BAB II

LANDASAN TEORI

2.1 Analisis Sentimen

Analisis sentimen adalah salah satu bidang pada *text mining* yang menganalisa sebuah pendapat, opini, evaluasi, sentimen, sikap atau penilaian seseorang terhadap individu, kelompok, produk, organisasi, masalah, peristiwa atau topik. (Sabily et al., 2019). Analisis Sentimen juga bisa diartikan sebagai riset komputasional dari sebuah opini dan emosi yang diekspresikan secara tekstual. Analisis Sentimen biasanya digunakan untuk menganalisa produk atau organisasi dalam rangka peningkatan kualitas dari produk atau organisasi nantinya. (Gunawan et al., 2017)

Analisis Sentimen terbagi menjadi dua kategori yaitu *Coarse-Grained Sentiment Analysis* dan *Fined-Grained Sentiment Analysis* (Sabily et al., 2019).

2.1.1 *Coarse-Grained sentiment analysis*

Coarse-Grained sentiment analysis adalah klasifikasi yang berorientasi pada sebuah dokumen secara keseluruhan. Klasifikasi jenis ini dibagi pada tiga yaitu positif, netral dan negatif.

2.1.2 *Fined-Grained sentiment analysis*

Fined-Grained sentiment analysis adalah klasifikasi yang orientasinya lebih spesifik, yaitu pada kalimat di sebuah dokumen.

2.2 Text Mining

Menurut (Pravina et al., 2019) yang mengutip dari (Adiwijaya, 2006) *Text Mining* merupakan proses mencari informasi, relasi dan fakta yang terdapat dalam sebuah teks pada proses analisa data. *Text Mining* dilakukan pada data berjumlah besar, dimensi besar, struktur teks yang kompleks dan tidak lengkap dan data yang memiliki *noise* yang tinggi.

Pada prosesnya, *Text mining* melalui beberapa tahapan, seperti ekstraksi teks, beberapa teknik tertentu, *pre-processing text,* pembobotan teks, serta analisis suatu teks. (Pravina et al., 2019)

2.3 Data Pre-Processing

Data Pre-Processing adalah langkah yang dilakukan setelah *dataset* terkumpul untuk membersihkan data, sehingga proses pada *machine learning* menjadi lebih cepat dan akurat. (Nurrohmat & SN, 2019). Tahapan pada *Data Pre-Processing* berbeda-beda pada setiap kasusnya, tergantung data ataupun model yang dibuat. Berikut sebagian dari langkah-langkah pada *Data Pre-Processing.*

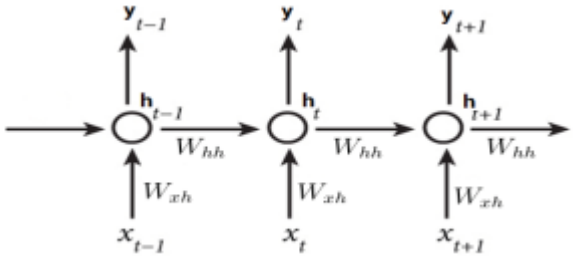
1. *Data Cleaning,* adalah proses yang dilakukan untuk membersihkan data dari karakter yang non-alfabetis. Contoh karakter yang akan dibersihkan adalah titik (.), koma (,), tanda tanya (?), tanda seru (!) dan karakter-karakter non-alfabetis lainnya. (Firmansyah et al., 2020)
2. *Stopword Removal,* adalah proses menghapus kata yang tidak memiliki arti dan tidak memberikan dampak pada proses pengklasifikasian sentimen.

Contoh dari *stopword* adalah ke, dari, pada, antara, dan kata-kata lainnya yang tidak memiliki arti. (Patel et al., 2021)

3. *Case Folding,* adalah proses mengubah bentuk huruf dari huruf kapital ke huruf kecil, sehingga huruf yang diterima hanya huruf ‘a’ sampai ‘z’ (Firmansyah et al., 2020).
4. *Stemming,* adalah proses untuk menghapus karakter yang tidak diperlukan yang bergabung dengan sebuah kata. Contohnya, kata “membantu” setelah proses *stemming* menjadi “bantu” (Patel et al., 2021).
5. *Tokenizing,* adalah proses untuk membagi kalimat menjadi beberapa bagian yang disebut dengan token. Contohnya, kalimat “semoga lebih baik lagi”, setelah proses *tokenization* menjadi “semoga, lebih, baik, lagi” (Patel et al., 2021).

2.4 Recurrent Neural Network (RNN)

Recurrent Neural Network (RNN) adalah salah satu model yang populer dalam *Natural Language Processing* (NLP). Dalam RNN, pemrosesan dilakukan berulang-ulang pada setiap elemennya, dengan keluaran yang dipengaruhi oleh hasil dari proses pada elemen sebelumnya. Gambar 2.1 menunjukkan arsitektur dari RNN (Yennimar et al., 2019).



