BAB 2

TINJAUAN PUSTAKA

2.1 Literatur Review

Dalam kajian literatur ini, penelitian ini mengenai Pengaruh *Word Normalization* dan *Levenshtein Distance* pada analisis sentiment *Cyberbullying* terhadap Komentar Instagram. Metode yang dilakukan pada penelitian ini adalah *Lexicon-Based* dalam perbaikan kata tidak baku. Hal ini berbagai penelitian telah dilakukan untuk mengevaluasi efektivitas algoritma dan teknik *Pre-Processing* dalam analisis sentimen, terutama dalam konteks deteksi *Cyberbullying* di instagram oleh Heri Santoso dkk(2023) menggunakan algoritma Forest Tree[2], Fauzan Baehaqi dan Nuri Cahyono (2021) menggunakan Naïve Bayes [3], Rizky Syarif et al. (2019) menggunakan Naïve Bayes dan Lexicon Based[4]. Masalah pada Rizky Syarif adalah Naïve Bayes menghasilkan akurasi yang jauh lebih tinggi (97%) dibandingkan Lexicon-Based (58%) padahal penelitian oleh Muhammad Wisnu Prayuda dkk(2022)[5], Penelitian yang dilakukan oleh Galuh Etha Pratiwi dkk yang melakukan [7] menyatakan bahwa analisis sentimen yang menggunakan metode *Lexicon-Based* memiliki akurasi cukup tinggi. Begitu pula penelitian yang dilakukan oleh Novrido Charibaldi dkk [8], Katarzyna Marszalek-Kowalewsk [10], Prananda Antinasari dkk [11] sebagai referensi pada penggunaan Word Normalization dalam perbaikan kata tidak baku dalam analisis sentimen. Sedangkan Penelitian yang dilakukan oleh M. Adnan Nur [12], Fahmi Reza Prasastio dkk [13] menunjukkan bahwa *Levenshtein Distance* sebagai koreksi kata salah eja dapat memiliki akurasi tinggi.

. Studi ini menunjukkan bahwa teknik *Pre-Processing*, seperti *Word Normalization* dan *Typo Checking* menggunakan Algoritma *Levenshtein Distance*, memainkan peran penting dalam meningkatkan akurasi dan efektivitas algoritma analisis sentimen, yang sangat relevan dalam deteksi *Cyberbullying* di media sosial.

Tabel 2. 1 Literatur Review

No	Judul dan Penulis	Tahun dan Penulis	Objek Penelitian	Perbandingan yang dijadikan alasan tinjaun penelitian
1	Deteksi Komentar *Cyberbullying* pada Media Sosial Instagram Menggunakan Algoritma *Random Forest*	2023 Penulis : Heri Santoso, Raissa Amanda Putri, Sahbandi	Komentar *Cyberbullying* di Instagram	Penelitian ini digunakan jadikan sebagai referensi untuk penelitian *Cyberbullying*
2	Analisis Sentimen Terhadap *Cyberbullying* Pada Komentar di Instagram Menggunakan Algoritma *Naïve Bayes*	2021 Penulis : Fauzan Baehaqi dan Nuri Cahyono	*Cyberbullying* Pada Komentar di Instagram	Penelitian ini digunakan jadikan sebagai referensi untuk penelitian *Cyberbullying*
3	Identifikasi *Cyberbullying* pada Komentar Instagram menggunakan Metode *Lexicon-Based* dan *Naïve Bayes Classifier* (Studi kasus:	2019 Penulis : Rizky Dhian Syarif, Anisa Herdiani, S.T., M.T., Widi Astuti, S.T., M.Kom	*Cyberbullying* pada Komentar Instagram	Penelitian ini digunakan sebagai patokan utama peneliti dalam menggunakan metode *Lexicon-Based* dan mendapatkan masalah dari penelitian ini.

