* akun detik.com. Metode yang digunakan adalah LDA. Hasil penelitian diperoleh 3 topik. Topik 1 terkait peristiwa alam atau bencana alam di Indonesia. Topik 2 terkait bahasan tokoh atau figur politik, isu serta peristiwa pemerintahan di Indonesia. Topik 3 terkait perlombaan atau *world cup*, kejuaraan piala dunia sepak bola yang diselenggarakan di Qatar pada tahun 2022.

Berita detikcom diteliti oleh (Matira, Junaidi, & Setiawan, 2023) menggunakan pemodelan topik. Variabel yang digunakan adalah judul berita *online*. Metode yang dipakai adalah LDA. Hasil penelitian ini diperoleh 3 topik, topik 1 membahas terkait konflik dan krisis negara, topik 2 membahas terkait dengan krisis kemanusiaan, dan topik 3 membahas terkait isu korupsi oleh pejabat negara.

Penelitian terkait BSI *Mobile* dilakukan oleh (Nugroho & M.Pudjihardjo, 2022) tentang pengaruh persepsi kemudahan, persepsi kegunaan, ketersediaan fitur dan literasi keuangan terhadap preferensi konsumen menggunakan BSI *Mobile*. Variabel yang digunakan adalah persepsi kemudahan, persepsi kegunaan, ketersediaan fitur dan literasi keuangan, dan preferensi penggunaan BSI *Mobile*. Hasil penelitian ini adalah persepsi kemudahan, ketersediaan fitur, dan literasi keuangan berpengaruh positif terhadap preferensi penggunaan BSI *Mobile*. Sedangkan persepsi kegunaan tidak berpengaruh terhadap preferensi penggunaan BSI *Mobile*.

Pengaruh promosi dan kualitas layanan terhadap keputusan nasabah menggunakan layanan BSI *Mobile* diteliti oleh (Devy & Fikriyah, 2022) di Bank Syariah Indonesia KC Surabaya Diponegoro. Variabel yang digunakan adalah promosi, kualitas layanan, dan keputusan nasabah dalam menggunakan layanan. Metode yang dipakai adalah Regresi Linier Berganda. Hasil penelitian ini

menunjukkan bahwa semakin baik promosi dan kualitas layanan yang ditawarkan kepada nasabah, maka nasabah akan semakin banyak menggunakan layanan BSI *Mobile*.

Penelitian terkait pengamatan tren ulasan hotel dilakukan oleh (Suparyati, Utami, & Fathurahman, 2022) pada Tripadvisor. Variabel yang digunakan adalah ulasan hotel pada Tripadvisor. Metode yang dipakai adalah LDA. Hasil penelitian ini menunjukkan bahwa tren ulasan lebih banyak membahas mengenai *location*, *service*, *hotel*, *breakfast*, *resort* dan *beach* namun terdapat beberapa sebaran kata yang kurang sesuai dengan bahasan topik (Suparyati, Utami, & Fathurahman, 2022).

Pemodelan topik diteliti juga oleh (Kannitha, Mustafid, & Kartikasari, 2022) tentang pemodelan topik pada keluhan pelanggan dalam media sosial twitter. Variabel yang digunakan adalah *tweets* dengan kata kunci "@FirstMediaCares" dan "@IndiHomeCare". Metode yang dipakai adalah LDA dengan estimasi *gibbs sampling*. Hasil penelitian ini terbentuk 10 topik dengan menggunakan nilai *loglikelihood* pada kata kunci "@FirstMediaCares" kemudian pada kata kunci "@IndiHomeCare" terbentuk 11 topik. Topik yang sering dikeluhkan pada First Media adalah internet yang mati ketika mereka sedang bekerja kemudian pada IndiHome adalah internet yang suka putus dan mati. Tingkat kesuaian topik dengan keluhan pelanggan melalui *tweets*, pada First Media sebesar 70%, sedangkan pada IndiHome sebesar 81,8%.

Penelitian dari (Nasution, Widodo, & Adhi, 2021) terkait sistem deteksi topik politik pada twitter. Variabel yang digunakan adalah *tweet hastag* #pemilu2019, #pilpres2019, dan #pileg2019. Metode yang dipakai adalah LDA. Hasil dari penelitian ini adalah jumlah topik 10 untuk pengujian 100, 1000, dan 6000 data dihasilkan rata-rata 90% benar untuk deteksi topik LDA, nilai tersebut juga masih dapat berubah tergantung seberapa bagus *input* ataupun optimalisasi yang dilakukan pada model sehingga LDA dirasa dapat digunakan untuk mendeteksi topik pada twitter dengan topik politik.

Pemodelan topik berita diteliti oleh (Nugraha & Mungaran, 2021) pada portal berita *online* berbahasa Indonesia. Variabel yang digunakan adalah teks artikel pada portal berita detikcom dengan kanal berita detikNews. Metode yang digunakan

adalah LDA. Hasil penelitian ini membentuk 5 topik dari total sebanyak 68.537 artikel dengan nilai koheren sebesar 0.67.

Penelitian yang dilakukan oleh (Prakerti dkk, 2020) tentang perilaku siswa menggunakan media pembelajaran daring. Variabel yang digunakan adalah *caption* yang di *posting* pada Instagram. Metode yang dipakai adalah LDA. Hasil penelitian ini diperoleh 4 topik. Topik 0 dengan 508 *Post* dan kata yang menjadi *highlight* sekolahdaring, dirumahaja, sekolahonline. Topik 1 dengan 156 *post* dan kata yang menjadi *highlight* tugas, sekolah, metabolism. Topik 2 dengan 150 *post* dan kata yang menjadi *highlight*tugasclassroombiologi, biologi, mipa. Topik 3 dengan 115 *post* dan kata yang menjadi *highlight* ipa, belajardirumah, sekolah daring.

Pembuatan kata kunci otomatis dalam artikel diteliti oleh (Wirasakti dkk, 2020) menggunakan pemodelan topik. Variabel yang digunakan adalah artikel *website/blog*. Metode yang dipakai adalah LDA dan *K-Means*. Hasil penelitian ini diperoleh 4 topik dengan kata yang mempunyai nilai probabilitas tertinggi adalah kata mesin, maksimal, varian, cx-8, mobil, dan mazda. Kata tersebut dapat digunakan sebagai rekomendasi dalam pembuatan kata kunci.

Penelitian dari (Alfanzar, Khalid, & Rozas, 2020) membahas terkait *topic modelling* skripsi. Variabel yang digunakan adalah *abstract* pada penelitian skripsi yang dilakukan oleh Program Studi Sastra Inggris UINSA. Metode yang dipakai adalah LDA. Hasil yang didapatkan dari 584 *abstract* skripsi terbentuk 3 topik. Kata-kata pada setiap topik sesuai dengan pembagian topik menurut konsentrasi Program Studi Sastra Inggris UINSA.

Tabel 2.1 Penelitian Terdahulu.

No	Penulis	Metode	Persamaan	Perbedaan
1	Muhammad Bahaudin Mahmud (2023)	*Naïve Bayes*	Data yang digunakan adalah ulasan pengguna BSI *Mobile* di *Google Play Store*	• Metode yang digunakan adalah *Latent Dirichlet Allocation* (LDA)
2	Reni dan Afrianti Elsye Vanomy (2023)	• *Latent Dirichlet Allocation* • *Content Analysis Method* • *Triangulation Method*	• Penggunaan metode *Latent Dirichlet Allocation* • Terdapat visualisasi *intertopic distance map*	• Data yang digunakan mengenai ulasan pengguna BSI *Mobile* di *Google Play Store* • Penentuan jumlah topik menggunakan *coherence score* • Terdapat visualisasi *wordcloud*

3	Hery Oktafiandi (2023)	*Latent Dirichlet Allocation*	• Penggunaan metode *Latent Dirichlet Allocation* • Penentuan jumlah topik menggunakan *coherence score* • Terdapat visualisasi *intertopic distance map*	• Data yang digunakan mengenai ulasan pengguna BSI *Mobile* di *Google Play Store* • Terdapat visualisasi *wordcloud* dan *intertopic distance map*
4	Muhammad Adrinta Abdurrazzaq (2023)	*Latent Dirichlet Allocation*	Penggunaan metode *Latent Dirichlet Allocation*	• Data yang digunakan mengenai ulasan pengguna BSI *Mobile* di *Google Play Store* • Penentuan jumlah topik tidak menggunakan nilai *perplexity* • Terdapat visualisasi *wordcloud*
5	Angga Reni Dwi Astuti dan Nuri Cahyo (2023)	*Latent Dirichlet Allocation*	• Penggunaan metode *Latent Dirichlet Allocation* • Terdapat visualisasi *intertopic distance map* • Terdapat visualisasi *wordcloud*	Data yang digunakan mengenai ulasan pengguna BSI *Mobile* di *Google Play Store*
6	Yayang Matira, Junaidi, dan Imam Setiawan (2023)	*Latent Dirichlet Allocation*	Penggunaan metode *Latent Dirichlet Allocation*	Data yang digunakan mengenai ulasan pengguna BSI *Mobile* di *Google Play Store*
7	Irvan Yusuf Nugroho dan M.Pudjihardjo (2022)	Regresi Linier Berganda	Objek penelitian terkait BSI *Mobile*	• Data yang digunakan mengenai ulasan pengguna BSI *Mobile* di *Google Play Store* • Metode yang digunakan adalah *Latent Dirichlet Allocation* (LDA)
8	Nurul Azizah Aunillah Devy dan Khusnul Fikriyah (2022)	Regresi Linier Berganda	Objek penelitian terkait BSI *Mobile*	• Data yang digunakan mengenai ulasan pengguna BSI *Mobile* di *Google Play Store* • Metode yang digunakan adalah *Latent Dirichlet Allocation* (LDA)
9	Suparyati, Emma Utami, dan Agus Fathurahman (2022)	*Latent Dirichlet Allocation*	Penggunaan metode *Latent Dirichlet Allocation*	• Data yang digunakan mengenai ulasan pengguna BSI *Mobile* di *Google Play Store*

				• Penentuan jumlah topik menggunakan *coherence score*
10	Diandra Zakeshia Tiara Kannitha, Mustafid, dan Puspita Kartikasari (2022)	*Latent Dirichlet Allocation*	Penggunaan metode *Latent Dirichlet Allocation*	• Data yang digunakan mengenai ulasan pengguna BSI *Mobile* di *Google Play Store* • Penentuan jumlah topik menggunakan *coherence score* • Terdapat visualisasi *wordcloud* dan *intertopic distance map*
11	Khairul Hudha Nasution, Widodo, dan Bambang Prasetya Adhi (2021)	*Latent Dirichlet Allocation*	Penggunaan metode *Latent Dirichlet Allocation*	• Data yang digunakan mengenai ulasan pengguna BSI *Mobile* di *Google Play Store* • Penentuan jumlah topik menggunakan *coherence score* • Terdapat visualisasi *wordcloud* dan intertopic *distance map*
12	Muhammad Andika Nugraha dan Lulu Chaerani Munggaran (2021)	*Latent Dirichlet Allocation*	• Penggunaan metode *Latent Dirichlet Allocation* • Penggunaan *coherence score* untuk penentuan jumlah topik • Terdapat visualisasi *intertopic distance map*	• Data yang digunakan mengenai ulasan pengguna BSI *Mobile* di *Google Play Store* • Terdapat visualisasi *wordcloud*
13	Aqila Intan Prakerti, Avelyna Ferariya Claresta, Muhammad Rasyid Kafif Ibrahim, dan Nur Aini Rakhmawati (2020)	*Latent Dirichlet Allocation*	Penggunaan metode *Latent Dirichlet Allocation*	• Data yang digunakan mengenai ulasan pengguna BSI *Mobile* di *Google Play Store* • Penentuan jumlah topik menggunakan *coherence score* • Terdapat visualisasi *wordcloud* dan *intertopic distance map*
14	Lucky Adhikrisna Wirasakti, Rony Permadi, Anggit Dwi Hartanto, dan Hartatik (2020)	• *Latent Dirichlet Allocation* • *K-Means*	Penggunaan metode *Latent Dirichlet Allocation*	• Data yang digunakan mengenai ulasan pengguna BSI *Mobile* di *Google Play Store* • Penentuan jumlah topik menggunakan *coherence score* • Terdapat visualisasi *wordcloud* dan *intertopic distance map*

15	Alif Iffan Alfanzar, Khalid, dan Indri Sudanawati Rozas (2020)	*Latent Dirichlet Allocation*	Penggunaan metode *Latent Dirichlet Allocation*	• Data yang digunakan mengenai ulasan pengguna BSI *Mobile* di *Google Play Store* • Penentuan jumlah topik menggunakan *coherence score* • Terdapat visualisasi *wordcloud* dan *intertopic distance map*

Berdasarkan penelitian terdahulu sudah banyak beberapa penelitian terkait pemodelan topik menggunakan metode LDA maupun terkait BSI *Mobile* namun masih terbatas penelitian terkait pemodelan topik ulasan pengguna BSI *Mobile* di *Google Play Store*. Oleh karena itu, pembaruan yang penulis lakukan adalah terkait implementasi LDA dalam pemodelan topik ulasan pengguna BSI *Mobile* di *Google Play Store*.

BAB III
LANDASAN TEORI

3.1. BSI *Mobile*

BSI *Mobile* adalah salah satu aplikasi *mobile banking* atau m-banking yang ada di Indonesia yang dimiliki oleh Bank Syariah Indonesia. BSI mempunyai visi *Top 10 Global Islamic Banking* dengan misi memberikan akses solusi keuangan syariah di Indonesia, menjadi bank besar yang memberikan nilai terbaik bagi para pemegang saham, dan menjadi perusahaan pilihan dan kebanggaan para talenta terbaik Indonesia (Lestari & Keumala, 2022). Pengelolaan bank tersebut disesuaikan dengan syariah Islam yang tidak digunakan oleh bank konvensional. Fitur BSI *Mobile* selain digunakan untuk memudahkan nasabah dalam bertransaksi namun juga menyediakan fasilitas untuk memberikan kemudahan nasabahnya dalam meningkatkan nilai spiritual dan berbagi kepada sesama melalui pembayaran Infak (Daudshah & Yetti, 2022).

3.2. *Text Mining*

Text mining merupakan proses eksplorasi dan analisis data teks tidak terstruktur dalam jumlah besar menggunakan perangkat lunak yang dapat mengidentifikasi konsep, pola, topik, kata kunci, dan atribut lain dalam data (Arni, 2018). Struktur fungsional *text mining* terdiri dari beberapa langkah, yaitu *preprocessing*, *core mining operations*, *presentation layer components and browsing functionality*, dan *refinement techniques* (Feldman & Sanger, 2007).

3.3. *Scrapping*

Scrapping merupakan proses pengambilan dokumen semi-terstruktur dari internet, biasanya dalam bentuk halaman *website*, menggunakan bahasa markup HTML atau XHTML, dan menganalisis dokumen tersebut untuk mengambil informasi spesifik dari halaman tersebut kemudian digunakan kembali di bidang lain (Anggraeni, 2019).

3.4. *Pre-Processing*

Preprocessing merupakan tahapan yang penting namun untuk *preprocessing* data teks masih terbilang cukup sulit karena teks adalah data yang sangat *raw* dan dapat memiliki arti yang berbeda dengan yang dimaksud oleh penulis serta bisa tidak sesuai tata bahasa karena perubahan budaya (Hakim B. , 2021).

3.4.1 *Remove Duplicate*

Remove Duplicate merupakan proses penghapusan data yang sama atau terambil secara berulang ketika melakukan *scrapping* (Putra & Juanita, 2021).

3.4.2 *Remove Emoji*

Emoji adalah simbol yang menggambarkan emosi namun sekarang beberapa penulisan *emoji* tidak lagi diartikan sebagai emosional dari penulis sehingga dapat tidak digunakan (Yerzi & Sibaroni, 2021). Hal tersebut dikarenakan emoji sering dianggap sebagai pelengkap pada komunikasi *nonverbal*, contohnya penggunaan *emoji* 😊 pada kata maupun kalimat yang menunjukkan kondisi marah (Zain & Isam, 2019).

3.4.3 *Case Folding*

Case folding adalah penyelarasan *case* dalam dokumen karena tidak semua teks dalam dokumen menggunakan huruf kapital sehingga tahapan ini digunakan untuk mengubah seluruh teks dalam dokumen menjadi huruf kecil atau *lowercase* sehingga mempermudah pencarian (Alita & Rahman, 2020).

3.4.4 *Spelling Normalization*

Spelling Normalizations merupakan tahap memperbaiki kata yang tidak baku, singkat atau salah eja agar menjadi kata baku sesuai dengan KBBI (Manurung, Matondang, & Prasvita, 2022). Contoh proses ini seperti penggantian kata “siiiaaappppp” menjadi "siap", "waaaahhh" menjadi "wah", "guwee" menjadi "saya", "eloo" menjadi "kamu", "laen" menjadi "lain", "pengen" menjadi "ingin", "knp" menjadi "kenapa", dan seterusnya (Khomsah & Aribowo, 2020).