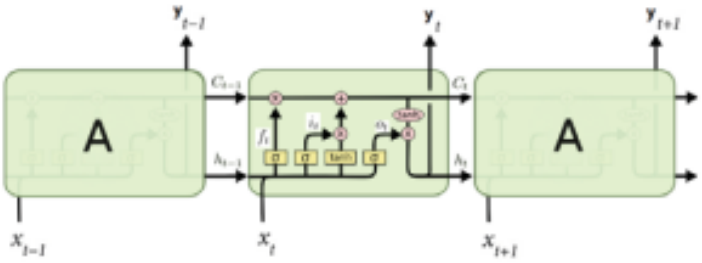
Gambar 2.1 Arsitektur *Recurrent Neural Network* (RNN)

(Yennimar et al., 2019)

Untuk menyimpan data dari proses sebelumnya, RNN melakukan perulangan dalam arsitekturnya sehingga informasi dari proses sebelumnya dapat tersimpan. Dalam arsitekturnya, *hidden layer* dalam RNN memakai fungsi aktivasi *sigmoid*.

2.5 Long-Short Term Memory (LSTM)

Dalam segi arsitektur, *Long-Short Term Memory* (LSTM) tidak ada perbedaan dengan RNN. LSTM dapat melakukan komputasi *hidden state* yang berfungsi untuk menyimpan *long-term dependencies.* LSTM merupakan salah satu arsitektur *neural network* yang cukup baik digunakan dalam penanganan data yang bersifat sekuensial. Proses pada RNN dapat dilihat pada gambar 2.2 (Yennimar et al., 2019).



Gambar 2.2 Arsitektur LSTM (Yennimar et al., 2019)

Pada gambar 2.2, setiap garis adalah jalan untuk vektor dari satu *node output* menuju *node input* lain. Lingkaran-lingkaran berwarna merah muda mewakili proses yang dilakukan, seperti penambahan vector. Sedangkan kotak-kotak berwarna kuning mewakili *learning layer* dari *neural networks.* Garis bercabang menunjukkan data disalin dan dikirimkan ke tempat lain, sedangkan garis yang menyatu menunjukkan penyatuan data.

2.6 Confusion Matrix

Confusion Matrix merupakan sebuah alat yang memberikan informasi tentang perbandingan hasil prediksi dari model klasifikasi dengan klasifikasi sebenarnya (Lionovan et al., 2017). Pada *confusion* matrix, terdapat 4 (empat) bagian, yaitu :

1. *True Positive* (TP) adalah kondisi dimana model memberikan prediksi klasifikasi benar dan klasifikasi sebenarnya juga benar;
2. *True Negative* (TN) adalah kondisi dimana model memberikan prediksi klasifikasi salah dan klasifikasi sebenarnya juga adalah salah;
3. *False Positive* (FP) adalah kondisi dimana model memberikan prediksi klasifikasi benar tapi klasifikasi sebenarnya adalah salah;

4. *False Negative* (FN) adalah kondisi dimana model memberikan prediksi klasifikasi salah tapi klasifikasi sebenarnya adalah benar.

Tabel 2.1 menunjukan ilustrasi dari bagian-bagian pada *confusion matrix*.

Tabel 2.1 *Confusion Matrix*

	Actual True	*Actual False*
Predicted True	*True Positive*	*False Positive*
Predicted False	*False Negative*	*True Negative*

Berdasarkan Tabel 2.1 akan didapatkan nilai dari akurasi, presisi dan *recall*. Akurasi merupakan perbandingan antara prediksi benar pada bagian positif dan negatif dengan keseluruhan data. Akurasi dihitung dengan menggunakan persamaan 2.1.

$$Akurasi = \frac{TP + TN}{TP + FP + TN + FN} \qquad (2.1)$$

Presisi merupakan perbandingan antara prediksi benar pada bagian positif dengan keseluruhan data yang diprediksi positif. Presisi dihitung dengan menggunakan persamaan 2.2.

$$Presisi = \frac{TP}{TP + FP} \qquad (2.2)$$

Recall merupakan perbandingan antara prediksi benar pada bagian positif dengan keseluruhan data positif. Recall dihitung dengan menggunakan persamaan 2.3.

