

**TESIS**

**PERBANDINGAN METODE LSTM DAN BILSTM PADA ANALISIS  
SENTIMEN OPINI PUBLIK MENGENAI UNDANG-UNDANG KPK**



Disusun oleh:

**Nama : Ardian Nur Romadhan**  
**NIM : 20.77.1251**  
**Konsentrasi : Informatics Technopreneurshp**

**PROGRAM STUDI S2 TEKNIK INFORMATIKA**  
**PROGRAM PASCASARJANA UNIVERSITAS AMIKOM YOGYAKARTA**  
**YOGYAKARTA**

**2023**

**TESIS**

**PERBANDINGAN METODE LSTM DAN BILSTM PADA ANALISIS  
SENTIMEN OPINI PUBLIK MENGENAI UNDANG-UNDANG KPK**

**COMPARISON OF THE LSTM AND BILSTM METHODS ON PUBLIC  
OPINION SENTIMENT ANALYSIS REGARDING THE KPK LAW**

Diajukan untuk memenuhi salah satu syarat memperoleh derajat Magister



Disusun oleh:

**Nama : Ardian Nur Romadhan**  
**NIM : 20.77.1251**  
**Konsentrasi : Informaties Technopreneurshlp**

**PROGRAM STUDI S2 TEKNIK INFORMATIKA**  
**PROGRAM PASCASARJANA UNIVERSITAS AMIKOM YOGYAKARTA**  
**YOGYAKARTA**  
**2023**

**HALAMAN PENGESAHAN**

**PERBANDINGAN METODE LSTM DAN BILSTM PADA ANALISIS  
SENTIMEN OPINI PUBLIK MENGENAI UNDANG-UNDANG KPK**

**COMPARISON OF THE LSTM AND BILSTM METHODS ON PUBLIC  
OPINION SENTIMENT ANALYSIS REGARDING THE KPK LAW**

Dipersiapkan dan Disusun oleh

**Ardian Nur Romadhan**

**20.77.1251**

Telah Diujikan dan Dipertahankan dalam Sidang Ujian Tesis  
Program Studi S2 Teknik Informatika  
Program Pascasarjana Universitas AMIKOM Yogyakarta  
pada hari Rabu, 1 November 2023

Tesis ini telah diterima sebagai salah satu persyaratan  
untuk memperoleh gelar Magister Komputer

Yogyakarta, 1 November 2023  
**Rektor**

**Prof. Dr. M. Suyanto, M.M.**  
**NIK. 190302001**

## HALAMAN PERSETUJUAN

### PERBANDINGAN METODE LSTM DAN BILSTM PADA ANALISIS SENTIMEN OPINI PUBLIK MENGENAI UNDANG-UNDANG KPK

### COMPARISON OF THE LSTM AND BILSTM METHODS ON PUBLIC OPINION SENTIMENT ANALYSIS REGARDING THE KPK LAW

Dipersiapkan dan Disusun oleh

**Ardian Nur Romadhan**

20.77.1251

Telah Ditujikan dan Dipertahankan dalam Sidang Ujian Tesis  
Program Studi S2 Teknik Informatika  
Program Pascasarjana Universitas AMIKOM Yogyakarta  
pada hari Rabu, 1 November 2023

**Pembimbing Utama**

Prof. Dr. Ema Utami, S.Si., M.Kom  
NIK. 190302035

**Pembimbing Pendamping**

Anggit Dwi Hartanto, M.Kom  
NIK. 190302163

**Anggota Tim Penguji**

Dr. Andi Sunyoto, M.Kom  
NIK. 190302052

Dr. Kumara Ari Yuana, S.T., M.T  
NIK. 190302575

Prof. Dr. Ema Utami, S.Si., M.Kom  
NIK. 190302035

Tesis ini telah diterima sebagai salah satu persyaratan  
untuk memperoleh gelar Magister Komputer.

Yogyakarta, 1 November 2023

**Direktur Program Pascasarjana**

Prof. Dr. Kusriani, M.Kom.  
NIK. 190302106

## HALAMAN PERNYATAAN KEASLIAN TESIS

Yang bertandatangan di bawah ini,

Nama mahasiswa : Ardian Nur Romadhan  
NIM : 20.77.1251  
Konsentrasi : Informatics Technopreneurship

Menyatakan bahwa Tesis dengan judul berikut:  
**PERBANDINGAN METODE LSTM DAN BILSTM PADA ANALISIS SENTIMEN OPINI PUBLIK MENGENAI UNDANG-UNDANG KPK**

Dosen Pembimbing Utama : Prof. Dr. Ema Utami, S.Si., M.Kom  
Dosen Pembimbing Pendamping : Anggit Dwi Hartanto, M.Kom

1. Karya tulis ini adalah benar-benar ASLI dan BELUM PERNAH diajukan untuk mendapatkan gelar akademik, baik di Universitas AMIKOM Yogyakarta maupun di Perguruan Tinggi lainnya
2. Karya tulis ini merupakan gagasan, rumusan dan penelitian SAYA sendiri, tanpa bantuan pihak lain kecuali arahan dari Tim Dosen Pembimbing
3. Dalam karya tulis ini tidak terdapat karya atau pendapat orang lain, kecuali secara tertulis dengan jelas dicantumkan sebagai acuan dalam naskah dengan disebutkan nama pengarang dan disebutkan dalam Daftar Pustaka pada karya tulis ini.
4. Perangkat lunak yang digunakan dalam penelitian ini sepenuhnya menjadi tanggung jawab SAYA, bukan tanggung jawab Universitas AMIKOM Yogyakarta
5. Pernyataan ini SAYA buat dengan sesungguhnya, apabila di kemudian hari terdapat penyimpangan dan ketidakbenaran dalam pernyataan ini, maka SAYA bersedia menerima SANKSI AKADEMIK dengan pencabutan gelar yang sudah diperoleh, serta sanksi lainnya sesuai dengan norma yang berlaku di Perguruan Tinggi

Yogyakarta, 01 November 2023  
Yang Menyatakan,



Ardian Nur Romadhan

## HALAMAN PERSEMBAHAN

Alhamdulillah dengan rasa syukur yang mendalam dengan telah diselesaikannya tesis ini berkat segala nikmat dan kasih sayang yang telah diberikan oleh Allah Subhanahu Wa Ta'ala, maka penulis mempersembahkan tesis ini kepada semua pihak yang terlibat secara langsung ataupun tidak langsung dalam proses pembuatan tesis.

1. Orang tua dan Mertua yang tercinta
2. Isteri yang tercinta, Endras Dinawati
3. Ananda tersayang, Alesha Kenziewidjaya
4. Ananda tersayang, Ayesha Kenziewidjaya
5. Seluruh keluarga besar yang dengan setia memberikan dorongan moral selama penulis belajar di Universitas Amikom Yogyakarta.



## KATA PENGANTAR



*Assalamu'alaikum warohmatullahi wabarakatuh,*

Puji dan syukur penulis panjatkan kehadiran Allah Subhanahu Wa Ta'ala atas nikmat dan karunia-Nya sehingga penulis dapat menyelesaikan tesis yang berjudul “Perbandingan Metode LSTM dan BiLSTM Pada Analisis Sentimen Opini Publik Mengenai Undang-Undang KPK”. Penulis menyadari tidak akan dapat menyelesaikan tesis ini dengan baik tanpa bimbingan, saran dan motivasi dari berbagai pihak. Peneliti mengucapkan terimakasih yang sebesar-besarnya kepada:

1. Prof. Dr. M. Suyanto, M.M, selaku Rektor Universitas Amikom Yogyakarta.
2. Prof. Dr. Kusriani, M.Kom, selaku Direktur Program Pascasarjana Universitas Amikom Yogyakarta.
3. Prof. Dr. Ema Utami, S.Si., M.Kom, selaku Wakil Direktur Program Pascasarjana Universitas Amikom Yogyakarta sekaligus selaku Pembimbing Utama, yang telah menyetujui atas permohonan penulisan tesis ini dan senantiasa memberikan pengarahan dan bimbingan kepada penulis.
4. Anggit Dwi Hartanto, M.Kom, selaku Pembimbing Pendamping yang senantiasa memberikan pengarahan dan bimbingan dalam menghadapi permasalahan yang membuat penulis dapat bertahan terhadap kesulitan dan mampu memelihara semangat hingga karya ini selesai.

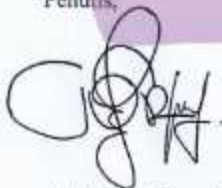
5. Tim Penguji Dr. Andi Sunyoto, M.Kom dan Dr. Kumara Ari Yuana, S.T., M.T, Segenap Dosen dan Karyawan Program Pascasarjana Universitas Amikom Yogyakarta yang telah memberikan bekal kepada penulis dengan ilmu pengetahuan yang sangat bermanfaat dalam hidup dan berkarya dalam masyarakat.
6. Teman-teman PJJ Magister Teknik Informatika Program Pascasarjana Universitas Amikom Yogyakarta, yang senantiasa berbagi perkembangan ilmu pengetahuan yang bermanfaat.
7. Semua pihak yang tidak dapat penulis sebutkan satu persatu, *syukron jazakumullah khoirun katsiron.*

Semoga Allah Subhanahu Wa Ta'ala memberikan balasan yang lebih kepada pihak yang telah membantu penulis menyelesaikan tesis ini. Semoga tesis ini dapat bermanfaat bagi masyarakat.

*Wasalamualaikum warohmatullahi wabarakatuh.*

Yogyakarta, 01 November 2023

Penulis,



Ardian Nur Romadhan



## DAFTAR ISI

HALAMAN JUDUL .....	ii
HALAMAN PENGESAHAN .....	iii
HALAMAN PERSETUJUAN.....	iv
HALAMAN PERNYATAAN KEASLIAN TESIS .....	v
HALAMAN PERSEMBAHAN .....	vi
KATA PENGANTAR .....	vii
DAFTAR ISI.....	ix
DAFTAR TABEL.....	xii
DAFTAR GAMBAR.....	xiii
DAFTAR LAMPIRAN.....	xiv
INTISARI .....	xv
<i>ABSTRACT</i> .....	xvi
<b>BAB I PENDAHULUAN.....</b>	<b>1</b>
1.1. Latar Belakang Masalah .....	1
1.2. Rumusan Masalah.....	6
1.3. Batasan Masalah .....	6
1.4. Tujuan Penelitian.....	7
1.5. Manfaat Penelitian .....	7
<b>BAB II TINJAUAN PUSTAKA .....</b>	<b>9</b>
2.1. Tinjauan Pustaka.....	9
2.2. Keaslian Penelitian .....	14
2.3. Landasan Teori .....	17

2.3.1. Sentimen Analisis .....	17
2.3.2. Penambangan Teks ( <i>Text Mining</i> ) .....	17
2.4. Pembobotan .....	19
2.5. Klasifikasi .....	20
2.5.1. <i>Convolutional Neural Network</i> (CNN) .....	20
2.5.2. <i>Long Short-Term Memory</i> (LSTM) .....	21
2.5.3. <i>Bidirectional Long Short-Term Memory</i> (BiLSTM) .....	24
2.5.4. Integrasi LSTM dan BiLSTM dengan CNN .....	25
2.5.5. Integrasi CNN dengan LSTM dan BiLSTM .....	25
2.6. <i>Confusion matrix</i> .....	26
<b>BAB III METODE PENELITIAN .....</b>	<b>27</b>
3.1. <i>Dataset</i> .....	27
3.2. <i>Preprocessing Data</i> .....	27
3.3. <i>Splitting data</i> .....	31
3.4. Evaluasi Model .....	31
3.5. Alur Penelitian .....	31
3.6. <i>Wordcloud</i> .....	33
3.7. Dasar Model Eksperimen .....	34
<b>BAB IV HASIL PENELITIAN DAN PEMBAHASAN .....</b>	<b>36</b>
4.1. Perbandingan Sentimen .....	36
4.2. Perbandingan <i>Wordcloud</i> .....	37
4.3. Pengujian Model .....	40

4.4. Pengujian Data Testing .....	42
4.5. Hasil Klasifikasi .....	46
4.5.1. Hasil Klasifikasi Eksperimen Pertama .....	46
4.5.2. Hasil Klasifikasi Eksperimen Kedua .....	47
4.5.3. Akurasi Terbaik Setiap Model Eksperimen .....	48
4.6. Pembahasan .....	48
2.6.1. Perbandingan Model Eksperimen .....	48
2.6.2. Evaluasi Performansi .....	49
4.2.3. Perbandingan Hasil Penelitian .....	51
4.2.4. Hasil uji sentimen .....	52
4.2.5. Kelebihan dan Kekurangan Penelitian .....	53
4.2.6. Kontribusi Keilmuan .....	54
BAB V PENUTUP .....	55
5.1. Kesimpulan .....	55
5.2. Saran .....	56
DAFTAR PUSTAKA .....	57
Lampiran .....	62

## DAFTAR TABEL

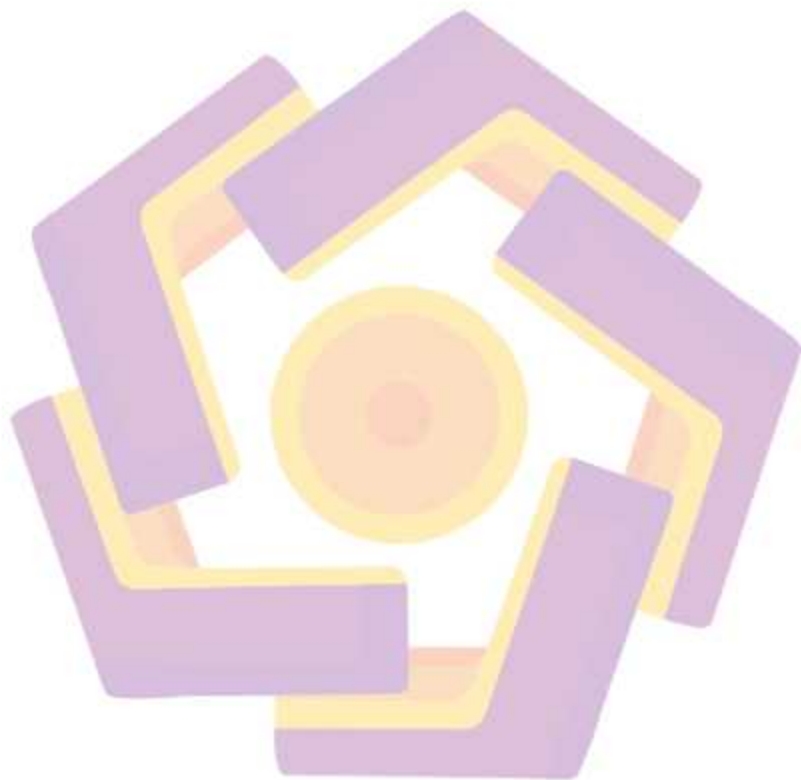
Tabel 1.1. <i>Gap Dataset</i> .....	4
Tabel 2.1. Matriks <i>literature review</i> dan posisi penelitian .....	14
Tabel 3.1. Sampel Tahapan <i>Preprocessing Data</i> .....	30
Tabel 4.1. <i>Confusion Matrix</i> Model LSTM+CNN .....	43
Tabel 4.2. <i>Confusion Matrix</i> Model BiLSTM+CNN .....	44
Tabel 4.3. <i>Confusion Matrix</i> Model CNN+LSTM .....	45
Tabel 4.4. <i>Confusion Matrix</i> Model CNN+BiLSTM .....	45
Tabel 4.5. Akurasi Terbaik .....	48
Tabel 4.6. Perbandingan Performasi .....	49
Tabel 4.7. Penelitian Terdahulu .....	51
Tabel 4.8. Hasil Prediksi .....	52

## DAFTAR GAMBAR

Gambar 3.1. Alur Penelitian .....	32
Gambar 3.2. <i>Wordcloud</i> .....	34
Gambar 4.1. Diagram Pie Perbandingan Sentimen.....	37
Gambar 4.2. <i>Wordcloud</i> pada Sentimen Negatif .....	38
Gambar 4.3. <i>Wordcloud</i> pada Sentimen Positif.....	38
Gambar 4.4. <i>Wordcloud</i> pada Sentimen Netral .....	39
Gambar 4.5. Sampel Pengujian Model LSTM+CNN .....	41
Gambar 4.6. Sampel Pengujian Model BiLSTM+CNN .....	41
Gambar 4.7. Sampel Pengujian Model CNN+LSTM.....	41
Gambar 4.8. Sampel Pengujian Model CNN+BiLSTM .....	42
Gambar 4.9. <i>Confusion Matrix</i> Model LSTM+CNN.....	43
Gambar 4.10. <i>Confusion Matrix</i> Model BiLSTM+CNN.....	44
Gambar 4.11. <i>Confusion Matrix</i> Model CNN+LSTM.....	44
Gambar 4.12. <i>Confusion Matrix</i> Model CNN+BiLSTM.....	45
Gambar 4.13. Hasil Klasifikasi Eksperimen Pertama.....	46
Gambar 4.14. Hasil Klasifikasi Eksperimen Kedua .....	47
Gambar 4.15. Perbandingan Model Eksperimen .....	49
Gambar 4.16. Perbandingan Performasi .....	50

## DAFTAR LAMPIRAN

Lampiran 1. <i>Dataset</i> .....	62
Lampiran 2. <i>Source Code</i> .....	63





## INTISARI

Tujuan penelitian yang berjudul “Perbandingan Metode LSTM dan BiLSTM Pada Analisis Sentimen Opini Publik Mengenai Undang-Undang KPK” ini adalah melakukan pengklasifikasian opini publik menggunakan integrasi model LSTM dan BiLSTM dengan CNN berdasarkan sentimen positif, negative, dan netral serta mengetahui tingkat akurasi model eksperimen LSTM+CNN, BiLSTM+CNN, CNN+LSTM, dan CNN+BiLSTM dalam preferensi sentiment analisis opini publik menggunakan pemodelan data teks. Topik UU KPK adalah bahasan yang menarik untuk dilakukan penelitian dan masih sedikit penelitian yang menerapkan metode *deep learning* yaitu LSTM dan BiLSTM dengan CNN untuk analisis sentimen pada UU KPK. Terdapat dua eksperimen dalam uji coba klasifikasi yaitu eksperimen pertama adalah menggunakan model LSTM+CNN dan BiLSTM+CNN dan eksperimen kedua menggunakan CNN+LSTM dan CNN+BiLSTM. Dataset yang digunakan adalah data *tweet* terkait UU KPK berbahasa Indonesia yang diambil dari Twitter. Hasil yang didapat menunjukkan bahwa metode *deep learning* LSTM dan BiLSTM dengan kombinasi CNN terbukti dapat menjadi alternatif dalam melakukan analisis sentimen opini publik menggunakan pemodelan data teks. Performa terbaik diperoleh pada eksperimen model BiLSTM+CNN sebesar 88.82%.

