ANALISIS SENTIMEN MENGGUNAKAN *LONG SHORT-TERM MEMORY* TERHADAP ULASAN RUMAH MAKAN PADANG PAYAKUMBUAH

SKRIPSI

Oleh :
SALMA CHESHA PUTRI PRAMUDYA WARDANI
NIM. 210605110013

PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2025

ANALISIS SENTIMEN MENGGUNAKAN *LONG SHORT-TERM MEMORY* TERHADAP ULASAN RUMAH MAKAN PADANG PAYAKUMBUAH

SKRIPSI

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

Oleh :
SALMA CHESHA PUTRI PRAMUDYA WARDANI
NIM. 210605110013

PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM MALANG
2025

HALAMAN PERSETUJUAN

ANALISIS SENTIMEN MENGGUNAKAN *LONG SHORT-TERM MEMORY* TERHADAP ULASAN RUMAH MAKAN PADANG PAYAKUMBUAH

SKRIPSI

Oleh :
SALMA CHESHA PUTRI PRAMUDYA WARDANI
NIM. 210605110013

Telah Diperiksa dan Disetujui untuk Diuji:
Tanggal: 5 Mei 2025

Pembimbing I,	Pembimbing II,
Prof. Dr. Muhammad Faisal, M.T	Ajib Hanani, S.Kom, M.T
NIP. 19740510 200501 1 007	NIP. 19840731 202321 1 013

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

Dr. Ir. Kachrul Kurniawan, M.MT., IPU
NIP. 19771020 200912 1 001

HALAMAN PENGESAHAN

ANALISIS SENTIMEN MENGGUNAKAN *LONG SHORT-TERM MEMORY* TERHADAP ULASAN RUMAH MAKAN PADANG PAYAKUMBUAH

SKRIPSI

Oleh :
SALMA CHESHA PUTRI PRAMUDYA WARDANI
NIM. 210605110013

Telah Dipertahankan di Depan Dewan Penguji Skripsi
dan Dinyatakan Diterima Sebagai Salah Satu Persyaratan
Untuk Memperoleh Gelar Sarjana Komputer (S.Kom)
Tanggal: 20 Mei 2025

Susunan Dewan Penguji

Ketua Penguji	: Prof. Dr. Suhartono, S.Si., M.Kom NIP. 19680519 200312 1 001	( )
Anggota Penguji I	: Syahiduz Zaman, M.Kom NIP. 19700502 200501 1 005	( )
Anggota Penguji II	: Prof. Dr. Muhammad Faisal, M.T NIP. 19740510 200501 1 007	( )
Anggota Penguji III	: Ajib Hanani, S.Kom, M.T NIP. 19840731 202321 1 013	( )

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

Dr. Fachrul Kurniawan, M.MT., IPU
NIP. 19771020 200912 1 001

PERNYATAAN KEASLIAN TULISAN

Saya yang bertanda tangan di bawah ini:

Nama	: Salma Chesha Putri Pramudya Wardani
NIM	: 210605110013
Fakultas / Program Studi	: Sains dan Teknologi / Teknik Informatika
Judul Skripsi	: Analisis Sentimen Menggunakan *Long Short Term Memory* Terhadap Ulasan Rumah Makan Padang Payakumbuah

Menyatakan dengan sebenarnya bahwa Skripsi yang saya tulis ini benar-benar merupakan hasil karya saya sendiri, bukan merupakan pengambil alihan 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 skripsi ini merupakan hasil jiplakan, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Malang, 20 November 2025
Yang membuat pernyataan,

Salma Chesha Putri Pramudya Wardani
NIM. 210605110013

MOTTO

"Life begins at the end of your comfort zone"

“Janganlah hendaknya kamu kuatir tentang apapun juga, tetapi nyatakanlah dalam segala hal keinginanmu kepada Allah dalam doa dan permohonan dengan ucapan syukur”

HALAMAN PERSEMBAHAN

Dengan rasa syukur kepada Allah SWT atas segala rahmat dan kemudahan-Nya, sehingga skripsi ini dapat terselesaikan dengan baik. Karya ini penulis persembahkan kepada Mama tersayang, Sutini dan Papa tersayang, Sucipto yang selalu mengalirkan kasih sayang, usaha terbaik, nasehat yang membangun, dukungan baik materi maupun non materi, dan do'a yang tiada henti. Kakak tersayang, Thanthio Eka Putra Pradana dan Evy Novitasari yang menjadi panutan dan dorongan untuk terus maju hingga skripsi ini dapat diselesaikan. Segenap keluarga besar, yang mengiringi perjalanan penulis dengan do'a dan dukungan. Aster Teknik Informatika angkatan 2021, yang telah berjuang bersama dari awal hingga akhir. Terakhir, kepada diri sendiri terima kasih karena sudah berusaha, bertahan, dan berjuang sejauh ini sampai semua terselesaikan.

KATA PENGANTAR

Assalamu'alaikum Warahmatullahi Wabarakatuh

Puji syukur senantiasa penulis panjatkan kehadirat Allah SWT, atas berkat rahmat dan limpahan hidayah-Nya, sehingga penulis dapat menyelesaikan skripsi yang berjudul "Analisis Sentimen Menggunakan *Long Short-Term Memory* Terhadap Ulasan Rumah Makan Padang Payakumbuah" dengan baik dan lancar sebagai salah satu persyaratan untuk memperoleh gelar sarjana pada program studi Teknik Informatika jenjang Strata-1 Universitas Islam Negeri Maulana Malik Ibrahim Malang. Shalawat serta salam tetap tercurahkan kepada Nabi Muhammad SAW, yang telah membawa kita dari zaman jahiliyah menuju zaman yang lurus yaitu Addinul Islam.

Penulis mengucapkan rasa terima kasih yang begitu besar kepada seluruh pihak yang memberikan dukungan dan motivasi kepada penulis sehingga dapat menyelesaikan skripsi ini. Ucapan terima kasih ini penulis disampaikan kepada:

1. Prof. Dr. M. Zainuddin, M.A., selaku rektor Universitas Islam Negeri Maulana Malik Ibrahim Malang.
2. Prof. Dr. Hj. Sri Harini, M.Si., selaku dekan Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang.
3. Dr. Ir. Fachrul Kurniawan, M.MT., IPU., selaku Ketua Program Studi Teknik Informatika Universitas Islam Negeri Maulana Malik Ibrahim Malang.
4. Prof. Dr. Muhammad Faisal, M.T., selaku dosen pembimbing I yang sudah sabar dan mendukung penulis dengan memberikan bimbingan, motivasi,

kritik dan saran untuk penulis dalam mengerjakan dan menyelesaikan skripsi ini.

5. Ajib Hanani, S.Kom, M.T., selaku dosen pembimbing II yang sudah memberikan motivasi, mendukung dan membimbing penulis untuk mengerjakan dan menyelesaikan skripsi ini.
6. Prof. Dr. Suhartono, S.Si, M.Kom dan Syahiduz Zaman, M.Kom., selaku dosen penguji I dan II yang sudah memberikan masukan dan saran kepada penulis untuk menyelesaikan skripsi ini.
7. Mama Sutini dan Papa Sucipto, selaku orang tua penulis yang senantiasa memberikan do'a tiada henti, semangat, dukungan, motivasi dan nasehat untuk bertanggung jawab dalam menyelesaikan skripsi ini. Menjadi garda terdepan untuk penulis dalam memberikan kekuatan dalam menghadapi tantangan dalam mengerjakan skripsi ini. Pilar utama dalam kesehatan dan kesuksesan perjalanan penulis menulis skripsi.
8. Fatchurrohman, M.Kom., selaku wali dosen yang sudah membantu penulis selama masa studi Program Studi Teknik Informatika.
9. Nia Faricha S, Si selaku admin Program Studi Teknik Informatika yang sudah membantu penulis selama masa studi dan menyelesaikan skripsi ini.
10. Segenap dosen dan jajaran staff Program Studi Teknik Informatika yang telah memberikan ilmu dan pengetahuan selama penulis masa studi.
11. Thanthio Eka Putra Pradana selaku kakak tersayang yang menjadi motivasi dan inspirasi penulis dalam menuju kesuksesan. Memberi motivasi,

dukungan penuh dan do'a untuk penulis selama masa studi sampai menyelesaikan skripsi ini.

