

**LATENT DIRICHLET ALLOCATION (LDA) UNTUK  
MENGETAHUI TOPIK PEMBICARAAN WARGANET  
TWITTER TENTANG *OMNIBUS LAW***

Skripsi :

Sebagai Salah Satu Syarat untuk Memperoleh

Gelar Sarjana Komputer (S.Kom)



Oleh :

**M. LUVIAN CHISNI CHILMI**

11150910000040

**PROGRAM STUDI TEKNIK INFORMATIKA**

**FAKULTAS SAINS DAN TEKNOLOGI**

**UNIVERSITAS ISLAM NEGERI SYARIF HIDAYATULLAH**

**JAKARTA**

**2021 M / 1442 H**

**LATENT DIRICHLET ALLOCATION (LDA) UNTUK  
MENGETAHUI TOPIK PEMBICARAAN WARGANET  
TWITTER TENTANG *OMNIBUS LAW***

Skripsi

Sebagai Salah Satu Syarat untuk Memperoleh Gelar

Sarjana Komputer (S.Kom)

Fakultas Sains dan Teknologi

Univeristas Islam Negeri Syarif Hidayatullah Jakarta

Oleh :

M. LUVIAN CHISNI CHILMI

11150910000040

**PROGRAM STUDI TEKNIK INFORMATIKA**

**FAKULTAS SAINS DAN TEKNOLOGI**

**UNIVERSITAS ISLAM NEGERI SYARIF HIDAYATULLAH**

**JAKARTA**

**2021 M / 1442 H**

**LEMBAR PERSETUJUAN**  
**LATENT DIRICHLET ALLOCATION (LDA) UNTUK**  
**MENGETAHUI TOPIK PEMBICARAAN WARGANET**  
**TWITTER TENTANG OMNIBUS LAW**



Ketua Program Studi Teknik Informatika

  
**Imam Marzuki Shofi, M.T**  
NIP. 19720205 200801 1 010

## LEMBAR PENGESAHAN

Skripsi Berjudul “*LATENT DIRICHLET ALLOCATION (LDA) UNTUK MENGETAHUI TOPIK PEMBICARAAN WARGANET TWITTER TENTANG OMNIBUS LAW*” telah diujikan dan dinyatakan lulus dalam sidang munaqasah Fakultas Sains dan Teknologi UIN Syarif Hidayatullah Jakarta pada 25 Juni 2021. Skripsi ini telah diterima sebagai salah satu syarat memperoleh gelar Sarjana Komputer (S.Kom) pada Program Studi Teknik Informatika.

Jakarta, 25 Juni 2021

### Tim Penguji,

Pengaji I

Husni Teja Sukmana, S.T., M.Sc., Ph.D

NIP. 19771030 200112 1 003

Pengaji II

Fitri Mintarsih, M.Kom

NIP. 19721223 200710 2 004

### Tim Pembimbing,

Pembimbing I

Fenty Eka Muzayyana Agustin, M.Kom

NIP. 19760805 200912 2 003

Pembimbing II

Rizal Broer Bahaweres, M.Kom

NIP. 19780424 200801 2 022

### Mengetahui,

Dekan

Fakultas Sains dan Teknologi



Nasrudin Hakiem, S.Si., M.T., Ph.D

NIP. 19710608 200501 1 005

Ketua Program

Studi Teknik Informatika

Imam Marzuki Shofi, M.T

NIP. 19720205 200801 1 010

## **PERNYATAAN ORISINALITAS**

Dengan ini saya menyatakan bahwa:

1. Skripsi ini merupakan hasil karya asli saya yang diajukan untuk memenuhi salah satu persyaratan memperoleh gelar Strata 1 (S1) di UIN Syarif Hidayatullah Jakarta.
2. Semua sumber yang saya gunakan dalam penulisan ini telah saya cantumkan sesuai dengan ketentuan yang berlaku di UIN Syarif Hidayatullah Jakarta.
3. Apabila di kemudian hari terbukti karya ini bukan hasil karya asli saya atau merupakan hasil jiplakan karya orang lain, maka saya bersedia menerima sanksi yang berlaku di UIN Syarif Hidayatullah Jakarta.

Jakarta, 25 Juni 2021



M. Luvian Chisni Chilmi

11150910000040

## **PERNYATAAN PERSETUJUAN PUBLIKASI SKRIPSI**

Sebagai civitas akademik UIN Syarif Hidayatullah Jakarta, saya yang bertanda tangan di bawah ini:

Nama : M. Luvian Chisni Chilmi  
NIM : 11150910000040  
Program Studi : Teknik Informatika  
Fakultas : Sains dan Teknologi  
Jenis Karya : Skripsi

demi pembuatan ilmu pengetahuan, menyetujui untuk memberikan kepada UIN Syarif Hidayatullah Jakarta Hak Bebas Royalti Noneksklusif (*Non-exclusive Royalty Free Right*) atas karya ilmiah yang berjudul:

### **LATENT DIRICHLET ALLOCATION (LDA) UNTUK MENGETAHUI TOPIK PEMBICARAAN WARGANET TWITTER TENTANG OMNIBUS LAW**

beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Noneksklusif ini UIN Syarif Hidayatullah Jakarta berhak menyimpan, mengalihmedia/ formatkan, mengelola dalam bentuk pangkalan data (*database*), merawat, dan mempublikasikan tugas akhir saya selama tetap mencantumkan nama saya sebagai penulis / pencipta dan sebagai pemilik Hak Cipta. Demikian pernyataan ini saya buat dengan sebenarnya.

Jakarta, 25 Juni 2021

M. Luvian Chisni Chilmi



(.....)

## KATA PENGANTAR

Puji syukur penulis panjatkan kehadirat Allah SWT yang telah melimpahkan rahmat, taufik serta hidayah-Nya sehingga penulis dapat menyelesaikan skripsi ini. Proses penyelesaian skripsi ini tidak lepas dari bantuan, dukungan, saran, serta kritik dari berbagai pihak, oleh karena itu penulis ingin menyampaikan terima kasih kepada :

1. Allah SWT yang telah memberikan nikmat, rahmat, dan karunia-Nya kepada penulis.
2. Imam Marzuki Shofi, M.T selaku Ketua Program Studi Teknik Informatika dan Bapak Andrew Fiade, M.Kom selaku Sekretaris Program Studi Teknik Informatika.
3. Bapak dan Ibu Dosen Program Studi Teknik Informatika Fakultas Sains dan Teknologi UIN Syarif Hidayatullah Jakarta yang telah memberikan ilmu kepada penulis didalam maupun diluar kelas.
4. Kedua orang tua penulis, Ibu Eva dan Bapak Nasrullah yang tiada hentinya mendoakan, memberi dukungan, membiayai dalam segala hal serta menyayangi penulis.
5. Elda Suci Yala Merru, yang senantiasa menjadi pendamping penulis, mengingatkan untuk segera menyelesaikan skripsi, bertukar pikiran, memberikan semangat, dukungan, dan doa tiada hentinya.
6. Adik penulis, Zidan Aji Musyafa dan Ali Masykur Falahuddin yang senantiasa mengingatkan, memberikan doa, dukungan, dan motivasi kepada penulis.
7. Adhyaksa yang telah membantu penulis mengajarkan & menyelesaikan teknis aplikasi pada skripsi penulis.
8. Arif Rohman, Zainal Muttaqin dan Berlian Muhammad yang telah memberikan dukungan moril kepada penulis agar cepat menyelesaikan skripsi ini.

9. Teman-teman grup #Calonsarjana yang selalu menyemangati dan mendoakan agar bisa segera wisuda tahun 2020 ini. Semoga doa ini juga berbalik kepada kalian, aamiin yaa rabbal alamin!
10. Teman-teman ISDP yang telah memberikan dukungan pada penulis serta motivasi untuk segera menyelesaikan skripsi ini.
11. Teman-teman mahasiswa Program Studi Teknik Informatika angkatan 2015, yang telah memberikan masukan serta motivasi kepada penulis.
12. Semua pihak yang tidak dapat penulis sebut satu persatu yang telah membantu dalam penyelesaian penulisan skripsi ini.

Penulis menyadari bahwa skripsi ini masih memiliki banyak kekurangan. Oleh sebab itu, penulis mengharapkan saran dan kritik yang bersifat membangun untuk dapat memperbaiki dan menyempurnakan skripsi ini sehingga dapat bermanfaat bagi semua pembaca yang mana dapat disampaikan melalui *email* penulis: luvian.chisni15@mhs.uinjkt.ac.id

Jakarta, 25 Juni 2021



M. Luvian Chisni Chilmi

11150910000040

Nama : M. Luvian Chisni Chilmi

Program Studi: Teknik Informatika

Judul : *Latent Dirichlet Allocation (LDA) Untuk Mengetahui Topik Pembicaraan Warganet Tentang Omnibus Law*

## ABSTRAK

*Omnibus law* menimbulkan pro dan kontra di seluruh lapisan masyarakat sejak pernyataan Presiden Jokowi mengenai rencana pengeluaran *omnibus law* hingga disahkan menjadi undang - undang pada tanggal 5 Oktober 2020. Kini *omnibus law* menjadi topik serta isu yang sedang ramai diperbincangkan, baik di kalangan masyarakat secara langsung maupun melalui media *social* seperti Twitter. Tujuan penelitian ini adalah mengetahui gambaran umum dari perbincangan publik tentang *omnibus law*. Metode *Latent Dirichlet Allocation (LDA)* digunakan untuk pemodelan topik dalam penelitian ini. Dari penelitian ini diperoleh jumlah topik sebanyak 5 dengan *coherence score* sebesar 0.5644. Kesimpulan dari setiap topiknya yaitu topik ke-1 mengenai penolakan UU Omnibus Law Cipta Kerja oleh sejumlah lapisan masarakat dan kepala daerah. Topik ke-2 mengenai kekhawatiran masyarakat terhadap penyebaran virus Corona yang disebabkan oleh pelanggaran kebijakan PSBB yang dilakukan oleh demonstran dalam unjuk rasa omnibus law. Topik ke-3 mengenai dampak UU omnibus law cipta kerja terhadap kawasan hutan dan lingkungan. Topik ke-4 mengenai keistimewaan yang didapat warga asing karena omnibus law. Topik ke-5 kerusuhan yang terjadi pada saat demo tolak omnibus law di berbagai wilayah.

**Kata Kunci :** *Omnibus Law, Topic Modelling, Latent Dirichlet Allocation (LDA)*

Author : M. Luvian Chisni Chilmi  
Major : Informatics Engineering  
Title : Latent Dirichlet Allocation (LDA) To Know The Topic of Citizen's Conversation About *Omnibus Law*

## ABSTRACT

Omnibus law raises pros and cons in all walks of life since President Jokowi's statement on the omnibus law spending plan until it was passed into law on October 5, 2020. Now omnibus law is a topic and issue that is being discussed, both in the community directly and through social media such as Twitter. The purpose of this study is to know the general picture of the public discussion about omnibus law. The Latent Dirichlet Allocation (LDA) method was used for topic modeling in this study. From this study obtained the number of topics as many as 5 with a coherence score of 0.5644. The conclusion of each topic is the 1st topic concerning the rejection of the Omnibus Law Cipta Kerja by a several layers of society and regional heads. Topic 2 concerning public concerns about the spread of corona virus caused by violations of PSBB policies committed by protesters in omnibus law rallies. Topic 3 on the impact of omnibus law copyright work on forest and environmental areas. The 4th topic concerns the privileges that foreigners get because of the omnibus law. The 5th topic of unrest that occurred during the demonstration rejected omnibus law in various regions.

**Keywords :** *Omnibus Law, Topic Modelling, Latent Dirichlet Allocation (LDA)*

## DAFTAR ISI

<b>LEMBAR PERSETUJUAN .....</b>	<b>ii</b>
<b>LEMBAR PENGESAHAN .....</b>	<b>iii</b>
<b>PERNYATAAN ORISINALITAS.....</b>	<b>iv</b>
<b>PERNYATAAN PERSETUJUAN PUBLIKASI SKRIPSI.....</b>	<b>v</b>
<b>KATA PENGANTAR.....</b>	<b>vi</b>
<b>ABSTRAK .....</b>	<b>viii</b>
<b>ABSTRACT .....</b>	<b>ix</b>
<b>DAFTAR ISI.....</b>	<b>x</b>
<b>DAFTAR GAMBAR.....</b>	<b>xv</b>
<b>DAFTAR TABEL .....</b>	<b>xvii</b>
<b>BAB 1 PENDAHULUAN .....</b>	<b>1</b>
1.1    Latar Belakang .....	1
1.2    Rumusan Masalah .....	11
1.3    Batasan Masalah.....	11
1.4    Tujuan Penelitian.....	11
1.5    Manfaat Penelitian.....	11
1.5.1    Manfaat Bagi Penulis .....	12
1.5.2    Manfaat Bagi Universitas.....	12
1.5.3    Manfaat Bagi Masyarakat .....	12
1.6    Metode Penelitian.....	12
1.6.1    Metode Pengumpulan Data .....	12
1.6.2    Metode <i>Lifecycle Of Text Mining</i> .....	13
1.7    Sistematika Penulisan.....	13
<b>BAB 2 TINJAUAN PUSTAKA DAN LANDASAN TEORI.....</b>	<b>15</b>

2.1	<i>Omnibus Law</i> .....	15
2.2	<i>Twitter</i> .....	17
2.2.1	<i>Glosarium Twitter</i> .....	17
2.2.2	<i>Streaming API</i> .....	18
2.3	<i>Text Mining</i> .....	18
2.4	<i>Preprocessing</i> .....	19
2.4.1	<i>Case Folding</i> .....	20
2.4.2	<i>Remove Punctuation</i> .....	20
2.4.3	<i>Stopwords</i> .....	20
2.4.4	<i>Tokenizing</i> .....	20
2.5	<i>Python</i> .....	21
2.5.1	<i>Packages</i> .....	21
2.5.2	<i>Gensim</i> .....	22
2.5.3	<i>pyLDAvis</i> .....	23
2.6	<i>Bigram dan Trigram</i> .....	23
2.7	<i>Term Frequency-Invers Document Frequency (TF-IDF)</i> .....	24
2.8	<i>Word Embedding</i> .....	26
2.9	<i>Word2Vec</i> .....	28
2.9.1	<i>Skip-gram</i> .....	30
2.9.2	<i>Continous Bag of Word (CBOW)</i> .....	32
2.10	<i>Topic Modeling</i> .....	34
2.11	<i>Latent Dirichlet Allocation (LDA)</i> .....	35
2.12	<i>Topic Coherence</i> .....	37
2.13	<i>Use Case Diagram</i> .....	37
2.14	<i>Activity Diagram</i> .....	38

2.15	Metode Pengumpulan Data .....	40
2.16	Studi Pustaka .....	41
2.17	Pengamatan .....	41
2.18	Metode <i>Lifecycle of Text Mining</i> .....	41
2.18.1	<i>Problem Definition</i> .....	42
2.18.2	<i>Selecting Text Mining Approach</i> .....	42
2.18.3	<i>Data Collection</i> .....	42
2.18.4	<i>Text Standarization</i> .....	42
2.18.5	<i>Text Preprocessing</i> .....	43
2.18.6	<i>Feature Extraction</i> .....	43
2.18.7	<i>Analysis</i> .....	43
2.18.8	<i>Discovery</i> .....	43
2.19	Literatur Sejenis .....	44
<b>BAB 3 METODOLOGI PENELITIAN .....</b>		<b>50</b>
3.1	Metode Pengumpulan Data .....	50
3.1.1	Pengamatan .....	50
3.1.2	Studi Pustaka.....	50
3.2	Metode <i>Lifecycle of Text Mining</i> .....	51
3.2.1	<i>Problem Definition</i> .....	51
3.2.2	<i>Selectinng Text Data Mining Approach</i> .....	51
3.2.3	<i>Data Collection</i> .....	51
3.2.4	<i>Text Standardization</i> .....	52
3.2.5	<i>Text Processing</i> .....	52
3.2.6	<i>Feature Extraction</i> .....	53
3.2.7	<i>Analysis</i> .....	53

3.2.8	<i>Discovery</i> .....	53
3.3	<i>Kerangka Berpikir</i> .....	54
<b>BAB 4 PERANCANGAN DAN PENGUJIAN .....</b>		<b>55</b>
4.1	<i>Problem Definition</i> .....	55
4.2	<i>Selectinng Text Data Mining Approach</i> .....	55
4.3	<i>Data Collection</i> .....	55
4.4	<i>Text Standardization</i> .....	57
4.5	<i>Text Processing</i> .....	57
4.5.1	<i>Tahap Pre-Processing</i> .....	58
4.5.2	<i>Caselolding</i> .....	58
4.5.3	<i>Remove Punctuation</i> .....	60
4.5.4	<i>Stopword</i> .....	61
4.5.5	<i>Tokenizing</i> .....	63
4.6	<i>Feature Extraction</i> .....	65
4.6.1	<i>Bigram &amp; Trigram Model</i> .....	65
4.6.2	Pembobotan <i>Term Frequency-Invers Document Frequency</i> (TF-IDF)	
	70	
4.7	<i>Analysis</i> .....	74
4.7.1	Alur Pemodelan Topik dengan <i>Latent Dirichlet Allocation</i> (LDA) 74	
4.7.2	Eksperimen Pemodelan Topik .....	77
4.7.3	Hasil <i>Topic Modelling</i> dengan <i>Latent Dirichlet Allocation</i> (LDA) 83	
<b>BAB 5 HASIL DAN PEMBAHASAN .....</b>		<b>89</b>
5.1	<i>Discovery</i> .....	89
5.1.1	Model LDA Topik ke-1 .....	89
5.1.2	Model LDA Topik ke-2 .....	92

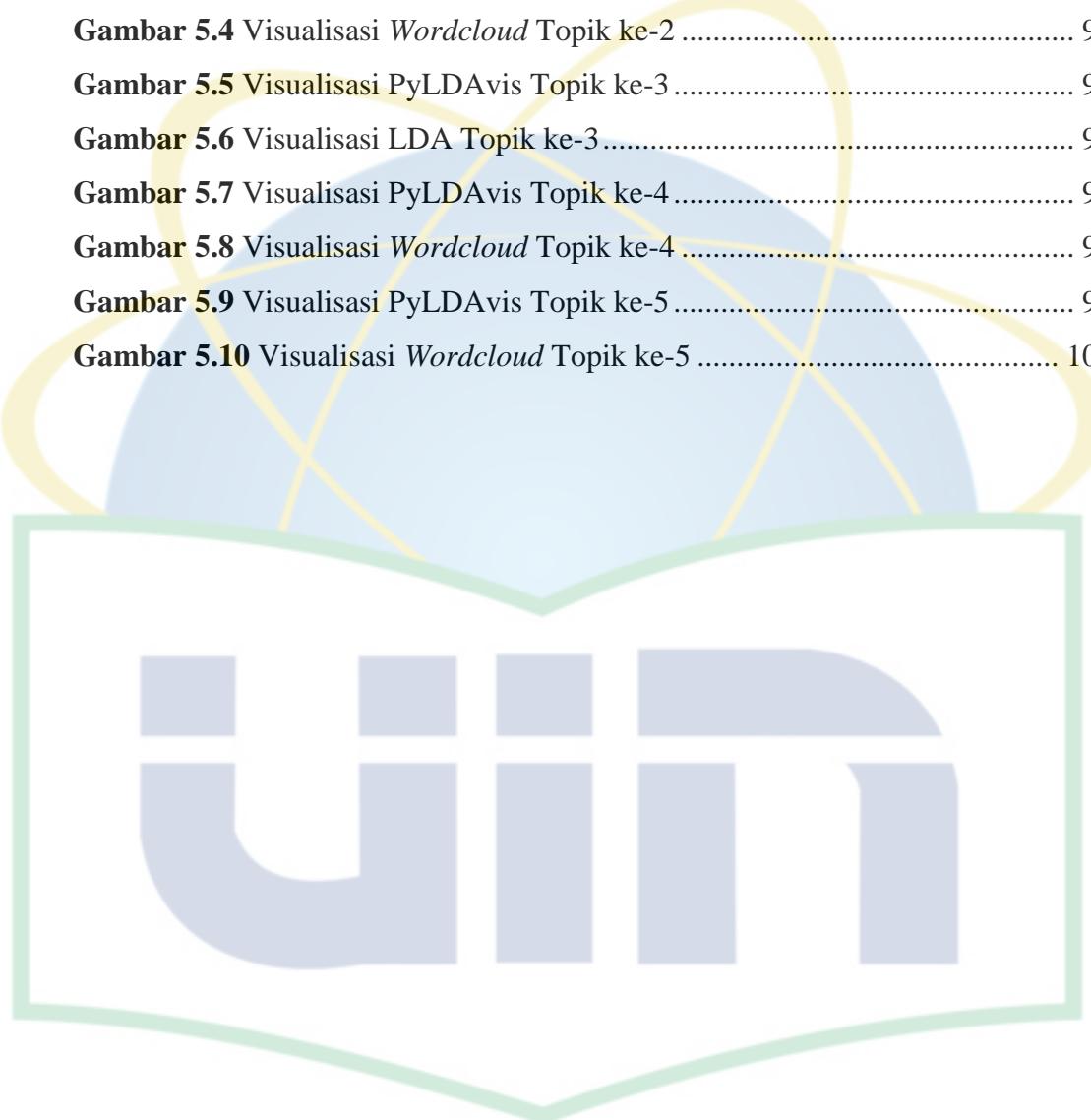
5.1.3	Model LDA Topik ke-3 .....	94
5.1.4	Model LDA Topik ke-4 .....	97
5.1.5	Model LDA Topik ke-5 .....	99
5.2	Hasil Penelitian dan Pembahasan .....	101
<b>BAB 6 PENUTUP.....</b>		<b>103</b>
6.1	Kesimpulan.....	103
6.2	Saran .....	104
<b>DAFTAR PUSTAKA .....</b>		<b>105</b>



## DAFTAR GAMBAR

<b>Gambar 2.1</b> One-hot encoding.....	27
<b>Gambar 2.2</b> Word Embedding .....	28
<b>Gambar 2.3</b> Contoh Word2Vec.....	29
<b>Gambar 2.4</b> Arsitektur Word2Vec Skip-gram .....	30
<b>Gambar 2.5</b> Ilustrasi Windowing Skip-gram.....	30
<b>Gambar 2.6</b> Ilustrasi Neural Network Skip-gram .....	30
<b>Gambar 2.7</b> Ilustrasi Feedforward Neural Network Skip-gram .....	31
<b>Gambar 2.8</b> Arsitektur Word2vec CBOW .....	32
<b>Gambar 2.9</b> Ilustrasi Windowing CBOW.....	32
<b>Gambar 2.10</b> Neural Network CBOW .....	33
<b>Gambar 2.11</b> Ilustrasi Feedforward Word2vec CBOW.....	33
<b>Gambar 2.12</b> Model Representasi LDA (Blei, et al., 2003) .....	35
<b>Gambar 2.13</b> Metode <i>Lifecycle of Text Mining</i> .....	41
<b>Gambar 3.1</b> Kerangka Berpikir .....	54
<b>Gambar 4.1</b> Flowchart Tahap Pengambilan Data .....	56
<b>Gambar 4.2</b> Flowchart Tahap Pre-Processing .....	58
<b>Gambar 4.3</b> Flowchart Stopword.....	62
<b>Gambar 4.4</b> Flowchart Tokenizing.....	64
<b>Gambar 4.5</b> Flowchart Bigram & Trigram.....	66
<b>Gambar 4.6</b> Flowchart Pembobotan TF-IDF.....	71
<b>Gambar 4.7</b> Flowchart Topik Modelling LDA .....	75
<b>Gambar 4.8</b> Pembentukan Dictionary dan Corpus.....	76
<b>Gambar 4.9</b> Pemodelan Topik Latent Dirichlet Allocation (LDA) .....	76
<b>Gambar 4.10</b> Grafik coherence score .....	83
<b>Gambar 4.11</b> Visualisasi PyLDAvis.....	86
<b>Gambar 4.12</b> Visualisasi Wordcloud Topik 1 .....	86
<b>Gambar 4.13</b> Visualisasi Wordcloud Topik 2 .....	87
<b>Gambar 4.14</b> Visualisasi Wordcloud Topik 3 .....	87
<b>Gambar 4.15</b> Visualisasi Wordcloud Topik 4 .....	87

<b>Gambar 4.16</b> Visualisasi <i>Wordcloud</i> Topik 5 .....	88
<b>Gambar 5.1</b> Visualisasi PyLDAvis Topik ke-1 .....	90
<b>Gambar 5.2</b> Visualisasi Wordcloud Topik ke-1 .....	90
<b>Gambar 5.3</b> Visualisasi PyLDAvis Topik ke-2 .....	92
<b>Gambar 5.4</b> Visualisasi <i>Wordcloud</i> Topik ke-2 .....	93
<b>Gambar 5.5</b> Visualisasi PyLDAvis Topik ke-3 .....	95
<b>Gambar 5.6</b> Visualisasi LDA Topik ke-3 .....	95
<b>Gambar 5.7</b> Visualisasi PyLDAvis Topik ke-4 .....	97
<b>Gambar 5.8</b> Visualisasi <i>Wordcloud</i> Topik ke-4 .....	98
<b>Gambar 5.9</b> Visualisasi PyLDAvis Topik ke-5 .....	99
<b>Gambar 5.10</b> Visualisasi <i>Wordcloud</i> Topik ke-5 .....	100



## DAFTAR TABEL

<b>Tabel 2.1</b> Input one-hot encoded .....	31
<b>Tabel 2.2</b> Use Case Diagram .....	38
<b>Tabel 2.3</b> Activity Diagram .....	39
<b>Tabel 2.4</b> Literatur Sejenis .....	47
<b>Tabel 4.1</b> Data Penulisan.....	56
<b>Tabel 4.2</b> Case Folding .....	59
<b>Tabel 4.3</b> Remove Punctuation.....	60
<b>Tabel 4.4</b> Stopword.....	62
<b>Tabel 4.5</b> Tokenizing .....	64
<b>Tabel 4.6</b> Bigram & Trigram .....	67
<b>Tabel 4.7</b> Output TF .....	71
<b>Tabel 4.8</b> Perhitungan TF-IDF .....	72
<b>Tabel 4.9</b> Contoh Hasil TF-IDF .....	73
<b>Tabel 4.10</b> Eksperimen Penentuan Jumlah Topik .....	78
<b>Tabel 4.11</b> Eksperimen Perbandingan LDA dengan LSA.....	80
<b>Tabel 4.12</b> Coherence Score.....	84
<b>Tabel 4.13</b> Hasil Pemodelan Topik LDA .....	84
<b>Tabel 4.14</b> Nilai Principal Component (PC).....	85
<b>Tabel 5.1</b> Model LDA Topik ke-1.....	89
<b>Tabel 5.2</b> Model LDA Topik ke-2.....	92
<b>Tabel 5.3</b> Model LDA Topik ke-3.....	94
<b>Tabel 5.4</b> Model LDA Topik ke-4.....	97
<b>Tabel 5.5</b> Model LDA Topik ke-5.....	99

# BAB 1

## PENDAHULUAN

### 1.1 Latar Belakang

Meningkatnya penggunaan teknologi informasi dan komunikasi khususnya internet dalam beberapa tahun terakhir ini telah memunculkan satu tren di kalangan masyarakat untuk menyampaikan aspirasinya melalui media online internet. Penyampaian aspirasi masyarakat tersebut dapat kita temukan dalam berbagai situs, blog, maupun jejaring sosial yang tersedia dalam media internet seperti Twitter (Yuniyati, et al., 2017). Twitter merupakan salah satu media sosial populer di dunia sejak pertama kali dipublikasikan pada tahun 2006 (Singh, 2015). Indonesia menggunakan Twitter sejak awal dipublikasikan dan termasuk dalam pengguna Twitter yang paling aktif. Terdapat sekitar 29 Juta pengguna Twitter di Indonesia. Pada tahun 2014, Indonesia menempati peringkat kelima sebagai negara dengan tweet terbanyak (Carley, et al., 2015).

Berbagai informasi dapat diperoleh dari media soial Twitter yang kini semakin banyak dimanfaatkan. Pengguna memanfaatkannya untuk berbagai kebutuhan misalnya untuk kebutuhan publik, pemerintahan, dan bisnis. *Tweet* yang dibagikan oleh pengguna Twitter mencakup beragam topik tertentu. Pengguna dapat saling berpendapat mengenai *tweet* yang dibagikan tersebut. Topik mewakili isi dari banyak *tweet* yang membahas mengenai satu konteks yang sama. Dari tweet tersebut dapat ditemukan topik utama yang sedang banyak diperbincangkan oleh pengguna saat itu dengan melakukan analisis topik (Utami, 2017). Analisis topik dilakukan untuk memberikan gambaran umum yang mewakili isi dari topik utama. Begitu banyak *tweet* yang dapat dianalisis untuk memperoleh topik utama. Diperlukan suatu metode yang cepat dan efisien yaitu dengan metode pemodelan topik. Pemodelan topik dapat digambarkan sebagai metode untuk menemukan kelompok kata (topik) dari kumpulan dokumen yang dapat merepresentasikan dengan baik informasi yang ada dalam kumpulan dokumen tersebut (Nair, 2016).

*Topic modelling* atau pemodelan topik merupakan metode *clustering* yang termasuk dalam *unsupervised learning*. Dalam *unsupervised learning* tidak ada label untuk suatu objek. Terdapat 3 tipe *clustering* yaitu *hard clustering*, *hierarchical clustering*, dan *soft/fuzzy clustering*. Pemodelan topik termasuk dalam *soft/fuzzy clustering* yang mana setiap objek dapat dimiliki lebih dari satu *cluster* dengan tingkat tertentu (Doig, 2015). Konsep *topic modeling* menurut Blei (Blei, 2012) terdiri dari entitas-entitas yaitu “kata”, “dokumen”, dan “corpora”. “Kata” dianggap sebagai unit dasar dari data diskrit dalam dokumen, didefinisikan sebagai item dari kosa kata yang diberi indeks untuk setiap kata unik pada dokumen. “Dokumen” adalah susunan N kata-kata. Sebuah corpus adalah kumpulan M dokumen dan corpora merupakan bentuk jamak dari corpus. Sementara “topic” adalah distribusi dari beberapa kosakata yang bersifat tetap. *Topic Modeling* merupakan dokumen teks yang terdiri dari kata-kata, topik yang dapat dituliskan dalam banyak dokumen dapat dinyatakan dengan kombinasi kata-kata yang saling terkait dan teknik yang dapat digunakan untuk menyimpulkan suatu topik yang tersembunyi dalam sebuah dokumen teks. Karena *topic modelling* ini mewakili dari setiap dokumen sebagai kombinasi kompleks dari beberapa topik dan setiap topiknya sebagai kombinasi kompleks dari beberapa kata, kemudian juga digunakan sebagai alat *text mining* untuk mengklasifikasikan sebuah dokumen berdasarkan hasil kesimpulan topik (Nugroho & Alamsyah, 2018).

Terdapat beberapa teknik yang dapat digunakan untuk pemodelan topik, diantaranya yaitu *Vector Space Model* (VSM), *Latent Semantic Analysis* (LSA), *Probabilistic Latent Semantic Analysis* (PLSA) dan *Latent Dirichlet Allocation* (LDA). Proses pemodelan topik pada VSM dilakukan dengan merepresentasikan setiap dokumen dalam bentuk vektor, kemudian tingkat kemiripan (*similarity*) antar dokumen dilakukan dengan menghitung penyimpangan sudut antar vektor. VSM bekerja dengan melihat kecocokan *term* (*term similarity*) antara *query* dan *corpus* dimana setiap *term* memiliki dependensi yang tinggi, oleh karena itu VSM bekerja sangat baik dalam kasus pencocokan kata kunci (Chawla & Singh, 2013). Namun, dikarenakan setiap *term* memiliki dependensi yang tinggi maka VSM mempunyai permasalahan yakni tidak mampu menangani masalah sinonim, selain itu VSM juga

merepresentasikan dokumen-dokumen ke dalam dimensi ruang (*space*) yang besar dan jarang (*sparse*) sehingga membuat akurasi penghitungan *similarity* menjadi rendah (Barde & Bainwad, 2017).

*Latent Semantic Analysis* (LSA) atau dikenal juga dengan nama *Latent Semantic Indexing* (LSI) adalah pengembangan dari metode VSM. LSA juga merupakan salah satu metode untuk melakukan pemodelan topik dengan merepresentasikan dokumen ke dalam ruang topik (*topic space*). LSA mengambil kata-kata penting dari informasi yang diberikan oleh dokumen dan menangkap kesamaan semantik (*semantic similarity*) antara kata-kata sehingga mampu mengatasi masalah sinonim (Xhu, et al., 2007), selain itu model LSA mereduksi dimensi ruang vektor sehingga mempunyai akurasi yang lebih baik daripada VSM (Barde & Bainwad, 2017). Namun, permasalahan yang timbul dalam penggunaan metode LSA ini adalah adanya faktor *polysemy* dalam pengelompokan kata (Hofmann, 2001). Permasalahan *polysemy* pada kata dapat diatasi dengan menggunakan varian dari LSA yang dikenal sebagai *Probabilistic Latent Semantic Analysis* (PLSA).

*Probabilistic Latent Semantic Analysis* (PLSA) adalah sebuah algoritma yang diterapkan untuk memperkirakan makna sekumpulan teks menjadi suatu *cluster* atau kelompok (kategori) tertentu sehingga mempermudah para analis untuk menarik suatu kesimpulan dari pengelompokan yang terbentuk (Hutomo & Zulhanif, 2013). Secara umum metode PLSA menggabungkan teori klasik tentang *vector space model*, *Singular Value Decomposition* (SVD) serta model variabel *latent*, yang diformulasikan dalam suatu bentuk model peluang dengan tujuan untuk mendapatkan suatu kelompok (*latent*) dari sekumpulan teks (*bag of words*) (Zulhanif, 2016). Metode LSA dan PLSA mengabaikan urutan kata (*word ordering*) dalam proses analisisnya, hal ini menjadi masalah dikarenakan beberapa *term* tertentu akan memiliki makna yang jauh, perkembangan selanjutnya pada analisis teks dalam pemodelan topik adalah seperti yang dikemukakan oleh Blei (2003) yaitu *Latent Dirichlet Allocation* (LDA) yang mana menggunakan model

latent variabel dengan pendekatan bayesian dalam penaksianan parameter modelnya.

*Latent Dirichlet Allocation* (LDA) merupakan pengembangan dari PLSA yang lebih stabil untuk mengolah data dalam jumlah besar (Blei, et al., 2003). Metode LDA mengasumsikan bahwa pada satu dokumen terdapat lebih dari satu topik, yang masing-masing merupakan distribusi melalui kosakata (Liu & Li, 2013). LDA adalah metode yang digunakan untuk melakukan analisis pada dokumen yang sangat besar. LDA juga digunakan untuk meringkas, melakukan pengelompokan, menghubungkan maupun memproses data. Model LDA merupakan model probabilistik generatif dari suatu korpus. Ide dasar model LDA adalah suatu dokumen yang direpresentasikan sebagai model campuran dari berbagai topik yang dapat disebut juga laten, setiap topik dikarakterisasikan oleh kata (Blei, et al., 2003). Terdapat variabel laten (tersembunyi) yang dapat menjelaskan variabel observasi (teramati) pada model probabilitas tersebut. Variabel observasi adalah dokumen, sedangkan variabel laten adalah topik yang ditentukan dari tiap kata pada dokumen. Distribusi yang kompleks menyebabkan perhitungan estimasi dari distribusi posterior untuk model LDA menjadi sangat sulit dilakukan secara manual (Kengken, 2014). Secara umum, cara kerja LDA adalah dengan memasukkan kumpulan dokumen dan beberapa parameter yang ditentukan. Kemudian dilakukan proses LDA sehingga menghasilkan model yang terdiri dari bobot yang dapat dinormalisasi terhadap probabilitas. Probabilitas muncul dalam 2 jenis : (a) probabilitas bahwa dokumen tertentu menghasilkan topik tertentu pada suatu posisi dan (b) probabilitas bahwa topik tertentu menghasilkan kata tertentu dari kumpulan perbendaharaan kata (Campbell, 2014).

Metode *Latent Dirichlet Allocation* (LDA) pada penelitian ini digunakan untuk menentukan pemodelan topik terhadap pendapat masyarakat mengenai *omnibus law*. Topik mengenai *omnibus law* dipilih dalam penelitian ini dikarenakan sejak disahkannya oleh DPR RI pada tanggal 5 Oktober 2020, *omnibus law* menjadi topik yang ramai diperbincangkan masyarakat (Tempo.co, 2020).

*Omnibus law* adalah undang-undang yang substansinya merevisi dan/atau mencabut banyak undang-undang. secara etimologinya atau asal usul katanya, *Omnibus* berasal dari bahasa latin *omnis* yang artinya banyak sedangkan *law* memiliki arti hukum. (Busroh, 2017). Konsep ini berkembang di negara - negara *common law* dengan sistem hukum *anglo saxon* seperti Amerika Serikat, Belgia, Inggris dan Kanada. Konsep *omnibus law* menawarkan pemberahan permasalahan yang disebabkan karena peraturan yang terlalu banyak (*over regulation*) dan tumpang tindih (*over lapping*). Pembahasan dalam *omnibus law* menekankan pada penyederhanaan regulasi sebagai upaya mereformasi regulasi ke arah yang lebih baik (Busroh, 2017).