Kata kunci: LSTM, BiLSTM, CNN, Word2Vec.

## ABSTRACT

*The aim of the research entitled "Comparison of LSTM and BiLSTM Methods in Public Opinion Sentiment Analysis Regarding the Corruption Eradication Committee Law" is to classify public opinion using the integration of LSTM and BiLSTM models with CNN based on positive, negative and neutral sentiments and to determine the level of accuracy of the LSTM experimental model +CNN, BiLSTM+CNN, CNN+LSTM, and CNN+LSTM in sentiment preferences for public opinion analysis using text data modeling. The topic of the Corruption Eradication Commission Law is an interesting topic for research and there is still little research that applies deep learning methods, namely LSTM and BiLSTM with CNN for sentiment analysis on the Corruption Eradication Commission Law. There are two experiments in the classification trial, namely the first experiment uses the LSTM+CNN and BiLSTM+CNN models and the second experiment uses CNN+LSTM and CNN+BiLSTM. The dataset used is tweet data related to the Indonesian Corruption Eradication Committee Law taken from Twitter. The results obtained show that the LSTM and BiLSTM deep learning methods with a combination of CNN have proven to be an alternative in conducting public opinion sentiment analysis using text data modeling. The best performance was obtained in the BiLSTM+CNN model experiment at 88.82%.*

*Keyword: LSTM, BiLSTM, CNN, Word2Vec.*

# BAB I

## PENDAHULUAN

### 1.1. Latar Belakang Masalah

Komisi Pemberantasan Korupsi (KPK) merupakan lembaga terdepan untuk memberikan integritas kinerja yang baik dalam memberantas tindak pidana korupsi yang dibentuk berlandaskan UU No. 30 Tahun 2002 dan Undang-Undang ini kemudian direvisi dengan terbitnya UU No. 19 Tahun 2019 (Manullang dkk., 2023; Pakpahan & Simanullang, 2022). Nurul Ghufron selaku wakil ketua KPK mengajukan uji materi (*judicial review*) terkait aturan batas usia pimpinan KPK Pasal 29 huruf e dan Pasal 34 UU KPK terkait masa jabatan pimpinan KPK (Argawati, 2021; Z. Putri, 2023). Mahkamah Konstitusi (MK) mengabulkan seluruh permohonan pengujian UU No. 19 Tahun 2019 tentang perubahan kedua atas UU No. 30 Tahun 2002. Amar putusan, mengadili, mengabulkan permohonan Pemohon untuk seluruhnya, kata Ketua MK Anwar Usman saat membacakan putusan UU KPK pada Kamis 25 Mei 2023 dengan didampingi delapan hakim konstitusi (Argawati, 2023). Fenomena ini menjadi isu yang kontroversial yang menimbulkan perbedaan opini masyarakat. Opini merupakan pernyataan atau pendapat terhadap suatu isu yang kontroversial akibat dari diskusi tentang fenomena yang menimbulkan perbedaan opini (Sudrajat & Rohida, 2022).

Sosial media banyak menghasilkan berbagai macam informasi baik politik, sosial, ekonomi, teknologi, atau ilmu pengetahuan. Berbagai platform media sosial seperti Twitter, Facebook, Instagram, Youtube dan lainnya sering kali dimanfaatkan

pengguna untuk berinteraksi dan beropini dengan mengunggah teks, gambar, audio, dan video di laman media sosial yang dimiliki. Sejumlah besar pengguna media sosial memungkinkan para peneliti untuk menggunakan unggahan seseorang menjadi data sentimen untuk diproses dan dianalisis. Media sosial Twitter merupakan salah satu media sosial yang populer, dimana berbagai opini publik tercurah dari berbagai fenomena yang ada di lingkungan masyarakat, sehingga *dataset* opini publik terhadap institusi dan individu banyak tersedia dan *open access* di sosial media twitter. Opini publik yang terdapat pada twitter dapat digunakan untuk melihat bagaimana pergerakan sentimen berjalan (D. D. Putri dkk., 2022).

Sentiment analisis berguna dalam mengembangkan sistem untuk menganalisis, mengidentifikasi, dan mengungkapkan pendapat dalam bentuk teks dan juga merupakan proses yang berfungsi untuk mengidentifikasi berupa opini atau sentimen dari isi dataset, berupa teks tentang topik atau peristiwa positif dan negatif (Berliana & Santoso, 2023; Fikri dkk., 2020). Sentimen yang terkandung dalam opini publik dapat mempengaruhi pandangan masyarakat terhadap suatu hal atau pilihan (Hermanto dkk., 2021). Polaritas positif berarti *tweet* sangat antusias terhadap kandidat tertentu. Sedangkan polaritas negatif merupakan argumen negatif atau ketidaksenangan terhadap kandidat (Macrohon dkk., 2022).

Berdasarkan *literature review* yang didapatkan oleh penulis mengenai topik politik pemerintahan atas pemberlakuan undang-undang baik UU / RUU / KUHP / RKUHP adalah bahasan yang menarik untuk dilakukan penelitian karena memiliki dampak langsung kepada unsur-unsur Negara Indonesia. Penelitian terkait sentimen analisis opini publik terhadap rencana / perubahan / penerapan undang-undang,



sebelumnya pernah dilakukan oleh para peneliti terdahulu seperti yang dilakukan oleh (Rozalina dkk., 2023) memanfaatkan platform media sosial Twitter untuk menganalisis *dataset* dan mengekstrak informasi opini publik terkait UU No. 3 Tahun 2022 tentang Ibu Kota Negara (IKN) menggunakan *big data* untuk mendapatkan sentimen netral, positif, dan negatif. Lebih lanjut pada penelitian yang dilakukan oleh (Budayani dkk., 2023) juga menggunakan *text mining* dengan mengolah data teks dan digunakan untuk mengklasifikasikan opini publik terkait UU TPKS menggunakan SVM. Penelitian serupa menggunakan *machine learning* juga diimplementasikan oleh (Ndruru, 2022) melakukan sentiment analisis terhadap UU Cipta Kerja dengan menggunakan *Naive Bayes Classifier* dan SVM. Selain memanfaatkan *machine learning* pada penelitian terdahulu juga didapati menggunakan basis *deep learning* seperti pada penelitian yang dilakukan oleh (Abdurrazzaq & Tjong, 2022) memanfaatkan Twitter sebagai sumber *dataset* untuk mengekstrak informasi opini sentimen pada KUHP baru menggunakan BERT. Sentimen analisis dengan memanfaatkan data teks (teknik *text mining*) merupakan bidang penelitian yang baru dan banyak diteliti khususnya dibidang linguistik. Dimana pengklasifikasian polaritas opini dari sumber data yang berjumlah banyak dapat memanfaatkan fungsi dari *text mining* (Iskandar & Nataliani, 2021).

Sementara pada penelitian ini penulis akan memanfaatkan *deep learning* dengan menggunakan *Long Short-Term Memory* (LSTM) dan *Bidirectional Long Short-Term Memory* (BiLSTM) dalam pengklasifikasian. Penggunaan algoritma LSTM dan BiLSTM sering digunakan pada penelitian terdahulu untuk membuat

prediksi dan klasifikasi yang berhubungan dengan waktu atau pemodelan data *time series*. Berbeda dengan penelitian terdahulu peneliti memanfaatkan pemodelan data *text* dimana *dataset* bersifat statis yang didapat dari platform media sosial Twitter sebagai sumber *dataset* dalam bentuk data teks.

Tabel 1.1. *Gap Dataset*

<i>Gap</i>	<i>Dataset</i>	<b>Peneliti</b>	
Terdapat perbedaan pemodelan data pada klasifikasi LSTM dan BiLSTM	<i>time series</i>	(Hermanto dkk., 2021)	Dataset situs detik finance - Judul artikel bahasa Indonesia
	statis	(Srivastava dkk., 2023)	Dataset Twitter - Analisis Sentimen

Berdasarkan penelitian empiris terdahulu pada Tabel 1.1 dapat diketahui bahwa penggunaan algoritma LSTM dan BiLSTM dapat melakukan prediksi dengan menggunakan dataset selain *time series*, hal ini menjadi alasan peneliti untuk melakukan eksperimen pada dataset selain *time series* untuk preferensi algoritma LSTM dan BiLSTM. Oleh karena itu, berdasarkan uraian dalam latar belakang masalah yang telah tersaji dapat diartikan bahwa penggunaan pemodelan dataset selain *time series* pada algoritma LSTM dan BiLSTM dapat menjadi peluang alternatif dalam melakukan preferensi sentiment analisis. Dengan demikian berdasarkan kesenjangan dataset pada penelitian terdahulu, maka peneliti akan mengkaji lebih jauh menggunakan pemodelan dataset bersifat statis dengan membandingkan akurasi dua metode *deep learning* secara bersama antara LSTM dan BiLSTM dengan hybrid menggunakan CNN. Penelitian ini menggunakan model Hybrid *deep learning*, arsitektur dikatakan hybrid karena terdiri dari lapisan awal *convolutional* LSTM atau BiLSTM yang akan menerima *embeddings* kata



untuk setiap token dalam judul sebagai input, dengan kata lain hybrid dalam eksperimen penelitian ini merupakan integrasi (kombinasi) model. *Deep learning* adalah bidang *artificial intelligence* yang terbukti mampu memberikan peningkatan akurasi sebanding dengan penambahan jumlah data (Sunyoto, 2019). Studi yang berkaitan dengan *deep learning* dan analisis sentimen telah banyak dilakukan. Namun, penelitian terdahulu berada dalam dua hal yang berbeda. Penelitian ini menggabungkan sentimen opini publik mengenai Undang-Undang KPK dan *deep learning* menggunakan metode LSTM dan BiLSTM dengan dengan integrasi model CNN. LSTM merupakan metode yang lebih baik dibandingkan dengan konvensional, metode LSTM sangat cocok untuk diterapkan pada sentiment analisis (Jelodar dkk., 2020). Sementara BiLSTM dipilih sebagai perbandingan akurasi dikarenakan metode ini cocok untuk klasifikasi teks dan analisis metode ini memberikan hasil yang baik karena beroperasi di kedua arah, *forward* atau maju dan *backward* atau mundur (Lin dkk., 2023).

Berdasarkan permasalahan dan sajian *literature review* yang sudah dijelaskan diatas, pada penelitian ini penulis akan melakukan penelitian dengan judul "Perbandingan Metode LSTM dan BiLSTM Pada Analisis Sentimen Opini Publik Mengenai Undang-Undang KPK". *Novelty* penelitian ini yaitu menerapkan metode *machine learning* pada model *deep learning* yaitu dengan menggunakan LSTM dan BiLSTM dengan integrasi model menggunakan CNN. Penelitian ini membagi dalam dua *scenario* eksperimen, yaitu skenario pertama menggunakan integrasi antara CNN+LSTM, CNN+BiLSTM dan skenario kedua integrasi antara LSTM+CNN dan BiLSTM+CNN. Hasil dari penelitian ini diharapkan dapat

menjadi alternatif dalam melakukan preferensi sentiment analisis terutama pada opini publik dengan menggunakan arsitektur dari *deep learning* dan dataset bersifat statis.

### 1.2. Rumusan Masalah

Adapun rumusan masalah pada penelitian ini sebagai berikut:

- a. Apakah integrasi model LSTM+CNN, BiLSTM+CNN, CNN+LSTM, dan CNN+LSTM dapat digunakan untuk klasifikasi opini publik dalam preferensi pemodelan data teks?
- b. Berapa tingkat akurasi model eksperimen menggunakan integrasi model LSTM+CNN, BiLSTM+CNN, CNN+LSTM, dan CNN+LSTM dalam preferensi pemodelan data teks?

### 1.3. Batasan Masalah

Batasan masalah pada penelitian ini adalah:

- a. Dataset yang digunakan merupakan kumpulan data teks (*text mining*) bahasa Indonesia yang terdapat di Twitter dengan jumlah *dataset* sebanyak 15.000. Objek yang akan diambil yaitu analisis sentimen terhadap uji materi (*judicial review*) dan amar putusan MK atas UU KPK Pasal 29 huruf e dan Pasal 34 yang sedang marak diperbincangkan oleh warganet, khususnya di twitter dengan *queries* = 'kpk', '#kpk', 'komisi pemberantasan korupsi', '#uukpk', '#kpkrakusjabatan'.

- b. *Preprocessing* menggunakan tahapan *Casefolding*, Normalisasi 1, *Tokenizing*, *Filtering (Stopword Removal)*, *Stemming*, Normalisasi 2.
- c. Pelabelan sentimen yang digunakan untuk membangun model *deep learning* yaitu secara manual dan menggunakan tiga label positif, negatif, dan netral.
- d. Model yang digunakan adalah LSTM dan BiLSTM dengan integrasi model menggunakan CNN.
- e. Pembobotan menggunakan pendekatan Word2Vec.
- f. Pembagian dataset training, validasi, dan tesing masing-masing 80%:10%:10%.