3.4.5 *Stopwords Removal*

Tahapan ini digunakan untuk menghilangkan kata yang tidak bermakna untuk kalimat yang ditempati sehingga perlu dibuang (Parasati, Bachtiar, & Setiawan, 2020). Contohnya adalah penghapusan kata sambung, kata depan, kata ganti, atau kata yang tidak ada hubunganya dengan analisis yang akan dilakukan (Salam, Zeniarja, & Khasanah, 2018).

3.4.6 *Tokenization*

Tokenizing merupakan proses pemecahan teks menjadi kata yang disebut dengan token (Manurung, Matondang, & Prasvita, 2022). Jika karakter ke-i bukan pemisah kata seperti titik (.), koma (,), spasi, dan pemisah lainnya maka akan digabungkan dengan karakter berikutnya (Salam, Zeniarja, & Khasanah, 2018).

3.4.7 *Stemming*

Stemming adalah proses merubah kata yang memiliki imbuhan menjadi kata dasar (Manurung, Matondang, & Prasvita, 2022). Imbuhan yang dihapus seperti *inflection suffixes* (-lah, -kah, -ku, dll), *derivational suffix* (-i, -kan, -an) dan *derivational prefix* (-be, -di, -me, -pe, -se, dan –te) (Prasastio, Heriyanto, & Kaswidjanti, 2022).

3.5. *Wordcloud*

Wordcloud atau awan kata pada dasarnya adalah visualisasi berupa representasi kata yang yang ukuran setiap kata menyesuaikan dengan frekuensi munculnya kata tersebut sehingga memudahkan pembaca untuk melihat kata-kata yang sering muncul dalam dokumen (Kabir, Ahmed, & Karim, 2020).

3.6. *Term Frequency-Inverse Document Frequency* (TF-IDF)

Metode TF-IDF (*Term Frequency Inverse Document Frequency*) adalah metode pembobotan hubungan kata (*term*) terhadap dokumen dengan cara menggabungkan 2 konsep, yaitu frekuensi kemunculan sebuah kata didalam sebuah dokumen tertentu dan *inverse* frekuensi dokumen yang mengandung kata tersebut (Karmayasa & Mahendra, 2012). Hasil pembobotan kata menggunakan TF-IDF berguna untuk mengetahui seberapa penting kata terhadap dokumen, serta bobot

setiap kata digunakan untuk analisis selanjutnya karena data yang telah melewati tahap *preprocessing* harus berbentuk numerik (Simatupang & Utomo, 2019).

Frekuensi kemunculan (*term frequency*) merupakan indikasi sejauh mana *term* mewakili isi dokumen yang mana jika semakin menonjol *term* yang muncul di dokumen maka semakin tinggi nilai kesesuaiannya (Karmayasa & Mahendra, 2012). Berikut terdapat beberapa formula yang digunakan pada TF (*Term Frequency*) (Sulehu dkk, 2019):

1. TF biner atau *binary* TF, jika suatu kata atau *term* ada pada dokumen maka diberi nilai satu (1) namun jika tidak ada maka diberi nilai nol (0).
2. TF murni atau *raw* TF, nilainya didasarkan pada jumlah kemunculan *term* atau kata dalam dokumen seperti jika muncul sebanyak 5 kali maka kata tersebut bernilai 5.
3. TF logaritmik, digunakan untuk menghindari dominasi dokumen yang berisi sedikit *term* dalam *query*, tetapi dengan frekuensi tinggi. Persamaan TF logaritmik adalah sebagai berikut:

$$TF = 1 + log\ (tf) \tag{3.1}$$

4. TF normalisasi, menggunakan perbandingan antara frekuensi suatu *term* dengan nilai maksimum dari kumpulan frekuensi *term* dalam dokumen.

$$TF = 0.5 + 0.5\ \times \left[\frac{tf}{maxtf}\right] \tag{3.2}$$

Pada penelitian ini, formula TF yang digunakan adalah TF murni atau *raw* TF. IDF (*Inverse Document Frequency*) adalah metode pembobotan yang dipadukan dengan TF yang menghitung banyaknya *term* tertentu dalam kumpulan dokumen (Sulehu dkk, 2019). Semakin sedikit jumlah dokumen yang mengandung *term* maka nilai IDF akan semakin besar (Sulehu dkk, 2019). Persamaan untuk perhitungan IDF adalah sebagai berikut (Sulehu dkk, 2019):

$$IDF_j = log\frac{D}{df_j} \tag{3.3}$$

dengan:

D : jumlah semua dokumen

df_j : jumlah dokumen yang mengandung *term*

Perhitungan IDF pada program *python* memanfaatkan *library sklearn*. Pada

library tersebut tidak menggunakan log namun menggunakan ln atau logaritma natural sehingga persamaan yang digunakan untuk menghitung IDF adalah sebagai berikut (Hendriawan, 2021):

$$IDF_j = ln\frac{D}{df_j} \tag{3.4}$$

Indeks j pada persamaan (3.4) menunjukkan urutan dokumen.

Persamaan untuk TF-IDF merupakan perkalian dari TF dengan IDF seperti berikut (Irmayati, 2018):

$$w_{ij} = tf_{ij} \times idf_j \tag{3.5}$$

$$w_{ij} = tf_{ij} \times ln\left(\frac{D}{df_j}\right) \tag{3.6}$$

dengan:

w_{ij} : bobot *term* t_j terhadap dokumen d_i

tf_{ij} : jumlah kemunculan *term* t_j dalam dokumen d_i

Indeks *j* dan *i* pada w_{ij} dan tf_{ij} dalam persamaan (3.5) dan (3.6) menunjukkan urutan *term* (*j*) dan urutan dokumen (*i*).

Berdasarkan persamaan (3.5) dan (3.6), berapapun nilai tf_{ij}, apabila $D = df_j$ maka hasilnya akan 0 untuk perhitungan IDF (Irmayati, 2018). Maka dari itu ditambahkan nilai 1 pada IDF sehingga persamaannya menjadi seperti berikut (Irmayati, 2018):

$$w_{ij} = tf_{ij} \times (ln\left(\frac{D}{df_j}\right) + 1) \tag{3.7}$$

Contoh perhitungan TF-IDF (Harishamzah, 2020), misal terdapat 3 dokumen:

d1 = saya berangkat ke kampus untuk belajar

d2 = ibu pergi berbelanja ke pasar

d3 = ayah ke kantor untuk bekerja

kemudian akan dihitung bobot dari kata “berangkat” pada dokumen ke-1 atau d1 dengan diketahui bahwa sebanyak 1 dokumen mengandung kata “berangkat”, jumlah dokumen sebanyak 3, dan kata “berangkat” muncul di dokumen ke-1 sebanyak 1 kali. Maka nilai TF-IDF:

$$IDF_j = ln\frac{D}{df_j}$$

$$IDF_1 = ln\frac{3}{1}$$

$$IDF_1 = 1{,}098$$

Kemudian untuk skor TF-IDF sebagai berikut:

$$w_{ij} = tf_{ij} \times (ln\left(\frac{D}{df_j}\right) + 1)$$

$$w = 1 \times (IDF_1 + 1)$$

$$w = 1 \times (1{,}098 + 1)$$

$$w = 2{,}098$$

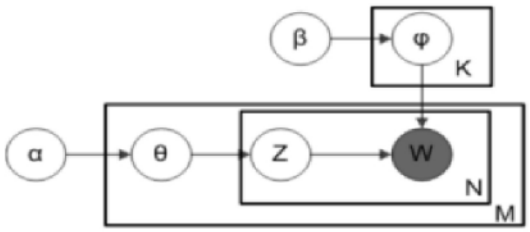
3.7. Pemodelan Topik

Pemodelan topik terdiri dari kata, dokumen, dan korpus. Kata adalah bagian dasar dari data diskrit, dokumen adalah urutan dari kumpulan kata, dan korpus adalah kumpulan dokumen (Blei, Ng, & Jordan, 2003). *Topic Modelling* atau pemodelan topik merupakan teknik yang digunakan dalam identifikasi pola pada korpus yang mempunyai teks berjumlah besar dengan kondisi teks terstruktur maupun tidak terstruktur yang dikelompokkan dengan cara mengelompokkan *term* korpus ke dalam kelompok *term* sehingga menghasilkan topik menggunakan pemrosesan kesamaan (Suparyati, Utami, & Fathurahman, 2022). Teknik ini ampuh untuk melakukan analisis dan pengelompokkan artikel secara efisien dalam skala besar oleh organisasi secara internal (Blad & Svensson, 2020).

Inputan yang digunakan pada pemodelan topik adalah vektor dari hasil pembobotan kata menggunakan TF-IDF (Damayanti, Purwitasari, & Suciati, 2018). Pemodelan topik juga merupakan bentuk pemodelan probabilistik, misalnya *Latent Dirichlet Allocation* (LDA) yang didasarkan pada pemahaman bahwa setiap teks dalam kumpulan teks terlihat seperti *bag of words* yang dihasilkan terhadap campuran topik yang diharapkan oleh penulisnya untuk didiskusikan (Febrianta, Widiyanesti, & Ramadhan, 2021). Selain LDA, ada juga model probabilistik berupa *Latent Semantic Analysis* (LSA) yang didasarkan pada kemiripan makna kata yang muncul dalam teks dan menunjukkan kata-kata dan teks menggunakan *vector space modelling* yang mengumpulkan data tekstual menjadi matriks istilah per dokumen dengan memperlihatkan frekuensi berbobot dari setiap istilah untuk merepresentasikan dokumen dalam kumpulan istilah (Ignatow & Mihalcea, 2018).

3.8. *Latent Dirichlet Allocation* (LDA)

LDA adalah model bayesian hierarki tiga tingkat dengan setiap *item* koleksi dimodelkan sebagai campuran terbatas pada kumpulan topik yang mendasarinya dan setiap topik dimodelkan sebagai campuran tak terbatas atas rangkaian probabilitas topik yang mendasarinya (Blei, Ng, & Jordan, 2003). LDA termasuk salah satu metode pemodelan topik yang paling banyak digunakan dibandingkan metode pemodelan topik lainnya (Vayansky & Kumar, 2020). Masalah *overfitting* pada *Probabilistic Latent Semantic Analysis* (PLSA) dapat diatasi menggunakan metode ini (Fernanda, 2021). LDA masuk kedalam *unsupervised learning* karena data yang digunakan tidak mempunyai label atau target (Fernanda, 2021). Metode ini didasarkan pada konsep probabilitas untuk menemukan kemiripan suatu dokumen dan mengelompokkan dokumen ke dalam beberapa topik atau kelompok (Kwartler, 2017). Data yang berjumlah sangat besar juga dapat diproses menggunakan metode ini (Yaman, Sartono, & Soleh, 2021). Disisi lain, LDA tidak dapat digunakan untuk mengklasifikasikan dokumen kedalam aspek tertentu secara langsung (Miller, Dligach, & Savova, 2016). *Output* yang dihasilkan dari metode ini berupa daftar topik yang diberikan bobot untuk setiap dokumen (Campbell, Hindle, & Stroulia, 2015). Penyajian model LDA dapat divisualisasikan seperti **Gambar 3.1**



Gambar 3.1 Representasi Model Grafis LDA.

Sumber : (Setijohatmo dkk, 2020)

Menurut (Blei, Ng, & Jordan, 2003), bentuk lingkaran pada **Gambar 3.1** merepresentasikan individual kata. Lingkaran berwarna abu-abu merepresentasikan variabel yang di observasi dan lingkaran kosong merepresentasikan variabel yang tidak secara langsung di observasi. α dan β sebagai parameter tingkat *corpus*. θ sebagai variabel tingkat dokumen(M). Z dan W sebagai variabel tingkat kata(N). φ merupakan distribusi kata terhadap topik dalam *corpus* dan K adalah kumpulan

topik (Setijohatmo dkk, 2020). Probabilitas dari sebuah *corpus* menurut (Blei, Ng, & Jordan, 2003) dapat dihitung menggunakan persamaan berikut:

$$p(D|\alpha,\beta) = \prod_{d=1}^{M} \int p(\theta_d|\alpha) \left(\prod_{n=1}^{N_d} \sum_{z_n} p(z_{dn}|\theta_d)\, p(w_{dn}|z_{dn},\beta) \right) d\theta_d \qquad (3.8)$$

dengan:

p = probabilitas

M = banyak dokumen dalam *corpus*

N = banyak kata dalam dokumen

α = banyak distribusi topik pada dokumen

β = banyak distribusi kata dalam topik

θ_d = distribusi topik untuk dokumen tertentu

z_{dn} = topik dari kata tertentu pada sebuah dokumen

w_{dn} = kata yang berkaitan dengan topik tertentu yang terdapat di dalam dokumen

Pada persamaan (3.8), indeks *d* menunjukkan urutan dokumen dan *n* menunjukkan urutan kata. Nilai α yang semakin besar menandakan campuran topik yang dibahas di dalam dokumen semakin banyak, nilai β yang semakin tinggi menandakan semakin banyaknya kata di dalam topik, dan semakin tingginya nilai θ maka topik dalam satu dokumen semakin banyak. Diketahui juga bahwa *D* menyatakan jumlah dokumen unik yang tersedia untuk algoritma, misalnya ukuran *corpus* (Hoffman, Blei, & Bach, 2010).

Sementara itu, probabilitas dari kemunculan sebuah dokumen dapat diketahui menggunakan persamaan berikut:

$$P(W,Z,\theta,\varphi|\alpha,\beta) = \prod_{j=1}^{M} P\left(\theta_j;\alpha\right) \prod_{i=1}^{K} P(\varphi_i;\beta) \prod_{t=1}^{N} P\left(Z_{j,t}|\theta_j\right) P\left(W_{j,t}|\varphi z_{j,t}\right) \qquad (3.9)$$

Indeks *j* menunjukkan urutan dokumen, *t* menunjukkan urutan kata, dan *i* menunjukkan urutan topik. $P(W,Z,\theta,\varphi|\alpha,\beta)$ merupakan probabilitas kemunculan dari sebuah dokumen, $\prod_{j=1}^{M} P\left(\theta_j;\alpha\right)$ dan $\prod_{t=1}^{N} P\left(Z_{j,t}|\theta_j\right)$ digunakan untuk menemukan topik, $\prod_{i=1}^{K} P(\varphi_i;\beta)$ dan $P\left(W_{j,t}|\varphi z_{j,t}\right)$ digunakan untuk menemukan kata dari suatu dokumen. $\prod_{j=1}^{M} P\left(\theta_j;\alpha\right)$ dan $\prod_{i=1}^{K} P(\varphi_i;\beta)$ merupakan *dirichlet distributions*, sedangkan $\prod_{t=1}^{N} P\left(Z_{j,t}|\theta_j\right)$ dan $P\left(W_{j,t}|\varphi z_{j,t}\right)$ merupakan multinomial *distributions*. $\prod_{j=1}^{M} P\left(\theta_j;\alpha\right)$ untuk menghitung probabilitas dari suatu dokumen apakah memuat topik tertentu (mengasosiasikan dokumen dengan topik

yang sesuai). $\prod_{i=1}^{K} P(\varphi_i; \beta)$ untuk menghitung probabilitas dari suatu topik yang memuat kata tertentu (mengkaitkan topik yang ada dengan kata-kata yang sesuai). $\prod_{t=1}^{N} P\left(Z_{j,t} \middle| \theta_j\right)$ untuk menghitung probabilitas dari topik berdasarkan nilai ditribusi topik dari dokumen. $P\left(W_{j,t} \middle| \varphi z_{j,t}\right)$ untuk menghitung berapa probabilitas suatu kata untuk diasosiasikan kedalam topik tertentu (Ridhwanullah, 2022).