12. Evy Novitasari selaku sepupu sekaligus kakak tersayang yang menjadi tempat berkeluh kesah, memberikan dukungan dan do'a serta semangat dalam menyelesaikan studi.
13. Kepada keluarga besar yang sudah memberikan do'a dan dukungan kepada penulis untuk tetap semangat menyelesaikan perkuliahan.
14. Sahabat sekamar, Najah Muchsin Sanin dan Sarah Arelia Rahmah yang sudah menemani penulis dari awal perkuliahan hingga saat ini. Sudah menjadi teman yang membantu penulis dalam perjalanan kuliahnya, menjadi tempat berkeluh kesah dan pendengar yang baik sekaligus konsultan untuk penulis.
15. Sahabat terdekat penulis, Mutiara Aprilia, Aisya Gusti, Fitria Susanti, Frisca Yuni, dan Nurjihan Nabilah yang sudah membantu dan mendukung penulis selama ini. Canda, tawa, suka dan duka yang dilewati bersama sudah memberikan warna kepada penulis selama masa perkuliahan.
16. Sahabat terbaik penulis, Safira Mutia, Serena Kalista, Ajeng Kusuma, Nadira Rahma, Agnes Felicia, Tasya Inaya dan Novita Imanuella yang sudah mendukung dan menjadi saksi selama ini dari masa bangku sekolah sampai bangku perkuliahan.
17. Sahabat penulis, Putri Purnamasari, Firzatul Mila, Daffa Pramuditya, Hamidah, Zidan, Heny, Adila, Atun, Rofidatus, Alima, Bibil, Nabila, Silvi, Azizah, Nadila, Khansa, Meilisa, Shafa, Izza, Sintia, Jihaan, Yoga, Varell,

Ganistya, Diva, Brenda, Khanza, Arini dan Yuddit yang senantiasa mendukung penulis dalam menjalani skripsi ini.

18. Kepada teman-teman angkatan 2021 “Aster” yang sudah menjadi teman dan saudara selama masa studi di Teknik Informatika.
19. Kepada teman-teman Medinfo Himpunan Mahasiswa Aktif 2023 yang sudah mendukung dan menjadi hiburan baru selama di perkuliahan.
20. Kepada Dra. Hj. Maftukhah selaku validator yang sudah membantu penulis dalam menyelesaikan skripsi ini.
21. Kepada Ibu Rob dan Bapak Mahfud selaku Ibu dan Ayah di Desa Kedok yang sudah memberi do’a, motivasi dan semangat untuk menjalankan skripsi hingga selesai.
22. Kepada seluruh pihak yang terlibat membantu, mendukung dan mendo’akan diberi kelancaran dalam menyelesaikan perkuliahan.
23. Terakhir, kepada diri sendiri yaitu Salma Chesha Putri Pramudya Wardani karena sudah berusaha, bertahan dan berjuang melewati setiap tantangan dari awal hingga berada di titik ini. Selamat karena sudah berhasil melewati satu titik terendah dalam perjalanan hidupnya. Terima kasih diriku karena dapat bertanggung jawab dengan apa yang sudah kamu pilih. Penulis menyadari berada di titik ini bukan karena keterpaksaan, tapi karena Allah SWT., tau bahwa penulis mampu melewatinya. Keringat dan air mata akan selalu terkenang dalam skripsi ini. Selamat menlanjutkan hidup diriku.

Penulis menyadari bahwa dalam penulisan skripsi masih terdapat kekurangan dan kesalahan. Penulis berharap semoga skripsi ini dapat memberikan manfaat kepada pemmbaca khususnya penulis sendiri.

Wassalamu'alaikum warahmatullahi wabarakatuh.

Malang, 20 Mei 2025

Penulis

DAFTAR ISI

DAFTAR GAMBAR

DAFTAR TABEL

ABSTRAK

Wardani, Salma Chesha Putri Pramudya. 2025. **Analisis Sentimen Menggunakan *Long Short-Term Memory* Terhadap Ulasan Rumah Makan Padang Payakumbuah.** Skripsi. Program Studi Teknik Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang. Pembimbing: (I) Prof. Dr. Muhammad Faisal, M.T (II) Ajib Hanani, S.Kom, M.T.

.

Kata Kunci: Analisis Sentimen, *K-Fold Cross Validation*, *Long Short-Term Memory* (LSTM), *Term Frequency-Inverse Document Frequency* (TF-IDF), Rumah Makan Padang Payakumbuah

Banyak rumah makan Padang di Indonesia, salah satunya Rumah Makan Padang Payakumbuah. Terdapat banyak ulasan dari pengunjung melalui platform seperti *Google Maps*. Ulasan-ulasan tersebut berisi opini pelanggan mengenai kualitas rumah makan tersebut. Namun, dengan jumlah ulasan yang banyak akan kesulitan untuk menganalisis sentimen. Penelitian ini bertujuan untuk mengetahui performa metode *Long Short-Term Memory* (LSTM) dalam menganalisis sentimen positif dan negatif dari ulasan pelanggan Rumah Makan Padang Payakumbuah. Data ulasan diperoleh melalui platform *Google Maps* di wilayah DKI Jakarta. Sebanyak 976 ulasan yang telah dilabeli positif dan negatif digunakan dalam penelitian ini. Tahapan yang dilakukan mencakup *preprocessing* data seperti data *cleaning*, *case folding*, *tokenization*, *stopword removal*, serta *stemming* dan normalisasi kata. Selanjutnya, pembobotan kata dilakukan menggunakan metode *Term Frequency-Inverse Document Frequency* (TF-IDF) untuk mempersiapkan data sebelum dimasukkan ke dalam model LSTM. Penelitian ini juga membandingkan hasil model LSTM pembagian data langsung tanpa *k-fold* dengan berbagai rasio 60:40, 70:30, 80:20 dan 90:10 dan menggunakan teknik *k-fold cross validation* dengan k=10, untuk mengukur performa model secara lebih akurat. Hasil penelitian menunjukkan bahwa model LSTM pada rasio 90:10 dengan model pembagian data tanpa *k-fold*, yaitu 90% data pelatihan dan 10% data pengujian mampu mendapatkan akurasi terbaik sebesar 84,69%. Sementara itu, penggunaan *k-fold cross validation* tidak memberikan peningkatan akurasi yang signifikan dibandingkan dengan model yang menggunakan pembagian data langsung. Penelitian ini diharapkan dapat memberikan kontribusi bagi Rumah Makan Padang Payakumbuah dalam meningkatkan kualitasnya berdasarkan analisis sentimen dari ulasan pelanggan.

ABSTRACT

Wardani, Salma Chesha Putri Pramudya. 2025. **Sentiment Analysis Using Long Short-Term Memory on Reviews of Padang Payakumbuah Restaurant.** Thesis. Informatics Engineering Study Program Faculty of Science and Technology Maulana Malik Ibrahim State Islamic University Malang. Advisor: (I) Prof. Dr. Muhammad Faisal, M.T (II) Ajib Hanani, S.Kom, M.T.