#### **1.4. Tujuan Penelitian**

Tujuan dari penelitian ini adalah:

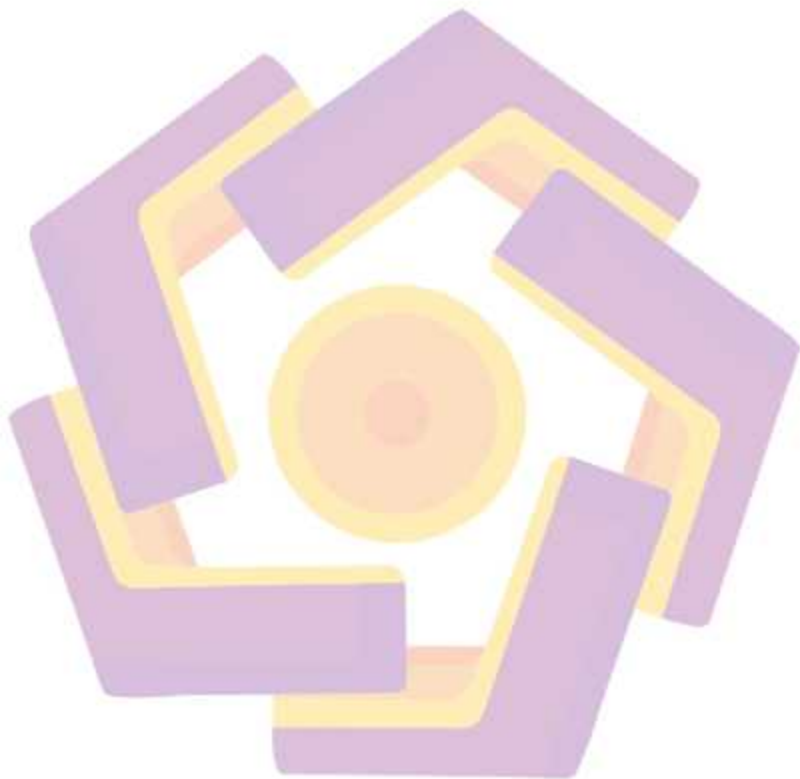
- a. Melakukan pengklasifikasian opini publik menggunakan integrasi model LSTM+CNN, BiLSTM+CNN, CNN+LSTM, dan CNN+LSTM berdasarkan preferensi pemodelan data teks.
- b. Mengetahui tingkat akurasi model eksperimen menggunakan integrasi model LSTM+CNN, BiLSTM+CNN, CNN+LSTM, dan CNN+LSTM dalam preferensi pemodelan data teks.

#### **1.5. Manfaat Penelitian**

Manfaat yang dapat diberikan oleh penelitian ini sebagai berikut:

- a. Memberikan model alternatif yang dapat melakukan klasifikasi opini publik berdasarkan cuitan di media sosial Twitter dengan menggunakan arsitektur *deep learning*.

- b. Dapat menjadi rekomendasi bagi pemerintah untuk mendapatkan pandangan publik, serta sebagai acuan dalam proses pengambilan keputusan sesuai kebutuhan.



## BAB II

### TINJAUAN PUSTAKA

#### 2.1. Tinjauan Pustaka

Pada penelitian terkait sentiment analisis mengenai topik politik pemerintahan atas pemberlakuan kebijakan baru / rencana / perubahan / penerapan Undang-Undang pernah dilakukan oleh para peneliti terdahulu. Pada penelitian yang diterbitkan pada tahun 2023 yaitu diantaranya oleh (Ashari dkk., 2023) pada penelitiannya mengklasifikasikan sentimen opini tentang pemindahan kebijakan kendaraan listrik di Indonesia menggunakan algoritma SVM, *Logistic Regression*, *Random Forest*, *Gradient Boosting*, CNN, dan RNN dengan total data yang digunakan adalah 7102 *tweet* dengan sampel *tweet* sebanyak 2847 dan hasil penelitiannya menunjukkan bahwa sebanyak 1586 *tweet* (55,71%) merespon positif dan 1261 (44,29%) merespon negative, SVM adalah model terbaik dengan akurasi 75,08%. Lebih lanjut pada penelitian yang dilakukan oleh (Gata dkk., 2023) mengklasifikasikan sentimen opini tentang *Omnibus Law* menggunakan algoritma NB, SVM dan K-NN dengan *tools rapidminer* dan hasil penelitiannya menunjukkan bahwa akurasi tertinggi dihasilkan oleh model algoritma SVM dengan sample *bootstrapping* dengan AUC 0,948 dan akurasi 85,88. Sedangkan pada penelitian yang dilakukan oleh (Amada dkk., 2023) mengklasifikasikan sentimen opini tentang Permenkominfo No.5 Tahun 2020 menggunakan algoritma SVM, NB dan K-NN dan hasil penelitiannya menunjukkan bahwa akurasi SVM 83%, NB 78%, dan K-NN 83%. Selain itu pada penelitian yang dilakukan oleh (Nurhusen dkk., 2023)



mengklasifikasikan sentimen opini tentang kebijakan kenaikan BBM menggunakan *logistic Regression* dibantu dengan *confusion matrix* untuk perhitungan evaluasi dan hasil penelitiannya menunjukkan bahwa sentimen positif mendapatkan nilai akurasi sebesar 38%, sentimen negatif sebesar 80% dan akurasi metode *regresi logistik* memperoleh 77%.

Pada penelitian yang diterbitkan pada tahun 2022 yaitu diantaranya oleh (Reiki dkk., 2022) mengklasifikasikan sentimen opini tentang UU No. 3 Tahun 2022 tentang pemindahan Ibu Kota Negara (IKN) menggunakan algoritma SVM dengan pembobotan kata TF-IDF, TF-RF, dan TF-BIN.ICF dan hasil penelitiannya menunjukkan bahwa pembobotan TF-IDF masih yang terbaik dibandingkan bobot term TF-BIN.ICF dan TF-RF, dengan skor akurasi 88,0% selisih 1,3% dengan pembobotan term TF-BIN.ICF. Lebih lanjut pada penelitian yang dilakukan oleh (Sandag dkk., 2022) mengklasifikasikan sentimen opini tentang kebijakan PPKM Covid-19 menggunakan algoritma LSTM, GRU, BiLSTM dan RNN dan hasil penelitiannya menunjukkan bahwa kinerja akurasi yang diperoleh dari keempat model tersebut antara lain LSTM 90%, GRU 89%, BiLSTM 90% dan RNN 85%, perbandingan hasil akurasi terbaik diperoleh dari model LSTM dan BiLSTM. Sedangkan pada penelitian yang dilakukan oleh (Andreas & Prabowo, 2022) mengklasifikasikan sentimen opini publik dalam bahasa Indonesia terhadap UU Cipta Kerja menggunakan dataset sejumlah 160 dari Twitter dengan metode *Naïve Bayes* dan hasil penelitiannya menunjukkan bahwa akurasi yang didapatkan selama penelitian adalah 85% dengan rata-rata presisi, recall, dan f1-score sebesar 85% Penelitian serupa dengan sentimen opini tentang RUU Cipta Kerja (*Omnibus Law*)



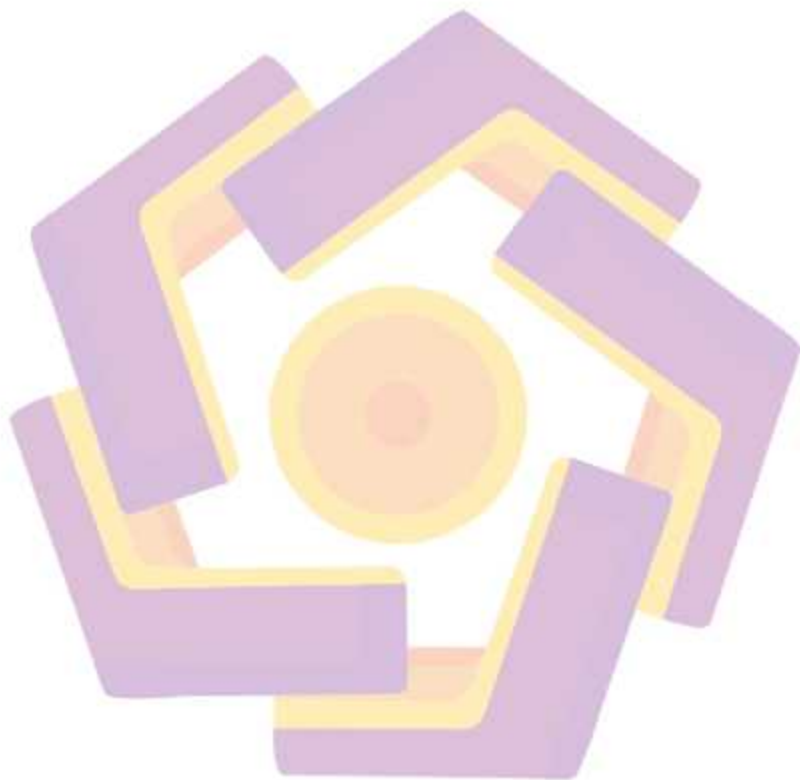
juga dilakukan oleh (Dauni dkk., 2022) menggunakan algoritma *convolutional neural network* yang merupakan salah satu metode *deep learning* dengan pembobotan kata Word2vec dan hasil penelitian menunjukkan bahwa hasil akurasi algoritma sebesar 84% dari 10 kali pengujian.

Pada penelitian yang diterbitkan pada tahun 2021 yaitu diantaranya oleh (Ahmad dkk., 2021) pada penelitiannya mengklasifikasikan ulasan untuk 8 topik kebijakan dalam *omnibus law* menggunakan algoritma SVM dan hasil penelitiannya menunjukkan bahwa dari percobaan didapatkan algoritma terbaik dengan tingkat akurasi 91,80% yaitu algoritma SVM dan topik terkait ketenagakerjaan merupakan topik yang paling banyak mendapat ulasan dan sentimen negatif dari masyarakat. Lebih lanjut pada penelitian yang dilakukan oleh (Sandryan dkk., 2021) mengklasifikasikan sentimen opini tentang Undang-Undang Cipta Kerja menggunakan algoritma *backpropagation* dengan pembobotan kata TF-IDF dan hasil penelitian menunjukkan bahwa akurasi sebesar 95%, precision sebesar 98%, recall sebesar 92,4%, dan f-measure sebesar 95,1%. Penelitian serupa dengan sentimen opini tentang RUU Cipta Kerja (*Omnibus Law*) juga dilakukan oleh (Nurdiansyah dkk., 2021) menggunakan algoritma *naïve bayes classifier* dengan pembobotan kata TF-IDF dan hasil penelitian menunjukkan bahwa nilai akurasi tertinggi sebesar 94% dengan data training yang digunakan sebesar 90%, data testing sebesar 10%. Selain itu pada penelitian yang dilakukan oleh (Augustia dkk., 2021) mengklasifikasikan sentimen opini tentang Omnibus Law menggunakan algoritma SVM dan NB dengan optimasi *Particle Swarm Optimization* (PSO) dan hasil penelitiannya menunjukkan bahwa memiliki

mendapatkan akurasi SVM 84,95% dan NB 87,53% dengan nilai *area under the curve* (AUC) 0,958 dan 0,754.

Pada penelitian yang diterbitkan pada tahun 2020 yaitu diantaranya oleh (Khurniawan & Ruldeviyani, 2020) pada penelitiannya mengklasifikasikan sentimen opini tentang Revisi UU Komisi Pemberantasan Korupsi (KPK) 2019 menggunakan algoritma SVM, *Decision Tree*, dan NB dengan total data yang digunakan adalah 7102 *tweet* dengan sampel *tweet* sebanyak 2847 dan hasil penelitiannya menunjukkan bahwa algoritma SVM memiliki performa dan akurasi tertinggi sebesar 81,70%, diikuti oleh Naive Bayes sebesar 80,90%, dan *Decision Tree* sebesar 74,55%. Lebih lanjut pada penelitian yang dilakukan oleh (Prastyo dkk., 2020) mengklasifikasikan sentimen opini tentang *Omnibus Law* menggunakan algoritma SVM dengan pembobotan kata TF-IDF 4 kernel, yaitu *Polynomial*, *Sigmoid*, *Linear*, dan *Radial Basis Function* (RBF) dan hasilnya menunjukkan bahwa fungsi kernel RBF pada SVM+TF-IDF menggunakan fitur 2000 mencapai kinerja terbaik dari semua pengujian fitur dalam akurasi, presisi, recall, dan *f-measure* dengan nilai 96,61%, 96,70%, 96,58%, 96,60%. Sedangkan pada penelitian yang dilakukan oleh (Helmiah dkk., 2020) mengklasifikasikan sentimen opini tentang RUU Cipta Kerja menggunakan dataset sejumlah 88820 dari Twitter dengan metode *Naïve Bayes Classifier* dan hasilnya menunjukkan bahwa dengan metode *Naïve Bayes* mendapat tingkat akurasi model sebesar 87.1%. Selain itu pada penelitian yang dilakukan oleh (Astiningrum & Batubulan, 2020) mengklasifikasikan sentimen opini tentang RKUHP menggunakan dataset sejumlah 1500 dari Twitter dengan metode SVM menggunakan pembobotan kata

TF-IDF dan hasil penelitiannya menunjukkan bahwa akurasi klasifikasi algoritma SVM menggunakan 70, 80, dan 90 persen data *training* dan hasil terbaiknya adalah 72%, 74% dan 81%.



## 2.2. Keaslian Penelitian

Tabel 2.1. Matriks *literature review* dan posisi penelitian

No	Judul	Peneliti, Media Publikasi, dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
1.	<i>Analyzing Public Opinion on Electrical Vehicles in Indonesia Using Sentiment Analysis and Topic Modeling</i>	(Ashari dkk., 2023) Publisher: IEEE	Untuk implementasi model analisis sentimen melalui media sosial Twitter.	SVM adalah model terbaik dengan akurasi 75,08%	Menggunakan metode <i>Machine Learning</i> selain SVM	Tindak Lanjut: Menggunakan <i>Deep Learning</i> (LSTM dan BiLSTM)
2.	<i>Twitter in analysis of policy sentiments of the omnibus law work creative design</i>	(Guta dkk., 2023) Publisher: AIP Conference Proceedings	Untuk menilai komentar tentang sentimen positif atau negatif omnibuslaw	Akurasi tertinggi dihasilkan oleh model algoritma SVM dengan sample bootstrapping dengan AUC 0,948 dan akurasi 85,88	Menggunakan <i>executable RapidMiner</i>	Tindak Lanjut: Menggunakan <i>executable google colaboratory</i>
3.	<i>Comparison of Term Weighting Methods in Sentiment Analysis of the New State Capital of Indonesia with the SVM Method</i>	(Reiki dkk., 2022) Publisher: IJoICT	Penelitian ini membandingkan pembobotan term tf-idf, tf-rf, dan tf-bin.icf dengan metode klasifikasi SVM pada topik baru Ibukota Negara Indonesia.	Hasil pembobotan tf-idf masih yang terbaik dibandingkan bobot term tf-bin.icf dan tf-rf, dengan skor akurasi 88,0% selisih 1,3% dengan pembobotan term tf-bin.icf.	Penelitian lebih lanjut juga dapat dilakukan untuk topik analisis sentimen lainnya dan dengan kategori multi kelas.	Tindak Lanjut: Pembagian data latih dan uji sebanyak 4 kali yaitu 90%:10%, 80%:20%, 70%:30%, dan 60%:40%

Tabel 2.1. (Lanjutan)

No	Judul	Peneliti, Media Publikasi, dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
4.	<i>Sentiment Analysis of Government Policy Regarding PPKM on Twitter Using LSTM</i>	(Sandag dkk., 2022) Publisher: IEEE	Untuk analisis sentimen pada isu PPKM Indonesia dengan menggunakan model deep learning yaitu LSTM	Akurasi yang diperoleh dari keempat model tersebut antara lain LSTM 90%, GRU 80%, BiLSTM 90% dan RNN 85%.	Menggunakan satu metode yaitu LSTM	Tindak Lanjut: Membandingkan 2 metode yaitu LSTM dan BiLSTM
5.	<i>Twitter User Sentiment Analysis For RUU Omnibus Law Using Convolutional Neural Network</i>	(Dauni dkk., 2022) Publisher: IEEE	Untuk membuat analisis opini dari pengguna media sosial Twitter tentang topik RUU Omnibuslaw menggunakan metode CNN yang merupakan salah satu metode Deep Learning	Hasil akurasi algoritma sebesar 84% dari 10 kali pengujian	Berdasarkan 2.820 data tweet	Tindak Lanjut: Menggunakan dataset lebih dari 2.820
6.	<i>Sentiment Analysis of the New Indonesian Government Policy (Omnibus Law) on Social Media Twitter</i>	(Sukma dkk., 2020) Publisher: IEEE	Untuk mengetahui tingkat kualitas keputusan dan tingkat penerimaan, digunakan metodologi SVM dan analisis sentimen untuk mengklasifikasikan ulasan untuk 8 topik kebijakan omnibus law	Algoritma terbaik dengan tingkat akurasi 91,80% yaitu algoritma SVM	Dalam metode penelitian yang digunakan tidak dapat mengidentifikasi bahasa daerah dan bahasa gaul (slang)	Tindak Lanjut: Normalisasi menggunakan kamus yang dapat mengubah kata-kata atau bahasa slang menjadi kata baku



Tabel 2.1. (Lanjutan)

No	Judul	Peneliti, Media Publikasi, dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
7.	<i>Twitter Sentiment Analysis: Case Study on the Revision of the Indonesia's Corruption Eradication Commission (KPK) Law 2019</i>	(Khurniawan & Ruldeviyani, 2020)  Publisher: IEEE	Untuk menganalisis sentimen opini publik terhadap Revisi UU KPK 2019	Hasilnya menunjukkan algoritma SVM memiliki performa dan akurasi tertinggi sebesar 81,70%, diikuti oleh <i>Naive Bayes</i> sebesar 80,90%, dan <i>Decision Tree</i> sebesar 74,55%	Untuk menyempurnakan penelitian ini di masa mendatang, kami menyarankan peneliti selanjutnya untuk mengolah dan menterjemahkan kosa kata ke dalam bahasa formal yang baku.	Tindak Lanjut: Menggunakan kamus menterjemahkan kosa kata ke dalam bahasa formal yang baku.
8.	<i>Indonesian Sentiment Analysis: An Experimental Study of Four Kernel Functions on SVM Algorithm with TF-IDF</i>	(Prastyo dkk., 2020)  Publisher: IEEE	Untuk membandingkan empat fungsi kernel pada SVM dengan TF-IDF	Fungsi kernel RBF pada SVM+TF-IDF menggunakan fitur 2000 mencapai kinerja terbaik dari semua pengujian fitur dalam akurasi, presisi, recall, dan <i>f-measure</i> dengan nilai 96.61%, 96.70%, 96.58%, 96.60%	Menggunakan satu pembobotan kata TF-IDF	Tindak Lanjut: Membandingkan 2 pembobotan yaitu TF-IDF dan Word2Vec



### 2.3. Landasan Teori

Terdapat beberapa landasan teori yang dibutuhkan dalam penelitian ini, mulai dari landasan teori untuk preferensi sentimen analisis, penambangan data (*text mining*), klasifikasi, model LSTM, BiLSTM, dan CNN.