Menurut (Steyvers & Griffiths, 2006), probabilitas kata pada topik dapat diketahui menggunakan persamaan berikut:

$$P(z_i = j | z_{-i}, w_i, d_i, .) \sim \frac{C_{w_i j}^{WT} + \beta}{\sum_{w=1}^{W} C_{wj}^{WT} + W\beta} \frac{C_{d_i j}^{DT} + \alpha}{\sum_{t=1}^{T} C_{d_i t}^{DT} + T\alpha} \quad (3.10)$$

dengan:

$(z_i = j)$ = penugasan topik dari kata ke-*i* pada topik ke-*j*

z_{-i} = penetapan topik untuk semua kata

"·" = semua informasi lain yang diketahui atau diamati

w_i = kata ke-*i*

d_i = dokumen ke-*i*

C^{WT} = matriks hitungan dengan dimensi $W \times T$

C^{DT} = matriks hitungan dengan dimensi $D \times T$

C_{wj}^{WT} = berapa kali kata w ditugaskan pada topik ke-*j*, selain topik saat ini

$C_{d_j}^{DT}$ = berapa kali topik ke-*j* ditugaskan pada beberapa kata dalam dokumen *d*, selain dokumen saat ini

Pada persamaan (3.10), $\frac{C_{w_i j}^{WT} + \beta}{\sum_{w=1}^{W} C_{wj}^{WT} + W\beta}$ adalah probabilitas kata *w* di bawah topik j sedangkan $\frac{C_{d_i j}^{DT} + \alpha}{\sum_{t=1}^{T} C_{d_i t}^{DT} + T\alpha}$ adalah probabilitas topik ke-*j* berada di bawah distribusi topik saat ini untuk dokumen *d*. Setelah banyak kata yang telah ditetapkan pada topik ke-*j* (di seluruh dokumen) maka akan meningkatkan kemungkinan menetapkan kata tertentu dari kata tersebut pada topik ke-*j*. Pada saat yang sama, jika topik ke-*j* telah digunakan berkali-kali dalam satu dokumen maka akan meningkatkan kemungkinan setiap kata dari dokumen tersebut akan ditempatkan ke topik ke-*j*. Oleh karena itu, kata-kata ditetapkan ke topik bergantung pada seberapa besar kemungkinan kata tersebut untuk suatu topik, serta seberapa dominan suatu topik dalam dokumen. Menurut (Setijohatmo dkk, 2020), nilai

standar untuk α adalah 50/T dengan T adalah jumlah topik di setiap dokumen dan nilai standar untuk β adalah 0,001. Selain itu, (Setijohatmo dkk, 2020) memberikan simulasi perhitungan probabilitas kata pada topik dengan diketahui terdapat 3 dokumen (D1, D2, dan D3) dan 2 topik tersembunyi (T1 dan T2). Inisialisasi topik secara random dan distribusi topik dapat dilihat pada **Tabel 3.1** dan **Tabel 3.2**.

Tabel 3.1 Inisialisasi Topik secara Random.

D1	Money	Bank	loan	bank	money	money	Bank	Loan
	2	2	1	2	1	2	1	2
D2	Money	Bank	bank	river	loan	stream	Bank	money
	1	2	1	2	1	2	1	2
D3	River	Bank	stream	bank	river	river	Stream	bank
	1	2	1	2	1	2	1	2

Tabel 3.2 Distribusi Topik.

	T1	T2
money	2	3
bank	3	6
loan	2	1
river	2	2
stream	2	1

	D1	D2	D3
T1	3	4	4
T2	5	4	4

Perhitungan probabilitas kata "money" pada dokumen 1 di setiap topik dapat dilihat sebagai berikut:

$$P(z_i = T1|z_{-i}, \text{money}, d_i, .) \sim \frac{2 + 0{,}01}{9 + 5 \times 0{,}01} \times \frac{3 + 25}{4 + 2 \times 25} = 0{,}10$$

$$P(z_i = T2|z_{-i}, \text{money}, d_i, .) \sim \frac{2 + 0{,}01}{10 + 5 \times 0{,}01} \times \frac{4 + 25}{3 + 2 \times 25} = 0{,}11$$

Jadi, kata "money" ditetapkan ke dalam T2 karena mempunyai probabilitas yang lebih besar dibandingkan dengan T1.

3.9. *Topic Coherence*

Pemilihan topik yang optimal pada pemodelan LDA dilihat dari *coherence score* setiap topik yang mana topik yang mempunyai *coherence score* tertinggi adalah topik yang optimal (Fernanda, 2021). Nilai ini digunakan sebagai pembeda antara topik yang dapat dilakukan interpretasi secara *semantic* dengan topik hasil temuan inferensi statistik (Stevens et al, 2012). *Topic coherence* dapat dihitung menggunakan formula berikut (Qomariyah, Irawan, & Fithriasari, 2019):

$$coherence\ (V) = \sum_{(v_i, v_j) \in V} score(v_i, v_j, \epsilon) \quad (3.11)$$

$$score(v_i, v_j, \epsilon) = log \frac{D(v_i, v_j)+\epsilon}{D(v_j)} \quad (3.12)$$

dengan :

V = sekumpulan kata yang menjelaskan topik

ϵ = faktor pemulusan yang menjamin bahwa skor akan kembali menjadi bilangan *real* (asli)

$D(v_i, v_j)$ = jumlah kata v_i dan v_j, i dan j sebagai pembeda dalam urutan kata (misal = kata ke-1 menggunakan v_i dan kata ke-2 menggunakan v_j)

$D(v_j)$ = jumlah kemunculan kata v_j

Menurut (Stevens et al, 2012), nilai ϵ adalah 1 karena $log(0)$ sehingga tidak berefek pada nilai *topic coherence* namun nilai $\epsilon \leq 1$ (misal = 10^{-12}) akan menunjukkan performa dari *topic coherence* yang lebih baik karena dapat mengurangi skor pada kata yang sama sekali tidak terkait dengan topik.

BAB IV
METODOLOGI PENELITIAN

4.1. Populasi dan Sampel Penelitian

Populasi dari penelitian ini adalah ulasan pengguna BSI *Mobile* di *Google Play Store*. Total ulasan pengguna BSI *Mobile* di *Google Play Store* pada tanggal 22 November 2022 sebanyak 118.000 ulasan (Google Play, 2023). Sampel yang digunakan dalam penelitian ini sebesar ± 15% dari populasi atau sebanyak 17.757 ulasan pengguna BSI *Mobile* di *Google Play Store*. Ulasan tersebut diambil sejak tanggal 11 Maret 2022 hingga 21 November 2022. Data yang diambil terdiri dari 4 kolom, yaitu userName, at, content, dan score namun pada penelitian ini hanya kolom content yang digunakan.

4.2. Jenis dan Sumber Data

Penelitian ini menggunakan data primer, yaitu ulasan pengguna BSI *Mobile* di *Google Play Store*. Pengambilan data tersebut dilakukan dengan cara *scrapping* menggunakan bantuan *Google Colab*. Perintah untuk melakukan *scrapping* ulasan pengguna BSI *Mobile* di *Google Play Store* seperti berikut.

```
from google_play_scraper import Sort, reviews
result, continuation_token = reviews(
'com.bsm.activity2',
lang='id',
country='id',
sort=Sort.NEWEST,
count=20000,
filter_score_with=None
)
```

4.3. Definisi Variabel

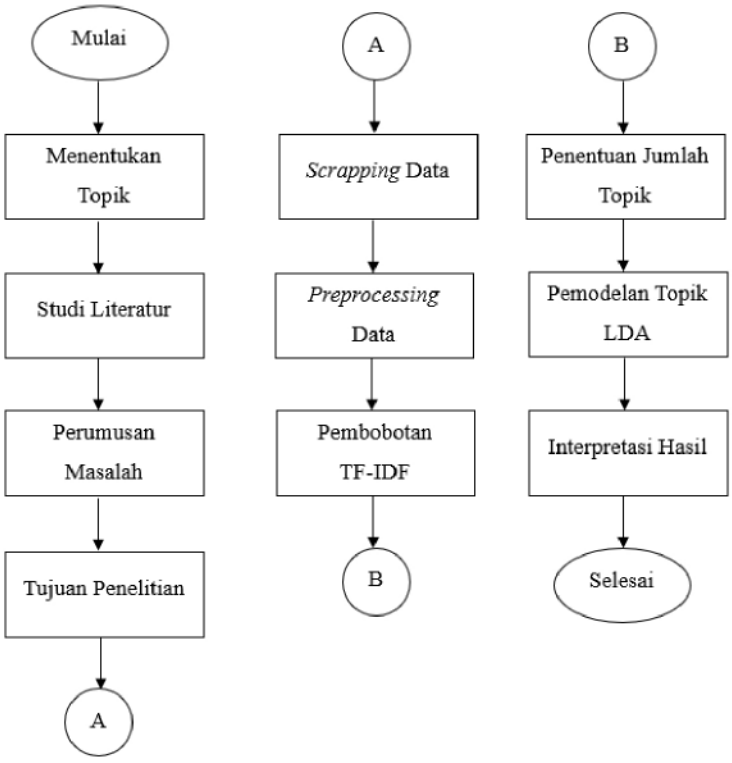
Variabel yang digunakan pada penelitian ini adalah variabel ulasan. Variabel ulasan pada penelitian ini merupakan pendapat berupa komentar maupun saran terkait BSI *Mobile* yang ada di *Google Play Store*.

4.4. Metode Analisis

Analisis pada penelitian ini menggunakan bantuan *software Microsoft Excel* dan *Google Colab*. Metode analisis yang digunakan pada penelitian ini adalah pemodelan topik menggunakan *Latent Dirichlet Allocation* (LDA). LDA merupakan salah satu jenis metode pemodelan topik yang sering digunakan. Pada analisis ini bertujuan untuk mengetahui hasil pemodelan topik ulasan pengguna BSI *Mobile* di *Google Play Store*.

4.5. Tahapan Penelitian

Tahapan dalam pelaksanan penelitian ini terlihat pada **Gambar 4.1**



Gambar 4.1 Tahapan Penelitian.

1. Mulai
2. Menentukan topik penelitian
3. Melakukan studi literatur terkait topik penelitian yang sudah ditentukan agar mendapatkan permasalahan yang akan diteliti.
4. Mengidentifikasi rumusan masalah.
5. Mengidentifikasi tujuan penelitian.
6. *Scrapping* data ulasan pengguna BSI *Mobile* di *Google Play Store*.

7. Melakukan *preprocessing* data meliputi *remove duplicate* data, *remove emoji*, *remove punctuation and number*, *case folding*, *spelling normalization*, stopwords *removal*, *tokenization*, *stemming*, dan penggabungan kata majemuk.
8. Melakukan pembobotan TF-IDF.
9. Penentuan jumlah topik.
10. Melakukan pemodelan topik menggunakan LDA.
11. Interpretasi hasil analisis pemodelan topik menggunakan LDA.

BAB V
HASIL DAN PEMBAHASAN

5.1. Pengambilan data

Pengambilan data ulasan pengguna BSI *Mobile* di *Google Play Store* dilakukan dengan cara *scrapping* melalui *Google Colab*. Proses ini perlu pemasangan *package google play scraper* yang menyediakan API untuk membantu *scrapping* di *Google Play Store.*

Ulasan pengguna BSI *Mobile* di *Google Play Store* yang didapatkan sebanyak 20.000 ulasan terbaru menggunakan Bahasa Indonesia. Ulasan tersebut ditulis oleh pengguna BSI *Mobile* dalam rentang waktu 11 Maret 2022 hingga 21 November 2022. Data yang terambil terdiri dari 4 kolom, yaitu userName, at, content, dan score. userName adalah nama akun pengguna BSI *Mobile* di *Google Play Store*, at adalah waktu yang menunjukkan kapan ulasan ditulis oleh pengguna BSI *Mobile* di *Google Play Store*, content adalah isi ulasan pengguna BSI *Mobile* terkait BSI *Mobile*, dan score adalah rating yang diberikan oleh pengguna BSI *Mobile* terkait BSI *Mobile*. Data tersebut dapat dilihat pada **Tabel 5.1**

Tabel 5.1 5 Ulasan Teratas Pengguna BSI *Mobile* di *Google Play Store*.

userName	at	content	score
RAKA PERMANA NAZIF	21/11/2022 05:48	kok aplikasinya berhenti trs ya?...masalahnya apa?...signal ama penyimpanan aman trs masalahnya apa?	1
Fauzan Ahmad	21/11/2022 05:13	Aplikasinya setelah diapdet makin keren	5
Rizka Aristiani	21/11/2022 04:39	Aplikasinya bagus dan sangat membantu, selama memakainya tidak pernah ada kendala dan error. Jadi makin nyaman pakai aplikasi BSI sekarang. Mantabbb	5
Nadila Fitriani	21/11/2022 04:32	Seneng deh pakai aplikasinya kalau gini, selain memudahkan user bertransaksi, tampilan visualnya juga sangat menarik sehingga memanjakan mata. Semoga konsisten terus seperti ini ya, dan mungkin bisa lebih ditingkatkan lagi jadi lebih baik lagi.	5
Ami Oktaviani	21/11/2022 04:26	Apps nya mudah banget dipakai	5

5.2. *Preprocessing* Data

Tahapan ini digunakan untuk mempersiapkan data agar dapat dilakukan analisis lebih lanjut. Persiapan data yang dilakukan pada penelitian ini meliputi *remove duplicate*, *remove emoji*, *remove punctuation and number*, *case folding*, *spelling normalization*, *stopwords removal*, *tokenization*, *stemming*, dan penggabungan kata majemuk. Serangkain proses *preprocessing* telah menyeleksi data ulasan pengguna BSI *Mobile* di *Google Play Store* sehingga ulasan yang digunakan untuk analisis pemodelan topik sebanyak 17.757 ulasan.

5.2.1 *Remove Duplicate*

Penghapusan data duplikat diperlukan untuk menghindari penggunaan ulasan negatif maupun positif yang dikirimkan secara berturut-turut. Penghapusan ulasan yang dianggap duplikat didasarkan pada kolom userName, content, dan score yang sama. Proses ini dilakukan menggunakan fungsi *Remove Duplicates* pada *Microsoft Excel*.

5.2.2 *Remove Emoji*

Emoji atau karakter gambar perlu dilakukan penghapusan karena isi ulasan yang digunakan pada penelitian ini berupa teks. Perbedaan ulasan sebelum dan sesudah dilakukannya *remove emoji* dapat dilihat pada **Tabel 5.2**

Tabel 5.2 Contoh Proses *Remove Emoji*.

Sebelum	Sesudah
Aplikasi harusnya memudahkan bukan mempersulit ,,tiap aktivitasi verifikasi wajah selalu gagal ,,dan endingnya harus ke kantor dan ngantri ,,,	Aplikasi harusnya memudahkan bukan mempersulit ,,tiap aktivitasi verifikasi wajah selalu gagal ,,dan endingnya harus ke kantor dan ngantri ,,,

5.2.3 *Remove Punctuation and Number*

Penghapusan tanda baca dan nomor pada ulasan dapat dilakukan menggunakan modul RegEx atau Ekspresi Reguler di program *python*. Proses tersebut dapat dilihat pada **Tabel 5.3**

Tabel 5.3 Proses *Remove Punctuation and Number*.

Sebelum	Sesudah
Aplikasi harusnya memudahkan bukan mempersulit ,,tiap aktivitasi verifikasi wajah selalu gagal ,,dan endingnya harus ke kantor dan ngantri ,,,	Aplikasi harusnya memudahkan bukan mempersulit tiap aktivitasi verifikasi wajah selalu gagal dan endingnya harus ke kantor dan ngantri

5.2.4 *Case Folding*

Proses ini digunakan untuk merubah kata pada ulasan yang masih menggunakan huruf kapital menjadi huruf kecil. Hal tersebut dilakukan agar terlihat lebih seragam dalam segi penulisan. Proses *case folding* dapat dilihat pada **Tabel 5.4.**

Tabel 5.4 Proses *Case Folding*.

Sebelum	Sesudah
Aplikasi harusnya memudahkan bukan mempersulit tiap aktivitasi verifikasi wajah selalu gagal dan endingnya harus ke kantor dan ngantri	aplikasi harusnya memudahkan bukan mempersulit tiap aktivitasi verifikasi wajah selalu gagal dan endingnya harus ke kantor dan ngantri

5.2.5 *Spelling Normalization*

Proses ini berupa perbaikan kata yang merupakan bahasa *slang* atau gaul dan kata yang penulisannya tidak benar (*typo*). Hasil dari proses tersebut dapat dilihat pada **Tabel 5.5.**

Tabel 5.5 Proses *Spelling Normalization*.

Sebelum	Sesudah
aplikasi harusnya memudahkan bukan mempersulit tiap aktivitasi verifikasi wajah selalu gagal dan endingnya harus ke kantor dan ngantri	aplikasi harusnya memudahkan bukan mempersulit tiap aktivitasi verifikasi wajah selalu gagal dan akhirnya harus ke kantor dan antre

5.2.6 *Stopwords Removal*

Stopwords removal adalah proses mengeluarkan kata yang tidak berguna atau tidak memberikan informasi penting. Kata tersebut seperti kata depan, kata hubung, nama objek yang diteliti (aplikasi, bsi, *mobile*), dan lain-lain. Hasil dari proses ini dapat dilihat pada **Tabel 5.6**

Tabel 5.6 Proses *Stopwords Removal*.

Sebelum	Sesudah
aplikasi harusnya memudahkan bukan mempersulit tiap aktivitasi verifikasi wajah selalu gagal dan akhirnya harus ke kantor dan antre	mempersulit aktivitasi verifikasi wajah gagal kantor antre

5.2.7 *Tokenization*

Tokenization adalah proses memecah kalimat menjadi per kata. Proses ini akan menghasilkan token yang berguna untuk pembersihan data tahap selanjutnya. Hasil proses ini dapat dilihat pada **Tabel 5.7.**

Tabel 5.7 Proses *Tokenization*.

Sebelum	Sesudah
mempersulit aktivitasi verifikasi wajah gagal kantor antre	['mempersulit', 'aktivasi', 'verifikasi', 'wajah', 'gagal', 'kantor', 'antre']

5.2.8 *Stemming*

Stemming digunakan untuk mengubah kata menjadi kata dasar dengan cara menghapus imbuhan dari kata tersebut. Hasil dari proses tersebut dapat dilihat pada **Tabel 5.8**

Tabel 5.8 Proses *Stemming*.

