

**PEMODELAN TOPIK TERKAIT ULASAN *VIDEO GAME* DENGAN GENRE  
*BATTLE ROYALE* MENGGUNAKAN METODE *BERTOPIC* DENGAN FITUR  
*GUIDED TOPIC MODELLING***

**SKRIPSI**



**Disusun Oleh :**

**Gibran Giffari Priyatna**

**11180940000073**

**PROGRAM STUDI MATEMATIKA**

**FAKULTAS SAINS DAN TEKNOLOGI**

**UNIVERSITAS ISLAM NEGERI SYARIF HIDAYATULLAH**

**JAKARTA**

**2022 M/1444 H**

**PEMODELAN TOPIK TERKAIT ULASAN *VIDEO GAME* DENGAN  
GENRE *BATTLE ROYALE* MENGGUNAKAN METODE *BERTOPIC*  
DENGAN FITUR *GUIDED TOPIC MODELLING***

**SKRIPSI**

**Diajukan untuk Memenuhi Salah Satu Persyaratan  
dalam Memperoleh Gelar Sarjana Matematika (S.Mat)**

**Disusun Oleh :**

**Gibran Giffari Priyatna**

**11180940000073**

**PROGRAM STUDI MATEMATIKA**

**FAKULTAS SAINS DAN TEKNOLOGI**


**UNIVERSITAS ISLAM NEGERI SYARIF HIDAYATULLAH**

**JAKARTA**

**2023 M/1445 H**

## **PERNYATAAN**

DENGAN INI SAYA MENYATAKAN BAHWA SKRIPSI INI BENAR-BENAR  
HASIL KARYA SENDIRI YANG BELUM PERNAH DIAJUKAN SEBAGAI  
SKRIPSI ATAU KARYA ILMIAH PADA PERGURUAN TINGGI ATAU  
LEMBAGA MANAPUN



Jakarta, 20 Januari 2023

Gibran Giffari Priyatna  
NIM. 11180940000073

## LEMBAR PENGESAHAN

Skripsi ini berjudul “Pemodelan Topik Terkait Ulasan *Video Game* dengan Genre *Battle Royale*” Menggunakan Metode *BERTopic* dengan Fitur *Guided Topic Modelling* ” yang ditulis oleh **Gibran Giffari Priyatna NIM. 11180940000073**. Skripsi ini telah diterima untuk memenuhi salah satu persyaratan dalam memperoleh gelar sarjana strata satu (S1) Program Studi Matematika.

**Menyetujui,**

Pembimbing I

Pembimbing II

Muhaza Liebenlito, M.Si

NIDN. 2003098802

M. Irvan Septiar Musti, S.Si., M.Si.

NUP. 9920113224

Penguji I

Penguji II

Taufik Edy Sutanto, M.Sc.Tech.,Ph.D

NIP. 197905302006041002

Dr. Nina Fitriyati, S.Si., M.Kom.

NIP. 197401252003122001

**Mengetahui,**

Dekan Fakultas Sains dan Teknologi

Kepala Program Studi Matematika

Ir. Nashrul Hakiem, S.Si., M.T., Ph.D.

NIP. 197106082005011005

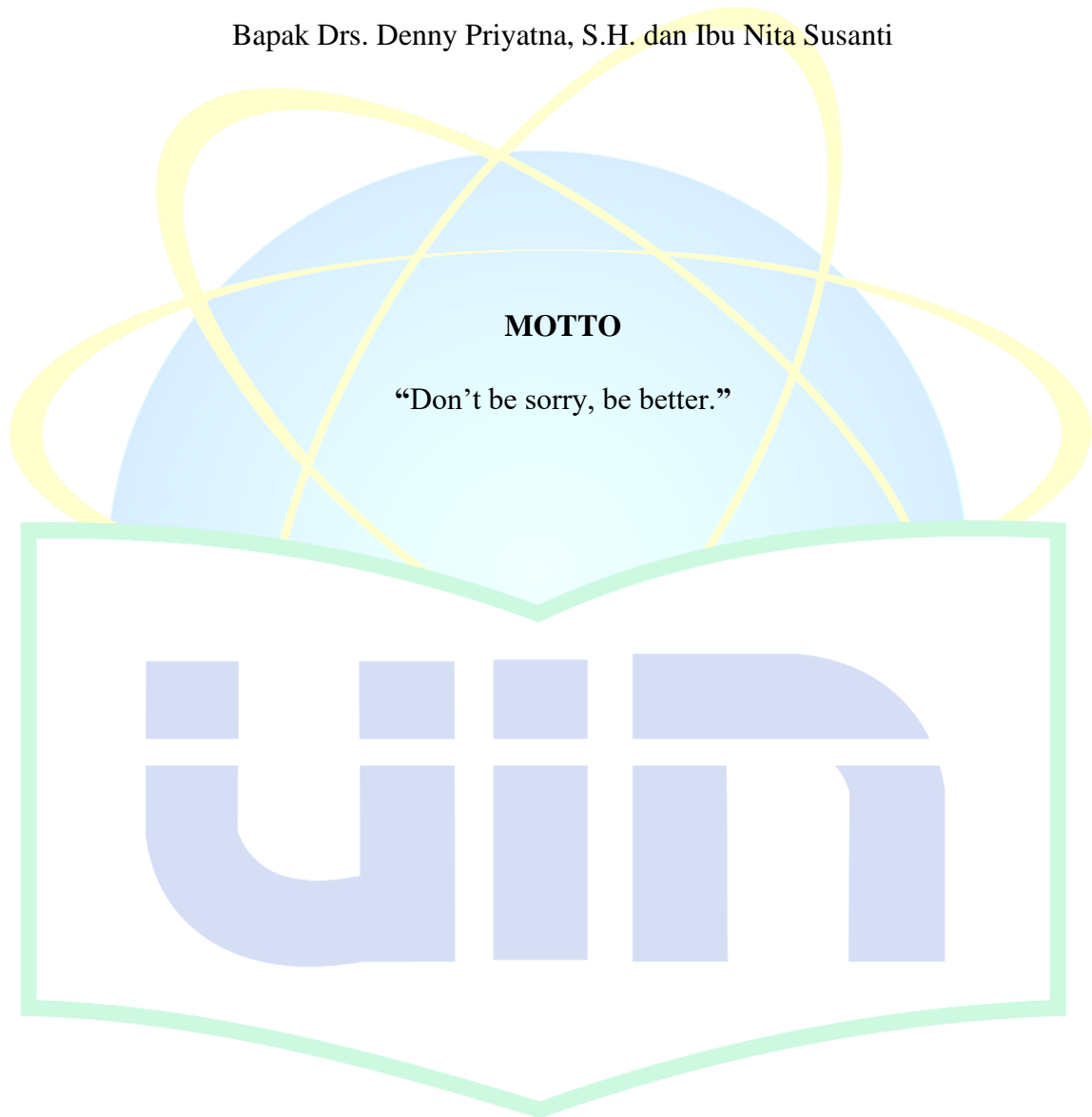
Dr. Suma'Inna, M.Si.

NIP. 197912082007012015

## **PERSEMBAHAN**

Skripsi ini saya persembahkan untuk kedua orang tua saya yang tidak pernah lelah  
untuk mencurahkan kasih sayangnya kepada saya

Bapak Drs. Denny Priyatna, S.H. dan Ibu Nita Susanti



## ABSTRAK

**Gibran Giffari Priyatna**, Pemodelan Topik Terkait Ulasan *Video Game* dengan Genre *Battle Royale* Menggunakan Metode *BERTopic* dengan Fitur *Guided Topic Modelling*.  
Dibawah bimbingan **Muhaza Liebenlito, M.Si.** dan **Muhammad Irvan Septiar Musti, M.Si.**

Pesatnya Perkembangan Industri *video game* pada beberapa tahun terakhir membuat banyak kemunculan studio pengembang *video game* baru. Untuk membuat sebuah *video game* yang menarik dan dapat dinikmati oleh banyak pemain, para studio pengembang baru harus memberikan hal yang inovatif tanpa mengabaikan masukan – masukan serta keinginan para pemain. Dikarenakan pemain suatu *video game* sangat lah banyak maka ulasan *video game* yang di terima pengembang juga sangat banyak sehingga tidak memungkinkan untuk dibaca satu persatu. Maka dari itu dalam penelitian kali ini Peneliti akan mencoba membantu para studio pengembang *video game* untuk mendapatkan masukan serta kritik dan saran pemain untuk *video game* yang telah mereka buat berdasarkan ulasan *video game* yang berjumlah sangat banyak menggunakan metode pemodelan topik untuk mendapat gambaran besar tentang apa saja yang dikeluhkan pemain dalam *game* tersebut. Metode pemodelan topik yang akan digunakan kali ini adalah *BERTopic*, yang merupakan metode pemodelan topik berbasis *sentence embedding* dengan arsitektur *neural network* sehingga metode ini dapat mengelompokkan kata sesuai konteksnya dalam suatu kalimat. Metode *BERTopic* juga dilengkapi dengan fitur *Guided Topic Modelling* yaitu metode pemodelan topik dengan menentukan topik-topik yang ingin difokuskan diawal proses. Dengan data ulasan yang ada metode *BERTopic* mampu mengekstrak topik-topik yang ada dengan baik, hal ini ditandai dengan hasil evaluasi *coherence score* yang cukup baik. Topik yang dihasilkan juga relevan dan dapat dengan mudah diinterpretasi.

**Kata Kunci :** Pemodelan Topik, *Video Game*, *Ulasan*, *BERT*, *BERTopic*, *Guided Topic Modelling*

## ABSTRACT

**Gibran Giffari Priyatna**, Topic Modeling based on Video Game Reviews with Battle Royale Genre Using the BERTopic Method with the Guided Topic Modeling Feature. Under the guidance of **Muhaza Liebenlito, M.Si.** and **Muhammad Irvan Septiar Musti, M.Si.**

The rapid development of the video game industry in recent years has led to the emergence of many new video game development studios. To make a video game that is interesting and can be enjoyed by many players, new development studios must provide innovative things without ignoring the inputs and wishes of the players. . Because there are so many video game players, the video game reviews that developers receive are also very large, so it's impossible to read one by one. Therefore in this study the researcher will try to help video game development studios to get input as well as player criticism and suggestions for the video games they have made based on the large number of video game reviews using the topic modeling method to get a big picture of what that the players complain about in the game. The topic modeling method that will be used this time is BERTopic, which is a sentence embedding-based topic modeling method with a neural network architecture so that this method can group words according to their context in a sentence. The BERTopic method is also equipped with the Guided Topic Modeling feature, which is a topic modeling method by determining the topics you want to focus on at the beginning of the process. With the existing review data, the BERTopic method is able to extract existing topics well, this is indicated by the results of a fairly good coherence score evaluation. The resulting topics are also relevant and can be easily interpreted.

**Keywords:** Topic Modeling, Video Games, Reviews, BERT, BERTopic, Guided Topic Modeling.

## KATA PENGANTAR

*Assalamu'alaikum Warrahmatullahi Wabarakatuh*

Alhamdulillah kita panjatkan puji dan syukur penulis kepada Allah SWT, karena atas karunia-Nya dapat menyelesaikan sebuah penelitian yang berjudul “Pemodelan Topik Terkait Ulasan *Video Game* dengan Genre *Battle Royale*” dengan baik dan lancar. Sholawat serta salam tidak lupa penulis panjatkan pada Baginda Nabi Besar Muhammad SAW, semoga kita semua mendapatkan *syafaat* beliau di *yaumul akhir* nanti. Tujuan penelitian ini adalah untuk memenuhi salah satu syarat dalam memperoleh gelar sarjana strata satu (S1) pada Program Studi Matematika UIN Syarif Hidayatullah Jakarta.

Penulis sangat menyadari bahwa penelitian ini dapat diselesaikan dengan baik dengan bantuan dari banyak pihak. Untuk itu, pada kesempatan ini penulis ingin menyampaikan terima kasih kepada :

1. Bapak Ir. Nashrul Hakiem, S.Si., M.T., Ph.D. selaku Dekan Fakultas Sains dan Teknologi.
2. Ibu Dr. Suma’Inna, M.Si. selaku Kepala Prodi Matematika dan Ibu Irma Fauziah, M.Sc. selaku Sekretaris Prodi Matematika, yang senantiasa membantu administrasi saya sebagai mahasiswa.
3. Bapak Muhaza Liebenlito, M.Si. selaku pembimbing skripsi I dan Bapak Muhammad Irvan Septiar Musti, M.Si. selaku pembimbing skripsi II, yang telah membantu dan meluangkan waktunya untuk penulis dalam melakukan penelitian ini.
4. Ibu Dr. Suma’Inna, M.Si. dan Dr. Gustina Elfiyanti, M.Si. selaku penguji yang senantiasa memberikan kritik dan saran yang membangun dalam penulisan skripsi ini.
5. Bapak Mahmudi, M.Si. selaku dosen pembimbing akademik yang selama kuliah selalu memberikan bimbingan serta arahnya.



6. Bapak dan Ibu dosen di prodi matematika yang memberikan ilmunya yang sangat luar biasa kepada saya.
7. Ayahanda dan Ibunda tercinta Alm. Bapak Drs. Denny Priyatna, S.H. dan Ibu Nita Susanti yang dengan luar biasa bekerja keras untuk menyekolahkan anaknya dan mendidik saya selama saya ada di dunia ini.
8. Seluruh teman-teman yang berkontribusi dalam pembuatan skripsi ini terutama Maftuh Mashuri, S.Mat., Fadlan Bima hermawan, S.Mat., dan Ade Cici ,S.Si.
9. Semua teman-teman yang telah memberikan dukungan serta *support* selama penulisan penelitian ini terutama Lukman Hakim, Ahmad Khairul Umam, Renaldy Indra Oetama, Dewi Syifa Andini, dan Sheila Azzahra.
10. Nadin Amizah, S.I.Kom., yang dengan lagu-lagunya membuat penulis bersemangat untuk menyelesaikan penelitian ini.
11. Kawan-kawan matematika 2018 yang sudah menjadi angkatan yang luar biasa, menjadi rumah, teman bercanda dan teman diskusi

Penulis menyadari bahwa penelitian ini tidak lepas dari banyak kekurangan. Oleh karena itu, penulis berharap agar para pembaca sekalian dapat memberikan kritik maupun saran yang dapat membangun tersebut agar dapat menjadi bahan evaluasi untuk penulis kedepannya. Terima kasih.

*Wassalamu'alaikum Warrahmatullahi Wabarakatuh*

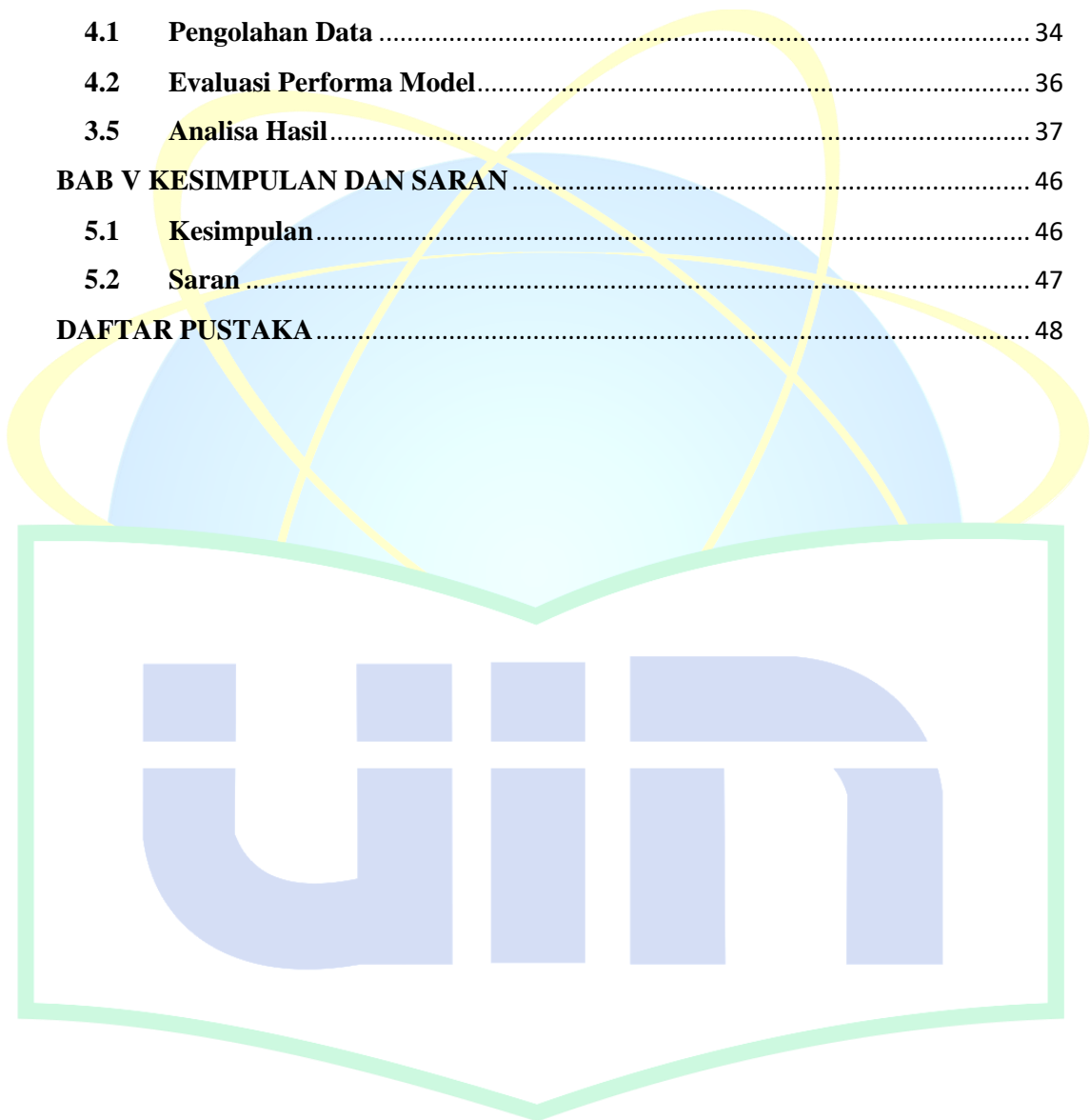
Jakarta, 20 Januari 2023

Penulis

## DAFTAR ISI

<b>PERNYATAAN.....</b>	<b>ii</b>
<b>LEMBAR PENGESAHAN .....</b>	<b>iii</b>
<b>PERSEMBAHAN .....</b>	<b>iv</b>
<b>ABSTRAK .....</b>	<b>v</b>
<b>ABSTRACT.....</b>	<b>vi</b>
<b>KATA PENGANTAR.....</b>	<b>vii</b>
<b>DAFTAR ISI.....</b>	<b>ix</b>
<b>DAFTAR GAMBAR.....</b>	<b>xi</b>
<b>BAB I PENDAHULUAN.....</b>	<b>1</b>
<b>1.1 Latar Belakang.....</b>	<b>1</b>
<b>1.2 Rumusan Masalah .....</b>	<b>5</b>
<b>1.3 Batasan Masalah .....</b>	<b>5</b>
<b>1.4 Tujuan Penelitian.....</b>	<b>6</b>
<b>1.5 Manfaat Penelitian.....</b>	<b>6</b>
<b>BAB II LANDASAN TEORI .....</b>	<b>7</b>
<b>2.1 Text Mining .....</b>	<b>7</b>
<b>2.2 Web Scrapping .....</b>	<b>8</b>
<b>2.3 Preprocessing .....</b>	<b>8</b>
<b>2.4 BERT.....</b>	<b>9</b>
<b>2.5 Dimensionality Reduction.....</b>	<b>10</b>
<b>2.6 Clustering .....</b>	<b>11</b>
<b>2.7 Class Based TF - IDF .....</b>	<b>12</b>
<b>2.8 Evaluasi Model .....</b>	<b>13</b>
<b>BAB III METODOLOGI PENELITIAN .....</b>	<b>15</b>
<b>3.1 Data Penelitian .....</b>	<b>15</b>
<b>3.2 Tahapan Penelitian .....</b>	<b>16</b>
<b>3.3 Preprocessing Text .....</b>	<b>19</b>
<b>3.4 Pemodelan Topik .....</b>	<b>20</b>
<b>3.4.1 Sentence-BERT .....</b>	<b>21</b>
<b>3.4.2 UMAP.....</b>	<b>22</b>