#### 2.3.1. Sentimen Analisis

Analisis sentimen merupakan sebuah studi komputasi tentang opini, evaluasi, sikap, subjektivitas dan pandangan yang diungkapkan dalam teks (Chaturvedi dkk., 2017). Analisis sentimen adalah bidang studi yang menganalisis pendapat orang, opini, evaluasi, penilaian, sikap, dan emosi terhadap entitas, oleh karenanya analisis sentimen menjadi populer dalam proses data sosial seperti Twitter dan media kolaboratif online lainnya (Hussein, 2018; Joshi dkk., 2017). Objek analisis sentimen berupa produk atau layanan yang ulasannya telah dipublikasikan di internet. Analisis sentimen adalah bidang yang menarik untuk dikembangkan dalam dunia digital sebab pada saat ini masyarakat secara umum mengutarakan perasaan melalui dunia maya. Menganalisis sentimen masyarakat dapat menghasilkan informasi-informasi baru yang dapat diolah sehingga menghasilkan pengetahuan baru.

#### 2.3.2. Penambangan Teks (*Text Mining*)

*Text mining* merupakan proses penggalian informasi terbaru yang berfungsi untuk menganalisis keterhubungan antara suatu dokumen dengan dokumen lain (Gata & Purnomo, 2016). *Text mining* (penambangan teks) merupakan penggalian

data yang bersifat tekstual. Penggalan teks artinya teks dianalisa dalam bentuk elektronik. *Text mining* merupakan proses analisa kata atau data lain yang bukan numerik. *Text mining* bisa berupa suatu analisa untuk memperoleh arti dan membuat rangkuman melalui pemrosesan teks, menganalisa setiap teks melalui kata kunci, membagi tema dan dokumen serta mendapatkan informasi dari kata-kata penting secara semantik dan kalimat yang relevan. *Text mining* merupakan teknik yang biasanya digunakan untuk menyelesaikan permasalahan classification, clustering, information retrieval dan information extraction.

Penambangan teks merupakan bagian dari penambangan data, yang dapat didefinisikan sebagai proses mengekstraksi pengetahuan dari teks atau dokumen (Jo, 2019). Menurut (Kantardzic, 2020) menjelaskan bahwa bentuk paling dasar dari informasi yaitu teks dan hampir 80% dari informasi perusahaan termuat dalam dokumen teks, sehingga tantangan terkait penambangan teks menjadi isu saat ini. Penambangan teks dapat diyakini memiliki potensi komersial yang lebih tinggi daripada penambangan data tradisional dengan data yang terstruktur.

Penambangan teks dapat mengekstraksi bagian penting dalam dokumen teks, mengelompokkan dokumen, dan membuat ringkasan pada dokumen yang tidak mungkin dilakukan secara manual oleh manusia (Jo, 2019). Beberapa tujuan dari penambangan teks, dijelaskan oleh (Kantardzic, 2020) adalah memberikan gambaran umum terkait isi topik apa yang ada pada dokumen-dokumen, meningkatkan efisiensi dan efektivitas pencarian informasi pada dokumen, dan mendeteksi informasi atau dokumen yang duplikat. Menurut (Jo, 2019), perbedaan antara penambangan teks dengan sistem temu balik yaitu dari hasil output.

Penambangan teks digunakan sebagai pengetahuan yang diperlukan dan digunakan secara langsung untuk membuat keputusan, sedangkan sistem temu balik adalah beberapa item data dari pengambilan informasi.

Sumber data pada penambangan teks dapat berupa teks pada suatu dokumen berita, jurnal, maupun surat (Jo, 2019). Pada sebuah paragraf pada suatu dokumen dapat berupa kombinasi dari kalimat yang diurutkan untuk menjaga konsistensi pada sub topik tertentu dengan tujuan untuk mencari kata-kata yang dapat mewakili dokumen-dokumen tersebut (Amantillah dkk., 2019). Sehingga hasil dari penambangan teks tersebut didapatkan suatu pengetahuan yang digunakan pada kebijakan yang tepat pada suatu perusahaan. Penambangan teks dapat dilakukan dengan klasifikasi atau melihat jumlah kata yang sering muncul atau word cloud (Mihuandayani dkk., 2018).

#### **2.4. Pembobotan**

Seleksi fitur merupakan proses untuk memilih term (kata/istilah) yang digunakan pada proses training dan pada penelitian ini menggunakan pendekatan Word2Vec.

*Word2vec* adalah salah satu metode *embedding word* yang berguna untuk merepresentasikan kata menjadi sebuah vektor (Nurrohmat & Azhari, 2019). 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.

## 2.5. Klasifikasi

Klasifikasi adalah proses menemukan model dari sebuah data. Tujuan dari klasifikasi adalah untuk mengambil suatu keputusan dengan memprediksi suatu kasus berdasarkan hasil klasifikasi yang diperoleh. Dalam proses pengklasifikasian terdapat 3 proses yang dilakukan yaitu :

- 1) Proses training. Pada proses ini dilakukan training set yang sudah diketahui label-labelnya untuk membangun model.
- 2) Proses testing. Proses ini untuk mengetahui keakuratan model yang dibangun pada proses training, umumnya digunakan data yang disebut data test set untuk memprediksi label.
- 3) Proses validasi. Pada proses ini digunakan untuk proses validasi model dan mencegah *overfitting*.

### 2.5.1. *Convolutional Neural Network (CNN)*

*CNN* merupakan jaringan saraf tiruan yang memberi umpan balik dimana jaringan saraf mempertahankan struktur hierarki dengan mempelajari representasi fitur internal dan menggeneralisasi fitur-fitur dalam masalah gambar secara umum seperti pengenalan objek dan masalah *computer vision*. Penggunaannya tidak terbatas pada gambar, bisa juga digunakan untuk mencapai hasil dalam masalah pemrosesan bahasa alami dan pengenalan ucapan.



### 2.5.2. Long Short-Term Memory (LSTM)

LSTM pertama kali dikenalkan oleh Hochreiter dan Schmidhuber dan saat ini menjadi arsitektur RNN yang populer digunakan (Suyanto dkk., 2019). LSTM dapat melakukan pelatihan dan mengatasi masalah vanishing gradient yang menjadi kesulitan pada *Recurrent Neural Network* (RNN). LSTM dapat mempelajari pola panjang dari data berurut karena mencegah situasi *vanishing gradient* (Putra, 2019). Tetapi, LSTM tetap memiliki prinsip yang sama dengan RNN dan yang membedakan dengan RNN yaitu isi *cellnya*. *Recurrent Neural Network* (RNN) sederhana karena dengan *cell* yang hanya berisi 1 layer *neuron* dengan fungsi aktivasi *tanh*.

Ide dasar dari LSTM adalah adanya jalur yang menghubungkan antara *cell state* ( $C_{t-1}$ ) sebelumnya dengan *cell state* yang sekarang ( $C_t$ ). Dengan jalur tersebut, suatu informasi pada *cell state* dapat dengan mudah diteruskan ke *cell state* berikutnya dengan beberapa modifikasi yang diperlukan. Nilai *cell state* merupakan vektor yang dirancang untuk menyimpan informasi tentang konteks dari suatu data. Langkah pertama dalam LSTM adalah menentukan informasi yang akan dibuang dari *cell state* ( $C_{t-1}$ ) menggunakan fungsi sigmoid yang disebut sebagai *forget gate* ( $f_t$ ). Nilai nol menandakan bahwa informasi akan dibuang sedangkan satu berarti informasi diteruskan.

$$f_t = \sigma(W_f \cdot [S_{t-1}, X_t] + B_f) \dots \dots \dots \text{Persamaan (2)}$$

Dimana:

$f_t$  = forget gate

$\sigma$  = fungsi sigmoid



- $W_f$  = nilai weight untuk forget gate  
 $S_{t-1}$  = nilai output sebelum order ke t  
 $X_t$  = nilai input pada order ke t  
 $B_f$  = nilai bias pada forget gate

Langkah berikutnya adalah menentukan informasi yang akan ditambahkan ke cell state ( $C_t$ ). Langkah ini memproses hasil penggabungan dari  $s_{t-1}$  dan  $x_t$  menggunakan dua fungsi, yaitu fungsi sigmoid sebagai input gate dan fungsi tanh sebagai intermediate gate. Hasil dari kedua fungsi tersebut dikalikan untuk mendapatkan informasi yang akan ditambahkan pada cell state ( $C_t$ ). Persamaan matematika dari kedua fungsi terdapat pada persamaan (2).

$$i_t = \sigma(W_i [S_{t-1}, X_t] + B_i) \dots \dots \dots \text{Persamaan (3)}$$

$$C_t = \tanh(W_c [S_{t-1}, X_t] + B_c) \dots \dots \dots \text{Persamaan (4)}$$

$$C_t = F_t * C_{t-1} + i_t * C_t \dots \dots \dots \text{Persamaan (5)}$$

Dimana:

- $i_t$  = *input gate*  
 $W_i$  = nilai *weight* untuk *input gate*  
 $S_{t-1}$  = nilai *output* sebelum order ke t  
 $X_t$  = nilai *input* pada order ke t  
 $B_i$  = nilai bias pada *input gate*  
 $C_t$  = *cell state*  
 $\tanh$  = fungsi tanh

- $W_c$  = nilai *weight* untuk *cell state*  
 $B_c$  = nilai bias untuk *cell state*  
 $C_{t-1}$  = *cell state* sebelum order ke t

Setelah itu, tambahkan dengan *output* dari *forget gate* pada langkah pertama. Langkah terakhir adalah menentukan output dari unit LSTM. Untuk menghasilkan *output*, perlu menghitung sigmoid dari gabungan  $s_{t-1}$  dan  $x_t$  yang disebut sebagai *output gate*. *Output gate* ini menentukan seberapa besar nilai dari *cell state* akan dihasilkan pada  $s_t$ . kemudian hitung nilai fungsi tanh dari *cell state* ( $C_t$ ) dan kalikan dengan nilai dari *output gate*. Hasil perkaliannya tersebut menjadi *output* dari unit LSTM yang terdapat pada persamaan (5)

$$O_t = \sigma(w_o \cdot [s_{t-1}, x_t] + b_o) \dots \dots \dots \text{Persamaan (6)}$$

$$S_t = w_o * \tanh(C_t) \dots \dots \dots \text{Persamaan (7)}$$

Dimana:

- $O_t$  = *output gate*  
 $\sigma$  = fungsi aktivasi sigmoid  
 $w_o$  = nilai *weight* untuk *output gate*  
 $S_{t-1}$  = nilai *output* sebelum order ke t  
 $X_t$  = nilai *input* pada order ke t  
 $b_o$  = nilai bias pada *output gate*  
 $S_t$  = nilai *output* order t  
 $\tanh$  = fungsi tanh  
 $C_t$  = *cell state*

### 2.5.3. Bidirectional Long Short-Term Memory (BiLSTM)

BiLSTM adalah variasi dari *recurrent neural network* (RNN) yang dikembangkan oleh Schuster dan Paliwal untuk melatih model *neural network* menggunakan *sequential* data dengan input masa lalu dan masa depan (Schuster & Paliwal, 1997). *Bidirectional LSTM* merupakan jaringan syaraf dari *long short-term memory* (LSTM) yang terdiri dari dua lapisan jaringan saraf LSTM. Lapisan LSTM lanjutan untuk memodelkan konteks sebelumnya dan lapisan LSTM terbelakang untuk memodelkan masing-masing berikutnya (Ertugrul & Karagoz, 2018). Pada kasus klasifikasi, *bidirectional lstm* lebih akurat dan lebih cepat dibandingkan LSTM (Graves & Schmidhuber, 2005). *Bidirectional LSTM* memanfaatkan konteks sebelumnya dan konteks setelahnya dengan memproses data dari dua arah dengan *hidden layer* terpisah (Maimaiti dkk., 2017). *Bidirectional LSTM* dikatakan tumpukan dari LSTM. *Forward layer* digunakan untuk konteks sebelumnya dan *backward layer* digunakan untuk konteks setelahnya (Lample dkk., 2016). Keluaran dari kombinasi dua arah *hidden layer* terdapat pada persamaan (8).

$$y_t = w_{\vec{h}y} \vec{h}_t + \bar{w}_{hy} \bar{h}_t \dots \dots \dots \text{Persamaan (8)}$$

Dimana:

- $y_t$  = *output gate* LSTM dua arah
- $w_{\vec{h}y}$  = nilai bobot *output gate* LSTM maju
- $\vec{h}_t$  = nilai keluaran LSTM maju

$\bar{w}_{hy}$  = nilai bobot untuk output gate LSTM mundur

$h_{\bar{t}}$  = nilai keluaran LSTM mundur

#### 2.5.4. Integrasi LSTM dan BiLSTM dengan CNN

Model LSTM+CNN dan BiLSTM+CNN terdiri dari lapisan LSTM atau BiLSTM awal yang akan menerima *embeddings* kata untuk setiap token dalam judul sebagai input. Intuisi adalah dimana token outputnya akan menyimpan informasi tidak hanya dari token awal, tetapi juga token sebelumnya. Dengan kata lain, lapisan LSTM atau BiLSTM menghasilkan pengkodean baru untuk input asli. Output dari layer LSTM atau BiLSTM kemudian diumpankan ke dalam lapisan konvolusi yang diharapkan akan mengekstrak fitur lokal. Akhirnya Output lapisan konvolusi akan dikumpulkan ke dimensi yang lebih kecil dan pada akhirnya dikeluarkan sebagai label positif, negative, atau netral.

#### 2.5.5. Integrasi CNN dengan LSTM dan BiLSTM

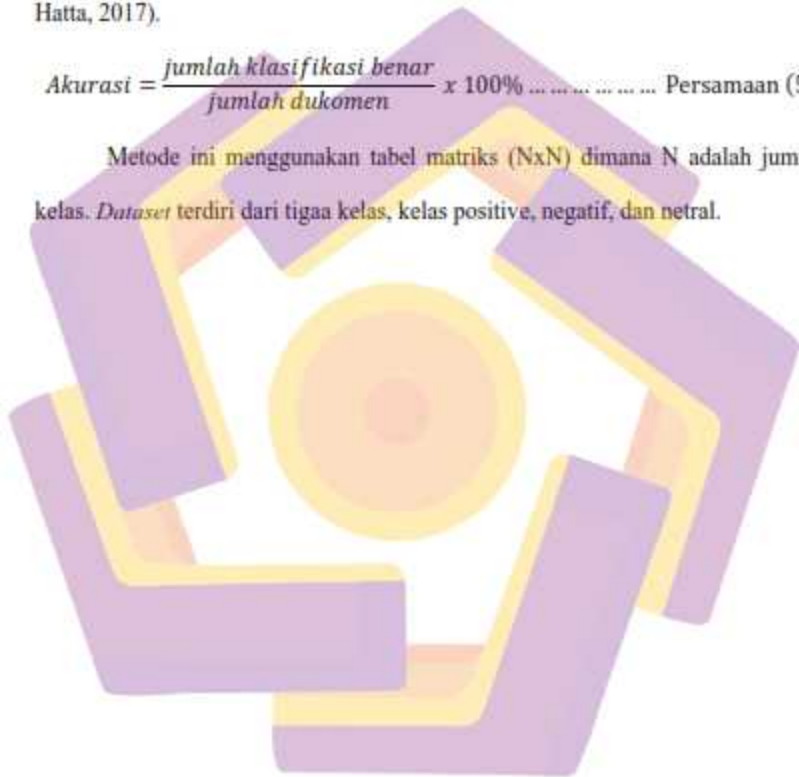
Model CNN+LSTM dan CNN+BiLSTM terdiri dari lapisan *Convolutional* awal yang akan menerima *embeddings* kata untuk setiap token dalam judul sebagai input. Output yang dihasilkan dari lapisan CNN akan dikumpulkan ke dimensi yang lebih kecil kemudian diumpankan ke dalam lapisan LSTM atau BiLSTM yang diharapkan akan mengekstrak fitur lokal. Pada akhirnya dikeluarkan sebagai label positif, negative, atau netral.