Sebelum	Sesudah
mempersulit aktivitasi verifikasi wajah gagal kantor antre	sulit aktivitasi verifikasi wajah gagal kantor antre

5.2.9 Penggabungan Kata Majemuk

Proses ini digunakan ketika terdapat 2 kata yang bersandingan memberikan makna sama sehingga jika tidak digabungkan dapat memberikan makna berbeda pada hasil analisis selanjutnya. Kata majemuk yang dimaksud seperti *customer service*, nomor rekening, tidak bisa, dan lain-lain. Hasil dari proses ini dapat dilihat pada **Tabel 5.9**

Tabel 5.9 Proses Penggabungan Kata Majemuk.

Sebelum	Sesudah
sulit aktivitasi verifikasi wajah gagal kantor antre	sulit aktivitasi verifikasi_wajah gagal kantor antre

5.3. Pembobotan TF-IDF

TF-IDF merupakan metode pembobotan hubungan kata yang dihitung dari frekuensi kemunculan kata tersebut sehingga dapat diketahui seberapa penting kata tersebut pada dokumen. Hasil dari proses TF-IDF dapat dilihat pada **Tabel 5.10**

Tabel 5.10 Hasil Pembobotan TF-IDF.

D	abal	abjad	….	tidak_valid	tidak_verifikasi	…	zakat	zaman
1	0	0	….	0	0	….	0	0
2	0	0	….	0	0	….	0	0
3	0	0	….	0	0	….	0	0
4	0	0	….	0	0	….	0	0
5	0	0	….	0	0	….	0	0
6	0	0	….	0	0	….	0	0
7	0	0	….	8,838624934	0	….	0	0
…	…	…	….	…	…	….	….	….
17757	0	0	….	0	0	….	0	0

Pada **Tabel 5.10** terlihat skor pembobotan TF-IDF, D merupakan jumlah dokumen atau baris, yaitu sebanyak 17.757 dan *term* atau kata sebanyak 2013. Proses penentuan skor TF-IDF secara manual dapat dijelaskan sebagai berikut:
Ketika yang akan dicari adalah kata "tidak_valid" pada dokumen ke-7 (D_7) dengan diketahui bahwa sebanyak 7 dokumen mengandung kata "tidak_valid", jumlah dokumen sebanyak 17.757, dan kata "tidak_valid" muncul di dokumen ke-7 sebanyak 1 kali. Maka didapatkan nilai IDF seperti berikut ini:

$$IDF_j = ln\frac{D}{df_j}$$

$$IDF_7 = ln\frac{17757}{7}$$

$$IDF_7 = 7{,}838624934$$

Kemudian untuk skor TF-IDF sebagai berikut:

$$w_{ij} = tf_{ij} \times (ln\left(\frac{D}{df_j}\right) + 1)$$

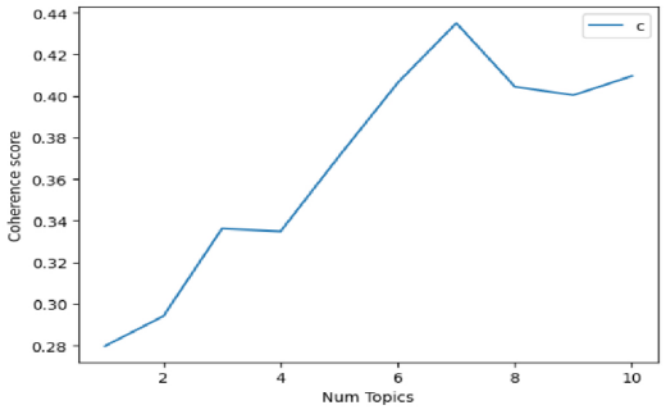
$$w = 1 \times (IDF_7 + 1)$$

$$w = 1 \times (7{,}838624934 + 1)$$

$$w = 8{,}838624934$$

5.4. Pemodelan Topik

Metode pemodelan topik yang digunakan penulis adalah LDA. Penggunaan LDA memberikan gambaran topik yang paling banyak dibicarakan pada data ulasan pengguna BSI *Mobile* di *Google Play Store*. Berdasarkan grafik *coherence score* pada **Gambar 5.1.** dapat diketahui banyak model yang baik digunakan pada penelitian ini.



Gambar 5.1 Grafik *Coherence Score*.

Grafik *coherence score* pada **Gambar 5.1.** menunjukkan bahwa *coherence score* tertinggi adalah 0,43 dengan *num topics* atau jumlah topik sebanyak 7. Secara lebih rinci *coherence score* di setiap *num topics* dapat dilihat pada **Tabel 5.11** berikut.

Tabel 5.11 *Coherence Score*.

Num Topics	*Coherence Score*	*Num Topics*	*Coherence Score*
1	0,279642135	6	0,40631943
2	0,2940825337	7	0,4350406246
3	0,336188032	8	0,4044120311
4	0,3347872146	9	0,4003916539
5	0,3710660207	10	0,4095745254

Berdasarkan hasil *coherence score* pada **Gambar 5.1** dan **Tabel 5.11** maka pada penelitian ini akan dibentuk sebanyak 7 topik model.

Penelitian dari (Roder, Both, & Hinneburg, 2015) memberikan studi kasus untuk mengaplikasikan rumus *topic coherence* dengan diketahui bahwa sekumpulan kata yang menjelaskan topik = {*game*, *sport*, *ball*, *team*}. Maka perhitungan *coherence score* menggunakan rumus (3.11) dan (3.12) dari topik tersebut adalah seperti berikut:

$$coherence\ (V) = [score(sport, game) + score(ball, sport) + score(team, ball)]$$

$$= \left[\left(log \frac{P(sport,game)+\epsilon}{P(game)}\right) + \left(log \frac{P(ball,sport)+\epsilon}{P(sport)}\right) + \left(log \frac{P(team,ball)+\epsilon}{P(ball)}\right)\right]$$

$$coherence\ (V) = \left[\left(log \frac{1+1}{1}\right) + \left(log \frac{1+1}{1}\right) + \left(log \frac{1+1}{1}\right)\right]$$

$$coherence\ (V) = [(log\ (2)) + (\log\ (2)) + (\log\ (2))]$$

$$coherence\ (V) = [0{,}30102999566 + 0{,}30102999566 + 0{,}30102999566]$$

$$coherence\ (V) = 0{,}90308998698$$

Beradasarkan perhitungan diatas dapat diketahui *coherence score* dari topik sebesar 0,90308998698.

5.4.1 Model LDA Topik 1

Model LDA dari topik 1 dapat dilihat pada **Tabel 5.12**

Tabel 5.12 Model LDA dari Topik 1.

Topik 1
0,273*"mudah" + 0,105*"transaksi" + 0,062*"nyaman" + 0,047*"fitur_lengkap" + 0,043*"pakai" + 0,039*"aman" + 0,039*"guna" + 0,033*"fitur" + 0,031*"lengkap" + 0,020*"gampang"

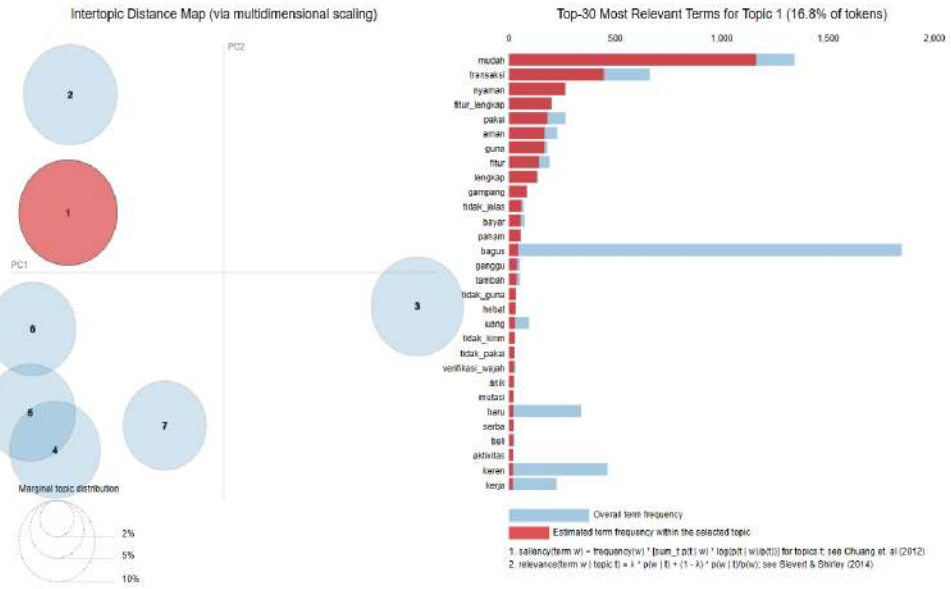
Model LDA dari topik 1 terlihat seperti pada **Tabel 5.12** terdapat 10 kata teratas yang mempunyai kontribusi tertinggi terhadap topik 1, yaitu kata "mudah" dengan nilai probabilitas sebesar 0,273, kata "transaksi" dengan nilai probabilitas sebesar 0,105, kata "nyaman" dengan nilai probabilitas sebesar 0,062, kata "fitur_lengkap" dengan nilai probabilitas sebesar 0,047, kata "pakai" dengan nilai probabilitas sebesar 0,043, kata "aman" dengan nilai probabilitas sebesar 0,039, kata "guna" dengan nilai probabilitas sebesar 0,039, kata "fitur" dengan nilai probabilitas sebesar 0,033, kata "lengkap" dengan nilai probabilitas sebesar 0,031, dan kata "gampang" dengan nilai probabilitas 0,020.

Berikut *wordcloud* dari kumpulan kata yang berada di topik 1 seperti pada **Gambar 5.2.**

Gambar 5.2 *Wordcloud* Topik 1.

Wordcloud topik 1 pada **Gambar 5.2** memberikan informasi bahwa 10 kata yang paling sering muncul adalah kata "mudah", "transaksi", "fitur_lengkap", "nyaman", "pakai", "aman", "fitur", "guna", "lengkap", dan "lengkap" yang ditunjukkan dengan ukuran huruf paling besar.

Visualisasi berupa *intertopic distance map* dari topik 1 dapat dilihat pada **Gambar 5.3**



Gambar 5.3 Visualisasi *Intertopic Distance Map* dari Topik 1.

Diagram pada **Gambar 5.3** menampilkan jarak topik 1 dengan topik lainnya. PC1 merupakan sumbu x dan PC2 merupakan sumbu y. Nilai PC1 dari topik 1 adalah -0,073933 dan nilai PC2 adalah 0,141553 sehingga topik 1 berada di kuadran 2. Letak topik 1 di kuadran 2 memperlihatkan bahwa topik 1 mempunyai jarak yang dekat dengan topik 2 yang berarti bahwa tingkat kesamaan antara topik 1 dengan topik 2 tinggi, sedangkan jarak topik 1 dengan topik 3, 4, 5, 6, dan 7 jauh yang berarti bahwa tingkat kesamaan topik 1 dengan topik 3, 4, 5, 6, dan 7 rendah. Selain itu, **Gambar 5.3** juga memuat *bar chart top 30 most relevant terms for topics 1* yang menunjukkan frekuensi kemunculan 30 kata paling relevan untuk topik 1, warna merah pada *bar chart* tersebut menunjukkan frekuensi dari kata yang muncul pada topik 1 dan warna biru menunjukkan keseluruhan frekuensi dari kata. 10 kata teratas pada *bar chart* tersebut sesuai dengan 10 kata pada **Tabel 5.12**. Topik 1 mempunyai *marginal topic distribution* sebesar 16,8% yang berarti bahwa persentase kata pada topik 1 memuat sebesar 16,8% dari keseluruhan topik dalam ulasan.

5.4.2 Model LDA Topik 2

Topik 2 mempunyai Model LDA seperti pada **Tabel 5.13**

Tabel 5.13 Model LDA dari Topik 2.

Topik 2
0,319*"bantu" + 0,161*"tidak_bisa" + 0,111*"keren" + 0,054*"cepat" + 0,032*"transaksi" + 0,024*"baru" + 0,018*"praktis" + 0,014*"rating" + 0,013*"hapus" + 0,013*"maintenance"

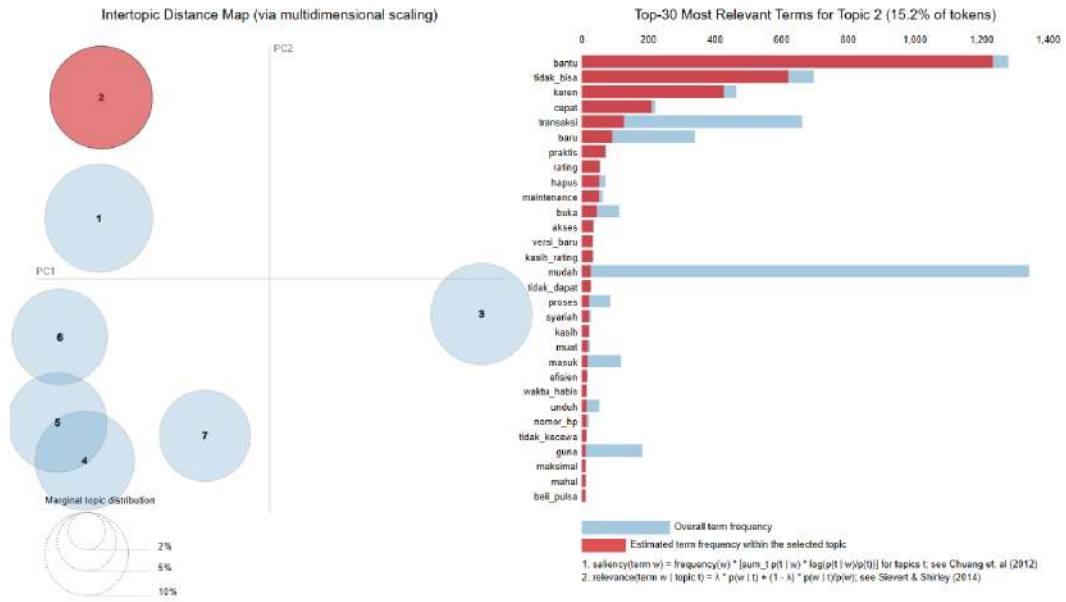
Pada **Tabel 5.13** terlihat bahwa pada topik 2 terdapat 10 kata teratas yang mempunyai kontribusi tertinggi, yaitu kata "bantu" dengan nilai probabilitas sebesar 0,319, kata "tidak_bisa" dengan nilai probabilitas sebesar 0,161, kata "keren" dengan nilai probabilitas sebesar 0,111, kata "cepat" dengan nilai probabilitas sebesar 0,054, kata "transaksi" dengan nilai probabilitas sebesar 0,032, kata "baru" dengan nilai probabilitas sebesar 0,024, kata "praktis" dengan nilai probabilitas sebesar 0,018, kata "rating" dengan nilai probabilitas sebesar 0,014, kata "hapus" dengan nilai probabilitas sebesar 0,013, dan kata "maintenance" dengan nilai probabilitas sebesar 0,013.

Kumpulan kata yang berada pada topik 2 dapat dilihat pada visualisasi *wordcloud* seperti **Gambar 5.4**

Gambar 5.4 *Wordcloud* Topik 2.

Berdasarkan **Gambar 5.4** diketahui bahwa kata "bantu", "tidak_bisa", "keren", "cepat", "transaksi", "praktis", "baru", "maintenance", "rating", dan "hapus" adalah 10 kata yang paling sering muncul ditunjukkan dengan ukuran huruf paling besar.

Visualisasi berupa *intertopic distance map* dari topik 2 dapat dilihat pada **Gambar 5.5**



Gambar 5.5 Visualisasi *Intertopic Distance Map* dari Topik 2.

Intertopic distance map dari topik 2 seperti **Gambar 5.5** memperlihatkan jarak topik 2 dengan topik lainnya. Nilai PC1 dari topik 2 adalah -0,070919 dan nilai PC2 adalah 0,299006 sehingga topik 2 berada di kuadran 2. Letak topik 2 di kuadran 2 memperlihatkan bahwa topik 2 mempunyai jarak yang dekat dengan topik 1 yang berarti bahwa tingkat kesamaan antara topik 2 dengan topik 1 tinggi, sedangkan jarak topik 1 dengan topik 3, 4, 5, 6, dan 7 jauh yang berarti bahwa tingkat kesamaan topik 2 dengan topik 3, 4, 5, 6, dan 7 rendah. Selain itu, **Gambar 5.5** juga memuat *bar chart top 30 most relevant terms for topics 2* yang menunjukkan frekuensi kemunculan 30 kata paling relevan untuk topik 2, warna merah pada *bar chart* tersebut menunjukkan frekuensi dari kata yang muncul pada topik 2 dan warna biru menunjukkan keseluruhan frekuensi dari kata. 10 kata teratas pada *bar chart* tersebut sesuai dengan 10 kata pada **Tabel 5.13**. Topik 2 mempunyai *marginal topic distribution* sebesar 15,2% yang berarti bahwa persentase kata pada topik 2 memuat sebesar 15,2% dari keseluruhan topik dalam ulasan.