	Pemilihan Presiden Indonesia Tahun 2019)			
4	Penerapan Metode *Lexicon Based* untuk Menganalisis Sentimen Terhadap Mudik Lebaran	2022 Penulis : Muhammad Wisnu Prayuda, Angga Aditya Permana	Mudik Lebaran	Penelitian ini digunakan jadikan sebagai referensi untuk penelitian Metode *Lexicon-Based*
5	Analisis Sentimen Ulasan Pengguna pada Aplikasi Threads dengan Metode *Lexicon Based* dan *Naïve Bayes Classifier*	2023 Penulis: Solagratia Saron Tandiapa, Gladly Caren Rorimpandey	Aplikasi Threads	Penelitian ini digunakan jadikan sebagai referensi untuk penelitian Metode *Lexicon-Based*
6	Analisis Sentimen Brand Ambassador Artis Korea Selatan pada Produk Indonesia dengan Lexicon	2022 Penulis: Galuh Etha Pratiwi, Tiani Wahyu Utami, Rochdi Wasono	Artis Korea sebagai BA pada produk indonesia	Penelitian ini digunakan jadikan sebagai referensi untuk penelitian Metode *Lexicon-Based*

7	Comparison of the Effect of *Word Normalization* on *Naïve Bayes Classifier* and *K-Nearest Neighbor* Methods for Sentiment Analysis	2024 Penulis: Novrido Charibaldi, Atania Harfiani, Oliver Samuel Simanjuntak	Opini Publik pada BPJS Kesehatan	Penelitian ini digunakan jadikan sebagai referensi untuk Teknik *Word Normalization*
8	The Impact of *Text Normalization* on Multiword Expressions Discovery in Persian	2021 Penulis: Katarzyna Marszalek-Kowalewska	Bahasa persia	Penelitian ini digunakan jadikan sebagai referensi untuk Teknik *Word Normalization*
9	Normalization of Unstructured and Informal Text in Sentiment Analysis	2018 Penulis : Muhammad Javed dan Shahid Kamal	Unstructured and Informal Text	Penelitian ini digunakan jadikan sebagai referensi untuk Teknik *Word Normalization*
10	Analisis Sentimen Tentang Opini Film Pada Dokumen Twitter Berbahasa	2017 Penulis : Prananda Antinasari, Rizal Setya	Opini Film Pada Dokumen Twitter	Penelitian ini digunakan jadikan sebagai referensi untuk Teknik *Word Normalization* dan *Levenshtein Distance*

	Indonesia Menggunakan *Naive Bayes* Dengan Perbaikan Kata Tidak Baku	Perdana, M. Ali Fauzi	Berbahasa Indonesia	
11	Perbandingan *Levenshtein Distance* dan *Jaro-Winkler Distance* untuk Koreksi Kata dalam *Pre - Processing* Analisis Sentimen Pengguna Twitter	2021 Penulis: M. Adnan Nur	*Pre-Processing* Analisis Sentimen Pengguna Twitter	Penelitian ini digunakan jadikan sebagai referensi untuk Teknik Leveishtein Distance
12	Analisis Sentimen Vaksin Covid-19 Menggunakan Algoritma *Naive Bayes* dan Koreksi Kata *Levenshtein Distance*	2022 Penulis: Fahmi Reza Prasastio, Heriyanto, dan Wilis Kaswidjanti	Vaksin Covid-19	Penelitian ini digunakan jadikan sebagai referensi untuk Teknik *Leveishtein Distance*

2.2 Cyberbullying

Cyberbullying merupakan suatu perlaku dan perbuatan yang dilakukan dengan sengaja dan secara berulang-ulang dengan tindakan berupa tekanan, intimidasi,

pelecehan, perkataan dan perbuatan kasar secara verbal yang dilakukan melalui media internet yaitu media sosial di dunia maya[1].

2.3 Analisis sentimen

Analisis sentimen atau opinion mining merupakan bagian dari text mining, dimana pada proses ini akan dilakukan ekstrak, mengolah, dan memahami data yang berbentuk tekstual untuk mendapatkan informasi sentiment yang terkandung dalam suatu kalimat[16].

2.4 Perbaikan kata tidak baku

Kata tidak baku pada sebuah opini dikarenakan penulisan kata singkat ,penggunaan bahasa modern atau slang dan penulisaan ejaan yang salah.perbaikan dilakukan untuk kata tidak baku menjadi kata baku. Menurut Buntoro Perbaikan kata tidak baku atau normalisasi bahasa adalah proses yang digunakan untuk mengubah kata-kata yang tidak baku menjadi kata baku sesuai dengan Kamus Besar Bahasa Indonesia (KBBI)[11].

2.5 Typo Checking

Typo checking adalah proses pemeriksaan kata untuk mendeteksi kata yang salah eja dan memberikan kandidat kata yang benar[9].