There are many Padang restaurants in Indonesia, one of which is Rumah Makan Padang Payakumbuah. There are many reviews from visitors through platforms such as Google Maps. These reviews contain customer opinions about the quality of the restaurant. However, with a large number of reviews it will be difficult to analyze sentiment. This study aims to determine the performance of the Long Short-Term Memory (LSTM) method in analyzing positive and negative sentiments from customer reviews of Rumah Makan Padang Payakumbuah. Review data is obtained through the Google Maps platform in the DKI Jakarta area. A total of 976 reviews that have been labeled positive and negative are used in this study. The stages carried out include data preprocessing such as data cleaning, case folding, tokenization, stopword removal, as well as stemming and word normalization. Furthermore, word weighting is performed using the Term Frequency-Inverse Document Frequency (TF-IDF) method to prepare the data before being incorporated into the LSTM model. This research also compares the results of the LSTM model of direct data sharing without k-fold with various ratios of 60:40, 70:30, 80:20 and 90:10 and uses the k-fold cross validation technique with k=10, to measure the performance of the model more accurately. The results showed that the LSTM model at a ratio of 90:10 with a data division model without k-fold, namely 90% *training* data and 10% *testing* data was able to get the best accuracy of 84.69%. Meanwhile, the use of k-fold cross validation does not provide a significant increase in accuracy compared to models that use direct data sharing. This research is expected to contribute to Payakumbuah Padang Restaurant in improving its quality based on sentiment analysis of customer reviews.

Key words: K-Fold Cross Validation, Long Short-Term Memory (LSTM), Payakumbuah Padang Restaurant, Sentiment Analysis, Term Frequency Inverse Document Frequency (TF-IDF)

الملخص

ورداني، سلمى تشيشا فراموديا فوتري. ٢٠٢٥. تحليل المشاعر باستخدام ذاكرة طويلة وقصيرة المدى تجاه تعليقات مطعم فادانج فاياكومبوه. البحث الجامعي. قسم الهندسة المعلوماتية، كلية العلوم والتكنولوجيا بجامعة مولانا مالك إبراهيم الإسلامية الحكومية مالانج. المشرف الأول: أ. د. محمد فيصل، الماجستير؛ المشرف الثاني: عجيب حناني، الماجستير.

الكلمات الرئيسية: تحليل مشاعر، تحقق متقاطع K-Fold، ذاكرة طويلة وقصيرة المدى (LSTM)، تردد الكلمة–تردد مستند عكسي (TF-IDF)، مطعم فادانج فاياكومبوه.

مطاعم فادانج كثيرة في إندونيسيا، أحدها هو مطعم فادانج فاياكومبوه. هناك تعليقات من الزوار من خلال منصات مثل جوجل الخرائط. تحتوي هذه المراجعات على آراء العملاء على جودة المطعم. ومع ذلك، مع عدد المراجعات التي سيجدها الكثيرون صعوبة في تحليل المشاعر. يهدف هذا البحث إلى معرفة أداء طريقة ذاكرة طويلة وقصيرة المدى (LSTM) في تحليل المشاعر الإيجابية والسلبية من تعليقات عملاء مطعم فاياكومبوه. تم الحصول على بيانات المراجعات من خلال منصة جوجل الخرائط في منطقة جاكرتا. تم استخدام ٩٧٦ تعليقًا تم تصنيفها على أنها إيجابية وسلبية في هذا البحث. شملت المراحل المدروسة معالجة مسبقة للبيانات مثل تنظيف البيانات، وتقليل حالة النص، وتحليل الكلمات، وإزالة الكلمات الشائعة، وكذلك تحديد الجذور وتطبيع الكلمات. علاوة على ذلك، يتم ترجيح الكلمة باستخدام طريقة تردد الكلمة–تردد المستند العكسي (TF-IDF) لإعداد البيانات قبل إدراجها في نموذج LSTM. يقارن هذا البحث أيضًا نتائج نموذج LSTM لتقسيم البيانات مباشرة دون استخدام تقنية k-fold مع نسب مختلفة ٦٠:٤٠ ٧٠:٣٠ ٨٠:٢٠، و٩٠:١٠، بالإضافة إلى استخدام تقنية تحقق متقاطع k-fold مع ك=١٠، لقياس أداء النموذج بدقة أكبر. أظهرت نتائج البحث أن نموذج LSTM عند نسبة ٩٠:١٠ بتقسيم البيانات دون استخدام k-fold، أي ٩٠% من بيانات التدريب و١٠% من بيانات الاختبار، استطاع تحقيق أفضل دقة بلغت ٨٤.٦٩%. بينما لم توفر استخدام تقنية تحقق متقاطع k-fold زيادة ملحوظة في الدقة مقارنة بالنموذج الذي استخدم تقسيم البيانات المباشر. يأمل أن يسهم هذا البحث في تحسين جودة مطعم فادانج فاياكومبو في ضوء تحليل المشاعر من تعليقات العملاء.

BAB I

PENDAHULUAN

1.1 Latar Belakang

Rumah makan merupakan suatu jenis usaha yang bergerak di bidang kuliner dan menyediakan berbagai macam hidangan untuk konsumsi, baik makanan maupun minuman, dengan dilengkapi berbagai fasilitas dan peralatan yang dibutuhkan untuk proses memasak, penyimpanan, hingga penyajian. Sebuah rumah makan, terdapat variasi hidangan makanan, ada rumah makan modern yang menyajikan makanan *western* dan ada rumah makan tradisional yang menyajikan makanan khas asli Indonesia (Muthia, 2018). Salah satu restoran yang menyediakan makanan tradisional yaitu restoran atau rumah makan Padang, merupakan makanan asli Minang (Sartika et al., 2024).

Sudah banyak rumah makan Padang yang tersebar di seluruh Indonesia, salah satunya rumah makan Padang Payakumbuah, yang didirikan oleh seorang influencer Bernama Arief Muhammad dan sudah memiliki beberapa cabang di kota besar seperti Jakarta, Surabaya, Semarang dan Malang. Rumah makan Padang Payakumbuah pertama kali dibuka pada 24 Juli 2022 dengan cabang pertama di Serpong Tangerang Selatan, sampai saat ini rumah Makan Padang Payakumbuah sudah memiliki 16 cabang antara lain berlokasi di Ciater BSD, Gading Serpong, Veteran Bintaro, Bogor, Tebet, Kemang, Juanda Bekasi, Islamic Vilage, Menteng Cikini, Citarum Bandung, Cibubur, Depok, Semarang, Gubeng Surabaya, Benhil dan yang baru buka yaitu di Malang (Jurianto, 2023).

Dengan banyaknya cabang rumah makan Padang membuat pengunjung mencari saran dan rekomendasi mengenai rumah makan Padang Payakumbuah mana yang akan mereka datangi mulai dari rasa masakan, tempat, dan pelayanan dari pengunjung lain, dengan menggunakan situs ulasan online salah satunya dari *Google Maps* (Muthia, 2018). *Google Maps* merupakan peta online dari Google yang dapat diakses melalui *browser* atau perangkat *mobile*, kemampuan dari *Google Maps* dapat melihat peta jalan, fitur *street view*, kondisi jalur lalu lintas secara *realtime*, perubahan rute menyesuaikan transportasi, dan mencari seluruh lokasi restoran, kantor, rumah, dan lainnya disertai fitur ulasan (Wijayanto et al., 2020).

Ulasan *Google Maps* membantu para pengunjung untuk mengetahui kualitas dari setiap cabang rumah makan Padang Payakumbuah sehingga pengunjung dapat membuat keputusan yang lebih baik. Namun, seringkali ulasan-ulasannya kurang lengkap atau tidak sesuai dengan kenyataan. Tidak menutup kemungkinan terdapat pengunjung yang jahil untuk menulis ulasan tidak jujur terhadap rumah makan Padang Payakumbuah.