## 2.6. Confusion matrix

*Confusion matrix* merupakan metode untuk menghitung tingkat akurasi, dengan menghitung jumlah klasifikasi benar dan salah dari sebuah metode klasifikasi berbanding dengan data sesungguhnya atau klasifikasi target (Agus & Hatta, 2017).

$$\text{Akurasi} = \frac{\text{jumlah klasifikasi benar}}{\text{jumlah dokumen}} \times 100\% \dots \dots \dots \text{Persamaan (5)}$$

Metode ini menggunakan tabel matriks (NxN) dimana N adalah jumlah kelas. *Dataset* terdiri dari tiga kelas, kelas positive, negatif, dan netral.





## BAB III METODE PENELITIAN

### 3.1. Dataset

Data yang digunakan dalam penelitian ini adalah data yang didapatkan dari media sosial Twitter dengan jumlah dataset yang di *scrapping* sebanyak 15.000. Dataset tersebut merupakan data yang didapatkan dari media sosial Twitter, yang terdiri label positif dan negatif. Isi cuitan pada dataset yang digunakan, sudah dilakukan seleksi untuk memastikan bahwa akun pengguna tersebut adalah asli dengan mengacu pada:

- a. Melakukan postingan tweet seperti “kpk, #kpk, komisi pemberantasan korupsi, #uukpk, #kpkrakusjabatan ....” atau kata kunci lainnya yang relevan.
- b. Penggunaan tagar yang tidak jelas, seperti #korupsitowerbts, dan lainnya

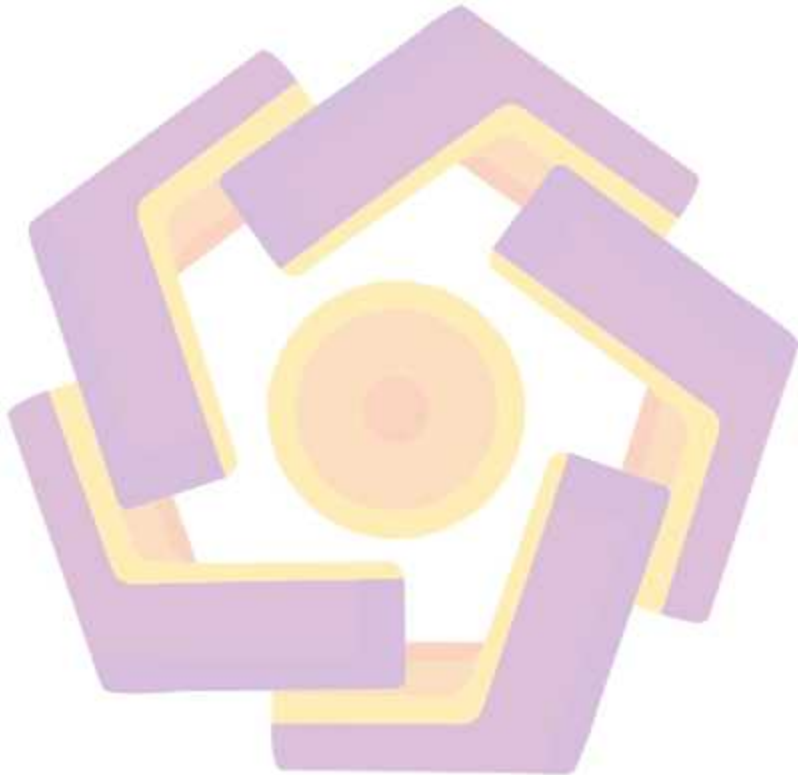
### 3.2. Preprocessing Data

Pada bagian ini akan dilakukan beberapa tahapan, yaitu menyeleksi tweet yang menggunakan bahasa Indonesia, mengubah teks menjadi huruf kecil atau *case folding*, menghapus link, fitur twitter seperti #, RT, @, angka dan simbol, normalisasi kata slang atau kata lainnya, *stemming* dan tokenisasi (Utami dkk., 2019).

Tahapan *preprocessing* meliputi : *Case Folding* → Normalisasi 1 → *Tokenizing* → *Filtering (Stopword Removal)* → *Stemming* → Normalisasi 2.

Tahapan tersebut dilakukan dengan bahasa pemrograman *Python* dengan menggunakan perangkat lunak *Google colab* dan bantuan sastrawi dan *Natural Language Toolkit (NLTK)*. *Case Folding* merupakan tahapan yang mengubah kata-kata pada teks menjadi huruf kecil. *Number, Link, and Punctuation Removal* merupakan tahapan untuk menghapus angka, URL atau link serta simbol-simbol yang tidak dibutuhkan. *Normalization 1* merupakan tahapan untuk memperbaiki kata menjadi kata normal. Normalisasi pada penelitian ini menggunakan kamus alay dari indo bermacam dimana kamus alay ini akan mengubah kata-kata atau bahasa slang menjadi kata baku. *Tokenizing* memisahkan setiap kata yang menyusun suatu dokumen. Umumnya setiap kata terpisahkan dengan kata yang lain oleh karakter spasi, sehingga proses tokenisasi mengandalkan karakter spasi pada dokumen untuk melakukan pemisahan kata (Kane dkk., 2016). Setelah melalui proses tokenisasi maka kalimat tersebut menjadi sekumpulan array yang setiap selnya berisi kata-kata yang ada pada kalimat tersebut. *Filtering (Stopword Removal)* pada tahap ini dilakukan dengan menghapus karakter khusus pada *dataset* atau kata yang tidak memiliki arti, adapun karakter khusus yang dihapus adalah tanda baca (poin (.), koma (,), tanda tanya, tanda seru, angka numerik dan karakter lainnya. *Stemming* merupakan tahapan untuk melakukan konversi kata menjadi kata dasar berdasarkan imbuhan (“komentar” menjadi “komentar”). *Normalization 2* pada tahapan ini sama dengan normalisasi 1 yaitu menyesuaikan kata yang tidak sesuai dengan KBBI. Untuk Normalisasi 2 ini menggunakan kamus *indonesian-word*. Jika masih terdapat kata yang tidak terdapat pada kamus, maka kata tersebut tetap akan dilakukan proses selanjutnya

dikarenakan bisa menjadi fitur unik pada analisis sentimen opini publik mengenai UU KPK. *Source Code* yang menggunakan dengan bahasa pemrograman Python yang digunakan pada tahapan preprocessing dapat dilihat pada Lampiran. Sampel perubahan data pada tahapan *preprocessing* data dapat dilihat pada Tabel 3.1.



Tabel 3.1. Sampel Tahapan *Preprocessing Data*

<i>Text_Clean</i>	<i>Normalization 1</i>	<i>Tokenizing</i>	<i>Filtering (Stopword Removal)</i>	<i>Stemming</i>	<i>Normalization 2</i>	<i>Text</i>	<i>Label</i>
yusufdumdum kok fitnah memang itu ada nya kpk perpanjangan jabatan saja rakyat tidak di libatkan mk se enak jidat meraka	yusufdumdum kok fitnah memang itu ada nya kpk perpanjangan jabatan saja rakyat tidak di libatkan mk se enak jidat meraka	['yusufdumdum', 'kok', 'fitnah', 'memang', 'itu', 'ada', 'nya', 'kpk', 'perpanjangan', 'jabatan', 'saja', 'rakyat', 'tidak', 'di', 'libatkan', 'mk', 'se', 'enak', 'jidat', 'meraka']	['yusufdumdum', 'kok', 'fitnah', 'memang', 'nya', 'kpk', 'perpanjangan', 'jabatan', 'rakyat', 'libatkan', 'mk', 'se', 'enak', 'jidat', 'meraka']	['yusufdumdum', 'kok', 'fitnah', 'memang', 'nya', 'kpk', 'panjang', 'jabat', 'rakyat', 'libat', 'mk', 'se', 'enak', 'jidat', 'raka']	['kok', 'fitnah', 'memang', 'kpk', 'panjang', 'jabat', 'rakyat', 'libat', 'enak', 'jidat', 'raka']	kok fitnah memang kpk panjang jabat rakyat libat enak jidat raka	negative
denny indrayana perpanjangan masa jabatan pimpinan kpk adalah politik httpstcoscmtodraz	denny indrayana perpanjangan masa jabatan pimpinan kpk adalah politik httpstcoscmtodraz	['denny', 'indrayana', 'perpanjangan', 'masa', 'jabatan', 'pimpinan', 'kpk', 'adalah', 'politik', 'httpstcoscmtodraz']	['denny', 'indrayana', 'perpanjangan', 'masa', 'jabatan', 'pimpinan', 'kpk', 'politik', 'httpstcoscmtodraz']	['denny', 'indrayana', 'panjang', 'masa', 'jabat', 'pimpin', 'kpk', 'politik', 'httpstcoscmtodraz']	['panjang', 'masa', 'jabat', 'pimpin', 'kpk', 'politik']	panjang masa jabat pimpin kpk politik	neutral
fahrihamzah merenungi keputusan revisi uu kpk	fahrihamzah merenungi keputusan revisi uu kpk	['fahrihamzah', 'merenungi', 'keputusan', 'revisi', 'uu', 'kpk']	['fahrihamzah', 'merenungi', 'keputusan', 'revisi', 'uu', 'kpk']	['fahrihamzah', 'renung', 'putus', 'revisi', 'uu', 'kpk']	['renung', 'putus', 'revisi', 'kpk']	renung putus revisi kpk	negative
kpkri politik cerdas berintegritas	kpkri politik cerdas berintegritas	['kpkri', 'politik', 'cerdas', 'berintegritas']	['kpkri', 'politik', 'cerdas', 'berintegritas']	['kpkri', 'politik', 'cerdas', 'integritas']	['politik', 'cerdas', 'integritas']	politik cerdas integritas	positive

### 3.3. *Splitting data*

*Splitting data* digunakan untuk memisahkan data awal menjadi data latih dan data uji. Data latih merupakan data yang akan digunakan untuk melakukan pelatihan data, sedangkan data uji digunakan sebagai pengujian terhadap model eksperimen yang dilatih dengan menggunakan LSTM+CNN, BiLSTM+CNN, CNN+LSTM, dan CNN+LSTM. Pada penelitian ini *splitting data* dilakukan sebanyak 1 kali yaitu Training, Validasi, dan Tesing masing-masing 80%:10%:10%.

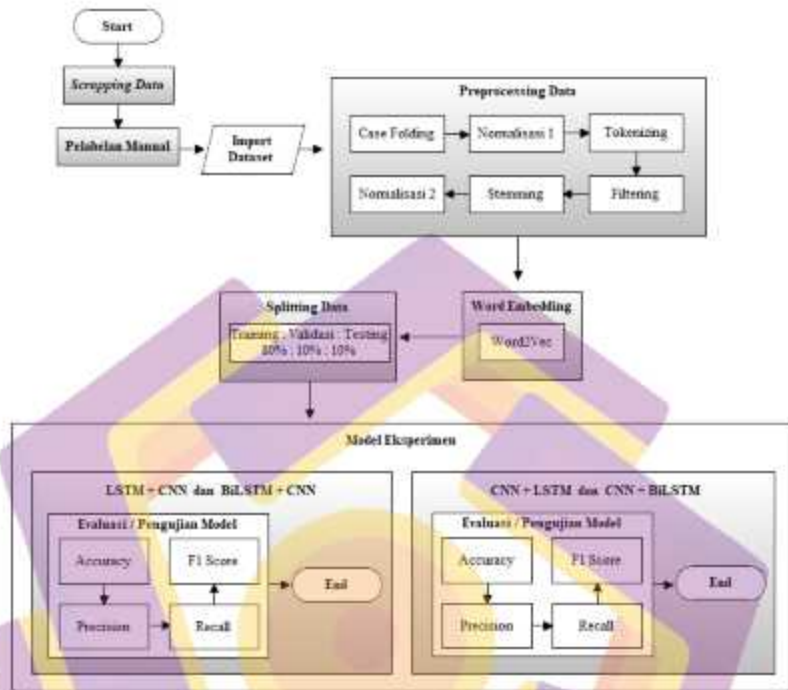
### 3.4. Evaluasi Model

Evaluasi model dapat dilihat dari akurasi dan *f-score/f-measure*. Akurasi adalah ukuran kinerja paling intuitif dan ini hanya rasio pengamatan yang diprediksi dengan benar terhadap pengamatan total, Sedangkan *f-measure* adalah salah satu perhitungan evaluasi yang menggabungkan nilai *persisi* dan juga *recall* dalam keadaan tertentu.

### 3.5. Alur Penelitian

Alur penelitian ini dapat dilihat pada Gambar 1, dimulai dari pengumpulan data, *preprocessing* data hingga pelatihan model dan pengujian model sehingga didapatkan hasil akurasi.





Gambar 3.1. Alur Penelitian

Alur penelitian ini dimulai dengan *import dataset*, lalu akan dilakukan *preprocessing data* seperti *case folding*, menghapus angka, link, hingga pembobotan kata *Word2Vec*. Proses pemisahan label dilakukan untuk memperkecil kelas pada label, sehingga menjadi 2 label yaitu positif dan negatif. Proses pemisahan data dilakukan sebanyak 1 kali yaitu Training, Validasi, dan Tesing masing-masing 80%:10%:10% proporsi itu dipilih karena untuk mengetahui akurasi terbaik dari setiap rasio. Terakhir, proses pelatihan dan pengujian model eksperimen untuk mendapatkan akurasi.

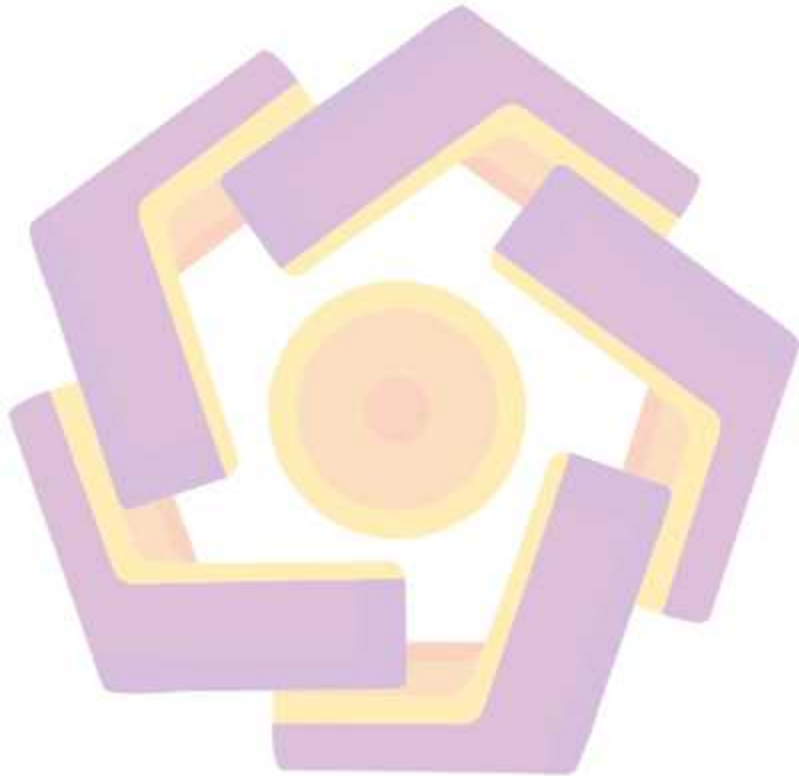
### 3.6. *Wordcloud*

Menurut (Qeis, 2015) *wordcloud* merupakan sebuah sistem yang dapat menghasilkan visualisasi dari setiap kata-kata dalam sebuah kalimat dengan memberikan sebuah penekanan pada frekuensi kemunculan kata terkait dalam wacana yang tertulis. Penggunaan *wordcloud* dapat memudahkan pembaca atau peneliti dikarenakan mampu memberikan gambaran dengan cepat mengenai isi dari suatu kalimat.