3.4.3	HDBSCAN .....	27
3.4.4	Class Based TF-IDF .....	31
3.5	<i>Guided Topic Modelling</i> .....	32
3.6	Visualisasi dan Interpretasi.....	33
<b>BAB IV HASIL DAN PEMBAHASAN .....</b>		<b>34</b>
4.1	Pengolahan Data .....	34
4.2	Evaluasi Performa Model.....	36
3.5	Analisa Hasil.....	37
<b>BAB V KESIMPULAN DAN SARAN .....</b>		<b>46</b>
5.1	Kesimpulan.....	46
5.2	Saran .....	47
<b>DAFTAR PUSTAKA .....</b>		<b>48</b>



## DAFTAR GAMBAR

<b>Gambar 1.1.</b> Jumlah Pendapatan Industri Video Game dibanding Industri Hiburan Lainnya[1].	2
<b>Gambar 1.2.</b> Jumlah Pemain Video Game di Seluruh Dunia Pada Tahun 2020[1]	3
<b>Gambar 3.1.</b> Diagram Alur Penelitian	17
<b>Gambar 3.2.</b> Diagram Proses SBERT	22
<b>Gambar 3.3.</b> Algoritma UMAP	23
<b>Gambar 3.4.</b> Algoritma Pembuatan himpunan simplisial fuzzy lokal	24
<b>Gambar 3.5.</b> Algoritma Penghitungan Normalisasi	24
<b>Gambar 3.6.</b> Algoritma Spectral Embedding untuk Permulaan	25
<b>Gambar 3.7.</b> Algoritma Optimalisasi Embedding	26
<b>Gambar 3.8.</b> Contoh Visualisasi Data 3 Dimensi Menjadi 2 Dimensi	26
<b>Gambar 3.9.</b> Algoritma Utama HDBSCAN	27
<b>Gambar 3.10.</b> Core Distance dalam Algoritma HDBSCAN	28
<b>Gambar 3.11.</b> Persebaran Kepadatan	28
<b>Gambar 3.12.</b> Contoh Penetapan Threshold yang Terlalu Tinggi	29
<b>Gambar 3.13.</b> Contoh Penetapan Threshold yang Terlalu Rendah	30
<b>Gambar 3.14.</b> Core Stability	31
<b>Gambar 3.15.</b> Alur Proses Metode "Guided Topic Modelling"	33
<b>Gambar 4.1</b> Perbandingan Jumlah Kata Sebelum dan Sesudah Preprocessing	35
<b>Gambar 4.2</b> Runningtime Algoritma BERTopic dengan "Guided Topic Modelling"	36
<b>Gambar 4.3</b> Coherence Score Algoritma BERTopic dengan "Guided Topic Modelling"	37
<b>Gambar 4.4</b> Paramater dan Seed Topic Yang digunakan Selama Proses	38
<b>Gambar 4.5</b> Kumpulan Kata dan Representasi Dokumen Topik 1	38
<b>Gambar 4.6</b> Kumpulan Kata dan Representasi Dokumen Topik 2	39
<b>Gambar 4.7</b> Kumpulan Kata dan Representasi Dokumen Topik 3	40
<b>Gambar 4.8</b> Kumpulan Kata dan Representasi Dokumen Topik 4	40
<b>Gambar 4.9</b> Kumpulan Kata dan Representasi Dokumen Topik 5	41
<b>Gambar 4.10</b> Kumpulan Kata dan Representasi Dokumen Topik 6	42
<b>Gambar 4.11</b> Kumpulan Kata dan Representasi Dokumen Topik 7	42
<b>Gambar 4.12</b> Kumpulan Kata dan Representasi Dokumen Topik 8	43
<b>Gambar 4.13</b> Kumpulan Kata dan Representasi Dokumen Topik 9	44
<b>Gambar 4.14</b> Kumpulan Kata dan Representasi Dokumen Topik 10	44

## BAB I

### PENDAHULUAN

Pada bab ini akan dijelaskan dari mana latar belakang penelitian ini berasal, tujuan dan manfaat penelitian ini serta rumusan dan batasan masalah dalam penelitian ini.

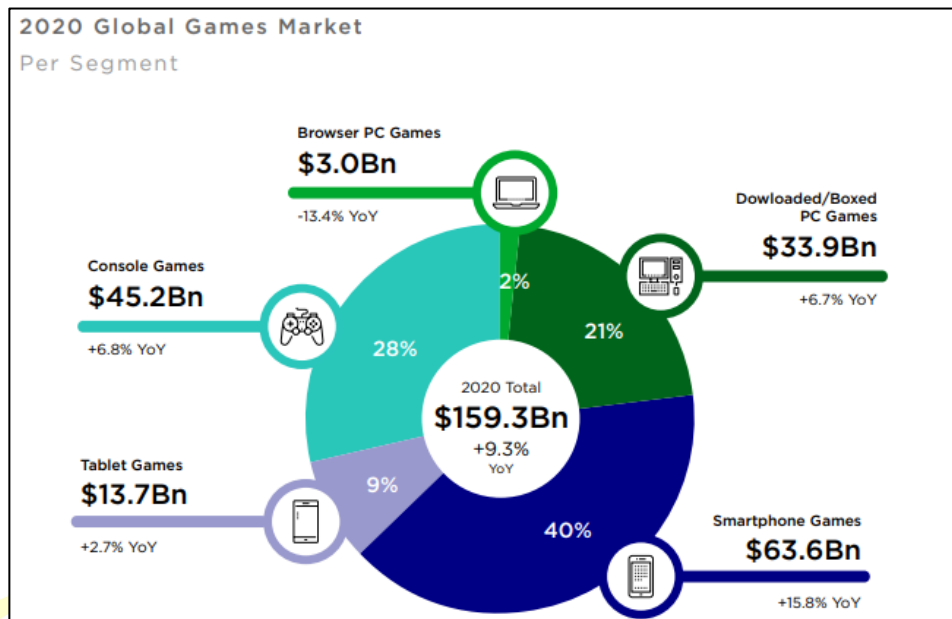
#### 1.1 Latar Belakang

Perkembangan teknologi dan informasi yang semakin pesat adalah hal yang tidak dapat disangkal lagi. Tidak terkecuali dalam bidang hiburan, banyak sekali bidang hiburan yang juga berkembang seiring berkembangnya teknologi. Salah satu bidang industri hiburan yang sedang berkembang dengan sangat pesat adalah industri permainan video atau *video game*. *Video game* sangat digemari oleh banyak orang tanpa memandang kalangan usia atau pun jenis kelamin. Walaupun demikian, sebagai seorang muslim kita tidak dilarang untuk mencari hiburan seperti memainkan *video game* asalkan dengan tetap memperhatikan ketentuan-ketentuan syariat. Seperti peringatan yang telah tertera pada Al-Quran surat Muhammad ayat 36 yang berbunyi :

إِنَّمَا الْحَيَاةُ الدُّنْيَا لَعِبٌّ وَلَهُوَ وَإِنْ تُؤْمِنُوا وَتَتَّقُوا يُؤْتِكُمْ أَجْرَكُمْ وَلَا يَسْلُكْكُمْ أَمْوَالَكُمْ

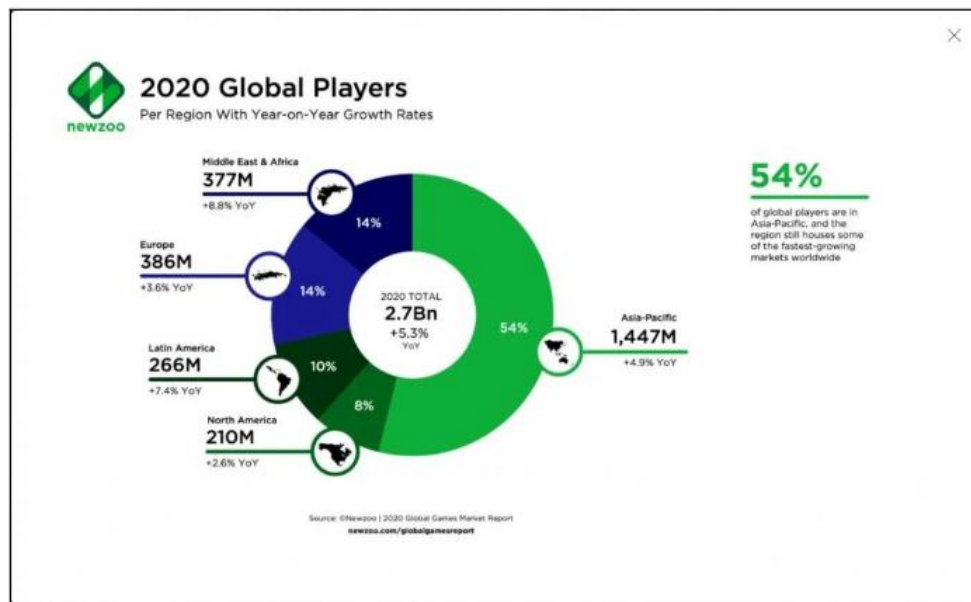
*“Sesungguhnya kehidupan dunia itu hanyalah permainan dan senda gurau. Jika kamu beriman serta bertakwa, Allah akan memberikan pahala kepadamu dan Dia tidak akan meminta hartamu.” (Muhammad : 36).*

Berkembangnya industri *video game* sendiri ditandai dengan besarnya *revenue* yang dihasilkan dari industri ini. dilansir dari [www.newzoo.com](http://www.newzoo.com), industri *video game* adalah industri hiburan dengan valuasi terbesar saat ini, dengan nilai *revenue* sebesar 159.3 US dollar pada tahun 2020 atau setara dengan 2,4 kuadriliun rupiah dan diprediksi akan terus meningkat kedepannya [1], seperti yang dapat dilihat pada Gambar 1.1.



**Gambar 1.1.** Jumlah Pendapatan Industri Video Game dibanding Industri Hiburan Lainnya[1].

Perkembangan industri video game yang sangat pesat ini tidak terlepas dari meningkatnya para pemain video game di dunia seiring makin banyaknya platform yang bisa digunakan untuk bermain video game, mulai dari PC (Personal Computer), konsol seperti Playstation dan Xbox, hingga smartphone. Seperti yang telah dirangkum oleh newzoo, total pemain video game di seluruh dunia mencapai 2,7 miliar jiwa atau lebih dari 30% populasi dunia saat ini [1], seperti yang tertera pada Gambar 1.2. Selain meningkatnya jumlah pemain, bertambahnya jumlah studio dan pengembang-pengembang game juga merupakan penyebab meledaknya industri video game beberapa tahun terakhir.



**Gambar 1.2.** Jumlah Pemain Video Game di Seluruh Dunia Pada Tahun 2020[1]

Negara Indonesia sendiri turut merasakan dampak dari berkembangnya industri video game ini. Seperti yang dilansir oleh CNBC Indonesia, Dedi Suherman CEO dari Melon Indonesia (perusahaan yang bergerak di bidang OTT video game, musik dan film.) menjelaskan bahwa pada tahun 2021 Indonesia memiliki 113,9 juta pemain video game dan diprediksi akan mencapai 120,2 juta pada tahun 2024 atau setara dengan 43% populasi Indonesia saat ini [2].

Seiring bertambah besarnya pasar video game beberapa tahun terakhir, setiap developer game pun berusaha menciptakan game-game terbaik yang bisa menarik minat para pemain. Mulai dari menghadirkan game yang memiliki mekanisme gameplay terbaru, hingga menciptakan game yang memiliki basis cerita yang sangat kuat. demi menarik perhatian para pemain [3].

Dalam membuat game yang baik, selain dibutuhkan kreativitas dalam mengolah mekanisme gameplay dan jalan cerita, dibutuhkan juga pandangan-pandangan dari para pemain game itu sendiri. Masukan dari para pemain berupa komentar, pujian atau kritikan bisa menjadi salah satu aspek yang dipertimbangkan oleh sebuah developer game, baik disaat mengembangkan suatu game atau saat pemeliharaan game yang sudah ada [4].



Salah-satu game yang sempat ramai pada tahun 2018 adalah salah satu game yang mempopulerkan genre “*Battle Royale*”. game tersebut masih cukup populer hingga saat ini, pada puncaknya orang yang memainkan game ini mencapai 3.2 juta pemain dalam satu waktu [5]. Bahkan dengan kepopuleran ini, game tersebut berhasil mendapatkan *revenue* sebesar 333.6 milyar Dollar Amerika Serikat sepanjang tahun 2021 [6]. Game ini sangat digemari karena mempopulerkan mekanisme yang cukup revolusioner pada masanya. Mekanisme yang diusung oleh game ini berupa 100 atau lebih pemain dikumpulkan pada area yang sama tanpa perlengkapan apapun, lalu tugas pemain adalah bertahan hidup dengan mengumpulkan perlengkapan dan mengalahkan pemain lain dengan area yang terus mengecil seiring waktu hingga tersisa satu pemain atau satu tim yang bertahan. Mekanisme permainan diatas disebut “*Battle Royale*” [7], dikarenakan kepopuleran mekanisme permainan tersebut sekitar 30% dari seluruh pemain game di *PC* memainkan game dengan mode permainan ini [8].

Maka dari itu, dengan penelitian ini penulis yang juga merupakan seorang pemain video game tersebut, ingin mencoba mempermudah pengembang video game dalam mengekstraksi informasi-informasi penting serta pandangan yang bisa di dapatkan dari *feedback* yang diberikan oleh para pemain dalam kolom *review* pada salah satu platform distribusi video game digital. Dikarenakan jumlah pemain dari game ini mencapai jutaan maka ulasan yang masuk juga sangatlah banyak, hal tersebut menjadi alasan peneliti menggunakan metode *topic modelling* [9] yang dilengkapi dengan fitur “*Guided Topic Modelling*” untuk mengekstraksi informasi-informasi penting dari ulasan *video game* yang telah ditentukan.

Dalam penelitian ini, Penulis menggunakan metode yang dapat membantu menentukan topik apa saja yang sedang dibahas dalam kolom review pada platform tersebut yaitu metode *Bidirectional Encoder Representation from Transformer (BERT)* atau lebih tepatnya lagi menggunakan *BERTopic* [10], sebuah algoritma turunan dari BERT [11] yang berfokus pada *topic modelling*. Selain dapat membantu menentukan topik yang ada dalam ulasan, Penelitian ini juga akan membantu melihat topik apa saja yang muncul saat di berikan sebuah atau beberapa



kata kunci yang akan membantu mem fokuskan topik yang muncul sesuai dengan dengan yang diinginkan. Pemilihan kata kunci bisa dilakukan dengan analisa awal terhadap masalah yang kerap kali terjadi pada sebuah *video game*. Metode diatas disebut juga metode “*Guided Topic Modelling*” [10].

Penelitian ini mengambil referensi utama dari jurnal penelitian Muhammad Yudha dan kawan-kawan pada awal tahun 2021 [12] yang membahas pemodelan topik pada *video game indie* dan juga penelitian Vasudeva Raju dan kawan-kawan pada tahun 2022 [13] yang membahas pemodelan topik menggunakan algoritma *BERTopic*. Dan penelitian dari Marteen Grootendorst pada 2022 yang membahas tentang algoritma *BERTopic* [10].

Perbedaan penelitian ini dengan penelitian diatas adalah penulis menggunakan metode *BERTopic* berbasis sentence embedding dengan fitur “*Guided Topic Modelling*” dan menerapkannya dalam pemodelan topik pada ulasan video game pada platform pendistribusian video game digital.

## **1.2 Rumusan Masalah**

Berdasarkan latar belakang masalah yang dipaparkan diatas, maka rumusan masalah dalam penulisan laporan ini adalah sebagai berikut:

1. Topik apa saja yang dibicarakan oleh pemain dalam ulasan *video game* bergenre *battle royale*.
2. Bagaimana hasil yang diperoleh dengan menggunakan fitur “*Guided Topic Model*” dalam metode *BERTopic*

## **1.3 Batasan Masalah**

Agar penelitian lebih terarah dan jelas, maka dibuatlah batasan permasalahan yang mengacu pada indikator yang akan difokuskan, yaitu :

1. Data berupa sebagian besar text ulasan salah satu video game dengan genre “*Battle Royale*” dari tahun 2017 hingga tahun 2021.
2. Pada penelitian ini, data ulasan video game diambil secara “*Scrapping*” dari salah satu platform pendistribusian video game digital

3. Penelitian ini menggunakan Metode *BERTopic* yang telah tersedia dalam modul *BERTopic* dengan semua parameter dalam keadaan *default* kecuali :
  - a. *min\_df* = 10
  - b. *min\_topic\_size* = 100
  - c. *n\_gram\_range* = 1, 2
  - d. *nr\_topics* = 9
4. Analisis hasil terhadap topik yang didapatkan bersifat eksplorasi data sehingga interpretasi yang didapatkan kebanyakan adalah hipotesis yang perlu diteliti lagi di masa yang akan datang.