Salah satu negara yang mengadopsi konsep *omnibus law* adalah Serbia pada 2002 untuk mengatur status otonomi Provinsi Vojvodina. Undang - Undang yang dibentuk dengan konsep ini mencakup yurisdiksi pemerintah Provinsi Vojvodina mengenai budaya, pendidikan, bahasa, media, kesehatan, sanitasi, jaminan kesehatan, pensiun, perlindungan sosial, pariwisata, pertambangan, pertanian, dan olahraga. Selain Serbia, sebagaimana yang dipublikasi di *Privacy Exchange.org*. Konsep *omnibus law* juga sudah diadopsi oleh negara-negara seperti Argentina, Australia, Austria, Belgium, Canada, Chile, Czech Republic, Denmark, Estonia, Finland, France, Germany, Greece, Hungary, Iceland, Ireland, Israel, Italy, Japan, Latvia, Liechtenstein, Lithuania, Luxembourg, Malta ,The Netherlands, New Zealand, Norway, Poland, Portugal, Romania, Russia, Slovak Republic, Slovenia, Spain, Sweden, Switzerland, Taiwan, Thailand, dan United Kingdom (Antoni, 2020).

Konsep *omnibus law* sejatinya dapat menjadi solusi untuk menyederhanakan regulasi yang terlalu banyak, Regulasi merupakan sebuah peraturan yang diterbitkan dan digunakan untuk mengatur suatu kelompok tertentu (Pusat studi Hukum dan Kebijakan Indonesia, 2019). Negara Indonesia sendiri memiliki regulasi yang terbilang banyak, sepanjang tahun 2000-2015, pemerintah pusat telah mengeluarkan regulasi sebanyak 12.417 peraturan, kemudian regulasi pemerintahan sebanyak 2.446 peraturan (Bappenas, 2019). Sementara itu, produk

yang dikeluarkan oleh perda kabupaten / kota sebanyak 25.575 yang kemudian disusul regulasi perda provinsi sebanyak 3.177, sedangkan pada tahun 2017 angka regulasi di Indonesia mencapai 42.000 (Ramadhan, 2019).

Selain jumlahnya yang terlalu banyak, regulasi tersebut juga tumpang tindih, sehingga untuk memperbaiki satu persoalan tidak cukup hanya dengan merevisi satu undang-undang saja. Misalnya bila terdapat masalah pengaturan soal kehutanan yang mengharuskan regulasinya diperbaiki, maka yang harus direvisi adalah UU No. 41/1999 tentang Kehutanan. Namun, masih ada ganjalan dalam *beleid* lain, semisal UU No. 32/2009 tentang Perlindungan dan Pengelolaan Lingkungan Hidup (PPLH) atau UU No. 5/1960 tentang Peraturan Dasar Pokok-Pokok Agraria (Hukumonline.com, 2017). Selain regulasi yang terlalu banyak, terdapat beberapa permasalahan mendasar lainnya, pertama, tidak sinkronnya perencanaan peraturan perundangundangan, baik di tingkat pusat maupun daerah dengan perencanaan dan kebijakan pembangunan. Kedua, adanya kecenderungan peraturan perundangundangan menyimpang dari materi muatan yang seharusnya diatur. Ketiga, ketidaktaatan terhadap materi muatan tersebut memunculkan persoalan “*hiper-regulasi*”. Keempat, efektivitas peraturan perundang-undangan juga sering menjadi persoalan yang muncul pada saat implementasi. Keadaan diperburuk dengan tidak adanya prosedur pemantauan dan evaluasi peraturan perundangundangan serta ketiadaan lembaga khusus yang menangani seluruh aspek dalam sistem peraturan perundang-undangan (Antoni, 2020).

*Omnibus law* menyita perhatian masyarakat di Indonesia setelah Presiden RI menyampaikannya dalam pidato kenegaraan pada pelantikannya sebagai Presiden di hadapan sidang MPR pada 20 Oktober 2019. *Omnibus law* menjadi fokus presiden dengan tujuan agar dapat menyelesaikan permasalahan tumpang tindihnya regulasi dan birokrasi. Harapannya dengan adanya *omnibus law* tersebut dapat memberikan pelayanan yang baik bagi masyarakat dan menarik investor asing berinvestasi di Indonesia. *omnibus law* menjadi salah satu jalan keluar untuk menarik investor asing ke Indonesia. Peraturan yang banyak, sektoral dan terdapat yang saling tumpang tindih menjadi latar belakang dibuatnya *omnibus law*.

*Omnibus law* juga menjadi solusi bagi lambatnya proses legislasi di Indonesia. Hal tersebut nampak dalam kinerja dari DPR RI periode 2014-2019 yang hanya membuat 84 undang - undang dimana lebih sedikit dari pada DPR RI periode 2009-2014 yang menghasilkan 125 undang - undang dalam kurun waktu 5 (lima) tahun (Prabowo, et al., 2020).

*Omnibus law* menimbulkan pro dan kontra di seluruh lapisan masyarakat. Sejak pernyataan Jokowi mengenai rencana pengeluaran *omnibus law*, kini *omnibus law* menjadi topik serta isu yang sedang ramai diperbincangkan. Baik di kalangan masyarakat secara langsung maupun melalui media *social* seperti YouTube, Instagram, dan Twitter (Susanto, et al., 2020). Di dalam proses perancangan RUU *omnibus law* terdapat banyak sekali opini-opini masyarakat yang tidak setuju dengan banyak pasal dalam RUU ini. Adanya opini-opini publik ini tidak lain disebabkan karena pengerjaannya yang di *deadline* hanya selama 100 hari oleh Presiden Jokowi dan juga tidak melibatkan banyak pihak dalam pembuatannya (Kurniawan, 2020). Mengutip dari harian Tempo.co sejak disahkannya *omnibus law* oleh DPR RI pada 5 Oktober 2020, gelombang penolakan terhadap *omnibus law* oleh masyarakat terus terjadi, bahkan masyarakat ramai berdemonstrasi di jalanan dan di depan gedung DPR/MPR walaupun sedang dalam masa Pembatasan Sosial Berskala Besar (PSBB) karena pandemi *Covid-19*. Bukan hanya ramai dengan demonstran yang turun ke jalan untuk menyampaikan aspirasinya, masyarakat juga ramai menyampaikan aspirasi dalam media *social*. Dalam pemberitaan harian detik.com tagar yang berkaitan dengan tolak *omnibus law* menjadi *tranding topic* di Twitter.

Pada penelitian sebelumnya yang dilakukan oleh I Made Kusnanta Bramantya Putra (2017) mengenai Analisis Topik Informasi Publik Media Sosial di Surabaya Menggunakan *Pemodelan Latent Dirichlet Allocation* (LDA). Data yang digunakan oleh penulis yaitu laporan peristiwa atau kejadian yang diperoleh dari partisipasi masyarakat. Jumlah laporan perhari yang tinggi dan berbagai macam topik dari laporan tersebut menyebabkan kesusahan dalam mengidentifikasi suatu topiknya sehingga akan menghabiskan banyak waktu jika dilakukan secara

manual oleh manusia. Dengan demikian sangat dibutuhkan pemodelan topik yang dapat secara otomatis mengklasifikasi pesan media sosial tersebut ke dalam topik-topik yang muncul dari hasil pemodelannya. Pemodelan topik dilakukan dengan metode *Latent Dirichlet Allocation* (LDA). Dengan metode LDA didapatkan hasil jumlah topik yang terdapat dalam pesan media sosial yaitu 4 topik dengan nilai *perplexity* terbaik yaitu sebesar 213.41 dan diuji kemudahannya untuk diinterpretasi oleh manusia dengan uji koherensi topik yang terdiri dari *word intrusion task* dan *topic intrusion task*. Kesimpulannya dengan menggunakan metode LDA, model yang didapatkan dapat diinterpretasi manusia dengan baik.

Selanjutnya, pada penelitian yang dilakukan oleh Agustina (2017) mengenai Analisis dan visualisasi suara pelanggan pada pusat layanan pelanggan dengan pemodelan topik menggunakan *Latent Dirichlet Allocation* (LDA) studi kasus: PT. Petromkimia Gresik. Pada perusahaan tersebut salah satunya yaitu memproduksi pupuk dimana pupuk merupakan kebutuhan penting dalam bidang pertanian. Kepuasan pelanggan menjadi salah satu aspek yang diperhatikan oleh perusahaan salah satunya yaitu dengan adanya aplikasi Pusat Layanan Pelanggan yang berguna untuk mengoptimalkan suara pelanggannya. Pada pencatatan suara pelanggan yang telah masuk dengan berbagai media tetapi belum terdapat analisis terhadap topik apakah yang sering pelanggan suarakan maka dilakukan analisis topik dengan menggunakan pemodelan LDA. Didapatkan hasil yaitu 35 topik yang berhasil diidentifikasi kemudian dikelompokkan ke dalam 7 kategori. Nilai *perplexity* yang diperoleh sebesar 34.92 dengan standar deviasi 0.49 pada 20 iterasi dan nilai akurasinya sebesar 83.7%. Kemudian hasil identifikasi divisualisasikan dalam dashboard berbasis web.

Kemudian, penelitian lainnya yang dilakukan oleh Guntur Budi Herwanto (2018) mengenai dokumen *clustering* dengan LDA dan *ward hierarichal clustering*. Pada saat ini terdapat sangat banyak konten informasi dalam bentuk berita dari berbagai sumber setiap harinya. Dengan banyaknya konten tersebut menuntut organisasi konten yang baik untuk pencarian informasi yang diinginkan dapat dilakukan dengan mudah maka dilakukan *clustering* dokumen. Penelitian ini

menggunakan kombinasi pemodelan topik LDA dengan ward *hierarchical clustering*. Pada LDA digunakan untuk representasi vektor dokumen berupa distribusi topik dengan tujuan mengurai dimensi vektor yang biasanya terlalu panjang jika menggunakan tf-idf. Didapatkan hasil *silhouette coefficient* yang baik yaitu 0.7. Performa *silhouette coefficient* pada representasi pemodelan topik lebih baik dibandingkan dengan representasi dengan tf-idf.

Terakhir, penelitian yang dilakukan oleh Wirasakti, Permadi, Hartanto, & Hartatik (2020) mengenai Pembuatan Kata Kunci Otomatis Dalam Artikel Dengan Pemodelan Topik. Pada penulisan ini yaitu untuk mengetahui kata kunci yang cocok untuk digunakan dalam sebuah publikasi artikel pada sebuah blog dengan menggunakan model LDA yaitu model probabilitas yang dapat menghasilkan beberapa topik. Data diambil dari blog/artikel kemudian dilakukan pemotongan per bagian dan dilakukan preprocessing data, serta pengubahan vektor data menjadi corpus untuk dimodelkan dengan LDA, yang kemudian dilakukan clustering dengan menggunakan K-Means. Didapatkan hasil model LDA yaitu 4 topik dengan 8 kata. Yang memiliki nilai probabilitas tertinggi diantaranya mesin (0.09375857), maksimal (0.064600445), mazda (0.10009629), varian (0.07572112), cx-8 (0.10170187), mazda (0.101048954), mobil (0.09820121), dan mazda (0.05679208).

Berdasarkan penelitian sebelumnya, penulis membandingkan dengan penelitian sekarang sehingga menjadikan penelitian penulis berbeda dan memiliki keunikan tersendiri dari penelitian lainnya. Penulis ingin mengelompokkan tweet masyarakat mengenai tweet tentang *omnibus law* dengan LDA. Perbedaan penelitian ini dengan sebelumnya adalah iterasi yang digunakan dan penentuan kelompok yang optimum menggunakan *topik coherence*. Menurut Wisdom (2017) *topic coherence* dapat dianggap memberikan kemampuan interpretasi lebih baik terhadap hasil dari *topic modeling* dibandingkan dengan *perplexity*. Tujuan dilakukannya penelitian ini adalah untuk mengelompokkan pendapat warganet di Twitter mengenai *omnibus law* sehingga dapat ditemukan topik yang menjadi perhatian warganet mengenai *omnibus law*.

Berdasarkan pada latar belakang yang telah penulis uraikan sebelumnya, maka penulis akan melakukan penelitian terhadap pengguna (*user*) twitter terkait topik dengan *key word omnibus law* menggunakan metode LDA untuk menentukan topik modeling dengan menggunakan *topic coherence* untuk validasi topiknya. Berdasarkan permasalahan tersebut sebagai bahan kajian yang tertuang dalam bentuk skripsi dengan judul “**LATENT DIRICHLET ALLOCATION (LDA) UNTUK MENGETAHUI TOPIK PEMBICARAAN WARGANET TWITTER TENTANG OMNIBUS LAW**”.

Adapun integrasi keislaman yang berhubungan dengan penelitian penulis terkait dengan studi kasus *omnibus law* adalah sebagai berikut:

إِنَّ اللَّهَ يَأْمُرُكُمْ أَنْ تُؤْدُوا الْأَمَانَاتِ إِلَى أَهْلِهَا وَإِذَا حَكَمْتُمْ بَيْنَ النَّاسِ إِنْ تَحْكُمُوا بِالْعَدْلِ إِنَّ اللَّهَ يُعْلَمُ بِمَا يَعْمَلُونَ إِنَّ اللَّهَ كَانَ سَمِيعًا بَصِيرًا

“Sesungguhnya Allah menyuruhmu menyampaikan amanah kepada yang berhak menerimanya. Dan apabila kamu menetapkan hukum di antara manusia, hendaknya kamu menetapkannya dengan adil Sesungguhnya Allah sebaik-baik yang memberi pengajaran kepadamu. Sungguh, Allah Maha Mendengar lagi Maha Melihat”. (QS. An-Nisa : 58)”.

Menurut Ibnu Katsir, Surat An-Nisa Ayat 58 mengandung perintah untuk menegakkan keadilan di dalam ketetapan hukum di antara manusia. Seperti halnya diriwayatkan oleh Muhammad bin Ka’ab, Zaidbin Aslam, dan Sahru bin Jaushib, bahwa ayat ini berkaitan dengan *umara’* (pemegang pemerintahan) untuk memperlakukan hukum dengan adil. Sehingga ada dua pelajaran yang diperintahkan Allah dalam ayat ini, yaitu untuk menyampaikan amanah kepada yang berhak menerimanya dan berbuat adil kepada sesama manusia (Srifariyati, 2019).

## 1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah dijelaskan, maka dikembangkan suatu rumusan masalah pada penelitian ini yaitu sebagai berikut:

- a. Bagaimana hasil pengelompokan topik pembicaraan masyarakat tentang *omnibus law* di media sosial Twitter menggunakan metode LDA?
- b. Bagaimana hasil kesimpulan tentang topik yang ditemukan?

## 1.3 Batasan Masalah

Batasan-batasan masalah dalam penelitian ini adalah sebagai berikut:

- a. Sentimen publik hanya diambil dari data *twitter* dengan *keyword omnibus law* pada tanggal 05 Oktober 2020 sampai 16 Oktober 2020.
- b. Bahasa asing, bahasa daerah, dan penggunaan emoji pada *tweet* diabaikan.
- c. Data diolah dengan menggunakan *software Anaconda 3.5.3.1* dengan metode analisis yang digunakan yaitu *Topic Modelling* dengan *Latent Dirichlet Allocation (LDA)*.

## 1.4 Tujuan Penelitian

Berikut ini merupakan tujuan dari penelitian yang akan dilakukan:

1. Mengetahui gambaran umum *topic modelling* dari *tweet* tentang *omnibus law*.
2. Mengetahui implementasi pemodelan topik dengan metode *Latent Dirichlet Allocation (LDA)* pada data *tweet* tentang *omnibus law*.

## 1.5 Manfaat Penelitian

Berdasarkan dengan permasalahan dan tujuan penelitian, maka penulis mengharapkan penelitian ini dapat memberikan beberapa manfaat dibawah ini:

### 1.5.1 Manfaat Bagi Penulis

- a. Memberikan pengetahuan lebih kepada penulis dalam menganalisa *topic modelling* terhadap tweet dengan *keyword omnibus law* menggunakan metode *Latent Dirichlet Allocation* (LDA).
- b. Menerapkan ilmu-ilmu yang diperoleh selama perkuliahan di Universitas Islam Negeri Syarif Hidayatullah Jakarta khususnya *Big Data*.
- c. Sebagai salah satu syarat dalam penyelesaian gelar Strata Satu (S1) program studi Teknik Informatika fakultas Sains dan Teknologi Universitas Islam Negeri Syarif Hidayatullah Jakarta.
- d. Hasil penelitian ini dapat menjadi referensi penulis apabila hendak melanjutkan penelitian di masa yang akan datang.

### 1.5.2 Manfaat Bagi Universitas

- a. Mengetahui kemampuan mahasiswa dalam menguasai dan menerapkan ilmu yang telah didapatkan di bangku perkuliahan dan juga sebagai bahan evaluasi.
- b. Menjadi sebuah tolak ukur bagi universitas dalam menentukan keberhasilan dan kemampuan penulis dalam mengimplementasikan ilmu yang sudah didapatkan selama menempuh pendidikan perkuliahan di universitas.
- c. Sebagai referensi untuk penelitian selanjutnya.

### 1.5.3 Manfaat Bagi Masyarakat

- a. Memberikan informasi kepada masyarakat dalam menganalisa *topic modelling* terhadap topik dengan *keyword omnibus law* yang diambil dari Twitter.
- b. Hasil dari penelitian dapat digunakan untuk penelitian tahap selanjutnya.

## 1.6 Metode Penelitian

Dalam menyusun penelitian ini, penulis menggunakan dua metode yaitu metode pengumpulan data dan metode simulasi.

### 1.6.1 Metode Pengumpulan Data

#### a. Pengamatan

Pada metode ini, penulis melakukan pengamatan dan pencatatan secara sistematis terhadap segala aktivitas yang terjadi pada media sosial Twitter khususnya dengan *keyword omnibus law* yang diteliti oleh penulis.

## b. Studi Pustaka

Pada metode ini, penulis mengumpulkan data dan informasi dengan mencari data-data yang diperlukan dari berbagai buku, artikel, karya ilmiah, dan website yang berhubungan dengan materi penelitian. Penulis melakukan kajian terhadap penelitian sebelumnya yang memiliki keterkaitan dengan topik yang dibahas dalam penelitian ini. Perbandingan studi sejenis ini diperlukan untuk mendapatkan informasi landasan teori mengenai masalah yang akan diteliti. Studi literatur yang digunakan penulis berupa jurnal, skripsi, maupun karya publikasi lainnya.

### 1.6.2 Metode *Lifecycle Of Text Mining*

Penulis menggunakan metode *lifecycle of text mining* sebagai metode untuk menganalisa perbedaan hasil dari masing-masing masukan. Metode *lifecycle of text mining* ini merujuk pada metode yang digunakan oleh Anoud Shaikh *et al.* (2019) yang meliputi beberapa tahapan, antara lain:

1. *Problem Definition*
2. *Selecting TDM Approach*
3. *Data Collection*
4. *Text Standardization*
5. *Text Processing*
6. *Feature Extraction*
7. *Analysis*
8. *Discovery*

## 1.7 Sistematika Penulisan

Dalam menyusun penelitian ini penulis menggunakan sistematika penulisan sebagai berikut:

## BAB 1 PENDAHULUAN

Pada bab 1, yaitu pendahuluan ini dikemukakan latar belakang masalah, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, metode penelitian, dan sistematika penulisan.

**BAB 2****LANDASAN TEORI**

Pada bab 2, yaitu landasan teori ini membahas mengenai dasar-dasar teori dari penelitian dan permasalahan yang berhubungan dengan topik serta penelitian yang terkait yang pernah dilakukan sebelumnya dan rencana penelitian yang akan dilakukan dalam penelitian selanjutnya.

**BAB 3****METODOLOGI PENELITIAN**

Pada bab 3, yaitu metodologi penelitian ini membahas mengenai metode serta langkah-langkah yang digunakan penulis dalam melakukan penelitian yang dilakukan.

**BAB 4****ANALISIS, PERANCANGAN, DAN PENGUJIAN**

Pada bab 4, yaitu analisis, perancangan, dan pengujian ini menjelaskan mengenai simulasi sistem dari perancangan sampai pengujian sistem yang sesuai dengan metode yang digunakan pada sistem.

**BAB 5****HASIL DAN PEMBAHASAN**

Pada bab 5 ini, yaitu hasil dan pembahasan berisi tentang hasil pembahasan yang diperoleh dari sistem yang telah dibuat untuk mewujudkan tujuan dan manfaat dari penelitian yang ingin dicapai.

**BAB 6****PENUTUP**

Pada bab 6 ini, yaitu penutup yang berisi kesimpulan yang diperoleh dari hasil penelitian berdasarkan rumusan masalah yang telah dikemukakan, serta saran yang dapat membantu pengembangan sistem ini di masa yang akan datang..

**DAFTAR PUSTAKA**

## BAB 2

### TINJAUAN PUSTAKA DAN LANDASAN TEORI

#### 2.1 *Omnibus Law*

*Omnibus law* adalah undang-undang yang substansinya merevisi dan/atau mencabut banyak undang-undang. secara etimologinya atau asal usul katanya, omnibus berasal dari bahasa latin *omnis* yang artinya banyak sedangkan *law* memiliki arti hukum (Busroh, 2017). Konsep ini berkembang di negara - negara *common law* dengan sistem hukum *anglo saxon* seperti Amerika Serikat, Belgia, Inggris dan Kanada. Konsep *omnibus law* menawarkan pemberian perbaikan yang disebabkan karena peraturan yang terlalu banyak (*over regulasi*) dan tumpang tindih (*over lapping*). pembahasan dalam *omnibus law* menekankan pada penyederhanaan regulasi sebagai upaya mereformasi regulasi ke arah yang lebih baik (Busroh, 2017).

Salah satu negara yang mengadopsi konsep *omnibus law* adalah Serbia pada 2002 untuk mengatur status otonomi Provinsi Vojvodina. Undang - Undang yang dibentuk dengan konsep ini mencakup yurisdiksi pemerintah Provinsi Vojvodina mengenai budaya, pendidikan, bahasa, media, kesehatan, sanitasi, jaminan kesehatan, pensiun, perlindungan sosial, pariwisata, pertambangan, pertanian, dan olahraga. Selain Serbia, sebagaimana yang dipublikasi di *Privacy Exchange.org* (*a global information resource on consumers, commerce, and data protection worldwide National Omnibus Laws*), Konsep *omnibus law* juga sudah diadopsi oleh negara-negara seperti Argentina, Australia, Austria, Belgium, Canada, Chile, Czech Republic, Denmark, Estonia, Finland, France, Germany, Greece, Hungary, Iceland, Ireland, Israel, Italy, Japan, Latvia, Liechtenstein, Lithuania, Luxembourg, Malta, The Netherlands, New Zealand, Norway, Poland, Portugal, Romania, Russia, Slovak Republic, Slovenia, Spain, Sweden, Switzerland, Taiwan, Thailand, dan United Kingdom (Antoni, 2020).

Konsep *omnibus law* sejatinya dapat menjadi solusi untuk menyederhanakan regulasi yang terlalu banyak, Regulasi merupakan sebuah peraturan yang diterbitkan dan digunakan untuk mengatur suatu kelompok tertentu (Pusat studi Hukum dan Kebijakan Indonesia, 2019). Negara Indonesia sendiri memiliki regulasi yang terbilang banyak, sepanjang tahun 2000-2015, pemerintah pusat telah mengeluarkan regulasi sebanyak 12.417 peraturan, kemudian regulasi pemerintahan sebanyak 2.446 peraturan (Bappenas, 2019). Sementara itu, produk yang dikeluarkan oleh perda kabupaten / kota sebanyak 25.575 yang kemudian disusul regulasi perda provinsi sebanyak 3.177, sedangkan pada tahun 2017 angka regulasi di Indonesia mencapai 42.000 (Ramadhan, 2019).

Selain jumlahnya yang terlalu banyak, regulasi tersebut juga tumpang tindih, sehingga untuk memperbaiki satu persoalan tidak cukup hanya dengan merevisi satu undang-undang saja. Misalnya bila terdapat masalah pengaturan soal kehutanan yang mengharuskan regulasinya diperbaiki, maka yang harus direvisi adalah UU No. 41/1999 tentang Kehutanan. Namun, masih ada ganjalan dalam *beleid* lain, semisal UU No. 32/2009 tentang Perlindungan dan Pengelolaan Lingkungan Hidup (PPLH) atau UU No. 5/1960 tentang Peraturan Dasar Pokok-Pokok Agraria (Hukumonline.com, 2017). Selain regulasi yang terlalu banyak, terdapat beberapa permasalahan mendasar lainnya, pertama, tidak sinkronnya perencanaan peraturan perundangundangan, baik di tingkat pusat maupun daerah dengan perencanaan dan kebijakan pembangunan. Kedua, adanya kecenderungan peraturan perundangundangan menyimpang dari materi muatan yang seharusnya diatur. Ketiga, ketidaktaatan terhadap materi muatan tersebut memunculkan persoalan “*hiper-regulasi*”. Keempat, efektivitas peraturan perundang-undangan juga sering menjadi persoalan yang muncul pada saat implementasi. Keadaan diperburuk dengan tidak adanya prosedur pemantauan dan evaluasi peraturan perundangundangan serta ketiadaan lembaga khusus yang menangani seluruh aspek dalam sistem peraturan perundang-undangan (Antoni, 2020).

## 2.2 Twitter

Twitter merupakan sosial media masif yang berubah menjadi situs berbagi informasi dan berkomunikasi secara cepat. Kecepatan dan kemudahan twitter dalam hal publikasi, membuat twitter menjadi sebuah medium pilihan bagi pengguna untuk berkomunikasi setiap hari. Twitter mempunyai peran dan andil penting dalam pergerakan sosial-politik seperti Arab *Spring* dan *The Occupy Wall Street movement*. Twitter juga dapat digunakan untuk melakukan laporan kerusakan dan persiapan informasi terkait bencana pada saat bencana alam akan dan sedang terjadi (Kumar, 2013).

### 2.2.1 Glosarium Twitter

Glosarium Twitter berisi kosakata dan istilah yang sering digunakan untuk membahas fitur dan aspek dari Twitter. Berikut ini merupakan kosakata pada twitter berdasarkan Support Twitter (Twitter, 2018):

1. @. Simbol @ digunakan untuk memanggil nama pengguna dalam Tweet: "Halo @twitter!" Orang lain akan menggunakan @namapengguna Anda untuk menyebut Anda di Tweet dan mengirim *Direct Message* atau tautan ke profil Anda.
2. @username. Anda dikenali di Twitter melalui nama pengguna yang selalu diawali simbol @. Misalnya, Bantuan Twitter adalah @BantuanTwitter.
3. #hashtag. Hashtag adalah kata atau frasa yang diawali langsung dengan simbol #. Bila Anda melakukan klik atau menyentuh hashtag, Anda akan melihat Tweet lain yang berisi kata kunci atau topik yang sama.
4. Geolokasi. Dengan menambahkan lokasi pada Tweet (geolokasi atau geotag), pengguna yang melihat Tweet anda akan mengetahui lokasi Anda saat mengirimkan Tweet.
5. Time Stamp. Tanggal dan waktu ketika Tweet dikirim ke Twitter. Cap waktu Tweet terlihat sebagai teks abu-abu di setiap tampilan rincian Tweet
6. Following. Berlangganan ke sebuah akun Twitter disebut "mengikuti". Untuk mulai mengikuti, klik atau sentuh tombol ikuti di samping nama akun atau di halaman profil mereka untuk melihat Tweet mereka begitu mereka mengirim

sesuatu yang baru. Pengguna di Twitter dapat mengikuti atau berhenti mengikuti pengguna lain kapan pun, kecuali akun yang diblokir.

7. Follower. Mengikuti dihasilkan dari pengguna yang mengikuti akun Twitter Anda. Anda dapat mengetahui jumlah mengikuti (atau pengikut) yang Anda miliki dari profil Twitter anda.
8. Retweet. Tindakan menyebarkan Tweet akun lain ke semua pengikut Anda dengan mengeklik atau menyentuh tombol Retweet.

### 2.2.2 *Streaming API*

Streaming API merupakan fitur pada twitter yang membantu developer untuk melakukan akses secara langsung ke dalam stream global Twitter dengan latency yang rendah, sehingga memudahkan developer untuk melakukan pengambilan data. Beberapa tipe endpoint dalam *Streaming API* berdasarkan *Support Twitter* (Twitter, 2018):

1. *Public Streams*. Menyediakan streams yang berasal dari data publik yang bergabung dengan Twitter. Jenis endpoint ini berguna untuk mencari user tertentu, mencari topik, dan melakukan data *mining*.
2. *User Streams*. *Single-user streams* yang menyediakan seluruh data yang berkesesuaian dengan seluruh informasi mengenai urus pilihan.
3. *Site Streams*. Streams untuk melakukan pencarian data yang dikhususkan untuk mencari seluruh informasi pada banyak *user*. Endpoint ini mengharuskan developer untuk melakukan koneksi ke twitter dengan otentifikasi banyak *user*.

### 2.3 *Text Mining*

*Text mining* merupakan suatu teknik yang digunakan untuk menangani dari pengelompokan, klasifikasi, ekstraksi informasi dan pengambilan informasi (Hudaya, Fakhrurroja, & Alamsyah, 2019). *Text Mining* merupakan suatu analisis data yang terdapat bahasa alami dengan menggunakan teknik dan alat untuk merancang, menemukan dan mengekstrak pengetahuan pada data yang tidak terstruktur. Pada data yang tidak terstruktur sehingga dapat menjadi data dengan topik yang lebih terstruktur dan dapat lebih mudah dianalisis dengan penambangan teks berfungsi dengan mengubah kata atau kalimat (Kabiru & Sari, 2019). Pada

proses *text mining* terdapat empat tahap yaitu text preprocessing (pemrosesan awal terhadap teks), *text transformation* (transformasi teks), *feature selection* (pemilihan fitur), dan *pattern discovery* (penemuan pola) (Sanjaya & Absar, 2015).

*Text mining* dapat memberikan solusi dari permasalahan seperti pemrosesan, pengorganisasian atau pengelompokan dan menganalisa *unstructured text* dalam jumlah besar. Dalam memberikan solusi, *text mining* mengadopsi dan mengembangkan banyak teknik dari bidang lain, seperti *Data mining*, *Information Retrieval*, Statistik dan Matematik, *Machine Learning*, *Linguistic*, *Natural Language Processing* (NLP), dan *Visualization*. Tujuan dari *text mining* adalah mendukung proses *knowledge discovery* pada koleksi dokumen yang besar untuk mendapatkan informasi yang berguna dari sekumpulan dokumen. Adapun tugas khusus dari *text mining* antara lain yaitu pengkategorisasian teks (*text categorization*) dan pengelompokan teks (*text clustering*) (Feldman & Sanger, 2007).

#### 2.4 *Preprocessing*

*Preprocessing* merupakan suatu proses awal pada klasifikasi dokumen dengan tujuan untuk menyiapkan data agar menjadi terstruktur. Hasil yang diperoleh dari *preprocessing* akan berupa nilai numerik sehingga dapat dijadikan sebagai sumber data yang dapat diolah lebih lanjut (Gusti, et al., 2016).

Pada tahap *preprocessing* dimana tahap yang sangat penting dilakukan untuk menghilangkan kata-kata dalam dokumen yang tidak dibutuhkan. Tahap ini berpengaruh yang dapat memberikan hasil setelah dilakukan pengolahan data teks tersebut. Jika tahap ini dilakukan dengan baik maka hasil dari tahap *preprocessing* akan mendapatkan hasil yang baik pula. Pada proses *preprocessing* terdiri dari beberapa proses yaitu *case folding*, *Remove punctuation*, *stopword*, dan *tokenizing*.

#### 2.4.1 Case Folding

Tahap awal yang dilakukan dalam *preprocessing* yaitu *case folding* dimana pada proses ini akan mengubah karakter dari huruf besar menjadi huruf kecil. Kemudian karakter yang akan diterima hanya “a” hingga “z” (Gusti, et al., 2016). Dalam hal ini dengan tujuan untuk dapat menghindari adanya dua kata yang sama namun dianggap berbeda oleh program yang dikarenakan perbedaan huruf kapital dan huruf kecil (Bagus, 2017).

#### 2.4.2 Remove Punctuation

*Remove punctuation* merupakan proses yang dilakukan untuk membuang karakter yang tidak digunakan berupa tanda baca, angka, markup/html/tag, spesial karakter (\$, %, &, stc) (Cendana & Permana, 2019) atau yang biasa dikenal dengan istilah noise. Noise yaitu suatu bentuk data yang dimana nantinya akan mengganggu pada proses pengolahan tersebut (Bagus, 2017).

#### 2.4.3 Stopwords

*Stopword* merupakan proses penghilangan kata tidak penting pada deskripsi melalui pengecekan kata-kata hasil parsing dokumen apakah termasuk di dalam daftar kata tidak penting (stoplist) atau tidak. Jika termasuk di dalam stoplist maka kata-kata tersebut akan dihilangkan dari deskripsi sehingga kata-kata yang tersisa di dalam deskripsi dianggap sebagai kata-kata penting atau keywords (Gusti, et al., 2016).

#### 2.4.4 Tokenizing

*Tokenizing* merupakan membagi suatu teks atau pemotongan berdasarkan dari tiap kata yang menyusunnya (Gusti, et al., 2016). Dengan tujuan yaitu agar dalam proses yang selanjutnya menjadi lebih mudah yang mengenai penghitungan kata, pembobotan kata sampai dengan transformasi kedalam bentuk vektor yang berdimensi tinggi (Bagus, 2017).

## 2.5 Python

*Python* diembangkan oleh Guido van Rossum di belanda pada tahun 1990 sebagai kelanjutan dari bahasa pemograman ABC. Nama *Python* dipilih oleh Guido karena kecintaanya pada acara televisi *Monty Python's Flying Circus*. Guido terus mengembangkan *Python* bersama *Python Software Foundation*, sehingga *Python* menjadi bahasa pemrograman yang dipakai secara luas dalam industri dan pendidikan karena sederhana, ringkas, sintaks intuitif dan memiliki pustaka yang luas (Schuerer & Maufrais, 2010).

*Python* adalah bahasa pemrograman tingkat tinggi (*high-level programming language*), berjalan dengan sistem *interpreted*, dan bisa dipakai untuk berbagai jenis tujuan (*general-purpose*). Sebuah bahasa pemrograman disebut sebagai bahasa pemrograman tingkat tinggi (*high-level programming language*) karena perintah atau kode program yang dipakai sudah mirip dengan bahasa manusia (Kuhlman, 2013).

Beberapa fitur dan Kelebihan yang dimiliki *Python* menurut Jubilee Enterprise (Jubilee Enterprise, 2017), yaitu:

1. Memiliki koleksi *library* yang banyak. Telah tersedia modul siap pakai untuk berbagai keperluan
2. Memiliki struktur bahasa yang jelas, sederhana dan mudah dipelajari,
3. *Python* adalah pemrograman berorientasi objek (OOP). Data dalam *Python* adalah sebuah objek yang dibuat dari kelas (class).
4. Memiliki sistem pengolahan memori otomatis (*Garbage Collection*).
5. Bersifat modular, sehingga mudah dikembangkan dengan menciptakan modul baru.

### 2.5.1 Packages

*Python* memiliki suatu konsep untuk membungkus beberapa file menjadi suatu kesatuan yang bisa kita panggil (*import*) di file lain untuk kebutuhan *reusable*. Konsep itu dinamakan *package* dan *module*. *Package* merupakan kumpulan dari *module-module*, dimana *module* adalah file *Python* (.py) yang berisi kumpulan

*class*, fungsi, variabel, dan *code Python* lainnya (Jubilee Enterprise, 2017). Pada penelitian ini penulis menggunakan packages yang digunakan dari *Python* diantaranya yaitu *packages Gensim* dan *pyLDAvis*. Penjelasan mengenai *packages* yaitu sebagai berikut:

### 2.5.2 *Gensim*

*Gensim* merupakan *library* dari *python* untuk implementasi LDA. *Gensim* ini dirancang yang secara otomatis mengekstraksi topik semantik dari dokumen dengan seefisien mungkin menurut Rehurek (2019). *Gensim* memiliki kemampuan dapat melakukan pemrosesan pada data teks mentah. *Gensim* juga dapat menyediakan beberapa algoritma di dalamnya diantaranya yaitu *Latent Dirichlet Allocation* (LDA). Menurut Rehurek (2017) terdapat beberapa fitur yang tersedia dari *Gensim* yaitu sebagai berikut (Listari, 2019):

- a. *Memory Independence*, seluruh data corpus tidak perlu di alokasikan pada RAM atau pada memori karena dalam satu waktu pada saat proses *training* terhadap data *corpus* tersebut dilakukan.
- b. Mengimplementasikan secara efisien yang terkait dengan algoritma ruang vektor yang populer salah satunya yaitu *Term Frequency-Invers Document Frequency* (TF-IDF).
- c. Memiliki kemampuan untuk melakukan *similarity query* terhadap dokumen dalam representasi semantiknya.