Pradana, (2020) menambahkan bahwa *wordcloud* populer dalam dunia *Text Mining* dikarenakan mudah difahami, dengan menggunakan *wordcloud* gambaran frekuensi kata dapat divisualisasikan dengan cepat dalam bentuk yang menarik namun informatif. Ukuran *text* pada gambar di *wordcloud* tergantung dengan frekuensi data, semakin banyak frekuensi kata yang digunakan, maka semakin besar ukuran kata yang divisualisasikan di *wordcloud*, begitu juga sebaliknya semakin sedikit frekuensi kata yang digunakan maka semakin kecil ukuran kata yang divisualisasikan di *wordcloud*. Tampilan *wordcloud* dapat dilihat pada gambar 3.2.



evaluasi performansi klasifikasi. Data latih digunakan untuk melatih model agar representasi terhadap data, sedangkan data uji digunakan untuk mengevaluasi model tersebut. Hasil keluaran yang diharapkan adalah akurasi setiap model eksperimen.



## BAB IV

### HASIL PENELITIAN DAN PEMBAHASAN

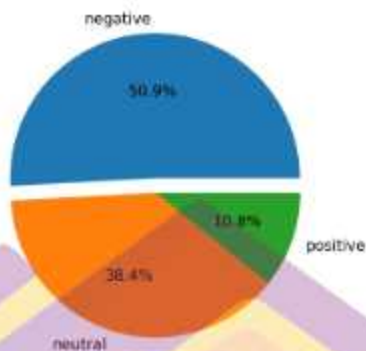
Proses pengujian dilakukan dengan melihat perbandingan sentiment kelas positif, negative, dan netral. Selanjutnya melihat performa model eksperimen LSTM+CNN, BiLSTM+CNN, CNN+LSTM, dan CNN+LSTM menggunakan pembobotan kata Word2Vec dalam melakukan klasifikasi. Pada proses ini dilakukan *split validation* untuk mengetahui akurasi terbaik dari setiap rasio data training dan data testing yang berbeda. Jumlah total data adalah 11.448 data dari data awal sejumlah 15.000 data.

#### 4.1. Perbandingan Sentimen

Perbandingan sentimen kelas positif, negatif dan netral. Sentimen opini publik terhadap UU KPK di media sosial Twitter menggunakan data testing sebesar 11.448 data. Hasil klasifikasi menunjukkan bahwa dominasi sentimen terbanyak ada pada sentimen kelas negatif yaitu dengan persentase tertinggi sebesar 50,90%, sedangkan sentimen kelas positif dengan persentase 10,08% dan untuk kelas netral 38,04%. Kelas negatif yang memiliki persentase lebih tinggi menandakan bahwa opini publik memiliki sentimen negatif yang cenderung tinggi terhadap UU KPK.



Sentiment Polarity pada Data



Gambar 4.1. Diagram Pie Perbandingan Sentimen

#### 4.2. Perbandingan *Wordcloud*

Analisis *wordcloud* dilakukan untuk mengetahui kata yang sering diungkapkan tentang UU KPK. *Wordcloud* digunakan untuk melihat gambaran dari kata yang sering muncul dalam keseluruhan data *tweet*. Pada penelitian ini akan ditambahkan *wordcloud* dari hasil klasifikasi negative, positif dan netral. *Wordcloud* pada sentimen negative ditunjukkan pada Gambar 4.2.

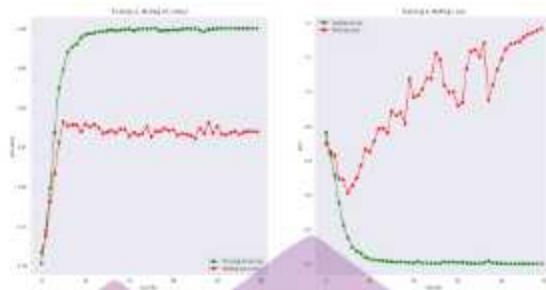




kpk, pimpin, jabat yang sering muncul pada ketiga sentimen tersebut. Hal ini menunjukkan pihak yang paling sering muncul pada *tweet* tentang UU KPK. Sedangkan perbedaan dari ketiga sentimen tersebut adalah kata-kata yang memiliki makna berseberangan dan bertolak belakang. Hal ini menunjukkan sudut pandang opini dari pengguna Twitter terhadap UU KPK.

#### 4.3. Pengujian Model

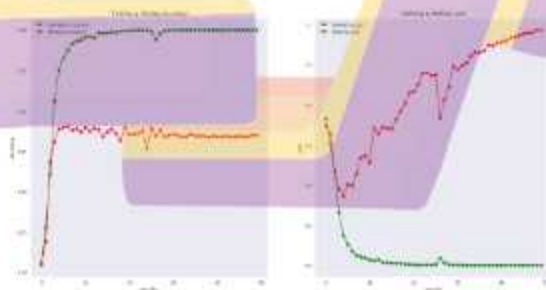
Pada tahapan ini dilakukan *training model* yang bertujuan untuk menguji model pada data *training* dengan parameter pada setiap model yang telah dibuat. Berdasarkan pengujian parameter yang telah dicoba, didapatkan model dengan nilai parameter terbaik adalah model CNN+BiLSTM dengan parameter Word2Vec dan *split validation* Training, Validasi, dan Tesing masing-masing 80%:10%:10%. Pengujian model menghasilkan nilai akurasi dan *loss* pada data *training* dan validasi. Proses *training* menggunakan jumlah *epoch* sebanyak 50. Dengan parameter yang sudah ditentukan akan dilihat berapa akurasi dari data *training* dan melihat nilai *loss* terendah. Model akan menyimpan *epoch* yang optimal pada nilai *loss* yang terendah selama proses *epoch* berlangsung. Berikut adalah sampel hasil pengujian model dapat dilihat pada Gambar 4.5 – Gambar 4.8.



Gambar 4.5. Sampel Pengujian Model LSTM+CNN

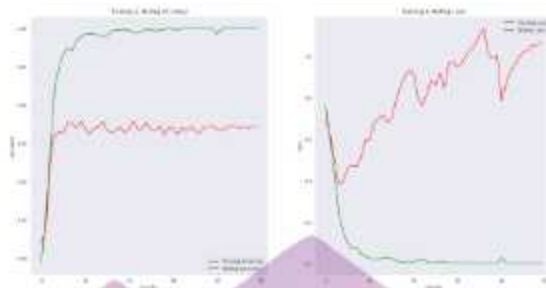


Gambar 4.6. Sampel Pengujian Model BiLSTM+CNN



Gambar 4.7. Sampel Pengujian Model CNN+LSTM





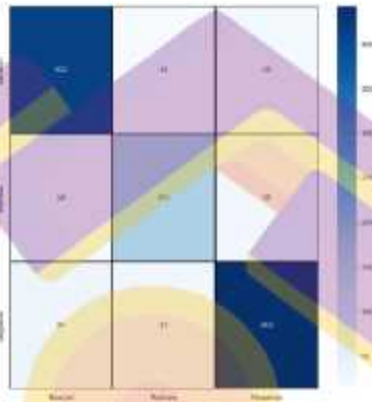
Gambar 4.8. Sampel Pengujian Model CNN+BiLSTM

Berdasarkan gambar pengujian diatas hasil yang paling baik ditunjukkan pada model BiLSTM+CNN Word2Vec yang menghasilkan nilai akurasi terendah pada data validasi *diepoch* pertama dengan nilai sebesar 0,6408 dan nilai *loss* tertinggi pada data training menghasilkan 1.5051. Pada epoch ke 50 model BiLSTM+ CNN Word2Vec menghasilkan akurasi yang baik dibanding dengan model yang lainnya. Perbandingan terlihat bahwa model sudah cukup baik dengan menunjukkan akurasi data training dengan validasi tidak berbeda jauh sehingga tidak mengalami *over fitting*. Selanjutnya model akan diuji kepercayaannya pada tahap pengujian dengan data testing.

#### 4.4. Pengujian Data Testing

Tahap ini merupakan tahap pengujian model dengan data testing. Data yang akan diuji adalah berjumlah 11.448 data *tweet* dengan jumlah label di kelas negatif 5.827, 4.396 kelas netral dan 1.236 kelas positif. Selanjutnya setelah model menghasilkan prediksinya di masing-masing kelas dilakukan perhitungan tingkat kepercayaan model dengan melihat akurasi, presisi dan *recall*. Tujuannya untuk

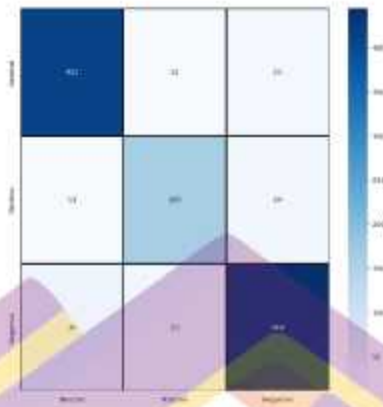
mengetahui seberapa dapat dipercaya model dalam memprediksi kelas. Hasil pengujian data testing ditunjukkan dengan *confusion matrix* seperti table dibawah ini. Tabel 4.1 – Tabel 4.4 menunjukan hasil *Confusion Matrix* masing – masing model



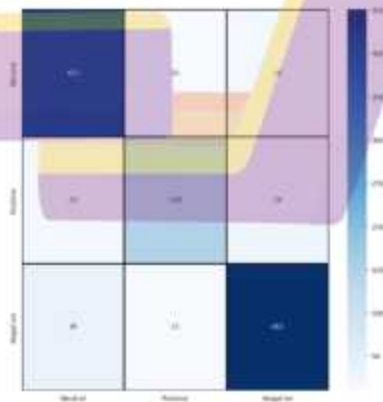
Gambar 4.9. *Confusion Matrix* Model LSTM+CNN

Tabel 4.1. *Confusion Matrix* Model LSTM+CNN

Aktual	Prediksi		
	Netral	Positif	Negatif
Netral	412	13	20
Positif	18	154	25
Negatif	34	27	442

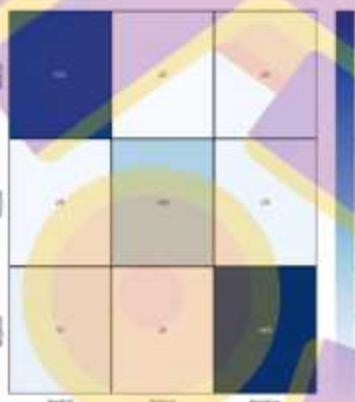
Gambar 4.10. *Confusion Matrix* Model BiLSTM+CNNTabel 4.2. *Confusion Matrix* Model BiLSTM+CNN

Aktual	Prediksi		
	Netral	Positif	Negatif
Netral	413	11	21
Positif	13	160	24
Negatif	34	25	444

Gambar 4.11. *Confusion Matrix* Model CNN+LSTM

Tabel 4.3. *Confusion Matrix* Model CNN+LSTM

Aktual	Prediksi		
	Netral	Positif	Negatif
Netral	415	10	20
Positif	20	148	29
Negatif	39	13	451

Gambar 4.12. *Confusion Matrix* Model CNN+BiLSTMTabel 4.4. *Confusion Matrix* Model CNN+BiLSTM

Aktual	Prediksi		
	Netral	Positif	Negatif
Netral	416	10	19
Positif	20	148	29
Negatif	42	16	445

Bedasarkan model yang telah dibuat dimana data test yang berlabel negatif benar diprediksi oleh model LSTM+CNN sebesar 412, sedangkan pada model BiLSTM+CNN sebesar 413, pada model CNN+LSTM sebesar 415 dan pada model CNN+BiLSTM sebesar 416. Dari *Confusion Matrix* pada Tabel 4.1 – Tabel 4.4

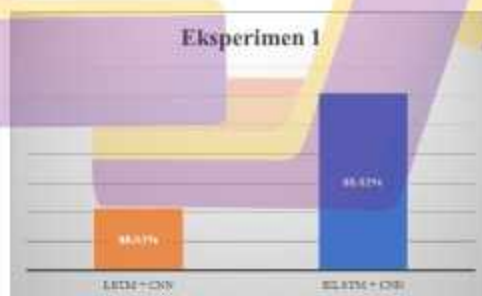
model yang menghasilkan nilai prediksi positif benar paling banyak adalah model BiLSTM+CNN, sedangkan model yang menghasilkan nilai negatif benar paling banyak adalah model CNN+BiLSTM.

#### 4.5. Hasil Klasifikasi

Pada klasifikasi sentimen opini publik mengenai UU KPK telah dilakukan eksperimen. Dasar eksperimen menggunakan parameter pembobotan kata *Word2Vec*, model (LSTM+CNN, BiLSTM+CNN, CNN+LSTM, dan CNN+BiLSTM) serta ukuran data Training, Validasi, dan Tesing masing-masing 80%:10%:10%.

##### 4.5.1. Hasil Klasifikasi Eksperimen Pertama

Eksperimen pertama dilakukan dengan model LSTM+CNN dan BiLSTM+CNN. Hasil klasifikasi skenario pertama dapat dilihat Gambar 4.13.



Gambar 4.13. Hasil Klasifikasi Eksperimen Pertama



Akurasi terbaik model LSTM+CNN dengan *Word2Vec* pada eksperimen pertama adalah 88.03%. Sedangkan akurasi model BiLSTM+CNN dengan *Word2Vec* pada eksperimen pertama adalah 88.82%.

#### 4.5.2. Hasil Klasifikasi Eksperimen Kedua

Eksperimen kedua merupakan eksperimen model klasifikasi yang dilakukan tidak jauh berbeda dengan eksperimen pertama, perbedaan yang dapat dilihat dengan model CNN+LSTM dan CNN+BiLSTM. Hasil eksperimen kedua dapat dilihat pada Gambar 4.14.



Gambar 4.14. Hasil Klasifikasi Eksperimen Kedua

Akurasi terbaik model CNN+LSTM dengan *Word2Vec* pada eksperimen pertama adalah 88.65%. Sedangkan akurasi model CNN+BiLSTM dengan *Word2Vec* pada eksperimen pertama adalah 88.03%.

### 4.5.3. Akurasi Terbaik Setiap Model Eksperimen

Rangkuman akurasi hasil klasifikasi terbaik setiap dimensi dan skenario dapat dilihat pada Tabel 4.5.

Tabel 4.5. Akurasi Terbaik

Model Eksperimen	Eksperimen 1		Eksperimen 2	
	LSTM + CNN	BiLSTM + CNN	CNN + LSTM	CNN + BiLSTM
Akurasi	88.03%	88.82%	88.65%	88.03%

Penjelasan parameter hasil akurasi pada Tabel 4.5. sebagai berikut:

1. Akurasi terbaik yang didapatkan pada eksperimen pertama sebesar 88.82% dengan model BiLSTM+CNN.
2. Akurasi terbaik yang didapatkan pada eksperimen model kedua sebesar 88.65% dengan model CNN+LSTM.

## 4.6. Pembahasan

Bagian ini akan membahas perbandingan hasil klasifikasi antara model eksperimen (LSTM+CNN, BiLSTM+CNN, CNN+LSTM, dan CNN+BiLSTM), rasio ukuran data, dan hasil uji sentimen.

### 2.6.1. Perbandingan Model Eksperimen

Perbandingan hasil performansi model dari setiap eksperimen dilakukan berdasarkan akurasi dari setiap klasifikasi terhadap rasio ukuran data. Hasil perbandingan dilihat pada Gambar 4.15.



Gambar 4.15. Perbandingan Model Eksperimen

Pada model eksperimen BiLSTM+CNN memiliki akurasi terbaik, model CNN+LSTM melakukan klasifikasi cukup baik daripada LSTM+CNN dan CNN+BiLSTM.