5.4.3 Model LDA Topik 3

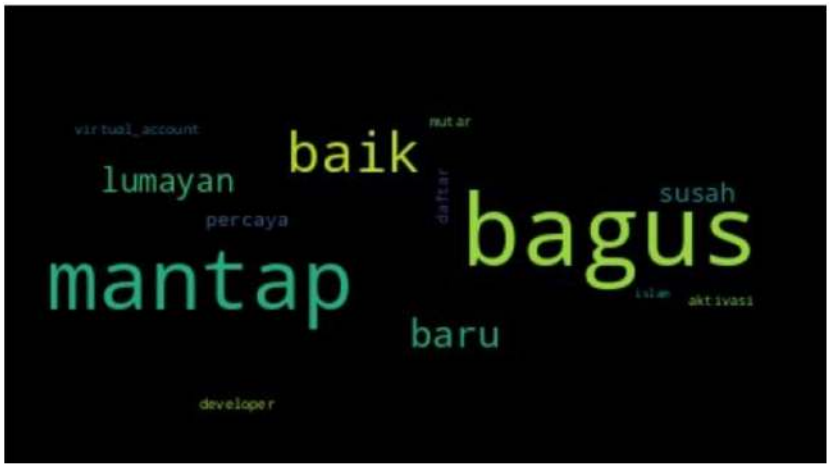
Hasil model LDA dari topik 3 ditunjukkan pada **Tabel 5.14**

Tabel 5.14 Model LDA dari Topik 3.

Topik 3
0,445*"bagus" + 0,315*"mantap" + 0,089*"baik" + 0,034*"baru" + 0,022*"lumayan" + 0,013*"susah" + 0,007*"percaya" + 0,004*"daftar" + 0,003*"mutar" + 0,003*"aktivasi"

Model LDA topik 3 seperti **Tabel 5.14** terlihat bahwa 10 kata teratas yang mempunyai kontribusi tertinggi terhadap topik 3, yaitu kata "bagus" dengan nilai probabilitas sebesar 0,445, kata "mantap" dengan nilai probabilitas sebesar 0,315, kata "baik" dengan nilai probabilitas sebesar 0,089, kata "baru" dengan nilai probabilitas sebesar 0,034, kata "lumayan" dengan nilai probabilitas sebesar 0,022, kata "susah" dengan nilai probabilitas sebesar 0,013, kata "percaya" dengan nilai probabilitas sebesar 0,007, kata "daftar" dengan nilai probabilitas sebesar 0,004, kata "mutar" dengan nilai probabilitas sebesar 0,003, dan kata "aktivasi" dengan nilai probabilitas sebesar 0,003.

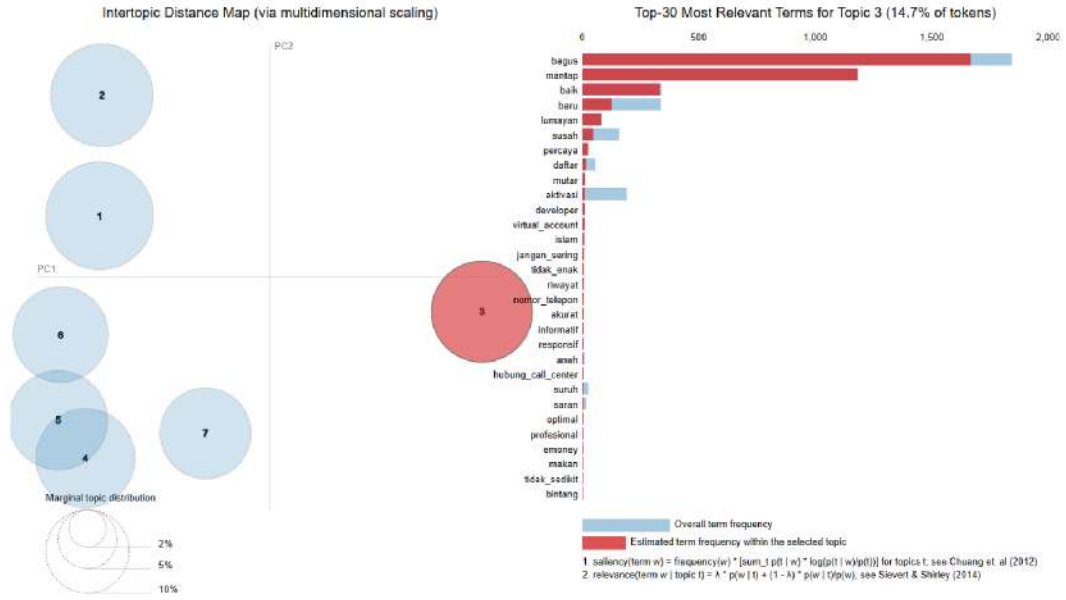
Wordcloud dari kumpulan kata yang berada di topik 3 dapat dilihat pada **Gambar 5.6**



Gambar 5.6 *Wordcloud* Topik 3.

Berdasarkan **Gambar 5.6** dapat diketahui bahwa 10 kata yang paling sering muncul adalah "bagus", "mantap", "baik", "lumayan", "baik", "baru", "susah", "percaya", "daftar", dan "mutar" yang ditunjukkan dengan ukuran huruf paling besar.

Visualisasi berupa *intertopic distance map* dari topik 3 dapat dilihat pada **Gambar 5.7**



Gambar 5.7 Visualisasi *Intertopic Distance Map* dari Topik 3.

Visualisasi pada **Gambar 5.7** memberikan informasi terkait jarak topik 3 dengan topik lainnya. Nilai PC1 dari topik 3 adalah 0,425685 dan nilai PC2 adalah 0,016308 sehingga topik 3 berada di kuadran 1. Letak topik 3 di kuadran 1 memperlihatkan bahwa topik 3 mempunyai jarak yang jauh dengan topik lainnya yang berarti bahwa tingkat kesamaan topik 3 dengan topik lainnya rendah. Selain itu, **Gambar 5.7** juga memuat *bar chart top 30 most relevant terms for topics 3* yang menunjukkan frekuensi kemunculan 30 kata paling relevan untuk topik 3, warna merah pada *bar chart* tersebut menunjukkan frekuensi dari kata yang muncul pada topik 3 dan warna biru menunjukkan keseluruhan frekuensi dari kata. 10 kata teratas pada *bar chart* tersebut sesuai dengan 10 kata pada **Tabel 5.14**. Topik 3 mempunyai *marginal topic distribution* sebesar 14,7% yang berarti bahwa persentase kata pada topik 3 memuat sebesar 14,7% dari keseluruhan topik dalam ulasan.

5.4.4 Model LDA Topik 4

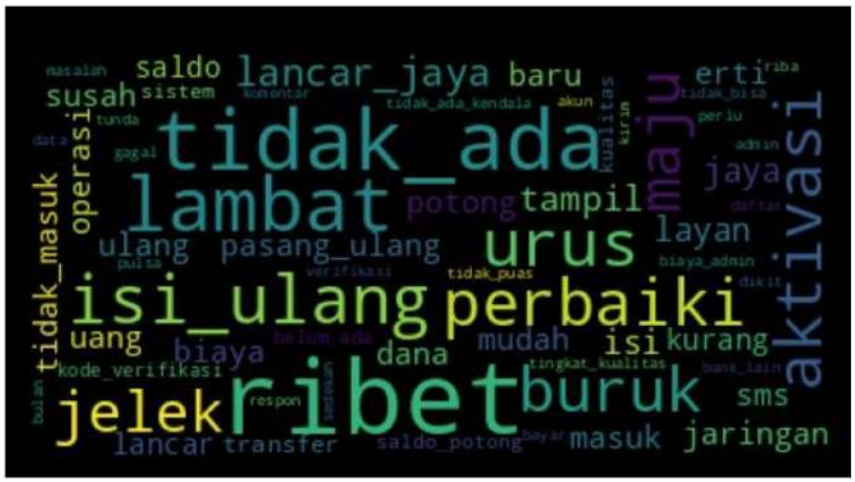
Topik 4 mempunyai model LDA seperti pada **Tabel 5.15**

Tabel 5.15 Model LDA dari Topik 4.

Topik 4
0,067*"ribet" + 0,043*"tidak_ada" + 0,035*"lambat" + 0,027*"isi_ulang" + 0,025*"urus" + 0,023*"aktivasi" + 0,023*"perbaiki" + 0,023*"buruk" + 0,021*"maju" + 0,020*"jelek"

Berdasarkan **Tabel 5.15** terlihat bahwa 10 kata teratas yang mempunyai kontribusi tertinggi terhadap topik 4, yaitu kata "ribet" dengan nilai probabilitas sebesar 0.067, kata "tidak_ada" dengan nilai probabilitas sebesar 0,043, kata "lambat" dengan nilai probabilitas sebesar 0,035, kata "isi_ulang" dengan nilai probabilitas sebesar 0,027, kata "urus" dengan nilai probabilitas sebesar 0,025, kata "aktivasi" dengan nilai probabilitas sebesar 0,023, kata "perbaiki" dengan nilai probabilitas sebesar 0,023, kata "buruk" dengan nilai probabilitas sebesar 0,023, kata "maju" dengan nilai probabilitas sebesar 0,021, dan kata "jelek" dengan nilai probabilitas sebesar 0,020.

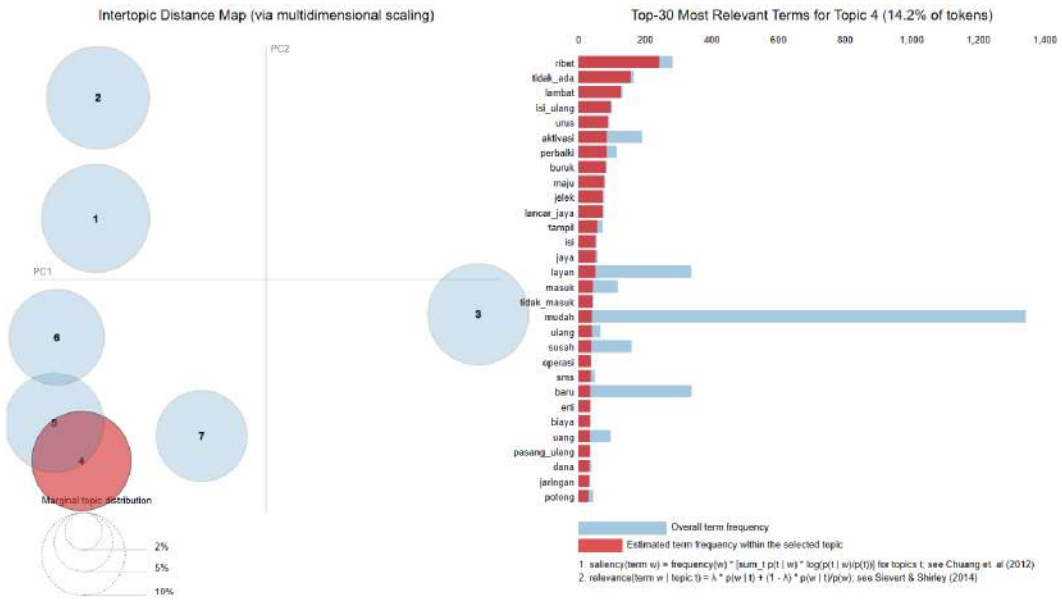
Wordcloud dari kumpulan kata yang berada di topik 4 seperti pada **Gambar 5.8**



Gambar 5.8 *Wordcloud* Topik 4.

Wordcloud topik 4 pada **Gambar 5.8** terlihat 10 kata yang paling sering muncul adalah "ribet", "tidak_ada", "lambat", "isi_ulang", "perbaiki", "aktivasi", "isi_ulang", "urus", "jelek", dan "buruk" yang ditunjukkan dengan ukuran huruf paling besar.

Intertopic distance map dari topik 4 dapat dilihat pada **Gambar 5.9**



Gambar 5.9 Visualisasi *Intertopic Distance Map* dari Topik 4.

Topik 4 mempunyai visualisasi *intertopic distance map* seperti **Gambar 5.9** yang memberikan informasi terkait jarak topik 4 dengan topik lainnya. Nilai PC1 dari topik 4 adalah -0.092352 dan nilai PC2 adalah -0.175229 sehingga topik 4 berada di kuadran 3. Letak topik 4 di kuadran 3 memperlihatkan bahwa topik 4 mempunyai jarak yang dekat dengan topik 5, 6, dan 7 yang berarti bahwa tingkat kesamaan antara topik 4 dengan topik 5, 6, dan 7 tinggi, sedangkan jarak topik 4 dengan topik 1, 2, dan 3 jauh yang berarti bahwa tingkat kesamaan topik 4 dengan topik 1, 2, dan 3 rendah. Selain itu, **Gambar 5.9** juga memuat *bar chart top 30 most relevant terms for topics 4* yang menunjukkan frekuensi kemunculan 30 kata paling relevan untuk topik 4, warna merah pada *bar chart* tersebut menunjukkan frekuensi dari kata yang muncul pada topik 4 dan warna biru menunjukkan keseluruhan frekuensi dari kata. 10 kata teratas pada *bar chart* tersebut sesuai dengan 10 kata pada **Tabel 5.15**. Topik 4 mempunyai *marginal topic distribution* sebesar 14,2% yang berarti bahwa persentase kata pada topik 4 memuat sebesar 14,2% dari keseluruhan topik dalam ulasan.

5.4.5 Model LDA Topik 5

Hasil model LDA dari topik 5 dapat dilihat pada **Tabel 5.16**

Tabel 5.16 Model LDA dari Topik 5.

Topik 5
0,065*"layan" + 0,054*"berkah" + 0,035*"senang" + 0,029*"simpel" + 0,025*"amanah" + 0,022*"customer_service" + 0,019*"takjub" + 0,019*"habis_pulsa" + 0,018*"tidak_perlu" + 0,017*"depan"

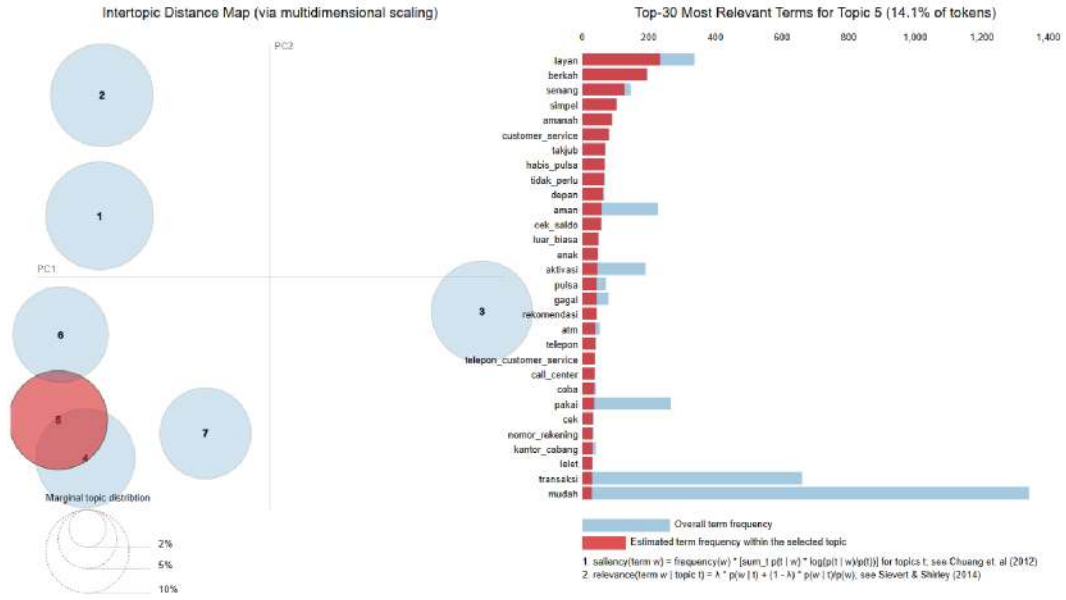
Model LDA dari topik 5 terlihat pada **Tabel 5.16** terdapat 10 kata teratas yang mempunyai kontribusi tertinggi terhadap topik 5, yaitu kata "layan" dengan nilai probabilitas sebesar 0,065, kata "berkah" dengan nilai probabilitas sebesar 0,054, kata "senang" dengan nilai probabilitas sebesar 0,035, kata "simpel" dengan nilai probabilitas sebesar 0,029, kata "amanah" dengan nilai probabilitas sebesar 0,025, kata "customer_service" dengan nilai probabilitas sebesar 0,022, kata "takjub" dengan nilai probabilitas sebesar 0,019, kata "habis_pulsa" dengan nilai probabilitas sebesar 0,019, kata "tidak_perlu" dengan nilai probabilitas sebesar 0,018, dan kata "depan" dengan nilai probabilitas sebesar 0,017.