Pada penelitian ini secara khusus menggunakan beberapa fitur dari *Gensim* yaitu *gensim.corpora* dan *gensim.model*. Pada modul *gensim.corpora* yang digunakan untuk membangun dictionary dari data teks sebelum melakukan proses pembuatan model LDA dengan memanggil *module Dictionary*. Kemudian modul *gensim.model* yang digunakan untuk membangun model LDA dengan memanggil dari modul *LdaModel*. Kemudian akan dipanggil pada saat melalukan proses perhitungan *coherence model* dengan meng-import modul *Coherence Model*.

### 2.5.3 *pyLDAvis*

PyLDAvis merupakan *library* yang terdapat pada *Python* untuk visualisasi model topik interaktif. Pada PyLDAvis ini dirancang untuk dapat membantu pengguna menafsirkan topik dalam model topik yang sesuai dengan kumpulan data teks. Packages ini juga mengekstraksi informasi dari model topik LDA yang dipasang untuk dapat menginformasikan visualisasi berbasis web interaktif. Visualisasi yang dimaksudkan yaitu untuk digunakan dalam notebook *IPython* namun dapat juga disimpan dalam file HTML yang berdiri sendiri untuk memudahkan berbagi (Pypi.org, 2018).

## 2.6 *Bigram* dan *Trigram*

*Bigram* dan *Trigram* merupakan bagian dari model *N-Gram*, model *N-Gram* adalah model probabilistic yang awalnya dirancang oleh ahli matematika dari Rusia pada awal abad ke-20 dan kemudian dikembangkan untuk memprediksi item berikutnya dalam urutan item. Item bisa berupa huruf / karakter, kata, atau yang lain sesuai dengan aplikasi. Salah satunya, model *N-Gram* yang berbasis kata digunakan untuk memprediksi kata berikutnya dalam urutan kata tertentu. Dalam arti bahwa sebuah *N-Gram* hanyalah sebuah wadah kumpulan kata dengan masingmasing memiliki panjang n kata. Sebagai contoh, sebuah n-gram ukuran 1 disebut sebagai *Unigram*, ukuran 2 sebagai *Bigram*, ukuran 3 sebagai *Trigram*, dan seterusnya (Hanafi, 2009).

Pada pembangkitan karakter, N-gram terdiri dari *substring* sepanjang n karakter dari sebuah *string* dalam definisi lain N-Gram adalah potongan sejumlah n karakter dari sebuah *string*. Metode *N-Gram* ini digunakan untuk mengambil potonganpotongan karakter huruf sejumlah n dari sebuah kata yang secara kontinuitas dibaca dari teks sumber hingga akhir dari dokumen. Sebagai contoh : kata "TEXT" dapat diuraikan ke dalam beberapa *N-Gram* berikut:

- *Uni-Gram* : T, E, X, T
- *Bi-Gram* : TE, EX, XT
- *Tri-Gram* : , TEX, EXT

Sedangkan penulis menggunakan *N-Gram* pada pembangkitan kata, metode *N-Gram* ini digunakan untuk mengambil potongan kata sejumlah n dari sebuah rangkaian kata (kalimat, paragraf, bacaan) yang secara kontinuitas dibaca dari teks sumber hingga akhir dari dokumen. Sebagai contoh : kalimat "Demo Tolak Omnibus Law." Dapat diuraikan ke dalam beberapa n-gram berikut:

- *Uni-Gram* : demo, tolak, omnibus, law
- *Bi-Gram* : demo tolak, tolak omnibus, omnibus law
- *Tri-Gram* : demo tolak omnibus, tolak omnibus law

Penulis menggunakan *bigram* dan *trigram* dalam penelitian ini untuk meminimalisir pemenggalan kata yang menghilangkan makna kata. Seperti kata "Omnibus Law" yang merupakan satu kesatuan kata namun terbentuk dari 2 kata. Penggunaan *bigram* dan *trigram* untuk mencari frequensi kemunculan kata yang sering muncul bergandengan dua kata yaitu *bigram*, bergandengan tiga kata yaitu *trigram*. Selain itu, keunggulan menggunakan *n-gram* yaitu tidak terlalu sensitif terhadap kesalahan penulisan yang terdapat pada suatu dokumen.

## 2.7 Term Frequency-Invers Document Frequency (TF-IDF)

Metode TF-IDF adalah suatu cara yang digunakan untuk memberikan bobot hubungan pada suatu term (kata) terhadap suatu dokumen, yang merupakan sebuah gabungan dari dua konsep perhitungan bobot yaitu gabungan dari metode *term frequency* (tf)/frekuensi kemunculan dari sebuah kata pada sebuah dokumen tertentu dan *metode inverse document frequency* (idf)/*inverse* dari frekuensi dokumen yang mengandung kata tersebut (Karmayasa, 2012).

Pembobotan dasar yang dilakukan dalam melakukan *term weighting* (pembobotan kata) yaitu menghitung frekuensi kemunculan kata dalam dokumen. Frekuensi kemunculan (TF) yaitu menunjukkan sejauh mana kata tersebut mewakili dari isi dokumen. Kemudian semakin besar dari kemunculan suatu kata (term) pada dokumen maka akan memberikan nilai kesesuaian yang semakin besar. Pada pembobotan juga akan memperhitungkan faktor dari kebalikan frekuensi dokumen yang mengandung suatu kata (IDF) (Karmayasa, 2012).

Pada *Term Frequency* terbagi menjadi beberapa jenis algoritma yaitu (Yoren, 2018):

- a. *Raw Term Frequency* (TF Murni), yaitu nilai TF yang didapatkan dengan menghitung frekuensi kemunculan suatu term pada dokumen. Jika muncul kata sebanyak 4 kali maka kata tersebut akan bernilai 4.
- b. *Binary Term Frequency* (TF Binary), yaitu penyeragaman bobot pada dokumen dimana jika muncul minimal satu kali dalam dokumen maka memiliki nilai 1 dan jika kata tidak muncul sama sekali maka memiliki nilai 0.
- c. *Logarithmic Term Frequency* (TF Logaritmik), yaitu mengurangi tingkat kepentingan kemunculan kata dalam menghitung bobot dokumen terhadap suatu kata dengan melakukan log pada TF yang dapat diperoleh dari persamaan berikut:

$$TF = 1 + \log(TF) \quad (0.1)$$

- d. *Augmented Term Frequency* (TF Normalisasi), yaitu nilai TF dengan menggunakan perbandingan antara frekuensi sebuah kata dengan jumlah keseluruhan kata pada dokumen.

$$TF = 0.5 + 0.5 \times \left( \frac{TF}{maxTF} \right) \quad (0.2)$$

Kemudian untuk mengitung nilai IDF yaitu sebagai berikut:

$$IDF_j = \ln \frac{D}{DF_j} \quad (0.3)$$

Keterangan :

D : jumlah semua dokumen yang ada dalam koleksi

$DF_j$  : jumlah dokumen yang mengandung term (j)

Rumus pada metode TF dan IDF merupakan gabungan dari formula perhitungan row TF dan Formula IDF. Pada penelitian ini jenis formula tf yang digunakan yaitu *Raw Term Frequency* (TF Murni). Sehingga jika digunakan dalam setiap kata pada setiap dokumen akan dihitung bobotnya dengan menggunakan rumus sebagai berikut:

$$W_{ij} = TF_{ij} * IDF_j \quad (0.4)$$

$$W_{ij} = TF_{ij} * \ln \frac{D}{DF_j} \quad (0.5)$$

Keterangan :

i = dokumen ke-d

j = kata ke-t dari kata kunci

$W_{ij}$  = bobot term (j) terhadap dokumen (i)

$TF_{ij}$  = jumlah kemunculan term (j) dalam dokumen (i)

Dapat dilihat pada persamaan, didapatkan hasil bahwa berapapun besarnya nilai  $TF_{ij}$  jika nilai  $D = DF_j$  maka akan didapatkan hasil 0 pada perhitungan  $IDF$ . Dengan begitu akan ditambahkan 1 pada sisi  $IDF$ , Sehingga perhitungan bobotnya dirumuskan menjadi sebagai berikut:

$$W_{ij} = TF_{ij} * \left( \ln \frac{D}{DF_j} + 1 \right) \quad (0.6)$$

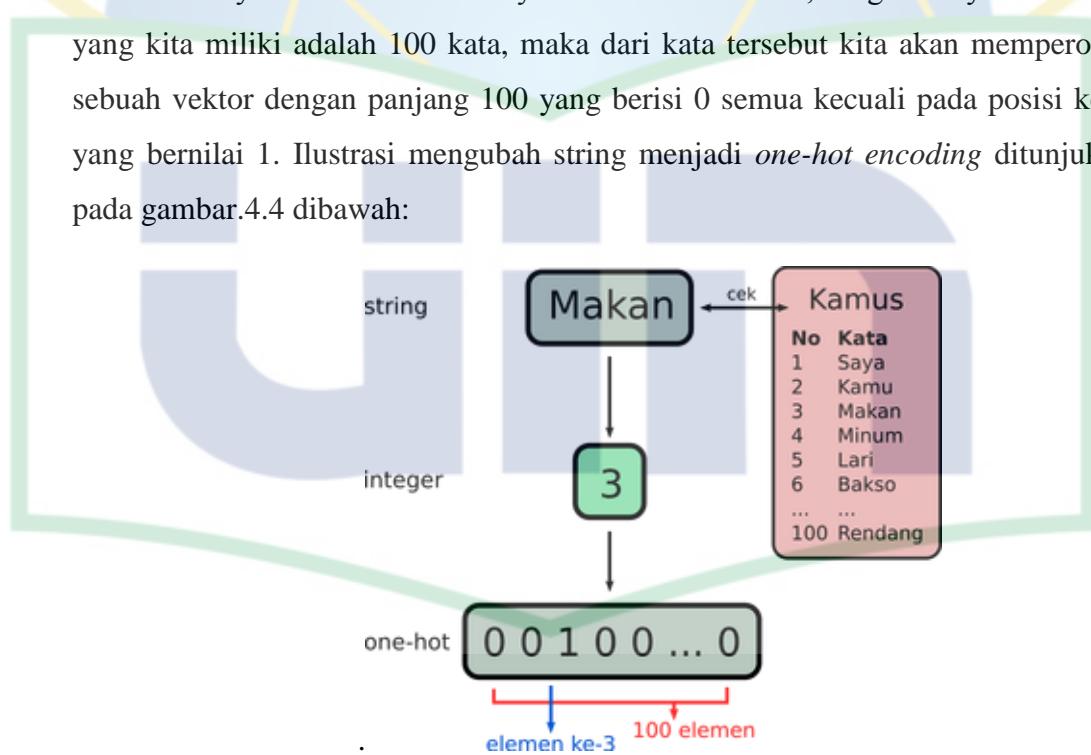
Fungsi dari metode TF-IDF ini yaitu untuk mencari representasi nilai dari setiap dokumen dalam koleksi yang dimiliki. Kemudian akan dibentuk suatu vektor antara dokumen dan term yang ditentukan berdasarkan nilai bobot term dalam dokumen. Semakin besar dari nilai perhitungan bobot yang didapatkan maka akan semakin tinggi pula tingkat similaritas dokumen terhadap term (Fattah, 2016).

## 2.8 Word Embedding

*Word Embedding* adalah proses konversi sebuah teks menjadi sebuah vektor atau *array* yang terdiri dari kumpulan angka. *Word Embedding* diperlukan sebagian besar *algoritma machine learning* dan arsitektur *deep learning* karena tidak mampu melakukan proses analisis pada input data berupa *strings* atau teks, oleh sebab itu word embedding dibutuhkan untuk mengubah teks menjadi angka sebagai input (Abdullah, 2018).

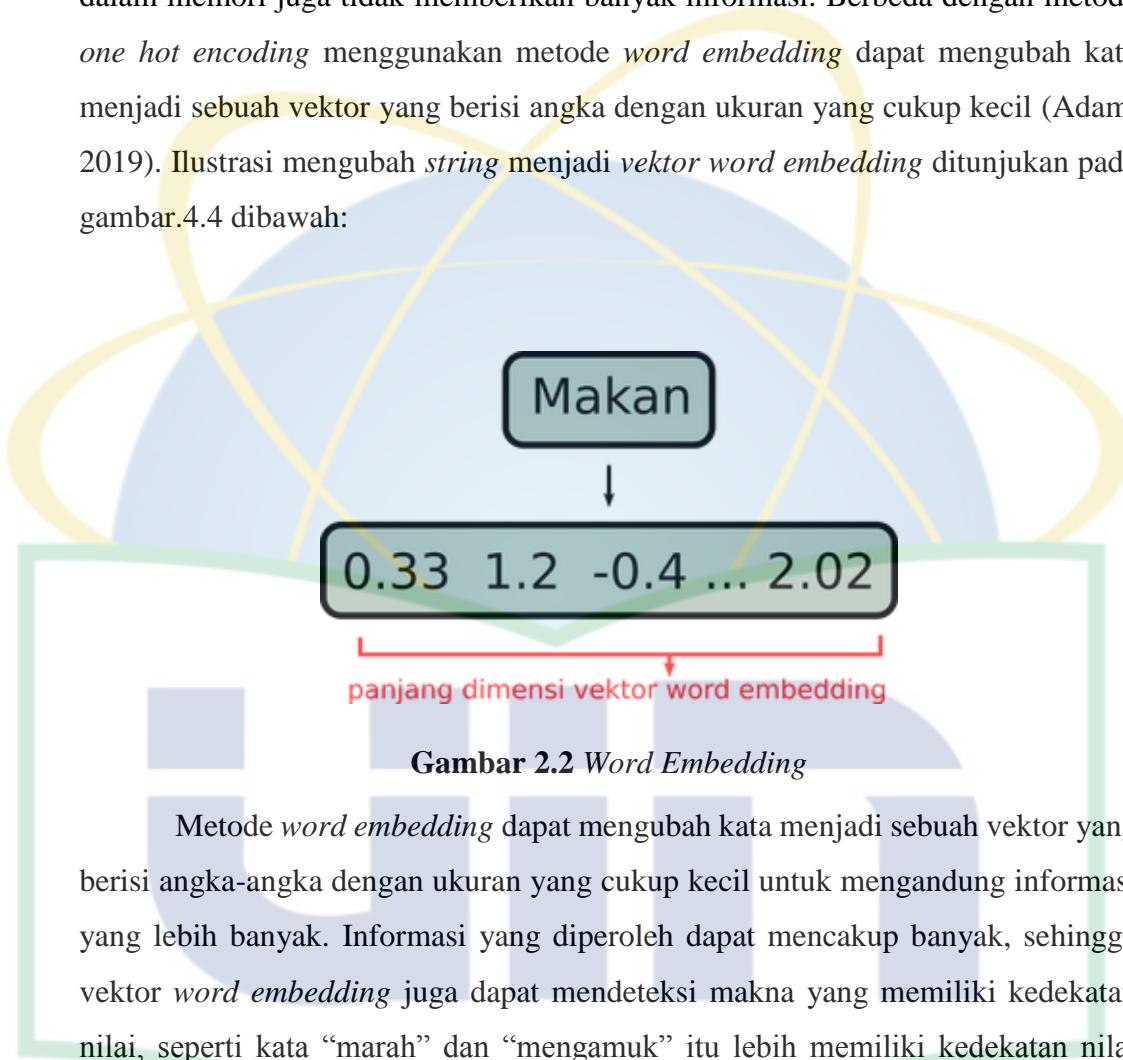
Sebagai contoh jika *model machine learning* menerima input sebuah teks, maka machine learning tidak bisa langsung menerima mentah-mentah teks tersebut. Machine learning membaca teks tersebut dengan cara membuat kamus kata berisikan semua kata yang terdapat dalam data set. Kemudian setiap menerima sebuah *string* kata, *string* tersebut diubah menjadi sebuah integer dengan memberinya nomor. Penomoran ini bisa ditentukan berdasarkan urutan di kamus kata yang kita miliki. Misalnya pada ilustrasi di bawah, string “Makan” menjadi angka 3, string “Lari” menjadi angka 5, dst.

Angka-angka tersebut selanjutnya diubah menjadi sebuah vektor (array 1 dimensi) yang memiliki panjang sepanjang banyak kata yang kita miliki di kamus. Array tersebut hanya akan bernilai 1 atau 0 menggunakan metode *one hot encoding*. Nilai 1 diposisikan pada indeks yang merupakan nomor kata tersebut sedangkan elemen lainnya bernilai 0. Contohnya untuk kata “makan”, dengan banyak kosakata yang kita miliki adalah 100 kata, maka dari kata tersebut kita akan memperoleh sebuah vektor dengan panjang 100 yang berisi 0 semua kecuali pada posisi ke 3 yang bernilai 1. Ilustrasi mengubah string menjadi *one-hot encoding* ditunjukkan pada gambar 4.4 dibawah:



Gambar 2.1 *One-hot encoding*

Jika kamus machine learning tersebut memiliki ukuran mencapai 10,000 kata, maka untuk setiap katanya akan di-convert menjadi vektor ukuran 10,000 yang hampir semua elemennya bernilai 0 semua. Metode ini selain kurang efisien dalam memori juga tidak memberikan banyak informasi. Berbeda dengan metode *one hot encoding* menggunakan metode *word embedding* dapat mengubah kata menjadi sebuah vektor yang berisi angka dengan ukuran yang cukup kecil (Adam, 2019). Ilustrasi mengubah *string* menjadi *vektor word embedding* ditunjukkan pada gambar.4.4 dibawah:



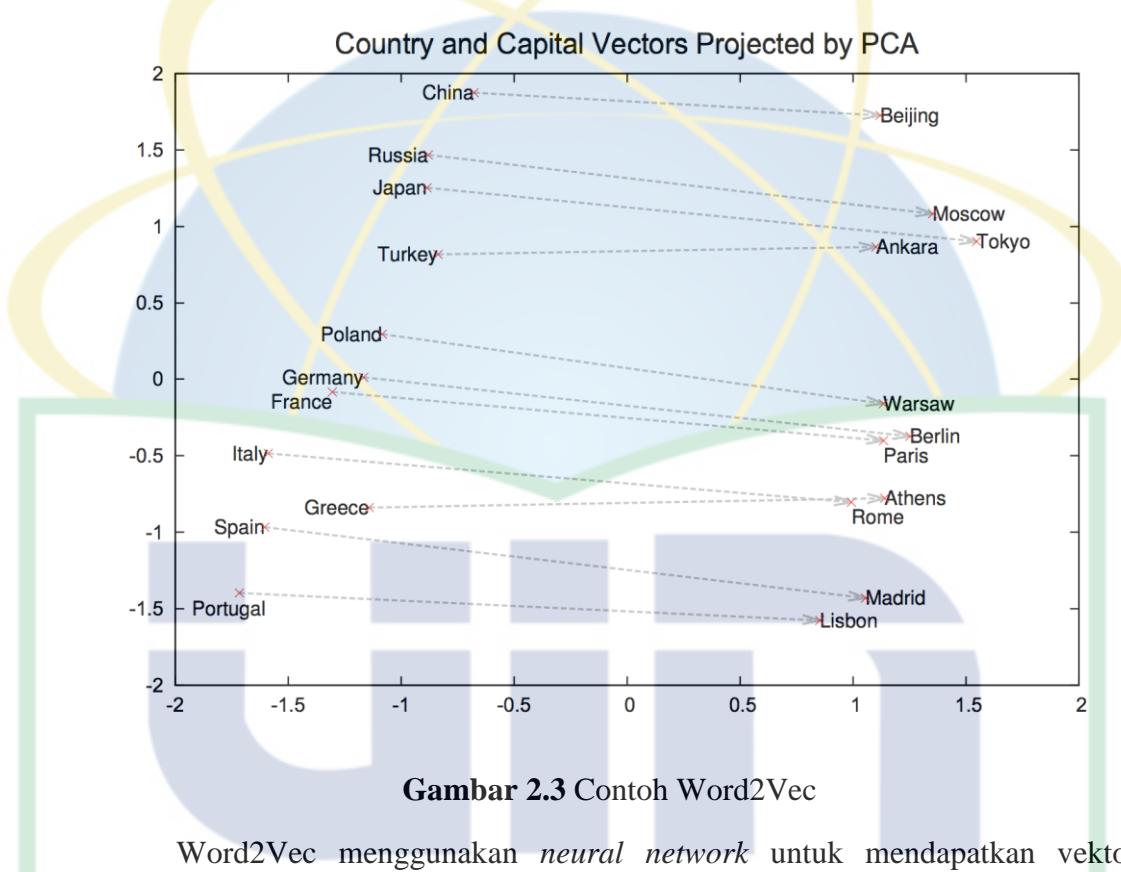
**Gambar 2.2 Word Embedding**

Metode *word embedding* dapat mengubah kata menjadi sebuah vektor yang berisi angka-angka dengan ukuran yang cukup kecil untuk mengandung informasi yang lebih banyak. Informasi yang diperoleh dapat mencakup banyak, sehingga vektor *word embedding* juga dapat mendekripsi makna yang memiliki kedekatan nilai, seperti kata “marah” dan “mengamuk” itu lebih memiliki kedekatan nilai dibandingkan kata “marah” dengan “bahagia”.

## 2.9 Word2Vec

Word2vec adalah salah satu metode *embedding word* yang berguna untuk merepresentasikan kata menjadi sebuah vektor dengan panjang N. Misalnya sebuah kata “Indonesia” di representasikan menjadi sebuah vektor dengan panjang 5 yaitu : [0.2, 0.4, -0.8, 0.9, -0.5]. Vektor tersebut tidak hanya merepresentasikan kata secara sintaktik tapi juga secara semantik atau secara makna (Mikolov, et al., 2013).

Sebagai contoh, apabila word2vec di train menggunakan korpus yang cukup lengkap, maka vektor representasi dari kata “Indonesia” akan berdekatan dengan vektor “Jakarta” sebagaimana vektor “Perancis” akan berdekatan dengan vektor “Paris”. Dengan kata lain, model word2vec akan memahami bahwa “Indonesia” dan “Jakarta memiliki hubungan yang sama dengan “Perancis” dan “Paris” yaitu negara dan ibukotanya.

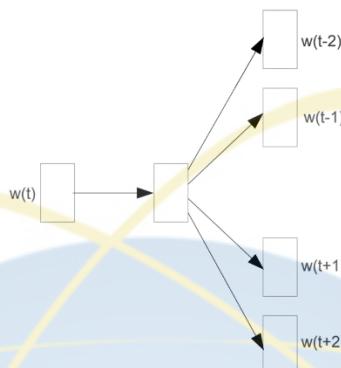


Word2Vec menggunakan *neural network* untuk mendapatkan vektor tersebut. Arsitektur Word2vec hanya terdiri dari 3 layer yaitu Input, *Projection (Hidden Layer)*, dan Output. Input pada Word2vec berbentuk *one-hot encoded vector* dengan panjang = jumlah kata unik pada data training.

Terdapat 2 jenis arsitektur neural network dari Word2Vec yaitu “*Skip-gram*” dan “*Continuous Bag of Word*” (CBOW).

### 2.9.1 Skip-gram

INPUT PROJECTION OUTPUT



**Gambar 2.4** Arsitektur Word2Vec Skip-gram

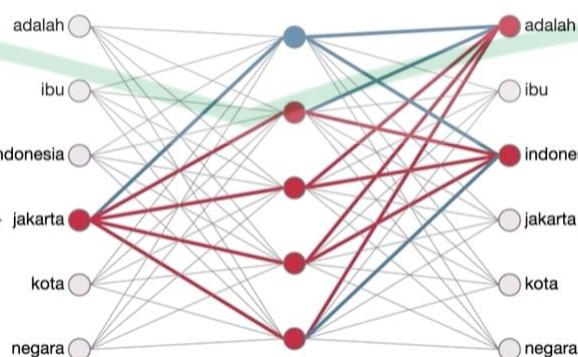
Tujuan dari arsitektur *skip-gram* adalah untuk memprediksi konteks (output) di sekitar *current word (input)*. Untuk memahaminya mari kita lihat ilustrasi berikut (Mikolov, et al., 2013).

Ibu kota Negara **Indonesia** adalah Jakarta

target      input      target

**Gambar 2.5** Ilustrasi Windowing Skip-gram

Misal data trainingnya adalah sebuah kalimat “Ibu kota Negara Indonesia adalah Jakarta” dengan *window size = 2*



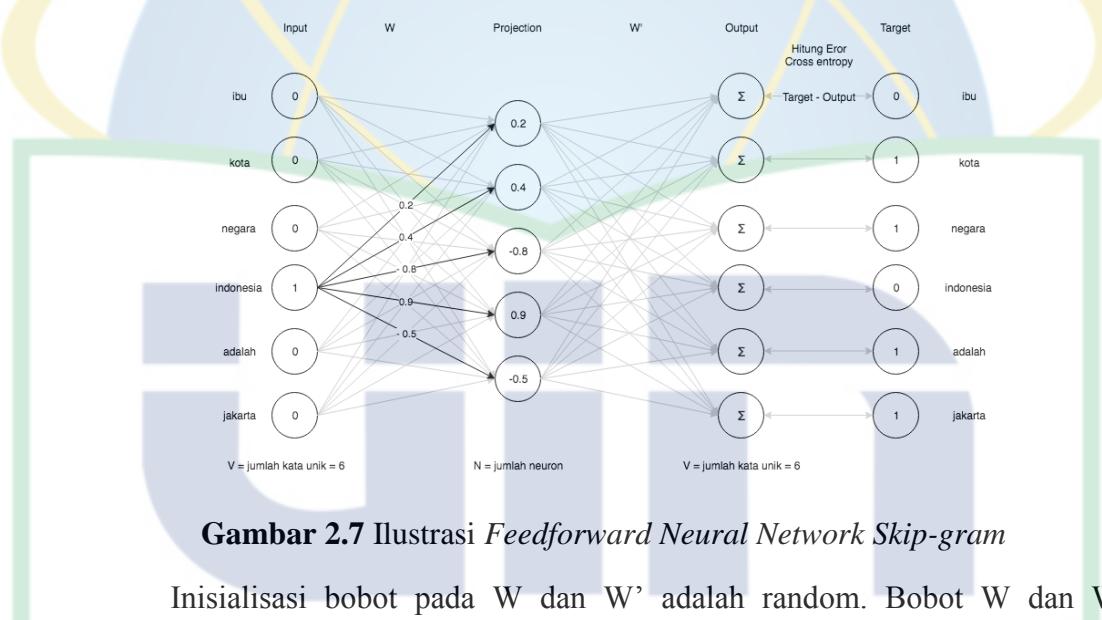
**Gambar 2.6** Ilustrasi Neural Network Skip-gram

Data input berbentuk *one-hot encoded vector* sehingga bentuk datanya adalah seperti berikut (anggap seluruh data di *e* terlebih dahulu):

**Tabel 2.1** Input one-hot encoded

ibu	[1,0,0,0,0,0]
kota	[0,1,0,0,0,0]
negara	[0,0,1,0,0,0]
indonesia	[0,0,0,1,0,0]
adalah	[0,0,0,0,1,0]
jakarta	[0,0,0,0,0,1]

Misal current word = Indonesia, maka ilustrasinya adalah sebagai berikut:

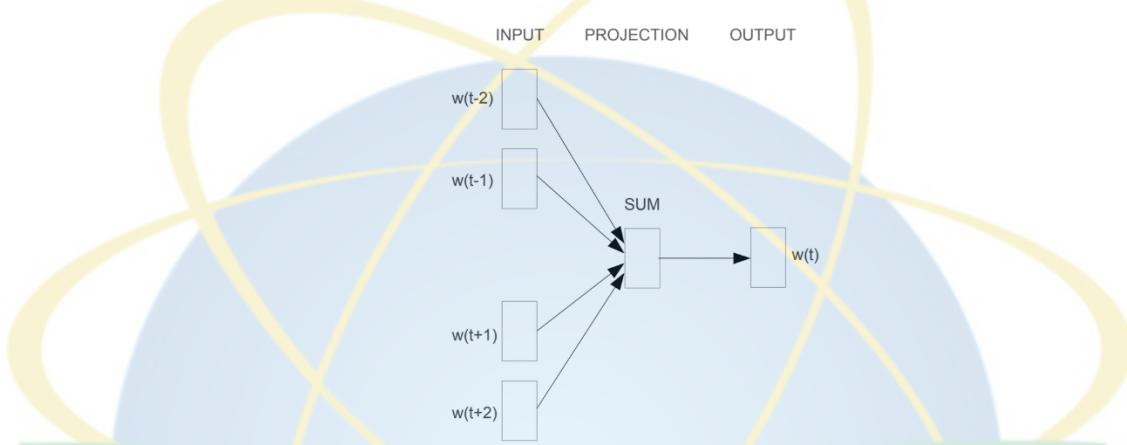


**Gambar 2.7** Ilustrasi Feedforward Neural Network Skip-gram

Inisialisasi bobot pada  $W$  dan  $W'$  adalah random. Bobot  $W$  dan  $W'$  merupakan matrik dengan ukuran  $W = V \times N$  dan  $W' = N \times V$ . Pada proses *feedforward*, vektor input akan di dot product dengan bobot  $W$  dan menghasilkan nilai pada layer projection. Kemudian layer projection di dot product dengan bobot  $W'$  dan menghasilkan vektor output. Setelah mendapatkan nilai output pada tahap *feedforward*, maka akan dihitung nilai eror nya dengan menggunakan metode *cross entropy* yaitu  $\text{Target} - \text{Output}$ . Selanjutnya adalah tahap *backpropagation* dengan memanfaatkan teknik *gradient descent* yaitu dengan melakukan *update* bobot  $W$  dan  $W'$ . Proses ini akan diulang kembali ke tahap *feedforward* hingga tercapai nilai eror minimum.

Setelah didapatkan nilai error minimum pada *cross entropy*, maka vektor yang merepresentasikan kata tersebut diambil dari bobot W dengan cara mengalikan dot product antara one-hot encoded vector masing-masing kata dengan bobot W, sedangkan bobot pada W' akan diabaikan.

### 2.9.2 Continous Bag of Word (CBOW)



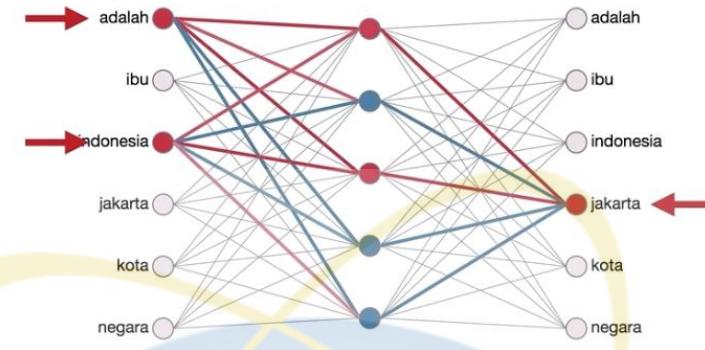
**Gambar 2.8** Arsitektur Word2vec CBOW

Arsitektur Word2Vec CBOW adalah kebalikan dari Word2vec *skip-gram*. Tujuannya adalah untuk memprediksi kata (*output*) ketika diberikan konteks disekitar kata tersebut (*input*). Berikut ini adalah ilustrasinya (Mikolov, et al., 2013).



**Gambar 2.9** Ilustrasi Windowing CBOW

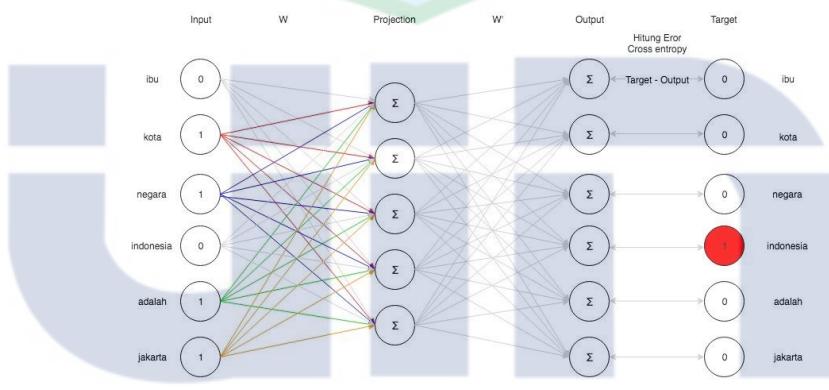
Dalam kalimat yang sama dengan yang dicontohkan sebelumnya yaitu kalimat “Ibu kota Negara Indonesia adalah Jakarta”



**Gambar 2.10 Neural Network CBOW**

Data input Word2Vec Skip-gram berbentuk *n-hot encoded vector*. Ketika proses *training* berjalan, kata yang sedang menjadi inputan akan bernilai 1 sedangkan kata yang lainnya akan bernilai 0. Karena targetnya hanya 1 kata, maka target output akan berbentuk *one-hot encoded vector*.

Berikut ini ilustrasi ketika target outputnya adalah kata “Indonesia”



**Gambar 2.11 Ilustrasi Feedforward Word2vec CBOW**

Prosesnya tidak jauh berbeda dengan arsitektur *Skip-gram*. Perbedaannya hanya terletak pada data inputnya yang berupa n-hot encoded vector dan target outputnya berupa one-hot encoded vector.

Setelah dilakukan training hingga mencapai eror minimum, maka kita dapat mengambil vektor representasi kata dengan cara mengalikan one-hot encoded vector masing kata dengan bobot W.

## 2.10 Topic Modeling

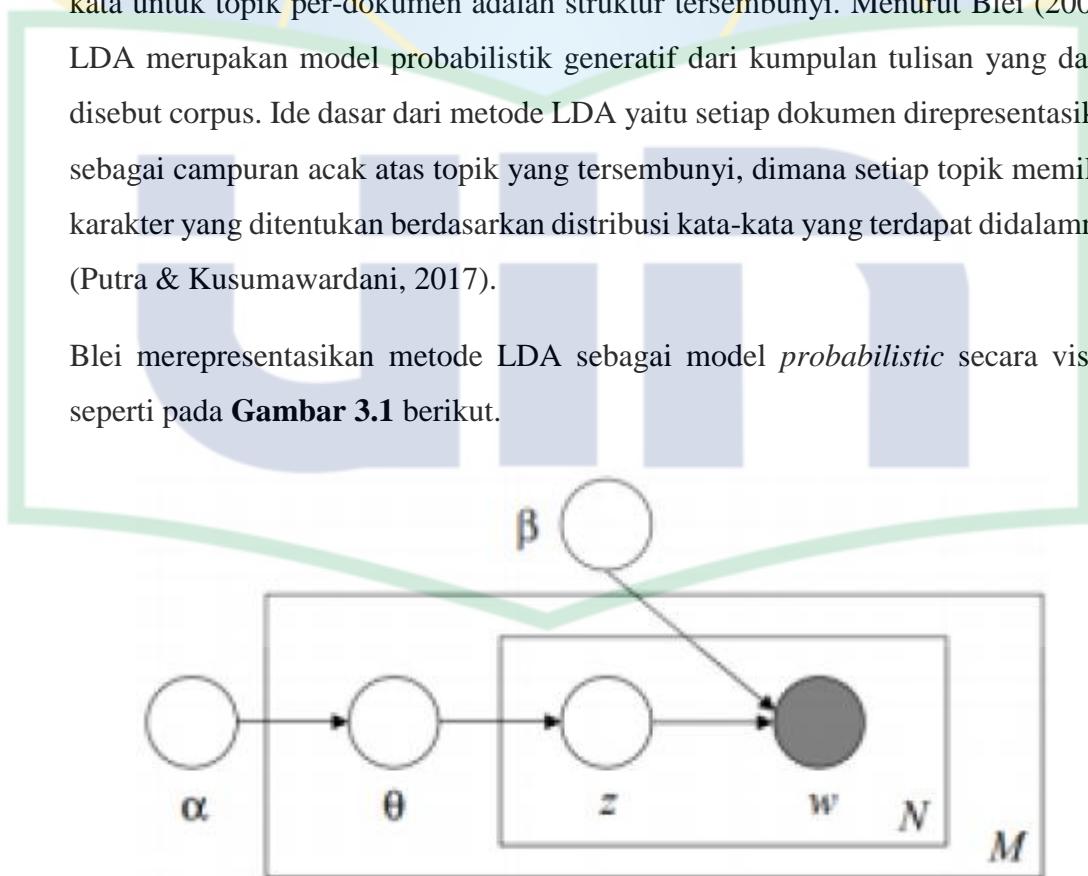
Konsep *topic modeling* menurut Blei (2003) terdiri dari entitas-entitas yaitu “kata”, “dokumen”, dan “corpora”. “Kata” dianggap sebagai unit dasar dari data diskrit dalam dokumen, didefinisikan sebagai item dari kosa kata yang diberi indeks untuk setiap kata unik pada dokumen. “Dokumen” adalah susunan N kata-kata. Sebuah corpus adalah kumpulan M dokumen dan corpora merupakan bentuk jamak dari corpus. Sementara “topic” adalah distribusi dari beberapa kosakata yang bersifat tetap. Secara sederhana, setiap dokumen dalam corpus mengandung proporsi tersendiri dari topik-topik yang dibahas sesuai kata-kata yang terkandung di dalamnya. Topic modeling merupakan sebuah topik yang terdiri dari kata-kata tertentu yang menyusun topik tersebut, dan dalam satu dokumen memiliki probabilitas masing-masing dari beberapa topik yang dihasilkan. Secara sederhana, *Topic Modeling* adalah algoritma yang memiliki tujuan untuk menemukan suatu topik yang tersembunyi dari rangkaian kata pada dokumen yang tidak terstruktur. Untuk menemukan topik yang berada antara teks tersebut dengan algoritma Topic Modeling menganalisis dari teks asli, bagaimana topik-topik dapat saling terhubung satu dengan yang lain, bagaimana tema-tema bisa berubah dari waktu ke waktu, sehingga bisa dikembangkan untuk pencarian, atau dengan meringkas teks yang terdapat dalam dokumen. (Putra & Kusumawardani, 2017).