Berkata jujur telah dijelaskan di dalam Al-Qur'an surah *Al-Ahzab* ayat 70, sebagai berikut:

يَا أَيُّهَا الَّذِينَ آمَنُوا اتَّقُوا اللَّهَ وَقُولُوا قَوْلًا سَدِيدًا

"Hai orang-orang yang beriman, bertakwalah kamu kepada Allah dan katakanlah perkataan yang benar." (*Q.S Al-Ahzab: 70*).

Menurut tafsir Ibnu Katsir, dalam ayat tersebut menjelaskan Allah SWT memerintahkan hamba-hamba-Nya yang beriman untuk tetap bertakwa dan

menyembah-Nya seolah-olah mereka melihat-Nya secara langsung. Mereka juga diperintahkan untuk selalu berkata jujur, tanpa kebohongan atau penyimpangan. Allah berjanji bahwa jika mereka mematuhi perintah-perintah-Nya, Dia akan memberikan pahala dengan memperbaiki amal perbuatan mereka, memberikan taufik untuk berbuat kebaikan, serta mengampuni dosa-dosa mereka yang telah lalu. Untuk dosa-dosa yang mungkin dilakukan di masa depan, Allah akan memberikan ilham agar mereka bertobat (Al-Sheikh, 2004).

Pelajaran yang dapat di ambil dari surah *Al-Ahzab* ayat 70 yaitu mendorong kita untuk memberikan ulasan yang jujur terhadap sesuatu. Namun banyaknya ulasan pada *Google Maps* membuat restoran kesulitan menentukan apakah ulasan yang diterima lebih banyak positif atau negatif. Oleh karena itu, penting untuk memproses teks ulasan secara sistematis. Analisis sentimen menjadi solusi yang diperlukan untuk menyaring ulasan dan mengidentifikasi apakah ulasan tersebut bersifat positif atau negatif (Amalsyah et al., 2025).

Analisis sentimen atau *opinion mining*, adalah cabang dari *Natural Language Processing* (NLP) yang bertujuan untuk menganalisis pendapat seseorang, sentimen seseorang, evaluasi seseorang, sikap seseorang dan emosi seseorang ke dalam bahasa tertulis (Deviyanto & Wahyudi, 2018). Tugas utama analisis sentimen yaitu mengkategorikan sentimen dalam teks, berupa pendapat seseorang di dalam dokumen atau sebuah kalimat dengan tujuan untuk mengetahui sentimen tersebut merupakan sentimen positif atau negatif. Oleh karena itu, sebuah bisnis atau lembaga terkait harus menerapkan analisis sentimen untuk mengetahui bagaimana masyarakat merespons suatu produk atau layanan tertentu. Ini perlu

dilakukan agar bisnis atau lembaga terkait dapat menggunakannya sebagai alat untuk mengevaluasi upaya untuk meningkatkan produk atau kinerja mereka (Rizqilla et al., 2025).

Dalam menentukan suatu ulasan dapat dilakukan secara manual, ulasan tersebut positif atau negatif. Namun, seiring waktu semakin banyak ulasan yang masuk, maka semakin butuh banyak waktu untuk menentukan ulasan positif atau negatif. Maka dari itu, dalam penelitian ini, peneliti menggunakan model *Deep Learning* untuk menganalisis sentiment. *Deep Learning* memiliki beberapa metode seperti *Artificial Neural Networks, Convolutional Neural Networks, Recurrent Neural Networks, Long Short Term Memory*. Metode yang digunakan untuk mengklasifikasi teks yaitu *Long Short Term Memory* (LSTM) (Tul et al., 2017).

Pada penelitian analisis sentiment evaluasi terhadap pengajaran dosen di Perguruan Tinggi digunakan metode LSTM. Hasil pengujian mengidentifikasi bahwa metode LSTM memiliki performa yang baik dalam mengklasifikasi polaritas evaluasi, dengan memiliki tingkat akurasi 91.08% (Amrustian et al., 2022). Penelitian lain untuk menganalisis sentimen terkait vaksin Covid-19 melalui tweet di Twitter dengan menggunakan algoritma *Long Short-Term Memory* (LSTM). Penelitian ini membandingkan metode LSTM dengan *Recurrent Neural Network* (RNN) dan *Naïve Bayes* untuk mengetahui metode yang memberikan akurasi lebih tinggi. Hasil penelitian menunjukkan bahwa LSTM memperoleh nilai akurasi tertinggi sebesar 80,34%, sementara RNN dan *Naïve Bayes* menghasilkan akurasi sebesar 74% dan 71% (Maharani et al., 2024).

Berdasarkan penerapan model analisis sentimen beberapa sumber jurnal penelitian sebelumnya, maka peneliti menggunakan LSTM dengan dilakukan represensentasi kata menggunakan *Term Frequency-Inverse Document Frequency* (TF-IDF) sebagai metode penelitian dengan judul Analisis Sentiment Terhadap Review Rumah Makan Padang Payakumbuah Menggunakan Metode *Long Short Term Memory*. Diharapkan penelitian ini dapat digunakan untuk mengukur sentimen positif dan negatif ulasan pelanggan terhadap rumah makan Padang Payakumbuah. Peneliti melakukan evaluasi kinerja model dengan hasil evaluasi berupa persentase parameter akurasi, presisi, *recall*, dan F1-*Score*. Seberapa banyak data yang diprediksi dengan benar, baik untuk data positif maupun negatif, dibandingkan dengan seluruh data yang diuji disebut akurasi. Dengan kata lain, akurasi menunjukkan seberapa sering model membuat prediksi yang benar dari semua data yang ada (Novaković et al., 2017).

1.2 Rumusan Masalah

Bagaimana performa metode *Long Short-Term Memory* (LSTM) dalam menganalisis sentimen positif dan negatif dari ulasan pelanggan Rumah Makan Padang Payakumbuah ?

1.3 Batasan Masalah

Guna menghindari perluasan ruang lingkup penelitian, peneliti perlu menentukan batasan masalah. Adapun batasan masalah pada penelitian tersebut yaitu:

1. Data ulasan yang digunakan hanya cabang yang ada di wilayah DKI Jakarta
2. Data diambil dengan API *Google Maps* dari SERPAPI dengan cara *crawling* data menggunakan Bahasa pemrograman phyton.
3. Data diambil dari *Google Maps*.
4. Label sentimen terdapat dua ulasan yaitu positif dan negatif.

1.4 Tujuan Penelitian

Tujuan dari penelitian ini untuk mengetahui performa metode *Long Short-Term Memory* (LSTM) dalam menganalisis sentimen positif dan negatif dari ulasan pelanggan Rumah Makan Padang Payakumbuah.

1.5 Manfaat Penelitian

Terdapat beberapa manfaat pada penelitian ini bagi pihak rumah makan dan bagi peneliti, berikut manfaat penelitian ini adalah:

1. Bagi pihak rumah makan Padang Payakumbuah, diharapkan dapat memberikan informasi sentimen positif atau negatif untuk evaluasi dan meningkatkan kualitas rasa makanan, tempat dan pelayan rumah makan Padang Payakumbuah.
2. Bagi peneliti, studi ini diharapkan dapat menjadi referensi terhadap penelitian yang akan datang.

BAB II

STUDI PUSTAKA

2.1 Penelitian Terkait

Penelitian yang dilakukan Auliya Rahman Isnain dan kawan-kawan pada tahun 2022, peneliti melakukan penelitian analisis perbandingan algoritma *Long Short Term Memory* dan *Naïve Bayes* untuk analisis sentimen terhadap kebijakan pemerintah mengenai New Normal ditengah pandemi COVID-19. Data diambil dengan *crawling* dari twitter dengan *keyword* "NewNormal", mendapat total data sebanyak 1590 *tweets*. Data dibagi menjadi 80% data uji dan 20% data latih. Kemudian data melewati beberapa tahapan *preprocessing* yaitu *case folding*, *normalization*, *tokenization*, *slangword*, dan *stopword*. Setelah melakukan *preprocessing* dilakukan vektorisasi, dalam penelitian ini yaitu menggunakan *Word2Vec*. Hasil dari penelitian ini perbandingan 2 metode antara *Long Short-Term Memory* mendapat akurasi, presisi, *recall* dan *f-measure* dengan nilai sebesar 83.33%, sedangkan *Naïve Bayes* mendapatkan nilai akurasi, presisi, *recall* dan *f-measure* sebesar 82% (Isnain et al., 2022).