2.6 Pre-Processing

Pre-Processing adalah tahapan mengolah data menjadi lebih terstruktur sebelum diolah. Pada tahap ini meliputi *Case Folding*, *Data Cleaning*, *Tokenizing*, *Word Normalization*, *Typo-Checking*, *Stopword Removal* dan *Stemming*.

2.7 Case Folding

Case Folding merupakan mengkonversi keseluruhan teks menjadi bentuk standar, yaitu huruf keci[4] . Contoh proses *case folding* pada dibawah ini.

Tabel 2. 2 Contoh Case Folding

No	Sebelum	Sesudah
1	"CONGRATS KAK ISYAN semoga karya nya apa yg dilakukan kakak sukses terus”	“congrats kak isyan semoga karya nya apa yg dilakukan kakak sukses terus”

2.8 Data Cleaning

Data cleaning adalah proses identifikasi, koreksi, dan penghapusan noise pada data. Dilakukan pengahapusan simbol,@ ,emoticon dan spasi tambahan[4] seperti contoh dibawah ini.

Tabel 2. 3 Contoh Data Cleaning

No	Sebelum	Sesudah
1	“@ayu.kinantii isyan skrg berubah ya:(baju nya nakal”	“ayu.kinantii isyan skrg berubah ya baju nya nakal”

2.9 Tokenizing

Tokenizing adalah kalimat akan dipecah berdasarkan spasi menjadi potongan kata [6]. Contohnya saya suka memasak menjadi “saya” “suka” “memasak”.

Tabel 2. 15 Contoh Tokenizing

No	Sebelum	Sesudah
1	“saya suka memasak”	“saya”,”suka”,”memasak”

2.10 Word Normalization

Normalisasi kata mengubah kata tidak baku menjadi kata atau bentuk baku menurut Kamus Besar Bahasa Indonesia (KBBI) [8][18].Langkah-langkah dalam proses Word Normalization sebagai berikut

Tabel 2. 4 Contoh Word Normalization

No	Sebelum	Sesudah
1	“gw mau main game mobile legend bareng lo sore ini”	“saya mau main game mobile legend bareng anda sore ini”

2.11 Stopword Removal

Stopword Removal melakukan proses dalam menghilangkan kata-kata yang tidak memiliki makna atau stop words agar terfokus pada kata-kata yang lebih bermakna[5].

Tabel 2. 5 Contoh Stopwords Removal

No	Sebelum	Sesudah
1	“menurut saya, cuaca hari ini sangat panas dan terik”	"cuaca hari panas terik."

2.12 Stemming

merupakan proses mencari kata dasar untuk memperkecil jumlah indeks yang berbeda dari suatu dokumen, dan juga untuk mengelompokkan kata yang memiliki kata dasar dan arti yang serupa. Contoh Stem (akar kata) adalah kata inti setelah imbuhan dihilangkan (awalan dan akhiran)[4].

Tabel 2. 6 Contoh Stemming

No	Sebelum	Sesudah
1	“saya suka memasak”	"saya suka masak."

2.13 Levenshtein Distance

Levenshtein Distance atau yang biasa disebut dengan *edit distance* adalah suatu metode yang dapat digunakan untuk mengatasi terjadinya kesalahan ejaan/Typo-Checking[8]. *Levenshtein Distance* memiliki 3 operasi yaitu Opeasi Penghapusan, operasi Penyisipan dan operasi Pergantian. algoritma *Levenshtein Distance* dibawah ini[28].

Tabel 2. 7 Pseucode Leveishtein Distance

Step	Description
1.	Set n to be the length of s. Set m to be the length of t. If n = 0, return m and exit. If m = 0, return n and exit. Construct a matrix containing 0..m rows and 0..n columns.
2.	Initialize the first row to 0..n. Initialize the first column to 0..m.
3	Examine each character of s (i from 1 to n).
4	Examine each character of t (j from 1 to m).
5	If s[i] equals t[j], the cost is 0. If s[i] doesn't equal t[j], the cost is 1. Set cell d[i,j] of the matrix equal to the minimum of:
6	a The cell immediately above plus 1: d[i-1,j] + 1. b The cell immediately to the left plus 1: d [i,j-1] + 1. c The cell diagonally above and to the left plus the cost: d [i-1,j-1] + cost.
7	After the iteration steps (3, 4, 5, 6) are complete, the distance is found in cell d[n,m].