Topic Modeling merupakan dokumen teks yang terdiri dari kata-kata, topik yang dapat dituliskan dalam banyak dokumen dapat dinyatakan dengan kombinasi kata-kata yang saling terkait dan teknik yang dapat digunakan untuk menyimpulkan suatu topik yang tersembunyi dalam sebuah dokumen teks. Karena topic modeling ini mewakili dari setiap dokumen sebagai kombinasi kompleks dari beberapa topik dan setiap topiknya sebagai kombinasi kompleks dari beberapa kata, kemudian juga digunakan sebagai alat text mining untuk mengklasifikasikan sebuah dokumen berdasarkan hasil kesimpulan topik (Nugroho & Alamsyah, 2018). Salah satu pemodelan topik yang paling populer yaitu Latent Dirichlet Allocation (LDA) (Putra & Kusumawardani, Analisis Topik Informasi Publik Media Sosial di Surabaya Menggunakan Pemodelan Latent Dirichlet Allocation (LDA), 2017).

## 2.11 Latent Dirichlet Allocation (LDA)

*Latent Dirichlet Allocation* merupakan salah satu metode yang dapat dipilih dalam melakukan analisis untuk dokumen yang memiliki ukuran sangat besar. LDA itu sendiri bisa digunakan untuk meringkas, melakukan klasterisasi, menghubungkan atau memproses data yang sangat besar dikarenakan LDA menghasilkan daftar topik yang diberi bobot untuk masing-masing dokumen. Distribusi yang digunakan yaitu distribusi *Dirichlet*, yang digunakan untuk memperoleh distribusi topik per-dokumen, dalam proses generatif, hasil yang didapatkan dari *Dirichlet* digunakan untuk mengalokasikan kata-kata dalam dokumen untuk topik yang berbeda. Pada LDA, dokumen-dokumen adalah objek yang bisa diamati, namun topik, distribusi topik per-dokumen, penggolongan setiap kata untuk topik per-dokumen adalah struktur tersembunyi. Menurut Blei (2003), LDA merupakan model probabilistik generatif dari kumpulan tulisan yang dapat disebut corpus. Ide dasar dari metode LDA yaitu setiap dokumen direpresentasikan sebagai campuran acak atas topik yang tersembunyi, dimana setiap topik memiliki karakter yang ditentukan berdasarkan distribusi kata-kata yang terdapat didalamnya (Putra & Kusumawardani, 2017).

Blei merepresentasikan metode LDA sebagai model *probabilistic* secara visual seperti pada **Gambar 3.1** berikut.



**Gambar 2.12** Model Representasi LDA (Blei, et al., 2003)

Dapat dilihat dari Gambar 2.12 diatas yaitu representasi metode LDA menurut (Blei, et al., 2003) dimana terdapat tingkatan pada pemodelan dengan LDA. Parameter  $\alpha$  dan  $\beta$  yaitu parameter distribusi topik yang berada pada tingkatan corpus, adalah kumpulan dari M dokumen. Untuk parameter  $\alpha$  yang digunakan dalam menentukan distribusi topik dokumen, jika nilai alpha semakin besar dalam suatu dokumen, menandakan bahwa campuran topik yang dibahas dalam dokumen semakin banyak. Untuk parameter  $\beta$  yang digunakan untuk menentukan distribusi kata dalam topik. Jika nilai beta semakin tinggi, maka semakin banyak kata-kata yang terdapat di dalam topik, namun jika nilai beta semakin kecil, maka semakin sedikit kata-kata yang terdapat di dalam topik sehingga topik tersebut mengandung kata-kata yang lebih spesifik. pada variabel  $\theta_m$  yaitu variabel yang berada di tingkat dokumen (M). Variabel  $\theta$  merepresentasikan distribusi topik untuk dokumen tersebut. Jika nilai  $\theta$  semakin tinggi, maka semakin banyak topik yang terdapat di dalam dokumen, jika nilai  $\theta$  semakin kecil, maka semakin spesifik pada topik tertentu. Pada variabel  $Z_n$  dan  $W_n$  yaitu variabel tingkat kata (N). Variabel Z merepresentasikan topik dari kata tertentu pada sebuah dokumen, pada variabel W merepresentasikan kata yang berkaitan dengan topik tertentu yang terdapat dalam dokumen. Berdasarkan penjelasan notasi sebelumnya, proses generatif pada LDA akan berkorespondensi pada joint distribution dari variabel yang tersembunyi dan variabel yang terobsesi. Berikut merupakan perhitungan probabilitas dari sebuah corpus berdasarkan notasi yang telah dijelaskan (Putra & Kusumawardani, Analisis Topik Informasi Publik Media Sosial di Surabaya Menggunakan Pemodelan Latent Dirichlet Allocation (LDA), 2017).

$$p(D|\alpha, \beta) = \prod_{d=1}^M \int p(\theta_d|\alpha) \left( \prod_{n=1}^{N_d} \sum_{Z_{dn}} p(Z_{dn}|\theta_d) p(W_{dn}|Z_{dn}, \beta) \right) d\theta_d(0,7)$$

Dapat dilihat bahwa pada notasi  $\beta$  mendeskripsikan topik, dimana pada setiap  $\beta$  merupakan distribusi dari sejumlah kata. Pada Variabel  $\theta_d$  adalah variabel level dokumen dengan satu kali sampel per dokumen yang merepresentasikan proporsi topik untuk dokumen ke  $d$ . Pada notasi  $Z_{dn}$  dan  $W_{dn}$  merupakan representasi

variabel di level kata dengan satu kali sampel untuk masing-masing kata pada setiap dokumen.

## 2.12 *Topic Coherence*

*Topic modeling* membahas mengenai kumpulan dari sebuah kata-kata dari sebuah dokumen ataupun corpus. Berdasarkan dari kata-kata yang terdapat dalam dokumen yang digunakan, penggalian dari relasi topik dilakukan dengan asumsi bahwa pada satu dokumen meliputi suatu set kecil dari topik yang ringkas, dimana topik-topik ini perlu dikorelasikan dengan interpretasi manusia. Pada penelitian ini akan menggunakan validasi topik dengan menggunakan *coherence topic* (Putra I. M., 2017).

*Topic coherence* yaitu dimana satu set dari kata-kata yang dihasilkan pada topik model dengan dinilai berdasarkan tingkat koherensi atau dalam diinterpretasi oleh manusia dengan tingkat kemudahannya. *Topic Coherence* mengukur nilai dari suatu topik dengan mengukur tingkat kesamaan semantik antara kata-kata yang ada dalam topik. Pengukuran ini dapat membantu dalam membedakan antara topik yang dapat diinterpretasi secara semantik dengan topik yang memiliki keterkaitan secara statistik (Putra, 2017). *Topic Coherence* merupakan suatu ukuran yang akan digunakan untuk mengevaluasi *Topic Modeling*, dimana jika *coherence* skor topik yang tinggi maka model yang dihasilkan tersebut yang baik. Menurut Wisdom (2017) *Topic Coherence* dapat dianggap memberikan kemampuan interpretasi lebih baik terhadap hasil dari *Topic Modelling* dibandingkan dengan *Perplexity*. Namun hasil dari matriks *perplexity* terkadang tidak memiliki korelasi yang baik pada interpretasi model oleh manusia (Listari, 2019).

## 2.13 *Use Case Diagram*

*Use case diagram* merupakan pemodelan untuk kelakukan (behavior) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Secara kasar, *use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu (Rossa, 2014).

**Tabel 2.2 Use Case Diagram**

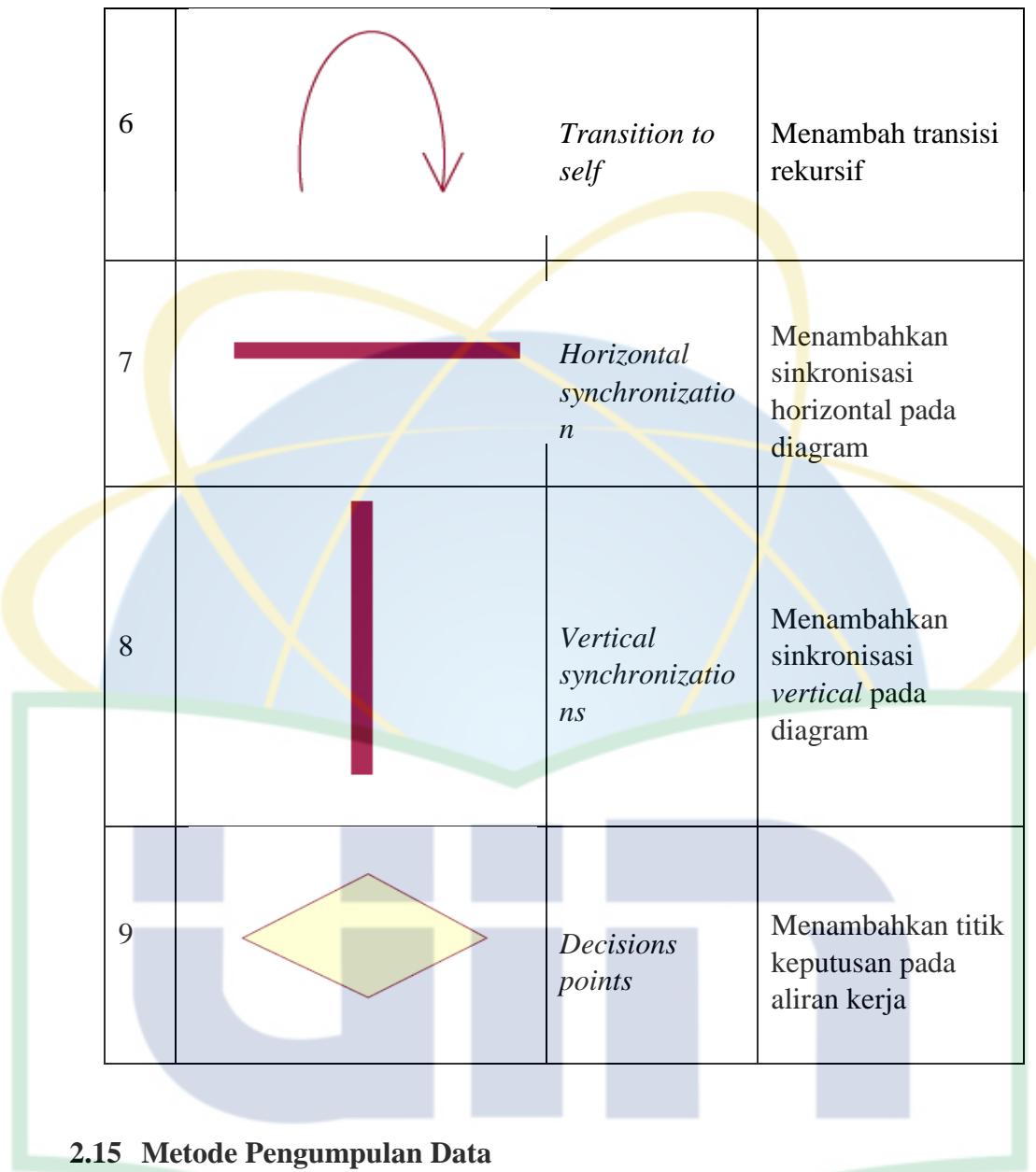
Simbol	Keterangan
	System Boundary menggambarkan batasan antara sistem dengan aktor
	Simbol ini menggambarkan interaksi antara aktor dengan sistem.
 Actor	Aktor menggambarkan pengguna sistem, dapat berupa manusia atau sistem terotomatisasi lain yang berinteraksi dengan sistem lain untuk berbagi, mengirim, dan menerima informasi.
	Menggambarkan hubungan antar aktor dan use case.

## 2.14 Activity Diagram

*Activity diagram* memodelkan alur kerja (*work flow*). Sebuah proses bisnis dan urutan aktifitas dalam suatu proses. Diagram ini sangat mirip dengan sebuah *flowchart* karena kita dapat memodelkan sebuah alur kerja dari aktifitas ke aktifitas lainnya atau dari suatu aktifitas ke dalam keadaan sesaat (*state*) (Rossa, 2014).

**Tabel 2.3 Activity Diagram**

No	Gambar	Nama	Fungsi
1		State	Menambahkan state untuk suatu objek
2		Activity	Menambahkan aktivitas baru pada diagram
3		Start state	Memperlihatkan dimana aliran kerja berawal
4		End state	Memperlihatkan dimana aliran kerja berakhir
5		State transition	Menambah transisi dari suatu aktivitas ke aktivitas yang lainnya



## 2.15 Metode Pengumpulan Data

Pengumpulan data adalah pengumpulan yang dilakukan untuk memperoleh informasi yang dibutuhkan dalam rangka mencapai tujuan penelitian. Tujuan yang diungkapkan dalam bentuk hipotesis merupakan jawaban sementara terhadap pertanyaan penelitian. Pengumpulan data dapat berupa suatu pernyataan (*statement*) tentang sifat, keadaan, kegiatan tertentu dan sejenisnya. Pengumpulan data dilakukan untuk memperoleh informasi yang dibutuhkan dalam rangka mencapai tujuan penelitian (Sugiyono, 2012).

## 2.16 Studi Pustaka

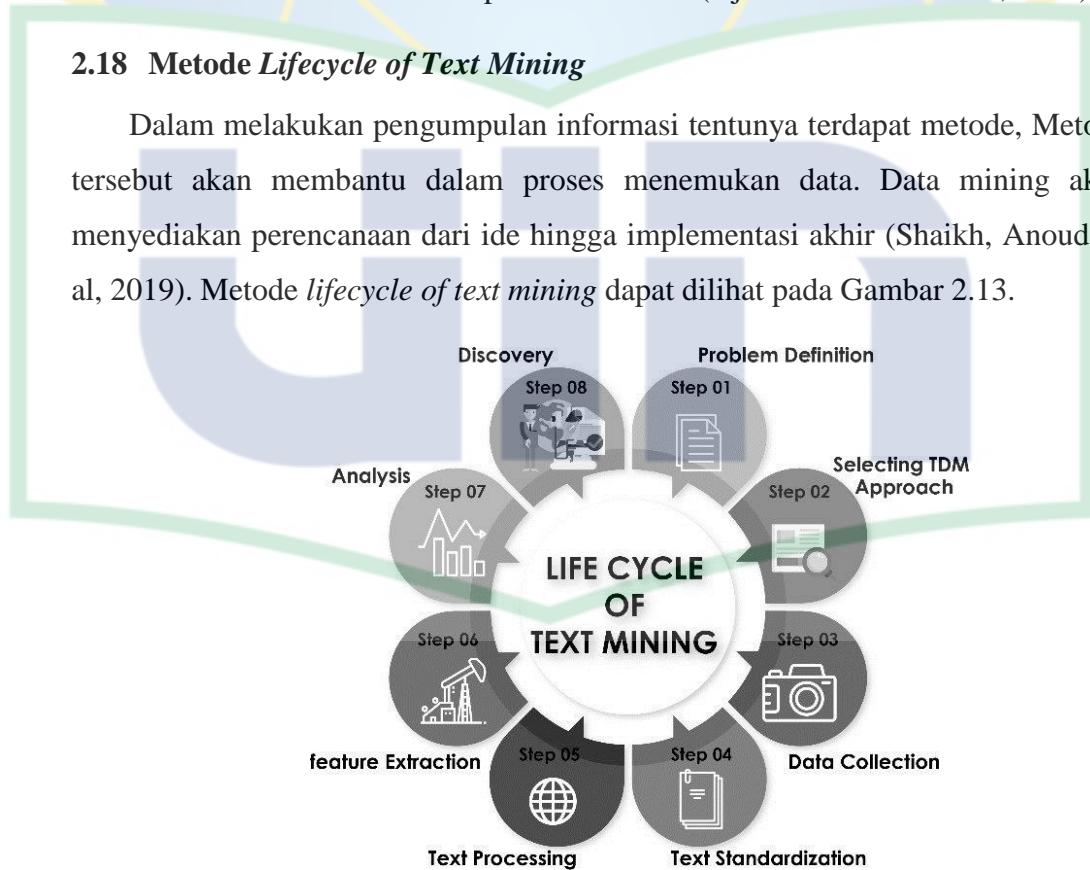
Studi Pustaka adalah kajian teoritis, referensi serta literatur ilmiah lainnya yang berkaitan dengan budaya, nilai dan norma yang berkembang pada situasi sosial yang teliti, selain itu studi pustaka sangat penting dalam melakukan penelitian, hal ini dikarenakan penelitian tidak akan lepas dari literatur-literatur ilmiah (Sugiyono, 2012).

## 2.17 Pengamatan

Pengamatan merupakan suatu cara untuk mengumpulkan data dengan jalan mengamati kegiatan yang sedang berlangsung. Pengamatan dapat dilakukan dengan partisipasi maupun tidak (Sudaryono, 2015). Pengamatan atau observasi merupakan cara lain untuk mengambil atau menggali data. Pengamatan dapat dilakukan dengan datang ke tempat para responden atau objek penelitian berada, kemudian melihat atau meneliti perilaku mereka (Djiwandono & Istiarto, 2015).

## 2.18 Metode *Lifecycle of Text Mining*

Dalam melakukan pengumpulan informasi tentunya terdapat metode, Metode tersebut akan membantu dalam proses menemukan data. Data mining akan menyediakan perencanaan dari ide hingga implementasi akhir (Shaikh, Anoud, et al, 2019). Metode *lifecycle of text mining* dapat dilihat pada Gambar 2.13.



Gambar 2.13 Metode *Lifecycle of Text Mining*

Tahapan metode *lifecycle of text mining* terbagi menjadi beberapa tahapan, yaitu: (Shaikh, Anoud, et al, 2019)

#### **2.18.1 *Problem Definition***

Tahap problem definition merupakan suatu proses identifikasi yang harus dilakukan untuk menentukan masalah yang menjadi dasar perumusan pada penelitian text mining. Tahap ini dilakukan dengan mendefinisikan masalah secara rinci untuk menentukan solusi pendekatan yang tepat dalam text mining. Identifikasi masalah yang dilakukan secara rinci akan memudahkan dalam menemukan solusi yang akurat dari masalah yang telah diidentifikasi.

#### **2.18.2 *Selecting Text Mining Approach***

Tahap selecting text mining Approach merupakan suatu proses yang dilakukan untuk menentukan metode pendekatan text mining yang tepat sebagai solusi dari masalah yang telah didefinisikan pada tahap problem definition. Tahap selecting text mining approach merupakan proses lanjutan dari tahap problem definition. Setelah tahap identifikasi masalah dilakukan dan ditemukan solusi yang tepat, maka dilanjutkan dengan proses selecting text mining approach agar didapatkan metode terbaik yang nantinya digunakan pada penelitian text mining.

#### **2.18.3 *Data Collection***

Tahap data collection merupakan tahap pengumpulan data yang digunakan sebagai input dalam analisa text mining. Tahap pengumpulan data dilakukan dengan menentukan jenis data yang digunakan serta bahan - bahan yang akan dibutuhkan pada penelitian text mining. Tahap pengumpulan data merupakan tahap yang penting untuk dilakukan dalam sebuah penelitian agar diperoleh informasi yang dibutuhkan untuk mencapai tujuan penelitian. Data yang diperoleh selanjutnya digunakan untuk proses analisa text mining.

#### **2.18.4 *Text Standardization***

Tahap text standardization merupakan tahap penyesuaian format data yang digunakan dengan bertujuan untuk mempermudah proses atau tahap text mining selanjutnya. Dokumen yang dikumpulkan dalam data collection bisa diperoleh

dalam berbagai bentuk, oleh sebab itu pada tahap ini semua data diubah menjadi format yang sesuai dengan standar format data text mining yang telah ditetapkan.

#### **2.18.5 *Text Preprocessing***

Tahap text-preprocessing merupakan tahapan awal dalam mengolah data input untuk mempersiapkan teks menjadi data yang lebih tersuktur sebelum diolah lebih lanjut. Preprocessing text dilakukan untuk tujuan penyeragaman dan kemudahan pembacaan dalam proses text mining selanjutnya. Preprocessing terdiri dari beberapa tahapan, yaitu mengubah semua huruf dalam dokumen menjadi huruf kecil, pemotongan string input berdasarkan tiap kata yang menyusunnya, dan tahap preprocessing lainnya yang membantu dalam pemrosesan lebih lanjut.

#### **2.18.6 *Feature Extraction***

Tahap feature extraction merupakan tahap pengambilan ciri – ciri yang unik dari data yang akan diolah dengan tujuan memperkecil jumlah data, mengambil informasi yang terpenting dari data yang diolah dan mempertinggi presisi pengolahan. Dengan feature extraction dapat diperoleh data yang lebih spesifik sehingga data yang tidak relevant tidak akan digunakan pada proses selanjutnya.

#### **2.18.7 *Analysis***

Tahap analysis merupakan analisa teks yang bertujuan untuk mengekstrak informasi yang dapat dibaca oleh sistem dari teks yang tidak terstruktur menjadi teks yang terstruktur untuk mengaktifkan pendekatan berbasis data dalam mengelola data. Fakta dan hubungan yang relevan ditentukan kemudian diekstraksi dalam bentuk terstruktur untuk peninjauan dan analisis yang lebih tepat. Fakta terhubung dengan cara baru untuk mensintesis pengetahuan sehingga menghasilkan wawasan yang dapat ditindaklanjuti. Untuk mengatasi ambiguitas bahasa manusia dan mencapai akurasi tinggi pada domain tertentu.

#### **2.18.8 *Discovery***

Tahap discovery merupakan tahap yang dilakukan untuk menemukan pola atau pengetahuan (knowledge) dari keseluruhan teks. Proses text mining diawali

dengan suatu data teks yang menghasilkan keluaran berupa pola sebagai hasil interpretasi atau evaluasi. Hasil interpretasi selanjutnya akan disajikan ke pengguna dalam bentuk visual. Namun, apabila hasil keluaran dari penemuan pola belum sesuai untuk aplikasi, dilanjutkan evaluasi dengan melakukan iterasi ke satu atau beberapa tahap sebelumnya hingga diperoleh hasil keluaran yang sesuai untuk aplikasi.

## 2.19 Literatur Sejenis

Pada penelitian ini, penulis melakukan studi literatur sejenis sebagai bahan untuk mengevaluasi serta referensi dari penelitian yang akan dilakukan. Hal ini bertujuan memperlihatkan perbedaan pada penelitian yang dilakukan penulis dibandingkan dengan penelitian-penelitian sebelumnya. Studi literatur yang digunakan penulis berupa jurnal, skripsi ataupun karya publikasi lainnya.

Penelitian yang dilakukan oleh Trias Santika (2019) mengenai Evaluasi *perplexity* untuk pemodelan topik diskusi agama Islam di media sosial Twitter Indonesia tahun 2006 – 2018 menggunakan *Latent Dirichlet Allocation* (LDA). Pembicaraan mengenai topik terkait diskusi agama Islam di Indonesia khususnya di media sosial Twitter cukup banyak dengan jumlah teks mencapai 421.053 *tweet*. Penelitian ini menggunakan data *tweet* tersebut yang diambil dari Maret 2006 hingga November 2018. Pada penelitian ini membuat pemodelan topik dengan metode LDA untuk mengamati pola perubahan perbincangan masyarakat di media sosial Twitter ketika membicarakan agama Islam disetiap tahunnya. Jumlah topik yang didapat pada penelitian ini menggunakan *perplexity* untuk mengevaluasi topik dan menentukan jumlah iterasinya. Hasil penelitian ini menunjukkan bahwa masing-masing tahun memiliki dua topik optimal, probabilitas kata-kata terkait agama Islam selalu muncul disetiap tahunnya dan terdapat pula kata-kata yang berkaitan dengan politik juga selalu muncul. Kesimpulan evaluasi *perplexity* untuk pemodelan topik menggunakan metode LDA pada studi kasus tersebut dapat menunjukkan bahwa pada masa ini pengguna Twitter sudah tidak lagi murni membicarakan agama Islam di media sosial tetapi sudah mencapurnya dengan kata-kata yang berhubungan dengan politik.

Penelitian yang dilakukan oleh Muhammad Diaz Elprana Rangkuti (2020) mengenai analisis topik komentar video beberapa akun Youtube *e-commerce* Indonesia menggunakan metode *Latent Dirichlet Allocation* (LDA). Beberapa perusahaan *e-commerce* di Indonesia saat ini mulai melakukan pemasaran produk atau aplikasi melalui Youtube. Penelitian ini menggunakan data pendapat masyarakat tentang *e-commers* di Indonesia yang terdapat pada kolom komentar Youtube dari tanggal 4 sampai 26 Oktober 2019. Tujuan penelitian ini untuk melihat topik apa saja yang sering dibahas masyarakat tentang *e-commers* di Indonesia. Metode *Latent Dirichlet Allocation* (LDA) digunakan untuk menentukan pemodelan topik dengan melakukan validasi topik menggunakan *topic coherence*. Topik dengan *coherence score* yang tertinggi menjadi topik uang paling optimal. Hasil yang didapatkan bahwa LZD memiliki *coherence score* tertinggi 0.533, TKPD memiliki *coherence score* 0.464, SHP memiliki *coherence score* 0.46, BKLPK memiliki *coherence score* 0.42.

Penelitian yang dilakukan oleh I Made Kusnanta Bramantya Putra (2017) mengenai Analisis Topik Informasi Publik Media Sosial di Surabaya Menggunakan Pemodelan *Latent Dirichlet Allocation* (LDA). Pada Radio Suara Surabaya yang mengembangkan tentang siaran interaktif dengan basis jurnalistik masyarakat, yang melibatkan partisipasi dari masyarakat dalam melaporkan peristiwa atau kejadian kepada penyiar radio yang bertugas. Jumlah laporan perhari yang tinggi dan berbagai macam topik dari laporan tersebut menyebabkan kesusahan dalam mengidentifikasi suatu topiknya sehingga akan menghabiskan banyak waktu jika dilakukan secara manual oleh manusia. Dengan demikian sangat dibutuhkan pemodelan topik yang dapat secara otomatis mengklasifikasi pesan media sosial tersebut ke dalam topik-topik yang muncul dari hasil pemodelannya. Pemodelan topik dilakukan dengan metode *Latent Dirichlet Allocation* (LDA). Dengan metode LDA didapatkan hasil jumlah topik yang terdapat dalam pesan media sosial yaitu 4 topik dengan nilai perplexity terbaik yaitu sebesar 213.41 dan diuji kemudahannya untuk diinterpretasi oleh manusia dengan uji koherensi topik yang terdiri dari word intrusion task dan topic intrusion task. Kesimpulannya dengan menggunakan metode LDA, model yang didapatkan dapat diinterpretasi manusia dengan baik.

Penelitian yang dilakukan oleh Agustina (2017) mengenai Analisis dan visualisasi suara pelanggan pada pusat layanan pelanggan dengan pemodelan topik menggunakan *Latent Dirichlet Allocation* (LDA) studi kasus: PT. Petromkimia Gresik. Pada perusahaan tersebut salah satunya yaitu memproduksi pupuk dimana pupuk merupakan kebutuhan penting dalam bidang pertanian. Kepuasan pelanggan menjadi salah satu aspek yang diperhatikan oleh perusahaan salah satunya yaitu dengan adanya aplikasi Pusat Layanan Pelanggan yang berguna untuk mengoptimalkan suara pelanggannya. Pada pencatatan suara pelanggan yang telah masuk dengan berbagai media tetapi belum terdapat analisis terhadap topik apakah yang sering pelanggan suarakan maka dilakukan analisis topik dengan menggunakan pemodelan LDA. Didapatkan hasil yaitu 35 topik yang berhasil diidentifikasi kemudian dikelompokkan ke dalam 7 kategori. Nilai *perplexity* yang diperoleh sebesar 34.92 dengan standar deviasi 0.49 pada 20 iterasi dan nilai akurasinya sebesar 83.7%. Kemudian hasil identifikasi divisualisasikan dalam dashboard berbasis web.

Penelitian yang dilakukan oleh Guntur Budi Herwanto Herwanto (2018) mengenai dokumen clustering dengan *Latent Dirichlet Allocation* dan *ward hierarichal clustering*. Pada saat ini terdapat sangat banyak konten informasi dalam bentuk berita dari berbagai sumber setiap harinya. Dengan banyaknya konten tersebut menuntut organisasi konten yang baik untuk pencarian informasi yang diinginkan dapat dilakukan dengan mudah maka dilakukan clustering dokumen. Penelitian ini menggunakan kombinasi pemodelan topik latent dirichlet allocation dengan *ward hierarchical clustering*. Pada LDA digunakan untuk representasi vektor dokumen berupa distribusi topik dengan tujuan mengurai dimensi vektor yang biasanya terlalu panjang jika menggunakan tf-idf. Didapatkan hasil *silhouette coefficient* yang baik yaitu 0.7. Performa *silhouette coefficient* pada representasi pemodelan topik lebih baik dibandingkan dengan representasi dengan tf-idf.

Penelitian yang dilakukan oleh Wirasakti, Permadi, Hartanto, & Hartatik (2020) mengenai Pembuatan Kata Kunci Otomatis Dalam Artikel Dengan Pemodelan Topik. Pada penulisan ini yaitu untuk mengetahui kata kunci yang

cocok untuk digunakan dalam sebuah publikasi artikel pada sebuah blog dengan menggunakan model LDA yaitu model probabilitas yang dapat menghasilkan beberapa topik. Data yang diambil yaitu berasal dari blog/artikel. lalu melakukan pemotongan artikel per bagian kemudian dilakukan preprocessing data, dan melakukan pengubahan vektor data menjadi corpus untuk dimodelkan dengan LDA, yang kemudian dilakukan clustering dengan menggunakan K-Means. Didapatkan hasil model LDA yaitu 4 topik dengan 8 kata. Yang memiliki nilai probabilitas tertinggi diantaranya mesin (0.09375857), maksimal (0.064600445), mazda (0.10009629), varian (0.07572112), cx-8 (0.10170187), mazda (0.101048954), mobil (0.09820121), dan mazda (0.05679208).

Pada referensi di atas kemudian dijadikan dalam bentuk Tabel 2.1 dibawah ini yang merupakan tabel rangkuman perbandingan penelitian terdahulu dengan penulisan ini tentang *Topic Modeling menggunakan Latent Dirichlet Allocation* (LDA) pada abstrak dokumen skripsi mahasiswa Statistika Universitas Islam Indonesia.

**Tabel 2.4** Literatur Sejenis

No	Penulis	Judul	Metode	Kelebihan	Kekurangan
1.	Trias Santika (2019)	Evaluasi perplexity untuk pemodelan topik diskusi agama Islam di media sosial Twitter Indonesia tahun 2006 – 2018 menggunakan Latent Dirichlet Allocation (LDA)	<i>Latent Dirichlet Allocation</i> (LDA)	Menggunakan jumlah data yang lebih banyak yaitu 421.053 tweet	Tidak menggunakan validasi topik coherence, tidak menggunakan bigram & Trigram model, tidak menggunakan word embedding

2.	Muhammad Diaz Elprana Rangkuti (2020)	Analisis topik komentar video beberapa akun Youtube <i>e-commerce</i> Indonesia menggunakan metode <i>Latent Dirichlet Allocation</i> (LDA)	<i>Latent Dirichlet Allocation</i> (LDA)	Menggunakan data dari Youtube	Tidak Menggunakan <i>bigram</i> dan <i>trigram</i> model, tidak menggunakan <i>word embedding</i>
3.	I Made Kusnanta Bramantya Putra (2017)	Analisis Topik Informasi Publik Media Sosial di Surabaya Menggunakan Pemodelan <i>Latent Dirichlet Allocation</i> (LDA)	<i>Latent Dirichlet Allocation</i> (LDA)	Menggunakan dua validasi Topik yaitu <i>Topic coherence</i> dan <i>Perplexity</i>	Tidak Menggunakan <i>bigram</i> dan <i>trigram</i> model.
4.	Agustina (2017)	Analisis dan visualisasi suara pelanggan pada pusat layanan pelanggan dengan pemodelan topik	<i>Latent Dirichlet Allocation</i> (LDA)	Menggunakan data dari berbagai media, SMS, Telepon dan media massa. Visualisasi	Tidak menggunakan <i>bigram</i> dan <i>trigram</i> model. Tidak menggunakan validasi Topik

				data dalam <i>dashboard</i> berbasis web	dengan Topic coherence
5.	Guntur Budi Herwanto (2018)	Dokumen <i>Clustering</i> Dengan <i>Latent Dirichlet Allocation</i> dan <i>Ward Hierarchical Clustering</i>	LDA dan <i>Ward Hierarchical Clustering</i>	Menggunakan kombinasi pemodelan topik <i>latent dirichlet allocation</i> (LDA) dengan <i>ward hierarchical clustering</i>	Tidak menggunakan bigram dan trigram model. Tidak menggunakan validasi Topik dengan Topic coherence
6.	Wirasakti, Permadi, Hartanto, & Hartatik (2020)	Pembuatan Kata Kunci Otomatis Dalam Artikel Dengan Pemodelan Topik	LDA dan <i>K-Means Clustering</i>	Menggunakan kombinasi pemodelan topik <i>latent dirichlet allocation</i> (LDA) dan <i>K-Means Clustering</i>	Tidak menggunakan bigram dan trigram model. Tidak menggunakan validasi Topik dengan Topic coherence

## **BAB 3**

### **METODOLOGI PENELITIAN**

Pada bab ini, penulis akan membahas tentang metode penelitian yang digunakan untuk menyusun skripsi ini. Metode penelitian yang digunakan dibagi menjadi dua bagian, yaitu metode pengumpulan data dan metode simulasi dengan didukung adanya kerangka berfikir pada akhir bab ini. Penjelasan lebih lengkap dalam bab ini adalah sebagai berikut:

#### **3.1 Metode Pengumpulan Data**

##### **3.1.1 Pengamatan**

Metode pengumpulan data dalam penelitian ini menggunakan pengamatan. Pada tahap pengamatan, penulis mengamati dan mengambil data dari API Twitter tentang tweet netizen terhadap *keyword omnibus law*. Penulis mengambil data pada periode 05 Oktober 2020 hingga 16 Oktober 2020. Penulis memanfaatkan fitur *developer* Twitter dalam pengambilan data yang dapat diakses pada website <https://developer.twitter.com/>. Data yang telah penulis dapatkan dari API Twitter, kemudian disimpan ke dalam format .CSV. Total data yang penulis ambil sebanyak 3.466 tweet.

##### **3.1.2 Studi Pustaka**

Penulis melakukan studi pustaka dengan melakukan pencarian data secara manual, yaitu mempelajari buku-buku yang berhubungan dengan konsep, analisis, dan simulasi serta jurnal dan skripsi yang sejenis. Selain itu, penulis juga melakukan pencarian data secara *online* melalui *browsing* internet dengan mengunjungi website yang berhubungan dengan topik ataupun konten yang membantu dalam penelitian ini dan melakukan pembelajaran terhadap jurnal dan skripsi yang relevan sesuai dengan topik penelitian yang dijadikan sebagai bahan acuan dalam pembuatan penelitian ini.

## 3.2 Metode *Lifecycle of Text Mining*

Metode yang penulis gunakan pada penelitian ini adalah metode *lifecycle of text mining*. Penulis memilih menggunakan metode *lifecycle of text mining* untuk menyesuaikan kebutuhan dalam melakukan penelitian ini. Metode *lifecycle of text mining* yang digunakan didasarkan pada landasan teori (2.18) yang telah dijelaskan sebelumnya. Berikut ini merupakan langkah-langkah metode *lifecycle of text mining*:

### 3.2.1 *Problem Definition*

Pada Tahap *problem definition*, penulis mendefinisikan masalah sesuai dengan latar belakang yang telah dijelaskan pada pendahuluan (1.1) maka dikembangkan suatu rumusan masalah pada penelitian ini yaitu bagaimana mengelompokan topik pembicaraan masyarakat tentang omnibus law di media sosial Twitter menggunakan metode LDA dan bagaimana mengambil kesimpulan tentang topik yang ditemukan.

### 3.2.2 *Selectinng Text Data Mining Approach*

Pada tahap *selectinng text data mining approach*, penulis menentukan pendekatan data mining yang digunakan pada penelitian ini mendapat reverensi dari hasil penelitian sebelumnya (dalam table literatur sejenis 2.4). Penelitian sebelumnya sama-sama menggunakan metode Latent Dirichlet Allocation (LDA). Adapun masalah pada penelitian ini adalah menentukan *Topic Modelling* dengan *Latent Dirichlet Allocation* (LDA) dalam satu studi kasus tweet tentang *omnibus law*. Dimana dalam penelitian sebelumnya kesimpulannya dengan menggunakan metode LDA, model yang didapatkan dapat diinterpretasi manusia dengan baik.