#### 1.4 Tujuan Penelitian

Selanjutnya adalah tujuan penelitian, berdasarkan uraian pada rumusan dan batasan masalah beberapa tujuan yang ingin dicapai pada penelitian kali ini adalah sebagai berikut :

1. Mengetahui topik apa saja yang dibicarakan oleh pemain dalam ulasan video game bergenre *battle royale*.
2. Mengetahui bagaimana performa “*Topic Modelling*” yang dihasilkan menggunakan fitur “*Guided Topic Model*” pada metode *BERTopic*.

#### 1.5 Manfaat Penelitian

Adapun manfaat yang dapat diambil dari penelitian ini adalah harapannya penelitian ini dapat mempermudah pengembang video game dalam mendapatkan informasi serta kritik dan masukan dari ulasan yang sangat banyak yang dibuat oleh pemain untuk *video game* yang mereka buat sebagai kontribusi penulis dalam dunia industri *video game*. Dan juga harapan penelitian ini dapat menjadi referensi dan memberikan gambaran bagi penelitian serupa dimasa yang akan datang.

## BAB II

### LANDASAN TEORI

Bab ini akan membahas tentang teori-teori yang melandasi penelitian ini. akan dijelaskan tentang komponen-komponen pemodelan topic (*Topic Modelling*) dan beberapa teori yang akan digunakan dalam penelitian ini.

#### 2.1 *Text Mining*

Seiring berkembangnya ilmu yang mempelajari data, data tidak terstruktur seringkali menjadi persoalan. Salah satu contoh data yang tidak terstruktur namun sangat sering ditemui dan biasanya memiliki informasi yang banyak adalah data teks. Mengolah data teks untuk menemukan pola dan informasi yang penting didalamnya biasa disebut *Text Mining* [14]. Dalam melakukan *Text Mining* berikut langkah - langkah yang diperlukan [15] :

1. Mengumpulkan informasi dari data yang tidak terstruktur
2. Mengubah informasi yang didapatkan menjadi data terstruktur
3. Identifikasi pola dari data terstruktur
4. Analisis pola yang didapatkan
5. Ekstrak informasi berharga dan disimpan dalam database

Proses yang dilakukan dalam *Text Mining* memang cukup sederhana namun prosesnya membutuhkan tahapan - tahapan serta perhitungan statistik [15]. Dalam perkembangannya, *Text Mining* memiliki banyak penerapan diantaranya, Sentimen Analisis, Pemodelan Topik dan masih banyak lagi [16].

Secara general, terdapat 4 proses yang perlu dijalankan dalam *Text Mining* untuk *Topic Modelling* dan dilakukan dalam penelitian ini, antara lain:

1. Pengambilan data dengan *scrapping* dari website yang ditentukan.
2. Pembersihan data dengan *text preprocessing*.
3. Pemodelan yang akan memproses data secara berulang (*looping*)
4. Dengan proses evaluasi dan validasi.
5. Visualisasi data dari hasil pemodelan data yang dilakukan.

## 2.2 *Web Scrapping*

Kolom ulasan merupakan salah satu fitur yang cukup penting yang harus dimiliki oleh sebuah platform pendistribusian video game digital. Banyak informasi yang bisa didapatkan dari kolom ulasan tersebut seperti, pendapat dari pemain mengenai suatu produk, berapa lama ia telah bermain, dan seberapa membantunya ulasan yang ia berikan.

Salah satu cara yang bisa dilakukan untuk mendapatkan informasi-informasi tersebut adalah *Web Scrapping* yaitu mengambil teks serta informasi dalam suatu web [17]. Melakukan *Web Scrapping* pada suatu web dapat membantu mengumpulkan informasi-informasi serta data dari web tersebut untuk selanjutnya bisa dianalisa dan didapatkan informasi yang berharga.

## 2.3 *Preprocessing*

Data yang telah kita dapatkan dari hasil *Web Scrapping* pasti memiliki banyak kata-kata yang berantakan yang memerlukan preprocessing [18] seperti menghapus simbol dan tanda baca, menghapus huruf yang berulang terlalu banyak dan lain - lain. Proses Processing yang dilakukan pada penelitian ini sebagai berikut:

### 1. *Case Folding*

*Case Folding* adalah mengubah semua karakter menjadi bentuk yang sama, contohnya mengubah semua huruf kapital menjadi huruf kecil [19]. Hal ini menjadi cukup penting karena seringkali penggunaan huruf kapital tidak konsisten dalam data ulasan, sehingga memerlukan proses *case folding*.

### 2. Penambahan Titik Pada Akhir Kalimat

Pada penelitian ini, salah satu prosedur yang akan dilakukan adalah "*Sentence Embedding*". Sesuai namanya, model ini bekerja dengan melakukan penyematan pada sebuah kata dengan melihat konteksnya pada suatu kalimat, sehingga menambahkan titik pada akhir kalimat menjadi hal yang cukup penting untuk dilakukan.

### 3. Menghapus simbol, angka, dan emotikon

Dalam menulis ulasan, banyak pemain yang menggunakan simbol atau angka pada tiap ulasan. Ada juga ulasan yang mengandung emotikon sebagai bentuk ekspresi pemain. Namun, ketiga faktor tersebut tidak akan kita teliti dalam penelitian kali ini. Maka dari itu, hal tersebut harus dihilangkan untuk memudahkan melakukan analisis data.

### 4. Menghapus huruf dan kata-kata yang berulang

Ada Kalanya ketika pemain menulis sebuah ulasan mereka mengulang huruf pada suatu kata terlalu banyak, mungkin hal tersebut menunjukkan penekanan terhadap kata tersebut atau bisa juga hanya sekedar kesalah penulisan. Pada penelitian kali ini huruf yang berulang akan dihapus guna menyeragamkan kata-kata yang ada dan juga mempermudah analisis.

### 5. Menghapus “Stopwords”

Stopword ialah kata umum yang biasa ditemukan tetapi tidak memiliki arti atau tidak berhubungan dengan informasi yang dipelajari, biasanya berupa kata depan atau subjek [20]. Maka dari itu diperlukan proses ini dengan memuat membuat kamus untuk menyesuaikan dengan data yang diteliti. Contoh kata *stopword* ialah “abang”, “atau”, “datang” dan sebagainya.

## 2.4 BERT

Pada tahun 2018, peneliti ahli dari Google AI Language mengembangkan model representasi bahasa terlatih, yang diberi nama dengan BERT atau *Bidirectional Encoder Representations from Transformers* [11]. Inovasi model arsitektur BERT yaitu representasi multi-layer dua arah (*bidirectional*) menggunakan *encoder* dari *Transformer* [21]. Transformers sendiri merupakan suatu mekanisme yang berfungsi untuk mempelajari adanya hubungan antar kata dalam teks [21]. Mekanisme yang terdapat pada *transformer*, yaitu:

## 1. *Encoder*

*Encoder* merupakan salah satu bagian dari arsitektur *Transformer* yang terdiri dari tumpukan 6 atau lebih lapisan identik. Setiap lapisan memiliki 2 sublapisan, yaitu lapisan *self-attention* dan *feed-forward neural network* [22]. Fungsi dari *self-attention layer* yaitu *encoder* membantu setiap *node* untuk fokus pada kata yang terlihat, serta mendapatkan konteks semantik dari kata tersebut. Lapisan *Self-attention* memungkinkan setiap posisi *encoder* menangani semua posisi lapisan sebelumnya dan posisi saat ini.

## 2. *Decoder*

*Decoder* juga terdiri dari tumpukan 6 atau lebih lapisan identik. *Decoder* berguna untuk menghasilkan urutan *output* yang dapat diprediksi. Setiap lapisan terdiri dari dua sublapisan, yaitu *self-attention layer* dan *feed-forward neural network* [22], tetapi diantara dua lapisan ini terdapat lapisan *attention layer* yang membantu *node* saat ini mendapatkan *key content* yang membutuhkan perhatian dengan cara melakukan *multi-head attention* dari *output* di *encoder* [23].

Dalam prosedur *BERTopic* yang akan digunakan dalam penelitian kali ini, kita akan menggunakan salah satu varian dari BERT yaitu, *Sentence – BERT* [24] atau SBERT. SBERT merupakan modifikasi dari jaringan pretrained BERT yang menggunakan struktur jaringan *siamese* sehingga dapat memperoleh *sentence embedding* yang bermakna secara semantik dan dapat dibandingkan menggunakan *cosine-similarity*.

## 2.5 *Dimensionality Reduction*

Dalam menjalankan prosedur *BERTopic* salah satu hal yang terpenting untuk dilakukan adalah mengurangi dimensi data yang dihasilkan saat melakukan proses “*Embedding*”. Biasanya data yang dihasilkan dari proses *embedding* setidaknya memiliki 384 dimensi [10], dan banyak dari algoritma “*Clustering*” kesulitan untuk memproses data yang memiliki dimensi terlalu tinggi. Maka dari itu solusinya adalah dengan mengurangi data dengan “*Dimensionality Reduction*”

[25], yaitu merubah data vektor yang memiliki dimensi tinggi menjadi dimensi yang lebih rendah sehingga data tersebut bisa diproses oleh algoritma *clustering*.

UMAP (*Uniform Manifold Approximation and Projection*) adalah teknik terbaru pembelajaran manifold untuk reduksi dimensi [26]. UMAP dibangun dari kerangka teoritis yang didasarkan pada geometri dan topologi aljabar Riemannian. Hasilnya adalah algoritma terukur praktis yang berlaku untuk data dunia nyata. Algoritma UMAP bersaing dengan t-SNE untuk kualitas visualisasi, dan bisa dibilang mempertahankan lebih banyak struktur global dengan kinerja waktu berjalan yang unggul. Selain itu, UMAP tidak memiliki keterbatasan komputasi dimensi embedding, membuatnya layak sebagai tujuan umum teknik pengurangan dimensi untuk machine learning.

## 2.6 Clustering

*Clustering* atau klasterisasi adalah metode pengelompokan data ke dalam beberapa *cluster* atau kelompok sehingga data dalam satu *cluster* memiliki tingkat kemiripan yang maksimum dan data antar *cluster* memiliki kemiripan yang minimum [27]. Dalam “*Topic Modelling*” *clustering* berguna untuk menglompokkan kata yang sebelumnya telah melewati tahap *preprocessing* sehingga terbentuk kelompok topik yang ada dalam suatu dokumen

Metode clustering yang digunakan dalam penelitian ini adalah HDBSCAN (*Hierarchical Density Based Spatial Clustering of Application with Noise*) yang merupakan metode pengelompokan bertingkat atau hierarkis yang berbasis kepadatan yang telah ditingkatkan secara teoritis dan praktikal, memberikan hirarki pengelompokan dimana cabang yang disederhanakan dari kelompok yang signifikan dapat dibangun [28]. Untuk mendapatkan partisi "flat" yang hanya terdiri dari cluster paling signifikan (mungkin sesuai dengan ambang kepadatan yang berbeda), kami mengusulkan ukuran stabilitas cluster baru, memformalkan masalah memaksimalkan stabilitas keseluruhan cluster yang dipilih, dan merumuskan algoritma yang menghitung solusi optimal untuk masalah ini.



## 2.7 Class Based TF - IDF

Prosedur klasik TF - IDF menggabungkan dua statistik yaitu, “*Term Frequency*” dan “*Inverse Document Frequency*” [29] sehingga terbentuk rumus berikut :

$$W_{t,d} = tf_{t,d} \cdot \log \left( \frac{N}{df_t} \right) \quad (2.1)$$

$tf_{t,d}$  = Frekuensi kata  $t$  pada dokumen  $d$

$df_t$  = Jumlah dokumen yang mengandung kata  $t$

$N$  = Jumlah dokumen di sebuah korpus  $N$

Di mana “*Term Frequency*” memodelkan frekuensi kata  $t$  dalam dokumen  $d$  dan “*Inverse Document Frequency*” mengukur berapa banyak informasi suatu istilah atau kata tersedia untuk dokumen dan dihitung dengan mengambil logaritma dari jumlah dokumen di sebuah korpus  $N$  dibagi dengan jumlah dokumen yang mengandung  $t$ .

Kemudian prosedur ini di generalisasikan ke dalam cluster dokumen. Pertama, semua dokumen dalam sebuah *cluster* diperlakukan sebagai satu dokumen hanya dengan menggabungkan dokumen. Kemudian, TF-IDF disesuaikan untuk representasi ini dengan menerjemahkan dokumen ke cluster, sehingga terbentuk rumus berikut:

$$W_{t,c} = tf_{t,c} \cdot \left( 1 + \frac{A}{tf_t} \right) \quad (2.2)$$

$tf_{t,c}$  = Frekuensi kata  $t$  pada cluster  $c$

$tf_t$  = Frekuensi kata  $t$  pada semua cluster

$A$  = Jumlah rata – rata kata per kelas  $A$



Dimana “*Term Frequency*” memodelkan frekuensi kata  $t$  di kelas  $c$  atau dalam contoh ini. Di Sini, kelas  $c$  adalah kumpulan dokumen yang digabungkan menjadi satu dokumen dalam *cluster*. Kemudian, “*Inverse Document Frequency*” diganti dengan “*Inverse Class Frequency*” untuk mengukur berapa banyak informasi yang disediakan sebuah kata untuk suatu kelas. Itu dihitung dengan mengambil logaritma dari jumlah rata-rata kata per kelas  $A$  dibagi dengan frekuensi kata  $t$  di semua kelas. Untuk hanya menampilkan nilai positif, kami menambahkan satu ke pembagian dalam logaritma.

Dengan demikian, prosedur TF-IDF berbasis kelas ini memodelkan pentingnya kata-kata dalam kelompok, bukan dokumen individu. Hal ini memungkinkan kita untuk menghasilkan distribusi topik-kata untuk setiap kelompok dokumen. Terakhir, dengan menggabungkan perwakilan c-TF-IDF secara iteratif dari topik yang paling tidak umum dengan topik yang paling umum. Dengan demikian, kita dapat mengurangi jumlah topik menjadi nilai yang ditentukan oleh pengguna.

## 2.8 Evaluasi Model

Evaluasi sebuah model penting dilakukan karena evaluasi model merupakan salah satu cara untuk melihat seberapa baik kinerja model. Dan juga dapat menjadi perbandingan model dengan model lainnya. Ada beberapa cara yang bisa dilakukan untuk mengevaluasi sebuah model contohnya, *running time* dan *coherence score* [30].

*Running time* akan menilai seberapa cepat model dapat memprediksi topik dari sebuah dokumen atau data set. Penilaian ini menjadi penting karena pada penerapannya kecepatan sebuah model dalam memprediksi topik adalah hal yang sangat di pertimbangkan nantinya ketika model akan diaplikasikan pada sebuah program.

Selain kecepatan, ketepatan sebuah model juga menjadi hal yang sangat dipertimbangkan. Salah satu cara untuk mengukur ketepatan atau seberapa baik performa sebuah model “*Topic Modelling*” adalah dengan mengukur “*Topic Coherence*” dari model tersebut [31]. Konsep dari topic coherence adalah menggabungkan sejumlah ukuran ke dalam kerangka kerja untuk mengevaluasi koherensi antara topik yang disimpulkan oleh sebuah model. Topic Coherence dapat dihitung dengan melakukan perbandingan antara pasangan kata-kata di dalam suatu topik tertentu yang dapat menghasilkan ukuran standar kualitas suatu topik, semakin tinggi nilai *coherence* akan semakin baik modelnya [31]. Terdapat dua tipe dalam menghitung *coherence score* yaitu [32] :

1. *Extrinsic UCI measure*

$$SCORE_{UCI}(w_i, w_j) = \log \frac{p(w_i, w_j)}{p(w_i)p(w_j)} \quad (2.3)$$

Dimana  $p(w_i) = \frac{D_{corpus}(w_i)}{D_{corpus}}$  dan  $p(w_i, w_j) = \frac{D_{corpus}(w_i, w_j)}{D_{corpus}}$

2. *Intrinsic UMass measure*

$$SCORE_{UMass}(w_i, w_j) = \log \frac{D(w_i, w_j) + 1}{D(w_i)} \quad (2.4)$$

$D$  = Jumlah total dokumen dalam korpus

$D(w_i)$  = Jumlah total dokumen yang mengandung kata  $w_i$

$D(w_i, w_j)$  = Jumlah total dokumen yang mengandung kata  $w_i$  dan  $w_j$

### BAB III

#### METODOLOGI PENELITIAN

Pada bab ini akan diuraikan langkah-langkah dalam melakukan penelitian ini. Beberapa hal yang akan diuraikan antara lain adalah sumber data penelitian, tahapan penelitian dan diagram alur penelitian.

##### 3.1 Data Penelitian

Data yang akan digunakan pada penelitian kali ini adalah data text ulasan salah satu *video game* dengan genre *battle royale* yang bersumber dari salah satu platform pendistribusian video game digital. Data yang akan diproses berisikan sebagian besar ulasan video game tersebut dari tahun 2017 hingga 2021 yang berjumlah 55.203 ulasan negatif dari sekitar 3,2 juta pemain.

Data yang digunakan dalam penelitian ini didapatkan menggunakan metode *scrapping* secara otomatis dengan bantuan *python* pada website platform pendistribusian video game digital tersebut. Total data awal yang didapatkan sebanyak 138.725 baris ulasan dengan 5 kolom yang berisikan : *date\_post*, *username*, *hour\_played*, *recommend* dan *review* dalam file yang tersimpan dalam format CSV (*Comma Separated Value*).