### 2.6.2. Evaluasi Performansi

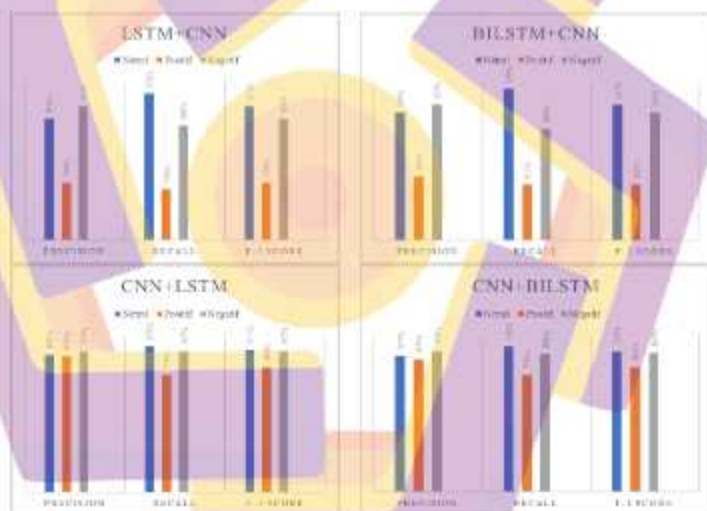
Setelah model menghasilkan prediksinya di masing-masing kelas dilakukan perhitungan tingkat kepercayaan model dengan melihat presisi, *recall* dan *f1-score*. Tujuannya untuk mengetahui seberapa besar persentase untuk dapat dipercaya sebagai model dalam memprediksi kelas. Evaluasi performansi berdasarkan rasio ukuran data dapat dilihat dari hasil presisi, *recall* dan *f1-score* setiap dimensi dan model pada Gambar 4.16 dan Tabel 4.6.

Tabel 4.6. Perbandingan Performansi

Model		<i>Precision</i>	<i>Recall</i>	<i>F-1 Score</i>
LSTM+CNN	Netral	89%	93%	91%
	Positif	79%	78%	79%
	Negatif	91%	88%	89%

Tabel 4.6. (Lanjutan)

Model		Precision	Recall	F-1 Score
BiLSTM+CNN	Netral	90%	93%	91%
	Positif	82%	81%	81%
	Negatif	91%	88%	90%
CNN+LSTM	Netral	88%	93%	91%
	Positif	87%	75%	80%
	Negatif	90%	90%	90%
CNN+BiLSTM	Netral	87%	93%	90%
	Positif	85%	75%	80%
	Negatif	90%	88%	89%



Gambar 4.16. Perbandingan Performansi

Dari hasil evaluasi performansi rasio ukuran data, didapati bahwa selisih nilai performansi sangat berdekatan dikarenakan dataset memiliki jumlah data *False Negatif* dan *False Positif* yang kurang mendekati (*Symmetric*). Karena jumlahnya

tidak mendekati, maka pada penelitian ini juga menggunakan *F1 Score* sebagai acuan.

#### 4.2.3. Perbandingan Hasil Penelitian

Bagian ini akan membahas hasil klasifikasi yang dilakukan perbandingan antara hasil penelitian ini dengan referensi rujukan yang disertai dengan sumber datasetnya. Perbandingan dapat dilihat pada Tabel 4.7 berikut:

Tabel 4.7. Penelitian Terdahulu

Model Eksperimen		Penelitian Terdahulu			Akurasi
1	LSTM + CNN	(Biswas & Dash, 2022)	Twitter – Covid19	75.72%	88.03%
	BiLSTM + CNN	(Duwairi dkk., 2021)	Twitter – Ujaran Kebencian Bahasa Arab	73.00%	88.82%
2	CNN + LSTM	(Hermanto dkk., 2021)	Situs Detik finance - Judul artikel bahasa Indonesia	74.00%	88.65%
	CNN + BiLSTM	(Srivastava dkk., 2023)	Twitter – Analisis Sentimen	80.00%	88.03%

Keterangan:

- Penelitian terdahulu adalah hasil akurasi penelitian terdahulu yang digunakan oleh penulis untuk referensi pada penelitian ini.
- Akurasi adalah hasil akurasi pada penelitian ini pada model eksperimen 1 dan model eksperimen 2 menggunakan dataset berkarakter klasifikasi statis.

Hasil performansi model LSTM+CNN membandingkan dengan penelitian yang dilakukan (Biswas & Dash, 2022). Dari hasil perbandingan didapatkan pada skenario (Biswas & Dash, 2022) sebesar 75.72%, sedangkan pada penelitian ini



mendapatkan akurasi sebesar 88.03%. Selanjutnya hasil performansi model BiLSTM+CNN membandingkan dengan penelitian yang dilakuka (Duwairi dkk., 2021). Dari hasil perbandingan didapatkan pada skenario (Duwairi dkk., 2021) sebesar 73.00%, sedangkan pada penelitian ini mendapatkan akurasi sebesar 88.82%.

Penelitian (Hermanto dkk., 2021) menggunakan pembobotan kata *Word2Vec* mendapatkan akurasi sebesar 74.00%, sedangkan pada penelitian ini mendapatkan akurasi sebesar 88.65%. Sedangkan hasil performansi model CNN+BiLSTM membandingkan dengan penelitian yang dilakuka (Srivastava dkk., 2023). Dari hasil perbandingan didapatkan hasil akurasi penelitian (Srivastava dkk., 2023) sebesar 80.00%, sedangkan pada penelitian ini mendapatkan akurasi sebesar 88.03%.

#### 4.2.4. Hasil uji sentimen

Pada bagian ini akan menjelaskan contoh hasil uji sentiment dengan menggunakan model yang sudah dilakukan training. Hasil dari prediksi sentiment positif, negative, dan netral dengan menggunakan data Twitter dapat dilihat pada Tabel 4.8.

Tabel 4.8. Hasil Prediksi

Kalimat yang diprediksi	Label Hasil Uji Sentimen
kpk	negative

*Source code* untuk hasil uji sentiment dalam memprediksi kalimat dapat dilihat sebagai berikut:

```

# Padding data
max_length1 = 20

# Contoh kalimat yang akan diprediksi
kalimat = "kpk"

# Konversi kalimat ke dalam bentuk sequence
sequence = tokenizer.texts_to_sequences([str(kalimat)])
sequence = pad_sequences(sequence, maxlen=max_length1,
padding='post')

# Melakukan prediksi
prediksi = modelBiLSTM.predict(sequence)

# Membulatkan nilai prediksi menjadi 0 atau 1
if prediksi < 0:
    label = "positive"
else:
    label = "negative"

print("Kalimat: ", kalimat)
print("Label: ", label)

1/1 [-----] - 1s 1s/step
Kalimat: kpk
Label: negative

```

#### 4.2.5. Kelebihan dan Kekurangan Penelitian

Selama proses training, validasi, dan tesing pada penelitian ini, terdapat beberapa kelebihan dan kelemahan. Adapun kelebihan yang didapatkan sebagai berikut:

- a. Eksperimen model dengan menggunakan integrasi model antara LSTM dan BiLSTM dengan CNN.

- b. Tahapan praproses data perlu diperbaiki sehingga dapat mengatasi kesalahan klasifikasi akibat penulisan komentar pengguna yang tidak sesuai dengan kaidah bahasa yang baik.

#### 4.2.6. Kontribusi Keilmuan

Kontribusi keilmuan pada arsitektur metode *deep learning* LSTM dan BiLSTM dengan kombinasi CNN terbukti dapat menjadi alternatif dalam melakukan analisis sentimen opini publik menggunakan pemodelan data *text* dimana pada penelitian terdahulu metode tersebut lebih banyak dan cocok digunakan pada pemodelan data *time series*.

Meskipun hasil eksperimen pada penelitian ini tidak mencapai akurasi > 90% melalui model hybrid yang di uji, namun penelitian ini masih dapat memberikan akurasi yang dapat memprediksi klasifikasi opini sentiment publik. Dalam pendekatan pada penelitian ini, peneliti menemukan keragaman baru mengenai ulasan teks Bahasa Indonesia, sehingga dilakukan dua kali normalisasi yaitu pertama menggunakan kamus alay dari indo *bercamari* dimana kamus alay ini akan mengubah kata-kata atau bahasa slang menjadi kata baku dan kedua menggunakan kamus indonesian-word. Sehingga dataset menjadi mudah diakses dan mudah dikelola.

## BAB V

### PENUTUP

#### 5.1. Kesimpulan

Berdasarkan rumusan masalah dan penjelasan yang telah peneliti deskripsikan pada bab sebelumnya, maka peneliti dapat menarik kesimpulan sebagai berikut:

1. Nilai akurasi pada model BiLSTM+CNN lebih tinggi dibandingkan dengan model eksperimen lainnya yaitu sebesar 88.82%.
2. Model BiLSTM+CNN merupakan model terbaik pada penelitian ini dibandingkan dengan model BiLSTM+CNN, CNN+LSTM, dan CNN+BiLSTM. Hasil akurasi algoritma terbaik pada penelitian ini tidak dapat di generalisasi. Hal ini dikarenakan kompleksitas *dataset* atau penggunaan *dataset* yang berbeda dapat menghasilkan tingkat akurasi yang tidak sama.
3. Dataset berkarakter statis pada penelitian ini terbukti dapat digunakan pada klasifikasi model hybrid yang di uji. Meskipun secara karakter LSTM dan BiLSTM memang lebih cocok untuk *time series trend prediction*.

## 5.2. Saran

Peneliti menyadari bahwa pendekatan klasifikasi masih memiliki beberapa kekurangan serta keterbatasan. Oleh karena itu, ada beberapa hal yang perlu dipertimbangkan untuk mengembangkan penelitian selanjutnya menjadi lebih baik yaitu:

1. Pada penelitian selanjutnya dapat menambahkan jumlah *dataset* yang digunakan dan melakukan perbandingan akurasi berdasarkan jumlah dataset yang lebih banyak.
2. Pada penelitian selanjutnya pelabelan manual perlu melibatkan pakar Bahasa Indonesia.
3. Pada penelitian selanjutnya dapat mengkombinasikan model-model algoritma yang lain untuk mendapatkan hasil akurasi yang lebih baik.
4. Pada penelitian selanjutnya dapat mencari algoritma yang bisa menangani masalah ketimpangan data agar hasil agar keputusan model yang dibuat tidak cenderung ke kelas yang paling dominan.



## DAFTAR PUSTAKA

- Abdurrazzaq, M. A., & Tjiong, E. L. (2022). Analisis Sentimen KUHP Baru Pada Data Twitter Menggunakan BERT. *Jurnal Komunikasi, Sains dan Teknologi*, 1(2), 38–43.
- Agus, F., & Hatta, H. R. (2017). Pengklasifikasian Dokumen Berbahasa Arab Menggunakan K-Nearest Neighbor. 18(1), 14.
- Ahmad, H., Asghar, M. U., Asghar, M. Z., Khan, A., & Mosavi, A. H. (2021). A hybrid deep learning technique for personality trait classification from text. *IEEE Access*, 9, 146214–146232.
- Amada, M., Munawar, M., & Pilliang, M. (2023). Sentiment Analysis of Content Permenkominfo No. 5 of 2020 Using a Classification Algorithm. *JITK (Jurnal Ilmu Pengetahuan dan Teknologi Komputer)*, 8(2), 125–132.
- Amanullah, R. F., Utami, E., & Sunyoto, A. (2019). Citation Detection on Scientific Journal Using Support Vector Machine. *2019 International Conference on Information and Communications Technology (ICOIACT)*. <https://doi.org/10.1109/icoiact46704.2019.8938522>
- Andreas, A., & Prabowo, Y. D. (2022). Analisis Sentimen Opini Publik dalam Bahasa Indonesia Terhadap UU Cipta Kerja Menggunakan Naïve Bayes. *KALBISIANA Jurnal Sains, Bisnis dan Teknologi*, 8(2), 2147–2161.
- Ashari, N., Al Firdaus, M. Z. M., Budi, I., Santoso, A. B., & Putra, P. K. (2023). Analyzing Public Opinion on Electrical Vehicles in Indonesia Using Sentiment Analysis and Topic Modeling. 461–465.
- Astiningrum, M., & Batubulan, K. S. (2020). Implementasi Analisis Sentimen Twitter Mengenai Opini Masyarakat Terhadap Rkuhp Tahun 2019. Seminar Informatika Aplikatif Polinema.
- Augustia, A. E., Taufan, R., Alkhalifi, Y., & Gata, W. (2021). Analisis Sentimen Omnibus Law Pada Twitter Dengan Algoritma Klasifikasi Berbasis Particle Swarm Optimization. *Paradigma*, 23(2), 158–166.
- Berliana, D. R., & Santoso, B. (2023). Elektabilitas Ridwan Kamil Dan Anies Baswedan Dalam Simulasi Pilpres 2024 Di Twitter (Analisis Jaringan Media Sosial Dan Analisis Sentimen Pengguna Twitter Terhadap #Ridwankamil Dan #Aniesbaswedan). *Mediakom: Jurnal Ilmu Komunikasi*, 6(2), 150–162.
- Biswas, S., & Dash, S. (2022). LSTM-CNN deep learning-based hybrid system for real-time COVID-19 data analysis and prediction using Twitter data. *Assessing COVID-19 and Other Pandemics and Epidemics using Computational Modelling and Data Analysis*, 239–257.

- Budayani, N. K. M., Slamet, I., & Handajani, S. S. (2023). *A Comparison of SVM Kernel Functions for Sentiment Analysis of UU TPKS*. 2, 761–765.
- Chaturvedi, S., Mishra, V., & Mishra, N. (2017). *Sentiment analysis using machine learning for business intelligence*. 2162–2166.
- Dauni, P., Ramdhani, M. A., Suryapratama, D. R., Zulfikar, W. B., & Fuadi, R. S. (2022). *Twitter User Sentiment Analysis For RUU Omnibus Law Using Convolutional Neural Network*. 1–6.
- Duwairi, R., Hayajneh, A., & Quwaider, M. (2021). A deep learning framework for automatic detection of hate speech embedded in Arabic tweets. *Arabian Journal for Science and Engineering*, 46, 4001–4014.
- Ertugrul, A. M., & Karagoz, P. (2018). *Movie genre classification from plot summaries using bidirectional lstm*. 248–251.
- Fikri, M. I., Sabrila, T. S., & Azhar, Y. (2020). Perbandingan metode naïve bayes dan support vector machine pada analisis sentimen twitter. *SMATIKA JURNAL: STIKI Informatika Jurnal*, 10(02), 71–76.
- Gata, W., & Purnomo. (2016). Akurasi Text Mining Menggunakan Algoritma K-Nearest Neighbour pada Data Content Berita SMS. *Jurnal Format*.
- Gata, W., Surohman, S., & Nawawi, H. M. (2023). *Twitter in analysis of policy sentiments of the omnibus law-work creative design*. 2714(1).
- Graves, A., & Schmidhuber, J. (2005). Framewise phoneme classification with bidirectional LSTM and other neural network architectures. *Neural networks*, 18(5–6), 602–610.
- Helmiah, N., Nooraeni, R., Nulkarim, A. R., Munia, N., Susanti, A., Putra, A. P., & Taufiqurrahman, F. (2020). Penerapan Metode Naïve Bayes dalam Analisis Persepsi Masyarakat mengenai Rencana Pengesahan RUU Omnibus Law di Bidang Investasi dan Ketenagakerjaan Tahun 2020 di Indonesia. *Jurnal MSA (Matematika dan Statistika serta Aplikasinya)*, 8(2), 48–57.
- Hermanto, D. T., Setyanto, A., & Luthfi, E. T. (2021). Algoritma LSTM-CNN untuk Binary Klasifikasi dengan Word2vec pada Media Online. *Creative Information Technology Journal*, 8(1), 64–77.
- Hussein, D. M. E.-D. M. (2018). A survey on sentiment analysis challenges. *Journal of King Saud University-Engineering Sciences*, 30(4), 330–338.

Iskandar, J. W., & Nataliani, Y. (2021). Perbandingan Naïve Bayes, SVM, dan k-NN untuk Analisis Sentimen Gadget Berbasis Aspek. *Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi)*, 5(6), 1120–1126.

Jelodar, H., Wang, Y., Orji, R., & Huang, S. (2020). Deep sentiment classification and topic discovery on novel coronavirus or COVID-19 online discussions: NLP using LSTM recurrent neural network approach. *IEEE Journal of Biomedical and Health Informatics*, 24(10), 2733–2742.

Jo, T. (2019). Text Mining: Concepts, Implementation, and Big Data Challenge. *Springer International Publishing*. <https://doi.org/10.1007/978-3-319-91815-0>

Joshi, A., Bhattacharyya, P., & Ahire, S. (2017). Sentiment resources: Lexicons and datasets. *A Practical Guide to Sentiment Analysis*, 85–106.