Berikut merupakan *wordcloud* dari kumpulan kata yang berada di topik 5 seperti pada **Gambar 5.10**

Gambar 5.10 *Wordcloud* Topik 5.

Berdasarkan **Gambar 5.10** terlihat bahwa 10 kata yang paling sering muncul adalah "layan", "berkah", "senang", "simpel", "amanah", "customer_service", "habis_pulsa", "takjub", "tidak_perlu", dan "cek_saldo" yang ditunjukkan dengan ukuran huruf paling besar.

Visualisasi berupa *intertopic distance map* dari topik 5 dapat dilihat pada **Gambar 5.11**



Gambar 5.11 Visualisasi *Intertopic Distance Map* dari Topik 5.

Visualisasi pada **Gambar 5.11** memberikan informasi jarak topik 5 dengan topik lainnya. Nilai PC1 dari topik 5 adalah -0.127998 dan nilai PC2 adalah -0.125399 sehingga topik 5 berada di kuadran 3. Letak topik 5 di kuadran 3 memperlihatkan bahwa topik 5 mempunyai jarak yang dekat dengan topik 4, 6, dan 7 yang berarti bahwa tingkat kesamaan antara topik 5 dengan topik 4, 6, dan 7 tinggi, sedangkan jarak topik 5 dengan topik 1, 2, dan 3 jauh yang berarti bahwa tingkat kesamaan topik 5 dengan topik 1, 2, dan 3 rendah. Selain itu, **Gambar 5.3** juga memuat *bar chart top 30 most relevant terms for topics 5* yang menunjukkan frekuensi kemunculan 30 kata paling relevan untuk topik 5, warna merah pada *bar chart* tersebut menunjukkan frekuensi dari kata yang muncul pada topik 5 dan warna biru menunjukkan keseluruhan frekuensi dari kata. 10 kata teratas pada *bar chart* tersebut sesuai dengan 10 kata pada **Tabel 5.16**. Topik 5 mempunyai *marginal topic distribution* sebesar 14,1% yang berarti bahwa persentase kata pada topik 5 memuat sebesar 14,1% dari keseluruhan topik dalam ulasan.

5.4.6 Model LDA Topik 6

Topik 6 mempunyai model LDA seperti pada **Gambar 5.17**

Tabel 5.17 Model LDA dari Topik 6.

Topik 6
0,179*"syukur" + 0,101*"puas" + 0,089*"manfaat" + 0,062*"lancar" + 0,031*"eror" + 0,022*"transfer" + 0,020*"buka_rekening" + 0,018*"buka" + 0,015*"mudah" + 0,014*"layan"

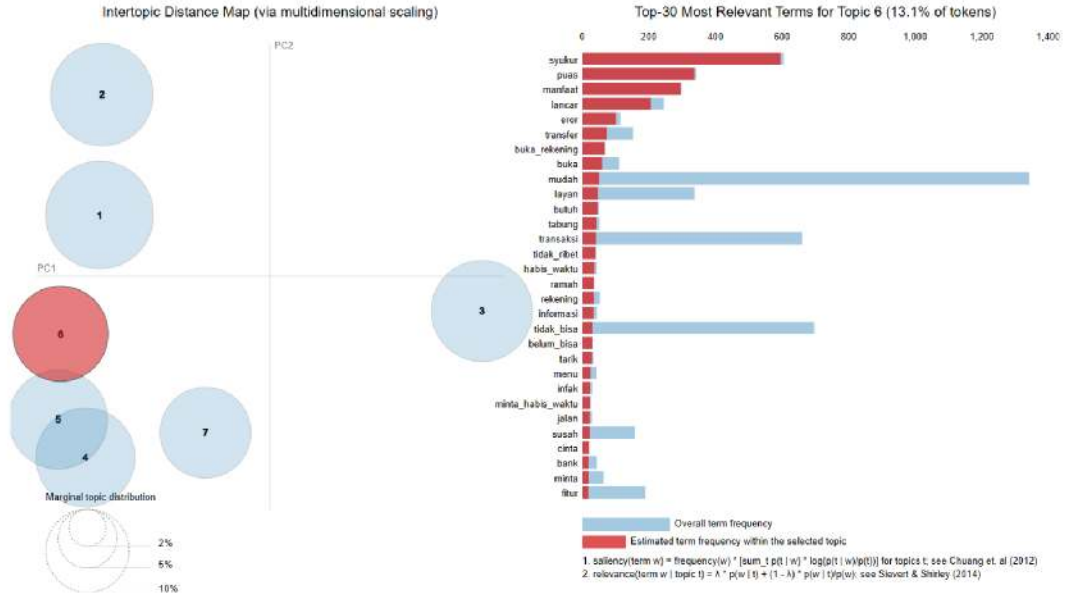
Model LDA topik 6 seperti **Tabel 5.17.** terdapat 10 kata teratas yang mempunyai kontribusi tertinggi terhadap topik 6, yaitu kata "syukur" dengan nilai probabilitas sebesar 0,179, kata "puas" dengan nilai probabilitas sebesar 0,101, kata "manfaat" dengan nilai probabilitas sebesar 0,089, kata "lancar" dengan nilai probabilitas sebesar 0,062, kata "eror" dengan nilai probabilitas sebesar 0,031, kata "transfer" dengan nilai probabilitas sebesar 0,022, kata "buka rekening" dengan nilai probabilitas sebesar 0,020, kata "buka" dengan nilai probabilitas sebesar 0,018, kata "mudah" dengan nilai probabilitas sebesar 0,015, dan kata "layan" dengan nilai probabilitas sebesar 0,014.

Berikut diberikan *wordcloud* dari kumpulan kata yang berada di topik 6 seperti pada **Gambar 5.12**

Gambar 5.12 *Wordcloud* Topik 6.

Wordcloud topik 6 pada **Gambar 5.12** dapat diketahui bahwa 10 kata yang paling sering muncul pada kumpulan kata di topik 6 adalah "syukur", "manfaat", "puas", "lancar", "eror", "buka_rekening", "transfer", "eror", "butuh", dan "transaksi" yang ditunjukkan dengan ukuran huruf paling besar.

Visualisasi berupa *intertopic distance map* dari topik 6 dapat dilihat pada **Gambar 5.13**



Gambar 5.13 Visualisasi *Intertopic Distance Map* dari Topik 6.

Diagram pada **Gambar 5.13** memberikan informasi jarak topik 6 dengan topik lainnya. Nilai PC1 dari topik 6 adalah -0.013547 dan nilai PC2 adalah -0.013547 sehingga topik 6 berada di kuadran 3. Letak topik 6 di kuadran 3 menunjukkan bahwa topik 6 mempunyai jarak yang dekat dengan topik 4, 5, dan 7 yang berarti bahwa tingkat kesamaan antara topik 6 dengan topik 4, 5, dan 7 tinggi, sedangkan jarak topik 6 dengan topik 1, 2, dan 3 jauh yang berarti bahwa tingkat kesamaan topik 6 dengan topik 1, 2, dan 3 rendah. Selain itu, **Gambar 5.13** juga memuat *bar chart top 30 most relevant terms for topics 6* yang menunjukkan frekuensi kemunculan 30 kata paling relevan untuk topik 6, warna merah pada *bar chart* tersebut menunjukkan frekuensi dari kata yang muncul pada topik 6 dan warna biru menunjukkan keseluruhan frekuensi dari kata. 10 kata teratas pada *bar chart* tersebut sesuai dengan 10 kata pada **Tabel 5.17**. Topik 6 mempunyai *marginal topic distribution* sebesar 13,1% yang berarti bahwa persentase kata pada topik 6 memuat sebesar 13,1% dari keseluruhan topik dalam ulasan.

5.4.7 Model LDA Topik 7

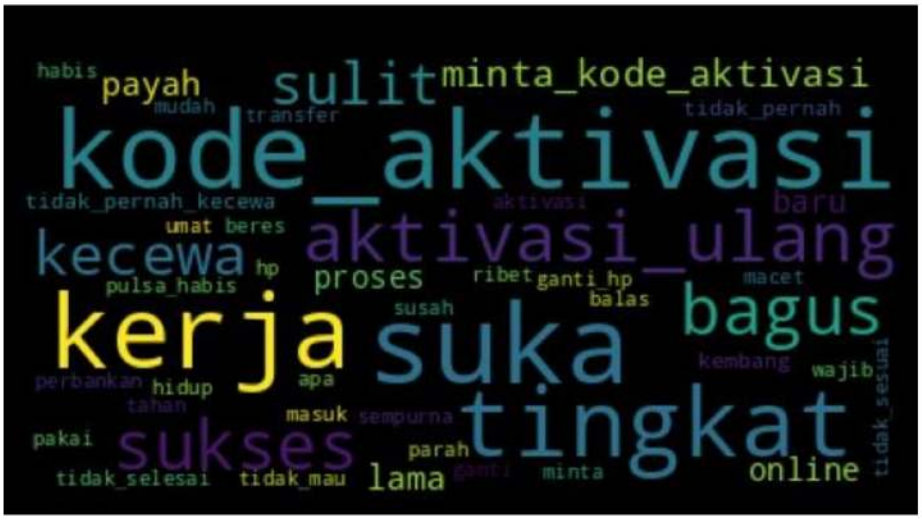
Model LDA dari topik 7 dapat dilihat pada **Tabel 5.18**

Tabel 5.18 Model LDA dari Topik 7.

Topik 7
0,077*"suka" + 0,076*"kode_aktivasi" + 0,066*"kerja" + 0,064*"tingkat" + 0,039*"aktivasi_ulang" + 0,038*"sukses" + 0,036*"bagus" + 0,028*"kecewa" + 0,026*"sulit" + 0,024*"minta_kode_aktivasi"

Berdasarkan **Tabel 5.18** terdapat 30 kata teratas yang mempunyai kontribusi tertinggi terhadap topik 7, yaitu kata "suka" dengan nilai probabilitas sebesar 0,077, kata "kode_aktivasi" dengan nilai probabilitas sebesar 0,076, kata "kerja" dengan nilai probabilitas sebesar 0,066, kata "tingkat" dengan nilai probabilitas sebesar 0,064, kata "aktivasi_ulang" dengan nilai probabilitas sebesar 0,039, kata "sukses" dengan nilai probabilitas sebesar 0,038, kata "bagus" dengan nilai probabilitas sebesar 0,036, kata "kecewa" dengan nilai probabilitas sebesar 0,028, kata "sulit" dengan nilai probabilitas sebesar 0,026, dan kata "minta_kode_aktivasi" dengan nilai probabilitas sebesar 0,024.

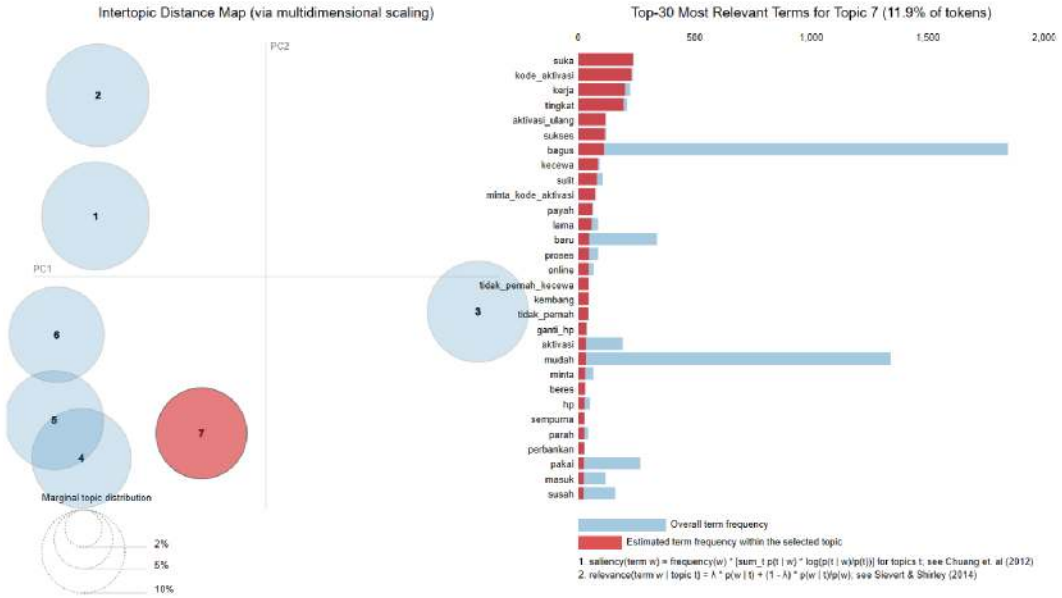
Wordcloud dari kumpulan kata yang berada di topik 7 seperti pada **Gambar 5.14**



Gambar 5.14 *Wordcloud* Topik 7.

Wordcloud topik 7 seperti **Gambar 5.14** dapat diketahui bahwa 10 kata yang paling sering muncul pada kumpulan kata di topik 7 adalah "suka", "kode_aktivasi", "kerja", "tingkat", "aktivasi_ulang", "sukses", "kecewa", "sulit", "minta_kode_aktivasi", dan "online" yang ditunjukkan dengan ukuran huruf paling besar.

Visualisasi berupa *intertopic distance map* dari topik 7 dapat dilihat pada **Gambar 5.15**



Gambar 5.15 Visualisasi *Intertopic Distance Map* dari Topik 7.

Visualisasi pada **Gambar 5.15** terlihat jarak topik 7 dengan topik lainnya. Nilai PC1 dari topik 1 adalah 0.064457 dan nilai PC2 adalah -0.142692. Topik 7 terletak pada kuadran 3 namun nilai PC1 dari topik 7 tidak sesuai dengan nilai PC1 di kudran 3 karena kuadran 3 mempunyai nilai PC1 (-) dan PC2 (-). Letak topik 7 di kuadran 3 memperlihatkan bahwa topik 7 mempunyai jarak yang dekat dengan topik 4, 5, dan 6 yang berarti bahwa tingkat kesamaan antara topik 7 dengan topik 4, 5, dan 6 tinggi, sedangkan jarak topik 7 dengan topik 1, 2, dan 3 jauh yang berarti bahwa tingkat kesamaan topik 7 dengan topik 1, 2, dan 3 rendah. Selain itu, **Gambar 5.3** juga memuat *bar chart top 30 most relevant terms for topics 7* yang menunjukkan frekuensi kemunculan 30 kata paling relevan untuk topik 7, warna merah pada *bar chart* tersebut menunjukkan frekuensi dari kata yang muncul pada topik 7 dan warna biru menunjukkan keseluruhan frekuensi dari kata. 10 kata teratas pada *bar chart* tersebut sesuai dengan 10 kata pada **Tabel 5.18**. Topik 7 mempunyai *marginal topic distribution* sebesar 11,9% yang berarti bahwa persentase kata pada topik 7 memuat sebesar 11,9% dari keseluruhan topik dalam ulasan.

5.5. Hasil Analisis

Berdasarkan model LDA dan visulisasi kata pada masing-masing model topik pada sub bab 5.4 maka dilakukan peringkasan hasil analisis LDA yang telah

dilakukan sehingga dapat diketahui topik apa saja yang banyak dibicarakan oleh pengguna BSI *Mobile* secara ringkas. Ringkasan hasil analisis LDA dapat dilihat pada **Tabel 5.19**

Tabel 5.19 Hasil Analisis LDA.