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

2.7 State of The Art

2.7.1 Penelitian Terkait

Beberapa peneliti sebelumnya telah melakukan penelitian terkait analisis sentimen menggunakan *long-short term memory*. (Lionovan et al., 2017) melakukan penelitian analisis sentimen pada kuesioner umpan balik menggunakan algoritma *long-short term memory*. Pada penelitian ini, hasil pengambilan kuesioner umpan balik akan diklasifikasikan menjadi 2 kelas, yaitu positif dan negatif. Pada penelitian ini, Lionovan, dkk. menggunakan *word2vec* pada *feature word embedding* dan *long-short term memory* pada bagian klasifikasi. Pengujian dilakukan dengan *confusion matrix* dengan rata-rata akurasi 89,16%. Akurasi penelitian masih dapat ditingkatkan dengan meningkatkan variasi data dan menambah jumlah data. Perbedaan penelitian ini dengan penelitian yang dilakukan adalah pada *feature word embedding*, dimana pada penelitian yang dilakukan menggunakan *GloVe*.

Kemudian, penelitian (Idris et al., 2019) melakukan analisis sentimen menggunakan *long-short term memory* untuk klasifikasi dan *word2vec* untuk *word embedding* pada klasifikasi konten radikalisme dari *tweet* berbahasa Indonesia. Pengujian dilakukan dengan metode k-fold cross validation dengan akurasi sebesar 81,60%. Penelitian ini dapat ditingkatkan akurasinya dengan melakukan pengujian untuk parameter yang digunakan, sehingga paramater pada model merupakan parameter terbaik dari pengujiannya. Perbedaan dengan penelitian yang dilakukan terletak pada bagian *word embedding* di mana pada penelitian yang dilakukan menggunakan *GloVe*.

Penelitian (Firmansyah et al., 2020) juga melakukan analisis sentimen tetapi dengan algoritma RNN untuk klasifikasi dan *word2vec* sebagai *word embedding* pada Klasifikasi Kalimat Ilmiah. Pada penelitian ini, Firmansyah, dkk. juga mencoba empat jenis metode optimasi, yaitu Adam, SGD, Adadelta dan Adamax. Hasilnya, optimasi dengan SGD mendapat tingkat akurasi terbaik yaitu sebesar 77,48%. Persebaran data yang kurang variatif menjadi salah satu alasan kurangnya akurasi pada ssetiap jenis optimasinya, sehingga perlu diuji peningkatan persebaran data untuk meningkatkan akurasinya. Perbedaan dengan penelitian yang dilakukan terletak pada metode klasifikasi dan word2vec, dimana pada penelitian yang dilakukan secara berturut-turut menggunakan LSTM dan *GloVe*.

Terakhir, penelitian (Temizkan, 2020) juga melakukan analisis sentimen menggunakan RNN sebagai metode klasifikasi pada *review* hotel. Hasilnya, akurasi yang dihasilkan adalah 79,85%. Perbedaan dengan penelitian yang dilakukan adalah pada penggunaan *word embedding*, dimana penelitian yang dilakukan menggunakan *word embedding* dari Keras dan penelitian penulis menggunakan *GloVe.*

2.7.2 Matriks Penelitian

Tabel 2.2 Matriks Penelitian

No	Judul Paper	Penulis	Algoritma	Fitur	Akurasi
1	Sentiment analysis for opinion IESM product with recurrent neural network approach based on long short term memory	Yennimar, dkk	LSTM		91,01%
2	Klasifikasi Teks Laporan Masyarakat pada Situs Lapor! menggunakan Recurrent Neural Network	Rozi, I.F., dkk	LSTM		88,82%
3	Klasifikasi Kalimat Ilmiah menggunakan Recurrent Neural Network	Firmansyah, M.R., dkk	LSTM	Word2vec	77,48%
4	Klasifikasi Topik dan Analisa Sentimen terhadap Kuesioner Umpan Balik menggunakan Metode Long Short Term Memory	Lionovan, D.A., dkk	LSTM	Word2vec	89,16%
5	Sentiment Analysis of Novel Review Using Long Short Term Memory Method	Nurrohmat, Muh Amin dan Azhari S.N.	LSTM		72,85%
6	Classification of Radicalism Content from Twitter Written in Indonesian Language using Long Short Term Memory	Idris, N.O., dkk	LSTM	Word2vec	81,60%
7	Klasifikasi Sentimen Ulasan Film Indonesia dengan Konversi Speech-to-Text (STT) Menggunakan Metode Convolutional Network (CNN)	Shafirra, N.A. dan Irhamah	CNN		83,7%
8	Sentiment Analysis using RNN and Google Translator	Mahajan, Dipti dan Chaudhary, D.K.	RNN		90,3%
9	Sentiment Analysis of US Airlines Tweets using LSTM/RNN	Monika, R., dkk.	LSTM	*GloVe*	76%
10	Sentiment Analysis for Hotel Reviews with Recurrent Neural Network Architecture	Karaoglan, K.M., dkk.	RNN	*Keras Word Embedding*	79,85%
11	Sentiment Analysis on Movie Review Using Deep Learning RNN Method	Patel Priya, dkk	RNN		94,61%