

**BIDIRECTIONAL GRU DENGAN ATTENTION MECHANISM
PADA ANALISIS SENTIMEN PLN MOBILE**

THESIS

**Oleh:
MOH. AINUR ROHMAN
NIM. 210605210005**



**PROGRAM STUDI MAGISTER INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2023**

HALAMAN PENGAJUAN

**BIDIRECTIONAL GRU DENGAN ATTENTION MECHANISM
PADA ANALISIS SENTIMEN PLN MOBILE**

THESIS

**Diajukan kepada:
Universitas Islam Negeri Maulana Malik Ibrahim Malang
Untuk memenuhi Salah Satu Persyaratan dalam
Memperoleh Gelar Magister Komputer (M.Kom)**

**Oleh:
MOH. AINUR ROHMAN
NIM. 210605210005**

**PROGRAM STUDI MAGISTER INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2023**

HALAMAN PENGAJUAN

**BIDIRECTIONAL GRU DENGAN ATTENTION MECHANISM
PADA ANALISIS SENTIMEN PLN MOBILE**

THESIS

**Diajukan kepada:
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang
Untuk memenuhi Salah Satu Persyaratan dalam
Memperoleh Gelar Magister Komputer (M.Kom)**

**Oleh:
MOH. AINUR ROHMAN
NIM. 210605210005**

**PROGRAM STUDI MAGISTER INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2023**

HALAMAN PERSETUJUAN

**BIDIRECTIONAL GRU DENGAN ATTENTION MECHANISM
PADA ANALISIS SENTIMEN PLN MOBILE**

THESIS

**Oleh:
MOH. AINUR ROHMAN
NIM. 210605210005**

Telah diperiksa dan disetujui untuk diuji:
Tanggal: 7 Maret 2023

Pembimbing I,



Prof. Dr. Suhartono, M.Kom
NIP. 19680519 200312 1 001

Pembimbing II,



Dr. Totok Chamidy, M.Kom
NIP. 19691222 200604 1 001

Mengetahui,
Ketua Program Studi Magister Informatika
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang




Dr. Totok Chamidy, M.Kom
NIP. 19691222 200604 1 001

HALAMAN PENGESAHAN

**BIDIRECTIONAL GRU DENGAN ATTENTION MECHANISM
PADA ANALISIS SENTIMEN PLN MOBILE**

THESIS

Oleh:
MOH. AINUR ROHMAN
NIM. 210605210005

Telah Dipertahankan di Depan Dewan Penguji Thesis
dan Dinyatakan Diterima Sebagai Salah Satu Persyaratan
Untuk Memperoleh Gelar Magister Komputer (M.Kom)
Tanggal: 16 Maret 2023

Susunan Dewan Penguji

Tanda Tangan

Penguji Utama : Dr. Fachrul Kurniawan ST., M.MT., IPM
NIP. 19771020 200912 1 001

Ketua Penguji : Dr. M. Ainul Yaqin, M.Kom
NIP. 19761013 200604 1 004

Sekretaris Penguji : Dr. Totok Chamidy, M.Kom
NIP. 19691222 200604 1 001

Anggota Penguji : Prof. Dr. Suhartono, M.Kom
NIP. 19680519 200312 1 001

()

()

()

()

Mengetahui,
Ketua Program Studi Magister Informatika
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang



Yoyo Crysdian
NIP. 19740424 200901 1 008

**SURAT PERNYATAAN
KEASLIAN TULISAN**

Saya yang bertanda tangan di bawah ini:

Nama : Moh. Ainur Rohman
NIM : 210605210005
Program Studi : Magister Informatika
Fakultas : Sains dan Teknologi

Menyatakan dengan sebenarnya bahwa Thesis yang saya tulis ini benar-benar merupakan hasil karya saya sendiri, bukan merupakan pengambilalihan data, tulisan atau pikiran orang lain yang saya akui sebagai hasil tulisan atau pikiran saya sendiri, kecuali dengan mencantumkan sumber cuplikan pada daftar pustaka.

Apabila dikemudian hari terbukti atau dapat dibuktikan Thesis ini hasil jiplakan, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Malang, 16 Maret 2023

Yang membuat pernyataan



Moh. Ainur Rohman

NIM. 210605210005

MOTTO

Imam Syafii RA

لَا يُدْرَكُ الْعِلْمُ إِلَّا بِالصَّبْرِ عَلَى الضَّرِّ

“Ilmu tidak akan didapati kecuali dengan bersabar atas kesulitan”

Imam Ath-Thabari RA

يُنْبَغِي لِطَالِبٍ أَنْ لَا يَدَعَ طَلَبَ الْعِلْمِ حَتَّى الْمَمَاتِ

“Sepantasnya seorang pelajar tidak meninggalkan menuntut ilmu hingga mati”

Imam Said bin Jubair RA

لَا يَزَالُ الرَّجُلَ عَالِمًا مَا تَعَلَّمَ

“Seseorang masih disebut alim selama dia belajar”

Imam Hasan Al-Basri

إِذَا اسْتَرَدَّلَ اللَّهُ عَبْدًا زَهَّدَهُ فِي الْعِلْمِ

“Jika Allah ingin menghinakan seorang hamba, Dia akan menjauhkannya dari ilmu”

HALAMAN PERSEMBAHAN

Puji syukur penulis panjatkan kehadirat Allah Subhanahu Wa Ta'ala, yang telah memberikan nikmat sehat, iman, Islam, rahmat dan hidayah sehingga penulis masih diberikan kesempatan untuk menyelesaikan Thesis ini, yang digunakan sebagai salah satu syarat untuk memperoleh gelar Magister Komputer (M.Kom) di Program Studi Magister Informatika, Fakultas Sains dan Teknologi, Universitas Islam Negeri Maulana Malik Ibrahim Malang. Walaupun jauh dari kata sempurna, namun penulis bangga telah mencapai pada titik ini, yang akhirnya Thesis ini bisa selesai di waktu yang tepat. Saya persembahkan Thesis ini untuk:

1. Tuhan semesta alam Allah Subhanahu Wa Ta'ala, yang telah memberikan rahmat dan karuna-Nya serta tidak lupa sholawat dan salam kepada Baginda Nabi Besar Muhammad Shallallahu Alaihi Wa Salam, sehingga saya dapat menyelesaikan Thesis ini.
2. Ibunda saya "Iin Maspufah" dan Ayahanda saya "Ir. Suyoko", yang selalu memberikan dukungan berupa doa, material, motivasi, pengorbanan dan nasehat secara ikhlas, lahir dan batin.
3. Seluruh dosen Program Studi Magister Informatika Universitas Islam Negeri Maulana Malik Ibrahim Malang yang tiada lelah memberikan ilmunya kepada saya.
4. Saudara kandung saya, Achmad Nurman Al-Farizi, yang telah mendoakan dan memberikan dukungan penuh kepada saya.
5. Sahabat saya Luban Abdi Susanto, S.Kom., Firman Wahyu Romadhan, A.Md. yang telah memberikan dukungan secara material maupun non-material.
6. Seluruh rekan Mahasiswa satu angkatan Program Studi Magister Informatika Universitas Islam Negeri Maulana Malik Ibrahim Malang yang telah memberikan banyak pengalaman berupa saran dan kritik yang membangun pada Thesis ini.
7. Semua pihak yang tidak dapat penulis sebut satu persatu, yang telah membantu dalam menyelesaikan Thesis ini.

KATA PENGANTAR



Alhamdulillah, segala puji bagi Allah Subhanahu Wa Ta'ala atas segala limpahan rahmat, rida, hidayah dan inayah-Nya sehingga Thesis dengan judul **“BIDIRECTIONAL GRU DENGAN ATTENTION MECHANISM PADA ANALISIS SENTIMEN PLN MOBILE”** ini penulis dapat selesaikan dengan baik dan lancar. Shalawat serta salam tetap turunkan kepada Baginda Nabi Besar Muhammad Shallallahu Alaihi Wa Salam yang telah menuntun kita dari zaman kegelapan menuju zaman yang terang benderang yakni Dienul Islam.

Thesis ini disusun untuk memenuhi persyaratan memperoleh gelar Magister Komputer (M.Kom) di Program Studi Magister Informatika Fakultas Sains dan Teknologi Universitas Negeri Islam Maulana Malik Ibrahim Malang. Dengan segala keterbatasan yang penulis miliki dan masih banyak kekurangan yang harus diperbaiki. Semoga hasil penelitian ini dapat berguna, khususnya bagi dunia pendidikan.

Pada penulisan Thesis ini, penulis banyak mendapat bantuan dari berbagai pihak. Oleh karena itu, ucapan terima kasih penulis sampaikan kepada:

1. Allah Subhanahu Wa Ta'ala atas segala rahmat dan hidayah-Nya sehingga Thesis ini dapat terselesaikan dengan baik.
2. Bapak Prof. Dr. H. M. Zainuddin, MA., selaku Rektor Universitas Islam Negeri Maulana Malik Ibrahim Malang.
3. Ibu Dr. Sri Harini, M.Si., selaku Dekan Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang.
4. Bapak Dr. Cahyo Crysdiyan, selaku Kepala Program Studi Magister Informatika Universitas Islam Negeri Maulana Malik Ibrahim.
5. Bapak Prof. Dr. Suhartono, M.Kom., dan Bapak Dr. Totok Chamidy, M.Kom., selaku Dosen Pembimbing yang telah meluangkan waktu serta dengan penuh kesabaran telah memberikan bimbingan dalam penyusunan Thesis.

6. Bapak Dr. Fachrul Kurniawan ST., M.MT., IPM dan Bapak Dr. M. Ainul Yaqin, M.Kom., selaku Dosen Penguji yang telah memberikan kritik dan saran yang membangun dalam penelitian Thesis.
7. Ibunda saya “Iin Maspufah” dan Ayahanda saya “Ir. Suyoko”, tercinta dengan penuh kesabaran dan pengorbanannya selalu memberikan dukungan dorongan, bantuan material maupun non-material agar penulis dapat menyelesaikan studi.
8. Teman-teman Magister Informatika Universitas Islam Negeri Maulana Malik Ibrahim Malang yang telah membantu dan memberi semangat dalam menyelesaikan Thesis.
9. Keluarga besar PLN Jember terutama Bapak Subechan dan Bapak Hafid yang sudah saya anggap bapak sendiri yang telah memberikan dukungan semangat.
10. Sahabatku Luban Abdi Susanto, S.Kom., Firman Wahyu Romadhan, A.Md., yang telah memberikan dukungan secara material maupun non-material.
11. Semua pihak yang tidak bisa penulis sebutkan satu persatu, terimakasih atas bantuan dan dukungannya.

Penulis menyadari Thesis ini masih jauh dari kata sempurna, karena hal tersebut tidak lepas dari kelemahan dan keterbatasan penulis. Akhirnya penulis berharap agar Thesis ini berguna sebagai tambahan Ilmu Pengetahuan serta dapat memberikan manfaat bagi semua pihak dan dijadikan implikasi selanjutnya bagi mahasiswa.

Malang, 16 Maret 2023

Penulis

DAFTAR ISI

| | |
|-----------------------------------|-------------|
| HALAMAN PENGAJUAN | i |
| HALAMAN PERSETUJUAN | iii |
| HALAMAN PENGESAHAN | iv |
| SURAT PERNYATAAN | v |
| MOTTO | vi |
| HALAMAN PERSEMBAHAN | vii |
| KATA PENGANTAR | viii |
| DAFTAR ISI | x |
| DAFTAR GAMBAR | xiii |
| DAFTAR TABEL | xiv |
| ABSTRAK | xv |
| ABSTRACT | xvi |
| ملخص | xvii |
| | |
| BAB I PENDAHULUAN | 1 |
| 1.1 Latar Belakang | 1 |
| 1.2 Pernyataan Masalah | 4 |
| 1.3 Tujuan Penelitian | 4 |
| 1.4 Manfaat Penelitian | 4 |
| 1.5 Batasan Masalah..... | 5 |
| 1.6 Sistematika Penulisan | 5 |
| | |
| BAB II STUDI PUSTAKA | 7 |
| 2.1 Analisis Sentimen | 7 |

| | |
|---|-----------|
| 2.1.1 Sentimen Klasifikasi (<i>Task</i>) | 8 |
| 2.1.2 <i>Deep Learning (Approaches)</i> | 8 |
| 2.1.3 <i>Document-Based (Level of Analysis)</i> | 9 |
| 2.2 <i>Bidirectional Gated Recurrent Unit (BiGRU)</i> | 10 |
| 2.3 <i>Attention Mechanism (Transformer)</i> | 12 |
| 2.4 Novelty Penelitian..... | 14 |
| 2.5 Kerangka Teori..... | 19 |
| BAB III METODOLOGI PENELITIAN | 21 |
| 3.1 Desain Penelitian..... | 21 |
| 3.2 Pengumpulan Data (<i>Data Gathering</i>) | 22 |
| 3.3 Pengolahan Data Awal (<i>Data Pre-processing</i>)..... | 23 |
| 3.4 Model yang Diusulkan (<i>Proposed Method</i>)..... | 27 |
| 3.5 Eksperimen dan Pengujian Model (<i>Model Test and Experiment</i>) | 31 |
| 3.6 Evaluasi dan Validasi Hasil (<i>Result Evaluation and Validation</i>) | 33 |
| 3.7 Desain Sistem..... | 34 |
| BAB IV HASIL DAN PEMBAHASAN..... | 36 |
| 4.1 <i>Experiment Environment</i> | 36 |
| 4.2 <i>Dataset</i> | 36 |
| 4.3 Indikator Evaluasi | 37 |
| 4.4 <i>Hyperparameter Setting</i> | 38 |
| 4.4.1 <i>Dropout</i> | 38 |
| 4.4.2 <i>Optimizer</i> | 41 |
| 4.4.3 <i>Batch Size</i> | 43 |

| | |
|--|-----------|
| 4.4.4 <i>Units</i> | 46 |
| 4.4.5 <i>Learning Rate</i> | 48 |
| 4.5 Komparatif Eksperimen | 50 |
| 4.5.1 Perbandingan menggunakan dan tanpa <i>global attention</i> | 51 |
| 4.5.2 Perbandingan dengan metode analisis sentimen yang lain | 52 |
| 4.6 Memahami Informasi Penting dalam Al-Quran..... | 55 |
| BAB V PENUTUP | 58 |
| 5.1 Kesimpulan | 58 |
| 5.2 Saran..... | 59 |
| DAFTAR PUSTAKA | 60 |

DAFTAR GAMBAR

| | |
|---|----|
| Gambar 2.1 Konsep analisis sentimen | 7 |
| Gambar 2.2 Model polarity classification pada DOCSA..... | 10 |
| Gambar 2.3 GRU architecture..... | 11 |
| Gambar 2.4 Arsitektur LSTM-GRU + GloVe | 14 |
| Gambar 2.5 Arsitektur CNN-BiLSTM + Doc2Vec | 15 |
| Gambar 2.6 Arsitektur BiLSTM+seninfo-TFIDF | 16 |
| Gambar 2.7 Arsitektur SAM-BiLSTM | 16 |
| Gambar 2.8 Arsitektur BiLSTM-HAN | 17 |
| Gambar 2.9 Arsitektur Glo-Att-BiGRU+Word2vec..... | 18 |
| Gambar 2.10 Kerangka teori | 19 |
| Gambar 3.1 Desain penelitian | 21 |
| Gambar 3.2 Text pre-processing | 23 |
| Gambar 3.3 Proses case folding | 24 |
| Gambar 3.4 Proses tokenization..... | 24 |
| Gambar 3.5 Proses filtering..... | 25 |
| Gambar 3.6 Proses stemming..... | 25 |
| Gambar 3.7 Proses vectorization..... | 26 |
| Gambar 3.8 Arsitektur model Bi-GRU dengan attention mechanism | 27 |
| Gambar 3.9 Arsitektur skip-gram | 28 |
| Gambar 3.10 Arsitektur bidirectional GRU | 29 |
| Gambar 3.11 Arsitektur global attention..... | 31 |
| Gambar 3.12 Desain sistem model..... | 34 |
| Gambar 4.1 Pengujian akurasi dan f1-score pada dropout | 40 |
| Gambar 4.2 Pengujian ROC-AUC dan precision-recall pada dropout | 40 |
| Gambar 4.3 Pengujian akurasi dan F1-Score pada optimizer | 42 |
| Gambar 4.4 Pengujian ROC-AUC dan precision-recall pada dropout | 43 |
| Gambar 4.5 Pengujian akurasi dan f1score pada batch size | 45 |
| Gambar 4.6 Pengujian ROC-AUC dan precision-recall pada batch size..... | 45 |
| Gambar 4.7 Pengujian akurasi dan f1-score pada units | 47 |
| Gambar 4.8 Pengujian ROC-AUC dan precision-recall pada units..... | 47 |
| Gambar 4.9 Pengujian akurasi dan f1-score pada learning rate..... | 49 |
| Gambar 4.10 Pengujian ROC-AUC dan precision-recall pada learning rate..... | 50 |
| Gambar 4.11 Pengujian akurasi dan f1-score pada global attention..... | 51 |
| Gambar 4.12 Pengujian ROC-AUC dan precision-recall pada global attention... | 52 |
| Gambar 4.13 Pengujian akurasi dan f1-score pada metode yang lain | 53 |
| Gambar 4.14 Pengujian ROC-AUC dan precision-recall pada metode yang lain | 53 |
| Gambar 4.15 Kurva ROC-AUC pada perbandingan model..... | 54 |
| Gambar 4.16 Kurva precision-recall pada perbandingan model..... | 54 |

DAFTAR TABEL

| | |
|--|----|
| Tabel 1.1 Manfaat penelitian..... | 4 |
| Tabel 2.1 Novelty penelitian..... | 18 |
| Tabel 3.2 Pengujian dengan dropout..... | 32 |
| Tabel 3.3 Pengujian dengan optimizer..... | 32 |
| Tabel 3.4 Pengujian dengan batch size | 32 |
| Tabel 3.5 Pengujian dengan units | 33 |
| Tabel 3.6 Pengujian dengan learning rate | 33 |
| Tabel 4.1 Contoh dataset..... | 37 |
| Tabel 4.2 Confusion matrix..... | 37 |
| Tabel 4.3 Hyperparameter list..... | 51 |

ABSTRAK

Rohman, Moh. Ainur. 2023. ***Bidirectional GRU dengan Attention Mechanism Pada Analisis Sentimen PLN Mobile***. Thesis. Program Studi Magister Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang. Pembimbing: (I) Prof. Dr. Suhartono, M.Kom. (II) Dr. Totok Chamidy, M.Kom.