### 3.2.3 *Data Collection*

Pada tahap *data collection*, penulis mengumpulkan data dengan mengambil tweet-tweet dari API Twitter sebelumnya dan penulis jadikan sebagai input data dalam aplikasi berbasis Python pada penelitian ini, *packages* yang digunakan dari python diantaranya yaitu packages *Gensim* dan *pyLDAvis*. Penulis melakukan *input*

data menggunakan beberapa fitur dari *Gensim* yaitu *gensim.corpora* dan *gensim.model*. Pada modul *gensim.corpora* yang digunakan untuk membangun *dictionary* dari data teks sebelum melakukan proses pembuatan model LDA dengan memanggil modul *Dictionary*. Kemudian modul *gensim.model* yang digunakan untuk membangun model LDA dengan memanggil dari modul *LdaModel*. Kemudian akan dipanggil pada saat melakukan proses perhitungan *coherence* model dengan meng-import modul *CoherenceModel*. Selanjutnya *output data* penulis menggunakan *library PyLDAvis* untuk visualisasi model topik interaktif. Pada PyLDAvis ini dirancang untuk dapat membantu pengguna menafsirkan topik dalam model topik yang sesuai dengan kumpulan data teks. *Packages* ini juga mengekstraksi informasi dari model topik LDA yang dipasang untuk dapat menginformasikan visualisasi berbasis web interaktif. Visualisasi yang dimaksudkan yaitu untuk digunakan dalam notebook *IPython* namun dapat juga disimpan dalam file *HTML* yang berdiri sendiri untuk memudahkan berbagi (pypi.org, 2018)

### 3.2.4 *Text Standardization*

Pada Tahap *text standardization*, penulis menyesuaikan format data yang digunakan dengan bertujuan untuk mempermudah proses atau tahap *text mining* selanjutnya. Dokumen yang telah dikumpulkan dalam tahap *data collection* diperoleh dalam berbagai bentuk, oleh sebab itu pada tahap ini semua data diubah menjadi format yang sesuai dengan standar format data *text mining* yang telah ditetapkan yaitu dengan format CSV.

### 3.2.5 *Text Processing*

Pada Tahap *text processing*, penulis mengolah data input untuk mempersiapkan teks menjadi data yang lebih tersuktur sebelum diolah lebih lanjut. Preprocessing text dilakukan untuk tujuan penyeragaman dan kemudahan pembacaan dalam proses *text mining* selanjutnya. Preprocessing terdiri dari beberapa tahapan, yaitu *case folding*, *Remove punctuation*, *stopword*, dan *tokenizing*.

### 3.2.6 Feature Extraction

Pada Tahap *feature extraction*, penulis melakukan proses pengambilan ciri – ciri yang unik dari data yang akan diolah dengan mengubah data yang berbentuk kata menjadi dalam bentuk numerik dengan menggunakan pembobotan TF-IDF, mengambil informasi yang terpenting dari data yang diolah dan mempertinggi presisi pengolahan menggunakan bigram dan trigram model. Dengan feature extraction dapat diperoleh data yang lebih spesifik sehingga data yang tidak relevant tidak akan digunakan pada proses selanjutnya.

### 3.2.7 Analysis

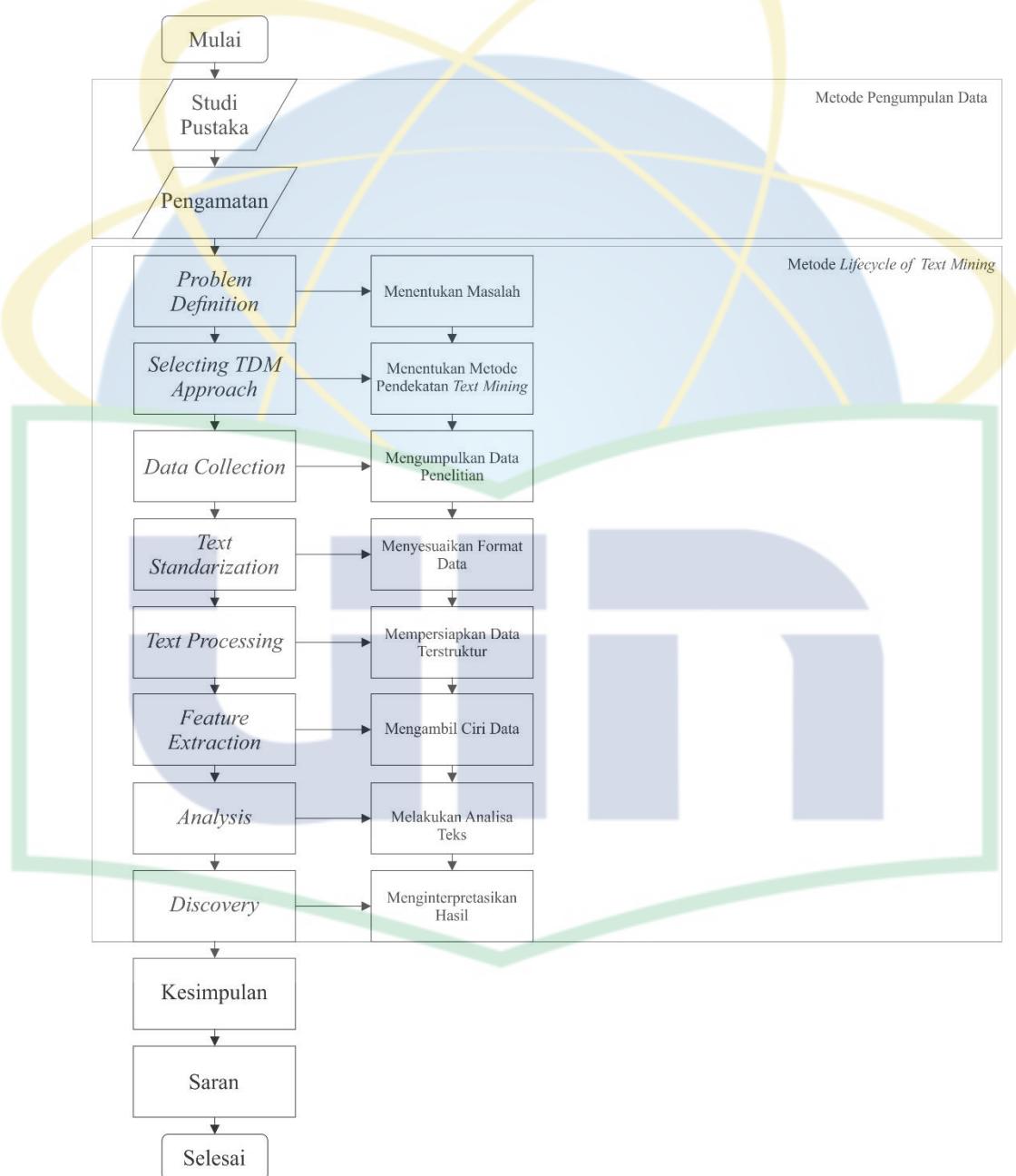
Pada tahap *analysis*, penulis melakukan *topic modelling* menggunakan metode *Latent Dirichlet Allocation* (LDA) menggunakan aplikasi berbasis *Python* yang telah penulis buat sebelumnya. *Analysis* dilakukan dengan mengambil data mentah yang telah tersimpan di database untuk menentukan hasil pemodelan dapat dilakukan dengan melihat pada visualisasi dari grafik *coherence score*. Dimana jika *coherence score* topik yang tinggi maka model yang dihasilkan tersebut yang baik. Setelah didapatkan jumlah topiknya berdasarkan grafik *coherence score*, kemudian akan didapatkan model *Latent Dirichlet Allocation* (LDA) berdasarkan banyak topiknya.

### 3.2.8 Discovery

Pada tahap *discovery*, penulis membuat kesimpulan dari pola atau pengetahuan (*knowledge*) yang ditemukan dari keseluruhan teks sebagai hasil interpretasi atau evaluasi. Pada tahap analysis bertujuan untuk meyakinkan hasil dari pemodelan topik menggunakan *Latent Dirichlet Allocation* (LDA) sesuai dengan yang telah penulis konspekan sebelumnya. Lalu, Pada tahap ini hasil yang mana merupakan tahapan terakhir dari metode *lifecycle of text mining*, penulis melakukan analisis terhadap output – output berdasarkan pengujian yang telah dilakukan.

### 3.3 Kerangka Berpikir

Setelah menguraikan penjelasan mengenai metode pengumpulan data dan metode simulasi, selanjutnya penulis menuangkan hasil dari penjelasan metode penelitian yang digunakan dalam penelitian ini ke dalam sebuah kerangka pemikiran pada gambar 3.1:



**Gambar 3.1** Kerangka Berpikir

## **BAB 4**

### **PERANCANGAN DAN PENGUJIAN**

#### **4.1 Problem Definition**

Pada tahap awal metode *lifecycle of text mining* ini yaitu mendefinisikan masalah (*problem definition*), penulis melakukan identifikasi masalah berdasarkan latar belakang penelitian. Studi kasus penelitian mengenai *omnibus law* yang sedang ramai diperbincangkan, bahkan menurut laporan dari Twitter Indonesia tagar tentang omnibuslaw menjadi pembahasan terpopuler sepanjang 2020 (cnnindonesia.com, 2020). Penulis mendefinisikan masalah penelitian pada text mining dengan topic modelling untuk mengetahui topik pembicaraan warganet Twitter tentang *omnibus law* menggunakan metode Latent Dirichlet Allocation (LDA). Data yang digunakan pada penelitian ini adalah komentar berbahasa Indonesia tentang kebijakan pemerintah yaitu Undang Undang *Omnibus Law*.

#### **4.2 Selectinng Text Data Mining Approach**

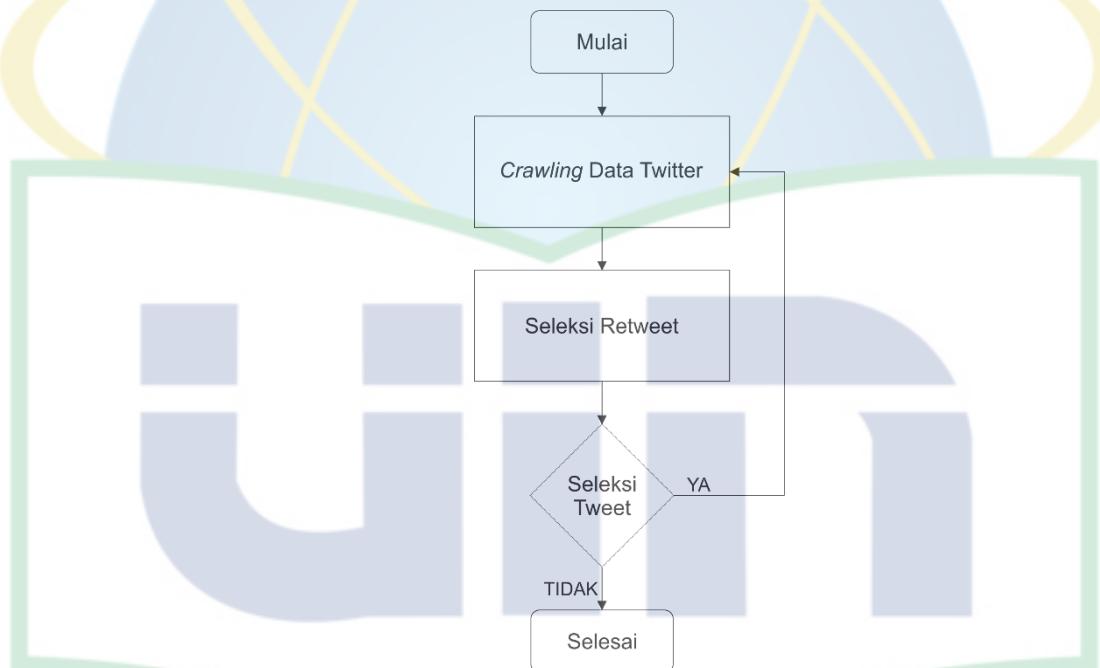
Pada tahap *selectinng text data mining approach*, penulis menentukan pendekatan data mining yang digunakan pada penelitian ini mendapat reverensi dari hasil penelitian sebelumnya (dalam table literatur sejenis 2.4). Penelitian sebelumnya sama-sama menggunakan metode Latent Dirichlet Allocation (LDA). Adapun masalah pada penelitian ini adalah menentukan *Topic Modelling* dengan *Latent Dirichlet Allocation* (LDA) dalam satu studi kasus tweet tentang *omnibus law*. Dimana dalam penelitian sebelumnya kesimpulannya dengan menggunakan metode LDA, model yang didapatkan dapat diinterpretasi manusia dengan baik.

#### **4.3 Data Collection**

Pada penelitian ini data yang diperoleh sebagai data input yaitu komentar-komentar masyarakat (*netizen*) terhadap kebijakan pemerintah tentang undang-undang *omnibus law* dari cuitan di Twitter. Dimana proses pengambilan data ini dinamakan dengan teknik crawling. Pada pengambilan data dari Twitter peneliti memanfaatkan yang namanya Twitter API. Periode pengambilan data dari tanggal 5 Oktober 2020 sampai 16 Oktober 2020. Selain data-data *tweet* yang dikumpulkan

pada penelitian ini yaitu kamus kata dasar KBBI, stopwords, dan kamus stopwords sastrawi. Adapun tahap pengambilan data adalah sebagai berikut:

- *Crawling Data*: Data dalam penelitian ini diambil dari twitter dengan menggunakan *keyword omnibus law*.
- Seleksi Retweet: Data *tweet* yang diambil secara bersamaan dilakukan proses seleksi retweet oleh sistem sehingga hanya tweet asli yang akan disimpan.
- Seleksi Tweet: Data *tweet* yang isinya hanya berupa mention atau hashtag ataupun keduanya diseleksi atau dibuang dari *database*.
- Tahapan *crawling* data selesai. *Flowchart* dari tahapan crawling data adalah sebagai berikut:



**Gambar 4.1** Flowchart Tahap Pengambilan Data

Berikut merupakan contoh beberapa data awal penulisan yang digunakan:

**Tabel 4.1** Data Penulisan

Tweet
Omnibus Law Pengaruhi IHSG, Simak Rekomendasi Saham Hari Ini <a href="https://t.co/V5gw44wm5t">https://t.co/V5gw44wm5t</a>

Setelah ada versi ke-3 yg 1052 halaman, sekarang ada versi ke-4 yg 1035 halaman. Entah mana yg benar. Omnibus Law cipta kerja ini memang barang ghoib.

Jokowi Sebut Rakyat Termakan Hoax Omnibus Law, ASMaPi: Hoax Versi Penguasa atau Rakyat Jelata? <https://t.co/zT8G9LDhxF>

Di dalam UU sapa jagat ini beberapa pasal menyebutkan keistimewaan yang didapatkan oleh para warga asing. <https://t.co/JGD17YGoXm>

Kompas saja memuat tulisan kelemahan2 Omnibus Law. Mulai dari Kehutanan, pertanian, an hak2 buruh dll. <https://t.co/0HVwxHjgwk>

Covid in my country is joke> yg akhrnya muncul dan terbanyak skg ke 9 di dunia> dijadikan cuan oleh tikus sampah> rakyat suruh diem di rumah> diemdiem pemerintah mau sah in omnibus law  
#jegalomnibuslawsampaibatal

#### **4.4 *Text Standardization***

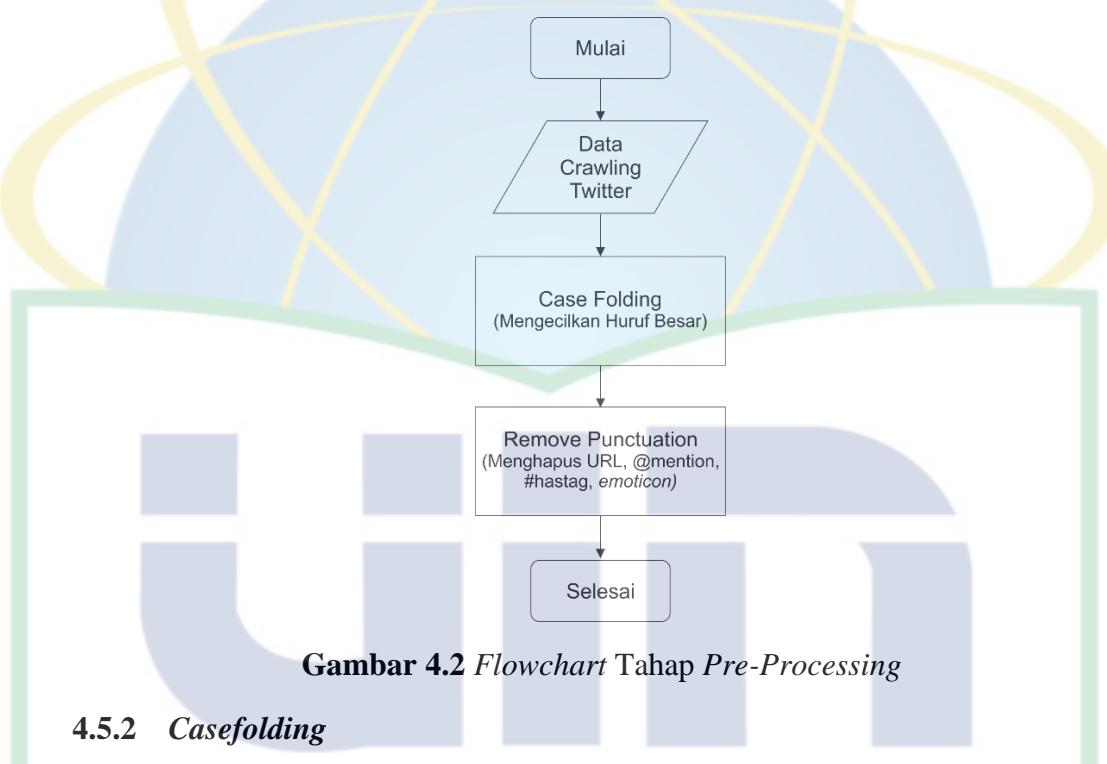
Pada Tahap *text standardization*, penulis menyesuaikan format data yang digunakan dengan bertujuan untuk mempermudah proses atau tahap *text mining* selanjutnya. Dokumen yang telah dikumpulkan dalam tahap *data collection* diperoleh dalam berbagai bentuk, oleh sebab itu pada tahap ini semua data diubah menjadi format yang sesuai dengan standar format data *text mining* yang telah ditetapkan yaitu dengan format CSV.

#### **4.5 *Text Processing***

Pada tahap *text processing* dimana tahap yang sangat penting dilakukan untuk menghilangkan kata-kata dalam dokumen yang tidak dibutuhkan. Tahap ini merupakan tahap pertama yang dilakukan dalam mulai menganalisis. Pada tahap preprocessing ini terdiri dari beberapa proses seperti *case folding*, *remove punctuation*, *stopword*, dan *tokenizing*.

#### 4.5.1 Tahap Pre-Processing

- *Case Folding* : Data *tweet* hasil *crawling* yang telah disimpan dipilih, lalu sistem akan merubah semua karakter huruf menjadi huruf kecil.
- Remove Punctuation : Setelah semua karakter huruf berubah menjadi kecil, maka dilakukan penghapusan URL, @mention, #hashtag, emoticon, serta angka.
- Tahapan *text-processing* selesai. *Flowchart* dari tahapan *text-processing* adalah sebagai berikut:



#### 4.5.2 Casefolding

*Case folding* dimana pada proses ini akan mengubah karakter dari huruf besar menjadi huruf kecil dengan tujuan jika terdapat kata yang sama namun penulisannya berbeda huruf kapital atau tidak maka tidak dianggap kata yang berbeda. Hasil dari case folding yaitu sebagai berikut

Hasil dari case folding yaitu sebagai berikut:

**Tabel 4.2 Case Folding**

Tweet Sebelum Case Folding	Tweet Sesudah Case Folding
Omnibus Law Pengaruhi IHSG, Simak Rekomendasi Saham Hari Ini <a href="https://t.co/V5qw44wm5t">https://t.co/V5qw44wm5t</a>	<u>o</u> mnibus <u>l</u> aw <u>p</u> engaruhi <u>i</u> hsg, <u>s</u> imak <u>r</u> ekomendasi <u>s</u> aham <u>h</u> ari <u>i</u> ni <a href="https://t.co/v5qw44wm5t">https://t.co/v5qw44wm5t</a>
Setelah ada versi ke-3 yg 1052 halaman, sekarang ada versi ke-4 yg 1035 halaman. Entah mana yg bener. Omnibus Law cipta kerja ini memang barang ghoib.	<u>s</u> etelah ada versi ke-3 yg 1052 halaman, sekarang ada versi ke-4 yg 1035 halaman. <u>e</u> ntah mana yg bener. <u>o</u> mnibus <u>l</u> aw <u>c</u> ipta <u>k</u> erja <u>i</u> n <u>m</u> emang <u>b</u> arang <u>g</u> hoib.
Jokowi Sebut Rakyat Termakan Hoax Omnibus Law, ASMaPi: Hoax Versi Penguasa atau Rakyat Jelata? <a href="https://t.co/zT8G9LDhxF">https://t.co/zT8G9LDhxF</a>	<u>j</u> okowi <u>s</u> ebut <u>r</u> akyat <u>t</u> ermakan <u>h</u> oax <u>o</u> mnibus <u>l</u> aw, <u>a</u> smapi: <u>h</u> oax <u>v</u> ersi <u>p</u> enguasa <u>a</u> tau <u>r</u> akyat <u>j</u> elata? <a href="https://t.co/zt8g9ldhxF">https://t.co/zt8g9ldhxF</a>
Di dalam UU sapu jagat ini beberapa pasal menyebutkan keistimewaan yang didapatkan oleh para warga asing. <a href="https://t.co/JGD17YGoXm">https://t.co/JGD17YGoXm</a>	<u>d</u> i dalam <u>u</u> u sapu jagat ini beberapa pasal menyebutkan keistimewaan yang didapatkan oleh para warga asing. <a href="https://t.co/jgd17ygoxm">https://t.co/jgd17ygoxm</a>
Kompas saja memuat tulisan kelemahan2 Omnibus Law. Mulai dari Kehutanan, pertanian, an hak2 buruh dll. <a href="https://t.co/0HVwxHjgwk">https://t.co/0HVwxHjgwk</a>	<u>k</u> ompas <u>s</u> aja <u>m</u> emuat <u>t</u> ulisan <u>k</u> elemahan2 <u>o</u> mnibus <u>l</u> aw. <u>m</u> ulai <u>d</u> ari <u>kehutanan, <u>pertanian, <u>a</u>n <u>h</u>ak2 <u>buruh <u>d</u>ll. <a href="https://t.co/0hvwxhjgwk">https://t.co/0hvwxhjgwk</a></u></u></u>
Covid in my country is joke> yg akhrnya muncul dan terbanyak skg ke 9 di dunia> dijadikan cuan oleh tikus sampah> rakyat suruh diem di rumah> diemdiem pemerintah mau sah in omnibus law #jegalomnibuslawsampaibatal	<u>c</u> ovid <u>i</u> n <u>m</u> y <u>c</u> ountry <u>i</u> s <u>j</u> oke> <u>y</u> g <u>a</u> khrnya <u>m</u> uncul <u>a</u> nd <u>t</u> erbanyak <u>s</u> kg <u>ke 9 <u>d</u>i <u>d</u>unia&gt; <u>d</u>ijadikan <u>c</u>uan <u>o</u>leh <u>t</u>ikus <u>s</u>ampah&gt; <u>r</u>akyat <u>s</u>uruh <u>d</u>iem <u>d</u>i <u>r</u>umah&gt; <u>d</u>iemdiem <u>pemerintah <u>mau</u> <u>s</u>ah <u>i</u>n <u>o</u>mnibus <u>l</u>aw <u>#j</u>egalomnibuslawsampaibatal</u></u>

Dapat dilihat pada Tabel 4.4 diatas merupakan hasil dari proses *case folding* dimana yang diberi garis bawah yaitu pada sebelumnya merupakan karakter huruf besar yang kemudian menjadi huruf kecil semua.

#### 4.5.3 Remove Punctuation

*Remove punctuation* merupakan proses yang dilakukan untuk membuang noise karakter berupa tanda baca, angka, markup/html/tag, spesial karakter (\$, %, &, stc) seperti yang terdapat dalam Tabel 4.5 berikut.

**Tabel 4.3 Remove Punctuation**

Tweet Sebelum Remove Punctuation	Tweet Sesudah Remove Punctuation
omnibus law pengaruhi ihsg <sub>2</sub> simak rekomendasi saham hari ini <a href="https://t.co/v5qw44wm5t">https://t.co/v5qw44wm5t</a>	omnibus law pengaruhi ihsg simak rekomendasi saham hari ini
setelah ada versi ke <sub>3</sub> yg <a href="#">1052</a> halaman <sub>2</sub> sekarang ada versi ke <sub>4</sub> yg <a href="#">1035</a> halaman <sub>2</sub> entah mana yg benar <sub>2</sub> omnibus law cipta kerja ini memang barang ghoib <sub>2</sub>	setelah ada versi ke yg halaman sekarang ada versi ke yg halaman entah mana yg benar omnibus law cipta kerja ini memang barang ghoib
jokowi sebut rakyat termakan hoax omnibus law <sub>2</sub> asmapi <sub>2</sub> hoax versi penguasa atau rakyat jelata <sub>2</sub> ? <a href="https://t.co/zt8g9ldhxf">https://t.co/zt8g9ldhxf</a>	jokowi sebut rakyat termakan hoax omnibus law asmapi hoax versi penguasa atau rakyat jelata
di dalam uu sapu jagat ini beberapa pasal menyebutkan keistimewaan yang didapatkan oleh para warga asing <sub>2</sub> <a href="https://t.co/jgd17ygoxm">https://t.co/jgd17ygoxm</a>	di dalam uu sapu jagat ini beberapa pasal menyebutkan keistimewaan yang didapatkan oleh para warga asing
kompas saja memuat tulisan kelemahan <sub>2</sub> omnibus law <sub>2</sub> mulai dari kehutanan <sub>2</sub> pertanian <sub>2</sub> an hak <sub>2</sub> buruh dll. <a href="https://t.co/0hvwxhjgwk">https://t.co/0hvwxhjgwk</a>	kompas saja memuat tulisan kelemahan omnibus law mulai dari kehutanan pertanian an hak buruh dll

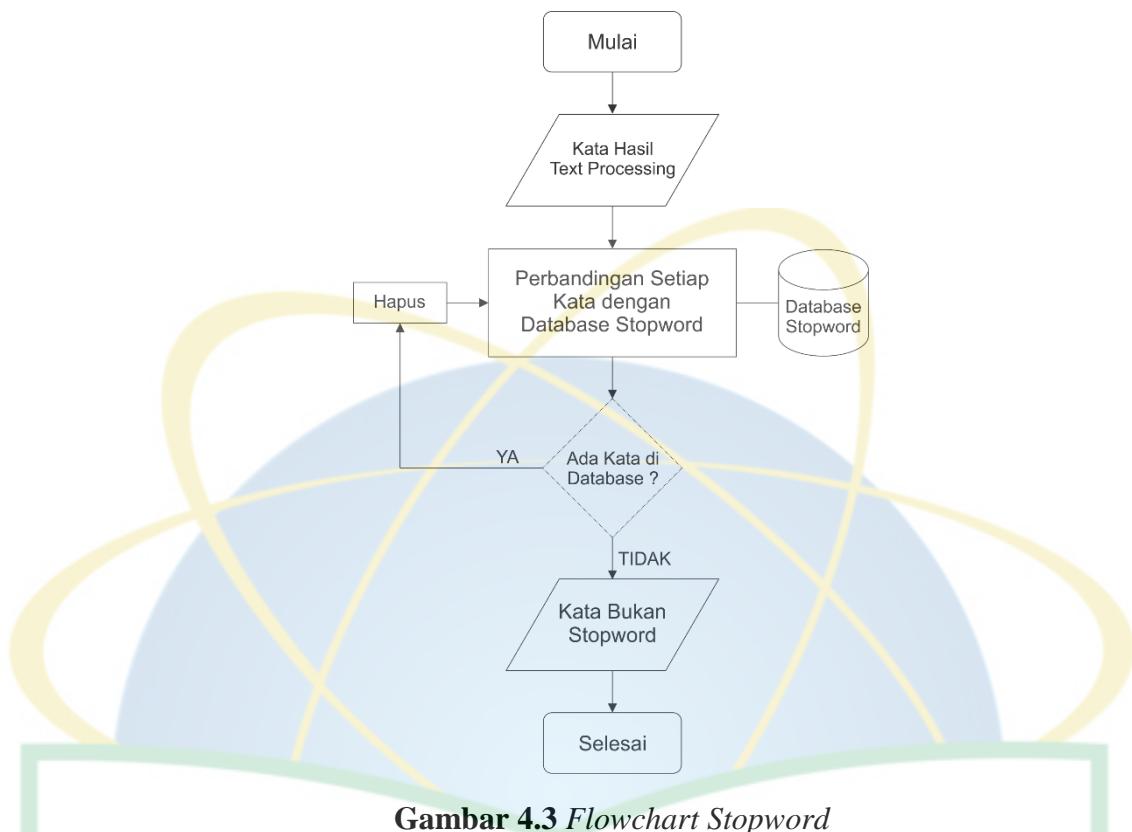
covid in my country is joke &gt; yg akhrnya muncul dan terbanyak skg ke 9 di dunia &gt; dijadikan cuan oleh tikus sampah &gt; rakyat suruh diem di rumah &gt; diemdiem pemerintah mau sah in omnibus law  #jegalomnibuslawsampaibatal	covid in my country is jokegt yg akhrnya muncul dan terbanyak skg ke di dunia gt dijadikan cuan oleh tikus sampah gt rakyat suruh diem di rumah gt diemdiem pemerintah mau sah in omnibus law
---	---

Dapat dilihat pada Tabel 4.5 diatas pada kolom sebelum yang diberi tanda underline dan berwarna yang kemudian dihilangkan sehingga pada kolom sesudah sudah tidak terdapat karakter-karakter tersebut.

#### 4.5.4 Stopword

*Stopword* merupakan proses penghilangan kata tidak penting pada deskripsi sehingga kata-kata yang tersisa di dalam deskripsi dianggap sebagai kata-kata penting atau *keywords*. Daftar kata tidak penting (*stoplist*) yang penulis gunakan dalam penelitian ini adalah database kamus stopword sastrawi. tahapan dari proses *stopword removal* sebagai berikut:

- Setelah melalui tahapan sebelumnya, hasilnya dibandingkan dengan kata-kata yang ada pada *database stopword*.
- Jika terdapat kata yang sama dengan kata pada *database stopword*, maka kata tersebut akan diseleksi atau dihapus oleh sistem.
- Jika terdapat kata yang berbeda / tidak sama dengan kata yang ada di *database stopword*, maka kata tersebut tidak akan diseleksi atau dibiarkan.
- Tahapan *stopword removal* selesai. *Flowchart* dari tahapan *stopword removal* adalah sebagai berikut:

**Gambar 4.3 Flowchart Stopword**

Hasil dari proses *stopword removal* sebagai berikut:

**Tabel 4.4 Stopword**

Tweet Sebelum Stopword Removal	Tweet Sesudah Stopword Removal
omnibus law pengaruhi ihsg simak rekomendasi saham hari <u>ini</u>	omnibus law pengaruhi ihsg simak rekomendasi saham hari
<u>setelah ada</u> versi <u>ke yg</u> halaman sekarang <u>ada</u> versi <u>ke yg</u> halaman entah mana <u>yg</u> bener omnibus law cipta kerja <u>ini</u> memang barang ghoib	versi halaman sekarang versi halaman entah mana bener omnibus law cipta kerja memang barang ghoib
jokowi sebut rakyat termakan hoax omnibus law asmapi hoax versi penguasa <u>atau</u> rakyat jelata	jokowi sebut rakyat termakan hoax omnibus law asmapi hoax versi penguasa rakyat jelata
<u>di dalam</u> uu sapu jagat <u>ini</u> beberapa pasal menyebutkan keistimewaan <u>yang</u> didapatkan <u>oleh para</u> warga asing	uu sapu jagat beberapa pasal menyebutkan keistimewaan didapatkan warga asing

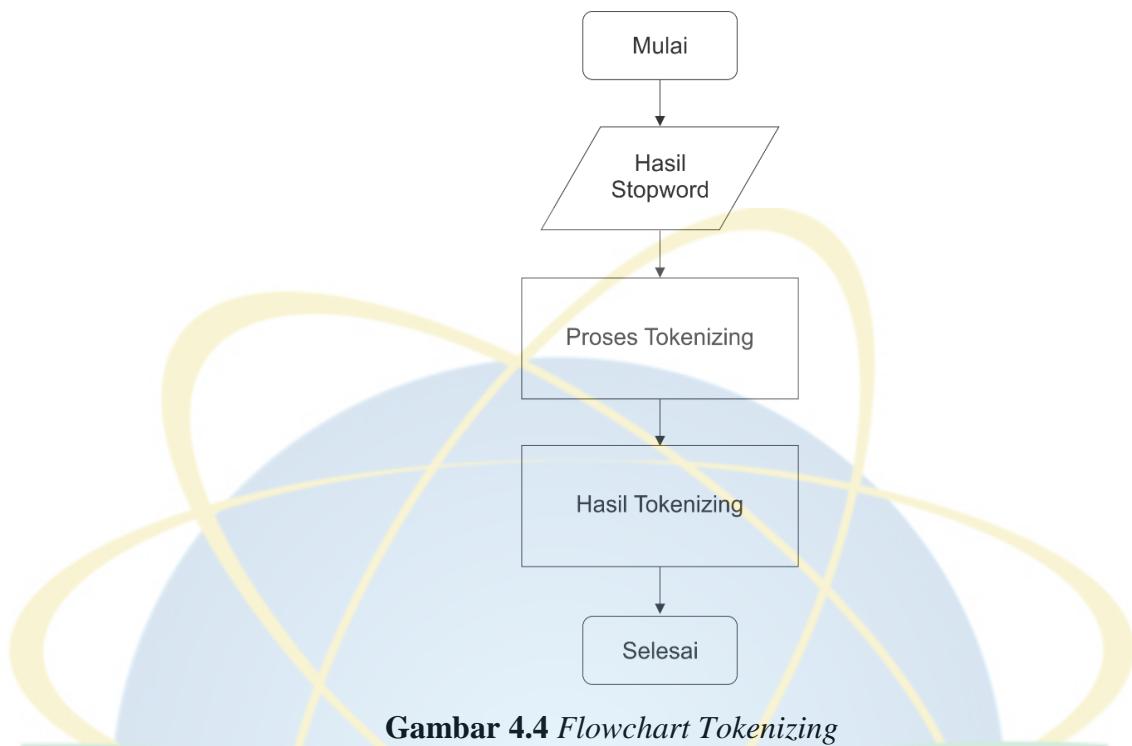
kompas <b>saja</b> memuat tulisan kelemahan omnibus law mulai <b>dari</b> kehutanan pertanian an hak buruh <b>dll</b>	kompas memuat tulisan kelemahan omnibus law mulai kehutanan pertanian an hak buruh
covid in my country is joke gt <b>yg</b> akhrnya muncul <b>dan</b> terbanyak skg <b>ke</b> <b>di</b> dunia gt dijadikan cuan <b>oleh</b> tikus sampah gt rakyat suruh diem <b>di</b> rumah gt diemdiem pemerintah mau sah in Omnibus Law	covid in my country is joke gt akhrnya muncul terbanyak skg dunia gt dijadikan cuan tikus sampah gt rakyat suruh diem rumah gt diemdiem pemerintah mau sah in Omnibus Law

Dapat dilihat pada Tabel 4.4 diatas pada kolom sebelum yang diberi tanda underline dan berwarna yang kemudian dihilangkan sehingga pada kolom sesudah tidak terdapat kata-kata tersebut.