2.14 Klasifikasi *Lexicon-Based*

Lexicon-Based merupakan kamus atau leksikon yang digunakan untuk pemilihan kata pada data atau dokumen [19][23]. Klasifikasi berbasis lexicon adalah metode pengklasifikasian teks yang menggunakan daftar kata atau frasa yang telah diberi label sebelumnya sebagai fitur untuk menentukan kategori atau kelas dari teks tersebut. Pendekatan ini dapat diterapkan dalam berbagai konteks, termasuk analisis sentimen, pengenalan entitas bernama, atau klasifikasi topik.

Dalam konteks analisis sentimen,Setiap kata yang ditemukan dalam *Lexicon-Based* dihitung polaritasnya. Polaritas dapat berupa skor yang menunjukkan tingkat kepositifan atau ke-negatifan kata tersebut. Misalnya, dalam analisis sentimen, *Lexicon* tersebut dapat berisi daftar kata-kata yang dikategorikan sebagai positif atau negatif. Teks kemudian diklasifikasikan berdasarkan distribusi kata-kata yang sesuai dengan kategori sentimen dalam *Lexicon-Based*. Pada proses ini memerlukan Indonesian sentiment(Inset) sebagai acuan pemberian nilai pada polaritas kata. Rumus dapat dilihat dibawah ini [4].

Skor positif

$$\sum_{i=1}^{n} Polaritas\ Positif \tag{2.1}$$

Skor Negatif

$$\sum_{i=1}^{n} Polaritas\ Negatif \tag{2.2}$$

Berdasarkan skor sentimen, kalimat dapat diklasifikasikan sebagai memiliki sentimen positif atau negatif. Jika skor sentimen lebih besar atau sama dengan dari nol, maka kalimat dianggap memiliki sentimen positif. Jika skor sentimen lebih kecil dari nol, maka kalimat dianggap memiliki sentimen negatif[4]. Skor polaritas per kata berdasarkan Inset Lexicon yang digunakan. Setiap kata dipasangkan dengan skor yang menunjukkan tingkat sentimen positif, negatif, pada kata tersebut dalam *Inset Lexicon*. Kemudian, skor polaritas keseluruhan dihitung berdasarkan skor polaritas dari setiap

kata yang muncul di dalamnya. Langkah –langkah klasifikasi *Lexicon-Based* dibawah ini. Menentukan kata untuk klasifikasi dengan *Lexicon-Based*[4]:

1. Setiap kata dalam kalimat akan diberi sebuah nilai yakni bernilai sesuai dengan polaritas yang sudah ditetapkan di kamus untuk kata positif dan negatif. Kemudian didapatkan total kata bersentimen pada kalimat tersebut.
2. Penanganan kata negasi: kata negasi seperti kata “tidak” pada kalimat contoh “Anda tidak bahagia” akan membalikkan orientasi sentimen. Dengan kata lain, kata bersentimen positif “bahagia” akan diberi nilai negative.
3. Pemberian skor pada kalimat: Skor digunakan untuk menentukan apakah sebuah kalimat bersentimen positif atau bersentimen negative.

Menurut Dian Noviani Syafar (2016) dalam Bahasa Indonesia terdapat empat kata negasi yang lazim digunakan yaitu tak atau tidak, bukan, jangan, dan belum[17]. Adapun pseucode yang dilakukan dalam penanganan negasi pada tabel dibawah ini[22]

Tabel 2. 8 Pseucode Penanganan Negasi

```
Let children(n) be the function that returns the children nodes of node n,
let leaf(node) be the function that returns true if node is a leaf let negation(t) be the
function that returns true if t is a negation word,
let score(t) be the sentiment score of term t
let rLeaves(n) be the leaves of the right sibling nodes of n.
1: procedure NH(node)
2:       if isLeaf(node) then
3:            if isNegation(node) then
4:                 score(node) ← 0
5:                 for each n in rLeaves(node) do
6:                       score(n) ← score(n)(−1)
7:       else
8:            for each t in children(node) do
9:          NH(t)
```

2.15 Confusion Matrix

Confusion matrix merupakan suatu metode yang digunakan untuk melakukan perhitungan akurasi pada konsep data mining[4]. Yang mana elemen diantaranya *Precision, Recall,* dan *Accuracy* seperti dibawah ini.