Kata kunci: analisis sentimen, attention mechanism, BiGRU, PLN Mobile, word2vec

Analisis sentimen adalah teknik yang digunakan untuk menganalisis pendapat, sentimen, sikap dan emosi orang terhadap entitas seperti layanan, produk, organisasi, peristiwa dan lain-lain. Ada tiga poin utama pada analisis sentimen yaitu task, approach, dan level of analysis. Beberapa pendekatan analisis sentimen banyak diusulkan salah satunya menggunakan deep learning seperti CNN, RNN, LSTM, dan BiGRU. Penelitian ini bertujuan untuk mengusulkan arsitektur analisis sentimen untuk mengatasi ketidakmampuan algoritma deep learning seperti LSTM dan GRU dalam menangkap informasi penting. Arsitektur yang diusulkan yaitu mengkombinasikan Bidirectional GRU (BiGRU) dengan attention mechanism menggunakan word2vec sebagai word embedding. Attention mechanism digunakan untuk menangkap kata yang penting sehingga arsitektur tersebut dapat memahami informasi yang penting. Kemudian, arsitektur yang diusulkan dilakukan perbandingan dengan metode CNN, CNN-GRU, CNN-LSTM, CNN-BiGRU, CNN-BiLSTM dengan menggunakan data ulasan PLN Mobile. Hasil eksperimen menunjukkan bahwa arsitektur analisis sentimen yang diusulkan memiliki akurasi dan f1-score yang lebih tinggi.

ABSTRACT

Rohman, Moh. Ainur. 2023. *Bidirectional GRU with Attention Mechanism on Sentiment Analysis of PLN Mobile*. Thesis. Master's Program in Informatics, Faculty of Science and Technology, Maulana Malik Ibrahim State Islamic University of Malang. Promotor: (I) Prof. Dr. Suhartono, M.Kom. (II) Dr. Totok Chamidy, M.Kom.

Sentiment analysis is a technique used to analyze people's opinions, sentiments, attitudes, and emotions toward entities such as services, products, organizations, events, and others. There are three main points in sentiment analysis: task, approach, and level of analysis. Several sentiment analysis approaches are proposed, including deep learning, such as CNN, RNN, LSTM, and BiGRU. This research proposes a sentiment analysis architecture to overcome the inability of deep learning algorithms such as LSTM and GRU to capture critical information. The proposed architecture combines Bidirectional GRU (BiGRU) with an attention mechanism using word2vec as word embedding. Attention mechanisms are used to capture important words so that the architecture can understand important information. Then, the proposed architecture was compared with CNN, CNN-GRU, CNN-LSTM, CNN-BiGRU, and CNN-BiLSTM methods using PLN Mobile review data. The experimental results showed that the proposed sentiment analysis architecture had higher accuracy and f1-score.

Keywords: attention mechanism, BiGRU, PLN Mobile, sentiment analysis, word2vec

ملخص

رحمن ، موه. اينور. 2023. GRU ثنائي الاتجاه مع آلية الانتباه على تحليل مشاعر الهاتف المحمول PLN. اطروحة. برنامج دراسة ماجستير المعلوماتية ، كلية العلوم والتكنولوجيا ، جامعة مولانا مالك إبراهيم الإسلامية الحكومية ، مالانج. المشرف: (I) سوهارتونو (II) توتوك شاميدي

الكلمات المفتاحية: تحليل المشاعر ، آلية الانتباه ، BiGRU ، PLN Mobile ، word2vec

تحليل المشاعر هو تقنية تستخدم لتحليل آراء الناس ومشاعرهم ومواقفهم وعواطفهم تجاه كيانات مثل الخدمات والمنتجات والمنظمات والأحداث وغيرها. هناك ثلاث نقاط رئيسية في تحليل المشاعر ، وهي المهمة والنهج ومستوى التحليل. تم اقتراح العديد من مناهج تحليل المشاعر ، أحدها يستخدم التعلم العميق مثل CNN و RNN و LSTM و BiGRU. يهدف هذا البحث إلى اقتراح بنية تحليل المشاعر للتغلب على عدم قدرة خوارزميات التعلم العميق مثل LSTM و GRU على التقاط المعلومات الهامة. تجمع البنية المقترحة بين GRU ثنائي الاتجاه (BiGRU) وآلية الانتباه باستخدام word2vec كتضمين للكلمات. تستخدم آليات الانتباه لالتقاط الكلمات المهمة حتى تتمكن البنية من فهم المعلومات المهمة. بعد ذلك ، تمت مقارنة البنية المقترحة مع طرق CNN و CNN-GRU و CNN-LSTM و CNN-BiGRU و CNN-BiLSTM باستخدام بيانات مراجعة PLN Mobile. أظهرت النتائج التجريبية أن بنية تحليل المشاعر المقترحة كانت أعلى دقة ودرجة F1.

BAB I

PENDAHULUAN

1.1 Latar Belakang

Analisis sentimen adalah salah satu bidang kajian di *Natural Language Processing* (NLP) (Han et al., 2020) untuk menganalisis pendapat, sentimen, sikap dan emosi orang terhadap entitas seperti layanan, produk, organisasi, peristiwa dan lain-lain (Bing Liu, 2012). Ada tiga poin utama pada analisis sentimen yaitu *task* (tugas), *approach* (pendekatan) dan *level of analysis* (level analisis) (Ligthart et al., 2021). *Task* mengacu pada tujuan yang ingin dicapai menggunakan teknik analisis sentimen, *approach* mengacu pada metodologi atau cara yang digunakan, sedangkan *level of analysis* mengacu pada tingkatan atau level dalam mengolah teks. Penelitian ini berfokus pada *approach* dalam analisis sentimen menggunakan pendekatan *deep learning*.

Deep learning adalah cabang dari *machine learning* yang menggunakan *deep neural network* (Ligthart et al., 2021). Saat ini *deep learning* telah banyak digunakan pada NLP (W. Li et al., 2020) terutama analisis sentimen dan mencapai hasil yang baik (Xu et al., 2019). Beberapa penelitian terkait analisis sentimen yang menggunakan pendekatan *deep learning* telah diusulkan. Pada penelitian (Rhanoui et al., 2019) mengusulkan model CNN-BiLSTM, model ini mengkombinasikan CNN dan *Bidirectional LSTM* untuk *document-level* analisis sentimen dan untuk *word embedding* menggunakan jenis Doc2vec. Hasil dari penelitian ini mencapai akurasi sebesar 90.66%. Penelitian (Ni & Cao, 2020) mengusulkan model kombinasi antara GRU dan LSTM dengan *word embedding*

menggunakan GloVe, penelitian ini bertujuan untuk mengatasi kekurangan pada RNN yaitu tidak bisa mempelajari *long-term information* pada teks. Hasil dari penelitian ini mencapai akurasi sebesar 87.10%. Penelitian (Xu et al., 2019) mengusulkan model kombinasi *word vector* dengan *sentiment information* dan TF-IDF, untuk algoritma *sequence learning*-nya menggunakan Bi-LSTM. Hasil dari penelitian ini mencapai F1-Score 92.18%.

Berdasarkan penelitian (Ni & Cao, 2020; Rhanoui et al., 2019; Xu et al., 2019), algoritma *deep learning* telah berhasil dalam menghasilkan *context information* dalam *sequence learning*. Namun, algoritma *deep learning* tidak dapat memfokuskan kata-kata yang paling berkontribusi, akibatnya tidak mampu menangkap informasi penting pada *context information*. Menangkap atau memahami informasi penting adalah suatu keharusan seperti pada QS. Shad ayat 29:

كُتِبَ أَنْزَلْنَاهُ إِلَيْكَ مُبْرَكًا لِيَذَّبَرُوا أَيَّتَهُ وَيَتَذَكَّرَ أُولُوا الْأَلْبَابِ

Artinya:

“Kitab (Al-Qur’an) yang Kami turunkan kepadamu penuh berkah agar mereka menghayati ayat-ayatnya dan agar orang-orang yang berakal sehat mendapat pelajaran.”

Menurut tafsir wajiz (Kementerian Agama RI), Allah menurunkan Al-Qur’an kepada Nabi Muhammad yang berisi banyak manfaat dan kebaikan. Dan Allah menurunkannya agar manusia menghayati dan memahami ayat-ayatnya dan agar orang-orang yang berakal sehat menggunakan akal budinya untuk mendapat pelajaran darinya dan mengamalkan kandungannya.

Untuk mengatasi masalah ini, penulis mengusulkan arsitektur yang diberi nama “Glo-att-bigru+word2vec” yang merupakan singkatan dari (Bidirectional GRU menggunakan *Global Attention* dan *Word2vec*). Arsitektur ini menggunakan *word2vec* untuk *word embedding*, Bidirectional GRU (BiGRU) untuk *sequence learning*. GRU dipilih karena dapat mengatasi *vanishing* dan *exploding gradient* pada RNN dan lebih sederhana dari *Long-Short Term Memory* (LSTM) (Ni & Cao, 2020) sedangkan Bi-GRU merupakan kombinasi dari dua *hidden layer* yaitu *forward* dan *backward*, dengan begini dapat mengakses konteks sebelumnya dan selanjutnya. *Global attention* digunakan untuk menangkap informasi penting dari *context information*.

Ide dasar dari arsitektur yang diusulkan adalah, *Global attention* digunakan untuk memberikan fokus (*attend*) yang berbeda dari *context information* yang dihasilkan oleh Bi-GRU dengan cara memberikan *weight* (bobot) yang berbeda ke dalam *variable-length* pada *sequence learning*. Kemudian fitur yang diekstraksi dari *global attention* digabungkan dan diklasifikasikan menggunakan *softmax classifier* sehingga model dapat menangkap informasi penting dari *sequence*. Model yang diusulkan akan diujikan menggunakan *dataset* ulasan PLN Mobile, kemudian dilakukan *hyperparameter setting* untuk dibandingkan dan membandingkan dengan model analisis sentimen yang lain.

1.2 Pernyataan Masalah

Berdasarkan latar belakang yang dibahas, algoritma *deep learning* seperti RNN, LSTM hanya dapat menangkap *context information* dan tidak dapat menangkap informasi penting dalam *sequence learning*. Oleh karena itu, rumusan masalah yang akan dijadikan pokok bahasan adalah “Seberapa baik arsitektur “Glo-att-bigru+word2vec” untuk menangkap informasi penting dari *context information* yang dihasilkan Bi-GRU?”

1.3 Tujuan Penelitian

Berdasarkan latar belakang yang dibahas, tujuan dari penelitian ini adalah:

1. Mengatasi ketidakmampuan algoritma *deep learning* seperti RNN, LSTM, dan GRU dalam menangkap informasi penting, menggunakan *attention mechanism*.
2. Menguji arsitektur yang diusulkan.

1.4 Manfaat Penelitian

Manfaat dari penelitian adalah sebagai berikut:

Tabel 1.1 Manfaat penelitian

| Bagi | Manfaat |
|---------------|--|
| Manajemen PLN | <ul style="list-style-type: none"> - Mengetahui loyalitas, kepuasan, psikologi, pengalaman pelanggan terkait menggunakan PLN Mobile. - Mendapatkan <i>insight</i> dari <i>feedback</i> pelanggan sebagai dasar evaluasi dan pembuatan <i>roadmap</i> promosi ke depan berdasarkan analisis sentimen. |
| Akademis | <ul style="list-style-type: none"> - Melahirkan khazanah pengetahuan yang baru, dengan munculnya arsitektur “glo-att-bigru+word2vec”. |

1.5 Batasan Masalah

Agar lingkup penelitian tidak meluas, maka penelitian ini terdapat batasan masalah sebagai berikut:

1. Dataset berasal dari *google playstore* yakni ulasan aplikasi PLN Mobile.
2. Label klasifikasi hanya ada dua, yakni: Positif dan Negatif.
3. *Word embedding* menggunakan Word2Vec.
4. *Attention mechanism* menggunakan *Global attention*.

1.6 Sistematika Penulisan

Tesis ditulis dengan mengikuti sistematika penulisan sebagai berikut:

Bab 1: Pendahuluan

Bab ini memuat mengenai latar belakang (*research background*), *research problem*, pernyataan masalah (*research question*), tujuan penelitian (*research objective*) sebagai kerangka utama penelitian. Dan ditambah tujuan penelitian, batasan masalah, dan sistematika penulisan.

Bab 2: Studi Pustaka

Bab ini menguraikan teori-teori dan konsep-konsep ilmiah yang akan digunakan penelitian melalui sumber-sumber yang kredibilitasnya dapat dipertanggung jawabkan.

Bab 3: Metodologi Penelitian

Bab ini menjelaskan cara / alur penelitian yang dilakukan secara sistematis. Pada tesis ini diuraikan mengenai arsitektur yang diusulkan dan beberapa skema pengujian yang akan dilakukan

Bab 4: Hasil dan Pembahasan

Bab ini memaparkan hasil penelitian yang dilakukan serta menyajikan hasil dari beberapa skema pengujian yang sudah dilakukan.

Bab 5: Penutup

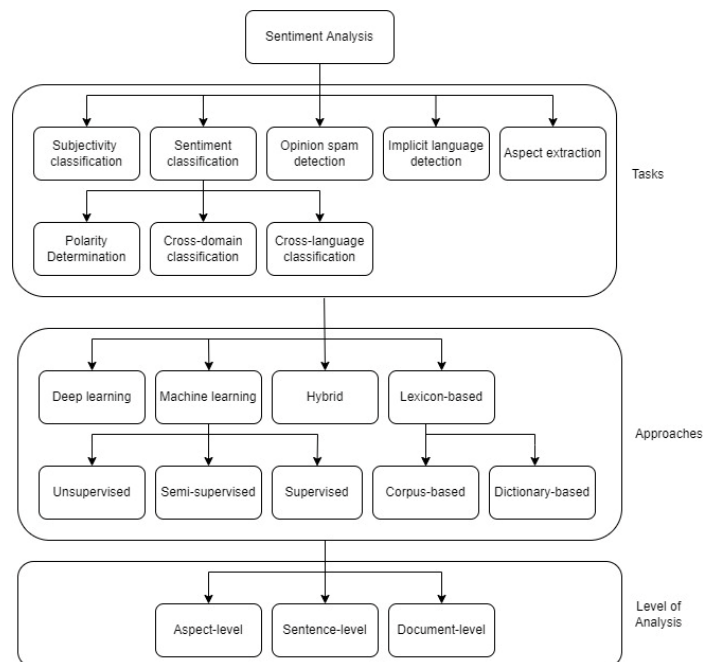
Bab ini merupakan kesimpulan dan menjawab dari pernyataan masalah. Dan juga saran penulis untuk peneliti selanjutnya.

BAB II

STUDI PUSTAKA

2.1 Analisis Sentimen

Analisis sentimen adalah ilmu yang digunakan untuk menganalisis pendapat, sentimen, sikap dan emosi orang terhadap entitas seperti layanan, produk, organisasi, peristiwa dan atribut-atributnya (Bing Liu, 2012). Analisis sentimen memungkinkan untuk mengetahui sentimen masyarakat tentang entitas tertentu untuk menciptakan sebuah pengetahuan yang dapat ditindaklanjuti. Pengetahuan ini digunakan untuk memahami, menjelaskan, dan memprediksi fenomena sosial (Pozzi et al., 2017). Pada penelitian (Hemmatian & Sohrabi, 2019; Kumar & Jaiswal, 2020; Mite-Baidal et al., 2018; Pozzi et al., 2017) menggambarkan terkait konsep-konsep sentimen analisis seperti pada gambar 2.1.



Gambar 2.1 Konsep analisis sentimen

2.1.1 Sentimen Klasifikasi (*Task*)

Sentimen klasifikasi adalah salah satu *task* dari analisis sentimen yang bertugas untuk mengidentifikasi polaritas sentimen di setiap dokumen teks. Umumnya, polaritas diklasifikasikan sebagai positif dan negatif. Beberapa penelitian menganggap kelas ketiga adalah sentimen netral. *Task* ini adalah salah satu yang paling banyak dikenal dan diteliti dalam analisis sentimen (Wang et al., 2014). Beberapa penelitian terkait sentimen klasifikasi:

Penelitian yang dilakukan (Duan et al., 2020) mengusulkan model *Generative Emotion Model with Categorized Words* (GEM-CW) untuk klasifikasi sentimen *stock message*. Peneliti menggunakan pendekatan *semi-supervised*. *Generative emotion model* memanfaatkan fitur-fitur pesan, emosi, dan kata secara bersamaan. Hasil eksperimen yang diusulkan menunjukkan model ini efektif, dengan akurasi 74.6%.

2.1.2 Deep Learning (*Approaches*)

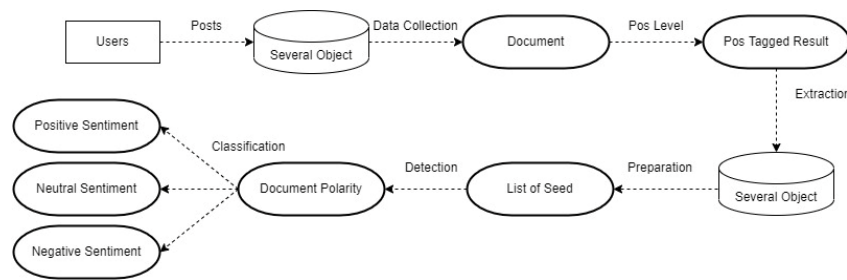
Deep learning adalah sub dari *machine learning* yang menggunakan *artificial neural network* (ANN) / jaringan syaraf tiruan. Belakangan ini *deep learning* telah banyak digunakan untuk analisis sentimen (Ligthart et al., 2021). *Deep learning* diterapkan ke banyak bidang NLP seperti teks, suara, bahasa dan memberikan hasil yang akurat. Klasifikasi teks adalah masalah fundamental di bidang NLP dan media sosial. Penggunaan *deep learning* pada analisis sentimen telah mencapai hasil yang luar biasa (Alshuwaier et al., 2022). Beberapa penelitian terkait analisis sentimen menggunakan pendekatan *deep learning*:

Penelitian yang dilakukan (Majumder et al., 2019) mengusulkan GRU berbasis *neural network* yang bisa dilatih pada *dataset* sentimen dan sarkas. Peneliti menunjukkan bahwa pendekatan berbasis *multitask-learning* memberikan hasil yang baik daripada hanya klasifikasi saja pada *dataset* sentimen dan sarkas. Untuk *dataset* sentimen memberikan presisi 79.89%, *recall* 74.86%, dan *f1-score* 77.30%, dan untuk *dataset* sarkas memberikan hasil presisi 87.41%, *recall* 87.03%, dan *f1-score* 86.97%.