**Tabel 3.1.** Data Awal Hasil *Scrapping*

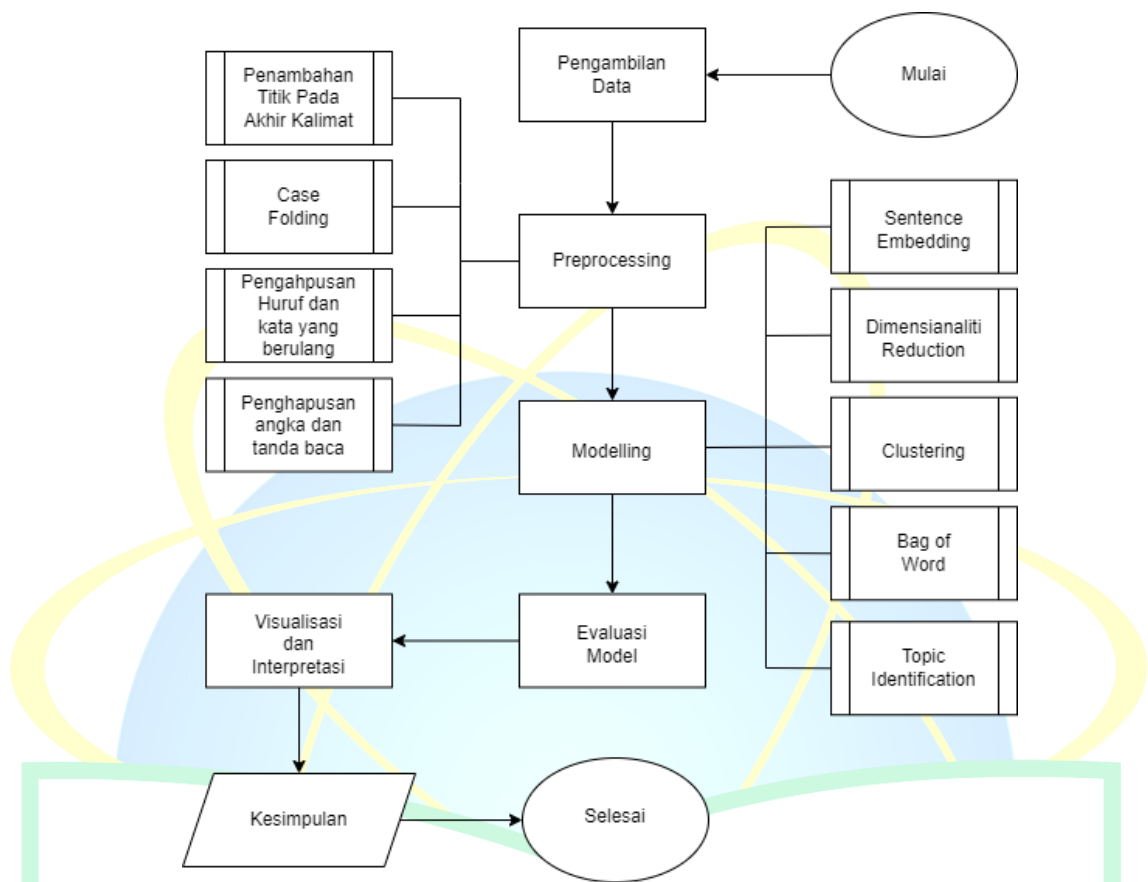
date_posted	username	hour_played	recommend	review
03/09/2022	frrtcht	170	Not Recommended	it allows them to acess and store (*and share) ridiculous amounts of information that they have no buisiness knowing (...)
05/08/2019	jaffaz	1499.4	Not Recommended	I've always asked myself, how can people play a game for 1000+ hours and then give it a thumbs down? (...)
24/06/2017	Chairman Bao	207.2	Recommended	PUBG is the type of game where it starts off with not caring too much if

				you die in the first few minutes (approx 90 players alive).(...)
28/06/2018	Lo Pan	180.2	Not Recommended	This game is like dating an alcoholic. They're a lot of fun to be around,(...)
11/01/2022	Garlic God	10.5	Not Recommended	Rather than fixing the cheating problem, let's just give it away for free! (...)

Selanjutnya data tersebut digabungkan dan disortir untuk diambil hanya ulasan dalam kategori *Not Recommended* dan pemainnya memiliki “*play hour*” sekurang-kurangnya lima jam. Hal ini dilakukan agar topik yang di hasilkan berfokus pada kekurangan dari *game* ini dan juga untuk menghindari ulasan yang ditulis oleh pemain yang tidak merasakan pengalaman game tersebut secara utuh. Hingga data yang tersisa untuk diproses berjumlah 59.472 ulasan untuk *video game* yang digunakan dalam penelitian ini.

### 3.2 Tahapan Penelitian

Sub-bab ini akan menjelaskan tahapan-tahapan yang dilakukan pada penelitian ini. Langkah-langkah akan diuraikan sejelas mungkin, berikut adalah tahapan yang peneliti lakukan selama melakukan penelitian seperti diagram alur penelitian pada Gambar 3.1:



**Gambar 3.1.** Diagram Alur Penelitian

1. Tahap pertama adalah proses pengambilan data dari kolom ulasan pada platform STEAM seperti yang telah di jelaskan pada subbab sebelumnya. Data diambil pada kolom komentar dari judul video game yang ingin diproses, pada peneltitian kali ini kita akan mengambil data ulasan dari video game yang bergenre *battle royale*. Data awal yang didapatkan berupa data dengan format CSV berjumlah 138.725 baris dengan 5 kolom yang berisikan: *date\_posted*, *username*, *hour\_played*, *recommend*, dan *review*. Kemudian data yang telah didapatkan akan disortir dengan mengambil hanya data ulasan pemain yang telah memainkan video game yang akan diteliti sekurang-kurangnya lima jam. Hal ini dilakukan untuk menghindari ulasan dari pemain yang belum merasakan fitur dari video game yang akan diteliti secara menyeluruh.

2. Data yang sudah dalam format *csv* lalu akan di *preprocessing*. *Preprocessing* dilakukan dengan menggunakan bahasa pemrograman python dengan modul bernama *regularexpression*. Pertama kita kan menghilangkan data ulasan yang hanya memiliki kurang dari satu kata atau kurang dari empat huruf, setelah dihilangkan data yang tersisa 132.943 baris yang akan dilakukan *preprocessing* lebih lanjut yakni penambahan titik pada akhir kalimat dikarenakan model yang berbasis *sentence embedding*. Kemudian akan dilanjutkan dengan *case folding*, penghapusan simbol dan tanda baca, penghapusan huruf dan kata-kata yang berulang dan yang terakhir *penghapusan stop word*. *Stopword* yang akan digunakan berformat *txt* yang berisi kata ganti, kata sambung, kata slang dan lain sebagainya. Sehingga data yang tersisa untuk diproses sebanyak 55.203 ulasan.

3. Setelah data melewati proses *preprocessing*, yang selanjutnya akan dilakukan adalah *modelling*. Data bersih yang didapat dari proses *preprocessing* akan dimulai dimasukkan kedalam tahapan prosedur *BERTopic*. Tahap pertama yang akan dilakukan adalah *sentence embedding* menggunakan *sentence-BERT*, Setiap kata dalam data ulasan akan diberikan bobot sesuai makna semantik dari kata tersebut dalam suatu kalimat, data yang didapatkan akan berupa kumpulan kata-kata dalam bentuk matriks berdimensi tinggi yang nantinya akan diproses menggunakan algoritma *clustering*. Selanjutnya data matriks yang diperoleh akan di proses dengan algoritma *clustering*. Namun untuk melakukan hal tersebut data matriks berdimensi tinggi yang didapatkan harus diubah menjadi data berdimensi rendah menggunakan algoritma *dimensionality reduction*, untuk penelitian kali ini akan digunakan metode *UMAP (Uniform Manifold Approximation and Projection)*. Setelah data yang akan diproses memiliki dimensi rendah, data akan diproses menggunakan algoritma *clustering*, pada penelitian kali ini akan digunakan algoritma *HDBSCAN (Hierarchical Density Based Spatial*

*Clustering of Application with Noise*). Hasilnya kata-kata dalam data yang masih berbentuk matriks akan dikelompokkan menjadi kelompok-kelompok (*cluster*) yang telah membentuk sejumlah topik yang dibicarakan dalam data ulasan. Selanjutnya data yang sebelumnya berbentuk matriks akan diubah menjadi menjadi bentuk semula dan di tokenisasi. Terakhir kata-kata dalam setiap *cluster* akan diberikan bobot menggunakan algoritma *c-TF-IDF* (*Class Based - Term Frequency-Invers Document Frequency*) dengan pemberian bobot tersebut akan terlihat mana kata yang paling identik dengan suatu topik (*cluster*) dan tidak dengan topik atau *cluster* yang lain.

4. Topik-topik yang telah diekstraksi dari data yang kita miliki akan divisualisasikan kedalam *barchart* untuk melihat kata apa saja yang mewakili topik-topik yang kita dapatkan sesuai bobotnya. selain memvisualisasikan topik, akan divisualisasikan juga persebaran *cluster* dalam data untuk melihat kinerja algoritma *clustering* yang dipakai dalam penelitian ini.

### **3.3 Preprocessing Text**

Pada tahap ini akan dijelaskan terkait dengan *preprocessing* dan hasilnya. Data teks ulasan memiliki format yang tidak terstruktur, data yang belum memiliki format yang terstruktur belum bisa diambil informasinya, sehingga perlu cara untuk membuatnya lebih terstruktur. *Preprocessing Text* bertujuan untuk membuat data lebih terstruktur dan mengurangi *noise* pada data seperti singkatan dan bentuk yang tidak beraturan. Hal tersebut membuat tahapan ini menjadi sangat penting karena pada tahapan inilah data dipersiapkan sehingga bisa dilakukan analisis lebih lanjut, beberapa proses didalam tahap *preprocessing* ini adalah sebagai berikut :

1. Menambahkan titik pada akhir kalimat
2. *Case Folding*
3. Menghapus tanda baca, angka dan simbol
4. Menghapus huruf dan kata-kata yang berulang
5. *Remove Stopwords*



Setelah dilakukan *preprocessing*, barulah data memiliki struktur dan format yang jelas dan akan lebih mudah untuk di proses. Berikut hasil dari *preprocessing* yang telah dilakukan

**Tabel 3.2.** Contoh *Preprocessing*

Kalimat Awal	game that makes me drown in adrenaline. Blood rushing through your brain. Can't breathe after every single fight. FPP mode is amazing. The only cons is performance but well it's only in beta. So, for me it's okay.
Penambahan titik	game that makes me drown in adrenaline. Blood rushing through your brain. Can't breathe after every single fight. FPP mode is amazing. The only cons is performance but well it's only in beta. So, for me it's okay.
Case Folding	game that makes me drown in adrenaline. blood rushing through your brain. can't breathe after every single fight. fpp mode is amazing. the only cons is performance but well it's only in beta. so, for me it's okay
Menghapus kata-kata yang ber ulang, simbol, tanda baca dan <i>stopwords</i>	game that makes me drown in adrenaline. blood rushing through your brain can t breathe after every single fight. fpp mode is amazing the only cons is performance but well it s only in beta. so, for me it s okay

### 3.4 Pemodelan Topik

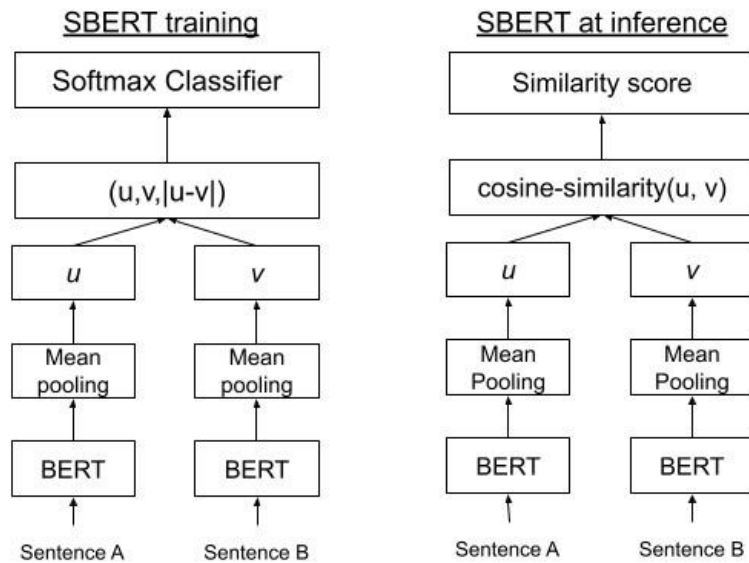
Tahap selanjutnya dalam penelitian ini adalah mengekstrak topik-topik yang dibicarakan pemain dari dalam data ulasan yang ada. Metode untuk untuk mengekstrak topik dari dalam dokumen disebut “*Topic modelling*” pemodelan topik, Dalam kali ini kita akan menggunakan metode pemodelan bernama *BERTopic*. *BERTopic* merupakan sebuah prosedur pemodelan topik yang memanfaatkan *sentence embedding* dalam memberikan makna semantik sebuah kata dalam kalimat dan juga menggunakan *class based TF-IDF* sebagai algoritma pemberi bobot untuk mengekstrak topik dari *cluster* yang terbentuk dari data. Selengkapnya Prosedur *BERTopic* adalah sebagai berikut :



### 3.4.1 Sentence-BERT

Tahap awal prosedur *BERTopic* adalah *sentence embedding*. Data ulasan yang telah melewati proses *preprocessing* akan dimasukkan ke dalam algoritma *sentence embedding*, pada penelitian kali ini algoritma *sentence embedding* yang akan digunakan adalah “*Sentence-BERT*” yang telah tersedia dalam modul “*SentenceTransformers*”. “*Sentences-BERT*” memiliki cara kerja yang serupa dengan “*BERT*” dengan beberapa perubahan seperti menghilangkan klasifikasi akhir serta menggunakan arsitektur “*Siamese Network*” yang berisi dua arsitektur *BERT* yang identik dan memiliki beban yang sama. Pertama, setiap ulasan akan diproses berpasangan dan diinput ke dalam masing-masing *BERT* dalam *network*.

Kata dalam setiap ulasan akan dikonversi ke dalam bentuk *list vector* menggunakan “*Wordpiece Embedding*” kemudian *list vector* tersebut di transformasi sehingga menghasilkan *sentence embedding* yang akan di *pooling* menggunakan metode *mean pooling*, *pooling* adalah teknik untuk menggeneralisir fitur dalam *network*. Dan *mean pooling* bekerja dengan merata-ratakan kumpulan fitur dalam *BERT*. Setelah *pooling* selesai, sekarang kita memiliki dua *embeddings*, satu untuk kalimat A dan satu untuk kalimat B. Saat model dilatih, *SBERT* menggabungkan dua *embeddings* yang kemudian dijalankan melalui *softmax classifier* [33] dan dilatih menggunakan fungsi *softmax-loss*. Pada inferensi, kedua penyematan tersebut kemudian dibandingkan menggunakan fungsi “*Cosine Similarity*” [34], yang akan menampilkan skor kesamaan untuk kedua kalimat tersebut. Berikut adalah diagram untuk *SBERT* saat *fine tuning* dan pada *Inference* pada Gambar 3.2. Hasilnya kita memiliki data ulasan yang telah berubah menjadi *list vector* yang memiliki 768 dimensi.



**Gambar 3.2.** Diagram Proses SBERT

### 3.4.2 UMAP

Tahap berikutnya setelah data ulasan sudah berbentuk *list vector* dan melewati tahap *embedding*, data tersebut akan memasuki tahap *clustering*. Namun hal tersebut belum bisa dilakukan karena *list vector* dari data ulasan yang ada masih memiliki dimensi yang terlalu tinggi, sehingga sulit untuk algoritma *clustering* untuk memproses data tersebut. Maka dari itu kita membutuhkan tahap ini yaitu *dimensionality reduction* dimana *list vector* dari data ulasan yang memiliki dimensi tinggi akan dikonversi menjadi data vektor dengan dimensi yang lebih rendah menggunakan algoritma UMAP (*Uniform Manifold Approximation and Projection*).

Algoritma UMAP akan membuat data vektor yang memiliki dimensi tinggi menjadi menjadi vektor berdimensi rendah dengan tetap menjaga struktur global dan lokal dari data tersebut [26]. UMAP bekerja dengan membuat representasi graf berdimensi tinggi dari data kemudian mengoptimalkannya menjadi graf berdimensi rendah tanpa menghilangkan struktur lokal serta global dari data. Untuk membuat permulaan graf berdimensi tinggi, UMAP membuat sesuatu yang disebut “Fuzzy

*Simplicial Complex*” [26] yang merupakan representasi graf berbobot dengan bobot sisi mewakili keterhubungan antara dua titik. Untuk menentukan keterhubungan, UMAP memperluas radius keluar dari setiap titik, menghubungkan titik-titik ketika jari-jari tersebut tumpang tindih. Memilih radius ini sangat penting, radius yang terlalu kecil akan menghasilkan cluster yang kecil dan terisolasi, sementara radius yang terlalu besar akan menghubungkan semuanya. UMAP mengatasi hal ini dengan memilih radius secara lokal, berdasarkan jarak ke tetangga terdekat ke- $n$  setiap titik. UMAP kemudian membuat grafik "fuzzy" dengan mengurangi kemungkinan koneksi saat radius bertambah. Terakhir, dengan menetapkan bahwa setiap titik harus terhubung setidaknya ke tetangga terdekatnya, UMAP memastikan bahwa struktur lokal dipertahankan seimbang dengan struktur global [35]. Secara lebih detail algoritma UMAP sebenarnya relatif jelas dapat dilihat pada Gambar 3.3.

---

**Algorithm 1** UMAP algorithm

---

```

function UMAP( $X, n, d, \text{min-dist}, \text{n-epochs}$ )

    # Construct the relevant weighted graph
    for all  $x \in X$  do
         $\text{fs-set}[x] \leftarrow \text{LOCALFUZZYSIMPLICIALSET}(X, x, n)$ 
     $\text{top-rep} \leftarrow \bigcup_{x \in X} \text{fs-set}[x]$     # We recommend the probabilistic t-conorm

    # Perform optimization of the graph layout
     $Y \leftarrow \text{SPECTRALEMBEDDING}(\text{top-rep}, d)$ 
     $Y \leftarrow \text{OPTIMIZEEMBEDDING}(\text{top-rep}, Y, \text{min-dist}, \text{n-epochs})$ 

    return  $Y$ 

```

---

**Gambar 3.3.** Algoritma UMAP

Saat melakukan penggabungan *fuzzy* dari himpunan *simplicial fuzzy* lokal, hal yang paling efektif adalah menggunakan probabilitas *t-conorm*. *Input* untuk algoritma UMAP adalah :  $X$ , dataset yang dimensinya akan dikurangi;  $n$ , ukuran lingkup yang akan digunakan metrik lokal;  $d$ , target dimensi data setelah dikurangi; *min-dist*, parameter algoritmik yang mengontrol tata letak; dan *n-epoch*, parameter yang mengontrol jumlah pekerjaan optimalisasi yang harus dilakukan.