#### 4.5.5 Tokenizing

*Tokenizing* merupakan membagi suatu teks atau pemotongan berdasarkan dari tiap kata yang menyusunnya agar memudahkan pada tahap selanjutnya. Adapun tahapan tokenizing sebagai berikut:

- Setelah melalui tahapan *stopword*, hasilnya dilakukan proses *tokenizing*
- Proses *tokenizing* membagi teks menjadi satu kata berdasarkan dari tiap kata yang menyusunnya.
- Hasil *tokenizing* disimpan untuk kemudian dilakukan proses selanjutnya.
- Tahapan *tokenizing* selesai. *Flowchart* dari tahapan *tokenizing* adalah sebagai berikut:



Hasi dari proses tokenizing adalah sebagai berikut

**Tabel 4.5** Tokenizing

Tweet Sebelum <i>Tokenizing</i>	Tweet Sesudah <i>Tokenizing</i>
Omnibus Law pengaruhi ihsg simak rekomendasi saham hari	'omnibus', 'law', 'pengaruhi', 'ihsg', 'simak', 'rekomendasi', 'saham', 'hari'
versi halaman sekarang versi halaman entah mana bener Omnibus Law cipta kerja memang barang ghoib	'versi', 'halaman', 'sekarang', 'versi', 'halaman', 'entah', 'mana', 'bener', 'omnibus', 'law', 'cipta', 'kerja', 'memang', 'barang', 'ghoib'
jokowi sebut rakyat termakan hoax Omnibus Law asmapi hoax versi penguasa rakyat jelata	'jokowi', 'sebut', 'rakyat', 'termakan', 'hoax', 'omnibus', 'law', 'asmapi', 'hoax', 'versi', 'penguasa', 'rakyat', 'jelata'
uu sapu jagat beberapa pasal menyebutkan keistimewaan didapatkan warga asing	'uu', 'sapu', 'jagat', 'beberapa', 'pasal', 'menyebutkan', 'keistimewaan', 'didapatkan', 'warga', 'asing'

	‘keistimewaan’, ‘didapatkan’, ‘warga’, ‘asing’
kompas memuat tulisan kelemahan Omnibus Law mulai kehutanan pertanian an hak buruh	‘kompas’, ‘memuat’, ‘tulisan’, ‘kelemahan’, ‘omnibus’, ‘law’, ‘mulai’, ‘kehutanan’, ‘pertanian’, ‘an’, ‘hak’ ‘buruh’
<i>Covid in my country is joke gt akhrnya muncul terbanyak skg dunia gt dijadikan cuan tikus sampah gt rakyat suruh diem rumah gt diemdiem pemerintah mau sah in omnibus law</i>	‘covid’, ‘in’, ‘my’, ‘country’, ‘is’, ‘joke’, ‘gt’, ‘akhrnya’, ‘muncul’, ‘terbanyak’, ‘skg’, ‘dunia’, ‘gt’, ‘dijadikan’, ‘cuan’, ‘tikus’, ‘sampah’, ‘gt’, ‘rakyat’, ‘suruh’, ‘diem’, ‘rumah’, ‘gt’, ‘diemdiem’, ‘pemerintah’, ‘mau’, ‘sah’, ‘in’ ‘omnibus’, ‘law’

#### 4.6 Feature Extraction

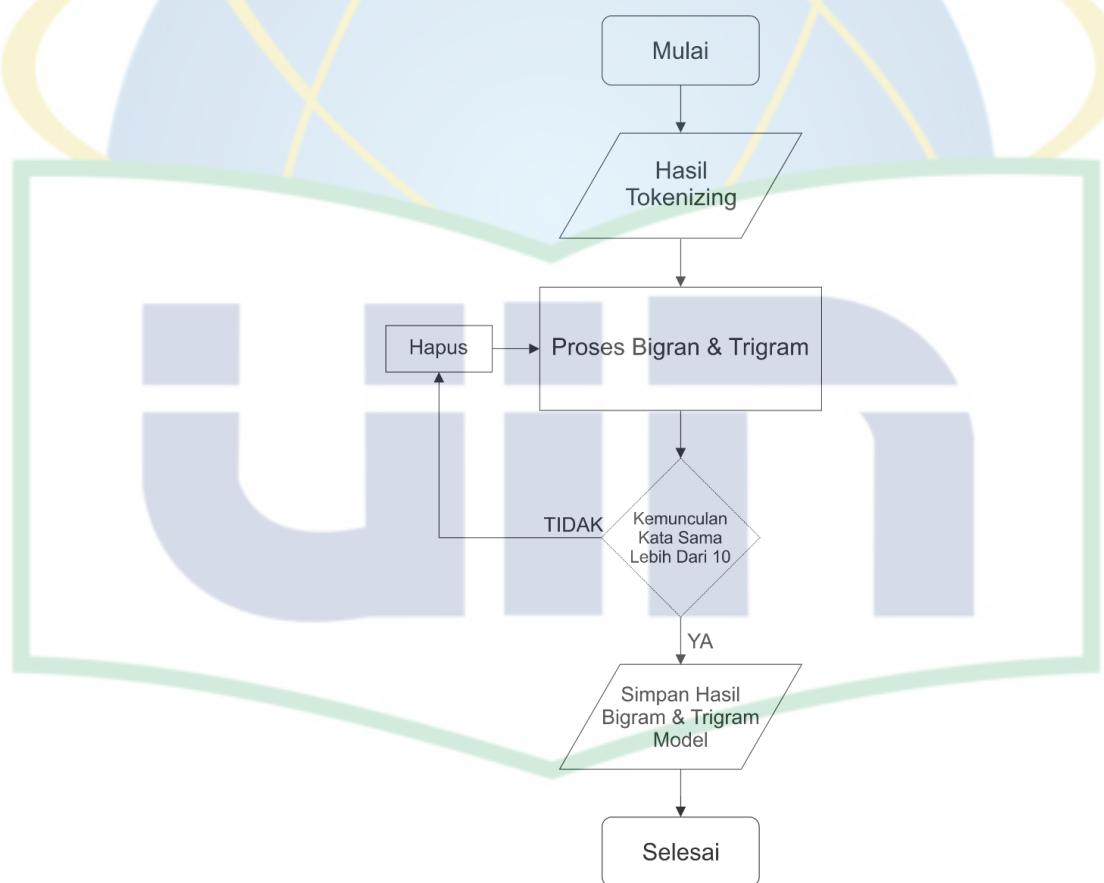
Setelah tahap *text preprocessing* dilakukan proses pengambilan informasi yang terpenting dari data yang diolah dan mempertinggi presisi pengolahan menggunakan bigram dan trigram model. Mengambil ciri – ciri yang unik dari data yang akan diolah dengan mengubah data yang berbentuk kata menjadi dalam bentuk numerik dengan menggunakan pembobotan TF-IDF, Dengan feature extraction dapat diperoleh data yang lebih spesifik sehingga data yang tidak relevant tidak akan digunakan pada proses selanjutnya.

##### 4.6.1 Bigram & Trigram Model

Pada tahap *bigram* dan *trigram* model bertujuan untuk mencari suatu makna kata yang disusun oleh dua kata dan tiga kata. Penulis menggunakan *bigram* dan *trigram* dalam penelitian ini untuk meminimalisir pemenggalan kata yang menghilangkan makna kata. Seperti kata “Omnibus Law” yang merupakan satu kesatuan kata namun terbentuk dari 2 kata. Penggunaan *bigram* dan *trigram* untuk mencari frequensi kemunculan kata yang sering muncul bergandengan dua kata

yaitu *bigram*, bergandengan tiga kata yaitu *trigram*. Adapun tahap pembuatan bigram dan trigram model adalah sebagai berikut:

- Setelah melalui tahapan tokenizing, hasilnya dibuat bigram dan trigram model.
- Proses bigram dan trigram : menyusun kata dengan dua kata yang bersebelahan dan tiga kata yang berdekatan.
- Kemunculan kata hasil proses bigram dan trigram apabila kemunculan kata lebih dari 10 maka hasil akan disimpan.
- Jika kemunculan kata TIDAK lebih dari 10 maka hasilnya akan dihapus.
- Tahapan pembuatan bigram dan trigram model selesai. Flowchart dari tahapan pembuatan bigram dan trigram model adalah sebagai berikut:



**Gambar 4.5 Flowchart Bigram & Trigram**

Hasil dari bigram dan trigram adalah sebagai berikut:

**Tabel 4.6 Bigram & Trigram**

Tweet Sebelum <i>Bigram</i> dan <i>Trigram</i>	Tweet Sesudah <i>Bigram</i>	Tweet Sesudah <i>Trigram</i>
‘omnibus’, ‘law’, ‘pengaruhi’, ‘ihsg’, ‘simak’, ‘rekомендации’, ‘saham’, ‘hari’	‘omnibus law’, ‘law pengaruhi’, ‘pengaruhi ihsg’, ‘ihsg simak’, ‘simak rekомендации’, ‘rekомендации saham’, ‘saham hari’	‘omnibus law pengaruhi’, ‘law pengaruhi ihsg’, ‘pengaruhi ihsg simak’, ‘ihsg simak rekомендации’, ‘simak rekомендации saham’, ‘rekомендации saham hari’
‘versi’, ‘halaman’, ‘sekarang’, ‘versi’, ‘halaman’, ‘entah’, ‘mana’, ‘bener’, ‘omnibus’, ‘law’, ‘cipta’, ‘kerja’, ‘memang’, ‘barang’, ‘ghoib’	‘versi halaman’, ‘halaman sekarang’, ‘sekarang versi’, ‘versi halaman’, ‘halaman entah’, ‘entah mana’, ‘mana bener’, ‘bener omnibus’, ‘omnibus law’, ‘law cipta’, ‘cipta kerja’, ‘kerja memang’, ‘memang barang’, ‘barang ghoib’	‘versi halaman sekarang’, ‘halaman sekarang versi’, ‘sekarang versi halaman’, ‘versi halaman entah’, ‘halaman entah mana’, ‘entah mana bener’, ‘mana bener omnibus’, ‘bener omnibus law’, ‘omnibus law cipta’, ‘law cipta kerja’, ‘cipta kerja memang’, ‘kerja memang barang’, ‘memang barang ghoib’
‘jokowi’, ‘sebut’, ‘rakyat’, ‘termakan’, ‘hoax’, ‘omnibus’, ‘law’, ‘asmapi’, ‘hoax’, ‘versi’,	‘jokowi sebut’, ‘sebut rakyat’, ‘rakyat termakan’, ‘termakan hoax’, ‘hoax omnibus’, ‘omnibus law’, ‘law asmapi’, ‘asmapi hoax’, ‘hoax versi’, ‘versi	‘jokowi sebut rakyat’, ‘sebut rakyat termakan’, ‘rakyat termakan hoax’, ‘termakan hoax omnibus’, ‘hoax omnibus law’, ‘omnibus law asmapi’,

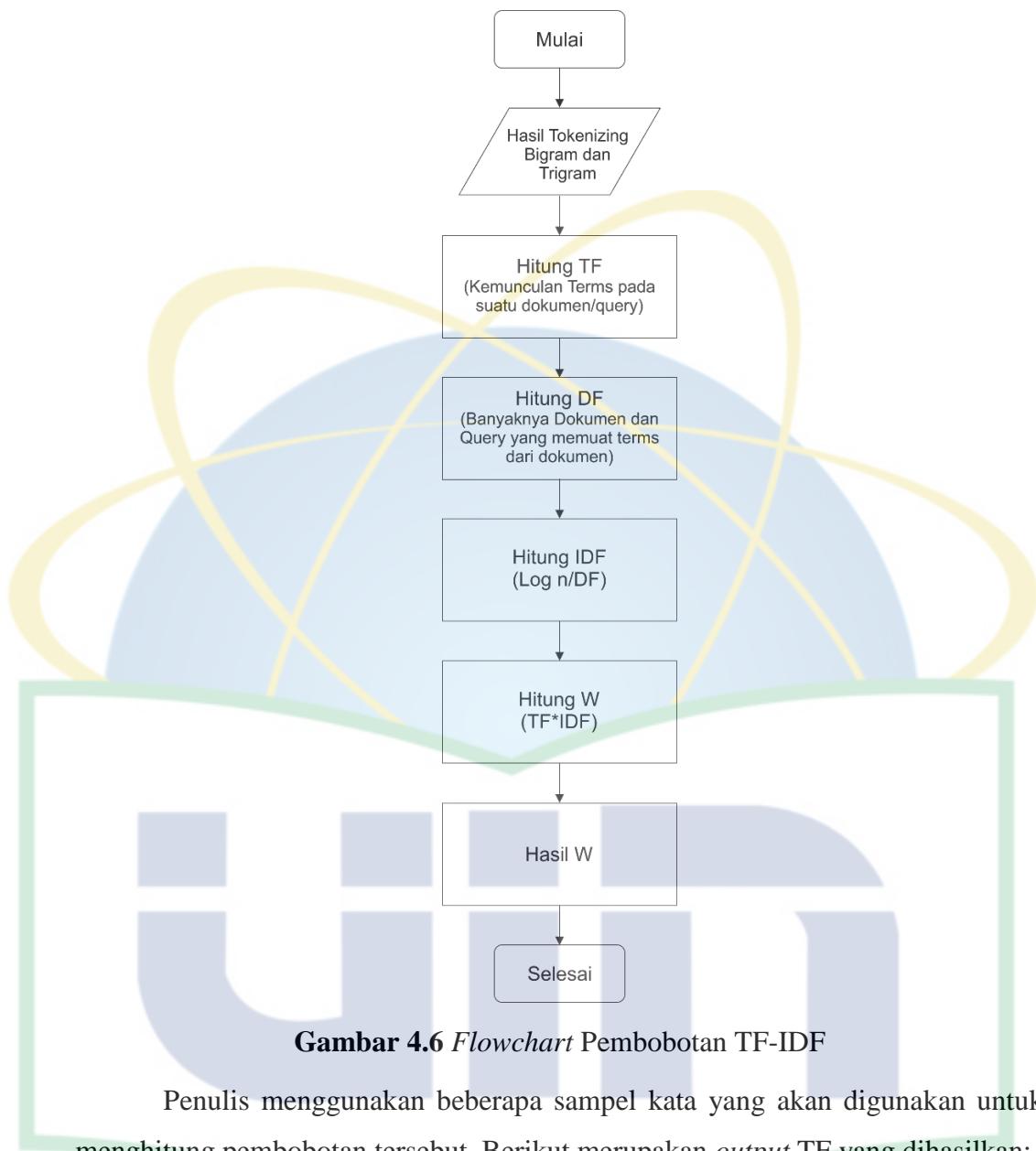
‘penguasa’, ‘rakyat’, ‘jelata’	penguasa’, ‘penguasa rakyat’, ‘rakyat jelata’	‘law asmapi hoax’, ‘asmapi hoax versi’, ‘hoax versi penguasa’, ‘versi penguasa rakyat’, ‘penguasa rakyat jelata’
‘uu’, ‘sapu’, ‘jagat’, ‘beberapa’, ‘pasal’, ‘menyebutkan’, ‘keistimewaan’, ‘didapatkan’, ‘warga’, ‘asing’	‘uu sapu’, ‘sapu jagat’, ‘jagat beberapa’, ‘beberapa pasal’, ‘pasal menyebutkan’, ‘menyebutkan keistimewaan’, ‘keistimewaan didapatkan’, ‘didapatkan warga’, ‘warga asing’	‘uu sapu jagat’, ‘sapu jagat beberapa’, ‘jagat beberapa pasal’, ‘beberapa pasal menyebutkan’, ‘pasal menyebutkan keistimewaan’, ‘menyebutkan keistimewaan didapatkan’, ‘keistimewaan didapatkan warga’, ‘didapatkan warga asing’
‘kompas’, ‘memuat’, ‘tulisan’, ‘kelemahan’, ‘omnibus’, ‘law’, ‘mulai’, ‘kehutanan’, ‘pertanian’, ‘an’, ‘hak’ ‘buruh’	‘kompas memuat’, ‘memuat tulisan’, ‘tulisan kelemahan’, ‘kelemahan omnibus’, ‘omnibus law’, ‘law mulai’, ‘mulai kehutanan’, ‘kehutanan pertanian’, ‘pertanian an’, ‘an hak’, ‘hak buruh’	‘kompas memuat tulisan’, ‘memuat tulisan kelemahan’, ‘tulisan kelemahan omnibus’, ‘kelemahan omnibus law’, ‘omnibus law mulai’, ‘law mulai kehutanan’, ‘mulai kehutanan pertanian’, ‘kehutanan pertanian an’, ‘pertanian an hak’, ‘an hak buruh’
‘covid’, ‘in’, ‘my’, ‘country’, ‘is’,	‘covid in’, ‘in my’, ‘my country’, ‘country is’, ‘is	‘covid in my’, ‘in my country’, ‘my country is’,

<p>‘joke’, ‘gt’,      ‘akhrnya’, ‘muncul’,      ‘terbanyak’, ‘skg’,      ‘dunia’, ‘gt’,      ‘dihadikan’, ‘cuan’,      ‘tikus’, ‘sampah’,      ‘gt’, ‘rakyat’,      ‘suruh’, ‘diem’,      ‘rumah’, ‘gt’,      ‘diemdiem’,      ‘pemerintah’, ‘mau’,      ‘sah’, ‘in’      ‘omnibus’, ‘law’</p>	<p>joke’, ‘joke gt’, ‘gt      akhrnya’, ‘akhrnya      muncul’, ‘muncul      terbanyak’, ‘terbanyak      skg’, ‘skg dunia’, ‘dunia      gt’, ‘gt dijadikan’,      ‘dihadikan cuan’, ‘cuan      tikus’, ‘tikus sampah’,      ‘sampah gt’, ‘gt rakyat’,      ‘rakyat suruh’, ‘suruh      diem’, ‘diem rumah’,      ‘rumah gt’, ‘gt diemdiem’,      ‘diemdiem pemerintah’,      ‘pemerintah mau’, ‘mau      sah’, ‘sah in’, ‘in omnibus’      ‘omnibus law’</p>	<p>‘country is joke’, ‘is joke      gt’, ‘joke gt akhrnya’, ‘gt      akhrnya muncul’,      ‘akhrnya muncul      terbanyak’, ‘muncul      terbanyak skg’, ‘terbanyak      skg dunia’, ‘skg dunia gt’,      ‘dunia gt dijadikan’, ‘gt      dijadikan cuan’,      ‘dihadikan cuan tikus’,      ‘cuan tikus sampah’,      ‘tikus sampah gt’,      ‘sampah gt rakyat’, ‘gt      rakyat suruh’, ‘rakyat      suruh diem’, ‘suruh diem      rumah’ ‘diem rumah gt’,      ‘rumah gt diemdiem’, ‘gt      diemdiem pemerintah’,      ‘diemdiem pemerintah      mau’, ‘pemerintah mau      sah’, ‘mau sah in’, ‘sah in      omnibus’, ‘in omnibus      law’</p>
--	---	--

#### 4.6.2 Pembobotan *Term Frequency-Invers Document Frequency* (TF-IDF)

Pada tahap ini merupakan tahap pembobotan yang dimana setelah dilakukan *preprocessing* pada sebelumnya akan dilakukan pengubahan data yang berbentuk kata menjadi dalam bentuk numerik dengan menggunakan pembobotan TF-IDF. Pembobotan TF-IDF adalah gabungan dari metode *Term Frequency* (TF) dengan metode *Inverse Document Frequency* (IDF). Adapun tahap pembobotan TF-TDF adalah sebagai berikut:

- Setelah melalui tahapan tokenizing dari bigram dan trigram model, hasilnya dilakukan pembobotan TF-IDF
- Menghitung *Term Frequency*: menghitung kemunculan dari sebuah kata pada sebuah dokumen
- Menghitung *Document Frequency*: menghitung frekuensi dokumen yang mengandung kata tersebut
- Menghitung *Inverse Document Frequency*: perhitungan dari bagaimana *term* didistribusikan secara luas pada dokumen.
- Rumus perhitungan  $IDF = \log n/DF$ .  $n$  adalah jumlah dokumen,  $DF$  adalah jumlah dokumen yang mengandung *term*.
- Menghitung *Weight*: menghitung bobot *term* terhadap dokumen
- Rumus perhitungan  $W = TF \times IDF$ .
- Hasil Pembobotan TF-IDF
- Tahapan pembobotan TF-IDF selesai. Flowchart dari tahapan pembobotan TF-IDF adalah sebagai berikut:



Penulis menggunakan beberapa sampel kata yang akan digunakan untuk menghitung pembobotan tersebut. Berikut merupakan *output* TF yang dihasilkan:

**Tabel 4.7 Output TF**

Dokumen	aab	aamiin	aasteno	....	bahas	bahasa	...
1	0	0	0	...	0	0	...
2	0	0	0	...	0	0	...
3	0	0	0	...	1	0	...
...	...	...	...	...	...	...	...
...	...	...	...	...	...	...	...

3464	0	0	0	...	0	0	...
3465	0	0	0	...	0	0	...
3466	0	0	0	...	0	0	...

Dapat dilihat pada Tabel 4.9 diatas merupakan hasil yang diperoleh *output* TF yang merupakan frekuensi kemunculan *term* (t) pada dokumen (D). Jika nilainya 0 maka kemunculan *term* (t) pada dokumen (D) tidak ada, namun jika bernilai 1 maka kemunculan *term* (t) pada dokumen (D) tersebut terdapat 1. Setelah didapatkan hasil TF kemudian dapat mencari hasil dari TF-IDF. Berikut ini merupakan output dari perhitungan TF-IDF.

**Tabel 4.8** Perhitungan TF-IDF

Dokumen	aab	aamiin	aasteno	....	bahas	bahasa	...
1	0	0	0	...	0	0	...
2	0	0	0	...	0	0	...
3	0	0	0	...	5.5398	0	...
...	...	...	...	...	...	...	...
...	...	...	...	...	...	...	...
3464	0	0	0	...	0	0	...
3465	0	0	0	...	0	0	...
3466	0	0	0	...	0	0	...

Dapat dilihat pada Tabel 4.10 diatas merupakan hasil yang diperoleh *output* perhitungan TF-IDF yang didapatkan dari perkalian nilai TF dengan IDF. Sebelum diperoleh nilai TF-IDF seperti yang terdapat dalam Tabel 4.10 diatas, maka penulis harus mendapatkan terlebih dahulu nilai TF dan IDF.

Peneliti menggunakan contoh untuk perhitungan TF-IDF secara manual yaitu kata “bahas” pada dokumen ke-3. Nilai TF sudah didapatkan pada Tabel 4.9 diatas, kemudian untuk menghitung nilai IDF diperlukan nilai DF yang merupakan jumlah dokumen yang mengandung *term* (t) atau terdapat berapa banyak dokumen

yang mengandung kata "bahas" yaitu sebanyak 37. Selanjutnya untuk menghitung nilai IDF, dimana IDF merupakan *inverse* dari DF. Dengan menggunakan persamaan untuk perhitungan manual IDF. Jumlah dokumen (D) pada penelitian ini sebanyak 3466 dokumen.

$$\text{IDF}_{\text{bahas}} = \ln \frac{3466}{37} = 4.539838557631320$$

Setelah didapatkan nilai TF dan IDF, kemudian untuk menghitung nilai TF-IDF yaitu dengan melakukan perkalian TF dengan IDF.

$$W = 1 * \left( \ln \frac{3466}{37} + 1 \right)$$

$$W = 1 * (4.539838557631320 + 1)$$

$$W = 1 * (5.539838557631320)$$

$$W = 5.539838557631320$$

$$W \approx 5.5398$$

Hasil perhitungan manual dari TF-IDF kata "bahas" pada dokumen ke-3 (D3) yang telah didapatkan kemudian dirangkum sebagai berikut.

Tabel 4.11 Contoh Hasil dari Perhitungan TF-IDF "bahas"

**Tabel 4.9** Contoh Hasil TF-IDF

Term (t)	TF	DF	IDF	TF-IDF
	D3			D3
bahas	1	37	4.539838557631320	5.539838557631320

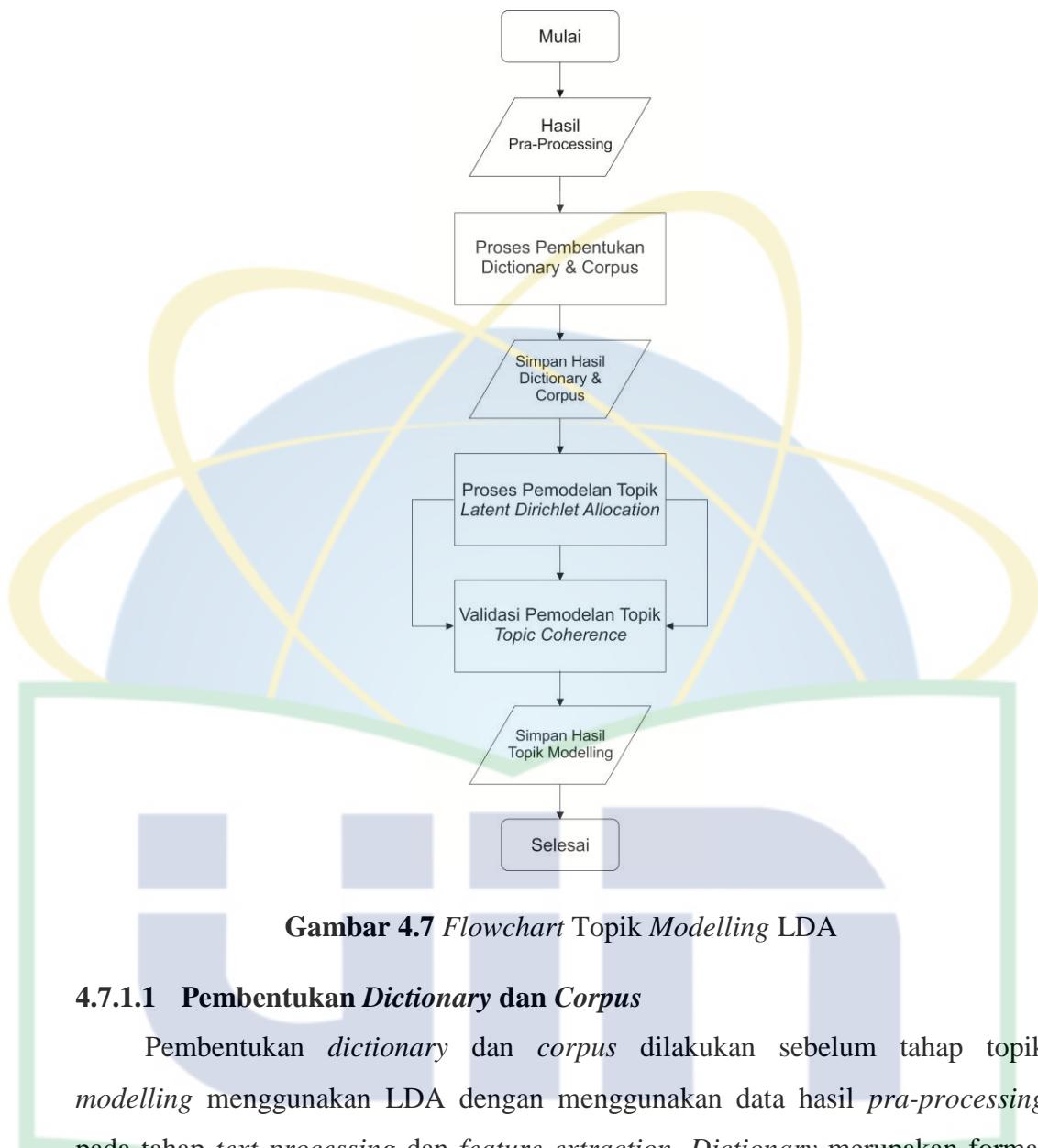
## 4.7 Analysis

Pada tahap *analysis* penulis melakukan pemodelan topik dengan metode *Latent Dirichlet Allocation* (LDA). Tahapan pemodelan topik dengan LDA merupakan tahapan yang dilakukan untuk membentuk model topik. Terdapat dua hal penting yang menjadi perhatian dalam tahap ini, yaitu alur pemodelan topik dengan LDA dan eksperimen pemodelan topik.

### 4.7.1 Alur Pemodelan Topik dengan *Latent Dirichlet Allocation* (LDA)

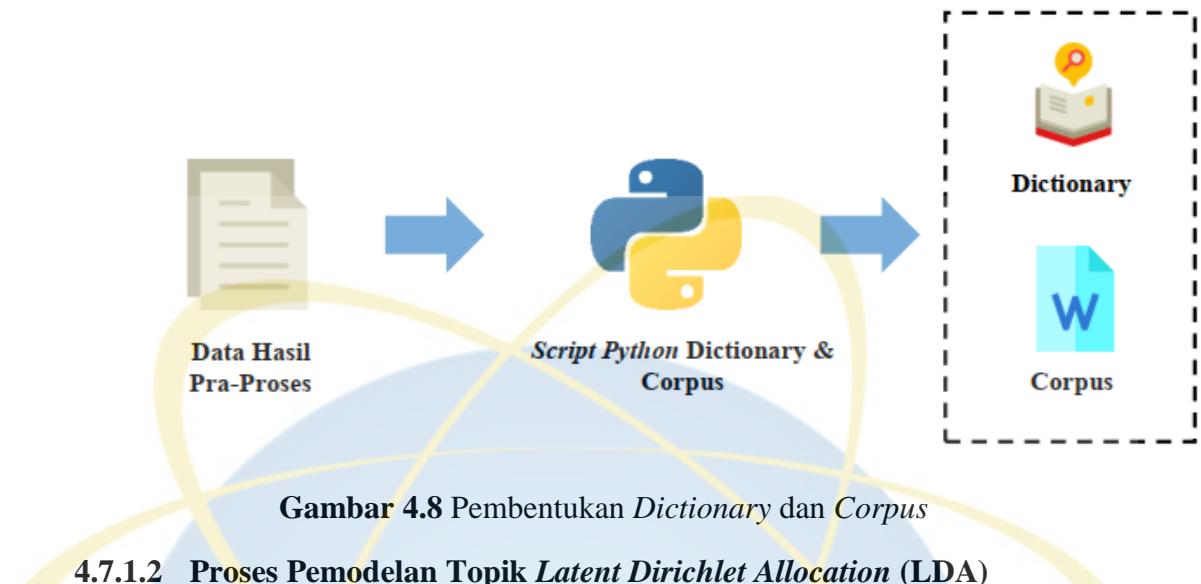
Adapun tahap alur pemodelan topik menggunakan LDA adalah sebagai berikut:

- Setelah melakukan tahap *pra-processing*, hasilnya digunakan untuk membuat pemodelan topik dengan LDA
- Sebelum pemodelan topik, terlebih dahulu dilakukan pembentukan *dictionary* dan *corpus*
- *Dictionary* merupakan format data yang mengandung himpunan kata unik dengan nomor indeks masing-masing
- *corpus* merupakan format data berbentuk *bag-of-word reference* yang akan digunakan untuk pembentukan model
- Proses pemodelan topik LDA dilakukan dengan menentukan jumlah topik yang menjadi kelompok klasterisasi kata dan penentuan jumlah passes dalam topic modeling yang merupakan jumlah iterasi proses pembelajaran dari training model.
- Validasi pemodelan topic dengan *topic coherence* bertujuan untuk memastikan model yang dibentuk dari proses topik modeling pada dokumen yang dihasilkan memiliki nilai probabilitas tertinggi.
- Hasil pemodelan topik LDA dengan jumlah topik yang sesuai berdasarkan nilai *topic coherence* tiap model.



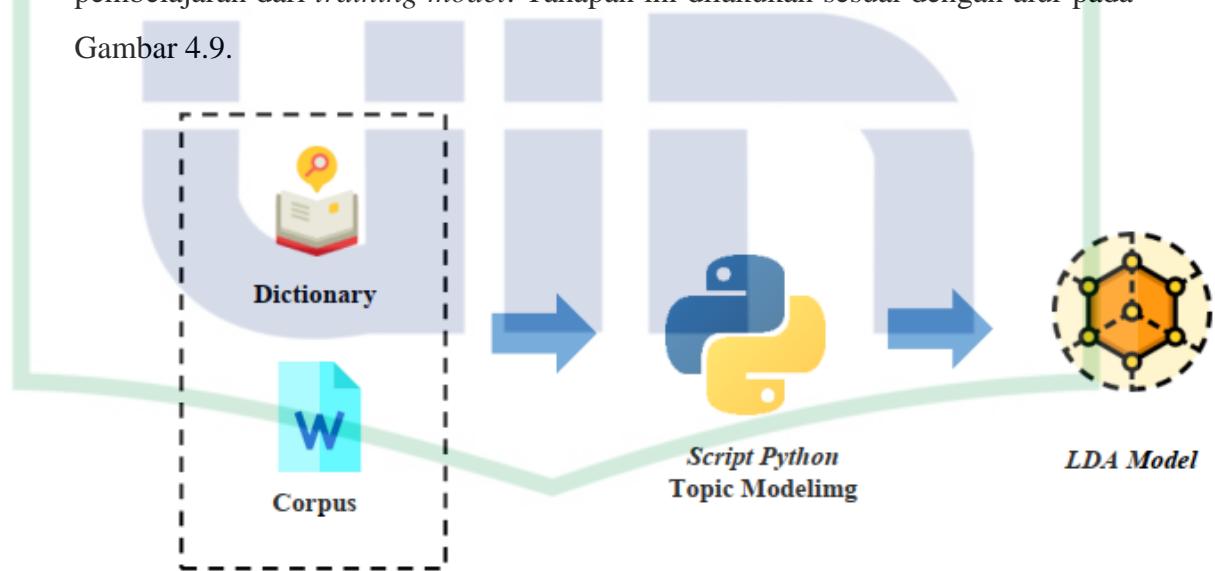
#### 4.7.1.1 Pembentukan *Dictionary* dan *Corpus*

Pembentukan *dictionary* dan *corpus* dilakukan sebelum tahap topik *modelling* menggunakan LDA dengan menggunakan data hasil *pra-processing* pada tahap *text processing* dan *feature extraction*. *Dictionary* merupakan format data yang mengandung himpunan kata unik dengan nomor indeks masing-masing. Sedangkan *corpus* merupakan format data berbentuk *bag-of-word reference* yang akan digunakan untuk pembentukan model. Tahapan ini dilakukan sesuai dengan alur pada Gambar 4.8.



#### 4.7.1.2 Proses Pemodelan Topik *Latent Dirichlet Allocation* (LDA)

Pemodelan topik metode *Latent Dirichlet Allocation* dilakukan dengan menggunakan *library ldamodel* dari *gensim*. Pembentukan model dilakukan dengan menentukan jumlah topik yang menjadi kelompok klasterisasi kata dan penentuan jumlah *passes* dalam *topic modelling* yang merupakan jumlah iterasi proses pembelajaran dari *training model*. Tahapan ini dilakukan sesuai dengan alur pada Gambar 4.9.



**Gambar 4.9 Pemodelan Topik *Latent Dirichlet Allocation* (LDA)**

#### 4.7.1.3 Validasi Pemodelan Topik (*Topic Coherence*)

Tahapan ini bertujuan untuk memastikan model yang dibentuk dari proses *topic modelling* pada dokumen yang dihasilkan memiliki nilai probabilitas

tertinggi, baik luaran berupa topik maupun kata-kata yang menyusun setiap topiknya. Dalam menentukan hasil pemodelan dapat dilakukan dengan melihat pada visualisasi dari grafik *coherence score*. Tahap ini dilakukan untuk menguji kemudahan topik model dalam kemampuannya untuk diinterpretasi oleh manusia. *Coherence score* merupakan suatu ukuran yang akan digunakan untuk mengevaluasi *topic modelling*, dimana jika *coherence score* topik yang tinggi maka model yang dihasilkan tersebut yang baik.

#### 4.7.2 Eksperimen Pemodelan Topik

Tahapan eksperimen pemodelan topik dengan *Latent Dirichlet Allocation* (LDA) merupakan tahapan yang dilakukan untuk membentuk model topik terbaik dengan melakukan eksperimen pada input parameter. Tahapan eksperimen pemodelan topik dilakukan dengan dua tahap perbandingan yaitu penentuan passes atau jumlah iterasi dan jumlah topik dan perbandingan hasil pemodelan topik menggunakan *Latent Dirichlet Allocation* (LDA) dengan *Latent Semantic Analysis* (LSA). Kedua eksperimen yang dilakukan akan dijelaskan masing-masing pada sub bab sebagai berikut.

##### 4.7.2.1 Eksperimen Penentuan Jumlah Topik

Salah satu eksperimen yang akan dilakukan untuk menentukan model topik terbaik yaitu dengan eksperimen penentuan jumlah topik. Eksperimen jumlah topik merupakan tahap yang penting yang akan menghasilkan model dengan nilai *coherence score* yang tinggi, semakin tinggi nilai *coherence score*, menunjukkan akurasi model lebih baik. Penentuan jumlah topik dilakukan dengan melakukan eksperimen pada nilai jumlah topik. Penentuan jumlah topik diawali dengan memberikan nilai mula-mula yaitu sebanyak 3 kali, yaitu 10, 20, 30 topik. Berdasarkan eksperimen jumlah topik yang telah dilakukan, nilai *coherence score* yang muncul dianalisis tren nilainya secara visual dan dilakukan penghitungan standar deviasinya. Sehingga pada akhirnya jumlah topik yang dipilih adalah jumlah topik yang memiliki nilai rata-rata paling tinggi dengan standar deviasi minimum. Perbandingan nilai *coherence score* dapat dilihat pada Tabel 4.10.

**Tabel 4.10** Eksperimen Penentuan Jumlah Topik

10 Topik		20 Topik		30 Topik	
Num Topics	Coherence Score	Num Topics	Coherence Score	Num Topics	Coherence Score
1	0.4869	1	0.4495	1	0.4869
2	0.3353	2	0.4757	2	0.3015
3	0.3861	3	0.3591	3	0.4736
4	0.443	4	0.5049	4	0.3652
5	0.5418	5	0.5644	5	0.5261
6	0.4164	6	0.4272	6	0.517
7	0.434	7	0.5246	7	0.4521
8	0.4684	8	0.441	8	0.4624
9	0.4708	9	0.4833	9	0.4735
10	0.4845	10	0.4216	10	0.4744
		11	0.4778	11	0.4786
		12	0.4622	12	0.4907
		13	0.4668	13	0.4695
		14	0.4854	14	0.5064
		15	0.5147	15	0.5165
		16	0.4793	16	0.4896
		17	0.4876	17	0.5143
		18	0.485	18	0.5034
		19	0.4839	19	0.5
		20	0.5029	20	0.513
				21	0.4913
				22	0.5366
				23	0.5156

	24	0.5195
	25	0.4972
	26	0.5255
	27	0.525
	28	0.5105
	29	0.5118
	30	0.5356

Berdasarkan eksperimen dari 3 parameter iterasi yang telah dilakukan didapatkan hasil pada masing-masing iterasi topik. Iterasi 10 topik dihasilkan jumlah topik terbaik dengan nilai *coherence score* tertinggi sebesar 0.5418 yaitu pada *num topics* 5. Selanjutnya pada pengujian 20 topik didapatkan hasil nilai dengan nilai *coherence score* tertinggi sebesar 0.5644 pada *num topics* 5. Pengujian terakhir yakni pada 30 topik didapatkan hasil nilai *coherence score* tertinggi sebesar 0.5366 pada *num topics* 22. Nilai *coherence score* yang dihasilkan dari setiap iterasi topik kemudian dirata-rata sesuai dengan *num topics* pada masing-masing iterasi. Nilai rata-rata tersebut digunakan sebagai penentuan pemodelan topik untuk mengetahui pembicaraan warganet *twitter* tentang *Omnibus Law* dengan *Latent Dirichlet Allocation* (LDA). berdasarkan hasil nilai rata-rata yang dilakukan, penulis memilih *num topics* 5 yang digunakan sebagai pemodelan topik untuk mengetahui pembicaraan warganet *twitter* tentang *Omnibus Law* dengan *Latent Dirichlet Allocation* (LDA) karena pada *num topics* 5 memiliki rata-rata paling tinggi dibanding dengan nilai rata-rata *num topics* lain dengan standar deviasi minimum 0.5441.