Kane, S. N., Mishra, A., & Dutta, A. K. (2016). *Preface: International conference on recent trends in physics (ICRTP 2016)*. 755(1), 0–5.

Kantardzic, M. (2020). *Data Mining: Concepts, Models, Methods, and Algorithms, 3rd Edition*. Wiley-IEEE Press. <http://93.174.95.29/main/38960843B7D6FD8B71F3CE97F7D5B5BB>

Khuriawan, F. S., & Ruldeviyani, Y. (2020). *Twitter Sentiment Analysis: Case Study On The Revision of The Indonesia's Corruption Eradication Commission (KPK) Law 2019*. 1–6.

Lample, G., Ballesteros, M., Subramanian, S., Kawakami, K., & Dyer, C. (2016). Neural architectures for named entity recognition. *arXiv preprint arXiv:1603.01360*.

Macrohon, J. J. E., Villavicencio, C. N., Inbaraj, X. A., & Jeng, J.-H. (2022). A Semi-Supervised Approach to Sentiment Analysis of Tweets during the 2022 Philippine Presidential Election. *Information*, 13(10), 484.

Maimaiti, M., Wumaier, A., Abiderexiti, K., & Yibulayin, T. (2017). Bidirectional long short-term memory network with a conditional random field layer for Uyghur part-of-speech tagging. *Information*, 8(4), 157.

Manullang, S. O., Kusumadewi, Y., Verawati, V., Siburian, H. K., Siburian, H., & Sipayung, B. (2023). Problematika Hukum atas Pembentukan Perubahan Kedua atas UU KPK. *Journal on Education*, 5(2), 4885–4897.

Mihuandayani, Utami, E., & Luthfi, E. T. (2018). Text Mining Based on Tax Comments as Big Data Analysis Using SVM and Feature Selection. *International Conference on Information and Communications Technology (ICOIACT)*. <https://doi.org/10.1109/icoiact.2018.8350743>



Ndruru, A. (2022). Analisis Sentimen UU Cipta Kerja Melalui Omnibus Law Menggunakan Naive Bayes Classifier (NBC) Dan Support Vector Machine (SVM). *Pelita Informatika: Informasi dan Informatika*, 10(3), 85–90.

Nurdiansyah, Y., Rahman, F., & Pandunata, P. (2021). Analisis Sentimen Opini Publik Terhadap Undang-Undang Cipta Kerja pada Twitter Menggunakan Metode Naive Bayes Classifier. 3, 201–212.

Nurhusen, M. R., Indra, J., & Baihaqi, K. A. (2023). Analisis Sentimen Pengguna Twitter Terhadap Kenaikan Harga Bahan Bakar Minyak (BBM) Menggunakan Metode Logistic Regression. *JURNAL MEDIA INFORMATIKA BUDIDARMA*, 7(1), 276–282.

Nurrohmat, M. A., & Azhari, S. (2019). Sentiment Analysis of Novel Review Using Long Short-Term Memory Method. *IJCCS (Indonesian Journal of Computing and Cybernetics Systems)*, 13(3), 209–218.

Pakpahan, S., & Simanullang, A. (2022). Analisis Sentimen Integritas KPK Tahun 2021 Pencegahan Korupsi pada Twitter KPK Menggunakan Metode K-Nearest Neighbor dan Naïve Bayes. *Citra Sains Teknologi*, 2(1), 63–73.

Prastyo, P. H., Ardiyanto, I., & Hidayat, R. (2020). *Indonesian Sentiment Analysis: An Experimental Study of Four Kernel Functions on SVM Algorithm with TF-IDF*. 1–6.

Pratama, B. T., Utami, E., & Sunyoto, A. (2019). The Impact of Using Domain Specific Features on Lexicon Based Sentiment Analysis on Indonesian App Review. *2019 International Conference on Information and Communications Technology (ICOIACT)*. <https://doi.org/10.1109/ICOIACT46704.2019.8938419>

Putra, J. W. G. (2019). Pengenalan konsep pembelajaran mesin dan deep learning. *Department of Computer Science, Tokyo Institute of Technology*.

Putri, D. D., Nama, G. F., & Sulistiono, W. E. (2022). Analisis Sentimen Kinerja Dewan Perwakilan Rakyat (DPR) Pada Twitter Menggunakan Metode Naive Bayes Classifier. *Jurnal Informatika dan Teknik Elektro Terapan*, 10(1).

Reiki, M. K. A., Sibaroni, Y., & Setiawan, E. B. (2022). Comparison of Term Weighting Methods in Sentiment Analysis of the New State Capital of Indonesia with the SVM Method. *International Journal on Information and Communication Technology (IJoICT)*, 8(2), 53–65.

Rozalina, G. M., Anwar, V. L. N., & Dewi, P. N. (2023). Big Data Dan Sentimen Analisis Twitter Dalam Bingkai Pemindehan Ibu Kota Negara. *Prosiding Konferensi Nasional Sosiologi (PKNS)*, 1(1), 389–393.

Sandag, G. A., Soegiarto, E. H. E., Laoh, L., Gunawan, A., & Sondakh, D. (2022). *Sentiment Analysis of Government Policy Regarding PPKM on Twitter Using LSTM*. 1–6.

Sandryan, M. K., Rahayudi, B., & Ratnawati, D. E. (2021). Analisis Sentimen Pada Media Sosial Twitter Terhadap Undang-Undang Cipta Kerja Menggunakan Algoritma Backpropagation dan Term Frequency-Inverse Document Frequency. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer e-ISSN, 2548, 964X*.

Schuster, M., & Paliwal, K. K. (1997). Bidirectional recurrent neural networks. *IEEE transactions on Signal Processing, 45*(11), 2673–2681.

Setiawan, S. E. B., & Nugraha, F. N. (2019). Implementation of Decision Tree C4.5 for Big Five Personality Predictions with TF-RF and TF-CHI2 on Social Media Twitter. *2019 International Conference on Computer, Control, Informatics and Its Applications (IC3INA)*. <https://doi.org/10.1109/IC3INA48034.2019.8949601>

Srivastava, T., Arora, D., & Sharma, P. (2023). *Sentiment Analysis of COVID-19 Tweets Using BiLSTM and CNN-BiLSTM*. 523–535.

Sudrajat, T., & Rohida, L. (2022). *Efek Media Massa dalam Pembentukan Opini Publik di Masa Pandemi Covid-19*. 4, 519–524.

Sukma, E. A., Hidayanto, A. N., Pandesenda, A. I., Yahya, A. N., Widharto, P., & Rahardja, U. (2020). *Sentiment Analysis of the New Indonesian Government Policy (Omnibus Law) on Social Media Twitter*. 153–158.

Sulistiani, H., & Tjahyanto, A. (2017). Comparative Analysis of Feature Selection Method to Predict Customer Loyalty. *IPTEK Journal of Engineering, 3*(1), 1. <https://doi.org/10.12962/joe.v3i1.2257>

Utami, E., Hartanto, A. D., Adi, S., Oyong, I., & Raharjo, S. (2019). Profiling analysis of DISC personality traits based on Twitter posts in Bahasa Indonesia. *Journal of King Saud University - Computer and Information Sciences, S131915781930744X*. <https://doi.org/10.1016/j.jksuci.2019.10.008>

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2017). Attention is all you need. *Advances in neural information processing systems, 30*.

Wilie, B., Vincentio, K., Winata, G. I., Cahyawijaya, S., Li, X., Lim, Z. Y., Soleman, S., Mahendra, R., Fung, P., & Bahar, S. (2020). IndoNLU: Benchmark and resources for evaluating Indonesian natural language understanding. *arXiv preprint arXiv:2009.05387*.



## LAMPIRAN

### Lampiran 1. Dataset

Justifikasi terhadap *dataset* pada penelitian ini. Untuk sumber *dataset* atau *collecting* data diambil dari Twitter dengan menggunakan *scraping* dari <https://github.com/JustAnotherArchivist/snscrape.git> dengan mengambil 15.000 data. *Source code* untuk *scraping dataset* dapat dilihat sebagai berikut:

```
!pip install pandas
!pip install
git+https://github.com/JustAnotherArchivist/snscrape.git

import pandas as pd
import snscrape.modules.twitter as sntwitter
import itertools

queries = ['kpk', '#kpk', 'komisi pemberantasan
korupsi', '#uukpk', '#kpkkrakusjabatan']
since_date = '2023-05-01'
until_date = '2023-07-10'

tweets = []
max_tweets = 15000
counter = 0

for query in queries:
    query_string = query
    scraper = sntwitter.TwitterSearchScraper(query_string)

    for tweet in scraper.get_items():
        tweets.append([tweet.user.username, tweet.date,
tweet.content])
        counter += 1

    if counter >= max_tweets:
        break

if counter >= max_tweets:
    break
```

## Lampiran 2. Source Code

### Tahap Preprocessing :

#### Casefolding

```
df = data
import string
def preprocess_text(kalimat):
    # Mengubah kalimat menjadi huruf kecil
    lower_case = kalimat.lower()

    # Menghapus angka dari kalimat
    hasil = re.sub(r"\d+", "", lower_case)

    # Menghapus tanda baca dari kalimat
    hasil =
hasil.translate(str.maketrans("", "", string.punctuation))

    # Menghapus spasi pada awal dan akhir kalimat
    hasil = hasil.strip()

    # Menghapus Username Twitter
    hasil = re.sub('@[\w]+', '', hasil)

    # Menghapus https dan http
    hasil = re.sub(r"(?:\B|http?://|https?://|www)\S+", "",
hasil)

    # Menghilangkan Tanda Baca
    hasil = hasil.translate(str.maketrans('', '',
string.punctuation))

    # Mengganti karakter HTML dengan tanda patik
    hasil = re.sub('<.*?>', ' ', hasil)

    # Mempertimbangkan huruf dan angka
    hasil = re.sub('[^a-zA-Z0-9]', ' ', hasil)

    # Mengganti line baru dengan spasi
    hasil = re.sub("\n", " ", hasil)

    # Menghapus single char
    hasil = re.sub(r"\b[a-zA-Z]\b", " ", hasil)
```

```

# Menghilangkan emoji
emoji_pattern = re.compile("[
    u"\U0001F600-\U0001F64F" #
emotikon wajah
    u"\U0001F300-\U0001F5FF" #
simbol & benda
    u"\U0001F680-\U0001F6FF" #
transportasi & simbol peralatan
    u"\U0001F1E0-\U0001F1FF" #
bendera negara
    u"\U00002500-\U00002BEF" #
karakter CJK (Chinese, Japanese, Korean)
    u"\U00002702-\U000027B0" #
simbol & tanda
    u"\U00002702-\U000027B0"
    u"\U000024C2-\U0001F251"
    u"\U0001F926-\U0001F937"
    u"\U00010000-\U0010ffff"
    u"\u200d"
    u"\u2640-\u2642"
    u"\u2600-\u2B55"
    u"\u23cf"
    u"\u23e9"
    u"\u231a"
    u"\u3030"
    u"\ufe0f"
"]+", flags=re.UNICODE)

hasil = emoji_pattern.sub(r'', hasil)

# Memisahkan dan menggabungkan kata
hasil = ' '.join(hasil.split())

return hasil
df['text_clean'] = df['text'].apply(preprocess_text)
df.head()

```

## Normalisasi 1

```

kamus = pd.read_csv('kamusalay.csv', header=None)
kamus.columns = ['kata_asli', 'kata_normal']
def normalisasi_teks(teks, kamus):
    kata_asli = teks.split()
    kata_normal = []

```

```

for kata in kata_asli:
    normal = kamus[kamus['kata_asli'] ==
kata]['kata_normal'].values
    if normal:
        kata_normal.append(normal[0])
    else:
        kata_normal.append(kata)

return ' '.join(kata_normal)

df['normal'] = df['text_clean'].apply(normalisasi_teks,
kamus=kamus)
data.head()

```

### ***Tokenizing***

```

import nltk
nltk.download('all')
from nltk.tokenize import word_tokenize
def tokenize_text(kalimat):
    tokens = nltk.tokenize.word_tokenize(kalimat)
    return tokens
df['token'] = df['normal'].apply(tokenize_text)
df.head()

```

### ***Filtering (Stopword Removal)***

```

from Sastrawi.StopWordRemover.StopWordRemoverFactory import
StopWordRemoverFactory
factory = StopWordRemoverFactory()
stopwords = factory.get_stop_words()
print(stopwords)
def stopwords_text(tokens):
    cleaned_tokens = []
    for token in tokens:
        if token not in stopwords:
            cleaned_tokens.append(token)
    return cleaned_tokens
df['stop'] = df['token'].apply(stopwords_text)
df.head()

```

## Stemming

```
from Sastrawi.Stemmer.StemmerFactory import StemmerFactory
stem_factory = StemmerFactory()
stemmer = stem_factory.create_stemmer()
def stemming_text(tokens):
    hasil = [stemmer.stem(token) for token in tokens]
    return hasil
df['stemmed'] = df['stop'].apply(stemming_text)
df.head()
```

## Normalisasi 2

```
with open('indonesian-words.txt', 'r', encoding='utf-8') as file:
    kamus_bahasa = set(file.read().splitlines())
def normalisasi_teks_part2(tokens, kamus, kamus_bahasa):
    kata_normal = []
    for kata in tokens:
        normal = kamus[kamus['kata_asli']] ==
        kata]['kata_normal'].values
        if normal:
            kata_normal.append(normal[0])
        else:
            kata_normal.append(kata)
    kata_normal = [kata for kata in kata_normal if kata in
    kamus_bahasa]
    return kata_normal
df['normal_2'] = df['stemmed'].apply(normalisasi_teks_part2,
    kamus=kamus, kamus_bahasa=kamus_bahasa)
df.head()
data = df
data = data[data['normal_2'].map(lambda x: len(x) > 0)]
data.head()
data.shape
```

## Training LSTM CNN Model

```
# Define the model architecture (LSTM-CNN)
model = Sequential()
model.add(Embedding(vocab_size, output_dim=EMBEDDING_DIM,
    weights=[embedding_vectors], input_length=20, trainable=True))
model.add(LSTM(units=128, return_sequences=True))
```



```

model.add(Dropout(0.1))
model.add(Conv1D(filters=64, kernel_size=3,
activation='relu'))
model.add(MaxPooling1D(pool_size=2))
model.add(Flatten()) # Menambahkan lapisan Flatten sebelum
Dense layer
model.add(Dense(3, activation='softmax'))
model.compile(optimizer='adam',
loss='categorical_crossentropy', metrics=['acc'])

```

### Training BiLSTM CNN Model

```

# Define the model architecture (BiLSTM-CNN)
model1 = Sequential()
model1.add(Embedding(vocab_size, output_dim=EMBEDDING_DIM,
weights=[embedding_vectors], input_length=20, trainable=True))
model1.add(Bidirectional(LSTM(64, return_sequences=True)))
model1.add(Dropout(0.1))
model1.add(Conv1D(filters=64, kernel_size=3,
activation='relu'))
model1.add(MaxPooling1D(pool_size=2))
model1.add(Flatten()) # Menambahkan lapisan Flatten sebelum
Dense layer
model1.add(Dense(3, activation='softmax'))
model1.compile(optimizer='adam',
loss='categorical_crossentropy', metrics=['accuracy'])

```

### Training CNN LSTM Model

```

# Define the model architecture (CNN-LSTM)
model = Sequential()
model.add(Embedding(input_dim=vocab_size, output_dim=100,
weights=[embedding_vectors], input_length=20, trainable=True))
model.add(Conv1D(filters=64, kernel_size=3,
activation='relu'))
model.add(MaxPooling1D(pool_size=2))
model.add(LSTM(units=128))
model.add(Dropout(0.1))
model.add(Dense(3, activation='softmax'))
model.compile(optimizer='adam',
loss='categorical_crossentropy', metrics=['acc'])

```

## Training CNN BiLSTM Model

```
# Define the model architecture (CNN-BiLSTM)
modell = Sequential()
modell.add(Embedding(input_dim=vocab_size, output_dim=100,
weights=[embedding_vectors], input_length=20, trainable=True))
modell.add(Conv1D(filters=64, kernel_size=3,
activation='relu'))
modell.add(MaxPooling1D(pool_size=2))
modell.add(Bidirectional(LSTM(64)))
modell.add(Dropout(0.1))
modell.add(Dense(3, activation='softmax'))
modell.compile(optimizer='adam',
loss='categorical_crossentropy', metrics=['accuracy'])
```