Selanjutnya, fungsi individu untuk membangun himpunan simplisial *fuzzy* lokal, menentukan *spectral embedding*, dan mengoptimalkan penyisipan terkait dengan himpunan *fuzzy cross entropy*, dijelaskan lebih rinci di bawah.

---

**Algorithm 2** Constructing a local fuzzy simplicial set

---

```

function LOCALFUZZYSIMPLICIALSET( $X, x, n$ )
  knn, knn-dists  $\leftarrow$  APPROXNEARESTNEIGHBORS( $X, x, n$ )
   $\rho \leftarrow$  knn-dists[1]                                # Distance to nearest neighbor
   $\sigma \leftarrow$  SMOOTHKNNDIST(knn-dists,  $n, \rho$ )        # Smooth approximator to
  knn-distance
  fs-set0  $\leftarrow X$ 
  fs-set1  $\leftarrow \{([x, y], 0) \mid y \in X\}$ 
  for all  $y \in$  knn do
     $d_{x,y} \leftarrow \max\{0, \text{dist}(x, y) - \rho\} / \sigma$ 
    fs-set1  $\leftarrow$  fs-set1  $\cup ([x, y], \exp(-d_{x,y}))$ 
  return fs-set

```

---

**Gambar 3.4.** Algoritma Pembuatan himpunan simplisial *fuzzy* lokal

Pada Gambar 3.4 algoritma 2 menjelaskan konstruksi himpunan simplisial *fuzzy* lokal. Untuk merepresentasikan himpunan simplisial *fuzzy*, kami bekerja dengan citra himpunan fuzzy 0 dan 1, yang kami nyatakan sebagai fs-set0 dan fs-set1. Satu bisa bekerja dengan simplisitas tingkat tinggi juga, tetapi implementasi tersebut tidak digunakan saat ini. Kita dapat membangun himpunan simplisial *fuzzy* lokal ke titik tertentu  $x$  dengan menemukan  $n$  nearest neighbor, menghasilkan normalisasi yang sesuai jarak pada manifold, dan kemudian mengubah ruang metrik hingga  $a$  himpunan sederhana melalui fungsi *FinSing*, yang dalam kasus ini diterjemahkan menjadi eksponensial dari jarak negatif.

---

**Algorithm 3** Compute the normalizing factor for distances  $\sigma$

---

```

function SMOOTHKNNDIST(knn-dists,  $n, \rho$ )
  Binary search for  $\sigma$  such that  $\sum_{i=1}^n \exp(-(knn-dists_i - \rho) / \sigma) = \log_2(n)$ 
  return  $\sigma$ 

```

---

**Gambar 3.5.** Algoritma Penghitungan Normalisasi

Pada Gambar 3.5 algoritma 3 menjelaskan penghitungan normalisasi. Daripada langsung menggunakan jarak ke- $n^{\text{th}}$  *nearest neighbor* sebagai normalisasi, kami menggunakan jarak KNN *smoothed version* yang memperbaiki kardinalitas himpunan fuzzy 1-*simplices* ke nilai tetap.  $\log_2(n)$  dipilih untuk tujuan ini berdasarkan percobaan empiris.

---

**Algorithm 4** Spectral embedding for initialization

---

```

function SPECTRALEMBEDDING(top-rep, d)
   $A \leftarrow$  1-skeleton of top-rep expressed as a weighted adjacency matrix
   $D \leftarrow$  degree matrix for the graph  $A$ 
   $L \leftarrow D^{1/2}(D - A)D^{1/2}$ 
  evec  $\leftarrow$  Eigenvectors of  $L$  (sorted)
   $Y \leftarrow$  evec[1.. $d + 1$ ] # 0-base indexing assumed
  return  $Y$ 

```

---

**Gambar 3.6.** Algoritma *Spectral Embedding* untuk Permulaan

Pada Gambar 3.6 algoritma 4 menjelaskan *spectral embedding*. *Spectral embedding* dilakukan dengan mempertimbangkan salah satu kerangka dari representasi topologi *fuzzy* global sebagai grafik berbobot dan menggunakan metode spektral standar pada Laplacian yang dinormalisasi simetris.

Komponen utama terakhir dari UMAP adalah optimalisasi *embedding* melalui minimalisasi himpunan *fuzzy cross entropy* seperti yang dijelaskan oleh algoritma 5 pada Gambar 3.7.

---

**Algorithm 5** Optimizing the embedding

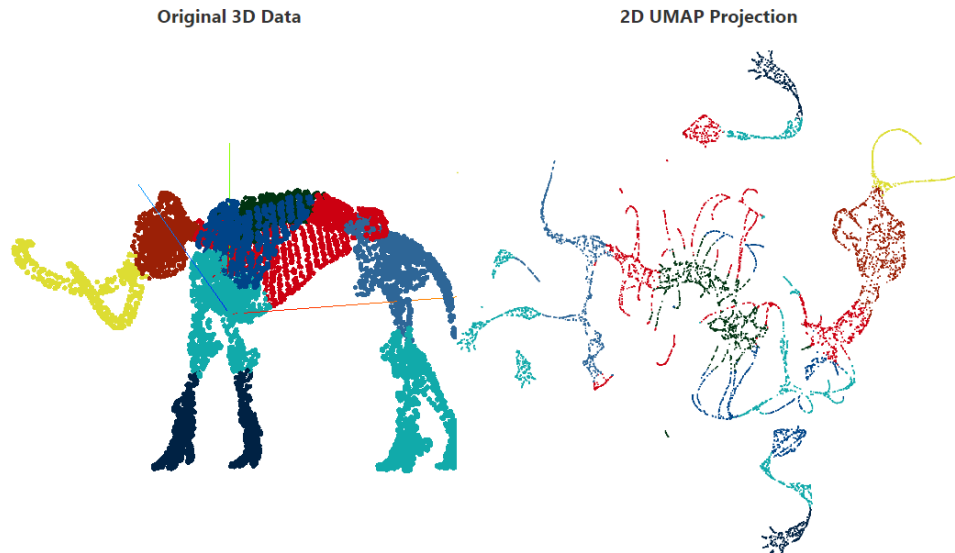
---

**function** OPTIMIZEEMBEDDING(top-rep,  $Y$ , min-dist, n-epochs) $\alpha \leftarrow 1.0$ Fit  $\Phi$  from  $\Psi$  defined by min-dist**for**  $e \leftarrow 1, \dots, \text{n-epochs}$  **do**    **for all**  $([a, b], p) \in \text{top-rep}_1$  **do**        **if** RANDOM( )  $\leq p$  **then**      # Sample simplex with probability  $p$              $y_a \leftarrow y_a + \alpha \cdot \nabla(\log(\Phi))(y_a, y_b)$         **for**  $i \leftarrow 1, \dots, \text{n-neg-samples}$  **do**             $c \leftarrow \text{random sample from } Y$              $y_a \leftarrow y_a + \alpha \cdot \nabla(\log(1 - \Phi))(y_a, y_c)$  $\alpha \leftarrow 1.0 - e/\text{n-epochs}$ **return**  $Y$ 

---

**Gambar 3.7.** Algoritma Optimalisasi *Embedding*

Berikut contoh visualisasi data berdimensi tinggi (3 dimensi) yang telah di reduksi dimensinya menjadi data dengan dimensi yang lebih rendah (2 dimensi), dapat dilihat pada Gambar 3.8.

**Gambar 3.8.** Contoh Visualisasi Data 3 Dimensi Menjadi 2 Dimensi



Dengan metode diatas maka data ulasan yang berbentuk vektor dengan dimensi tinggi dapat diubah menjadi data dengan dimensi yang lebih rendah sehingga data ulasan yang ada dapat dilanjutkan menuju proses *clustering*.

### 3.4.3 HDBSCAN

Setelah vektor dari data ulasan memiliki dimensi yang lebih rendah, data tersebut bisa untuk diproses dengan algoritma *clustering* agar kata-kata dalam data ulasan bisa dikelompokkan sesuai dengan kelompoknya. Pada penelitian kali ini algoritma *clustering* yang akan digunakan adalah HDBSCAN (*Hierarchical Density Based Spatial Clustering of Application with Noise*). HDBSCAN bekerja berdasarkan kepadatan persebaran data dan juga HDBSCAN lebih efektif dalam memproses data yang kurang terstruktur karena HDBSCAN meninggalkan *noise* pada proses *clustering* data.

#### Algorithm 1. HDBSCAN main steps

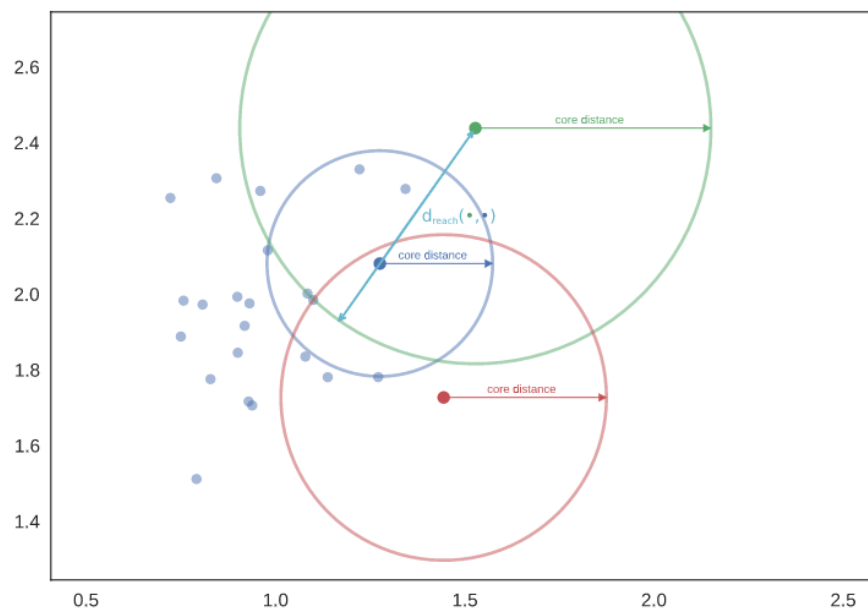
1. Compute the core distance w.r.t.  $m_{pts}$  for all data objects in  $\mathbf{X}$ .
2. Compute an MST of  $G_{m_{pts}}$ , the Mutual Reachability Graph.
3. Extend the MST to obtain  $MST_{ext}$ , by adding for each vertex a “self edge” with the core distance of the corresponding object as weight.
4. Extract the HDBSCAN hierarchy as a dendrogram from  $MST_{ext}$ :
  - 4.1 For the root of the tree assign all objects the same label (single “cluster”).
  - 4.2 Iteratively remove all edges from  $MST_{ext}$  in decreasing order of weights (in case of ties, edges must be removed simultaneously):
    - 4.2.1 Before each removal, set the dendrogram scale value of the current hierarchical level as the weight of the edge(s) to be removed.
    - 4.2.2 After each removal, assign labels to the connected component(s) that contain(s) the end vertex(-ices) of the removed edge(s), to obtain the next hierarchical level: assign a new cluster label to a component if it still has at least one edge, else assign it a null label (“noise”).

**Gambar 3.9.** Algoritma Utama HDBSCAN

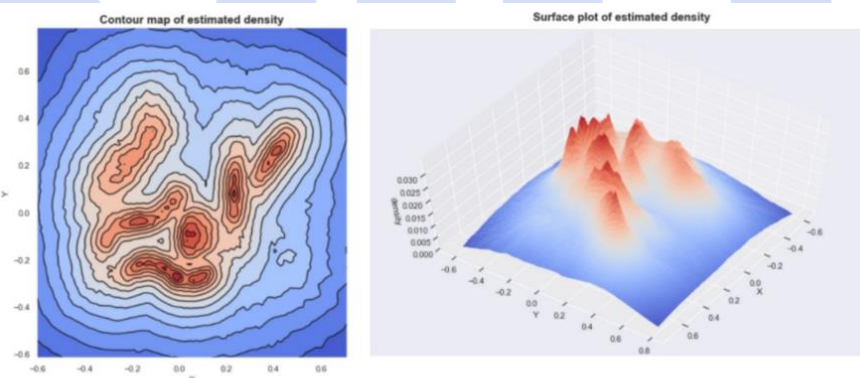
Tahapan dari proses HDBSCAN dapat dilihat pada algoritma di Gambar 3.9, Berikutnya algoritma HDBSCAN akan dijelaskan secara lebih sederhana dan disertai ilustrasi untuk lebih mudah dimengerti.

Data yang telah berbentuk vektor berdimensi rendah akan di-*input* kedalam algoritma HDBSCAN, pertama HDBSCAN akan memperkirakan kepadatan di sekitar titik-titik tertentu pada data. Salah satu cara untuk melakukan hal ini adalah dengan menggunakan “*Core Distance*” [28] yaitu mengukur jarak suatu titik

menuju tetangga terdekat ke- $K$ . Titik dengan daerah yang lebih padat akan memiliki "Core Distance" yang lebih kecil dan titik dengan 9 daerah yang lebih renggang akan memiliki "Core Distance" yang lebih besar itulah yang membuat algoritma ini menjadi "Density Based" [36]. Contoh penggunaan "Core Distance" dalam algoritma HDBSCAN dapat dilihat pada gambar 3.10. Setelah metode ini diaplikasikan pada semua titik maka akan terlihat persebaran kepadatan pada data seperti terlihat pada Gambar 3.11.



**Gambar 3.10.** Core Distance dalam Algoritma HDBSCAN

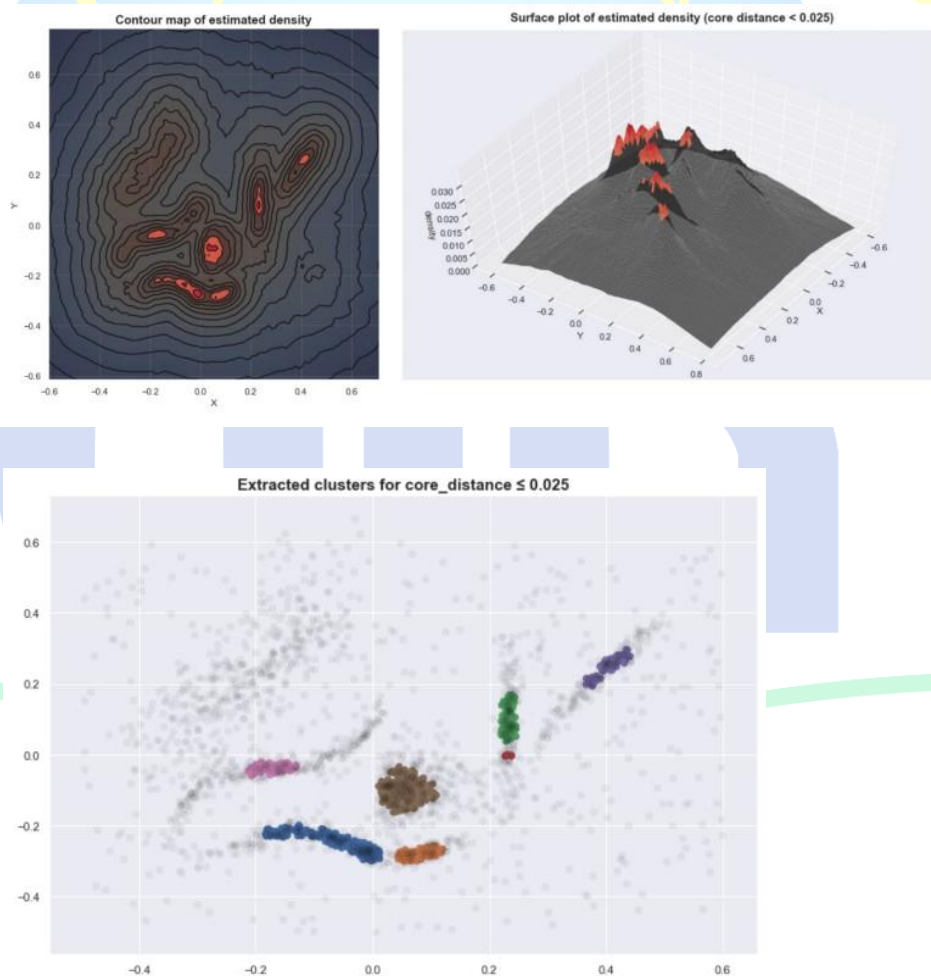


*Estimated densities from our sample data set*

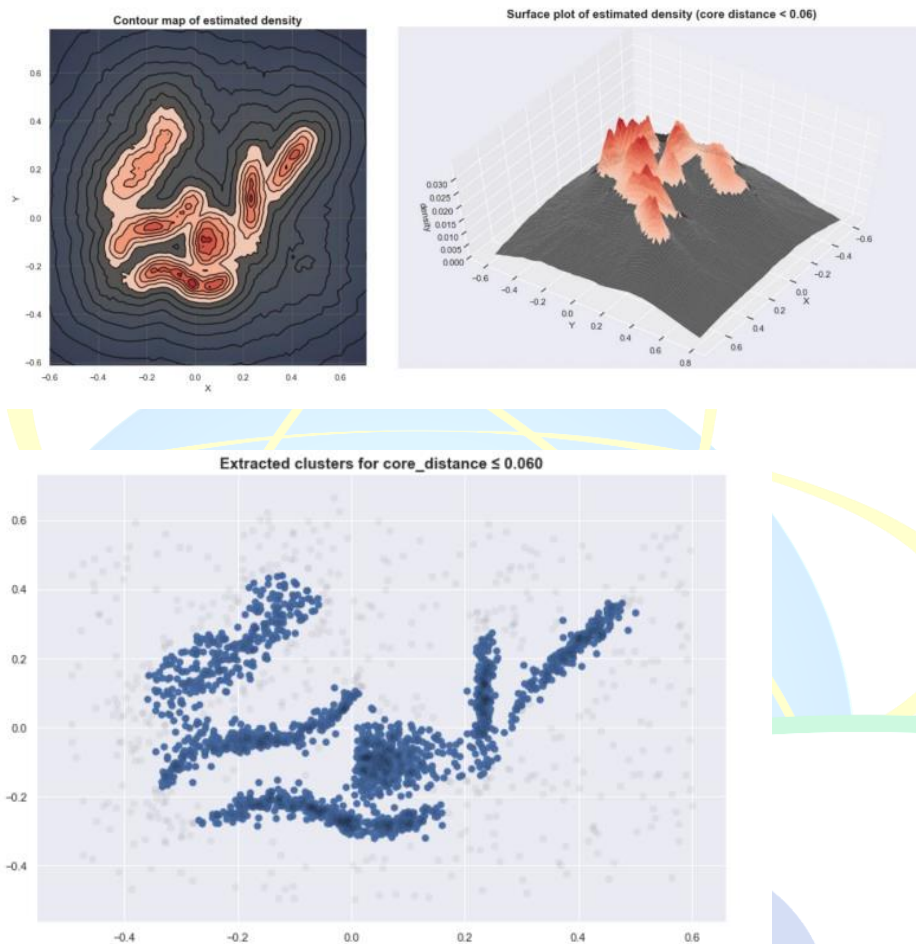
**Gambar 3.11.** Persebaran Kepadatan



Tahapan selanjutnya adalah mengekstrak *cluster* dari data yang telah terbentuk persebaran kepadatan datanya. Salah satu cara untuk mengekstrak sebuah *cluster* adalah dengan mengambil ambang batas global untuk semua *cluster*. Dengan mendapatkan titik yang kepadatannya diatas ambang batas dan menggabungkan titik-titik tersebut maka terciptalah *cluster-cluster* dari data yang dimiliki. Namun hal ini memiliki beberapa kekurangan, jika ambang batas yang ditetapkan terlalu tinggi maka akan terlalu banyak titik yang dianggap sebagai *noise* seperti pada Gambar 3.12 begitu juga sebaliknya, jika ambang batas yang ditetapkan terlalu rendah maka semua titik pada data akan dianggap menjadi satu *cluster* seperti pada Gambar 3.13. Cara Ini adalah pendekatan yang dilakukan oleh DBSCAN (*Density Based Spatial Clustering of Application with Noise*) [36].