#### 4.7.2.2 Eksperimen perbandingan hasil pemodelan topik menggunakan *Latent Dirichlet Allocation* (LDA) dengan *Latent Semantic Analysis* (LSA)

Eksperimen pemodelan topik yang dilakukan berikutnya adalah eksperimen dengan melakukan perbandingan hasil pemodelan topik menggunakan *Latent Dirichlet Allocation* (LDA) dengan *Latent Semantic Analysis* (LSA). Tahap

eksperimen ini dilakukan dengan menentukan nilai *coherence score* pada masing-masing iterasi topik dari LDA dan LSA. Setelah dilakukan perhitungan nilai *coherence score* didapatkan hasil sebagai berikut.

**Tabel 4.11** Eksperimen Perbandingan LDA dengan LSA

10 Topik			
LDA		LSA	
Num Topics	Coherence Score	Num Topics	Coherence Score
1	0.4869	1	0.4102
2	0.3353	2	0.4711
3	0.3861	3	0.5076
4	0.443	4	0.3693
5	0.5418	5	0.3397
6	0.4164	6	0.3619
7	0.434	7	0.412
8	0.4684	8	0.4127
9	0.4708	9	0.3424
10	0.4845	10	0.4982
20 Topik			
LDA		LSA	
Num Topics	Coherence Score	Num Topics	Coherence Score
1	0.4495	1	0.3991
2	0.4757	2	0.4045
3	0.3591	3	0.4768
4	0.5049	4	0.5049
5	0.5644	5	0.507
6	0.4272	6	0.4083
7	0.5246	7	0.3412
8	0.441	8	0.4184
9	0.4833	9	0.4848

10	0.4216	10	0.397
11	0.4778	11	0.4164
12	0.4622	12	0.4794
13	0.4668	13	0.3385
14	0.4854	14	0.434
15	0.5147	15	0.4002
16	0.4793	16	0.4113
17	0.4876	17	0.4888
18	0.485	18	0.3837
19	0.4839	19	0.4639
20	0.5029	20	0.4249
30 Topik			
LDA		LSA	
Num Topics	Coherence Score	Num Topics	Coherence Score
1	0.4869	1	0.4301
2	0.3015	2	0.411
3	0.4736	3	0.4363
4	0.3652	4	0.3405
5	0.5261	5	0.3614
6	0.517	6	0.3702
7	0.4521	7	0.3796
8	0.4624	8	0.4761
9	0.4735	9	0.4231
10	0.4744	10	0.3596
11	0.4786	11	0.3974
12	0.4907	12	0.4683
13	0.4695	13	0.4144
14	0.5064	14	0.3395
15	0.5165	15	0.41
16	0.4896	16	0.441

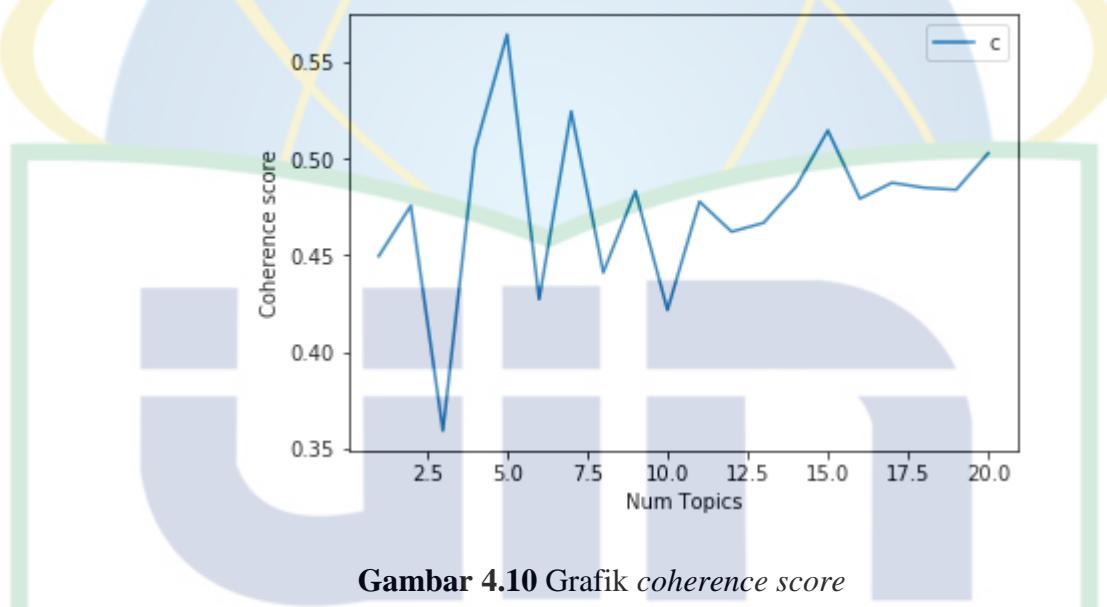
17	0.5143	17	0.4259
18	0.5034	18	0.4458
19	0.5	19	0.4118
20	0.513	20	0.4659
21	0.4913	21	0.4307
22	0.5366	22	0.4395
23	0.5156	23	0.422
24	0.5195	24	0.427
25	0.4972	25	0.4393
26	0.5255	26	0.4042
27	0.525	27	0.4039
28	0.5105	28	0.5099
29	0.5118	29	0.4991
30	0.5356	30	0.4021

Berdasarkan hasil dari Tabel 4.11 di atas, didapatkan nilai *coherence score* dari masing-masing iterasi topik LDA dan LSA. Nilai *coherence score* pada tabel LDA dan LSA memiliki perbedaan yaitu, tabel nilai *coherence score* tabel LDA memiliki nilai rata-rata 0.5 yang menunjukkan bahwa nilai tersebut tergolong sedang untuk nilai *coherence score*. Hal ini sesuai dengan teori penggolongan nilai *coherence score* yang menyebutkan bahwa penggolongan nilai *coherence score* terbagi menjadi 7 golongan yaitu, 0.3 is bad, 0.4 is low, 0.5 is okay, 0.6 might be as good as it is going to get, 0.7 is nice, 0.8 is unlikely, dan nilai 0.9 is probably wrong (Schervish, Seidenfeld, & Kadane, 2010). Sedangkan pada tabel LSA rata-rata menunjukkan nilai *coherence score* sebesar 0.4 yang tergolong rendah.

Berdasarkan hasil nilai *coherence score* tersebut membuktikan bahwa pemodelan topik menggunakan LDA merupakan pemodelan terbaik jika dibanding dengan LSA. Hal ini sesuai dengan teori yang disebutkan oleh Blei, et al. 2003 yang menyebutkan bahwa *Latent Dirichlet Allocation* merupakan algoritma pemodelan topik yang stabil untuk mengolah data dalam jumlah besar.

#### 4.7.3 Hasil *Topic Modelling* dengan *Latent Dirichlet Allocation* (LDA)

Dalam menentukan hasil pemodelan dapat dilakukan dengan melihat pada visualisasi dari grafik *coherence score*. *Coherence score* merupakan suatu ukuran yang akan digunakan untuk mengevaluasi *topic modelling*, dimana jika *coherence score* topik yang tinggi maka model yang dihasilkan tersebut yang baik. Grafik dari *coherence score* yang dihasilkan terdapat naik dan turun, penulis telah melakukan perbandingan *coherence score* pada eksperimen pemodelan topik dengan menentukan penentuan jumlah topik dimana topik dimulai dari 1 sampai 10, 1 sampai 20 dan 1 sampai 30. Dari hasil eksperimen tersebut jumlah topik dalam pemodelan topik dengan nilai *coherence score* terbaik sebanyak 5 topik.



**Gambar 4.10** Grafik *coherence score*

Berdasarkan Gambar 4.14 diatas didapatkan informasi bahwa pada grafik *coherence score* diatas memiliki nilai yang naik turun berdasarkan *num topics* nya. Hasil *num topics* dengan grafik nilai tertinggi terdapat pada jumlah topik sebanyak 5 topik yang memiliki nilai *coherence* terbesar yaitu sebesar 0.5644, dengan begitu jumlah topik tersebut yang akan menjadi acuan untuk membuat model selanjutnya. pada tabel berikut ini merupakan *coherence score* yang dihasilkan:

**Tabel 4.12 Coherence Score**

<i>Num Topic</i>	<i>Coherence Score</i>	<i>Num Topic</i>	<i>Coherence Score</i>
1	0.4495	11	0.4778
2	0.4757	12	0.4633
3	0.3591	13	0.4668
4	0.5049	14	0.4854
5	0.5644	15	0.5147
6	0.4272	16	0.4793
7	0.5246	17	0.4876
8	0.441	18	0.485
9	0.4833	19	0.4839
10	0.4216	20	0.5029

Setelah didapatkan jumlah topiknya berdasarkan grafik *coherence score*, kemudian akan didapatkan model LDA berdasarkan banyak topiknya yaitu 5 dengan jumlah kata yang ditampilkan dalam model 10 kata yang memiliki bobot masing-masing dari tiap kata tersebut.

**Tabel 4.13 Hasil Pemodelan Topik LDA**

Model LDA Topik 1
'0.018*"cipta_kerja" + 0.007*"kepala_daerah" + 0.007*"demo_to_lak" + 0.006*"kerja" + 0.006*"cipta" + 0.006*"tolak" + 0.006*"demo" + 0.005*"uu_cipta" + 0.004*"catatan_perjalanan" + 0.004*"februari_oktober"'
Model LDA Topik 2
'0.024*"cekal_korona" + 0.013*"korona" + 0.013*"cekal" + 0.006*"ikut_demo" + "0.006*"presiden_jokowi" + 0.005*"demo" + 0.005*"pemerintah" + 0.004*"cipta_kerja" + 0.004*"sama_sekali" + 0.004*"februari"'
Model LDA Topik 3
'0.015*"uu_cipta" + 0.009*"uu_ciptaker" + 0.008*"cipta_kerja" + 0.008*"kawasan_hutan" + 0.006*"kerja" + 0.006*"cipta" + 0.005*"hoax" + 0.005*"ciptaker" + 0.005*"undang_undang" + 0.004*"baca"'
Model LDA Topik 4
'0.018*"melly_goeslaw" + 0.009*"cipta_kerja" + 0.009*"warga_a_sing" + '0.008*"sederet_keistimewaan" + 0.007*"uu_cipta" + 0.007*"melly" + 0.006*"goeslaw" + 0.006*"asing" + 0.005*"keistimewaan" + 0.005*"ruu_cipta"'

Model LDA Topik 5
'0.013*"gas_air" + 0.012*"ngaco_senen" + 0.012*"jongkok_lalu" + 0.007*"surabaya" + 0.007*"polisi" + 0.007*"demo" + 0.007*"m ata" + 0.006*"gas" + 0.006*"senen" + 0.006*"air"'

Kemudian setelah diperoleh model LDA maka model tersebut dapat dilihat dengan visualisasi PyLDAvis dan keterkaitan antar kata yang dihasilkan. Pada panel sisi kiri merupakan pemetaan jarak dari antar topik (*intertopic distance map*) via *multidimensional scaling* yang terdapat juga cluster topik yang berbentuk lingkaran dengan nomor tertentu pada setiap cluster topik. Sedangkan pada panel sisi kanan terdapat 30 buah terminologi yang paling relevan untuk topik tertentu.

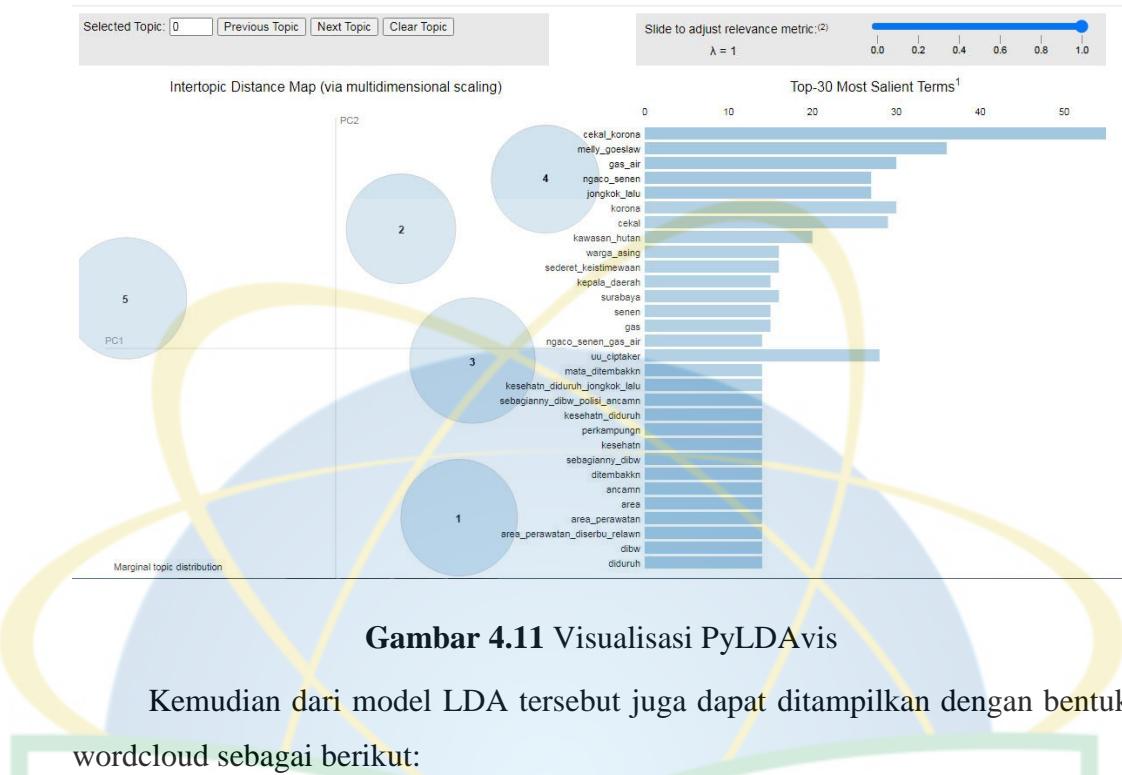
Pada letak untuk tiap topiknya yaitu tergantung pada titik koordinat dari masing-masing topik dengan melihat berdasarkan principal component (PC). Nilai PC untuk tiap topiknya yaitu pada tabel berikut.

**Tabel 4.14** Nilai Principal Component (PC)

Topik	PC1	PC2
1	0.026551	-0.058874
2	0.050556	-0.009772
3	0.029410	-0.066098
4	0.072049	0.111971
5	-0.178566	0.022772

Dapat dilihat pada Tabel 4.14 diatas yang merupakan nilai principal component (PC) untuk masing-masing topiknya yang dipakai dalam visualisasi PyLDAvis. Untuk nilai PC1 yang digunakan sebagai titik koordinat pada sumbu X, sedangkan nilai PC2 yang digunakan sebagai titik koordinat pada sumbu Y.

Selanjutnya dapat dilihat pada Gambar 4.15 dibawah jika dipilih salah satu topik maka lingkarannya akan berubah warna menjadi merah yang kemudian pada bar chart panel sisi kanan akan berubah berwarna merah yang memperlihatkan estimasi *term frequency* pada topik yang pilih.



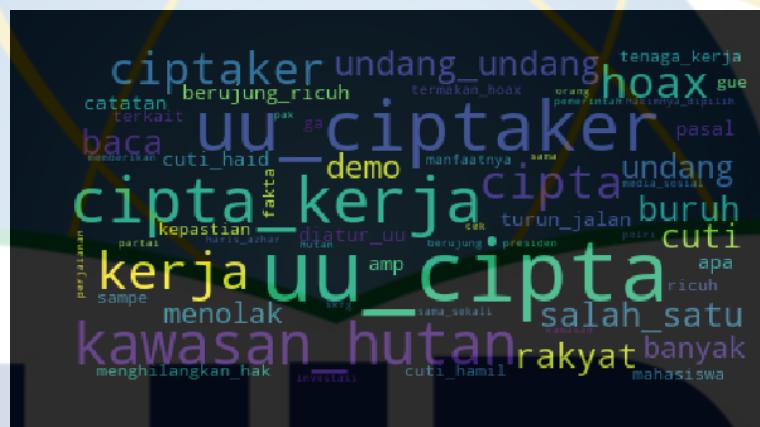
Kemudian dari model LDA tersebut juga dapat ditampilkan dengan bentuk wordcloud sebagai berikut:



## Gambar 4.12 Visualisasi Wordcloud Topik 1



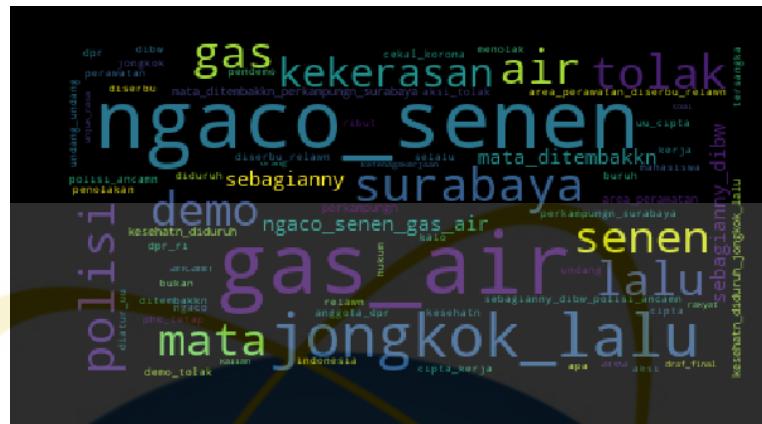
### Gambar 4.13 Visualisasi Wordcloud Topik 2



**Gambar 4.14** Visualisasi Wordcloud Topik 3



**Gambar 4.15** Visualisasi Wordcloud Topik 4



### **Gambar 4.16** Visualisasi Wordcloud Topik 5

Selanjutnya pada tahap interpretasi data yang didapat pada setiap topiknya akan di bahas pada bab *discovery*.

## BAB 5

# HASIL DAN PEMBAHASAN

### 5.1 *Discovery*

Dalam tahapan *discovery*, penulis melakukan analisis dari hasil pemodelan topik tweet *omnibus law* dengan *Latent Dirichlet Allocation* (LDA). Berdasarkan *coherence score* dengan nilai tertinggi maka jumlah topik yang didapatkan yaitu sebanyak 5 topik. *Output* penelitian ini penulis dapatkan dari aplikasi yang penulis bangun untuk membantu proses penelitian dengan menggunakan bahasa pemrograman *python*. Selanjutnya penulis melakukan interpretasi data *output* yang telah didapat. Berikut penulis jelaskan analisis *output* dari penelitian ini:

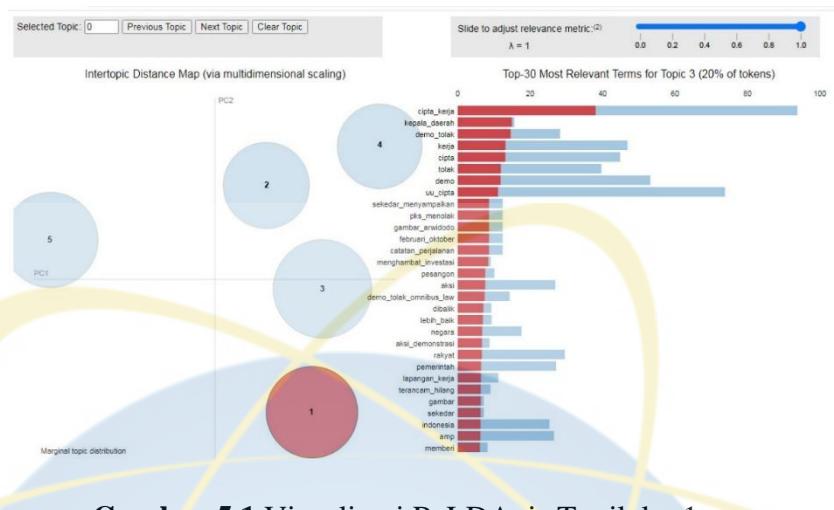
#### 5.1.1 Model LDA Topik ke-1

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

**Tabel 5.1** Model LDA Topik ke-1

Model LDA Topik 1
'0.018*"cipta_kerja" + 0.007*"kepala_daerah" + 0.007*"demo_tolak" + 0.006*"kerja" + 0.006*"cipta" + 0.006*"tolak" + 0.006*"demo" + 0.005*"uu_cipta" + 0.004*"catatan_perjalanan" + 0.004*"februari_oktober",'

Berdasarkan hasil perolehan data pada pemodelan topik LDA maka hasil tersebut dapat divisualisasikan dan ditampilkan dalam dua visualisasi data yaitu dengan visualisasi PyLDAvis dengan keterkaitan antar kata yang dihasilkan dan ditampilkan dalam bentuk wordcloud, visualisasi PyLDAvis topik 1 dapat dilihat pada Gambar 5.1 di bawah ini



**Gambar 5.1** Visualisasi PyLDAvis Topik ke-1

Berikut bentuk wordcloud hasil perolehan dari model LDA topik satu yang ditampilkan pada Gambar 5.2 di bawah ini



**Gambar 5.2** Visualisasi Wordcloud Topik ke-1

Berdasarkan model LDA pada Topik 1 didapatkan hasil kata-kata yang sering muncul dan berkaitan satu sama lain adalah kata-kata cipta kerja, kepala daerah, demo tolak, kerja, cipta, tolak, demo, uu cipta, catatan perjalanan, dan Februari Oktober, sehingga dari 10 kata yang diperoleh dapat diinterpretasikan menjadi pembahasan sebagai berikut:

#### 1. Demo tolak UU cipta kerja *omnibus law*

Pembahasan yang paling dominan dan sering muncul dalam *omnibus law* berdasarkan hasil pemodelan topik LDA adalah mengenai demo tolak UU cipta

kerja. Setiap topik dari hasil pemodelan topik LDA terdapat kata cipta kerja, diantaranya membahas demo tolak uu cipta kerja, selain itu hasil perolehan data pemodelan topik LDA diperkuat kembali dengan kutipan oleh CNN Indonesia yang menyatakan adanya pasal kontroversial dalam Omnibus cipta kerja yang mendapat sorotan dari banyak elemen masyarakat, sehingga dapat dikatakan bahwa *omnibus law* cipta kerja ini menjadi topik utama dalam pembahasan *omnibus law*.

## 2. Keterlibatan Kepala Daerah terhadap terhadap penolakan *omnibus law*

Adanya keterlibatan kepada daerah dalam pergerakan menolak *omnibus law* menjadi hal yang sering muncul dan di bahas oleh kalangan masyarakat. Hal ini dikarenakan Kepala Daerah dianggap mendukung dan ada di pihak masyarakat dalam menolak *omnibus law*. Menurut tribunnews.com beberapa kepala daerah yang turut andil dalam penolakan *omnibus law* adalah Gubernur Jawa Barat, Gubernur Daerah Istimewa Yogyakarta, Gubernur Kalimantan Barat, Gubernur Sumatera Barat, Bupati Bandung Barat, Bupati Bandung, Bupati Subang, Bupati Garut, Bupati Tegal, Bupati Limapuluh Kota, dan Walikota Malang.

## 3. Catatan perjalanan *omnibus law* dari bulan februari hingga oktober

Rekam jejak atau catatan perjalanan omnibuslaw mulai bulan Februari hingga bulan Oktober menjadi topik akhir yang paling banyak di bahas. Kutipan oleh Indonesia.go.id menjelaskan bahwa rekam jejak perjalan dimulai pada bulan Februari tanggal 3 setelah prsiden mengirimkan draf RUU Cipta Kerja kepada DPR hingga 5 oktober RUU Cipta kerja disahkan menjadi UU Cipta Kerja.

Sehingga berdasarkan beberapa data yang diperoleh dapat disimpulkan bahwa pada pemodelan LDA topik 1 membahas tentang penolakan UU *omnibus law* Cipta Kerja oleh sejumlah lapisan masarkat dan kepala daerah, serta membicarakan mengenai perjalanan RUU *omnibus law* cipta kerja pada bulan Februari hingga disahkan menjadi UU cipta kerja pada bulan oktober. Pernyataan dari pembahasan di atas dapat memperkuat keakurasi data hasil pemodelan topik LDA yang digunakan sesuai dengan informasi dan berita yang beredar.

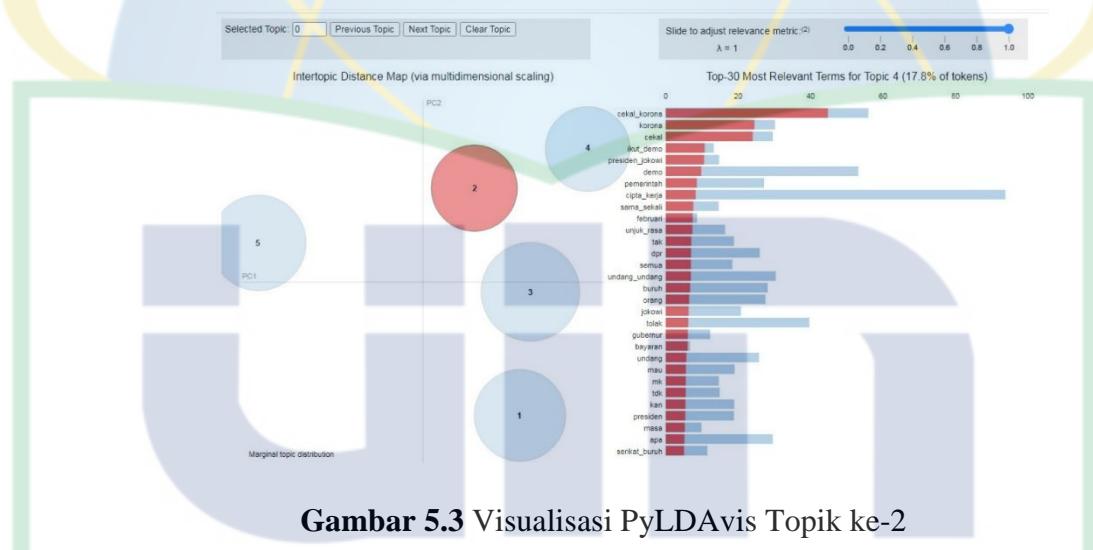
### 5.1.2 Model LDA Topik ke-2

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

**Tabel 5.2** Model LDA Topik ke-2

Model LDA Topik 2
'0.024*"cekal_korona" + 0.013*"korona" + 0.013*"cekal" + 0.006*"ikut_demo" + "0.006*"presiden_jokowi" + 0.005*"demo" + 0.005*"pemerintah" + 0.004*"cipta_kerja" + 0.004*"sama_sekali" + 0.004*"februari"'

Berdasarkan hasil perolehan data pada pemodelan topik LDA maka hasil tersebut dapat divisualisasikan dan ditampilkan dalam dua visualisasi data yaitu dengan visualisasi PyLDAvis dengan keterkaitan antar kata yang dihasilkan dan ditampilkan dalam bentuk wordcloud, visualisasi PyLDAvis topik 1 dapat dilihat pada Gambar 5.3 di bawah ini



**Gambar 5.3** Visualisasi PyLDAvis Topik ke-2

Berikut bentuk wordcloud hasil perolehan dari model LDA topik satu yang ditampilkan pada Gambar 5.4 di bawah ini



**Gambar 5.4** Visualisasi Wordcloud Topik ke-2

Berdasarkan model LDA pada Topik 2 didapatkan hasil kata-kata yang sering muncul dan berkaitan satu sama lain adalah kata-kata cekal korona, korona, cekal, ikut demo. Presiden jokowi, demo, pemerintah, cipta kerja, sama sekali, Februari, sehingga dari 10 kata yang diperoleh dapat diinterpretasikan menjadi pembahasan sebagai berikut:

## 1. Corona di demo *omnibus law*

Dalam topik 2 ini banyak masyarakat baik di kehidupan sehari-hari maupun di media sosial yang membicarakan mengenai virus corona pada demo omnibuslaw. Hal ini disebabkan karena unjuk rasa yang dilakukan oleh demonstran terjadi pada saat pandemi *Covid-19* dan di masa Pembatasan Sosial Berskala Besar (PSBB) yang dilakukan untuk pencegahan penularan *Covid-19*, sedangkan demonstran yang melakukan unjuk rasa rata-rata tidak mematuhi protocol kesehatan dan kebijakan PSBB yang sedang dilakukan, sehingga muncul kekhawatiran masyarakat akan adanya penyebaran virus Corona yang tak terkendalikan. Hal ini sesuai dengan kabar berita yang dirilis oleh detik.com yang menyatakan bahwa adanya unjuk rasa *omnibus law* di masa pandemi *Covid-19* menjadikan adanya cluster baru penyebaran virus corona.

## 2. Presiden jokowi dan pemerintah

Wacana Presiden beserta jajarannya untuk menyusun *omnibus law* saat pertama kali dilantik kembali diperbincangkan oleh masyarakat. Masyarakat

menganggap adanya rencana untuk menyusun serta merealisasikan *omnibus law* dalam waktu yang cepat merupakan hal yang dianggap terburu-buru dan dapat mengakibatkan hal-hal yang tak diinginkan di kemudian hari. Selain masyarakat, banyak platform media yang turut membahas adanya wacana Presiden dan Pemerintah dapat merealisasikan *omnibus law*, hal ini sesuai informasi berita yang diterbitkan oleh Suara.com yang menyatakan bahwa Presiden Jokowi ingin *omnibus law* segera diselesaikan dan meminta agar *omnibus law* tak hanya jadi wacana.

Berdasarkan interpretasi data model topik LDA topik ke-2 dapat disimpulkan bahwa adanya kekhawatiran masyarakat terhadap penyebaran virus Korona yang disebabkan oleh pelanggaran kebijakan PSBB yang dilakukan oleh demonstran dalam unjuk rasa *omnibus law*, serta *omnibus law* mulai menjadi fokus presiden berserta pemerintah untuk segera direalisasikan. Banyaknya berita maupun tanggapan masyarakat mengenai hal tersebut membuktikan kecocokan data dari hasil pemodelan Topik LDA yang didapatkan.

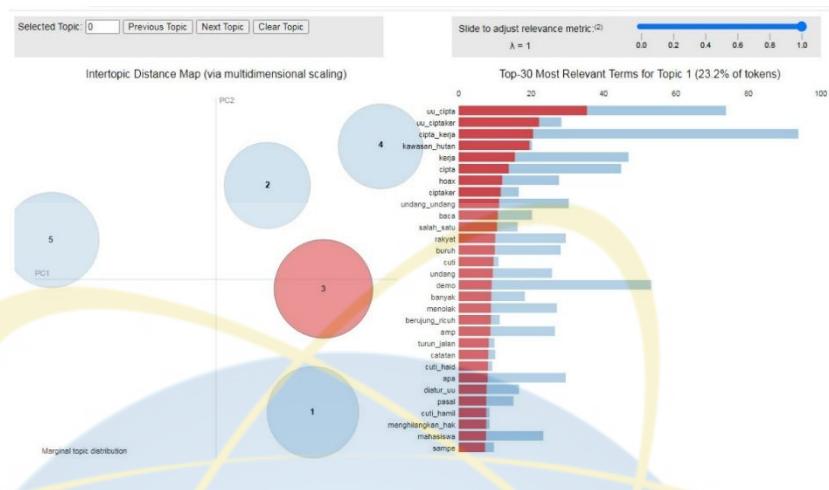
### 5.1.3 Model LDA Topik ke-3

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

**Tabel 5.3** Model LDA Topik ke-3

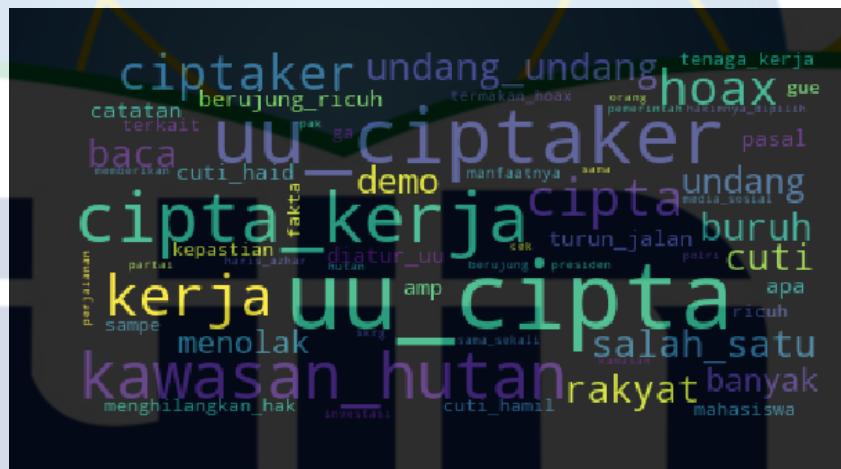
Model LDA Topik 3
'0.015*"uu_cipta" + 0.009*"uu_ciptaker" + 0.008*"cipta_kerja" + 0.008*"kawasan_hutan" + 0.006*"kerja" + 0.006*"cipta" + 0.0 05*"hoax" + 0.005*"ciptaker" + 0.005*"undang_undang" + 0.004* "baca"'

Berdasarkan hasil perolehan data pada pemodelan topik LDA maka hasil tersebut dapat divisualisasikan dan ditampilkan dalam dua visualisasi data yaitu dengan visualisasi PyLDAvis dengan keterkaitan antar kata yang dihasilkan dan ditampilkan dalam bentuk wordcloud, visualisasi PyLDAvis topik 1 dapat dilihat pada Gambar 5.5 di bawah ini



**Gambar 5.5** Visualisasi PyLDAvis Topik ke-3

Berikut bentuk wordcloud hasil perolehan dari model LDA topik satu yang ditampilkan pada Gambar 5.6 di bawah ini



**Gambar 5.6** Visualisasi LDA Topik ke-3

Berdasarkan model LDA pada Topik 3 didapatkan hasil kata-kata yang sering muncul dan berkaitan satu sama lain adalah kata-kata UU cipta, UU ciptaker, cipta kerja, kawasan hutan, kerja, cipta, hoax, ciptaker, undang undang, baca, sehingga dari 10 kata yang diperoleh dapat diinterpretasikan menjadi pembahasan sebagai berikut:

1. UU cipta kerja kawasan hutan / *cluster* tentang lingkungan hidup salah satu cluster paling banyak dalam pembahasan UU *omnibus law* cipta kerja yaitu mengenai kawasan hutan. Banyak masyarakat yang membahas

mengenai dampak potensi kebjakan *omnibus law* terhadap sektor kehutanan dan lingkungan. Menurut Susanti dan Richwanudin (2020) terdapat 4 potensi dampak *omnibus law* di sektor kehutanan dan lingkungan yaitu 1). Kemudahan pemberian perizinan tanpa pertimbangan aspek ekologis sangat riskan terhadap dampak lingkungan yang akan ditimbulkan kedepannya. 2) Pemanfaatan kawasan hutan lindung semakin tak terproteksi.3) Hilangnya AMDAL sebagai pintu gerbang terakhir penyelamatan lingkungan. 4) Semakin mudahnya perubahan peruntukan dan fungsi kawasan hutan serta penggunaan kawasan hutan sehingga dikhawatirkan akan semakin banyaknya konversi kawasan hutan yang tak sesuai fungsi kawasannya lagi yang dilakukan oleh oknum-oknum yang berkuasa tanpa pengawasan dan sepenuhnya masyarakat.

## 2. Hoax tentang *omnibus law*

Pembahasan paling banyak pada topik 3 yaitu pembahasan mengenai hoax tentang *omnibus law*, diantara banyaknya pembahasan mengenai *omnibus law* terdapat beberapa pihak yang memanfaatkan situasi untuk menyebarkan informasi palsu (Hoax), namun beberapa masyarakat telah mewaspadai hal tersebut dan banyak menyarankan kepada yang lain untuk membaca UU *omnibus law* secara rinci. Hal ini selaras dengan pernyataan Kementerian Komunikasi dan Informatika (Kemenkominfo) yang menilai masih banyak disinformasi atau penyampaian informasi yang salah soal omnibuslaw dan terus memantau penyebaran informasinya di media sosial. Hal itu dilakukan untuk mencegah opini-opini yang keliru

Berdasarkan model LDA pada Topik 3 didapatkan kesimpulan yang sesuai dengan hasil pemodelan pada topik 3 yang membahas mengenai dampak uu omnibuslaw cipta kerja terhadap kawasan hutan dan lingkungan, serta banyaknya berita hoax yang beredar tentang informasi tentang omnibusaw. Pemodelan Topik LDA memberikan hasil yang tidak bertolak belakang dengan data dan fakta yang ada.