Aktual	Prediksi	
	Bullying	Non-Bullying
Bullying	**TP**	**FN**
Non Bullying	**FP**	**TN**

$$Recall = \frac{TP}{FN+TP} \quad (2.3)$$

$$Precision = \frac{TP}{FP+TP} \quad (2.4)$$

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (2.5)$$

Accuracy digunakan dalam menyatakan persentase jumlah tuple pada data uji yang telah diklasifikasikan dengan benar. *Recall* merupakan ukuran kelengkapan dari besar persentase tuple non-bullying yang dilabeli non-bullying. *Precision* merupakan ukuran kepastian dari besar persentase tuple yang benar dilabeli non-bullying[16].dengan keterangan pada *Confusion Matrix* seperti dibawah ini.

a. *True Positives* (TP): Komentar yang bullying pada label asli dan diklasifikasikan sebagai bullying.
b. *True Negatives* (TN): Komentar yang non-bullying pada label asli dan diklasifikasikan sebagai non-bullying.
c. *False Positives* (FP): Komentar yang non-bullying pada label asli tapi diklasifikasikan sebagai bullying.
d. *False Negatives* (FN): Komentar yang bullying pada label asli tapi diklasifikasikan sebagai non-bullying.

2.16 Python

Bahasa pemograman Python pertama kali dikembangkan pada tahun 1991 oleh guido ve Rossum, seorang programmer Belanda. Python adalah bahasa pemograman interpretatif yang dapat digunakan di berbagai platform dengan filosofi perancangan yang berfokus pada tingkat keterbacaan kode dan merupakan salah satu bahasa popular yang berkaitan dengan *Data Science, Machine learning* dan *Internet of Things*(IoT)[30].

2.17 Metode Beriorientasi Fungsi

Pada tahap-tahap yang dilakukan Metode beriorientasi fungsi dimana Perangkat lunak dianggap sebagai kumpulan fungsi atau proses transformasi data sebagai berikut[29].

1 Data masukan
2 Proses transformasi
3 Data keluaran/hasil transformasi
4 Keadaan awal atau akhir
5 Perubahan (dari keadaan awal ke akhir)
6 Aksi untuk mengubah keadaan
7 Sudut pandang pengembangan dengan metode ini adalah aspek fungsional dan perilaku sistem. Pengembang harus mengetahui fungsi-fungsi atau proses-proses apa saja yang ada di dalam sistem, data apa saja yang menjadi masukkannya, dimana data tersebut disimpan, transformasi apa yang akan dilakukan terhadap data tersebut, dan apa yang menjadi hasil transformasinya. Selain itu, pengembang, pengembang harus mngetahui keadaan, perubahan, kondisi dan aksi dari sistem.
8 Strategi utama untuk menangani kompleksitas pembangunan perangkat lunak adalah dekomposisi permasalahan menjadi bagian-bagian kecil yang dapat dikelola.
9 Pada metode beriorientasi fungsi atau diagram aliran data(DFD), dekomposisi permasalahan dilakukan berdasarkan fungsi, mulai dari diagram konteks sampai proses-proses yang paling kecil.

2.18 Data Flow Diagram (DFD)

Data Flow Diagram (DFD) adalah alat pembuatan model yang memungkinkan professional sistem untuk menggambarkan sistem sebagai suatu jaringan proses fungsional yang dihubungkan satu sama lain dengan alur data, baik secara manual maupun komputerisasi. DFD beriorientasi pada konsep dekomposisi dapat digunakan untuk penggambaran analisa maupun rancangan sistem yang mudah dikomunikasikan oleh professional sistem kepada pemakai maupun pembuat program.

1 Teknik penggambaran Data Flow Diagram(DFD)

Langkah-langkah yang perlu dilakukan untuk membuat DFD sebagai berikut.

a) Idetifikasi terlebih dahulu semua entitas luar yang terlihat di sistem.
b) Identifikasi semua input dan output yang terlibat dengan entitas luar.
c) Buat diagram konteks, diagram ini adalah diagram level tertinggi dari DFD yang menggambarkan hubungan sistem dengan lingkungan luarnya.