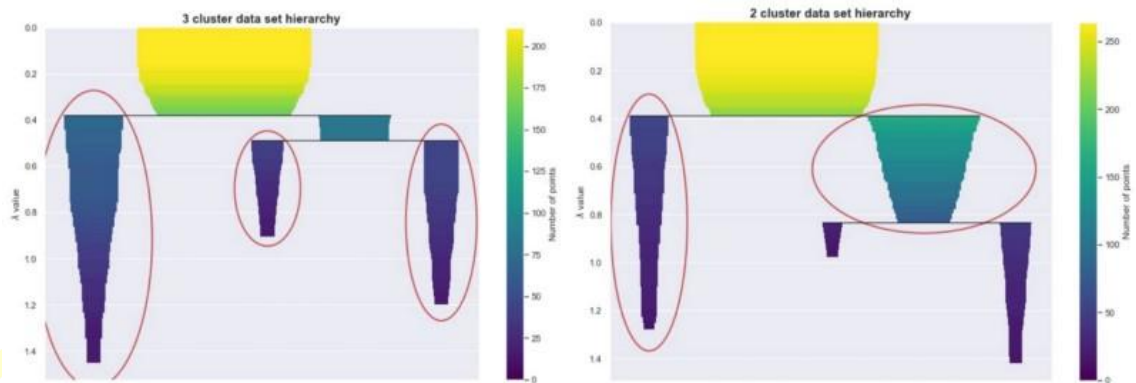
**Gambar 3.12.** Contoh Penetapan *Threshold* yang Terlalu Tinggi



**Gambar 3.13.** Contoh Penetapan *Threshold* yang Terlalu Rendah

Untuk menanggulangi hal tersebut HDBSCAN akan membuat hierarki untuk menemukan titik mana yang akhirnya bergabung menjadi satu *cluster*. Untuk memilah titik mana yang akan ‘bertahan’, HDBSCAN menggunakan “*Cluster Stability*” [28] untuk menentukannya. Ketika dua titik yang memiliki alas yang sama, untuk menentukan apakah masing-masing titik tersebut adalah sebuah *cluster* adalah dengan melihat perbandingan volume alas dan puncaknya. Ketika kedua titik tadi merupakan dua *cluster* maka volume dari dua titik tersebut akan lebih besar dari pada alasnya, Namun, ketika dua titik tersebut sebenarnya adalah fitur dari sebuah *cluster* maka volume alasnya akan lebih besar dari pada volume dua titik tersebut seperti pada Gambar 3.14. Dengan cara tersebut, HDBSCAN dapat

memutuskan apakah *cluster* akan dibagi menjadi *subcluster* berdasarkan titik yang ada atau tidak. Dengan metode tersebut data ulasan yang ada telah dibagi menjadi *cluster*.



**Gambar 3.14. Core Stability**

#### 3.4.4 Class Based TF-IDF

Setelah data terbagi menjadi *cluster*, tahap selanjutnya adalah merepresentasikan topik yang terbentuk dalam setiap *cluster*. Data ulasan yang ada telah menjadi beberapa *cluster* masih berbentuk data vektor, untuk menentukan topik apa yang ada dalam sebuah *cluster*, data vektor harus diubah menjadi kumpulan token kata dan setiap *cluster* akan menjadi sebuah “Bag of Word” dengan cara tokenisasi menggunakan modul “CountVectorizer”. Setelah semua *cluster* berubah menjadi kumpulan kata, barulah kita akan mengekstrak topik dari tiap *cluster* menggunakan algoritma *c-TF-IDF* [10].

Bobot akan diberikan pada setiap kata menggunakan rumus *c-TF-IDF* (2.2) dimana setiap kata akan dibandingkan kepentingannya diantara dokumen yang ada. Semakin suatu kata identik dengan suatu *cluster* dan tidak untuk *cluster* yang lain maka kata tersebut akan memiliki bobot yang tinggi dalam *cluster* tersebut. Dari sini data ulasan sudah berubah menjadi *cluster-cluster* yang memiliki representasi topik dan bisa diinterpretasi pada setiap *cluster*-nya.

### 3.5 *Guided Topic Modelling*

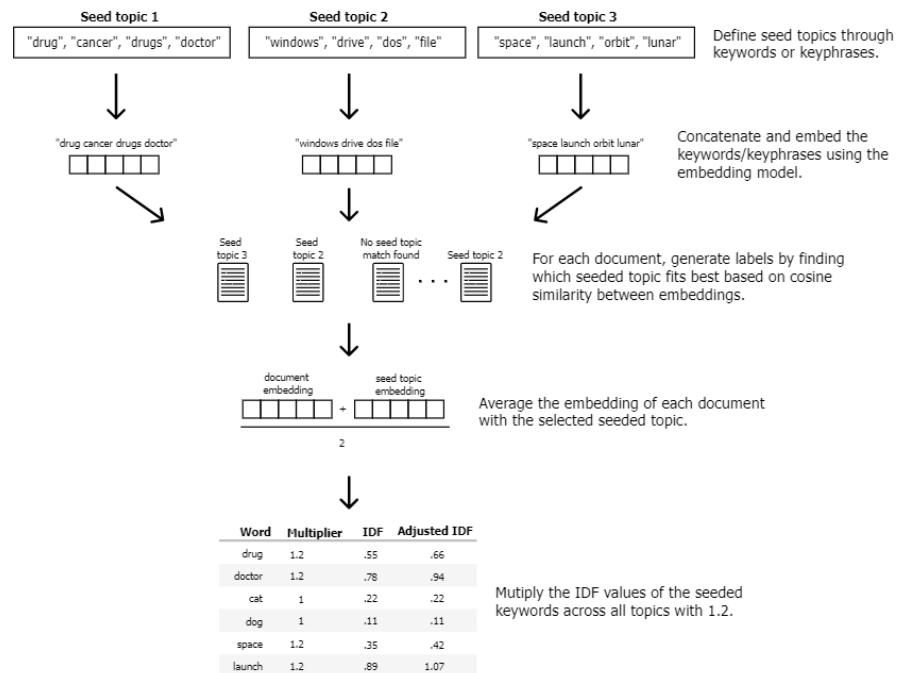
“*Guided Topic Modelling*” atau pemodelan topik terarah merupakan kumpulan teknik memandu pendekatan pemodelan topik dengan menetapkan sejumlah “*Seeded Topic*” atau topik unggulan pada permulaan proses sehingga topik yang akan dihasilkan akan memiliki kecenderungan dengan topik yang sudah ditetapkan diawal [37].

Sebagai contoh, penerapan “*Guided Topic Modelling*” dapat dilakukan pada sebuah aplikasi, misal diketahui ada sebuah masalah pada sistem *login* pada aplikasi tersebut. Untuk mengetahui keluhan-keluhan pengguna mengenai masalah tersebut kita dapat memodelkan topik dari *feedback* pengguna dengan menetapkan beberapa *seeded topic* terlebih dahulu seperti, “*login*”, “*bug*”, “*password*” dan “*username*” agar topik yang muncul berkaitan dengan kata-kata tersebut sehingga didapatkan gambaran yang lebih jelas mengenai kata-kata dalam *seeded topic* yang telah ditetapkan. Pada penelitian ini *seed topic* yang akan digunakan di tentukan dengan analisa awal terhadap masalah yang sering muncul dalam sebuah *video game* online, sehingga *seed topic* yang digunakan pada penelitian ini adalah :

1. *Seed topic 1 : hack, cheat, region, lock*
2. *Seed topic 2 : server, network, connect, ping*
3. *Seed topic 3 : bug, lag, crash, glitch*
4. *Seed topic 4 : microtransaction, buy, skin, crate*

Dalam *BERTopic*, fitur “*Guided Topic Modelling*” bekerja dengan dua tahap. Pertama, semua *seeded topic* akan digabungkan dan di-*input* kedalam algoritma *sentence embedding* untuk mendapatkan *embedding* dari *seeded topic* yang telah ditentukan. Kemudian *embedding* tersebut akan dibandingkan dengan hasil *embedding* dari dokumen yang ada menggunakan “*Cosinus Similarity*” jika sebuah dokumen dinilai mirip dengan *seeded topic* maka dokumen tersebut akan diberi label topik tersebut, jika sebuah dokumen lebih mirip dengan rata-rata *embedding* dokumen maka dokumen tersebut akan diberi label -1. Semua dokumen yang memiliki label *seeded topic* akan diteruskan menuju UMAP untuk membuat pendekatan *semi-supervised* [37].

Kedua, semua kata dalam *seeded topic* akan dipasangkan dengan pengali yang lebih besar dari 1. Hal itu akan membuat nilai IDF dari kata tersebut meningkat pada semua topik (*cluster*) sehingga akan meningkatkan kemungkinan kata-kata dalam *seeded topic* akan muncul pada representasi suatu topik [37].



**Gambar 3. 15** Alur Proses Metode "*Guided Topic Modelling*"

### 3.6 Visualisasi dan Interpretasi

Tahap akhir dari penelitian ini adalah melakukan visualisasi dari topik yang berhasil diekstrak dari data ulasan dan menginterpretasikannya. Setelah data ulasan di proses menggunakan prosedur *BERTopic* maka akan terlihat topik apa saja yang dibicarakan pemain dalam ulasan tersebut. Untuk mempermudah analisa, kata-kata dalam setiap topik akan divisualisasikan dengan probabilitasnya. Selain kata dalam topik, akan dilakukan juga visualisasi persebaran *cluster* dalam data ulasan yang kita miliki untuk melihat seberapa baik algoritma *clustering* bekerja. Visualisasi dari hasil yang didapatkan akan mempermudah proses interpretasi topik dan pengimplementasiannya pada game tersebut.

## BAB IV

### HASIL DAN PEMBAHASAN

Pada bab ini akan ditampilkan dan jelaskan hasil dari proses pemodelan topik yang telah dilakukan, dari mulai pengolahan data, performa dari algoritma pemodelan pemodelan topik yang dipakai dan juga analisa dari hasil pemodelan topiik pada penelitian kali ini.

#### 4.1 Pengolahan Data

Data ulasan dari pemain tentang game PUBG yang didapatkan tersimpan dalam format csv. Data yang sudah dalam format csv selanjutnya akan masuk ke dalam tahap *preprocessing*, karena data yang digunakan merupakan data teks yang tidak terstruktur maka *preprocessing* yang dilakukan adalah *case folding*, *remove number*, *remove emoticons*, dan *remove stopwords*. Hasil tahap *preprocessing* seperti pada Tabel 4.1.

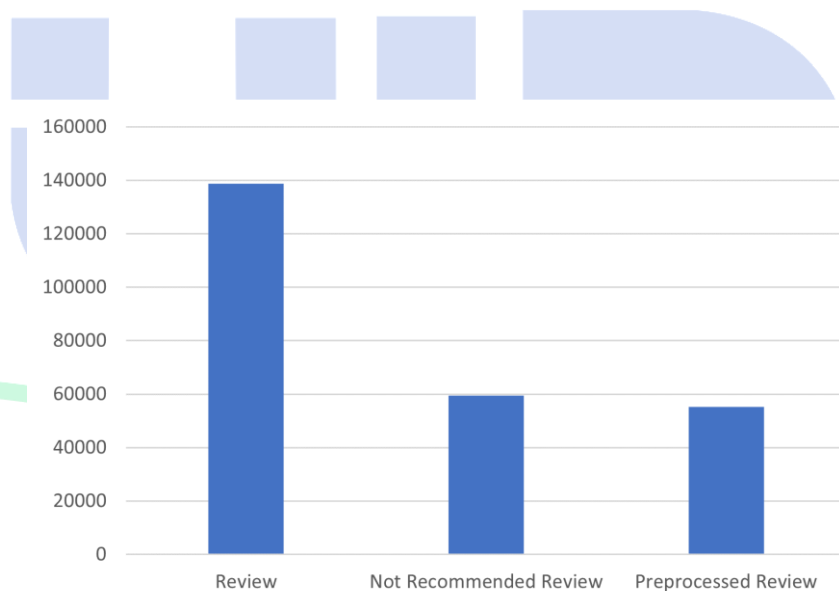
**Tabel 4.1.** Hasil *Preprocessing* Data Ulasan Game Battle Royale

Review	Preprocessed Review
This is the end of Early Acces journey. This was my first BattleRoyale experience. I had a lot of fun in this game. Developers worked very hard to make game better and testers didn't leave them alone. Thanks for all Bluehole! You guys did good job in a short time. I hope everything is gonna be better.	this the end of early acces journey this was my first battleroyale experience i had a lot of fun in this game developers worked very hard to make game better and testers didn t leave them alone thanks for all bluehole you guys did good job in a short time i hope everything is gonna better
game that makes me drown in adrenaline. Blood rushing through your brain. Can't breathe after every single fight. FPP mode is amazing. The only cons is performance but well it's only in beta. So, for me it's okay.	game that makes me drown in adrenaline blood rushing through your brain can t breathe after every single fight fpp mode is amazing the only cons is performance but well it s only in beta so for me it s okay



Can't wait until the graphics settings gets optimised...as of now not even a GTX 1080 can run the highest settings smoothly.	can t wait until the graphics settings gets optimised as of now not even a gtx can run the highest settings smoothly
we need Death playbackwe need Death playbackwe need Death playbackI know it's not fair, but you can show it after the whole team dies. then we will know who is the hacker.	we need death playbacki know it s not fair but you can show it after the whole team dies then we will know who is the hacker

Pada Tabel 4.1 terlihat bahwa ada perbedaan pada data ulasan sebelum dan sesudah dilakukan *preprocessing*. Sebelum *preprocessing* terlihat banyak ulasan yang mengandung emoji, kata-kata berulang, serta tanda baca. Hal itu terhapus setelah di *preprocessing*. Proses *case folding* juga terlihat berhasil karena pada kolom tweet yang telah dibersihkan sudah tidak ada kata yang menggunakan huruf kapital. Menghilangkan karakter tidak valid dan angka juga telah dilakukan pada proses *preprocessing*.

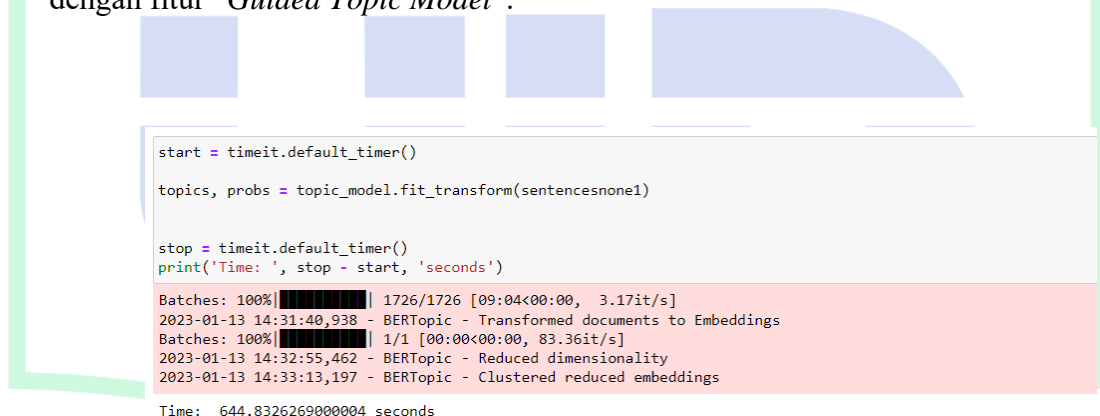


**Gambar 4.1** Perbandingan Jumlah Kata Sebelum dan Sesudah *Preprocessing*

Selanjutnya Gambar 4.1 menunjukkan histogram perbedaan jumlah ulasan pada data sebelum dan sesudah *preprocessing*. Terlihat perbedaan jumlah ulasan pada data yang menunjukkan bahwa jumlah ulasan yang muncul pada histogram setelah *preprocessing* jauh lebih sedikit dibanding yang sebelum di *preprocessing*, jumlah ulasan yang awalnya sebanyak 138.725 baris berkurang menjadi 55.203 baris. Hal tersebut terjadi dikarenakan tahap *preprocessing* yang menghapus kata-kata dan komponen lain yang tidak dibutuhkan, dan banyak dari ulasan hanya berisi komponen-komponen yang tidak dibutuhkan sehingga sebagian data ulasan dihapus.