2.1.3 Document-Based (Level of Analysis)

Document-based sentiment analysis (DOCSA) adalah tingkatan analisis sentimen yang sederhana dan menganggap seluruh dokumen sebagai satu komponen (entitas) analisis dan berasal dari satu *stakeholder* (pemegang opini) (Ligthart et al., 2021). Ada tiga tugas pokok dalam DOCSA, yaitu: 1) *extraction opinion words*, 2) *polarity detection*, dan 3) *polarity classification*. *Polarity classification* menentukan dokumen / entitas sebagai positif, negatif atau netral (Alshuwaier et al., 2022). *Polarity classification* diilustrasikan pada gambar 2.2. Beberapa penelitian terkait DOCSA:

Penelitian yang dilakukan (Rhanoui et al., 2019) mengusulkan model CNN-BiLSTM untuk DOCSA. Peneliti menggunakan *dataset* koran nasional dan internasional berbahasa Prancis total 2003 dengan rincian *positive* 474, *neutral* 1247, dan *negative* 282. Untuk *word embedding* menggunakan Doc2vec dan hasil akurasi 90.66%.



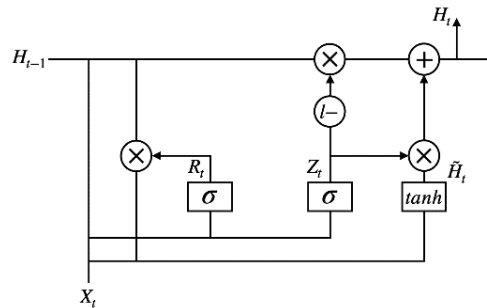
Gambar 2.2 Model polarity classification pada DOCSA

2.2 Bidirectional Gated Recurrent Unit (BiGRU)

GRU adalah salah satu mekanisme dari RNN yang mirip dengan LSTM (Bhuvaneshwari et al., 2019). GRU pertama kali diusulkan Gers, dkk. pada tahun 2014 yang merupakan model sederhana dari LSTM. Ada dua *gate* pada GRU yaitu *forget gate* dan *input gate* yang kemudian diteruskan ke *update gate*. Dari *update gate* informasi diteruskan secara selektif ke *hidden layer* untuk mengurangi masalah *gradient* saat mengingat informasi (Zeng et al., 2020). Karena kinerjanya mirip LSTM, GRU cocok digunakan pada penelitian ini dengan karakteristik yang sederhana, parameter yang sedikit, kemampuan menangani *overfitting* yang lebih baik, dan kecepatan konvergensi yang lebih cepat. Untuk lebih jelasnya pada gambar 2.4. Sedangkan Bi-GRU sendiri GRU yang bekerja dari dua arah (Lynn et al., 2019).

Pada penelitian (Han et al., 2020) mengusulkan *Pretraining and Multi-task learning model based on Double BiGRU* (PM-DBiGRU). Pertama peneliti menggunakan bobot *pretraining* dari sentimen klasifikasi ulasan obat untuk Menginisialisasi bobot model usulan peneliti. Kemudian dua Bi-GRU diterapkan untuk menghasilkan representasi semantic dari target dan ulasan obat. Hasil eksperimen menunjukkan bahwa usulan peneliti dapat meningkatkan kinerja

klasifikasi sentimen pada *aspect-based* dibandingkan dengan arsitektur lainnya dengan akurasi 78.26% dan *f1-score* 77.75%.



Gambar 2.3 GRU architecture

Berikut Formula GRU:

Reset gate:

$$r_t = \sigma(W_r[h_{t-1}, x_t] + b_r) \dots\dots\dots (2.1)$$

Candidate activation vector

$$\tilde{h}_t = \tanh(W_h + [r_t * h_{t-1}, x_t] + b_h) \dots\dots\dots (2.2)$$

Update gate

$$z_t = \sigma(W_z[h_{t-1}, x_t] + b_z) \dots\dots\dots (2.3)$$

Candidate activation vector

$$\tilde{h}_t = \tanh(W_h + [z_t * h_{t-1}, x_t] + b_h) \dots\dots\dots (2.4)$$

Hasil hidden gate

$$h_t = (1 - z_t) * h_t + z_t * \tilde{h}_t \dots\dots\dots (2.5)$$

Dimana:

x_t : *input vector*

h_t : *output vector*

\tilde{h}_t : *candidate activation vector*

z_t : *update gate vector*

r_t : *reset gate vector*

W, b : *parameter matrices and vector*

2.3 Attention Mechanism (Transformer)

Attention mechanism adalah mekanisme untuk memfokuskan setiap hasil *output* terhadap target kata. Sehingga kata yang dihasilkan memiliki hubungan yang lebih baik sesuai konteks (Bahdanau et al., 2014). Saat ini, *attention mechanism* telah terbukti efektif untuk mendapatkan hasil yang baik dengan memilih informasi yang penting. *Attention mechanism* pertama kali diusulkan di bidang *computer vision*, tujuannya untuk meniru *attention mechanism* pada manusia dengan memberikan bobot yang berbeda untuk bagian gambar (W. Li et al., 2020). (Bahdanau et al., 2014) menggunakan *attention mechanism* pada mesin penerjemah, dan ini merupakan penerapan *attention mechanism* pertama kali pada bidang NLP. Saat ini, *attention mechanism* telah menunjukkan hasil yang baik dalam bidang NLP, dengan mengimprove *deep neural network* (DNN) dan membiarkan DNN belajar sendiri dimana harus fokus pada konteks (Yadav et al., 2021). Untuk lebih jelas, berikut langkah-langkah *attention mechanism* bekerja:

1. Alignment Score

Alignment Score digunakan untuk mengukur seberapa baik kata *input sequence* dengan *output* pada posisi (t) tersebut. Caranya adalah dengan mengambil *encoded* pada *hidden state* (h_i) dan *decoder output* sebelumnya (S_{t-1}) untuk menghitung nilai $e_{t,i}$. *Alignment model* disimbolkan dengan fungsi $a(\cdot)$. Berikut persamaan dari *alignment score*.

$$e_{t,i} = a(s_{t-1}, h_i) \dots\dots\dots (2.6)$$

2. Attention Weight

Setelah ketemu nilai (S_{t-1}) pada perhitungan *alignment score*. Kemudian menghitung bobot/*weight* $(a_{t,i})$. yaitu dengan memasukkan *alignment score* ke dalam fungsi *softmax*. Fungsi aktivasi *softmax* akan menghasilkan probabilitas yang digunakan untuk mewakili bobot pada setiap *input sequence*, semakin tinggi bobot maka semakin tinggi pula pengaruh atau kontribusi yang diberikan. Berikut persamaan dari *attention weight*.

$$a_{t,i} = \text{softmax}(e_{t,i}) \dots\dots\dots (2.7)$$

3. Context Vector

Langkah terakhir adalah *context vector*. *Context vector* berfungsi untuk menghitung *output* akhir dari *decoder*. *Context vector* adalah jumlah dari *weight* $(a_{t,i})$ dan *hidden state decoder* (h_i) . Berikut persamaan dari *context vector*.

$$c_t = \sum_{i=1}^T a_{t,i} h_i \dots\dots\dots (2.8)$$

Dari persamaan ini bisa dilihat bahwa $(a_{t,i})$ mengisyaratkan seberapa banyak (h_i) berkontribusi pada konteks.

Penelitian (Zhu et al., 2019) mengusulkan model LSTM dua arah untuk meningkatkan *attention*. *Attention mechanism* digunakan secara langsung untuk mempelajari bobot setiap kata pada sentimen kalimat berdasarkan *word vector*. LSTM digunakan untuk mempelajari informasi semantic teks, dan meningkatkan klasifikasi melalui *parallel fusion*. penelitian (Ma et al., 2018) mengusulkan metode baru untuk *aspect-based*. Metode ini menggunakan *layer attention mechanism* yang terdiri dari *target-level attention* dan *sentence-level attention* untuk menambah jaringan LSTM. Eksperimen dilakukan pada dua dataset publik,

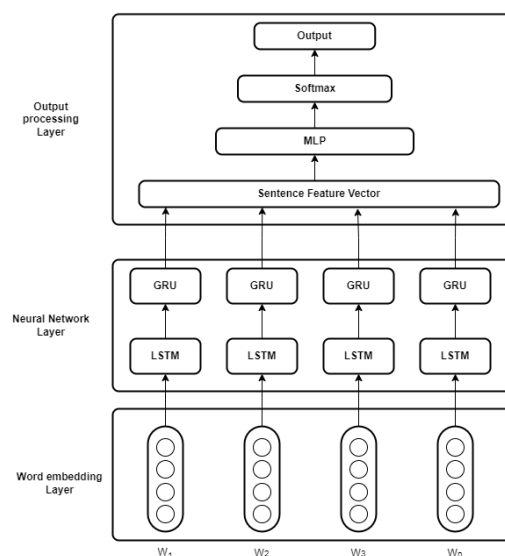
dan hasilnya menunjukkan bahwa arsitektur *attention* yang diusulkan dikombinasikan dengan *Sentic LSTM* dapat mengungguli metode terbaru dalam *task* klasifikasi emosi.

2.4 Novelty Penelitian

Sudah banyak penelitian terkait analisis sentimen yang telah diusulkan. Beberapa penelitian tentu memiliki ciri khas dan karakteristik tersendiri. Ciri khas dan karakteristik penelitian pada analisis sentimen itu sendiri ada pada *dataset*, *word embedding layer*, *neural network layer* dan *classifier*. Beberapa usulan arsitektur metode analisis sentimen sebagai berikut:

1. Arsitektur LSTM-GRU + GloVe

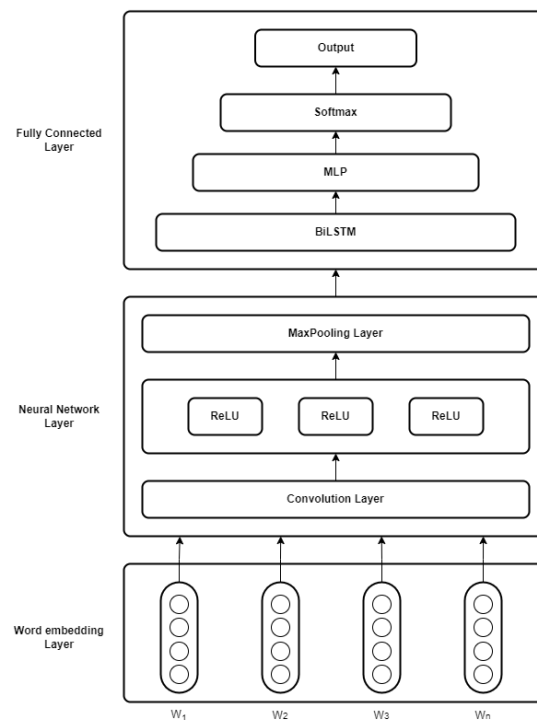
Arsitektur ini diusulkan oleh (Ni & Cao, 2020). Pada *word embedding layer* menggunakan *GloVe*, *neural network layer* menggunakan gabungan LSTM dan GRU, dan untuk *classifier* menggunakan *softmax*. Berikut arsitektur LSTM-GRU + GloVe:



Gambar 2.4 Arsitektur LSTM-GRU + GloVe

2. Arsitektur CNN-BiLSTM + Doc2vec

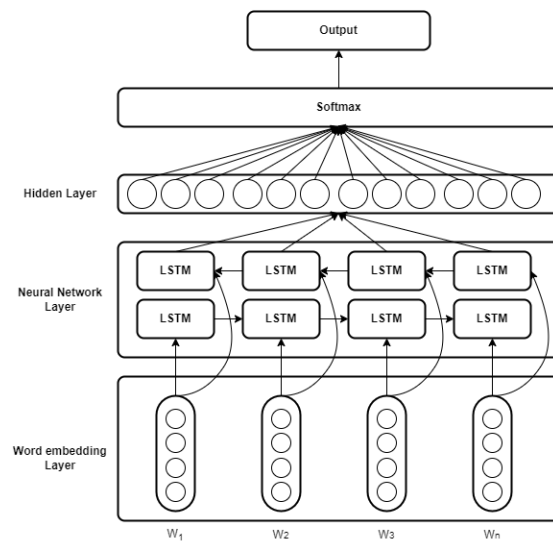
Arsitektur ini diusulkan oleh (Rhanoui et al., 2019). Pada *word embedding layer* menggunakan *Doc2Vec*, *neural network layer* menggunakan gabungan CNN dan BiLSTM, dan *classifier* menggunakan *softmax*. Berikut arsitektur CNN-BiLSTM + Doc2vec:



Gambar 2.5 Arsitektur CNN-BiLSTM + Doc2Vec

3. Arsitektur BiLSTM + Seninfo-TFIDF

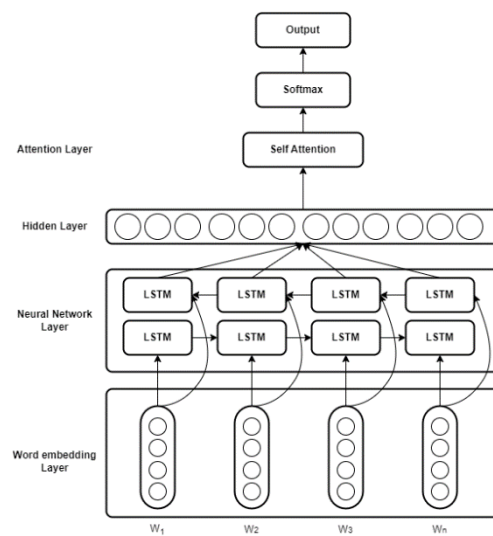
Arsitektur ini diusulkan oleh (Xu et al., 2019). Pada *word embedding layer* menggunakan gabungan *sentiment information* (Seninfo) dan TFIDF, *neural network layer* menggunakan Bi-LSTM, dan *classifier* menggunakan *softmax*. Berikut arsitektur BiLSTM+Seninfo-TFIDF:



Gambar 2.6 Arsitektur BiLSTM+seninfo-TFIDF

4. Arsitektur SAM-BiLSTM + GloVe

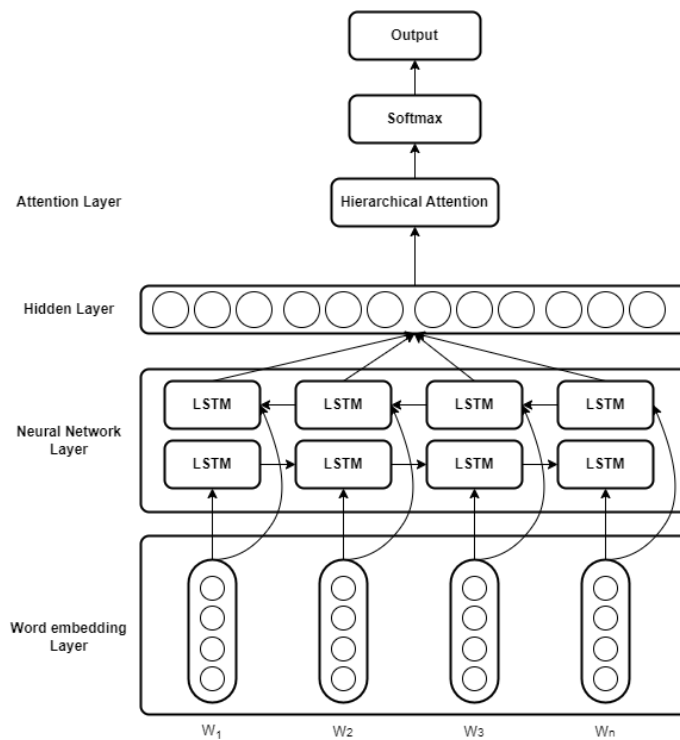
Arsitektur ini diusulkan oleh (W. Li et al., 2020). Pada *word embedding layer* menggunakan GloVe, *neural network layer* menggunakan Bi-LSTM, *attention layer* menggunakan *self-attention*, dan *classifier* menggunakan *softmax*. Berikut arsitektur SAM-BiLSTM:



Gambar 2.7 Arsitektur SAM-BiLSTM

5. Arsitektur BiLSTM-HAN

Arsitektur ini diusulkan oleh (J. Li et al., 2019). Pada *word embedding layer* menggunakan *one-hot encoding*, *neural network layer* menggunakan BiLSTM, *attention layer* menggunakan *hierarchical attention*, dan *classifier* menggunakan *softmax*. Berikut arsitektur BiLSTM-HAN:

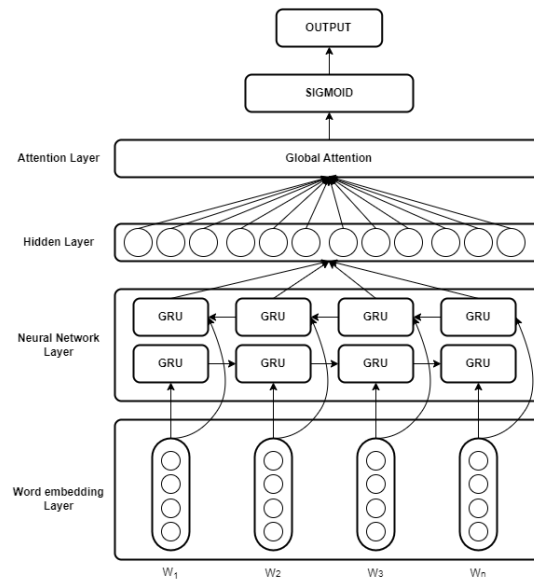


Gambar 2.8 Arsitektur BiLSTM-HAN

6. Arsitektur Glo-Att-BiGRU+Word2vec

Arsitektur ini merupakan arsitektur analisis sentimen yang diusulkan. Pada *word embedding layer* menggunakan *word2vec*, *neural network layer* menggunakan BiGRU, *attention layer* menggunakan *global*

attention, dan *classifier* menggunakan *sigmoid*. Berikut arsitektur Glo-Att-BiGRU+Word2vec:



Gambar 2.9 Arsitektur Glo-Att-BiGRU+Word2vec

Pemaparan masing-masing arsitektur analisis sentimen yang diusulkan sebelumnya (J. Li et al., 2019; W. Li et al., 2020; Ni & Cao, 2020; Rhanoui et al., 2019; Xu et al., 2019) bertujuan dalam rangka menyusun *novelty* penelitian. Adapun *novelty* pada penelitian ini terdapat pada tabel 2.1:

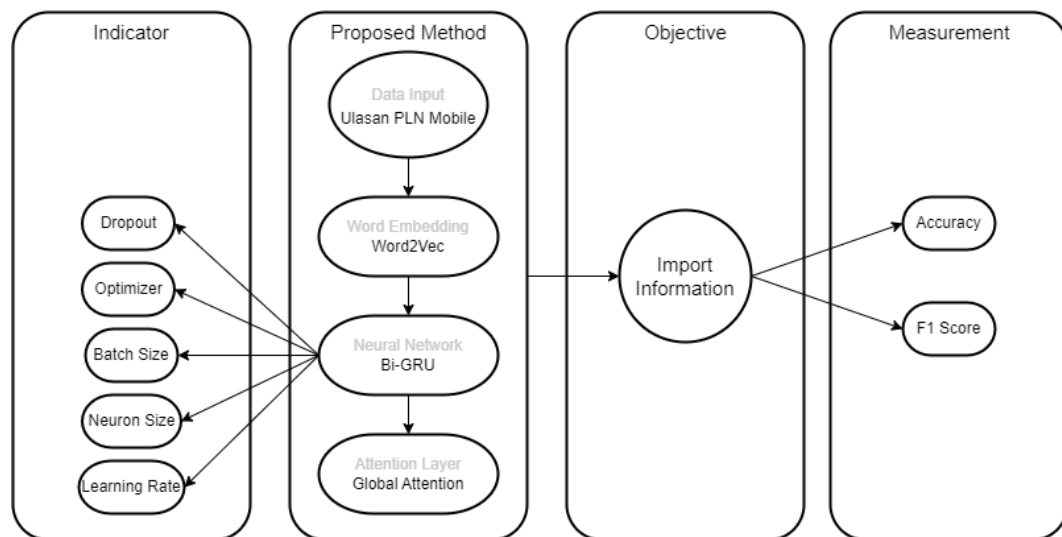
Tabel 2.1 Novelty penelitian

| Arsitektur | Word embedding | Neural network | Attention | Classifier |
|---------------------------|------------------|----------------|------------------------|------------|
| (Ni & Cao, 2020) | GloVe | LSTM+GRU | - | Softmax |
| (Rhanoui et al., 2019) | Doc2Vec | CNN+BiLSTM | - | Softmax |
| (Xu et al., 2019) | Seminfo+TFIDF | BiLSTM | - | Softmax |
| (W. Li et al., 2020) | GloVe | BiLSTM | Self-attention | Softmax |
| (J. Li et al., 2019) | One-hot encoding | BiLSTM | Hierarchical-attention | Softmax |
| Arsitektur yang diusulkan | Word2vec | BiGRU | Global-attention | Sigmoid |

Berdasarkan matriks penelitian pada tabel 2.1, maka *novelty* penelitian ini ada empat, yaitu ada pada: *word embedding layer*, *neural network layer*, *attention layer*, dan *classifier*.