Dalam penelitian yang dilakukan Muhammad Afrizal Amrustian, Widi Widayat, dan Arif Muhammad Wirawan, mereka menganalisis sentimen terhadap evaluasi pengajaran dosen oleh mahasiswa di perguruan tinggi menggunakan metode *Long Short Term Memory* (LSTM). Dataset yang dipakai berjumlah 2.280, masukan teks dikumpulkan dua kali dalam satu semester dengan panjang setiap teks berkisar antara 3 hingga 50 kata. Sebelum pemodelan, peneliti menerapkan

praproses data yang mencakup *labelling* (menghapus kolom tak relevan dan menambahkan kolom label positif atau negatif), konversi data dari *Excel* ke *list*, *array*, dan JSON, *balancing* sehingga tiap kelas sentimen terdiri dari 1.120 sampel, serta pembuatan *word vector* menggunakan *Word2Vec* untuk membentuk matriks *embedding*. *Embedding* berdimensi 25 ini kemudian dijadikan bobot awal pada lapisan *Embedding* dalam arsitektur *Sequential* Keras yang dilengkapi lapisan LSTM, *dense*, dan *dropout* untuk klasifikasi sentimen. Data selanjutnya dibagi menjadi set latih dan uji, lalu model dilatih hingga 10 *epoch*. Pada *epoch* kesepuluh diperoleh akurasi latih tertinggi 91,08% dan akurasi uji 89,73% dengan kurva akurasi latih dan uji yang stabil tanpa indikasi *overfitting* (Amrustian et al., 2022).

Dalam penelitian sebelumnya yang dilakukan Yuliana Romadhoni dan kawan-kawan, meneliti analisis sentimen terhadap PERMENDIKBUD No.30 pada media sosial twitter dengan perbandingan metode menggunakan *Long Short Term Memory* dan *Naïve Bayes*. Dataset berasal dari twitter yang di *crawling* dan mendapat data sebanyak 2265 *tweets*. Dilakukan tahap *preprocessing* mulai dari *cleaning* data, *case folding*, *tokenization* dan *stemming*. Hasil dari *preprocessing* menyisakan 471 data yang akan digunakan dalam penelitian. Penelitian ini melakukan pembobotan kata menggunakan metode *Term Frequency Inverse Document Frequency* (TF-IDF). Data dibagi menjadi 80% data uji dan 20% data latih. Hasil dari penelitian ini LSTM memiliki nilai akurasi lebih tinggi yaitu 77%, presisi dengan nilai 84%, *recall* sebesar 75%, dan *F1-Score* sebesar 80%. Sedangkan *Naïve Bayes*, mendapatkan nilai akurasi sebesar 76%, presisi, *recall* dan *F1-Score* sebesar 75% (Romadhoni & Holle, 2022).

Pada penelitian lainnya melakukan perbandingan metode untuk analisis sentimen terhadap kinerja kepolisian Indonesia menggunakan *Multinomial Naïve Bayes*, *Long Short-Term Memory* dan *Lexicon Based*. Dengan *crawling* data twitter sebanyak 14.000 data yang dibagi menjadi 80% data uji dan 20% data latih. Penelitian ini melalui tahapan *preprocessing* yaitu data *cleaning*, *case folding*, *tekonizing*, *stopwords*, *stemming* dan pembobotan dengan TF-IDF. *Lexicon Based* bertujuan untuk mengoptimalisasi proses tahapan *text mining*. Hasil penelitian ini menggunakan metode *Multinomial Naïve Bayes* mendapat nilai akurasi 78%, presisi sebesar 78.5%, *recall* dan *F1-Score* 77%. Sedangkan *Long Short-Term Memory* mendapat nilai akurasi, presisi, *recall*, dan *F1-Score* sebesar 87% (Nurpandi et al., 2024).

Penelitian yang dilakukan Safrizal Ahmad dan kawan-kawan, meneliti analisis sentiment *product tools dan home* menggunakan metode CNN dan LSTM. Metode CNN digunakan untuk sentimen hubungan jangka pendek antar kata, sedangkan metode LSTM digunakan untuk sentimen jangka panjang dalam teks. Dataset diambil dari ulasan produk Amazon kategori *product tools dan home* dengan data sebanyak 409.499 ulasan, kemudian data dibagi menjadi 80:20 yaitu 80% data uji dan 20% data *testing*. Pada penelitian ini ekstraksi fitur yang dipakai dalam CNN adalah ConV10 dan Maxpooling1D. *Epoch* yang digunakan sebagai melatih model CNN dan LSTM yaitu 20 *epoch*. Hasil dalam penelitian ini metode LSTM masih lebih unggul dibanding CNN, nilai akurasi dan *recall* metode LSTM sebesar 60.24%. Sedangkan nilai akurasi dan *recall* metode CNN sebesar 57.99% (Ahmad et al., 2023).

Penelitian-penelitian sebelumnya sebagai referensi dalam penelitian ini ditunjukkan pada Tabel 2.1.

Tabel 2.1 Penelitian Terkait

Peneliti	Judul	Metode	Hasil
(Isnain et al., 2022)	Analisis Sentimen Algoritma LSTM dan *Naïve Bayes* untuk Analisis Sentimen	*Long Short-Term Memory* dan *Naïve Bayes* dengan fitur *Word2Vec*	Metode *Long Short-Term Memory* memiliki akurasi tertinggi dibandingkan metode *Naïve Bayes.* Metode *Long Short-Term Memory* memiliki nilai akurasi, presisi,*recall* dan *f-measure* 83.33% dan *Naïve Bayes* memiliki nilai akurasi, presisi, *recall* dan *f-measure* 82%.
(Amrustian et al., 2022)	Analisis Sentimen Evaluasi Terhadap Pengajaran Dosen di Perguruan Tinggi Menggunakan Metode LSTM	*Long Short-Term Memory*	Hasil akurasi dari metode *Long Short-Term Memory* sebesar 91,08%.
(Romadhoni & Holle, 2022)	Analisis Sentimen Terhadap PERMENDIKBUD No.30 pada Media Sosial Twitter Menggunakan Metode *Naïve Bayes* dan LSTM	*Long Short-Term Memory* dan *Naïve Bayes* dengan metode pembobotan kata TF-IDF	Metode *Long Short-Term Memory* memiliki nilai akurasi sebesar 77%, presisi sebesar 84%, *recall* sebesar 75% dan *F1-Score* 80% dan *Naïve Bayes* memiliki nilai akurasi sebesar 76%, presisi, *recall* dan *F1-Score* sebesar 75%.
(Nurpandi et al., 2024)	Analisis Sentimen Terhadap Kinerja Kepolisian Indonesia Menggunakan Metode Multinomial *Naïve Bayes, Long Short-Term Memory, dan Lexicon-Based*	*Long Short-Term Memory* dan *Multinomial Naïve Bayes* dengan metode pembobotan kata TF-IDF dan proses optimalisasi *text mining Lexcon Based*	Metode *Long Short-Term Memory* memiliki nilai akurasi, presisi, *recall* dan *F1-Score* sebesar 87% dan *Multinomial Naïve Bayes* memiliki nilai akurasi sebesar 78%, presisi sebesar 78.5%, *recall* dan *F1-Score* sebesar 77%.
(Ahmad et al., 2023)	Analisis Sentimen *Product Tools* & *Home* Menggunakan Metode CNN dan LSTM	*Long Short-Term Memory* dan *Convolutional Neural Networks* dengan esktraksi fitur menggunakan ConV10 dan Maxpooling1D	Hasil dari penelitian ini *Long Short-Term Memory* menjadi nilai akurasi yang lebih unggul dengan nilai akurasi dan *recall* sebesar 60.24% dan *Convolutional Neural Networks* memiliki nilai akurasi dan *recall* sebesar 57.99%