Topik	Model
# 1 transaksi mudah dan aman disertai fitur yang lengkap	0,273*"mudah" + 0,105*"transaksi" + 0,062*"nyaman" + 0,047*"fitur_lengkap" + 0,043*"pakai" + 0,039*"aman" + 0,039*"guna" + 0,033*"fitur" + 0,031*"lengkap" + 0,020*"gampang"
# 2 aplikasi tidak bisa diperbarui maupun dihapus dan sering *maintenance*	0,319*"bantu" + 0,161*"tidak_bisa" + 0,111*"keren" + 0,054*"cepat" + 0,032*"transaksi" + 0,024*"baru" + 0,018*"praktis" + 0,014*"rating" + 0,013*"hapus" + 0,013*"maintenance"
# 3 adanya pembaruan aplikasi membuat pengguna mengalami kesusahan ketika daftar dan aktivasi	0,445*"bagus" + 0,315*"mantap" + 0,089*"baik" + 0,034*"baru" + 0,022*"lumayan" + 0,013*"susah" + 0,007*"percaya" + 0,004*"daftar" + 0,003*"mutar" + 0,003*"aktivasi"
# 4 fitur isi_ulang ribet dan aktivasi lambat	0,067*"ribet" + 0,043*"tidak_ada" + 0,035*"lambat" + 0,027*"isi_ulang" + 0,025*"urus" + 0,023*"aktivasi" + 0,023*"perbaiki" + 0,023*"buruk" + 0,021*"maju" + 0,020*"jelek"
# 5 layanan customer service menghabiskan pulsa	0,065*"layan" + 0,054*"berkah" + 0,035*"senang" + 0,029*"simpel" + 0,025*"amanah" + 0,022*"customer_service" + 0,019*"takjub" + 0,019*"habis_pulsa" + 0,018*"tidak_perlu" + 0,017*"depan"
# 6 pengguna bersyukur atas kebermanfaatan aplikasi	0,179*"syukur" + 0,101*"puas" + 0,089*"manfaat" + 0,062*"lancar" + 0,031*"eror" + 0,022*"transfer" + 0,020*"buka_rekening" + 0,018*"buka" + 0,015*"mudah" + 0,014*"layan"
# 7 sulit meminta kode aktivasi	0,077*"suka" + 0,076*"kode_aktivasi" + 0,066*"kerja" + 0,064*"tingkat" + 0,039*"aktivasi_ulang" + 0,038*"sukses" + 0,036*"bagus" + 0,028*"kecewa" + 0,026*"sulit" + 0,024*"minta_kode_aktivasi"

BAB VI
PENUTUP

6.1. Kesimpulan

Berdasarkan hasil dan pembahasan pada bab 5, maka diberikan kesimpulan sebagai berikut:

1. Pemodelan topik ulasan pengguna BSI *Mobile* menggunakan metode *Latent Dirichlet Allocation* (LDA) didapatkan hasil bahwa jumlah model topik terbaik sebanyak 7 model topik dengan *coherence score* sebesar 0,4350406246. Bahasan pada 7 topik tersebut adalah sebagai berikut:
   1) Topik 1 membahas terkait transaksi mudah dan aman disertai fitur yang lengkap.
   2) Topik 2 membahas terkait aplikasi tidak bisa diperbarui maupun dihapus dan sering *maintenance*.
   3) Topik 3 membahas terkait adanya pembaruan aplikasi membuat pengguna mengalami kesusahan ketika daftar dan aktivasi.
   4) Topik 4 membahas terkait fitur isi_ulang ribet dan aktivasi lambat.
   5) Topik 5 membahas terkait layanan customer service menghabiskan pulsa.
   6) Topik 6 membahas terkait pengguna bersyukur atas kebermanfaatan aplikasi.
   7) Topik 7 membahas terkait sulit meminta kode aktivasi.

6.2. Saran

Saran yang dapat diberikan berdasarkan hasil penelitian ini adalah sebagai berikut:

1. Pada penelitian ini data yang digunakan ± 15% dari total ulasan pengguna BSI *Mobile* di *Google Play Store* sehingga dapat dilakukan penambahan data agar hasil lebih menggambarkan seluruh ulasan yang ada.
2. Hasil penelitian ini menunjukkan bahwa topik 7 mempunyai nilai PC1 yang tidak sesuai dengan letak kuadran pada visualisasi *intertopic distance maps*. Hal tersebut dikarenakan adanya tahapan penggabungan kata majemuk secara manual sebelum masuk ke tahapan N-gram (pada penelitian ini menggunakan bigram dan trigram). Penelitian selanjutnya dapat melakukan evaluasi terhadap

algoritma LDA yang menggunakan tahap penggabungan kata majemuk secara manual.

3. Penelitian selanjutnya dapat melakukan pengembangan metode LDA dengan metode yang lain.

DAFTAR PUSTAKA

Abdurrazzaq, M. A. (2023). Analisis Ulasan Aplikasi MyPertamina Menggunakan Topic Modeling dengan Latent Dirichlet Allocation. *Kalbiscentia, Jurnal Sains dan Teknologi*, 1-6.

Alfanzar, A. I., Khalid, & Rozas, I. S. (2020). Topic Modelling Skripsi Menggunakan Metode Latent Dirichlet Allocation. *Jurnal Sistem Informasi*, 7-13.

Alita, D., & Rahman, A. (2020). Pendeteksian Sarkasme pada Proses Analisis Sentimen Menggunakan Random Forest Classifier. *Jurnal Komputasi*, 50-58.

Anggraeni, D. T. (2019). Forecasting Harga Saham Menggunakan Metode Simple Moving Average Dan Web Scrapping. *Jurnal Ilmiah MATRIK*, 234-241.

Arni, U. D. (2018). Apa Itu Text Mining ? *Garuda Cyber Indonesia*.

Astuti, A. R., & Cahyono, N. (2023). Analisis Topic Modelling Persepsi Pengguna Internet Menggunakan Metode Latent Dirichlet Allocation. *Indonesian Journal of Computer Science (IJCS)*, 326-334.

Blad, J., & Svensson, K. (2020). *Exploring NMF and LDA Topic Models of Swedish News Articles.* Uppsala: UPPSALA UNIVERSITET.

Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent Dirichlet Allocation. *Journal of Machine Learning Research 3*, 993-1022.

Campbell, J. C., Hindle, A., & Stroulia, E. (2015). Latent Dirichlet Allocation: Extracting Topics from Software Engineering Data. *Elsevier*, 139-159.

Damayanti, P., Purwitasari, D., & Suciati, N. (2018). Eliminasi Data Non-Topic menggunakan Pemodelan Topik untuk Peringkasan Otomatis Data Tweet dengan Konteks Covid-19. *Jurnal Teknologi Informasi dan Ilmu Komputer (JTIIK)* , 199-208.

Daudshah, F., & Yetti, F. (2022). Faktor-Faktor Yang Mempengaruhi Intensi Nasabah Berinfak Pada BSI Mobile. *VEMAR (Veteran Economics, Management, & Accounting Review)*, 74-87.

Devy, N. A., & Fikriyah, K. (2022). Pengaruh Promosi dan Kualitas Layanan terhadap Keputusan Nasabah Menggunakan Layanan BSI Mobile pada Bank Syariah Indonesia KC Surabaya Dipenogoro. *Jurnal Ilmu Komputer, Ekonomi dan Manajemen (JIKEM)*, 1386-1398.

Febrianta, M. Y., Widiyanesti, S., & Ramadhan, S. R. (2021). Analisis Ulasan Indie Video Game Lokal pada Steam Menggunakan Analisis Sentimen dan Pemodelan Topik Berbasis Latent Dirichlet Allocation. *Journal of Animation & Games Studies*, 117-144.

Febrianti, D., Hidayah, S. A., Abdullah, & Lawita, N. F. (2021). Penerapan Basis Data pada Perusahaan Perbankan (Studi Kasus Penerapan Mobile Banking pada Bank Syariah Indonesia). *Jurnal Pendidikan Tambusai*, 3686-3693.

Feldman, R., & Sanger, J. (2007). *The Text Mining Handbook.* New York: the United States of America by Cambridge University Press.

Fernanda, J. W. (2021). Pemodelan Persepsi Pembelajaran Online Menggunakan Latent Dirichlet Allocation. *Jurnal Statistika Universitas Muhammadiyah Semarang*, 79-85.

Google Play. (2023, January 26). Retrieved January 27, 2023, from Google Play: https://play.google.com/store/apps/details?id=com.bsm.activity2

Hakim, B. (2021). Analisa Sentimen Data Text Preprocessing Pada Data Mining Dengan Menggunakan Machine Learning. *Journal of Business and Audit Information Systems*, 16-22.

Harishamzah. (2020, April 13). *Perbandingan Perhitungan Bobot TF-IDF secara Manual dan Menggunakan Python*. Retrieved from Medium: https://medium.com/bisa-ai/perbandingan-perhitungan-bobot-tf-idf-secara-manual-dan-menggunakan-python-377392a165c6

Hasanah, U., Fitriani, N., & Hana, K. F. (2022). Analisis Penerapan Sharia Compliance Pada Produk Pembiayaan Bsi Kur Mikro Di Bank Syariah Indonesia. *Jurnal Manajemen dan Perbankan Syariah*, 2-12.

Hendriawan, R. Y. (2021). Analisis Sentimen Berbasis Aspek Pada Restoran dengan Metode Modified K-Nearest Neighbor (MKNN). *elibrary UNIKOM*, 1-118.

Hoffman, M. D., Blei, D. M., & Bach, F. (2010). Online Learning for Latent Dirichlet Allocation. *NeurIPS Proceedings*, 1-9.

Ignatow, G., & Mihalcea, R. (2018). *An Introduction to Text Mining: Research Design, Data Collection, and Analysis.* London: SAGE Publications.

Indah, R. N. (2022, February 20). *Apa itu G20 dan Manfaatnya untuk Indonesia*. Retrieved November 21, 2022, from Kementerian Keuangan Republik Indonesia: https://www.djkn.kemenkeu.go.id/kpknl-singkawang/baca-artikel/14747/Apa-itu-G20-dan-Manfaatnya-untuk-Indonesia.html

Irmayati, H. (2018). Analisis Algoritma Fuzzy Logic dalam Pengklasifikasian Tugas Akhir. *Komputika: Jurnal Sistem Komputer*, 71-77.

Kabir, A. I., Ahmed, K., & Karim, R. (2020). Word Cloud and Sentiment Analysis of Amazon Earphones Reviews with R Programming Language. *Informatica Economică*, 55-71.

Kannitha, D. Z., Mustafid, & Kartikasari, P. (2022). Pemodelan Topik Pada Keluhan Pelanggan Menggunakan Algoritma Latent Dirichlet Allocation Dalammedia Sosial Twitter. *Jurnal Gaussian,*, 266-277.

Karmayasa, O., & Mahendra, I. B. (2012). Implementasi Vector Space Model Dan Beberapa Notasi Metode Term Frequency Inverse Document Frequency (Tf-Idf) Pada Sistem Temu Kembali Informasi. *(JELIKU) Jurnal Elektronik Ilmu Komputer Udayana*, 1.

Kementerian Koordinator Bidang Perekonomian Republik Indonesia. (2021, November 19). *Presidensi Indonesia G20 - 2022, Momentum Branding Indonesia di Dunia Internasional*. Retrieved November 21, 2022, from Kementerian Koordinator Bidang Perekonomian Republik Indonesia: https://www.ekon.go.id/publikasi/detail/3469/presidensi-indonesia-g20-2022-momentum-branding-indonesia-di-dunia-internasional

Khomsah, S., & Aribowo, A. S. (2020). Model Text-Preprocessing Komentar Youtube Dalam Bahasa Indonesia. *Jurnal Resti*, 648-654.

Kwartler, T. (2017). *Text Mining in Practice with R.* Hoboken: John Wiley & Sons Ltd.

Lestari, M. A., & Keumala, D. (2022). Pengaturan Restrukturisasi Pembiayaan Di Bank Syariah Indonesia Selama Masa Pandemi Covid -19. *Reformasi Hukum Trisakti*, 145-154.

Mahmud, M. B. (2023). Analisis Sentimen Aplikasi Bsi Mobile Pada Ulasan Google Play Menggunakan Algoritma Naive Bayes. *Repository UIN Sunan Kalijaga Yogyakarta*.

Manurung, D. D., Matondang, N. H., & Prasvita, D. S. (2022). Analisis Sentimen pada Ulasan Aplikasi Jakarta Terkini (JAKI) di Google Play Store Menggunakan Metode Support Vector Machine. *SeminarNasionalMahasiswaIlmuKomputerdanAplikasinya(SENAMIKA)*, 158-167.

Matira, Y., Junaidi, & Setiawan, I. (2023). Pemodelan Topik pada Judul Berita OnlineDetikcom Menggunakan Latent Dirichlet Allocation. *Journal of Statistics and Its Application*, 53-63.

Miller, T. A., Dligach, D., & Savova, G. K. (2016). Unsupervised Document Classification with Informed Topic Models. *In Proceedings of the 15th Workshop on Biomedical Natural Language Processing*, 83-91.

Mutiarasari, K. A. (2022, November 14). *Kapan Indonesia Masuk G20? Ini Awal Mulanya*. Retrieved November 21, 2022, from detiknews: https://news.detik.com/berita/d-6404955/kapan-indonesia-masuk-g20-ini-awal-mulanya

Nasution, K. H., Widodo, & Adhi, B. P. (2021). Sistem Deteksi Topik Politik Pada Twitter Menggunakan Algoritma Latent Dirichlet Allocation. *Jurnal Pinter*, 34-42.

Nugraha, M. A., & Mungaran, L. C. (2021). Pemodelan Topik Berita pada Portal Berita Online Berbahasa Indonesia Menggunakan Latent Dirichlet Allocation (LDA). *Jikstik (Jurnal Ilmiah Komputasi STI&K)*, 173-180.

Nugroho, I. Y., & M.Pudjihardjo. (2022). Pengaruh Persepsi Kemudahan, Persepsi Kegunaan, Ketersediaan Fitur Dan Literasi Keuangan Terhadap Preferensi Konsumen Menggunakan Bsi Mobile. *Islamic Economics And Finance In Focus (IEFF)*, 135-147.

Oktafiandi, H. (2023). Implementasi LDA untuk Pengelompokan Topik Twitter Bertagar #Mypertamina. *Jurnal Ekonomi Dan Teknik Informatika*, 10-16.

Parasati, W., Bachtiar, F. A., & Setiawan, N. Y. (2020). Analisis Sentimen Berbasis Aspek pada Ulasan Pelanggan Restoran Bakso President Malang dengan Metode Naïve Bayes Classifier. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, 1090-1099.

Prakerti, A. I., Claresta, A. F., Kafif Ibrahim, M. R., & Rakhmawati, N. A. (2020). Model Latent Dirichlet Allocation Pada Perilaku Siswa Menggunakan Media Pembelajaran Daring. *Information Management For Educators And Professionals* , 35-44.

Prasastio, F. R., Heriyanto, & Kaswidjanti, W. (2022). Sentiment Analysis of the Covid-19 Vaccine Using the Naive Bayes Algorithm and Levenshtein Distance Word Correction. *Telematika: Jurnal Informatika dan Teknologi Informasi*, 91-104.

Putra, A. D., & Juanita, S. (2021). Analisis Sentimen Pada Ulasan Pengguna Aplikasi Bibit Dan Bareksa Dengan Algoritma KNN. *Jurnal Teknik Informatika dan Sistem Informasi*, 636-646.

Qomariyah, S., Irawan, N., & Fithriasari, K. (2019). Topic modeling Twitter data using Latent Dirichlet Allocation and Latent Semantic Analysis. *AIP Conference Proceedings 2194*, 1-7.

Reni, & Vanomy, A. E. (2023). The Role of E-Wallet's Actual Consumer in Indonesia with Policy Perspective and Consumer Perception Using Latent Dirichlet Allocation (LDA) Method. *PROFIT: Jurnal Administrasi Bisnis*, 42-54.

Ridhwanullah, D. (2022). Pemodelan Topik pada Cuitan tentang Penyakit Tropis di Indonesia dengan Metode Latent Dirichlet Allocation. *DSpace UII*.

Roder, M., Both, A., & Hinneburg, A. (2015). Exploring the Space of Topic Coherence Measures. *WSDM '15: Proceedings of the Eighth ACM International Conference on Web Search and Data Mining*, 399–408.

Salam, A., Zeniarja, J., & Khasanah, R. S. (2018). Analisis Sentimen Data Komentar Sosial Media Facebook Dengan K-Nearest Neighbor (Studi Kasus Pada Akun Jasa Ekspedisi Barang J&T Ekspress Indonesia). *SINTAK*, 480-486.

Saputro, W. (2022, October 19). *G20 Dorong Akses Digital: Bikin Transaksi Keuangan Mudah, Ekonomi Merekah*. Retrieved November 21, 2022, from kumparanBISNIS: https://kumparan.com/kumparanbisnis/g20-dorong-

akses-digital-bikin-transaksi-keuangan-mudah-ekonomi-merekah-1z4zClFOTcA/full

Setijohatmo, U. T., Rachmat, S., Susilawati, T., & Rahman, Y. (2020). Analisis Metoda Latent Dirichlet Allocation untuk Klasifikasi Dokumen Laporan Tugas Akhir Berdasarkan Pemodelan Topik. *Prosiding The 11th Industrial Research Workshop and National Seminar*, 402-408.

Simatupang, M. P., & Utomo, D. P. (2019). Analisis Testimonial dengan Menggunakan Algoritma Text Mining dan Term Frequency-Inverse Document Frequence (TF-IDF) pada Toko Allmeeart. *KOMIK (Konferensi Nasional Teknologi Informasi dan Komputer)*, 808-814.

Stevens, K., Kegelmeyer, P., Andrzejewski, D., & Buttler, D. (2012). Exploring Topic Coherence over many models and many topics. *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, s 952–961.

Steyvers, M., & Griffiths, T. (2006). *Probabilistic Topic Models.* Irvine: University of California.

Sulehu, M., Juhar, Rimalia, W., & Iskandar, A. (2019). Implementasi Metode Term Frequency-Inverse Document Frequency-Class Frequency untuk Peringkasan Berita Online. *Celebes Engineering Journal*, 54-61.

Suparyati, Utami, E., & Fathurahman, A. (2022). Pengamatan Tren Ulasan Hotel Menggunakan Pemodelan Topik Berbasis Latent Dirichlet Allocation. *Journal of Applied Informatics and Computing (JAIC)*, 71~77.