2.5 Kerangka Teori

Kerangka teori adalah suatu diagram yang menjelaskan garis besar dari alur sebuah penelitian. Kerangka teori dirancang berdasarkan rumusan masalah (*research question*), dan merepresentasikan suatu himpunan dari beberapa konsep serta hubungan antara konsep-konsep tersebut. Kerangka teori pada penelitian ini terdapat pada gambar 2.4



Gambar 2.10 Kerangka teori

Pada gambar 2.4 komponen kerangka teori dibagi menjadi empat, berikut penjelasan dari masing-masing komponen:

1. *Indicator*

Indicator (Indikator) adalah variabel yang dilakukan untuk observasi. Penelitian ini menggunakan *hyperparameter setting* pada Bi-GRU untuk tujuan observasi.

2. *Proposed Method*

Proposed method (model yang diusulkan) merupakan kombinasi dari BiGRU untuk *sequence learning* dan *Global Attention* untuk memfokuskan kata yang paling besar memberikan kontribusi. Dengan *Word embedding* menggunakan *word2vec*.

3. *Objective*

Tujuan (*objective*) pada penelitian adalah untuk menangkap informasi penting dari *context information* yang dihasilkan Bi-GRU.

4. *Measurement*

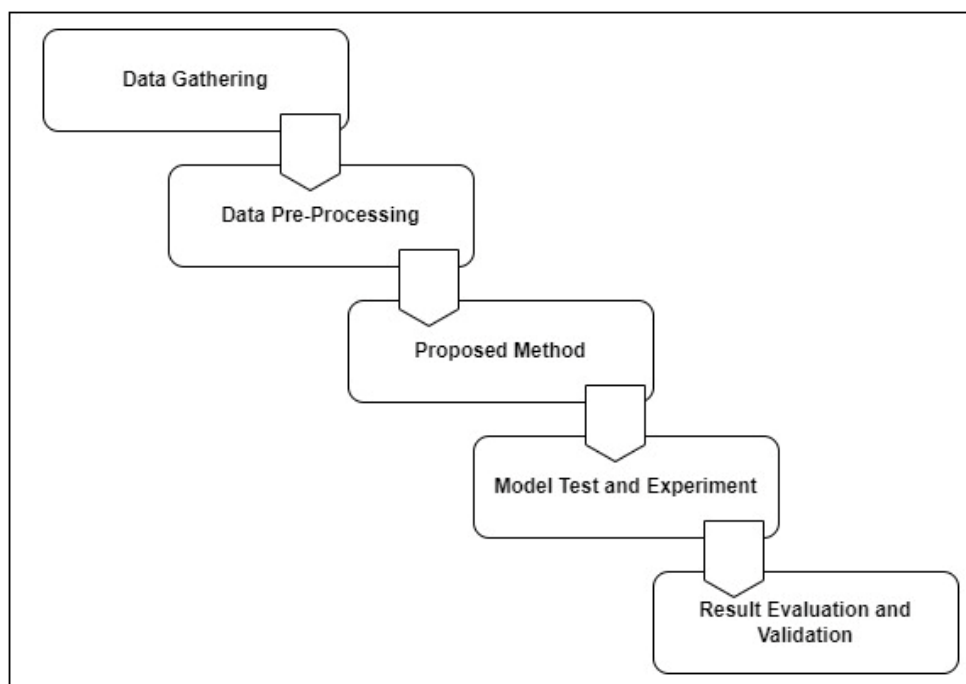
Pengukuran (*measurement*) menggunakan dua tolak ukur yaitu akurasi dan F1 score sebagai acuan dari keberhasilan menyelesaikan masalah.

BAB III

METODOLOGI PENELITIAN

3.1 Desain Penelitian

Metode penelitian yang dilakukan adalah metode eksperimen, dengan tahapan penelitian seperti pada gambar 3.1:



Gambar 3.1 Desain penelitian

1. *Data Gathering*

Pada tahapan ini, penulis mengumpulkan data-data yang terkait analisis sentimen pada ulasan PLN mobile yang berasal dari google playstore.

2. *Data Pre-Processing*

Pada tahapan ini, adalah tahap untuk melakukan pra-pemrosesan sebelum data dimodelkan. Ada enam tahap yaitu *case folding*, *tokenize*, *filtering*, *stemming*, *padding* dan, *vectorization*.

3. *Proposed Method*

Pada tahapan ini dipaparkan arsitektur model yang diusulkan, dalam penelitian ini model yang diusulkan adalah mengkombinasikan Bi-GRU dan *attention mechanism* untuk menangkap informasi penting.

4. *Model Test and Experiment*

Tahapan ini melakukan *testing* dengan *hyperparameter setting* pada model. *Hyper-parameter*-nya adalah *optimizer*, *dropout*, *batch size* dan *unit*.

5. *Result Evaluation and Validation*

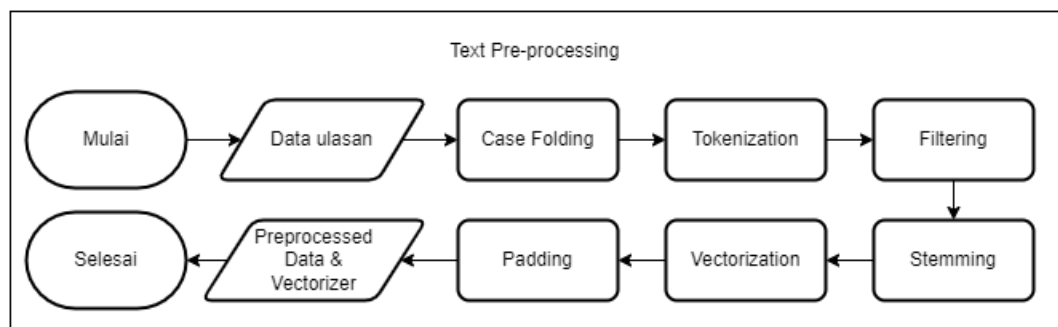
Tahapan ini memaparkan hasil evaluasi terhadap data latih dan data uji. Menggunakan tiga skema pengukuran. Satu yang utama adalah *confusion matrix* dua yang lain adalah ROC-AUC dan *precision-recall*.

3.2 Pengumpulan Data (*Data Gathering*)

Metode pengumpulan data menggunakan metode *scrapping*. Sumber data didapat dari ulasan aplikasi PLN *Mobile* di *google play store*. Atribut atau fitur yang dibutuhkan hanya dua, yakni: ulasan dan sentimen. Atribut ulasan berisi tentang ulasan pelanggan terhadap aplikasi PLN *Mobile* dan atribut sentimen berisi tentang penilaian pengguna terhadap aplikasi. Atribut sentimen hanya terdapat dua label (*binary*), yakni positif dan negatif. Untuk label netral dihapus karena menurut (Bing Liu, 2012) pada bukunya, label netral tidak terdapat sentimen atau opini, dan sifatnya objektifitas.

3.3 Pengolahan Data Awal (*Data Pre-processing*)

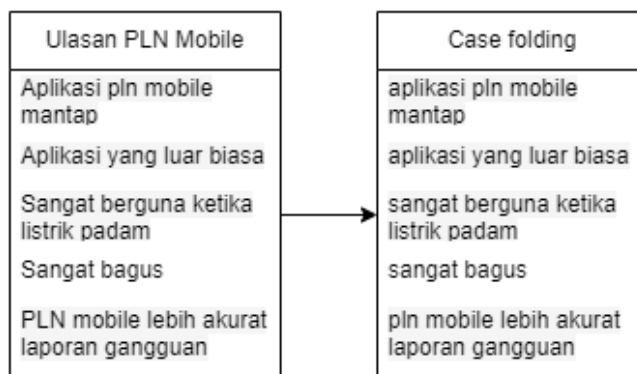
Penelitian ini menggunakan data teks, oleh karena itu proses *data pre-processing* dilakukan dengan teknik *text pre-processing*. Umumnya *text preparation* dan *pre-processing* dilakukan pada penelitian *Natural Language Processing* (NLP). Tujuan dilakukan *text pre-processing* pada penelitian ini untuk menstandarisasi data, mengurangi jumlah dimensi dalam dataset, dan mengurangi kompleksitas akan tetapi dengan informasi yang sama. Ada enam tahap *text pre-processing* pada penelitian ini seperti pada gambar 3.2.



Gambar 3.2 *Text pre-processing*

1. *Case folding*

Proses ini untuk mentransformasikan semua huruf dataset ulasan PLN Mobile menjadi huruf kecil, dan menghapus semua karakter selain huruf atau dianggap delimiter. Tujuan *case folding* untuk menyamakan istilah dan konteks. Pada gambar 3.3 adalah hasil dari data ulasan yang ditransformasikan ke *case folding*.



Gambar 3.3 Proses *case folding*

2. *Tokenization*

Setelah ulasan ditransformasikan ke *case folding*. Kalimat dijadikan token atau memecah dokumen/paragraf menjadi unit lebih kecil agar lebih mudah dipahami oleh mesin. Karena bagi mesin sulit ketika tidak dilakukan *tokenization*. Pada gambar 3.4 adalah hasil data ulasan yang dijadikan token per kata.

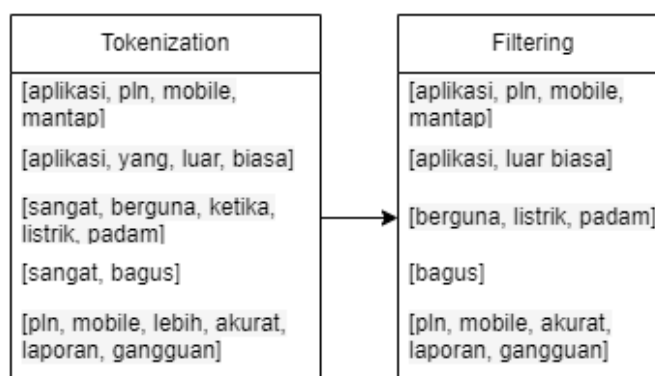


Gambar 3.4 Proses *tokenization*

3. *Filtering*

Setelah proses *tokenization*, data ulasan PLN Mobile dihilangkan kata-kata yang tidak mengandung makna atau arti, atau dalam bidang NLP istilahnya *stop word list*. Kata-kata yang dibuang seperti “di”, “ke”, “dari”, “yang”, dan lain-lain. Tujuannya adalah untuk mereduksi data sehingga

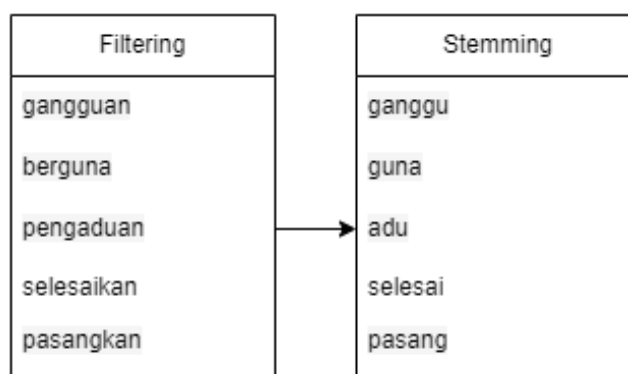
kompleksitas data berkurang. Pada gambar 3.5 adalah hasil proses dari *tokenization* ke *filtering*, bisa diperhatikan ada beberapa kata yang hilang pasca di *filtering*.



Gambar 3.5 Proses filtering

4. Stemming

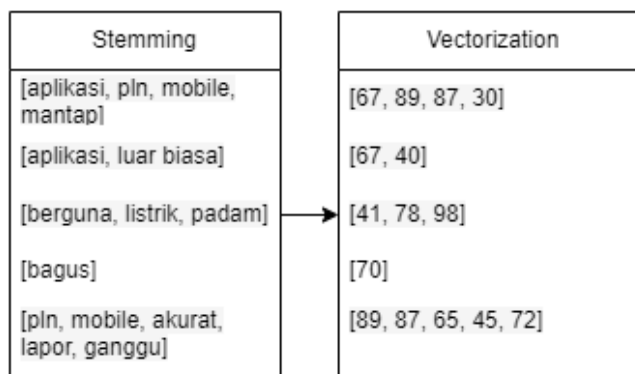
Setelah melakukan proses *filtering*, data ulasan diproses *stemming*. *Stemming* sendiri adalah proses mencari kata dasar dengan menghilangkan *prefix* dan *suffix*. Data ulasan dicari semua kata dasarnya untuk meningkatkan akurasi temu kembali informasi (*retrieval information*). Pada tabel 3.1 adalah hasil dari proses *stemming*.



Gambar 3.6 Proses stemming

5. *Vectorization*

Pada penelitian ini data ulasan PLN Mobile semuanya berbentuk teks, sedangkan komputer tidak mengenal karakter selain angka, komputer hanya menerima *input* angka. Setelah data dilakukan proses *stemming*, token-token pada dokumen/paragraf ditransformasikan menjadi sebuah bilangan (vektor), agar model *deep learning* mengenali dataset. Pada gambar 3.6 adalah proses *vectorization*, terlihat pada gambar ini yang tadinya teks berubah menjadi angka-angka.



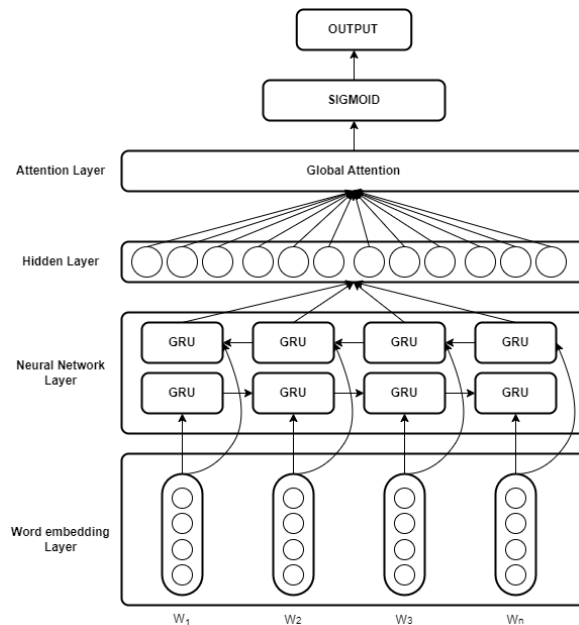
Gambar 3.7 Proses *vectorization*

6. *Padding*

Pada data ulasan PLN Mobile, bila dijadikan matrik masing-masing dokumen, maka akan terlihat bentuk (panjang) matrik tidak memiliki ukuran yang sama. Sedangkan ketika bekerja dengan *neural network* harus memiliki *input* dengan bentuk dan ukuran yang sama. Pada penelitian ini data ulasan dilakukan proses *padding* agar memiliki *input* teks dengan ukuran yang sama.

3.4 Model yang Diusulkan (*Proposed Method*)

Pada bagian ini, penulis memaparkan usulan metode secara rinci. Pada penelitian ini penulis mengusulkan *attention mechanism* yang digabungkan dengan Bi-GRU. Arsitektur *proposed method* ditunjukkan pada gambar 3.7.



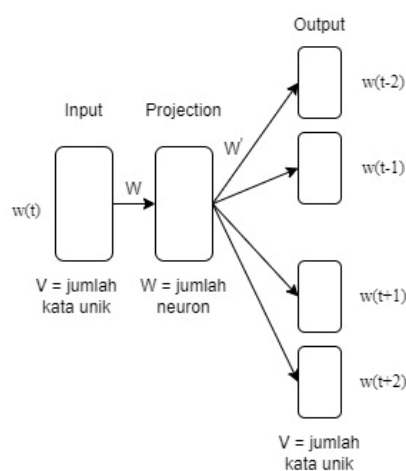
Gambar 3.8 Arsitektur model Bi-GRU dengan attention mechanism

Penulis mengusulkan penelitian analisis sentimen ulasan PLN Mobile terdiri dari: 1) *Task*: sentimen klasifikasi, 2) *Approach*: deep learning, 3) *Level-of Analysis*: Document Based. Dan terbagi tiga layer. Pertama, dilakukan *word embedding* sebelum *training* untuk merepresentasikan *word*. Kedua, proses *sequence learning* menggunakan Bi-GRU. Ketiga, *attention layer* untuk menangkap informasi penting dari Bi-GRU.