2.2 Analisis Sentimen

Sentiment analysis adalah proses analisis teks yang bertujuan untuk mengidentifikasi dan mengklasifikasikan opini atau emosi yang terkandung dalam suatu teks, seperti ulasan produk, komentar media sosial, atau artikel berita (Fitrana et al., 2024). Dengan menggunakan teknik pemrosesan bahasa alami (NLP) dan algoritma *machine learning*, *sentiment analysis* berfungsi untuk menentukan apakah suatu teks mengandung sentimen positif, negatif, atau netral. Proses ini membantu perusahaan atau organisasi memahami bagaimana konsumen atau publik merespons produk, layanan, atau isu tertentu. Dengan hasil analisis ini, bisnis dapat mengambil langkah-langkah strategis untuk meningkatkan kualitas produk atau layanan, serta merespons keluhan atau kritik dengan lebih cepat dan efisien (Vidya Chandradev et al., 2023).

Cara kerja analisis sentimen melewati beberapa tahapan. Pertama, tahap *preprocessing* yaitu teks diproses dengan menghilangkan elemen yang tidak relevan seperti tanda baca, angka, dan kata-kata tidak penting (*stopwords*), serta mengubahnya menjadi bentuk standar seperti huruf kecil. Kemudian, fitur teks diekstraksi menjadi representasi numerik menggunakan teknik seperti *Bag of Words* (BoW), *Term Frequency Inverse Document Frequency* TF-IDF, atau *word embeddings* seperti *Word2Vec* dan *GloVe*. Selanjutnya, model *machine learning* atau *deep learning* seperti *Naive Bayes*, SVM, *Convolutional Neural Networks* (CNN), atau *Long Short-Term Memory* (LSTM) digunakan untuk mengklasifikasikan teks ke dalam sentimen positif, negatif, atau netral. Terakhir,

model dievaluasi menggunakan data uji dan dinilai dengan metrik seperti akurasi, presisi, *recall*, dan *F1-Score* untuk mengukur kinerjanya (Haddi et al., 2013).

2.3 *Term Frequency-Inverse Document Frequency* (TF-IDF)

Term Frequency-Inverse Document Frequency (TF-IDF) adalah metode yang digunakan dalam *Natural Language Processing* untuk mengevaluasi seberapa penting suatu kata dalam dokumen relatif terhadap koleksi dokumen atau korpus dan mengubah data menjadi numerik dengan memberikan bobot pada setiap kata (Afandi et al., 2024). Metode TF-IDF dikenal karena efisiensi, kemudahan, dan hasil yang akurat dalam menghitung bobot kata (term) terhadap dokumen (Riyani et al., 2019). Metode ini menggabungkan dua komponen utama yaitu *Term Frequency* (TF) untuk mengukur seberapa sering suatu kata muncul dalam dokumen dan *Inverse Document Frequency* (IDF) untuk mengukur seberapa penting suatu kata dalam keseluruhan korpus. Kata yang sering muncul di banyak dokumen akan memiliki IDF yang rendah, sedangkan kata yang jarang muncul akan memiliki IDF yang tinggi (Kosasih & Alberto, 2021). Rumus implementasi *Term Frequency-Inverse Document Frequency* (TF-IDF) ditunjukkan pada persamaan 2.1, 2.2 dan 2.3:

$$TF = \frac{Jumlah\ frekuensi\ kata\ (w)\ dalam\ satu\ dokumen\ (d)}{Jumlah\ total\ kata\ dalam\ satu\ dokumen\ (d)} \tag{2.1}$$

$$IDF = \log \frac{Jumlah\ total\ dokumen}{Jumlah\ total\ dokumen\ dengan\ kata\ (w)} \tag{2.2}$$

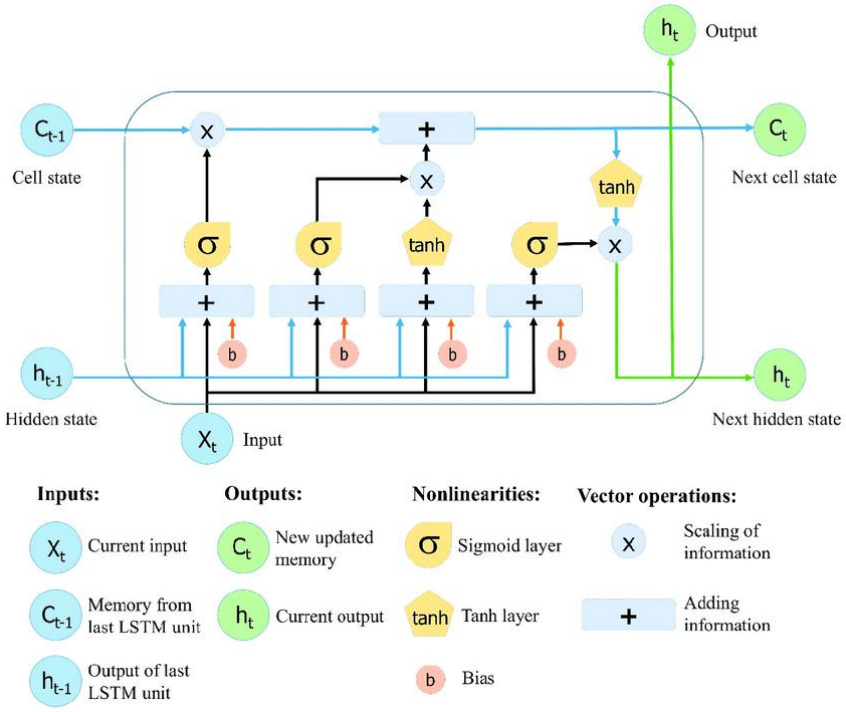
$$TF - IDF = TF\ x\ IDF \tag{2.3}$$

2.4 *Long Short-Term Memory* (LSTM)

Long Short-Term Memory (LSTM) merupakan salah satu jenis jaringan saraf tiruan dalam kategori *deep learning*, tepatnya dalam bagian *Recurrent Neural Networks* (RNN) yang khusus dirancang untuk menangani data sekuensial atau berurutan, seperti teks, audio, dan deret waktu. *Long Short-Term Memory* (LSTM) pertama kali diperkenalkan oleh Sepp Hochreiter dan Jürgen Schmidhuber pada tahun 1997 sebagai solusi atas kelemahan utama dari *Recurrent Neural Networks* (RNN) standar, yaitu *vanishing gradient* (Meriani & Rahmatulloh, 2024).

Vanishing gradient merupakan suatu masalah *Recurrent Neural Networks* (RNN) saat berusaha menyimpan informasi penting dari langkah-langkah awal dalam urutan data. Masalah ini terjadi ketika gradien, yang digunakan untuk memperbarui bobot jaringan selama pelatihan, menjadi sangat kecil seiring bertambahnya panjang urutan. Akibatnya, bobot yang seharusnya diubah untuk mempelajari pola dari urutan panjang tidak diperbarui secara efektif, yang dapat menyebabkan RNN gagal dalam mengambil *long-term dependencies* yang mengakibatkan akurasi menjadi turun dalam memprediksi (Cahyani et al., 2023).

LSTM dimodifikasi dengan menambahkan *memory cell* yang mampu menyimpan informasi dalam jangka waktu yang lama, fungsinya untuk memecahkan masalah *vanishing gradient*. Arsitektur LSTM hampir mirip dengan RNN, yang membedakan adalah proses *hidden state* Arsitektur LSTM dapat dilihat pada Gambar 2.1.