Vayansky, I., & Kumar, S. A. (2020). A review of topic modeling methods. *Information Systems*, 1-28.

Widyaningsih, M., S, F. R., & R, A. D. (2022). Pengaruh Persepsi Kegunaan, Kemudahan dan Pengetahuan Informasi Terhadap Keputusan Penggunaan Aplikasi BSI Mobile (Studi Kasus Nasabah BSI Kabupaten Sukoharjo). *Rizquna : Jurnal Hukum dan Ekonomi Syariah*, 1-24.

Wirasakti, L. A., Permadi, R., Hartanto, A. D., & Hartatik. (2020). Pembuatan Kata Kunci Otomatis Dalam Artikel Dengan Pemodelan Topik. *Jurnal Media Informatika Budidarma*, 27-31.

Yaman, A., Sartono, B., & M. Soleh, A. (2021). Pemodelan topik pada dokumen paten terkait pupuk di Indonesia berbasis Latent Dirichlet Allocation. *Berkala Ilmu Perpustakaan dan Informasi*, 168-180.

Yerzi, F. S., & Sibaroni, Y. (2021). Analisis Sentimen Terhadap Kebijakan Pemerintah Dalam Menangani Covid-19 Dengan Pendekatan Lexicon Based. *e-Proceeding of Engineering*, 11354-11366.

Zain, N. M., & Isam, H. (2019). Emoji dan ekspresi emosi dalam kalangan komuniti siber. *PENDETA Journal of Malay Language, Education and Literature*, 12-23.

LAMPIRAN

Lampiran 1 Data Ulasan Pengguna BSI *Mobile* di *Google Play Store*

	A	B	C
1	userName	at	content
2	RAKA PERMAN	21/11/2022 05:4	kok aplikasinya berhenti trs ya?...masalahnya apa?...signal ama penyimpanan aman trs masalahnya apa?
3	Fauzan Ahmad	21/11/2022 05:1	Aplikasinya setelah diapdet makin keren
4	Rizka Aristiani	21/11/2022 04:3	Aplikasinya bagus dan sangat membantu, selama memakainya tidak pernah ada kendala dan error. Jadi makin nyaman pakai aplikasi BSI sekarang. Mantebbb
5	Nadila Fitriani	21/11/2022 04:3	Seneng deh pakai aplikasinya kalau gini, selain memudahkan user bertransaksi, tampilan visualnya juga sangat menarik sehingga memanjakan mata. Semoga konsisten te
6	Ami Oktaviani	21/11/2022 04:2	Apps nya mudah banget dipakai
7	della kasagai	21/11/2022 03:5	Wah ternyata setelah update terbaru, apk BSI tambah keren. Maju terus BSI ..
8	Afin N Ikhsan	21/11/2022 03:4	Proses aktivasi yg tidak valid, no rek blm terbentuk, namun di minta memasukkan no rek. Pelayanan cs jg kurang baik, jawaban hanya template. Perbaiki lagi. Terimakasih
9	Terapi stroke	21/11/2022 03:4	Ok
10	Agus Muzlim	21/11/2022 03:4	Saya kasi B3 dulu, sebab untuk menu topup e-walet masih perlu diperbaiki terutama untuk gopay harusnya ada 2 jenis, yg pertama gopay untuk transaksi dan yang kedua g
11	Irawati Hamdan	21/11/2022 03:2	Mantap BSI 👍👍👍
12	JNE Tangerang	21/11/2022 01:2	ajib
13	saiful bahri	20/11/2022 22:5	Banyak kemudahan
14	Iqbal dhiafakhri	20/11/2022 17:0	Kenapa aplikasinya tidak bisa di uninstal, tidak bikin puas malah jadi masalah ke hp
15	safana nyanyak	20/11/2022 12:2	Good
16	KRC A	20/11/2022 12:0	Gak di update gak bisa dibuka, di update ngebugg. Tolong perbaikannya
17	Thambrin Thamt	20/11/2022 11:2	Aplikasi nya susah dibuka
18	Ken Arok	20/11/2022 11:1	Kenapa mau masuk login bsi mobile aja sulit bget,and mau ganti pin aja ga bisa,ribet bget amplikasinnya suka kuluar sendiri

Lampiran 2 *Scrapping*

```
!pip install google-play-scraper#https://pypi.org/project/google-play-scraper/

from google_play_scraper import app
import pandas as pd
import numpy as np
```

```
#Scrape desired number of reviews
from google_play_scraper import Sort, reviews
result, continuation_token = reviews(
'com.bsm.activity2',
lang='id', # defaults to 'en'
country='id', # defaults to 'us'
sort=Sort.NEWEST, # defaults to Sort.MOST_RELEVANT you can use Sort.NEWEST to get newst reviews
count=20000, # defaults to 100
filter_score_with=None # defaults to None(means all score) Use 1 or 2 or 3 or 4 or 5 to select certain score
)
```

```
dataskripsi = pd.DataFrame(np.array(result),columns=['review'])
dataskripsi = dataskripsi.join(pd.DataFrame(dataskripsi.pop('review').tolist()))
dataskripsi.head()
```

```
len(dataskripsi.index) #count the number of data we got
dataskripsi[['userName', 'at','content', 'score']].head() #preview userName, rating, date-time, and reviews only
data_mentahskripsi = dataskripsi[['userName', 'at','content', 'score']]
data_sortirskripsi= data_mentahskripsi.sort_values(by='at', ascending=False)
```

```
#Sort by Newst, change to True if you want to sort by Oldest.
data_sortirskripsi.head()
data_scrapskripsi = data_sortirskripsi[['userName', 'at','content', 'score']] #get userName, rating, date-time, and reviews only
data_scrapskripsi.head()
data_scrapskripsi.to_csv("skripsi.csv", index = False) #Save the file as CSV , to download: click the folder icon on the left. the csv file should be
```

Lampiran 3 *Preprocessing*

```
#membuka file
import pandas as pd
data = pd.read_excel('dataskripbaruteksaja.xlsx')
data.head()
```

```
import pandas as pd
import numpy as np
from nltk.corpus import stopwords
import re
import nltk
nltk.download('stopwords')
nltk.download('wordnet')
from nltk.stem import PorterStemmer
from nltk.stem import LancasterStemmer
```

```
import re#https://www.webhozz.com/code/python-regex/
```

```
pip install Sastrawi
```

```
# Removing Punctuation
data['text'] = data['text'].str.replace('[^\w\s]','')
data['text'].head()
```

```
#remove angka
data['text']=data['text'].str.replace(r'[\d+]', '')
data['text'].head()
```

```
import string
```

```
data['text']=data['text'].astype(str)
```

```
#lowercase / case folding -> https://adityarizki.net/belajarpython-8-operasi-case-folding-data-teks-menggunakan-library-nltk/
def clean_lower(lwr):
    text = lwr.lower() # lowercase text
    return text
# Buat kolom tambahan untuk data description yang telah dicasefolding
data['text'] = data['text'].apply(clean_lower)
casefolding=pd.DataFrame(data['text'])
casefolding
```

```
casefolding.to_csv("cleaningawal.csv", index = False)
```

spelling normalization (typo dan slang)

https://www.semanticscholar.org/paper/Spelling-Normalization-of-English-Student-Writings-Hong/bb9817276c89a0ad589c341c03492e37f091bb2d

```
import pandas as pd
data = pd.read_csv('cleaningawal.csv')
data.head()
```

```
data.head()
```

```
import re, string
import pandas as pd
```

```
# membaca file normalisasi
df_norm = pd.read_csv("kamus.txt")
# membuat kamus normalisasi (dictionary)
df_kamus = {}
for dt in df_norm.itertuples():
  df_kamus[dt[1]] = dt[2]
```

```
def preprocess(row):
  row['text'] = ' '.join([df_kamus[a] if a in df_kamus else a for a in row['text'].split()])
  return row
```

```
datab = data.apply(preprocess, axis=1)
datab.head()
```

```
datab.to_csv("typoslang.csv", index = False)
```

remove stopword (kt tidak penting)

https://rahmadya.com/2019/04/24/stopword-berbahasa-indonesia/

```
[1] import pandas as pd
    data = pd.read_csv("hasil_typoslang.csv")
    data.head()
```

```
!pip install Sastrawi
```

```
[3] from Sastrawi.StopWordRemover.StopWordRemoverFactory import StopWordRemoverFactory

    factory = StopWordRemoverFactory()
    stopwords = factory.get_stop_words()
    print(stopwords)
```

```
[4] # import StopWordRemoverFactory class
    from Sastrawi.StopWordRemover.StopWordRemoverFactory import StopWordRemoverFactory
    factory = StopWordRemoverFactory()
    stopword = factory.get_stop_words()
```

```
import requests
def stopwords():
    r = requests.get("https://raw.githubusercontent.com/masdevid/ID-Stopwords/master/id.stopwords.02.01.2016.txt").text
    data = []
    for x in r.split("\n"):
        data.append(x)
    return data
```

```
[6] stopwords()
```

```
# Import Stopword Factory class
from Sastrawi.StopWordRemover.StopWordRemoverFactory import StopWordRemoverFactory
```

```
[8] #Create factory
    factory = StopWordRemoverFactory()
    more_stopword = ['haha','moga','semoga','mohon','aplikasi','aplikasinya','gini','betulbetul','temanteman','mudahmudahan','benarbenar','memperolokolok
    stopwordplus = factory.get_stop_words()+stopwords()+more_stopword
    data['text'] = data['text'].apply(lambda x: " ".join(x for x in x.split() if x not in stopwordplus))
    data['text']
```

```
stopwordplus
```

```
text = data['text']
text_list = [i.split() for i in text]
print(len(text_list))
```

```
[11] import re
     # Function to Tokenize words
     def tokenize(text):
       tokens = re.split('\W+', text) #W+ means that either a word character (A-Za-z0-9_) or a dash (-) can go there.
       return tokens
     data['text'] = data['text'].apply(lambda x: tokenize(x.lower()))
     data.head()
```

```
data.to_csv("stopwordstoken.csv", sep=',')
```

```
# Stemming -> https://medium.com/@93Kryptonian/stemming-with-sastrawi-877cc40a37ad
# import StemmerFactory class
from Sastrawi.Stemmer.StemmerFactory import StemmerFactory
```

```
[13] # create stemmer
     factory = StemmerFactory()
     stemmer = factory.create_stemmer()
```

```
[14] # stemming process
     def kata_stem(teks):
       stem_teks = " ".join([stemmer.stem(i) for i in teks])
       return stem_teks
     data['text'] = data['text'].apply(lambda x: kata_stem(x))
     data['text'].head()
```

```
data.head()
```

Lampiran 4 Pemodelan Topik LDA

▾ TF IDF

▾ jika ada masalah di get_feature_names() ganti ke get_feature_names_out()

######## https://stackoverflow.com/questions/70640923/countvectorizer-object-has-no-attribute-get-feature-names-out

```
[ ] import pandas as pd
    data = pd.read_csv('bsimobile.csv')
    data.head()
```

```
from sklearn.feature_extraction.text import CountVectorizer
count_vectorizer = CountVectorizer(encoding='latin-1', ngram_range=(1, 1), tokenizer=None, analyzer = 'word', stop_words = stopwordplus)
countvec = count_vectorizer.fit_transform(data.text).toarray()
countvec
```

```
countvec2 = pd.DataFrame(countvec)
countvec2
```

```
kata_kata = count_vectorizer.get_feature_names_out()
countvec3 = pd.DataFrame(countvec, columns=kata_kata)
countvec3
```

```
from sklearn.feature_extraction.text import TfidfTransformer
transformer = TfidfTransformer(norm=None, use_idf=True, smooth_idf=False, sublinear_tf=False)
tfidf = transformer.fit_transform(countvec)
tfidf
```

```
tfidf1 = tfidf.toarray()
tfidf1
```

```
kata_kata2 = count_vectorizer.get_feature_names_out()
df1 = pd.DataFrame(tfidf1, columns=kata_kata2)
df1
```

```
[ ] df1.to_csv("tfidf.csv")
```

Pemodelan Topik

```
## sintaks pemodelan topik DSI
# https://colab.research.google.com/drive/16basitGM1BYDrcmDyFKAxFu997JyVqOT?usp=sharing#scrollTo=L-t6e6u9Do5y
```

```
#Read file as panda dataframe
df = pd.read_csv('bsimobile.csv')#create data frame

text = df['text']
text_list = []
for i in range(len(text)) :
    bbb = text[i].replace('[', '')
    bbb = bbb.replace(']', '')
    bbb = bbb.replace("'", "")
    bbb = bbb.replace(",", "")
    temp = []
    for j in bbb.split() :
        temp.append(j)
    text_list.append(temp)
```

```
print(len(text_list))
```

```
df.head()
```

```
pip install -U gensim
```

```
#Create Bigram & Trigram Models
from gensim.models import Phrases
# Add bigrams and trigrams to docs,minimum count 10 means only that appear 10 times or more.
bigram = Phrases(text_list, min_count=10)
trigram = Phrases(bigram[text_list])

for idx in range(len(text_list)):
    for token in bigram[text_list[idx]]:
        if '_' in token:
            # Token is a bigram, add to document.
            text_list[idx].append(token)
    for token in trigram[text_list[idx]]:
        if '_' in token:
            # Token is a bigram, add to document.
            text_list[idx].append(token)
```

```
from gensim import corpora, models
# Create a dictionary representation of the documents.
dictionary = corpora.Dictionary(text_list)

dictionary.filter_extremes(no_below=5, no_above=0.2)
#no_below (int, optional) - Keep tokens which are contained in at least no_below documents.
#no_above (float, optional) - Keep tokens which are contained in no more than no_above documents (fraction of total corpus size, not an absolute numb
```

```
# https://radimrehurek.com/gensim/tut1.html
#build corpus
doc_term_matrix = [dictionary.doc2bow(doc) for doc in text_list]

# the function doc2bow converts documents (a list of words) into the bag-of-words format
print(len(doc_term_matrix))
print(doc_term_matrix[100])

tfidf = models.TfidfModel(doc_term_matrix) #build TF-IDF model
corpus_tfidf = tfidf[doc_term_matrix]
```

```
[ ] from gensim.models.coherencemodel import CoherenceModel
from gensim.models.ldamodel import LdaModel
from gensim.corpora.dictionary import Dictionary
from numpy import array
#function to compute coherence values
def compute_coherence_values(dictionary, corpus, texts, limit, start, step):
    coherence_values = []
    model_list = []
    for num_topics in range(start, limit, step):
        model = LdaModel(corpus=corpus, id2word=dictionary, num_topics=num_topics, iterations=100)
        model_list.append(model)
        coherencemodel = CoherenceModel(model=model, texts=texts, dictionary=dictionary, coherence='c_v')
        coherence_values.append(coherencemodel.get_coherence())

    return model_list, coherence_values
```

```
[ ] start=1
    limit=11
    step=1
    model_list, coherence_values = compute_coherence_values(dictionary, corpus=corpus_tfidf,
                                                             texts=text_list, start=start, limit=limit, step=step)
    #show graphs
    import matplotlib.pyplot as plt
    x = range(start, limit, step)
    plt.plot(x, coherence_values)
    plt.xlabel("Num Topics")
    plt.ylabel("Coherence score")
    plt.legend(("coherence_values"), loc='best')
    plt.show()
```

```
# Print the coherence scores
for m, cv in zip(x, coherence_values):
    print("Num Topics =", m, " has Coherence Value of", round(cv, 10))#tergantung pd coherence values
```

```
from pprint import pprint
model = LdaModel(corpus=corpus_tfidf, id2word=dictionary,
random_state=0, num_topics=7)
pprint(model.print_topics())
```

```
#referensi mengganti jumlah kata mjd 30 => https://stackoverflow.com/questions/55815556/how-to-change-the-default-number-
model = LdaModel(corpus=corpus_tfidf, id2word=dictionary,random_state=0, num_topics=7)

for idx, topic in model.print_topics(7, 30):
    print('Topic: {} Word: {}'.format(idx, topic))
```

```
!pip install pyLDAvis==2.1.2
```

```
import gensim
import pyLDAvis.gensim;pyLDAvis.enable_notebook()

data = pyLDAvis.gensim.prepare(model, corpus_tfidf, dictionary)
print(data)
pyLDAvis.save_html(data, 'lda-bsimobile.html')
```

```
[ ] import matplotlib.pyplot as plt
    from wordcloud import WordCloud as wd

    for t in range(model.num_topics):
        plt.figure(figsize=(7,6))
        plt.imshow(wd(max_font_size=50, min_font_size=6).fit_words(dict(model.show_topic(t, 200))))
        plt.axis("off")
        plt.title("Topic #" + str(t))
        plt.savefig("wcld-topic-#"+str(t)+".png", facecolor='k', bbox_inches='tight')

    plt.show()
```

```
import pyLDAvis
import pyLDAvis.gensim
vis = pyLDAvis.gensim.prepare(model, corpus_tfidf, dictionary)
pyLDAvis.enable_notebook()
pyLDAvis.display(vis)
```