1. *Word Embedding*

Tradisional *text representation*, seperti *Bag of Words*, terdapat dua masalah, yaitu: kehilangan *learning representation* dan ukuran

dimensionality yang besar. Untuk mengatasi masalah ini, penelitian ini menggunakan *word embedding*, untuk merepresentasikan setiap kata yang mirip secara semantik. *Word embedding* yang digunakan adalah *word2vec* yang diusulkan oleh Mikolve (Mikolov et al., 2013). Arsitektur yang digunakan dari *word2vec* adalah skip-gram. Model dilatih dengan memaksimalkan *average log probability* di semua kata.



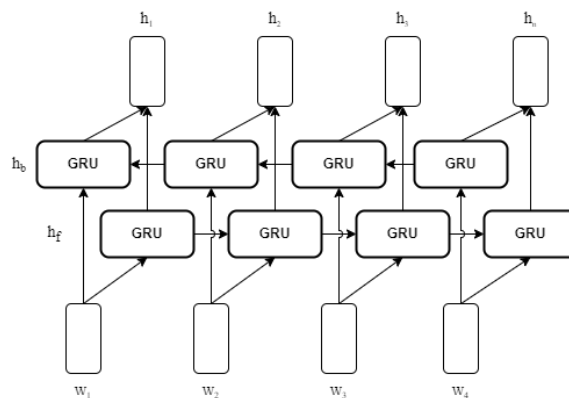
Gambar 3.9 Arsitektur skip-gram

Pada gambar 3.8 arsitektur skip-gram memberikan bobot pada W dan W' secara acak. W dan W' merupakan matrik dengan ukuran $W = V * N$ dan $W' = N * V$. Pada tahap *feedforward*, *input* di *dot product* dengan W dan menghasilkan nilai pada *hidden layer (projection)*. Kemudian *hidden layer* di *dot product* dengan W' dan menghasilkan nilai *output*, setelah itu akan dihitung nilai errornya menggunakan *cross entropy*. Pada tahap *backpropagation*, menggunakan teknik *gradient descent* yaitu dengan melakukan *upgrade* W dan W' . Proses iterasi akan terus berjalan sampai

nilai error minimum. Penelitian ini menggunakan ukuran dimensi setiap *word vector* sebesar 300.

2. Bi-GRU

Secara konsep, analisis sentimen adalah proses *sequence learning* dengan jenis *task many-to-one*. *Word representation* yang dihasilkan oleh *word2vec* tidak terdapat *sequence information*. Karena itu, Bi-GRU digunakan untuk *sequence modelling* yang dapat mengekstrak *context information* dari *word representation* yang dihasilkan *word2vec*.



Gambar 3.10 Arsitektur bidirectional GRU

Pada gambar 3.9 Bi-GRU mempelajari *context information* dari dua arah yaitu: *forward* dan *backward*, setelah itu menggabungkan keduanya. Bi-GRU terdiri dari *forward GRU* (direpresentasikan sebagai \overrightarrow{GRU}) untuk mempelajari *feature sequence* dari Lc_1 ke Lc_{100} dan *backward GRU* (direpresentasikan sebagai \overleftarrow{GRU}) untuk mempelajari *feature sequence* dari Lc_{100} ke Lc_1 . Berikut formula Bi-GRU secara detail:

$$\vec{h}_f = \overrightarrow{GRU}(Lc_n), n \in [1, 100] \dots\dots\dots (3.1)$$

$$\overleftarrow{h}_b = \overleftarrow{GRU}(Lc_n), n \in [100, 1] \dots\dots\dots (3.2)$$

3. *Global Attention*

Pada dokumen sentimen, setiap kata memiliki kontribusi yang berbeda terhadap konteks sentimen, dengan memberikan bobot yang berbeda pada kata, dapat memberikan fokus (*attend*) yang berbeda pada *context information* yang dihasilkan Bi-GRU. *Global attention* digunakan untuk memberikan bobot yang berbeda pada kata untuk meningkatkan pemahaman terhadap sentimen. *Global attention* dapat memfokuskan fitur dari *keyword* dan mengurangi fitur *non-keywords* menggunakan *fully-connected layer* dan *softmax function*. Berikut formula dari *global attention*:

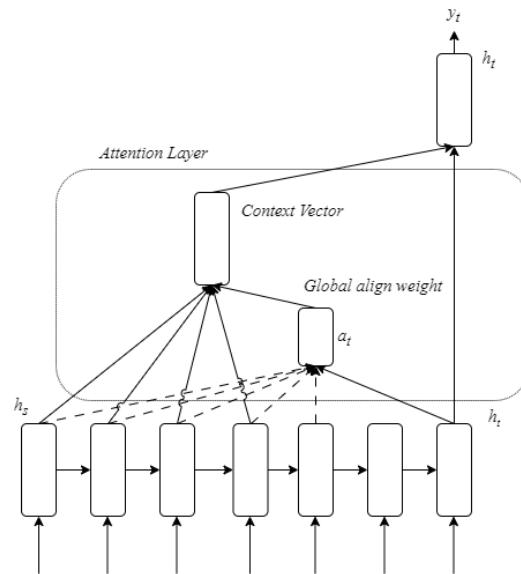
$$M = \tanh \left(\begin{bmatrix} W_h H \\ W_v v_a \otimes e_n \end{bmatrix} \right) \dots\dots (3.3)$$

$$a = \text{softmax}(W^T M) \dots\dots (3.4)$$

$$r = H a^T \dots\dots (3.5)$$

Dimana $H \in \mathbb{R}^{d \times N}$ adalah matriks yang terdiri dari *hidden vector* $[h_1, \dots, h_N]$ yang dihasilkan Bi-GRU, d adalah ukuran dari *hidden layer*. v_a adalah representasi *embedding* dan W adalah yang harus dipelajari selama *training*.

Pada formula 3.3 digunakan untuk mencari *alignment score* menggunakan *hyperbolic tangent* dan menghasilkan fitur yang lebih kompleks. Formula 3.4 digunakan untuk mencari *attention weight* dengan menambahkan *softmax* dan menghasilkan distribusi sentimen. Formula 3.5 digunakan untuk mencari *context vector*.



Gambar 3.11 Arsitektur global attention

Pada gambar 3.10 setiap *time step* t , *global attention* menghitung *alignment weight* a_t berdasarkan *target state* saat ini h_t . *context vector* c_t dihitung sebagai bobot rata-rata berdasarkan *alignment weight* dan semua *state*.

3.5 Eksperimen dan Pengujian Model (*Model Test and Experiment*)

Eksperimen dilakukan untuk menguji model yang diusulkan. Penelitian ini menggunakan empat skenario pengujian yang berbeda untuk mencari akurasi model yang terbaik. Skenarionya adalah dengan melakukan *hyperparameter setting*. Berikut skenario-skenario pengujian:

1. Melakukan *tuning* pada *dropout*

Skenario pertama dengan melakukan *tuning* pada *dropout*. Yaitu dengan mengurangi *unit neural network* pada ukuran tertentu. ukuran tersebut seperti yang tertera pada tabel 3.2:

Tabel 3.1 Pengujian dengan dropout

| Dropout | Accuracy | F1 |
|---------|----------|----|
| 0.2 | ? | ? |
| 0.3 | ? | ? |
| 0.5 | ? | ? |

2. Melakukan *tuning* pada *optimizer*

Skenario kedua dengan melakukan *tuning* pada *optimizer*. Yaitu dengan mencari *optimizer* mana yang dapat meminimalkan *error function* (*loss function*) hingga *cost* yang terkecil. *Optimizer* yang akan digunakan dalam pengujian seperti yang tertera pada tabel 3.3:

Tabel 3.2 Pengujian dengan optimizer

| Optimizer | Accuracy | F1 |
|------------------|----------|----|
| Adam | ? | ? |
| Gradient Descent | ? | ? |
| Adadelta | ? | ? |
| RMSProp | ? | ? |

3. Melakukan *tuning* pada *batch size*

Skenario ketiga dengan melakukan *tuning* pada *batch size*. Yaitu dengan menentukan jumlah sampel data *training* dalam satu kali iterasi. Jumlah sampelnya seperti yang tertera pada tabel 3.4:

Tabel 3.3 Pengujian dengan batch size

| Batch Size | Accuracy | F1 |
|------------|----------|----|
| 24 | ? | ? |
| 32 | ? | ? |
| 64 | ? | ? |
| 128 | ? | ? |

4. Melakukan *tuning* pada *units*

Skenario keempat dengan melakukan *tuning* pada *units*. Yaitu dengan menentukan jumlah *unit* atau *neuron* pada *hidden layer*. Jumlah *neuron* atau *unit*-nya sendiri tertera pada tabel 3.5:

Tabel 3.4 Pengujian dengan *units*

| Units | Accuracy | F1 |
|-------|----------|----|
| 32 | ? | ? |
| 64 | ? | ? |
| 128 | ? | ? |
| 256 | ? | ? |

5. Melakukan *tuning* pada *learning rate*

Skenario kelima dengan melakukan *tuning* pada *learning rate*. Yaitu dengan menentukan *learning rate* pada *optimizer*. *Learning rate* pada penelitian ini sendiri tertera pada tabel 3.6:

Tabel 3.5 Pengujian dengan *learning rate*

| Learning Rate | Accuracy | F1 |
|---------------|----------|----|
| 0.01 | ? | ? |
| 0.005 | ? | ? |
| 0.025 | ? | ? |
| 0.001 | ? | ? |

3.6 Evaluasi dan Validasi Hasil (*Result Evaluation and Validation*)

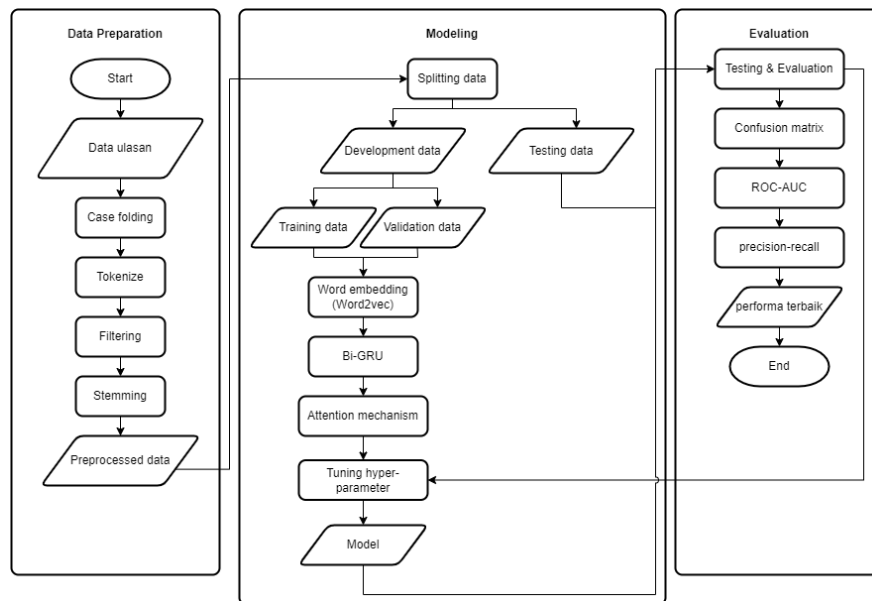
Dari hasil eksperimen dan pengujian model, tahap selanjutnya dilakukan evaluasi model untuk membahas hasil pengujian dan observasi model. Evaluasi model disini menggunakan tiga alat pengukuran, satu yang utama adalah *confusion matrix* dua yang lain adalah ROC-AUC dan *precision-recall*. *Confusion matrix* digunakan untuk mengukur seberapa baik arsitektur “glo-att-

bigru+word2vec” dalam mengatasi masalah menggunakan akurasi dan F1 score. Sedangkan ROC digunakan untuk mengukur akurasi dengan mencari *threshold* yang terbaik atau tanpa memperhatikan distribusi label dan AUC digunakan untuk menghitung ruang dibawah ROC *curve* dengan rentang nilai 0-1. Untuk *precision-recall* fungsinya sama dengan ROC-AUC akan tetapi digunakan untuk mengukur *data imbalance*.

3.7 Desain Sistem

Pada penelitian ini desain sistem model terbagi menjadi tiga bagian yaitu:

Data Preparation, Modelling, dan Evaluation seperti pada gambar 3.11.



Gambar 3.12 Desain sistem model

Berikut untuk detailnya:

1. *Data Preparation*

Data preparation berfungsi untuk memproses data sebelum data dimodelkan. dan ada empat tahap, yaitu: *case folding*, *tokenizing*, *filtering*, dan *stemming*.

2. *Modelling*

Setelah data bersih dari *noise*. Kemudian data dimodelkan. Disini data dibagi menjadi dua, yaitu: data latih dan data uji. Data latih digunakan untuk proses *training model*, dan data uji digunakan untuk *testing model*. Kemudian data dimodelkan menggunakan algoritma Bi-GRU dan *attention mechanism*.

3. *Evaluation*

Bagian ketiga adalah evaluasi. Bagian ini untuk mengukur seberapa besar akurasi Bi-GRU menggunakan *attention mechanism*. Dan seperti gambar 3.11 untuk proses *testing & evaluation* iterasi ke proses *hyperparameter setting* ini menandakan proses pengujian pada model.

BAB IV

HASIL DAN PEMBAHASAN

4.1 Experiment Environment

Bab ini merupakan tahap eksperimen dan pengujian, untuk mengetahui performa arsitektur yang diusulkan. Eksperimen ini terdapat beberapa pengujian yang dilakukan yaitu pengujian dengan *hyperparameter setting*, pengujian model menggunakan *global attention* dan tanpa *global attention*, dan pengujian dengan membandingkan metode analisis sentimen yang lain.

Platform eksperimen yang digunakan adalah Google Colaboratory Pro, dengan spesifikasi GPU NVIDIA A100, memory 90 GB, python 3.8.10 dan disk 166 GB. Library yang digunakan untuk eksperimen adalah *tensorflow*, *pandas*, *scikit-learn*, dan beberapa library pendukung yang lain.

4.2 Dataset

Dataset diambil / *scrapping* dari ulasan PLN mobile pada google playstore. *Dataset* dikumpulkan dari periode Agustus 2016 sampai dengan Februari 2023. *Dataset* yang digunakan untuk penelitian berjumlah 236905 berbahasa Indonesia. Sebelum dataset digunakan untuk uji coba metode yang diusulkan, *dataset* dilakukan *pre-processing* terlebih dahulu. *Pre-processing* yang dilakukan antara lain merubah *rating* menjadi label *positive* dan *negative*, *handling data imbalance*, dan *data cleaning*. Contoh *dataset* ditunjukkan pada tabel 4.1.

Tabel 4.1 Contoh dataset

| User | Review | Sentiment | Date |
|---------------------|---------------------------------------|-----------|------------|
| Valdo Rivaldo | bantu lapor ganggu listrik | positive | 2023-02-22 |
| Novi Hendri winanto | layan mantap | positive | 2023-02-22 |
| Rifqi Fauji | gak bisa cek sisa saldo guna aplikasi | negative | 2023-02-22 |
| Novy Tael | aplikasi mudah bantu | positive | 2023-02-22 |
| Rif YZ | kanapa nggak verifikasi | negative | 2023-02-21 |
| Udin Lenong | layan lambat | negative | 2023-02-21 |

4.3 Indikator Evaluasi

Eksperimen ini menggunakan tiga alat pengukuran yaitu *confusion matrix*, ROC-AUC dan *precision-recall*. *Confusion matrix* digunakan untuk mengukur F1-Score dan akurasi. F1-Score dipilih untuk mengukur model pada pengujian *dataset* yang *imbalance*, sedangkan akurasi dipilih untuk mengukur model pada pengujian *dataset* yang *balance* (setelah dilakukan SMOTE: *oversampling*).

Tabel 4.2 Confusion matrix

| | | Actual Value | |
|-----------------|---------|--------------|---------|
| | | Positif | Negatif |
| Predicted Value | Positif | TP | FP |
| | Negatif | FN | TN |

$$accuracy = \frac{TP+TN}{TP+TN+FP+FN} \dots\dots\dots (4.1)$$

$$precision = \frac{TP}{TP+FP} \dots\dots\dots (4.2)$$

$$recall = \frac{TP}{TP+FN} \dots\dots\dots (4.3)$$

$$F1 = 2 * \frac{(recall*precision)}{(recall+precision)} \dots\dots\dots (4.4)$$

ROC-AUC dan *precision-recall* digunakan untuk menentukan nilai *threshold* yang optimal dari arsitektur yang diusulkan. ROC-AUC digunakan untuk pengujian *dataset balance* sedangkan *precision-recall* digunakan untuk pengujian *dataset imbalance*.

$$TPR = \frac{TP}{TP+FN} \dots\dots\dots (4.5)$$

$$FPR = \frac{FP}{FP+TN} \dots\dots\dots (4.6)$$

4.4 Hyperparameter Setting

Untuk melatih model yang diusulkan agar lebih baik, melakukan *hyperparameter setting* (mencari *hyperparameter* terbaik) adalah hal yang penting. *Hyperparameter setting* pada model yang diusulkan mencakup *dropout*, *optimizer*, *batch size*, *units* dan *learning rate*. *Hyperparameter* terbaik akan diterapkan pada pengujian model selanjutnya, yaitu pengujian dengan menggunakan dan tanpa menggunakan *global attention* dan pengujian perbandingan dengan arsitektur yang lain.