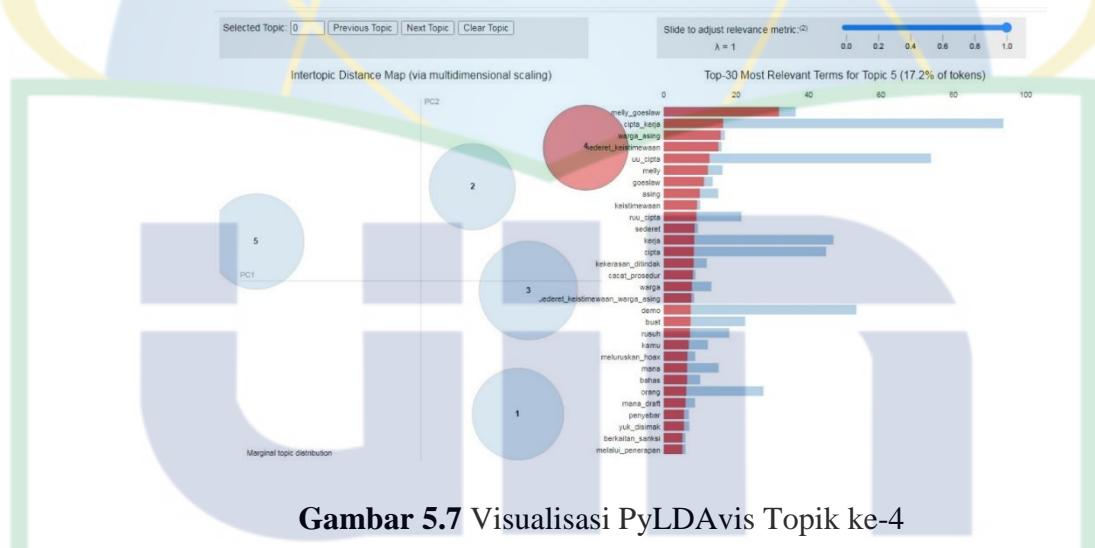
#### 5.1.4 Model LDA Topik ke-4

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

**Tabel 5.4** Model LDA Topik ke-4

Model LDA Topik 4
'0.018*"melly_goeslaw" + 0.009*"cipta_kerja" + 0.009*"warga_asing" + '0.008*"sederet_keistimewaan" + 0.007*"uu_cipta" + 0.007*"melly" + 0.006*"goeslaw" + 0.006*"asing" + 0.005*"keistimewaan" + 0.005*"ruu_cipta"'

Berdasarkan hasil perolehan data pada pemodelan topik LDA maka hasil tersebut dapat divisualisasikan dan ditampilkan dalam dua visualisasi data yaitu dengan visualisasi PyLDAvis dengan keterkaitan antar kata yang dihasilkan dan ditampilkan dalam bentuk wordcloud, visualisasi PyLDAvis topik 1 dapat dilihat pada Gambar 5.7 di bawah ini



**Gambar 5.7** Visualisasi PyLDAvis Topik ke-4

Berikut bentuk wordcloud hasil perolehan dari model LDA topik satu yang ditampilkan pada Gambar 5.8 di bawah ini



**Gambar 5.8** Visualisasi Wordcloud Topik ke-4

Berdasarkan model LDA pada Topik 4 didapatkan hasil kata-kata yang sering muncul dan berkaitan satu sama lain adalah kata-kata melly goeslaw, cipta kerja, warga asing, sederet keistimewaan, uu cipta, melly, doeslaw, asing, keistimewaan, RUU cipta, sehingga dari 10 kata yang diperoleh dapat diinterpretasikan menjadi pembahasan sebagai berikut:

1. Plesetan *omnibus law* menjadi Melly Goeslaw

Plesetan *omnibus law* menjadi Melly Goeslaw awalnya dimulai dari video seorang pria yang tengah mengkritik UU *omnibus law* (suara.com). selanjutnya plesetan itu ditiru oleh masyarakat di twetter baik untuk candaan atau bahkan sindiran terhadap *omnibus law*.

## 2. Keistimewaan warga asing

Salah satu pembahasan dalam omnibuslaw yaitu membahas mengenai Warga negara asing, masyarakat beranggapan bahwa warga asing dimanjakan dalam UU *omnibus law* Cipta Kerja. Menurut kutipan dari Detik.com dalam UU saku jagat beberapa pasal menyebutkan keistimewaan yang didapatkan oleh para warga asing seperti memperbolehkan warga asing mempunyai hak milik satuan rumah susun (sarasan). Hal ini menuai banyak kritikan masyarakat khusunya di twetter mengenai keistimewaan yang didapat warga asing dalam UU *omnibus law*.

Berdasarkan model LDA pada Topik 4 didapatkan kesimpulan yaitu mengenai bahan candaan dari masyarakat tentang plesetan *omnibus law* menjadi Melly Goeslaw. Selain itu netizen juga membahas mengenai keistimewaan yang didapat warga asing karena *omnibus law*. Pernyataan tersebut mengartikan bahwa pemodelan Topik LDA yang digunakan sudah tepat karena sesuai dengan informasi yang beredar.

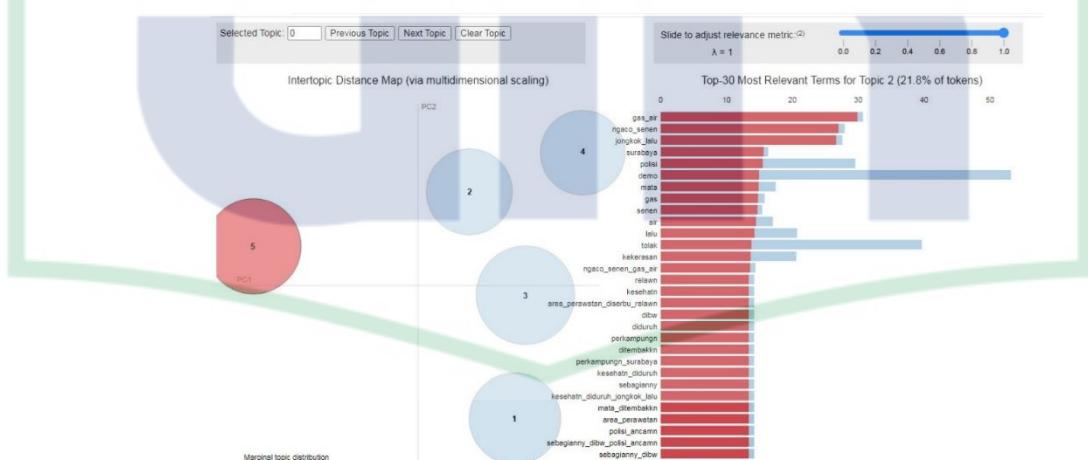
### 5.1.5 Model LDA Topik ke-5

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

**Tabel 5.5** Model LDA Topik ke-5

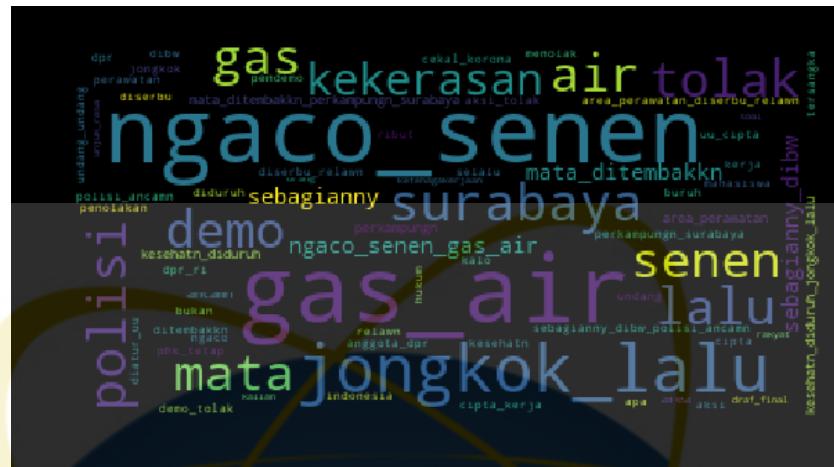
Model LDA Topik 5
' 0.013*"gas_air" + 0.012*"ngaco_senen" + 0.012*"jongkok_lalu" + 0.007*"surabaya" + 0.007*"polisi" + 0.007*"demo" + 0.007*"m ata" + 0.006*"gas" + 0.006*"senen" + 0.006*"air"'

Berdasarkan hasil perolehan data pada pemodelan topik LDA maka hasil tersebut dapat divisualisasikan dan ditampilkan dalam dua visualisasi data yaitu dengan visualisasi PyLDAvis dengan keterkaitan antar kata yang dihasilkan dan ditampilkan dalam bentuk *wordcloud*, visualisasi PyLDAvis topik 1 dapat dilihat pada Gambar 5.9 di bawah ini



**Gambar 5.9** Visualisasi PyLDAvis Topik ke-5

Berikut bentuk *wordcloud* hasil perolehan dari model LDA topik satu yang ditampilkan pada Gambar 5.10 di bawah ini



## Gambar 5.10 Visualisasi Wordcloud Topik ke-5

Berdasarkan model LDA pada Topik 2 didapatkan hasil kata-kata yang sering muncul dan berkaitan satu sama lain adalah kata-kata gas air, ngaco senen, jongkok lalu, surabaya, polisi, demo, mata, gas, senen, air, sehingga dari 10 kata yang diperoleh dapat diinterpretasikan menjadi pembahasan satu pembahasan yaitu Kerusuhan demo *omnibus law*

Pada topik 5 ini membahas mengenai kerusuhan demo *omnibus law* yang terjadi di jakarta di daerah pasar senen dan surabaya. Masyarakat menanggapi bahwa kerusuhan diakibatkan karena emosi yang tak terkendali dari demonstran dan provokator yang mungkin ikut hadir dalam demo omnibus law, hingga akhirnya Polisi bentrok dengan masa dan menembakan gas air mata. Polisi juga menangkap beberapa oknum demonstran sehingga menyuruhnya berbaris jongkok lalu dibawa ke kantor polisi.

Berdasarkan pernyataan tersebut maka didapatkan kesimpulan bahwa masyarakat banyak membahas mengenai kerusuhan yang terjadi pada saat demo tolak *omnibus law* di berbagai wilayah. Hasil yang diperoleh dari pemodelan topik LDA menunjukkan keselarasan data dengan fakta yang terjadi, hal ini mengindikasikan bahwa pemodelan topik LDA sudah tepat

## 5.2 Hasil Penelitian dan Pembahasan

Berdasarkan hasil penelitian yang telah diperoleh menggunakan metode LDA didapatkan sebanyak 5 jumlah topik (Topik 1- Topik 5). Topik ke-1 mengenai penolakan UU *omnibus law* Cipta Kerja oleh sejumlah lapisan masyarakat dan kepala daerah, serta membicarakan mengenai perjalanan RUU *omnibus law* cipta kerja pada bulan Februari hingga disahkan menjadi UU cipta kerja pada bulan Oktober. Sehingga hasil pemodelan topik menggunakan LDA dari tweet warganet tentang *omnibus law* pada media sosial Twitter didalam Topik 1 dapat digunakan sebagai sumber data oleh beberapa stakholder terkait seperti pemerintah pusat, pemerintah daerah, dan kementerian ketenaga kerjaan untuk menjadi bahan evaluasi terhadap *omnibus law* cipta kerja yang telah disahkan dan pertimbangan pemerintah dalam menentukan setiap kebijakan selanjutnya.

Topik ke-2 mengenai kekhawatiran masyarakat terhadap penyebaran virus Korona yang disebabkan oleh pelanggaran kebijakan PSBB yang dilakukan oleh demonstran dalam unjuk rasa *omnibus law*. Sehingga hasil pemodelan topik menggunakan LDA dari tweet warganet tentang *omnibus law* pada media sosial Twitter didalam Topik 2 dapat digunakan sebagai sumber data oleh beberapa stakholder terkait seperti pemerintah pusat, pemerintah daerah, dan kementerian kesehatan untuk menjadi bahan evaluasi kebijakan dan pertimbangan pemerintah dalam memperketat peraturan disiplin kesehatan dan meningkatkan kampanye protokol kesehatan dalam masa pandemi covid-19.

Topik ke-3 mengenai dampak UU omnibuslaw cipta kerja terhadap kawasan hutan dan lingkungan, serta banyaknya berita hoax yang beredar tentang informasi tentang omnibusaw. Sehingga hasil pemodelan topik menggunakan LDA dari tweet warganet tentang *omnibus law* pada media sosial Twitter didalam Topik 3 dapat digunakan sebagai sumber data oleh beberapa stakholder terkait seperti pemerintah pusat, kementerian Lingkungan Hidup dan Kehutanan, dan Kementerian Komunikasi dan Informatika untuk menjadi bahan evaluasi dan pertimbangan pemerintah dalam menentukan setiap kebijakan yang menyangkut kawasan hutan lindung. Serta

sosialisasi pemerintah tentang *omnibus law* dan segera melakukan klarifikasi berta hoax yang telah beredar.

Topik ke-4 mengenai bahan candaan dari masyarakat tentang plesetan omnibusaw menjadi melly goeslaw dan keistimewaan yang didapat warga asing karena *omnibus law*. Sehingga hasil pemodelan topik menggunakan LDA dari tweet warganet tentang *omnibus law* pada media sosial Twitter didalam Topik 4 dapat digunakan sebagai sumber data oleh beberapa stakholder terkait seperti pemerintah pusat, pemerintah daerah, kementerian ketenaga kerjaan dan kementerian agraria dan tata ruang untuk menjadi bahan evaluasi dan pertimbangan pemerintah dalam menentukan setiap kebijakan hak-hak untuk warga negara asing (WNA) di Indonesia.

Topik ke-5 kerusuhan yang terjadi pada saat demo tolak *omnibus law* di berbagai wilayah. Sehingga hasil pemodelan topik menggunakan LDA dari tweet warganet tentang *omnibus law* pada media sosial Twitter didalam Topik 5 dapat digunakan sebagai sumber data oleh beberapa stakholder terkait seperti pemerintah pusat, pemerintah daerah, dan kepolisian untuk menjadi bahan evaluasi dan pertimbangan dalam penegakan hukum serta penertiban demo unjuk rasa yang tidak tertib atau ricuh.

## **BAB 6**

### **PENUTUP**

#### **6.1 Kesimpulan**

Berdasarkan hasil pembahasan dari penelitian yang telah penulis lakukan, berikut merupakan kesimpulan pada penelitian ini:

1. Pemodelan topik dengan *Latent Dirichlet Allocation* (LDA) menggunakan data dari pendapat masyarakat tentang *omnibus law* di Twitter sebanyak 3644 *tweet*. Selanjutnya data *tweet* dilakukan *process preprocessing* yang memiliki tahapan *case folding*, *remove punctuation*, *stopword* dan *tokenizing*. Tahap bigram dan trigram digunakan sebelum melakukan proses pembobotan *term frequency-invers document frequency* (TF-IDF). Tahap *word embedding* menggunakan *library* word2vec untuk mengubah data string kata menjadi angka yang kemudian disimpan dalam *bag of word*. Selanjutnya data disiapkan menjadi *dictionary* dan *corpus* untuk proses pemodelan topik menggunakan LDA dengan menentukan jumlah topik berdasarkan *topic coherence score*.
2. Hasil dari analisis pemodelan topik dengan metode *Latent Dirichlet Allocation* (LDA) pada data pendapat masyarakat tentang *omnibus law* di twetter diperoleh jumlah topik sebanyak 5 dengan *coherence score* sebesar 0.5644. Kesimpulan yang dihasilkan yaitu sebagai berikut :
  - Model LDA topik ke-1 yaitu tentang penolakan UU *omnibus law* Cipta Kerja oleh sejumlah lapisan masarakat dan kepala daerah, serta membicarakan mengenai perjalanan RUU *omnibus law* cipta kerja pada bulan Februari hingga disahkan menjadi UU cipta kerja pada bulan oktober.
  - Model LDA topik ke-2 yaitu tentang kekhawatiran masyarakat terhadap penyebaran virus Korona yang disebabkan oleh pelanggaran kebijakan PSBB yang dilakukan oleh demonstran dalam unjuk rasa *omnibus law*, serta *omnibus law* mulai menjadi fokus presiden berserta pemerintah untuk segera direalisasikan.

- Model LDA topik ke-3 yaitu tentang dampak uu omnibuslaw cipta kerja terhadap kawasan hutan dan lingkungan, serta banyaknya berita hoax yang beredar tentang informasi tentang omnibusaw.
- Model LDA topik ke-4 yaitu tentang bahan candaan dari masyarakat tentang plesestan *omnibus law* menjadi Melly Goeslaw. Selain itu netizen juga membahas mengenai keistimewaan yang didapat warga asing karena *omnibus law*.
- Model LDA topik ke-5 yaitu kerusuhan yang terjadi pada saat demo tolak *omnibus law* di berbagai wilayah.

## 6.2 Saran

Dalam penelitian ini, penulis menyadari terdapat masih banyak kekurangan dan keterbatasan pada penelitian yang telah penulis kerjakan. Oleh sebab itu berdasarkan penelitian ini, penulis menyarankan untuk mengembangkan penelitian ini menjadi lebih baik dengan beberapa poin seperti berikut ini:

3. Jumlah data yang digunakan pada penelitian berikutnya bisa lebih banyak lagi.
4. Pada penelitian ini, data diambil dari media sosial Twitter, pada penelitian berikutnya diharapkan mengambil data dari media sosial lain seperti Youtube atau Instagram.
5. Dapat melalukan penyempurnaan proses *preprocessing* diantaranya yaitu pada bagian *stopwords* dengan menyesuaikan kata apa saja yang dapat dihilangkan yang tidak menghilangkan makna kata yang terkandung. Kemudian dapat ditambahkan dengan proses *stemming*.
6. Dapat mengembangkan pemodelan topik dengan metode yang lainnya.

## DAFTAR PUSTAKA

- Abdullah, A. H. (2018). *Word Embedding*. Retrieved Desember 27, 2020, from rpubs.com: <https://www.rpubs.com>
- Adam, R. (2019). *Word Embedding Bahasa Indonesia Menggunakan Fasttext*. Retrieved Desember 27, 2020 , from <https://structilmy.com>.
- Agustina. (2017). *Analisis Dan Visualisasi Suara Pelanggan Pada Pusat Layanan Pelanggan dengan Pemodelan Topik*. Surabaya: [Skripsi]. Institut Teknologi Sepuluh November.
- Antoni, P. (2020). Penerapan Omnibus Law Dalam Upaya Reformasi Regulasi. *Jurnal Legislasi Indonesia*, Vol 17 No. 1.
- Bagus, C. W. (2017). Text Mining pada Media Sosial Twitter (Studi Kasus: Masa Tenang Pilkada DKI 2017 Putaran 2). *Universitas Jember*.
- Bappenas. (2019). Pokok Permasalahan dan Regulasi di Indonesia. *Kajian Reformasi Regulasi di Indonesia*, Halaman 54.
- Barde, B. V., & Bainwad, A. M. (2017). An Overview of Topic Modeling Methods and Tools. *IEEE International Conference on Intelligent Computing and Control Systems Madurai*.
- Bbc.com. (2020). *Omnibus Law: Demo tolak UU Cipta Kerja di 18 provinsi diwarnai kekerasan, YLBHI: 'Polisi melakukan pelanggaran*. Retrieved Desember 27, 2020, from <https://www.bbc.com/indonesia/indonesia-54469444>
- Blei, D. (2012). Probabilistic topic models. *Communications of the ACM*, 55(4): 77-84.
- Blei, D., Jordan, M., & Ng, A. (2003). Latent dirichlet allocation. *Journal of Machine Learning Reserach*, 3: 993-1022.
- Busroh, F. (2017). Konseptualisasi Omnibus Law dalam menyelesaikan Permasalahan Regulasi Pertahanan. *Arena Hukum*, Vol. 10 (2) 241.

- Campbell, J., Hindle, A., & Stroulia, E. (2014). Latent Dirichlet Allocation : Extracting Topics from Software Engineering Data. 1–21.
- Carley, K., Kowalchuk, M., & Malik, M. (2015). Twitter usage in indonesia. *Center for the Computational Analysis of Social and Organizational Systems*.
- Cendana, M., & Permana, S. D. (2019). PRA-PEMROSESAN TEKS PADA GRUP WHATSAPP UNTUK PEMODELAN TOPIK. *Jurnal Mantik Penusa*, Volume 3, No. 3, 112.
- Chawla, I., & Singh, S. K. (2013). Performance evaluation of VSM and LSI models to determine Bug reports similarity. *IEEE Sixth International Conference on Contemporary Computing (IC3) Noida*.
- Cnbcindonesia.com. (2020). *Demo Omnibus Law Bisa Bikin Pasien Covid-19 Tambah 6.000/hari*. Retrieved Desember 27 , 2020, from <https://www.cnbcindonesia.com/news/20201009150955-4-193202/demo-omnibus-law-bisa-bikin-pasien-covid-19-tambah-6000-hari>
- Cnnindonesia.com. (2020). *Poin-poin Kontroversial Omnibus Law*. Retrieved Desember 27 , 2020, from Cnnindonesia.com: <https://www.cnnindonesia.com/nasional/20201024173045-12-562388/poin-poin-kontroversial-omnibus-law>
- Cnnindonesia.com. (2020). *Twitter Rilis Tagar Populer 2020 #GagalkanOmnibusLaw Teratas*. Retrieved Januari 10, 2021, from <https://www.cnnindonesia.com/teknologi/20201208110609-185-579296/twitter-rilis-tagar-populer-2020-gagalkanomnibuslaw-teratas>
- Detik.com. (2020). *Kominfo Ungkap Hoax dan Fakta Seputar Omnibus Law Cipta Kerja*. Retrieved Desember 27 , 2020, from <https://news.detik.com/berita/d-5206053/kominfo-ungkap-hoax-dan-fakta-seputar-omnibus-law-cipta-kerja>
- Detik.com. (2020). *Sederet Keistimewaan buat Warga Asing di Omnibus Law Ciptaker*. Retrieved Desember 27 , 2020, from

<https://finance.detik.com/berita-ekonomi-bisnis/d-5209514/sederet-keistimewaan-buat-warga-asing-di-omnibus-law-ciptaker>

- Djati, B. S. (2007). *Simulasi Teori dan Aplikasinya*. Yogyakarta: Andi.
- Djiwandono, & Istiarto, P. (2015). *Meneliti itu Tidak Sulit: Metodologi Penelitian Sosial dan Pendidikan Bahasa*. Yogyakarta.
- Doig, C. (2015). *Introduction to topic modeling in python*. Diambil kembali dari <http://chdoig.github.io/pytexas2015-topic-modeling>.
- Fattah, R. (2016). *Twitter Text Mining Untuk Informasi Gempa Bumi Menggunakan T-IDF Di Indonesia*. Malang: Universitas Islam Negeri Maulana Malik Ibrahim.
- Feldman, R., & Sanger, J. (2007). The Text Mining Handbook Advanced Approaches in Analyzing Unstructured Data. *Cambridge University Press*.
- Girolami, M., & Kaban, A. (2003). On an equivalence between PLSI and LDA Proceedings of the 26th annual international ACM SIGIR. *conference on Research and development in informaion retrieval*. SIGIR 03 433.
- Gusti, A. I., Akbar, A. L., & Akbar, M. S. (2016). Optimasi Naive Bayes Dengan Pemilihan Fitur Dan Pembobotan Gain Ratio. *Jurnal Ilmiah Teknologi Informasi*, 23-24.
- Hanafi, A. (2009). Pengenalan Bahasa Suku Bangsa Indonesia Berbasis Teks Menggunakan Metode N-gram. *IT Telkom*.
- Herwanto, G. W. (2018). Document Clustering Dengan Latent Dirichlet Allocation Dan Ward Hierarichal Clustering. *Jurnal Pseudocode*, Volume V Nomor 2. ISSN 2355-5920.
- Hofmann, T. (2001). Probabilistic Latent Semantic Indexing, Proceedings of the Twenty-Second Annual International SIGIR. *Conference on Research and Development in Information Retrieval (SIGIR-99)*.

- Hudaya, C., Fakhrurroja, H., & Alamsyah, A. (2019). Analisis Persepsi Konsumen Terhadap Brand Go-Jek Pada Media Sosial Twitter Menggunakan Metode Sentiment Analysis dan Topic Modelling. *Jurnal Mitra Manajemen (JMM Online)*, Vol. 3 No.6, 667.
- Hukumonline.com. (2017, February 17 ). *Menimbang Konsep Omnibus Law Bila Diterapkan di Indonesia*. Diambil kembali dari Hukumonline.com: <https://www.hukumonline.com/berita/baca/lt58a6fc84b8ec3/menimbang-konsep-omnibuslaw-bila-diterapkan-di-indonesia/>
- Hutomo, A., & Zulhanif. (2013). Analisis Keluhan Penumpang PT. Kereta Api Indonesia (Persero) Menggunakan LSA dan Analisis Korespondensi.
- Indonesia.go.id. (2020). *Perjalanan Omnibus Law Cipta Kerja Hingga Menjadi UU*. Dipetik Desember 27 , 2020, dari Indonesia.go.id: <https://indonesia.go.id/gallery/perjalanan-omnibus-law-cipta-kerja-hingga-menjadi-uu>
- Jubilee Enterprise. (2017). *Pemograman Python*. Jakarta: PT Elex Media Komputindo.
- Kabiru, I. N., & Sari, P. K. (2019). Analisa Konten Media Sosial E-Commerce Pada Instagram Menggunakan Metode Sentimen Analysis Dan LDA-Based Topic Modeling (Studi Kasus: Shopee Indonesia). *e-Proceeding of Management*, Vol.6, No.1, 14.
- Kakiay, T. J. (2004). *Pengantar Sistem Simulasi*. Yogyakarta: Andi.
- Karmayasa, O. (2012). Implementasi Vector Space Model dan Beberapa Notasi Metode Term Frequency Inverse Document Frequency (TF-IDF) Pada Sistem Temu Kembali Informasi. *Jurnal Elektronik Ilmu Komputer Udayana*, Volume 1 No 1, 2.
- Kengken, R. (2014). *Pemodelan Topik Untuk Media Sosial Menggunakan Latent Dirichlet Allocation*. Yogyakarta(ID): [skripsi]. Universitas Gadjah Mada.

- Kuhlman, D. (2013). Beginning Python, Advanced Python, and Python Exercises. *A Python Book.*
- Kumar, S. (2013). Chemistry and Biological Activities of Flavonoids: An Overview. *The ScientificWorld Journal*, 1-16.
- Kumparan.com. (2020). *Berbagai Kerusuhan yang Terjadi saat Demo Tolak Omnibus Law.* Retrieved Desember 27, 2020, from <https://kumparan.com/kumparannews/berbagai-kerusuhan-yang-terjadi-saat-demo-tolak-omnibus-law-1uMLXm9AwkH>
- Kurniawan, F. (2020). Problematika pembentukan ruu cipta kerja dengan konsep omnibus law pada klaster ketenagakerjaan pasal 89 angka 45 tentang pemberian pesangon kepada pekerja yang di phk. *Jurnal Panorama Hukum*, Vol. 5 No. 1.
- Listari. (2019). *Inisiasi Natural Language Processing (NLP) dan Klasifikasi Jenis Wisata Kuliner Untuk Program CHATBOT, (Studi Kasus : Informasi Wisata Kuliner Daerah Istimewa Yogyakarta)*. Yogyakarta: Statistika, UII.
- Liu, Z., & Li, M. (2013). High performance latent dirichlet allocation for text mining. (London: Brunel University).
- Mikolov, T., Chen, K., Corrado, & Dean, G. J. (2013). Efficient Estimation of Word Representations in Vector Space. *ICLR Workshop*, arXiv:1301.3781v3 [cs.CL].
- Mikolov, T., Sutskever, I., Chen, K., Corrado, K., & Dean, J. (2013). Distributed Representations of Words and Phrases and their Compositionality. *NIPS*.
- Nair, G. (2016). *Text mining 101: Topic modeling*. Diambil kembali dari <http://www.kdnuggets.com/2016/07/text-mining-101-topicmodeling.html>.
- Nugroho, D. D., & Alamsyah, A. (2018). Analisis Konten Pelanggan Airbnb Pada Network Sosial Media Twitter Content Analysis Of Airbnb Customer Based On Twitter Social Media. *e-Proceeding of Management*, 1623 & 1626.

- Prabowo, A. S., Triputra, A. N., & Junaidi, Y. (2020). Politik Hukum Omnibus Law di Indonesia. *Jurnal Pamator*, Volume 13 No. 1. doi:doi.org/10.21107/pamator.v13i1.6923
- Privacy Exchange.org. (2017). *A global information resource on consumers, commerce, and data protection worldwide National Omnibus Laws.* Retrieved from Privacy Exchange.org: <http://www.privacyexchange.org/legal/nat/omni/>
- Pusat studi Hukum dan Kebijakan Indonesia. (2019). PSHK Sampaikan Masukan Prolegnas dan Omnibus law. *PSHK*, Halaman 65.
- Putra, I. K. (2017). *Analisis Topik Informasi Publik Media Sosial di Surabaya Menggunakan Pemodelan Latent Dirichlet Allocation (LDA).* Surabaya: [skripsi]. Institut Teknologi Sepuluh November.
- Putra, I. M. (2017). *Analisis Topik Informasi Publik Media Sosial Di Surabaya Menggunakan Pemodelan Latent Dirichlet Allocation (LDA).* Surabaya: Institut Teknologi Sepuluh Nopember.
- Putra, I. M., & Kusumawardani, R. P. (2017). Analisis Topik Informasi Publik Media Sosial di Surabaya Menggunakan Pemodelan Latent Dirichlet Allocation (LDA). *JURNAL TEKNIK ITS*, Vol. 6, No. 2 , A312.
- Pypi.org. (2018, Juni 6). *pyLDAvis 2.1.2.* Diambil kembali dari Pypi.org: <https://pypi.org/project/pyLDAvis/>
- Ramadhan, F. (2019). *Demi Investasi dan Daya Saing Global Jokowi Usulkan Omnibus Law.* Dipetik Desember 26, 2020, dari Tempo.co: <https://grafis.tempo.co/read/1864/demi-investasi-dan-daya-saing-global-jokowi-usulkan-omnibus-law>
- Rangkuti, M. D. (2020). *Analisis Topik Komentar Vidio Beberapa Akun Youtube E-Commerce Indonesia Menggunakan Metode Latent Direchlet Allocation (LDA).* Jakarta: [skripsi]. UIN Syarif Hidayatullah.

- Rossa, A. (2014). *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek* (2nd ed.). Bandung: Informatika.
- Sanjaya, S., & Absar, A. E. (2015). Pengelompokan Dokumen Menggunakan Winnowing Fingerprint dengan Metode K-Nearest Neighbour. *Jurnal CoreIT*, Vol.1, No.2, 52.
- Santika, T. (2019). *Evaluasi Perplexity untuk Pemodelan Topik Diskusi Agama Islam di Media Sosial Twitter Indonesia Tahun 2006 – 2018 Menggunakan Latent Dirichlet Allocation (LDA)*. Jakarta: [skripsi]. UIN Syarif Hidayatullah.
- Sargent, R. G. (2013). An introduction to verification and validation of simulation models. *Proceedings of the 2013 Winter Simulation Conference*, (hal. 321–327).
- Schervish, M., Seidenfeld, T., & Kadane, J. J. (2010). Coherence with Proper Scoring Rules. *Sun Yat University*.
- Schuerer, K., & Maufrais, C. (2010). Introduction to Programming using Python. *Boston: Pearson*, hal. 1–242. ISBN: 0132747189.
- Shaikh, Anoud, et al. (2019). Bringing Shape to Textual Data – A Feasible Demonstration. *Mehran University Research Journal of Engineering & Technology*, Vol. 38, No. 4, 901-914 October 2019. doi:10.22581/muet1982.1904.04
- Singh, S. (2015). *Spatial temporal analysis od social media data*. Munich (GER): [tesis]: Technische Universitat Munchen.
- Sridadi, B. (2011). *Pemodelan dan Simulasi Sistem*. Bandung: Informatika.
- Srifariyati. (2019). Prinsip Kepemimpinan Dalam Perspektif QS. An-nisa: 58-59. *Jurnal Madaniyah*, Volume 9 Nomor 1. ISSN : 2086-3462.
- Suara.com. (2020). *Presiden Jokowi Resmi Teken UU Omnibus Law Cipta Kerja*. Retrieved Desember 27 , 2020, from

- <https://www.suara.com/news/2020/11/03/071713/presiden-jokowi-resmi-teken-uu-omnibus-law-cipta-kerja?page=all>
- Suara.com. (2020). *Sampai Trending, Lelaki Ini Kritik Omnibus Law Tapi Sebut Melly Goeslaw*. Retrieved Desember 27 , 2020, from <https://www.suara.com/entertainment/2020/10/11/195000/sampai-trending-lelaki-ini-kritik-omnibus-law-tapi-sebut-melly-goeslaw?page=all>
- Sudaryono. (2015). *Metodologi Riset Di Bidang TI (Panduan Praktis, Teori dan Contoh Kasus)*. Yogyakarta: Andi.
- Sugiyono. (2012). *Metode Penelitian Kuantitatif, Kualitatif, dan R&D* . Bandung: Alfabeta.
- Susanti, F. D., & Richwanudin, S. A. (2020, Oktober). *EMPAT Potensi Dampak Kebijakan Omnibus Law di Sektor Kehutanan dan Lingkungan*. Retrieved from Senjak Institute Fakultas Kehutanan UGM: <https://sebijak.fkt.ugm.ac.id/2020/10/06/empat-potensi-dampak-kebijakan-omnibus-law-di-sektor-kehutanan-dan-lingkungan/>
- Susanto, A., Adzka , A., Mulyani , C., Al Faruq, M., & Aldebarant, N. (2020). Menupas Omnnibus law. *Kajian Pembahasan Formil. Dewan Mahasiswa Justicia, Fakultas Hukum, Universitas Gadjah Mada*, Vol 5(1). Diambil kembali dari <http://demajusticia.org/rilis-kajian-mengupas-omnibus-law-bikin-gaklaw-jilid-i-pembahasan-formil/>
- Tempo.co. (2020, Oktober 17 ). *Tempo.co*. Diambil kembali dari Dua Pekan Gelombang Demo Omnibus Law di Jakarta: <https://metro.tempo.co/read/1396824/dua-pekan-gelombang-demo-omnibus-law-di-jakarta/full&view=ok>
- Tribunnews.com. (2020). *Daftar Gubernur/Bupati/Wali Kota dan DPRD yang Tolak Omnibus Law*. Dipetik Desember 27 , 2020, dari Tribunnews.com: <https://www.tribunnews.com/nasional/2020/10/09/daftar-gubernurbupatiwali-kota-dan-dprd-yang-tolak-omnibus-law>

- Twitter. (2018, August 10). *Twitter*. Diambil kembali dari Support Twitter:  
<https://help.twitter.com/id>
- Utami, K. P. (2017). *Analisis Topik Data Media Sosial Twitter Menggunakan Model Topik Latent Dirichlet Allocation*. Bogor: [skripsi] Fakultas Matematika Dan Ilmu Pengetahuan Alam IPB.
- Wirasakti, L. A., Permadi, R., Hartanto, A., & Hartamitik. (2020). Pembuatan Kata Kunci Otomatis Dalam Artikel Dengan Pemodelan Topik. *Jurnal Media Informatika Budidarma*, ISSN 2614-5278. DOI 10.30865/mib.v4i1.1707.
- Wisdom, A. (2017). *Topic Modeling*. Retrieved Desember 12 , 2020, from Medium:  
<https://medium.com/square-corner-blog/topic-modeling-optimizing-forhuman-interpretability-48a81f6ce0ed>
- Xhu, Z., Cai, P., Wiemer, Hastings, A. C., Graesser, & Namara, D. (2007). XHu, Z. Cai, P. Wiemer-Hastings, A. C. Graesser dan D. S. M. Namara. . Strengths, Limitations and Extentions of LSA. dalam *Handbook of Latent Semantic Analysis*, T. K. Landauer, D. S. M. Namara, S. Dennis dan W. Kintsch, Penyunt. New York. Routledge.
- Yoren. (2018). *Perbandingan Raw Tf dan Binary Tf Pada Sistem Pencarian Di Situs Museum Wayang Kekayon Yogyakarta*. Yogyakarta: [Skripsi]. Program Studi Teknik Informatika Jurusan Teknik Informatika Fakultas Sains dan Teknologi Universitas Sanata Dharma.
- Yuniyati, M., Akbar, A., & Adiputra, W. (2017). *Komunikasi Sosial dan Aspirasi Masyarakat dalam Pembangunan Daerah*. Yogyakarta: [tesis] S2 Ilmu Komunikasi UGM.
- Zulhanif. (2016). Pemodelan Topik Dengan Latent Dirichlet Allocation. *Seminar Nasional Pendidikan Matematika UNPAD*. ISSN: 2528-4630.