Gambar 2.1 Arsitektur LSTM. Sumber: (Le et al., 2019)

Gambar 2.1 merupakan arsitektur pada LSTM terdiri dari *memory cell*. *Cell state* dan *hidden state* diteruskan ke sel berikutnya dalam setiap *timestep*. Data ditransmisikan melalui *cell state*, yang memungkinkan data bergerak maju dengan sedikit perubahan. Tetapi dalam *cell state*, transformasi linier dapat terjadi. Dengan menggunakan gerbang *sigmoid*, data dapat ditambahkan atau dihapus dari *cell state*. Gerbang ini memiliki struktur yang menyerupai lapisan atau serangkaian operasi matriks, dan masing-masing memiliki berat yang berbeda untuk mengubah data (Le et al., 2019).

Berdasarkan Gambar 2.1 model LSTM memiliki 4 *gate* yaitu *forget gate* (f_t) berfungsi menentukan data yang akan dihapus dan digunakan untuk input, *input gate* (i_t) berfungsi memproses data yang dibawa ke titik masuk dari luar, *output gate* (o_t) berfungsi menghasilkan *output* LSTM *cell* dan memproses perhitungan

masuk (Hochreiter & Schmidhuber, 1997), dan *cell state* (C_t) . Tiap gate mengunakan nilai masukan x_t (vektor input pada timestep t) dan h_{t-1} (vektor hidden state dari *timestep* sebelumnya, yaitu t-1) dengan nilai bobot dan bias yang sudah ditentukan (Mutmatinah et al., 2024). Proses pengolahan data pada model LSTM, dimulai dari *forget gate* (f_t). Pada tahap ini, input yang diproses akan diolah dengan aturan bahwa informasi yang kurang penting atau tidak memiliki makna signifikan akan diabaikan. *Forget gate* merupakan *gate* awal yang dilalui oleh masukan menggunakan fungsi aktivasi *sigmoid*. Persamaan *forget gate* (f_t) ditunjukkan pada persamaan 2.4:

$$f_t = \sigma\,(W_f \cdot [h_{t-1} \cdot x_t] + b_f) \tag{2.4}$$

Keterangan:
a) f_t = *forget gate*
b) σ = fungsi aktivasi *sigmoid*
c) W_f = bobot *forget gate*
d) h_{t-1} = nilai *hidden state timestep* sebelumnya
e) x_t = nilai masukan
f) b_f = bias *forget gate*

Setelah itu, informasi diproses melalui *input gate* (i_t). Di sini, informasi diseleksi untuk memutuskan mana yang akan diperbarui ke dalam *cell state*, menggunakan fungsi aktivasi *sigmoid*. Pada tahap ini juga dihasilkan kandidat *cell state* baru ($C_t^{\sim}$) yang yang diperoleh menggunakan fungsi *tanh*. Persamaan *input gate* (i_t) dan *cell state* baru ($C_t^{\sim}$) ditunjukkan pada persamaan 2.5 dan 2.6:

$$i_t = \sigma\,(W_i \cdot [h_{t-1} \cdot x_t] + b_i) \tag{2.5}$$

$$C_t^{\sim} = \tanh\,(W_c \cdot [h_{t-1} \cdot x_t] + b_c) \tag{2.6}$$

Keterangan:
a) i_t = *input gate*
b) σ = fungsi aktivasi *sigmoid*
c) W_i = bobot *input gate*
d) h_{t-1} = nilai *hidden state timestep* sebelumnya

e) x_t = nilai masukan
f) b_i = bias *input gate*
g) $C_t^{\sim}$ = kandidat *cell state* baru
h) tanh = fungsi *tanh*
i) W_c = bobot
j) b_c = bias

Selanjutnya, mencari nilai pada *cell state* (C_t) dengan nilai yang sebelumnya sudah dihitung. Persamaan ditunjukkan pada persamaan 2.7:

$$C_t = f_t \,.\, C_{t-1} + i_t \,.\, C_t^{\sim} \tag{2.7}$$

Keterangan:
a) C_t = *cell state*
b) f_t = *forget gate*
c) C_{t-1} = nilai *cell state* sebelumnya
d) i_t = *input gate*
e) $C_t^{\sim}$ = kandidat *cell state* baru

Terakhir, *output gate* menggunakan fungsi aktivasi *sigmoid* akan menghasilkan nilai *output gate* (o_t) dan yang nanti akan digunakan untuk mendapat nilai *hidden state* (h_t) yang baru dengan mengalikan *output gate* (o_t) dan *cell state* (C_t) yang telah diproses dengan fungsi *tanh* (Widayat, 2021). Persamaan *output gate* (o_t) dan *hidden state* (h_t) ditunjukkan pada persamaan 2.8 dan 2.9:

$$o_t = \sigma\,(W_o \cdot [h_{t-1} \cdot x_t] + b_o) \tag{2.8}$$

$$h_t = o_t \cdot \tanh(C_t) \tag{2.9}$$

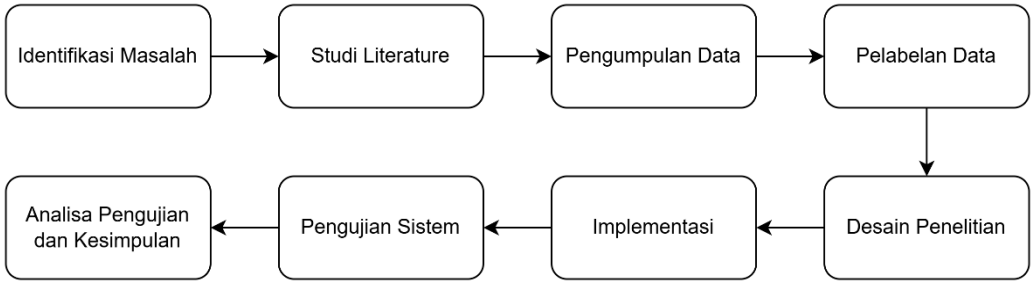
Keterangan:
a) o_t = *output gate*
b) σ = fungsi aktivasi *sigmoid*
c) W_o = bobot *output gate*
d) h_{t-1} = nilai *hidden state timestep* sebelumnya
e) x_t = nilai masukan
f) b_o = bias *output gate*
g) h_t = *hidden state* baru
h) tanh = fungsi *tanh*
i) C_t = *cell state*

BAB III

METODE PENELITIAN

3.1 Prosedur Penelitian

Rangkaian atau tahapan sistematis yang dilakukan oleh peneliti dalam Menyusun dan melaksanakan penelitian disebut sebagai prosedur penelitian. Prosedur penelitian mencakup proses mulai dari perencanaan, pengumpulan data, analisis data, hingga hasi penelitian. Gambar 3.1 merupakan prosedur atau tahapan yang akan dilakukan pada penelitian ini, agar penelitian dapat dilakukan secara terstruktur.



Gambar 3.1 Prosedur Penelitian

Prosedur penelitian ini dimulai dengan identifikasi masalah, dimana dilakukan perumusan latar belakang permasalahan sebagai alasan penelitian ini dilakukan. Pada tahap ini juga terdapat rumusan masalah serta batasan masalah agar penelitian lebih fokus dan terarah. Setelah identifikasi masalah, tahap selanjutnya studi literatur. Tahap ini dilakukan pencarian referensi yang relevan dengan topik penelitian, bertujuan untuk memperkaya wawasan dan menemukan celah penelitian yang dapat dikembangkan lebih lanjut. Selanjutnya terdapat tahap

pengumpulan data, data berupa ulasan, *rating*, *username* dan nama cabang rumah makan padang terhadap rumah makan Padang Payakumbuah melalui aplikasi *Google Maps*. Tahapan berikutnya yaitu pelabelan data, data yang sudah dikumpulkan diberi label atau diklasifikasi dengan kategori positif dan negatif. Selanjutnya terdapat tahap desain sistem, untuk tahapan implementasi sistem secara menyeluruh. Tahap impelentasi dan pengujian sistem dilakukan untuk mengimplementasi desain sistem yang sudah dibuat. Yang kemudian sistem tersebut diuji untuk mengetahui performanya. Tahap terakhir yaitu kesimpulan dan analisis pengujian, di mana hasil pengujian dievaluasi untuk melihat ke efektivitasan sistem dan melakukan perhitungan nilai akurasi yang telah dibuat oleh sistem.