4.4.1 Dropout

Dropout adalah teknik pada *neural network* untuk *regularization* dan mengurangi *overfitting* dengan cara menghilangkan sementara beberapa *neuron* pada *hidden layer*. Dengan menentukan *range dropout* yang tepat dapat

memberikan hasil model yang terbaik. Pada pengujian ini *range dropout* yang akan diuji adalah 0.2, 0.3, dan 0.5. Detail skenario pengujian sebagai berikut:

Skenario pengujian pada data *imbalance*:

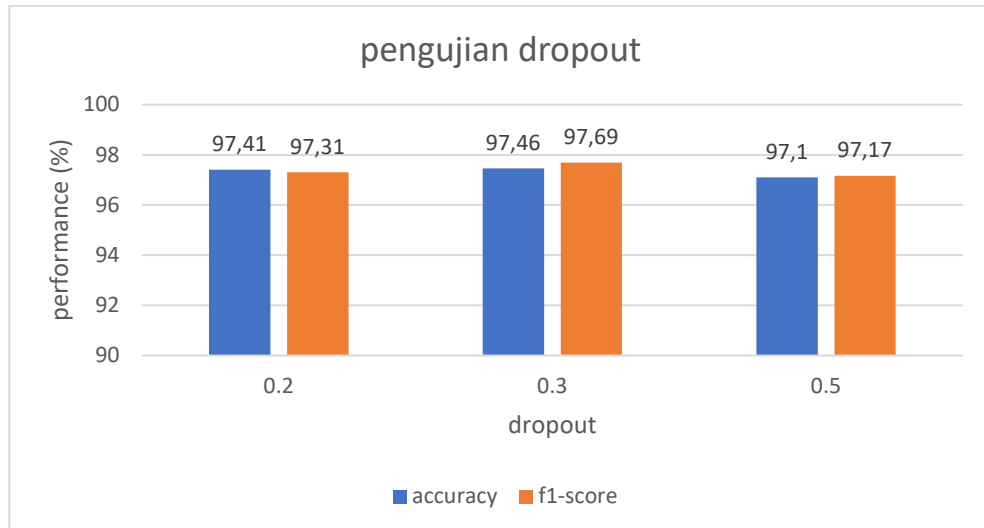
- Jumlah data yang digunakan sebanyak 236905 dengan label positif sebanyak 213214 dan label negatif sebanyak 21924.
- Teknik pembagian data untuk evaluasi model adalah split ratio dengan perbandingan 90:10, oleh karena itu jumlah data training sebanyak 213214 dan data testing sebanyak 23691.
- *Loss function* yang digunakan adalah *binary crossentropy*.
- *Optimizer* yang digunakan adalah adam.
- *Batch size* yang digunakan sebanyak 32.
- *Units* yang digunakan sebanyak 32 *neuron*.
- *Learning rate* yang digunakan sebesar 0.001.

Skenario pengujian pada data *balance*:

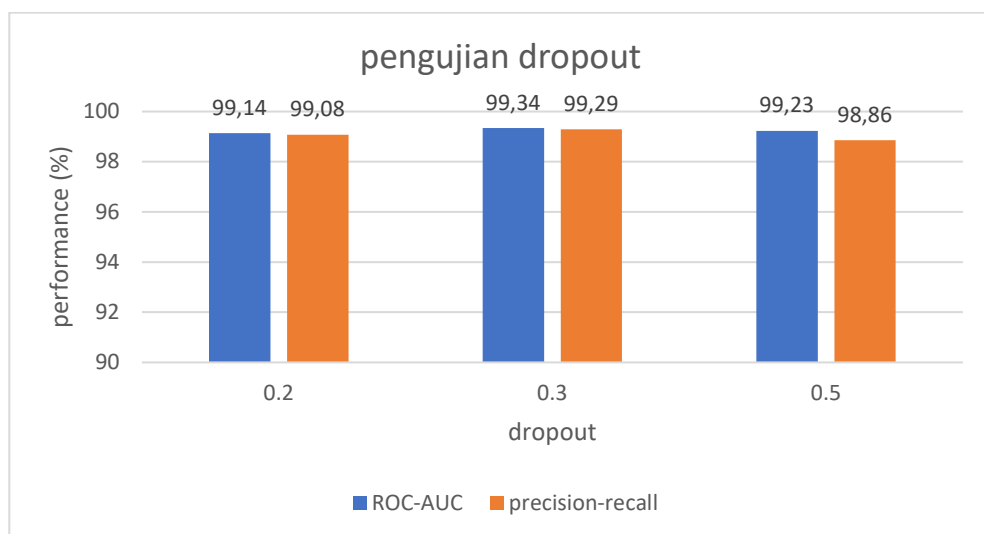
- Jumlah data yang digunakan sebanyak 429962 dengan label positif dan negatif sebanyak 214981.
- Teknik pembagian data untuk evaluasi model adalah split ratio dengan perbandingan 90:10, oleh karena itu jumlah data training sebanyak 386965 dan data testing sebanyak 42997.
- *Loss function* yang digunakan adalah *binary crossentropy*.
- *Optimizer* yang digunakan adalah adam.
- *Batch size* yang digunakan sebanyak 32.
- *Units* yang digunakan sebanyak 32 *neuron*.

- *Learning rate* yang digunakan sebesar 0.001.

Hasil pengujian seperti pada gambar 4.1 dan 4.2.



Gambar 4.1 Pengujian akurasi dan f1-score pada dropout



Gambar 4.2 Pengujian ROC-AUC dan precision-recall pada dropout

Pada gambar 4.1 dan gambar 4.2 menunjukkan *range dropout* terbaik ada pada *range* 0.3. Hal ini dikarenakan jika *range dropout* terlalu kecil dapat mengurangi model dalam menggeneralisasi data sehingga kemampuan model dalam memprediksi data yang baru menjadi berkurang. Sedangkan jika *dropout*

terlalu besar maka model dapat menjadi kurang efektif dalam mempelajari pola-pola yang kompleks pada data sehingga model kehilangan terlalu banyak informasi selama pelatihan, dan menyebabkan kinerja model menurun.

4.4.2 *Optimizer*

Optimizer adalah algoritma yang digunakan untuk mengoptimalkan atau meminimalkan nilai fungsi kerugian (*loss function*) pada saat melakukan *training* model *neural network*. *Optimizer* ini bekerja dengan mengubah bobot (*weights*) dan bias (*biases*) pada setiap iterasi *training*, sehingga memperbaiki prediksi model dan membuatnya lebih akurat. Pemilihan *optimizer* yang tepat harus dilakukan untuk meningkatkan akurasi model, menghindari *overfitting*, mengoptimalkan *learning rate*. Pada pengujian ini *optimizer* yang akan diuji adalah *gradient descent* (SGD), *adam*, *adadelta*, dan *RMSprop*. Detail skenario pengujian sebagai berikut:

Skenario pengujian pada data *imbalance*:

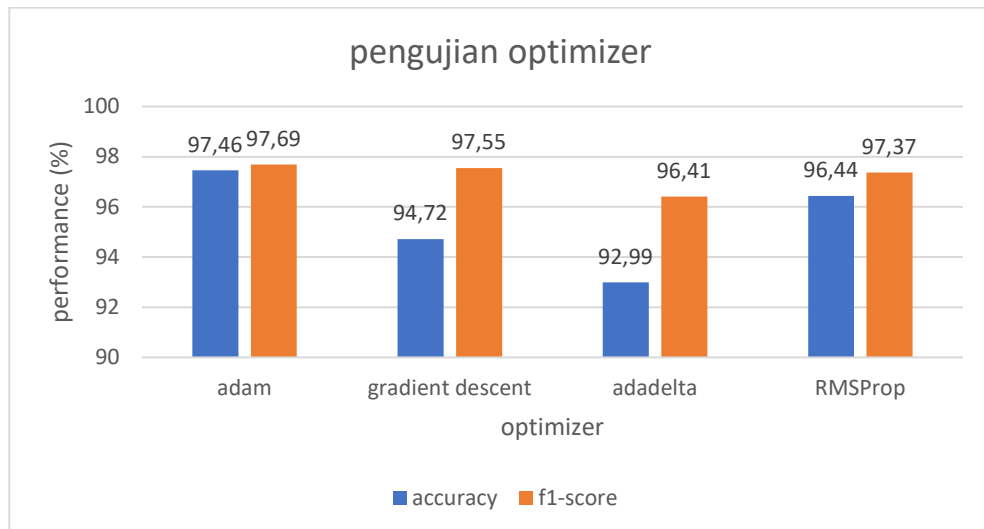
- Jumlah data yang digunakan sebanyak 236905 dengan label positif sebanyak 213214 dan label negatif sebanyak 21924.
- Teknik pembagian data untuk evaluasi model adalah split ratio dengan perbandingan 90:10, oleh karena itu jumlah data training sebanyak 213214 dan data testing sebanyak 23691.
- *Loss function* yang digunakan adalah *binary crossentropy*.
- *Dropout* menggunakan *range* 0.3.
- *Batch size* yang digunakan sebanyak 32.
- *Units* yang digunakan sebanyak 32 *neuron*.

- *Learning rate* yang digunakan sebesar 0.001.

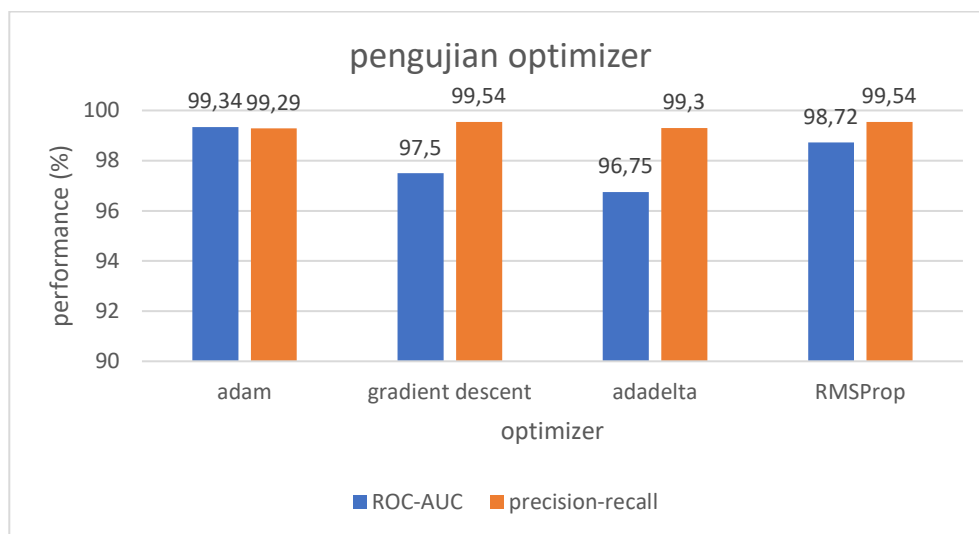
Skenario pengujian pada data *balance*:

- Jumlah data yang digunakan sebanyak 429962 dengan label positif dan negatif sebanyak 214981.
- Teknik pembagian data untuk evaluasi model adalah split ratio dengan perbandingan 90:10, oleh karena itu jumlah data training sebanyak 386965 dan data testing sebanyak 42997.
- *Loss function* yang digunakan adalah *binary crossentropy*.
- *Dropout* menggunakan *range* 0.3.
- *Batch size* yang digunakan sebanyak 32.
- *Units* yang digunakan sebanyak 32 *neuron*.
- *Learning rate* yang digunakan sebesar 0.001.

Hasil pengujian seperti pada gambar 4.3 dan 4.4.



Gambar 4.3 Pengujian akurasi dan F1-Score pada optimizer



Gambar 4.4 Pengujian ROC-AUC dan precision-recall pada dropout

Pada gambar 4.3 dan gambar 4.4 menunjukkan *optimizer* terbaik adalah *adam*. Hal ini dikarenakan *adam* (*adaptive moment estimation*) menggabungkan konsep dari SGD dan AdaGrad yaitu dengan memperbarui bobot dan bias dengan mengambil gradien rata-rata dan rata-rata kuadrat dari gradien, dan kemudian menyesuaikan *learning rate* untuk setiap parameter. Sedangkan AdaDelta dan RMSProp umumnya dipakai pada data non-stasioner seperti harga saham, suhu, dan pengunjung situs web.

4.4.3 Batch Size

Batch size adalah jumlah sampel data yang diberikan kepada model pada setiap iterasi training. Pada *neural network*, menentukan *batch size* yang tepat penting untuk dilakukan karena mempengaruhi waktu *training* dan akurasi model. Pada pengujian ini, jumlah *batch size* yang akan diuji sebanyak 24, 32, 64 dan 128. Detail skenario pengujian sebagai berikut:

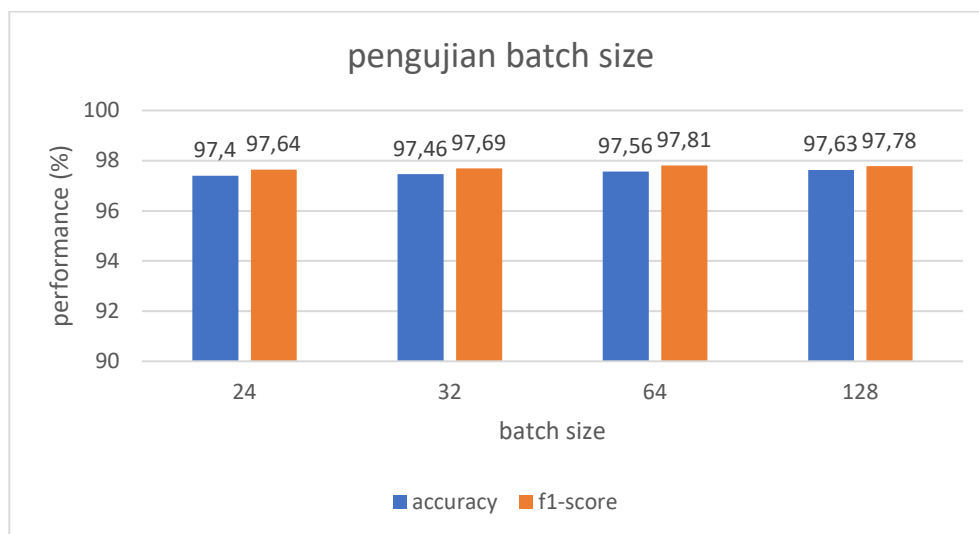
Skenario pengujian pada data *imbalance*:

- Jumlah data yang digunakan sebanyak 236905 dengan label positif sebanyak 213214 dan label negatif sebanyak 21924.
- Teknik pembagian data untuk evaluasi model adalah split ratio dengan perbandingan 90:10, oleh karena itu jumlah data training sebanyak 213214 dan data testing sebanyak 23691.
- *Loss function* yang digunakan adalah *binary crossentropy*.
- *Dropout* menggunakan *range* 0.3.
- *Optimizer* yang digunakan adalah adam.
- *Units* yang digunakan sebanyak 32 *neuron*.
- *Learning rate* yang digunakan sebesar 0.001.

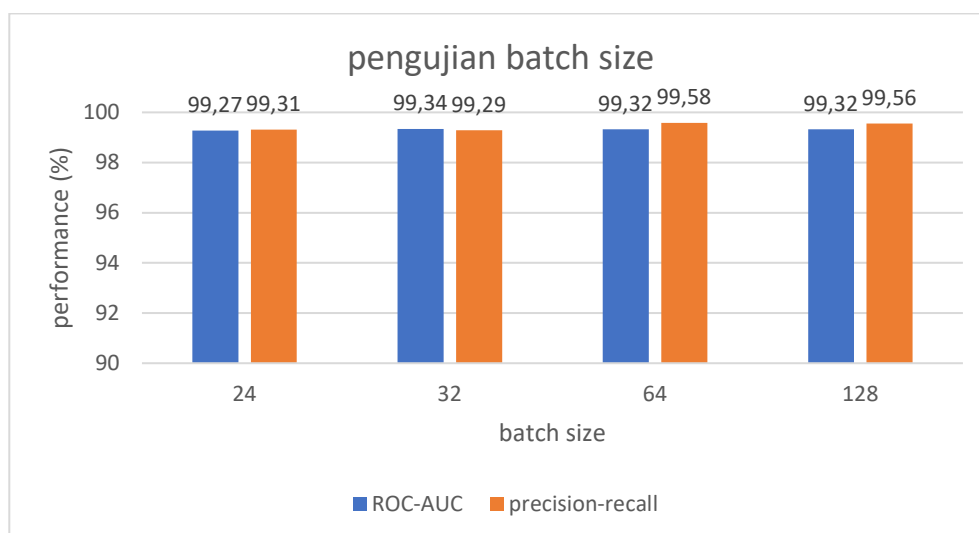
Skenario pengujian pada data *balance*:

- Jumlah data yang digunakan sebanyak 429962 dengan label positif dan negatif sebanyak 214981.
- Teknik pembagian data untuk evaluasi model adalah split ratio dengan perbandingan 90:10, oleh karena itu jumlah data training sebanyak 386965 dan data testing sebanyak 42997.
- *Loss function* yang digunakan adalah *binary crossentropy*.
- *Dropout* menggunakan *range* 0.3.
- *Optimizer* yang digunakan adalah adam.
- *Units* yang digunakan sebanyak 32 *neuron*.
- *Learning rate* yang digunakan sebesar 0.001.

Hasil pengujian seperti pada gambar 4.5 dan gambar 4.6.



Gambar 4.5 Pengujian akurasi dan f1score pada batch size



Gambar 4.6 Pengujian ROC-AUC dan precision-recall pada batch size

Pada gambar 4.5 pengujian akurasi terbaik ada pada *batch size* 128 sedangkan f1-score terbaik ada pada *batch size* 64. Hal ini dikarenakan *batch size* yang besar dapat membantu model lebih stabil dalam mempelajari pola-pola umum dalam data pelatihan, sehingga model lebih dapat digeneralisasi pada data yang tidak pernah dilihat sebelumnya.

4.4.4 Units

Units pada *neural network* merujuk pada jumlah *neuron* atau *node* yang terdapat pada *hidden layer* dari jaringan saraf tiruan. Dengan menentukan *units* yang tepat membuat model menggeneralisasi data lebih baik, akurasi yang tepat dan mengurangi kompleksitas pada *neural network*. Pada pengujian ini, jumlah *units* yang akan diuji sebanyak 32, 64, 128, 256.