## 4.2 Evaluasi Performa Model

Pada subbab ini akan di perlihatkan bagaimana performa yang dihasilkan dari algoritma *BERTopic* dengan fitur “*Guided Topic Modelling*” dalam mengekstrak topik dari data ulasan pemain terkait suatu *video game*. Performa dari algoritma *BERTopic* akan dievaluasi dengan melihat seberapa cepat algoritma menyelesaikan tugas yang diberikan dan seberapa besar “*Coherence Score*” yang dapat dihasil kan oleh algoritma tersebut. Data ulasan yang telah melewati tahap *preprocessing* data akan di-*input* kedalam algoritma *BERTopic* yang dilengkapi dengan fitur “*Guided Topic Model*”.



**Gambar 4.2** Runningtime Algoritma *BERTopic* dengan "*Guided Topic Modelling*"

Dapat dilihat pada Gambar 4.2 perolehan running time yang didapatkan algoritma *BERTopic*. Algoritma *BERTopic* dengan fitur “*Guided Topic Modelling*” dijalankan menggunakan perangkat dengan spesifikasi *processor* Core I5 10400f, VGA Nvidia GTX 1650, RAM 8GB dan VRAM 4GB menghasilkan running time



selama 10.7 menit. Waktu tersebut tergolong cukup lama untuk sebuah algoritma pemodelan topik, namun hal tersebut bisa diwajarkan mengingat algoritma *BERTopic* yang menggunakan dasar arsitektur neural network.

```
# Evaluate
coherence_model = CoherenceModel(topics=topic_words,
                                  texts=tokens,
                                  corpus=corpus,
                                  dictionary=dictionary,
                                  coherence='c_v')
coherence = coherence_model.get_coherence()
print(coherence)
```

0.5244758866170298

**Gambar 4.3** *Coherence Score* Algoritma *BERTopic* dengan "*Guided Topic Modelling*"

Kemudian pada Gambar 4.3 memperlihatkan coherence score yang diperoleh algoritma *BERTopic* dengan fitur "*Guided Topic Modelling*" dalam memproses data ulasan dari pemain suatu *video game*. Algoritma *BERTopic* berhasil mendapatkan coherence score yang terbilang cukup bagus untuk sebuah algoritma pemodelan topik yaitu **0,52** yang berarti kumpulan kata-kata dalam topik yang dihasilkan oleh algoritma *BERTopic* memiliki keterkaitan yang cukup tinggi dan relatif mudah untuk diinterpretasikan oleh manusia menjadi sebuah narasi.

Dengan demikian dapat diambil kesimpulan penggunaan fitur "*Guided Topic Modelling*" pada algoritma *BERTopic* sedikit menurunkan performa dari *BERTopic* itu sendiri. Pada Namun penurunan peforma ini tidak begitu signifikan untuk mengurangi keakuratan dari algoritma itu sendiri.

### 3.5 Analisa Hasil

Pada subbab ini akan dilakukan analisa terhadap kumpulan kata-kata dalam topik yang dihasilkan dari proses algoritma *BERTopic* dengan fitur "*Guided Topic Modelling*" pada data ulasan dari pemain pada sebuah *video game*. Proses algoritma *BERTopic* pada data ulasan pemain pada *video game* yang digeunakan pada penelitian ini menghasilkan 10 topik dari 55.203 ulasan negatif dari pemain.

```

empty_dimensionality_model = BaseDimensionalityReduction()
umap_model = UMAP(n_neighbors=10, n_components=10, min_dist=0.0, metric='cosine', random_state=42)
#umap_model = None

sentence_model = SentenceTransformer("all-mpnet-base-v2", device="cuda")

seed_topic_list = [
    ["hack", "cheat", "region", "lock"],
    ["server", "network", "connect", "ping"],
    ["bug", "lag", "crash", "glitch"],
    ["microtransaction", "buy", "skin", "crate"],
]

topic_model = BERTopic(
    embedding_model=sentence_model,
    umap_model=umap_model,
    vectorizer_model=vectorizer_model,
    seed_topic_list=seed_topic_list,
    top_n_words=10,
    min_topic_size=105,
    nr_topics=None,
    calculate_probabilities=True,
    verbose=True
)

```

**Gambar 4.4** Paramater dan *Seed Topic* Yang digunakan Selama Proses

Gambar 4.4 memperlihatkan parameter dan *seed topic* yang digunakan selama proses algoritma *BERTopic* berjalan. Dengan parameter yang diperlihatkan diatas algoritma *BERTopic* menghasilkan sebanyak 10 topik dari 55.203 data ulasan negatif yang ada. Topik-topik yang telah dihasilkan selanjutnya akan dianalisa sebagai berikut :

#### A. Topik 1

```
topic_model.get_topic(topic=0)
```

```

[('hackers', 0.023513313129902864),
 ('cheaters', 0.019580024889676115),
 ('chinese', 0.01748857262302646),
 ('region', 0.01568214646198426),
 ('lock', 0.01364471082898196),
 ('banned', 0.013355801511584714),
 ('china', 0.012114493648990008),
 ('many', 0.01177616203848011),
 ('bluehole', 0.01139276281335486),
 ('get', 0.011374755706979075)]

```

```
topic_model.get_representative_docs(0)
```

```

['oceania servers are garbage and never work. playing on any other server means unplayable lag. and t
hat s before we even start on the constant hackers and game crippling bugs.',
 'although it s buggy, unoptimised and makes me rage a helluva lot, i still love this game. nothing g
ets your heart pumping more than being in the last , surrounded by enemies and just crawling in grass
hoping you don t get spotted. new genre crawl simulator edit after a further or so hours i m now chan
ging my review. the game is now somewhat optimised and runs fairly smoothly on a high end pc, but now
the problem is hackers. every game you play i guarantee you, you will be faced with a hacker. it is a
bsolutely rampant, you can either get shot through a building from the other side of the map, or watc
h a guy trace you through walls till you re in line of sight to spray you down in the killcam. from w
hat it started out to be to how it is now is pretty sad and the onus lies completely on the devs for
not region locking and opting for a punkbuster style anti cheat program. don t waste your money on th
is trash.',
 'just sucks after hours. cant bring myself to play it much anymore. kinda just stagnated and didnt a
dd much to the game. also paid crates for a dollar game. lost to fortniteeveryone speaks chinese']

```

**Gambar 4.5** Kumpulan Kata dan Representasi Dokumen Topik 1

Gambar 4.5 memperlihatkan kumpulan kata yang muncul dalam topik 1. Kata yang muncul antara lain : *hacker*, *cheater*, *china*, *region*, dan lain-lain. Dari kata-kata tersebut bisa disimpulkan bahwa topik 1 berbicara terkait keluhan pemain yang sering menemui *hacker* dan *cheater* dalam game tersebut terutama *hacker* dari region china. Hal ini didukung juga oleh representasi dokumen yang muncul untuk topik 1 pada gambar 4.5 yang berisikan keluhan pemain tentang banyaknya *hacker* dan *cheater*. Topik ini juga berkaitan dengan *seed topic* pertama yang telah ditentukan sebelumnya

## B. Topik 2

```
topic_model.get_topic(topic=1)

[('servers', 0.08954760135956155),
 ('server', 0.08714971089747095),
 ('busy', 0.05471985967697865),
 ('servers busy', 0.04070615536384881),
 ('connection', 0.025060694837735747),
 ('fix', 0.023893603954138188),
 ('lag', 0.022972500128258334),
 ('connect', 0.02012379619353233),
 ('please', 0.019678804044869946),
 ('cant', 0.019082591682048896)]

topic_model.get_representative_docs(1)

['servers are trash.',
 'the servers are trash',
 'the content is rich, but the producer doesn t solve the server problem at all. almost every day the re is a problem with the server, and so far there is no improvement in appearance. junk attitude']
```

**Gambar 4.6** Kumpulan Kata dan Representasi Dokumen Topik 2

Selanjutnya pada gambar 4.6 menampilkan kumpulan kata yang muncul dalam topik 2. Kata yang muncul antara lain : *server*, *busy*, *connection*, *fix*, dan lain-lain. Dari kata-kata tersebut bisa disimpulkan bahwa topik 2 berbicara terkait keluhan pemain terhadap buruknya koneksi *server* yang dimiliki oleh *game battle royale* tersebut. Hal ini didukung juga oleh representasi dokumen yang muncul untuk topik 2 pada gambar 4.6 yang berisikan keluhan pemain tentang *server* yang selalu sibuk. Topik ini juga memiliki keterkaitan dengan *seed topic* kedua.

### C. Topik 3

```
topic_model.get_topic(topic=2)

[('crash', 0.08905732219004253),
 ('crashes', 0.07837977036641741),
 ('crashing', 0.07543344134591153),
 ('keeps', 0.02959871581411092),
 ('crashed', 0.026940043759798717),
 ('fix', 0.0259932927427986),
 ('cant', 0.025845078739799732),
 ('launch', 0.02495834356229798),
 ('error', 0.02338351556924288),
 ('always', 0.020969989731781557)]

topic_model.get_representative_docs(2)

['the games not working on my pc. game is stuck on the loading screen. i paid for this game and now i
t s not working. kindly help me with my situation',
 'always crash , trash game',
 'cant play it because it says im missing file privileges,bs']
```

**Gambar 4.7** Kumpulan Kata dan Representasi Dokumen Topik 3

Kemudian pada gambar 4.7 menampilkan kumpulan kata yang muncul dalam topik 3. Kata yang muncul antara lain : *keep*, *crashing*, *error*, *always* dan lain-lain. Dari kata-kata tersebut bisa disimpulkan bahwasannya topik 3 berbicara terkait keluhan pemain terhadap *video game* yang kerap kali mengalami *crash* dan *error* saat hendak dimainkan sehingga *game* tersebut tidak dapat dimainkan. Hal ini didukung juga oleh representasi dokumen yang muncul untuk topik 3 pada gambar 4.7 yang berisikan keluhan pemain tentang aplikasi *game* tersebut yang terjebak pada *loading screen* saat hendak dimainkan. Topik ini juga sesuai dengan *seed topic* ketiga yang diberikan sebelumnya.

### D. Topik 4

```
topic_model.get_topic(topic=3)

[('free', 0.15002196834981119),
 ('refund', 0.09671576819554605),
 ('now free', 0.08548306050926426),
 ('money back', 0.07412332734436906),
 ('back', 0.05632184070432244),
 ('money', 0.051912996067784274),
 ('now', 0.045927954205438916),
 ('free now', 0.04246832507486688),
 ('bought', 0.03966432371250631),
 ('paid', 0.036536953825838187)]

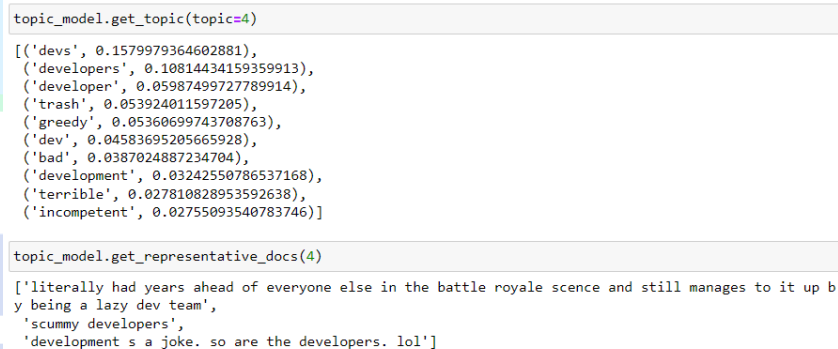
topic_model.get_representative_docs(3)

['being set up permanent feng ting account,just survive the jedi play eight hours. i need to protect
my interests. i m not buying one off games i need a refund. i don t need a one off game like this',
 'now its free to play ok i wasted my money',
 'give me refund bro i pay for this game now its free to play not fair bro']
```

**Gambar 4.8** Kumpulan Kata dan Representasi Dokumen Topik 4

Lalu pada gambar 4.8 menampilkan kumpulan kata yang muncul dalam topik 4. Kata yang muncul antara lain : *refund*, *money*, *back*, *now*, *free*, dan lain-lain. Dari kata-kata tersebut bisa disimpulkan bahwa topik 4 berbicara terkait kekesalan pemain yang merasa sia-sia telah mengeluarkan uangnya untuk membeli *game* tersebut karena *game* tersebut berubah menjadi *game free to play*. Hal ini didukung juga oleh representasi dokumen yang muncul untuk topik 4 pada gambar 4.8 yang berisikan tuntutan pemain yang meminta uangnya dikembalikan. Topik ini juga yang paling memiliki keterkaitan dengan *seed topic* keempat yaitu jual beli dalam *game* walaupun tidak memiliki kata-kata secara langsung.

#### E. Topik 5



```
topic_model.get_topic(topic=4)
[('devs', 0.1579979364602881),
 ('developers', 0.10814434159359913),
 ('developer', 0.05987499727789914),
 ('trash', 0.053924011597205),
 ('greedy', 0.05360699743708763),
 ('dev', 0.04583695205665928),
 ('bad', 0.0387024887234704),
 ('development', 0.03242550786537168),
 ('terrible', 0.027810828953592638),
 ('incompetent', 0.02755093540783746)]

topic_model.get_representative_docs(4)
['literally had years ahead of everyone else in the battle royale scene and still manages to it up b
y being a lazy dev team',
 'scummy developers',
 'development s a joke. so are the developers. lol']
```

**Gambar 4.9** Kumpulan Kata dan Representasi Dokumen Topik 5

Pada gambar 4.9 memperlihatkan kumpulan kata yang muncul dalam topik 5. Kata yang muncul antara lain : *developer*, *trash*, *greedy*, *incompetent*, dan lain-lain. Dari kata-kata tersebut bisa disimpulkan bahwa topik 5 berbicara terkait keluhan pemain yang merasa pihak pengembang dari *game battle royale* tersebut sangatlah buruk dan juga lebih mementingkan pendapatan dari pada memperhatikan keluhan pemain. Hal ini didukung juga oleh representasi dokumen yang muncul untuk topik 5 pada gambar 4.9 yang berisikan komentar pemain yang menyebut kan bahwa pengembang *game* ini adalah tim yang malas.

## F. Topik 6

```
topic_model.get_topic(topic=5)

[('fortnite', 0.4367312157091134),
 ('fortnite better', 0.24924084721888173),
 ('better', 0.13462480469995777),
 ('better fortnite', 0.12978752440462496),
 ('fortnite instead', 0.053102986636564806),
 ('fortnite free', 0.04861702162679914),
 ('go fortnite', 0.043065235362500356),
 ('free', 0.038276971548063844),
 ('like fortnite', 0.03087440134866386),
 ('instead', 0.03027300349001638)]

topic_model.get_representative_docs(5)

['more lag and bugs than gameplay, but still better than fortnite.',
 'fortnite is way better they actually give about their community and this game a running sim.',
 'thank you for giving us fortnite.']
```

**Gambar 4.10** Kumpulan Kata dan Representasi Dokumen Topik 6

Lalu pada gambar 4.10 menampilkan kumpulan kata yang muncul dalam topik 6. Kata yang muncul antara lain : *fortnite*, *better*, *go fortnite*, *like*, dan lain-lain. Dari kata-kata tersebut bisa disimpulkan bahwa topik 6 berbicara terkait komentar pemain yang membanding-bandingkan *game battle royale* ini dengan *game* lain yang memiliki mekanisme serupa. Hal ini didukung juga oleh representasi dokumen yang muncul untuk topik 6 pada gambar 4.10 yang berisikan komentar pemain yang mengatakan *game* lain yang memiliki mekanisme serupa lebih baik dari pada *game* ini.

## G. Topik 7

```
topic_model.get_topic(topic=6)

[('lag', 0.11462617078822018),
 ('laggy', 0.06353856598270521),
 ('lags', 0.05567164802150287),
 ('fps', 0.04793060921737142),
 ('lagging', 0.04739245337263478),
 ('update', 0.04478675774178083),
 ('fix lag', 0.038509125406898215),
 ('fix', 0.036194233456886056),
 ('stuttering', 0.03315692180364999),
 ('gtx', 0.028316954469497172)]

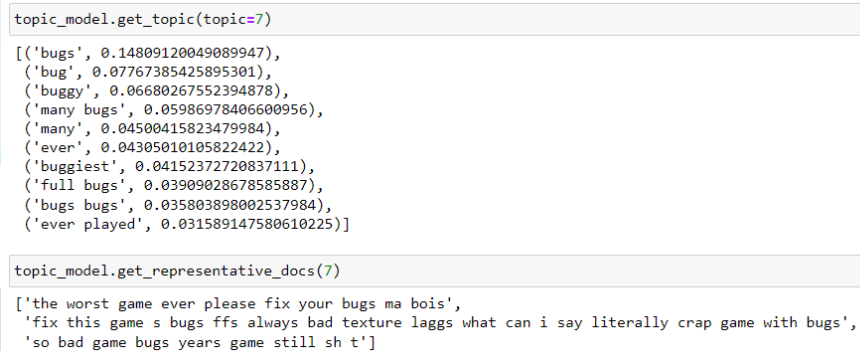
topic_model.get_representative_docs(6)

['too much lag', 'sever, always in lagging condition', 'sanhok map is too lag']
```

**Gambar 4.11** Kumpulan Kata dan Representasi Dokumen Topik 7

Selanjutnya pada gambar 4.11 menampilkan kumpulan kata yang muncul dalam topik 7. Kata yang muncul antara lain : *lag*, *fps*, *fix*, *stuttering* dan lain-lain. Dari kata-kata tersebut bisa disimpulkan bahwa topik 7 berbicara terkait keluhan pemain terhadap performa *game* yang buruk sehingga *game* tidak nyaman saat dimainkan. Hal ini didukung juga oleh representasi dokumen yang muncul untuk topik 7 pada gambar 4.11 yang berisikan keluhan pemain yang merasa *game* tersebut sangat sering mengalami *lag*. Topik ini juga memiliki keterkaitan dengan *seed topic* ketiga.