Skenario pengujian pada data *imbalance*:

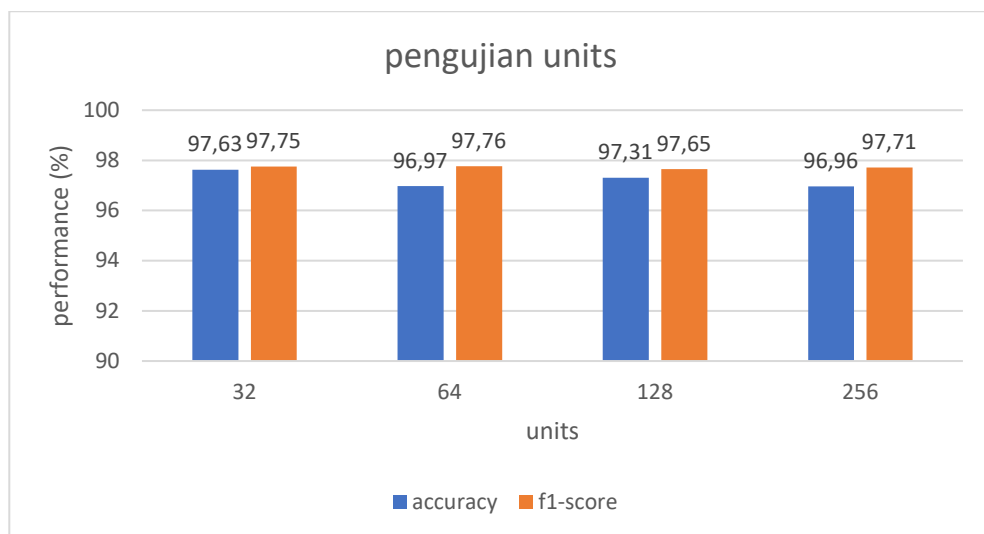
- Jumlah data yang digunakan sebanyak 236905 dengan label positif sebanyak 213214 dan label negatif sebanyak 21924.
- Teknik pembagian data untuk evaluasi model adalah split ratio dengan perbandingan 90:10, oleh karena itu jumlah data training sebanyak 213214 dan data testing sebanyak 23691.
- *Loss function* yang digunakan adalah *binary crossentropy*.
- *Dropout* menggunakan *range* 0.3.
- *Optimizer* yang digunakan adalah adam.
- *Batch size* yang digunakan sebanyak 64.
- *Learning rate* yang digunakan sebesar 0.001

Skenario pengujian pada data *balance*:

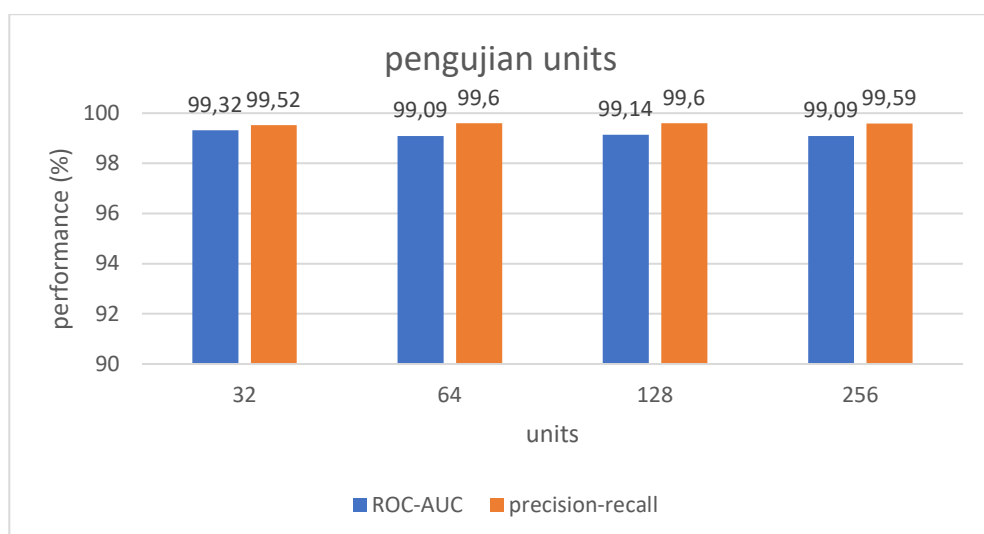
- Jumlah data yang digunakan sebanyak 429962 dengan label positif dan negatif sebanyak 214981.
- Teknik pembagian data untuk evaluasi model adalah split ratio dengan perbandingan 90:10, oleh karena itu jumlah data training sebanyak 386965 dan data testing sebanyak 42997.

- *Loss function* yang digunakan adalah *binary crossentropy*.
- *Dropout* menggunakan *range* 0.3.
- *Optimizer* yang digunakan adalah *adam*.
- *Batch size* yang digunakan sebanyak 128.
- *Learning rate* yang digunakan sebesar 0.001

Hasil pengujian seperti pada gambar 4.7 dan gambar 4.8.



Gambar 4.7 Pengujian akurasi dan f1-score pada units



Gambar 4.8 Pengujian ROC-AUC dan precision-recall pada units

Pada gambar 4.8 pengujian akurasi terbaik ada pada jumlah *units* 32 sedangkan f1-score terbaik ada pada jumlah *units* 64. Hal ini dikarenakan *units* 32 dan 64 membuat model tidak terlalu kompleks sehingga mudah dipahami sedangkan *units* 128 dan 256 membuat model sulit memahami untuk membuat prediksi karena terlalu kompleks.

4.4.5 *Learning Rate*

Learning rate adalah parameter yang digunakan dalam *neural network* untuk menentukan seberapa besar langkah yang diambil oleh algoritma saat memperbarui bobot atau parameter model pada setiap iterasi. Secara sederhana, *learning rate* dapat diartikan sebagai seberapa cepat atau lambat model belajar dari data yang diberikan. Menentukan *learning rate* yang tepat perlu dilakukan karena dapat meningkatkan konvergensi lebih cepat sehingga performa lebih baik, dan meningkatkan akurasi. Pada pengujian ini, beberapa *learning rate* yang akan diuji adalah 0.1, 0.005, 0.025, 0.001.

Skenario pengujian pada data *imbalance*:

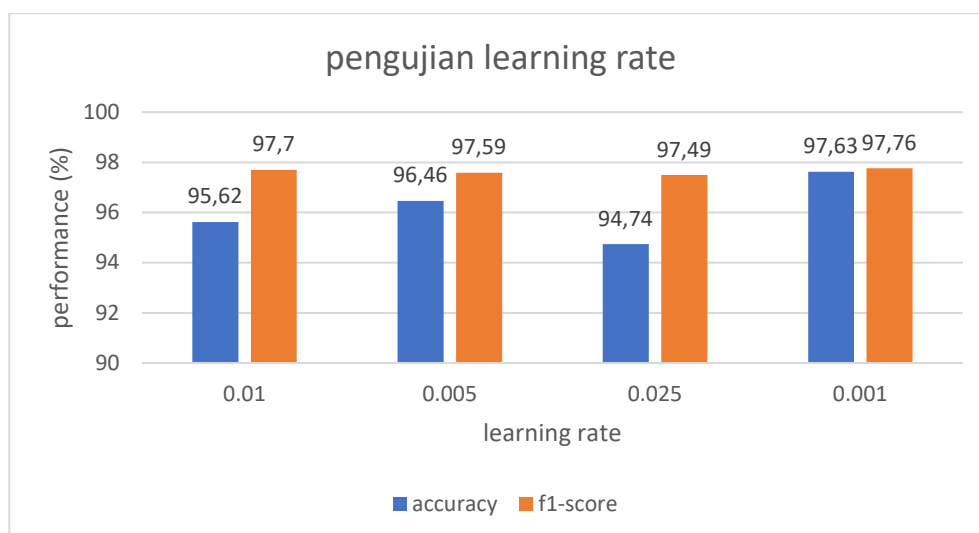
- Jumlah data yang digunakan sebanyak 236905 dengan label positif sebanyak 213214 dan label negatif sebanyak 21924.
- Teknik pembagian data untuk evaluasi model adalah split ratio dengan perbandingan 90:10, oleh karena itu jumlah data training sebanyak 213214 dan data testing sebanyak 23691.
- *Loss function* yang digunakan adalah *binary crossentropy*.
- *Dropout* menggunakan *range* 0.3.
- *Optimizer* yang digunakan adalah adam.

- *Batch size* yang digunakan sebanyak 64.
- *Units* yang digunakan sebanyak 64 *neuron*.

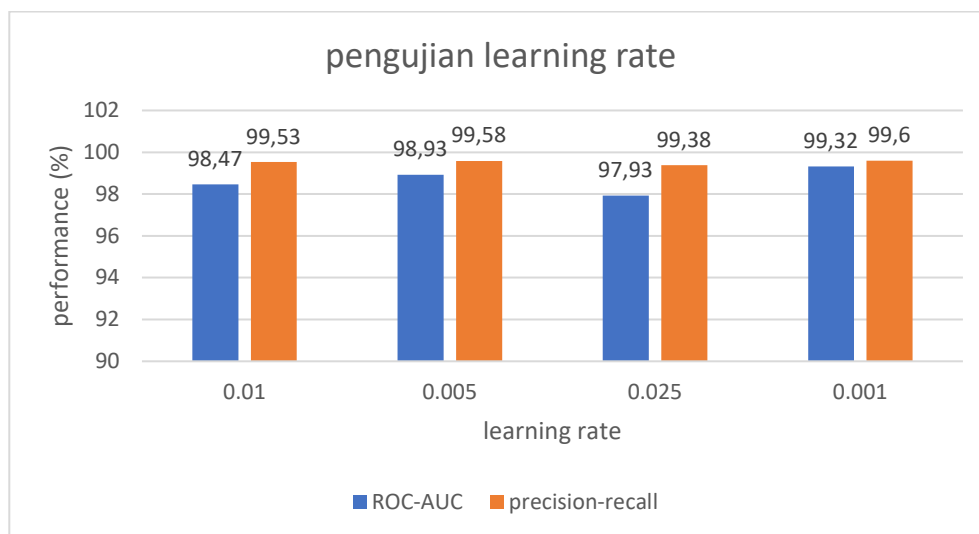
Skenario pengujian pada data *balance*:

- Jumlah data yang digunakan sebanyak 429962 dengan label positif dan negatif sebanyak 214981.
- Teknik pembagian data untuk evaluasi model adalah split ratio dengan perbandingan 90:10, oleh karena itu jumlah data training sebanyak 386965 dan data testing sebanyak 42997.
- *Loss function* yang digunakan adalah *binary crossentropy*.
- *Dropout* menggunakan *range* 0.3.
- *Optimizer* yang digunakan adalah adam.
- *Batch size* yang digunakan sebanyak 128.
- *Units* yang digunakan sebanyak 32 *neuron*.

Hasil pengujian seperti pada gambar 4.9 dan gambar 4.10.



Gambar 4.9 Pengujian akurasi dan f1-score pada learning rate



Gambar 4.10 Pengujian ROC-AUC dan precision-recall pada learning rate

Pada gambar 4.9 dan gambar 4.10 menunjukkan *learning rate* terbaik adalah 0.001. Hal ini dikarenakan *learning rate* 0.001 dapat membantu mencegah kehilangan informasi penting, karena model melakukan perubahan bobot yang lebih kecil pada setiap iterasi, sehingga dapat mempertahankan informasi yang lebih *smooth*.

4.5 Komparatif Eksperimen

Komparatif eksperimen digunakan untuk membandingkan dan mengevaluasi arsitektur yang diusulkan menggunakan data ulasan PLN Mobile yang *balance* dan *imbalace* dengan tujuan mengukur performanya melalui mekanisme tertentu. Mekanisme pada penelitian ini ada dua, yang pertama membandingkan menggunakan dan tanpa menggunakan *global attention* dan yang kedua membandingkan dengan metode analisis sentimen yang lain. Parameter yang digunakan untuk komparatif eksperimen seperti pada tabel 4.3.

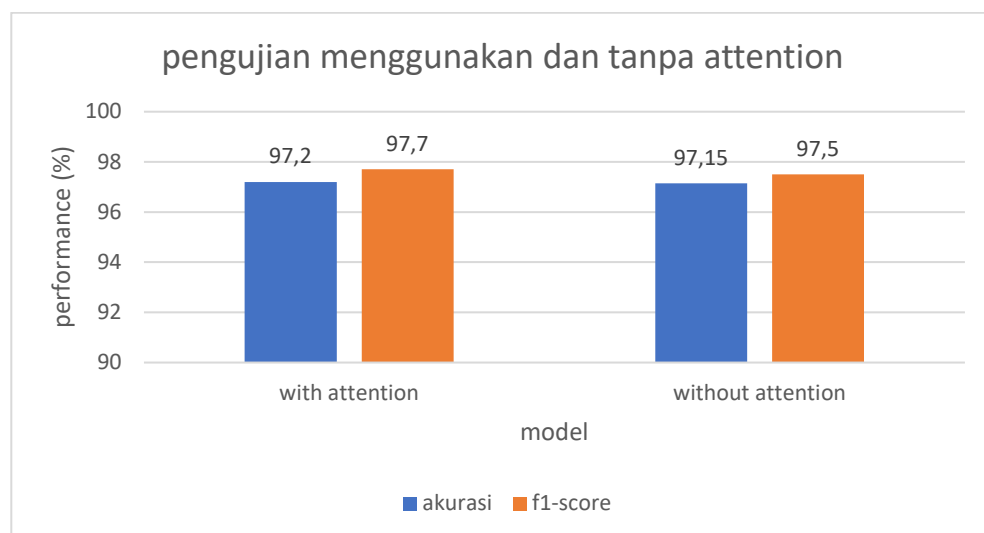
Tabel 4.3 Hyperparameter list

| Hyperparameter | Value (data balance) | Value (data imbalance) |
|----------------|----------------------|------------------------|
| dropout | 0.3 | 0.3 |
| optimizer | adam | adam |
| Batch size | 128 | 64 |
| units | 32 | 64 |
| learning rate | 0.001 | 0.001 |
| Loss function | binary crossentropy | binary crossentropy |

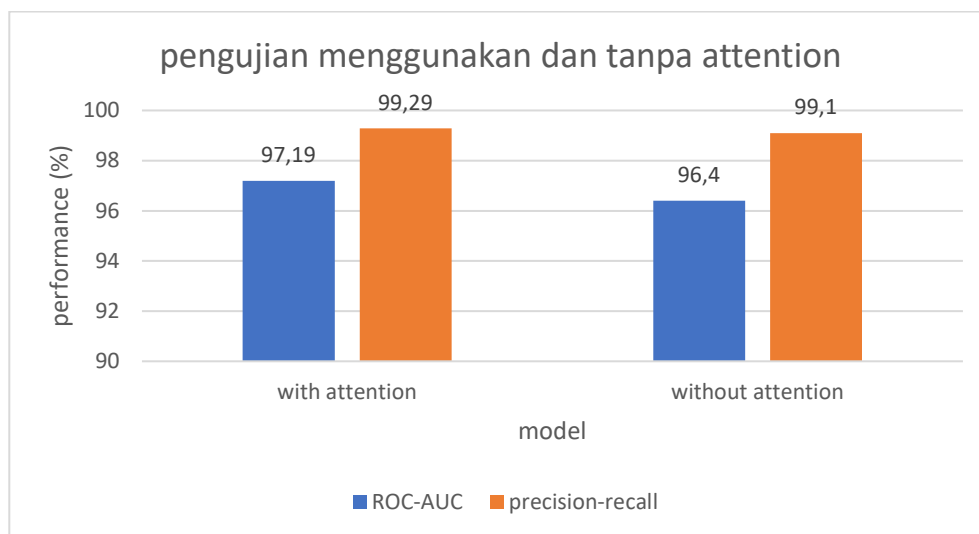
Daftar *hyperparameter* pada tabel 4.3 didapat dari pengujian *hyperparameter* pada bab 4.4.1 sampai 4.4.5 yang merupakan *hyperparameter* terbaik.

4.5.1 Perbandingan menggunakan dan tanpa *global attention*

Perbandingan menggunakan dan tanpa *global attention* bertujuan untuk menguji seberapa baik arsitektur ketika ditambahkan *global attention* seperti pada gambar 4.11 dan gambar 4.12.



Gambar 4.11 Pengujian akurasi dan f1-score pada global attention

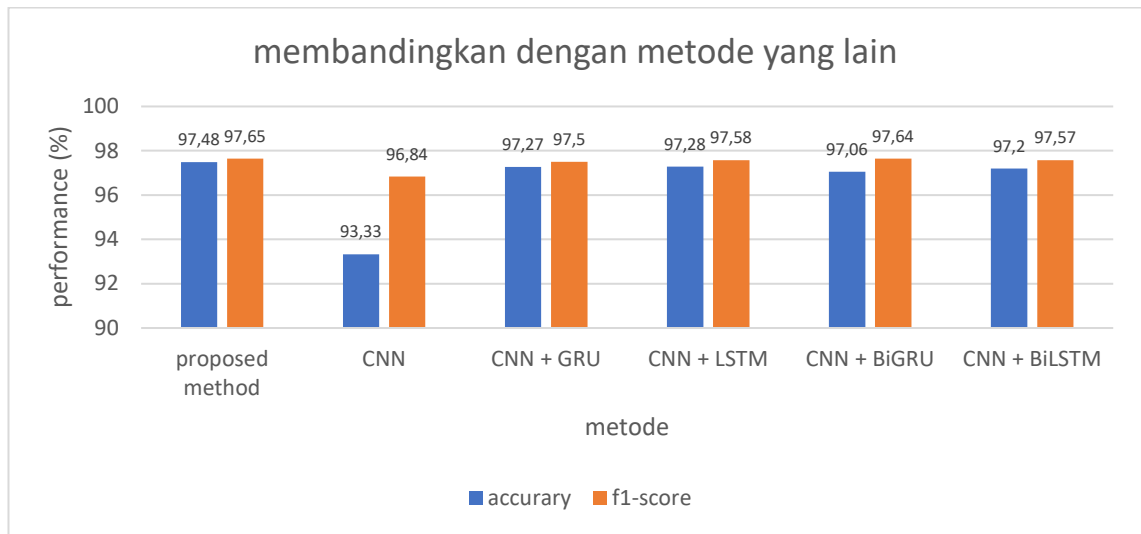


Gambar 4.12 Pengujian ROC-AUC dan precision-recall pada global attention

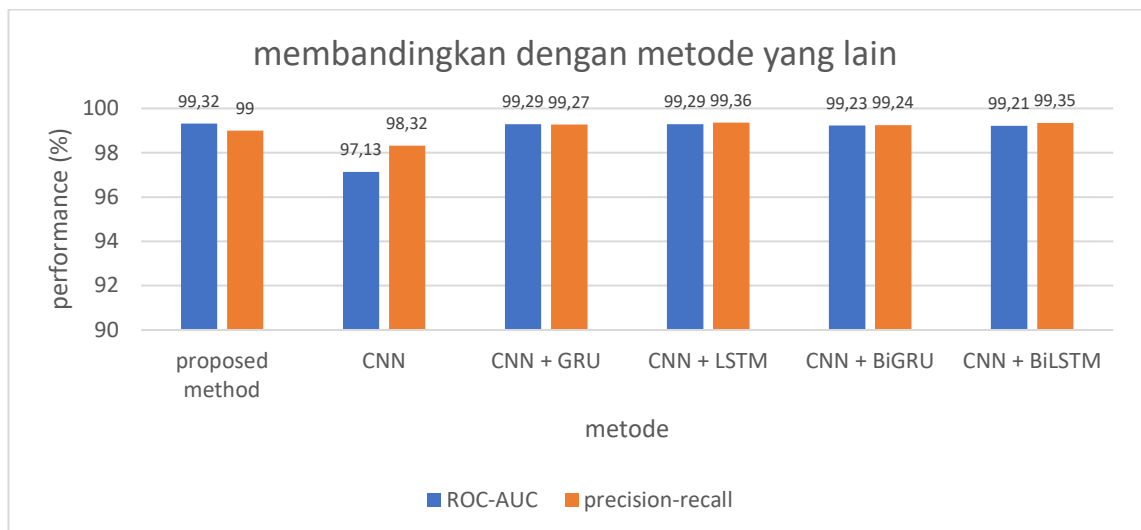
Pada gambar 4.11 dan gambar 4.12 menunjukkan ketika *global attention* diterapkan dalam arsitektur maka performa meningkat. Hal ini dikarenakan *global attention* memfokuskan kata-kata yang paling berkontribusi pada *sequence learning* sehingga mampu menangkap informasi penting dengan begitu performa menjadi meningkat.

4.5.2 Perbandingan dengan metode analisis sentimen yang lain

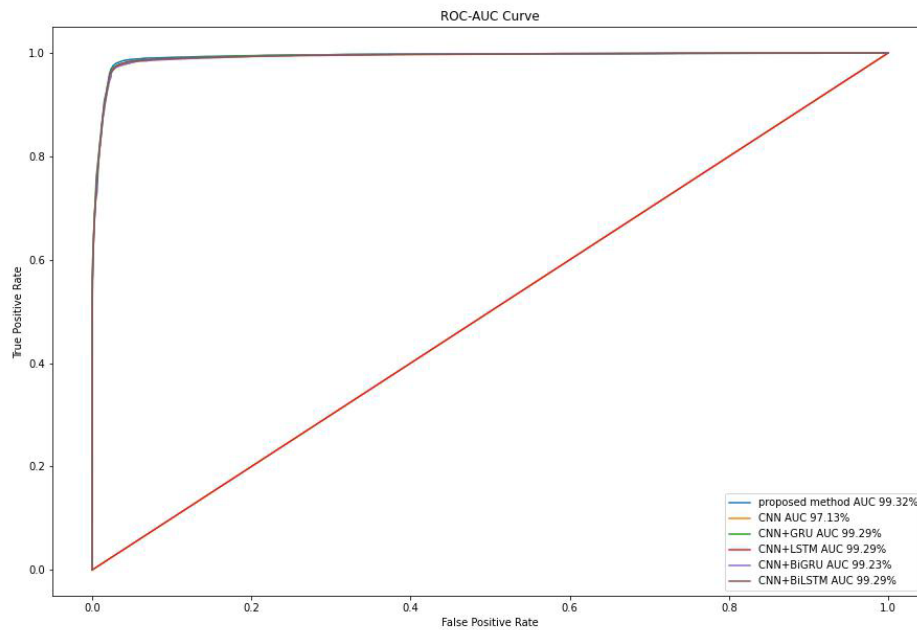
Untuk lebih membuktikan efektifitas metode analisis sentimen yang diusulkan pada penelitian ini. Metode yang diusulkan akan dibandingkan dengan metode analisis sentimen yang lain yaitu: CNN, CNN+GRU, CNN+LSTM, CNN+BiGRU, dan CNN+BiLSTM. Pengujian menggunakan data yang sama yaitu ulasan PLN Mobile yang *balance* dan *imbalance*. Hasil pengujian seperti pada gambar 4.13 sampai gambar 4.16.



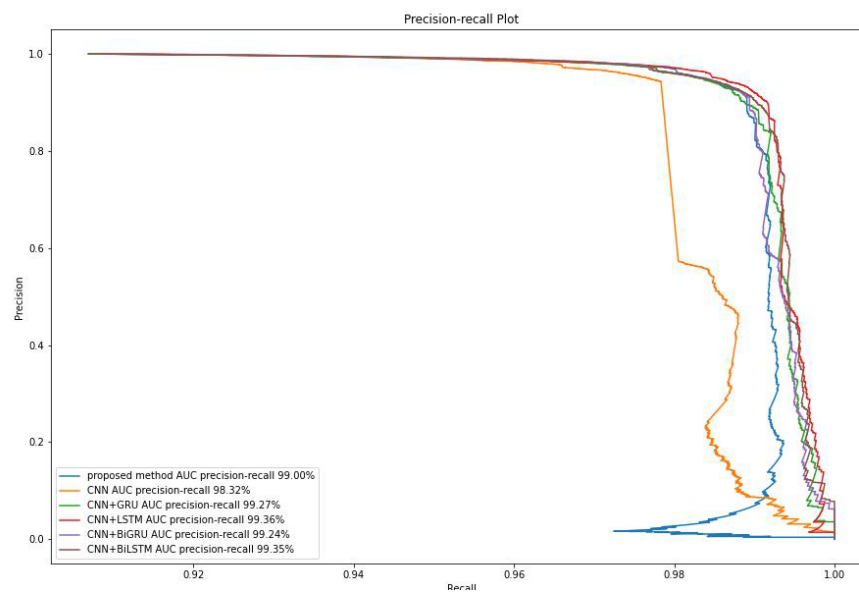
Gambar 4.13 Pengujian akurasi dan f1-score pada metode yang lain



Gambar 4.14 Pengujian ROC-AUC dan precision-recall pada metode yang lain



Gambar 4.15 Kurva ROC-AUC pada perbandingan model



Gambar 4.16 Kurva precision-recall pada perbandingan model

Pada gambar 4.13, gambar 4.14, gambar 4.16 menunjukkan pengujian model yang diusulkan mengungguli metode yang lain dengan hasil akurasi 97.48%, *f1-score* 97.65%, ROC-AUC 99.32%. Hal ini dikarenakan *global*

attention mampu menangkap informasi penting dari GRU, ditambah GRU menggunakan dua arah (*bidirectional*) sehingga dapat mempelajari semantik dari belakang ke depan (*forward*) dan dari belakang ke depan (*backward*). Ketika model CNN dikombinasikan dengan model LSTM, BiLSTM, GRU, BiGRU memang dapat mengekstraksi semantik seperti *global attention* akan tetapi pada kasus *sequence learning* CNN kurang efektif. Hal ini disebabkan ketika bekerja pada *sequence learning* CNN mengabaikan *sequence* pada data karena *convolution layer* hanya melihat blok-blok kecil dari data *input* dan tidak memiliki kemampuan untuk menangkap konteks dari *sequence* data. Akan tetapi pada gambar 4.17 terlihat metode yang diusulkan lebih kecil dari CNN+GRU, CNN+LSTM, CNN+BiGRU, dan CNN+BiLSTM ini dikarenakan parameter model yang kurang tepat untuk pengujian *precision-recall*.

4.6 Memahami Informasi Penting dalam Al-Quran

Memahami informasi penting dalam suatu konteks adalah hal yang krusial dilakukan, agar dapat memberikan gambaran inti terhadap sesuatu yang telah diketahui. Informasi penting ditemukan setelah memahami konteks dan mencari pokok yang dianggap penting. Jika dianalogikan dalam sastra, *context information* sama dengan pikiran pokok yaitu inti dari setiap teks yang ingin disampaikan penulis kepada pembaca. Dalam Al-Quran pun juga diperintahkan memahami kandungan-kandungan ayat dalam Al-Quran, sebagaimana QS. Shad 29.

كُتِبَ أَنْزَلْنَاهُ إِلَيْكَ مُبْرَكًا لِيَذَّبَرُوا إِلَيْهِ وَلِيَتَذَكَّرَ أُولُوا الْأَلْبَابِ

Artinya:

“Kitab suci (Al-Qur'an) yang Kami wahyukan kepadamu adalah penuh dengan keberkahan, sehingga diharapkan mereka dapat memahami dengan sepenuh hati makna dari setiap ayatnya, serta memberikan pembelajaran berharga bagi mereka yang memiliki akal sehat.”

Menurut tafsir Tahlili, Allah telah menurunkan Al-Quran kepada Rasulullah SAW dan pengikutnya sebagai kitab yang sempurna, berisi bimbingan yang bermanfaat untuk umat manusia. Bimbingan tersebut membantu manusia untuk hidup bahagia dan sejahtera di dunia dan di akhirat. Dengan merenungkan isi Al-Quran, manusia dapat menemukan cara untuk mengatur kehidupan mereka di dunia. Kisah dan perumpamaan dari umat terdahulu juga dapat dijadikan pelajaran dalam mencapai tujuan hidup dan mengatasi masalah yang dihadapi. Al-Quran diturunkan agar dipahami dengan benar dan diamalkan sesuai dengan petunjuk rasul dan ilmu pengetahuan yang relevan, baik yang berkaitan dengan bahasa maupun perkembangan masyarakat. Untuk memahami petunjuk dalam Al-Quran, manusia harus mengikuti tuntunan rasul dan menggunakan ilmu pengetahuan hasil pengalaman dan pemikiran mereka.

Menurut Al-hasan al-Bashri, ayat ini menjelaskan bahwa banyak orang yang menghafal Al-Quran tanpa memahami maknanya. Mereka mengabaikan ketentuan-ketentuan Al-Quran dan hanya fokus pada menghafal huruf-hurufnya. Sehingga, meskipun mereka mengatakan telah membaca seluruh Al-Quran, sebenarnya mereka telah melewatkan seluruh isi dan tuntunan Al-Quran pada perilaku dan budi pekerti mereka. Al-hasan al-Bashri menekankan bahwa

menghafal Al-Quran tanpa memahami dan mengamalkannya adalah sia-sia dan tidak berguna, dan orang-orang seperti itu tidak layak menjadi ahli hikmah dan guru. Dia berharap agar Allah tidak memperbanyak jumlah orang yang seperti itu.

Ibnu Mas'ud mengatakan: Orang-orang di kalangan kami biasanya belajar sepuluh ayat Al-Quran dengan sungguh-sungguh dan mereka tidak beralih ke ayat lain sebelum memahami sepenuhnya kandungan ayat-ayat tersebut dan menerapkan isinya dalam kehidupan mereka (Riwayat Ahmad).

BAB V

PENUTUP

5.1 Kesimpulan

Penelitian ini bertujuan untuk mengatasi masalah pada algoritma *deep learning* seperti LSTM dan GRU yang tidak bisa menangkap informasi penting pada *sequence learning*. Metode yang diusulkan untuk mengatasi masalah tersebut adalah menggunakan *bidirectional* GRU dengan mengkombinasikan *global attention* dan *word2vec* sebagai *word embedding*. *Global attention* bekerja dengan memfokuskan kata-kata yang paling berkontribusi sehingga membentuk seperti *keyword*. Model yang diusulkan kemudian dilakukan *hyperparameter setting* untuk menentukan parameter terbaik dan membandingkan dengan metode analisis sentimen yang lain.

Model tersebut dilatih menggunakan *dataset* ulasan PLN Mobile. Untuk memvalidasi model, penulis membandingkan dengan lima metode yaitu CNN, CNN+LSTM, CNN+GRU, CNN+BiLSTM, dan CNN+BiGRU dengan data yang sama hasilnya dapat disimpulkan bahwasanya metode yang diusulkan mengungguli dengan akurasi 97.48%, *f1-score* 97.65%, ROC-AUC 99.32%. Namun, untuk pengujian *precision-recall* lebih kecil dari CNN+GRU, CNN+LSTM, CNN+BiGRU, dan CNN+BiLSTM ini dikarenakan parameter model yang kurang tepat untuk pengujian *precision-recall*.

5.2 Saran

Adapun saran untuk penelitian selanjutnya adalah sebagai berikut:

1. Menggunakan *attention mechanism* yang lain seperti *self-attention*, *multi-head attention* dan *hierarchical attention* untuk meningkatkan performa.
2. Menerapkan pengujian *cross validation* pada saat *training* untuk mengurangi *overfitting* dan *underfitting*.
3. Menggunakan *word embedding* yang lain seperti *GloVe*, *FastText*.
4. Menerapkan model yang diusulkan untuk aplikasi yang nyata.

DAFTAR PUSTAKA

- Alshuwaier, F., Areshey, A., & Poon, J. (2022). Applications and Enhancement of Document-Based Sentiment Analysis in Deep learning Methods: Systematic Literature Review. *Intelligent Systems with Applications*, 15(April), 200090. <https://doi.org/10.1016/j.iswa.2022.200090>
- Bahdanau, D., Cho, K., Bengio, Y., By, P., & Aharoni, R. (2014). *Machine Translation is Everywhere*.
- Bhuvaneswari, A., Jones Thomas, J. T., & Kesavan, P. (2019). Embedded Bi-directional GRU and LSTM Learning Models to Predict Disaster on Twitter Data. *Procedia Computer Science*, 165, 511–516. <https://doi.org/10.1016/j.procs.2020.01.020>
- Bing Liu. (2012). *Sentiment Analysis and Opinion Mining*. 148, 148–162.
- Duan, J., Luo, B., & Zeng, J. (2020). Semi-supervised learning with generative model for sentiment classification of stock messages. *Expert Systems with Applications*, 158, 113540. <https://doi.org/10.1016/j.eswa.2020.113540>
- Han, Y., Liu, M., & Jing, W. (2020). Aspect-Level Drug Reviews Sentiment Analysis Based on Double BiGRU and Knowledge Transfer. *IEEE Access*, 8, 21314–21325. <https://doi.org/10.1109/ACCESS.2020.2969473>
- Hemmatian, F., & Sohrabi, M. K. (2019). A survey on classification techniques for opinion mining and sentiment analysis. *Artificial Intelligence Review*, 52(3), 1495–1545. <https://doi.org/10.1007/s10462-017-9599-6>
- Kumar, A., & Jaiswal, A. (2020). Systematic literature review of sentiment analysis on Twitter using soft computing techniques. *Concurrency and Computation: Practice and Experience*, 32(1), 1–29. <https://doi.org/10.1002/cpe.5107>
- Li, J., Xu, Y., & Shi, H. (2019). Bidirectional LSTM with Hierarchical Attention for Text Classification. *2019 IEEE 4th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC), Iaeac*, 456–459. <https://doi.org/10.1109/IAEAC47372.2019.8997969>
- Li, W., Qi, F., Tang, M., & Yu, Z. (2020). Bidirectional LSTM with self-attention mechanism and multi-channel features for sentiment classification. *Neurocomputing*, 387, 63–77. <https://doi.org/10.1016/j.neucom.2020.01.006>
- Ligthart, A., Catal, C., & Tekinerdogan, B. (2021). Systematic reviews in sentiment analysis: a tertiary study. In *Artificial Intelligence Review* (Vol. 54, Issue 7). Springer Netherlands. <https://doi.org/10.1007/s10462-021-09973-3>
- Lynn, H. M., Pan, S. B., & Kim, P. (2019). A Deep Bidirectional GRU Network Model for Biometric Electrocardiogram Classification Based on Recurrent Neural Networks. *IEEE Access*, 7, 145395–145405. <https://doi.org/10.1109/ACCESS.2019.2939947>
- Ma, Y., Peng, H., & Cambria, E. (2018). Targeted aspect-based sentiment analysis via embedding commonsense knowledge into an attentive LSTM. *32nd AAAI Conference on Artificial Intelligence, AAAI 2018*, 5876–5883. <https://doi.org/10.1609/aaai.v32i1.12048>
- Majumder, N., Poria, S., Peng, H., Chhaya, N., Cambria, E., & Gelbukh, A.

- (2019). Sentiment and Sarcasm Classification With Multitask Learning. *IEEE Intelligent Systems*, 34(3), 38–43. <https://doi.org/10.1109/MIS.2019.2904691>
- Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient Estimation of Word Representations in Vector Space. *1st International Conference on Learning Representations, ICLR 2013 - Workshop Track Proceedings*, 1–12. <http://arxiv.org/abs/1301.3781>
- Mite-Baidal, K., Delgado-Vera, C., Solís-Avilés, E., Espinoza, A. H., Ortiz-Zambrano, J., & Varela-Tapia, E. (2018). Sentiment analysis in education domain: A systematic literature review. *Communications in Computer and Information Science*, 883, 285–297. https://doi.org/10.1007/978-3-030-00940-3_21
- Ni, R., & Cao, H. (2020). Sentiment Analysis based on GloVe and LSTM-GRU. *Chinese Control Conference, CCC, 2020-July*, 7492–7497. <https://doi.org/10.23919/CCC50068.2020.9188578>
- Pozzi, F. A., Fersini, E., Messina, E., & Liu, B. (2017). Sentiment Analysis in Social Networks. In *Sentiment Analysis in Social Networks*. Elsevier. <https://doi.org/10.1016/C2015-0-01864-0>
- Rhanoui, M., Mikram, M., Yousfi, S., & Barzali, S. (2019). A CNN-BiLSTM Model for Document-Level Sentiment Analysis. *Machine Learning and Knowledge Extraction*, 1(3), 832–847. <https://doi.org/10.3390/make1030048>
- Wang, G., Sun, J., Ma, J., Xu, K., & Gu, J. (2014). Sentiment classification: The contribution of ensemble learning. *Decision Support Systems*, 57(1), 77–93. <https://doi.org/10.1016/j.dss.2013.08.002>
- Xu, G., Meng, Y., Qiu, X., Yu, Z., & Wu, X. (2019). Sentiment analysis of comment texts based on BiLSTM. *IEEE Access*, 7(c), 51522–51532. <https://doi.org/10.1109/ACCESS.2019.2909919>
- Yadav, R. K., Jiao, L., Goodwin, M., & Granmo, O. C. (2021). Positionless aspect based sentiment analysis using attention mechanism[Formula presented]. *Knowledge-Based Systems*, 226, 107136. <https://doi.org/10.1016/j.knosys.2021.107136>
- Zeng, L., Ren, W., & Shan, L. (2020). Attention-based bidirectional gated recurrent unit neural networks for well logs prediction and lithology identification. *Neurocomputing*, 414, 153–171. <https://doi.org/10.1016/j.neucom.2020.07.026>
- Zhu, Y., Wang, M., Liu, S., Song, C., Wang, Z., Wang, P., & Qin, X. (2019). Sentiment analysis based on attention mechanisms and bi-directional LSTM fusion model. *Proceedings - 2019 IEEE SmartWorld, Ubiquitous Intelligence and Computing, Advanced and Trusted Computing, Scalable Computing and Communications, Internet of People and Smart City Innovation, SmartWorld/UIC/ATC/SCALCOM/IOP/SCI 2019*, 865–868. <https://doi.org/10.1109/SmartWorld-UIC-ATC-SCALCOM-IOP-SCI.2019.00177>