#### H. Topik 8



```
topic_model.get_topic(topic=7)

[('bugs', 0.14809120049089947),
 ('bug', 0.07767385425895301),
 ('buggy', 0.06680267552394878),
 ('many bugs', 0.05986978406600956),
 ('many', 0.04500415823479984),
 ('ever', 0.04305010105822422),
 ('buggiest', 0.04152372720837111),
 ('fix this game s bugs ffs always bad texture laggs what can i say literally crap game with bugs',
 ('bugs bugs', 0.035803898002537984),
 ('ever played', 0.031589147580610225)]

topic_model.get_representative_docs(7)

['the worst game ever please fix your bugs ma bois',
 'fix this game s bugs ffs always bad texture laggs what can i say literally crap game with bugs',
 'so bad game bugs years game still sh t']
```

**Gambar 4.12** Kumpulan Kata dan Representasi Dokumen Topik 8

Kemudian pada gambar 4.12 menampilkan kumpulan kata yang muncul dalam topik 8. Kata yang muncul antara lain : *bug*, *many*, *full bugs* dan lain-lain. Dari kata-kata tersebut bisa disimpulkan bahwa topik 8 berbicara terkait keluhan pemain terhadap *game* yang sangat banyak memiliki *bug* dan cacat dalam *game* tersebut. Hal ini didukung juga oleh representasi dokumen yang muncul untuk topik 8 pada gambar 4.12 yang berisikan keluhan pemain tentang banyaknya *bug* dalam *game* tersebut dan meminta pengembang untuk segera memperbaiki hal tersebut. Seperti topik 3 dan topik 7, topik 8 memiliki keterkaitan dengan *seed topic* ketiga.



## I. Topik 9

```
topic_model.get_topic(topic=8)

[('fortnite', 0.05857050835100012),
 ('sue', 0.055267099423256597),
 ('suing', 0.040812438984931684),
 ('epic', 0.030495179424640896),
 ('battle', 0.027523633060020192),
 ('lawsuit', 0.026337076796898513),
 ('instead', 0.024024368656642706),
 ('royale', 0.023212097674836688),
 ('battle royale', 0.022734388431408847),
 ('epic games', 0.02124819935421169)]

topic_model.get_representative_docs(8)

['suing another dev for gamemode uninstall',
 'this review has been sued',
 'i wrote this review but all i got was sued']
```

**Gambar 4.13** Kumpulan Kata dan Representasi Dokumen Topik 9

Selanjutnya pada gambar 4.13 menampilkan kumpulan kata yang muncul dalam topik 9. Kata yang muncul antara lain : *fortnite*, *sue*, *epic games*, *lawsuit*, dan lain-lain. Dari kata-kata tersebut bisa disimpulkan bahwa topik 9 berbicara terkait komentar pemain terhadap berita penuntutan atas pelanggaran hak cipta terhadap pengembang *game battle royale serupa* oleh pengembang *game battle royale* ini. Hal ini didukung juga oleh representasi dokumen yang muncul untuk topik 9 pada gambar 4.13 yang berisikan komentar pemain yang membahas tentang kasus tersebut.

## J. Topik 10

```
topic_model.get_topic(topic=9)

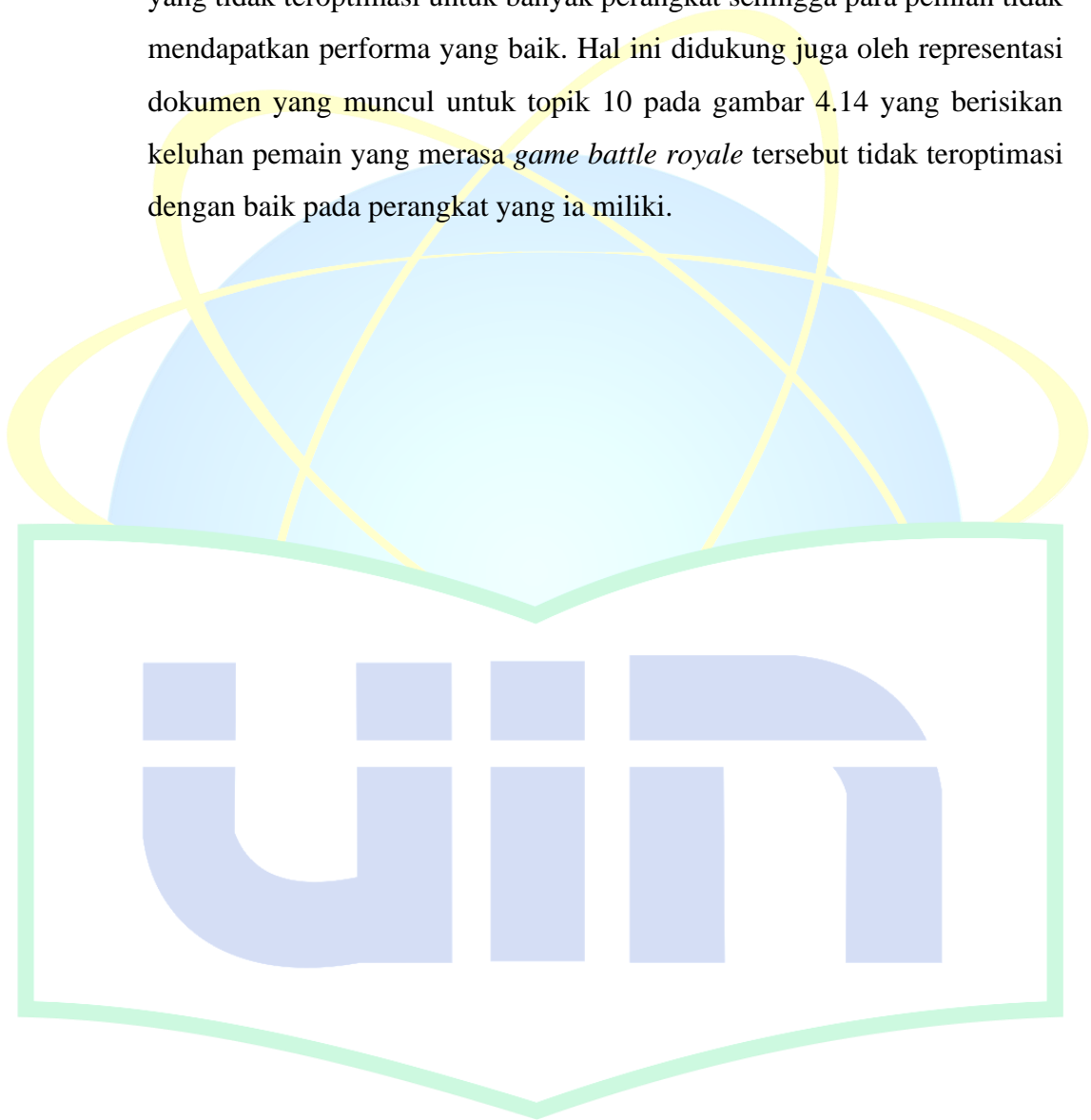
[('optimization', 0.3836420173677193),
 ('optimize', 0.20321581989228304),
 ('optimized', 0.1776210272509089),
 ('bad optimization', 0.12958308704182253),
 ('bad', 0.08927102647699453),
 ('optimised', 0.08890709633372412),
 ('optimisation', 0.08613330016833133),
 ('poorly optimized', 0.08488323552050991),
 ('poorly', 0.08266370725429441),
 ('poor', 0.07453067580891135)]

topic_model.get_representative_docs(9)

['poorly optimized, buy has great potential',
 'not good optimise with ti,',
 'un optimized piece of']
```

**Gambar 4.14** Kumpulan Kata dan Representasi Dokumen Topik 10

Terakhir pada gambar 4.14 menampilkan kumpulan kata yang muncul dalam topik 10. Kata yang muncul antara lain : *optimization*, *bad*, *poorly*, *optimized* dan lain-lain. Dari kata-kata tersebut bisa disimpulkan bahwa topik 9 berbicara terkait keluhan pemain terhadap *game battle royale* yang tidak teroptimasi untuk banyak perangkat sehingga para pemain tidak mendapatkan performa yang baik. Hal ini didukung juga oleh representasi dokumen yang muncul untuk topik 10 pada gambar 4.14 yang berisikan keluhan pemain yang merasa *game battle royale* tersebut tidak teroptimasi dengan baik pada perangkat yang ia miliki.



## BAB V

### KESIMPULAN DAN SARAN

Bab ini berisi tentang beberapa kesimpulan yang dapat diambil dari penelitian kali ini, serta memuat saran-saran yang bisa menjadi masukan untuk penelitian serupa di masa yang akan datang.

#### 5.1 Kesimpulan

Berdasarkan analisa terhadap hasil topik yang dapat diekstrak dari data ulasan pemain *game battle royale* yang digunakan pada penelitian ini menggunakan algoritma *BERTopic* dengan fitur “*Guided Topic Modelling*”, terdapat 10 topik dari 55.203 ulasan. Dari hasil analisa terhadap beberapa topik yang telah diekstrak, dapat disimpulkan bahwasannya banyak pemain PUBG yang mengeluhkan tantang banyaknya *hacker* dan *cheater* yang terdapat dalam game tersebut. Selain itu tidak sedikit pula pemain yang mengeluhkan tentang performa game tersebut pada *device* yang mereka miliki, baik itu dari sisi performa game itu sendiri dan juga dari sisi *server* yang dimiliki game tersebut dikarenakan game tersebut berbasis *online*. Kemudian, tidak sedikit juga diantara pemain yang merasa telah menyia-nyiakan uangnya untuk membeli game ini karena *game battle royale* tersebut sudah berubah menjadi *game free to play*. Dan juga ada pula pemain yang membahas tentang isu-isu pada industri *video game* yang berkaitan dengan game tersebut.

Dari segi performa, ketepatan yang dihasilkan oleh algoritma *BERTopic* dengan *Guided Topic Modelling* mendapatkan hasil yang cukup memuaskan, *coherence score* yang dihasilkan tergolong cukup baik di angka 0.5244758866 dengan 10 topik dari 55.203 ulasan. Sedangkan untuk kecepatan, *runningtime* yang dihasilkan adalah selama 10.7 menit. Hasil tersebut tampaknya sedikit lebih lama dari algoritma pemodelan topik lainnya.

## 5.2 Saran

Berdasarkan kesimpulan dari penelitian yang telah dilakukan diperoleh beberapa saran untuk penelitian serupa di masa yang akan datang yaitu :

1. Dalam penelitian ini algoritma pemodelan topik yang digunakan adalah *BERTopic* yang mana algoritma tersebut memiliki parameter yang sangat banyak dan beragam yang bisa disesuaikan untuk *dataset* yang ada, namun parameter yang digunakan pada penelitian ini lebih banyak menggunakan parameter *default*. Maka dari itu disarankan bagi penelitian selanjutnya memperdalam setiap parameter yang digunakan dalam *BERTopic* untuk mendapatkan *output* topik yang terbaik.
2. Penggunaan arsitektur *neural network* pada algoritma pemodelan topik membuat waktu yang dibutuhkan untuk memproses data menjadi lebih lama jika dibandingkan algoritma lain seperti LDA dan sebagainya. Dari hal tersebut disarankan bagi penelitian berikutnya untuk menemukan cara bagaimana algoritma *BERTopic* dan algoritma pemodelan topik yang menggunakan *neural network* lainnya dapat lebih efisien dalam segi waktu pemrosesan.
3. Terakhir, Peneliti menyarankan untuk penelitian serupa berikutnya agar bisa bekerja sama langsung dengan suatu studio pengembang game. Hal tersebut bertujuan untuk membuat penelitian seperti ini memiliki dampak yang nyata dan dapat diterapkan oleh studio pengembang game sesuai dengan apa yang terjadi dalam industri saat ini.

## DAFTAR PUSTAKA

- [1] New Zoo, "Global Games Market Report," 2020.
- [2] N. P. Bestari, "Wow! Tiga Tahun Lagi Pemain Game di RI Tembus 127 Juta Orang," 2022.
- [3] T. H. Apperley, "Genre and game studies: Toward a critical approach to video game genres," *Simul Gaming*, vol. 37, no. 1, pp. 6–23, Mar. 2006, doi: 10.1177/1046878105282278.
- [4] L. Caroux, K. Isbister, L. le Bigot, and N. Vibert, "Player-video game interaction: A systematic review of current concepts," *Computers in Human Behavior*, vol. 48. Elsevier Ltd, pp. 366–381, 2015. doi: 10.1016/j.chb.2015.01.066.
- [5] Steam, "PUBG : BATTLEGROUNDS Steam Chart," 2022.
- [6] E. Makuch, "PUBG Revenue Numbers Show How Much Bigger PUBG Mobile Is Than Console/PC," 2022.
- [7] C. GyuHyeok and K. Mijin, "Gameplay of Battle Royale Game by Rules and Actions of Play," *IEEE 7th Global Confrence on Consumer Electronics (GCCE 2018)*, 2018.
- [8] J. Kooistra, "THE SUCCESS OF BATTLE ROYALE GAMES BATTLE ROYALE: A MIX OF OLD AND NEW ELEMENTS," 2018.
- [9] S. I. Nikolenko, S. Koltcov, and O. Koltsova, "Topic modelling for qualitative studies," *J Inf Sci*, vol. 43, no. 1, pp. 88–102, Feb. 2017, doi: 10.1177/0165551515617393.
- [10] M. Grootendorst, "BERTopic: Neural topic modeling with a class-based TF-IDF procedure," Mar. 2022, [Online]. Available: <http://arxiv.org/abs/2203.05794>
- [11] J. Devlin Google and A. I. Language, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding (Bidirectional Encoder Representations from Transformers)."
- [12] M. Y. Febrianta *et al.*, "Analisis Ulasan Indie Video Game Lokal pada Steam Menggunakan Analisis Sentimen dan Pemodelan Topik Berbasis Latent Dirichlet Allocation."
- [13] V. Raju, B. Kumar Bolla, D. K. Nayak, and J. Kh, "Topic Modelling on Consumer Financial Protection Bureau Data: An Approach Using BERT Based Embeddings."
- [14] A. Kaushik and S. Naithani, "A Comprehensive Study of Text Mining Approach," 2016.

- [15] S. Dang and P. H. Ahmad, "Text Mining: Techniques and its Application Text Mining View project Text Mining: Techniques and its Application," *IJET/ International Journal of Engineering & Technology Innovations*, vol. 1, 2014.
- [16] I. H. Witten, "Text mining."
- [17] B. Zhao, "Web Scraping," in *Encyclopedia of Big Data*, Springer International Publishing, 2017, pp. 1–3. doi: 10.1007/978-3-319-32001-4\_483-1.
- [18] V. Gurusamy and A. Professor, "Preprocessing Techniques for Text Mining."
- [19] F. Kyoomarsi, H. Khosravi, E. Eslami, P. K. Dehkordy, and A. Tajoddin, "Optimizing text summarization based on fuzzy logic," in *Proceedings - 7th IEEE/ACIS International Conference on Computer and Information Science, IEEE/ACIS ICIS 2008, In conjunction with 2nd IEEE/ACIS Int. Workshop on e-Activity, IEEE/ACIS IWEA 2008*, 2008, pp. 347–352. doi: 10.1109/ICIS.2008.46.
- [20] S. N. Kane, A. Mishra, and A. Gaur, "Journal of Physics: Conference Series: Preface," *Journal of Physics: Conference Series*, vol. 534, no. 1. Institute of Physics Publishing, 2014. doi: 10.1088/1742-6596/534/1/011001.
- [21] A. Vaswani *et al.*, "Attention Is All You Need," Jun. 2017, [Online]. Available: <http://arxiv.org/abs/1706.03762>
- [22] M. Geva, R. Schuster, J. Berant, and O. Levy, "Transformer Feed-Forward Layers Are Key-Value Memories," Dec. 2020, [Online]. Available: <http://arxiv.org/abs/2012.14913>
- [23] H. Sch, "Introduction to Information Retrieval IIR 19: Web Search." [Online]. Available: <http://informationretrieval.org>
- [24] N. Reimers and I. Gurevych, "Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks," Aug. 2019, [Online]. Available: <http://arxiv.org/abs/1908.10084>
- [25] P. O. Box, L. van der Maaten, E. Postma, and J. van den Herik, "Tilburg centre for Creative Computing Dimensionality Reduction: A Comparative Review Dimensionality Reduction: A Comparative Review," 2009. [Online]. Available: <http://www.uvt.nl/ticc>
- [26] L. McInnes, J. Healy, and J. Melville, "UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction," Feb. 2018, [Online]. Available: <http://arxiv.org/abs/1802.03426>
- [27] P.-N. et al Tan, *Introduction to data mining*. Boston: Pearson Addison Wesley, 2006.
- [28] R. J. G. B. Campello, D. Moulavi, and J. Sander, "LNAI 7819 - Density-Based Clustering Based on Hierarchical Density Estimates."

- [29] T. Joachims, "A Probabilistic Analysis of the Rocchio Algorithm with TFIDF for Text Categorization," 1996.
- [30] H. M. Wallach, I. Murray, R. Salakhutdinov, and D. Mimno, "Evaluation Methods for Topic Models."
- [31] D. Mimno, H. M. Wallach, E. Talley, M. Leenders, and A. Mccallum, "Optimizing Semantic Coherence in Topic Models," Association for Computational Linguistics.
- [32] M. Röder, A. Both, and A. Hinneburg, "Exploring the space of topic coherence measures," in *WSDM 2015 - Proceedings of the 8th ACM International Conference on Web Search and Data Mining*, Feb. 2015, pp. 399–408. doi: 10.1145/2684822.2685324.
- [33] F. Gao, B. Li, L. Chen, Z. Shang, X. Wei, and C. He, "A softmax classifier for high-precision classification of ultrasonic similar signals," *Ultrasonics*, vol. 112, Apr. 2021, doi: 10.1016/j.ultras.2020.106344.
- [34] T. Kitasuka, M. Aritsugi, and F. Rahutomo, "Semantic Cosine Similarity," 2012. [Online]. Available: <https://www.researchgate.net/publication/262525676>
- [35] Nikolay Oskolkov, "How Exactly UMAP Works And why exactly it is better than tSNE," 2019.
- [36] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, "A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise," 1996. [Online]. Available: [www.aaai.org](http://www.aaai.org)
- [37] M. Grootendorst, "Guided Topic Modeling," 2022.