3.2 Pengumpulan Data

Proses mengumpulkan informasi dari berbagai sumber yang relevan disebut sebagai pengumpulan data. Data dibagi menjadi dua, terdapat data primer dan data sekunder. Data primer adalah data yang diperoleh secara langsung dari sumber aslinya melalui metode seperti wawancara, observasi, eksperimen, survei, atau kuesioner. Sedangkan data sekunder adalah data yang sudah tersedia dan diperoleh dari sumber-sumber lain, seperti buku, jurnal, laporan penelitian sebelumnya, atau basis data public (Gunawan et al., 2023). Data yang digunakan dalam penelitian ini data primer, data diperoleh dengan cara *crawling* pada *platform Google Maps* menggunakan bahasa pemrograman *pyhton*. *Crawling* data menggunakan *serpApi* untuk mengambil data dari *Google Maps*. *SerpAPI* merupakan layanan API yang memungkinkan digunakan para pengembang untuk melakukan web *scraping* atau

crawling dari berbagai mesin pencarian seperti Google, Bing, Yahoo, dan lainnya (Ma'ady et al., 2023). Dalam tahapan ini peneliti melakukan *crawling* ulasan terhadap rumah makan Padang Payakumbuah yang terletak di daerah Jakarta dan sekitarnya. Didapatkan 3 cabang rumah makan Padang Payakumbuah yaitu di daerah Cikini Menteng, Tebet, dan Kemang, dengan jumlah data yang diambil sebanyak 976 ulasan.

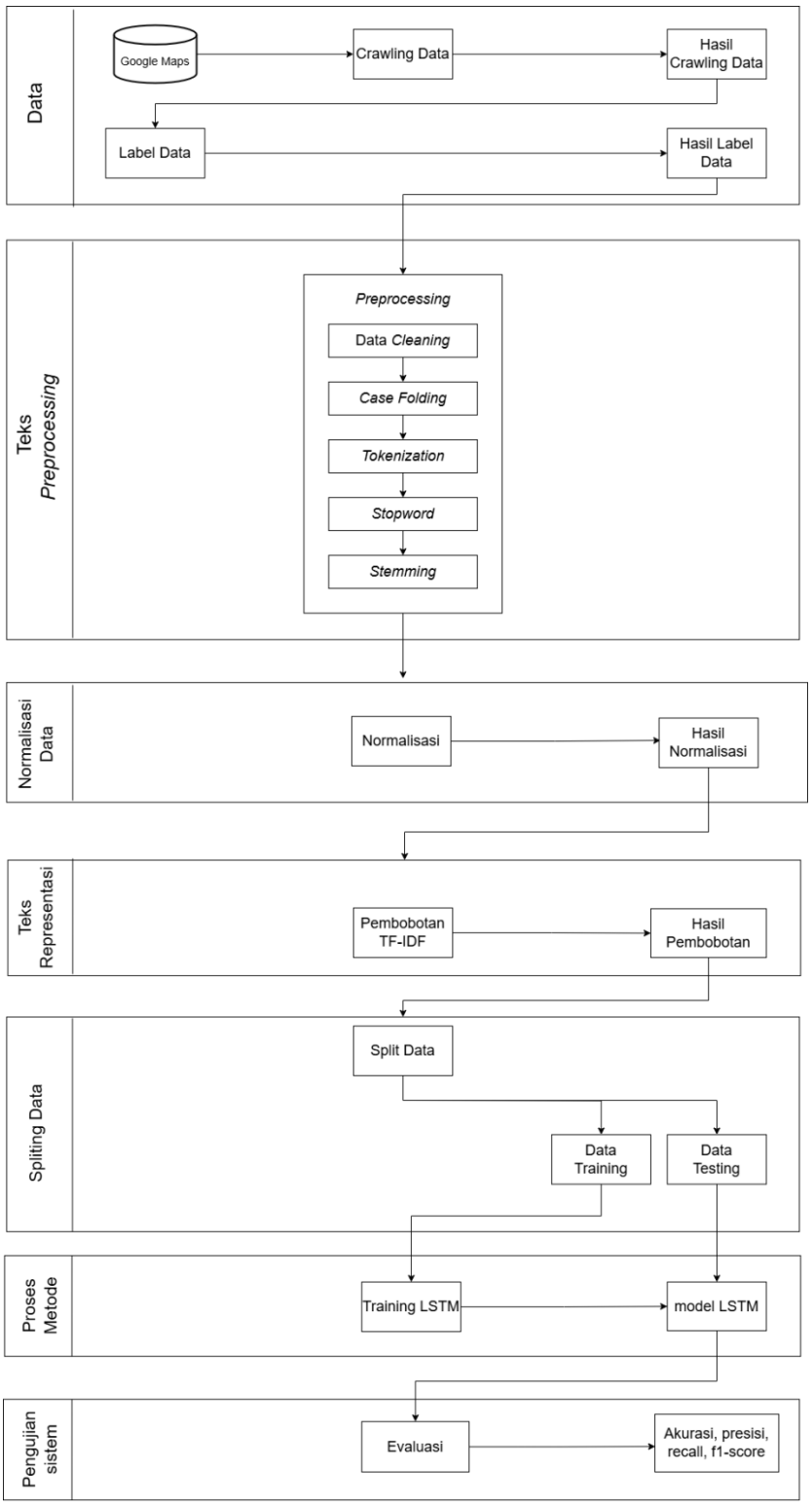
3.3 Pelabelan Data

Data yang sudah di *crawling* pada tahap ini akan diberi label secara manual menjadi dua kategori yaitu positif dan negatif. Label kemudian akan divalidasi oleh validator. Ahli Bahasa yang menjadi Validator untuk memvalidasi pelabelan pada penelitian ini yaitu oleh seorang guru Bahasa Indonesia Sekolah Menengah Atas Islam Kepanjen atas nama Dra. Hj. Maftukhah. Kemudian *dataset* akan disimpan dalam format *Excel* (xlsx). Contoh data ulasan yang sudah diberi label ditunjukkan pada Tabel 3.1 yang akan digunakan untuk sampel data pada tahap selanjutnya.

Tabel 3.1 Sampel Pelabelan Data

Dokumen	Ulasan	Kelas
D1	Makanannya enk tempat nya nyaman harga terjangkau.. ❤️‍🩹	Positif
D2	rasa kurang enak..nasi pera telor balado asem	Negatif
D3	pelayanan nya perfect. suasana nya nyaman. rasa makanan nya sangat enak. minuman nya juga segar	Positif

3.4 Desain Penelitian



Gambar 3.2 Desain Penelitian

Desain penelitian ditunjukkan pada Gambar 3.2 merupakan rencana atau kerangka kerja yang dibuat untuk membantu peneliti mengatur langkah-langkah dalam menjalankan penelitian. Desain ini digunakan agar penelitian berjalan dengan sistematis dan sesuai tujuan (Sugiyono, 2013). Dalam penelitian, desain ini mencakup bagaimana data dikumpulkan, diolah, dianalisis, dan bagaimana hasil akhirnya disimpulkan.

Pada penelitian ini, proses dimulai dengan mengumpulkan data (*crawling*) dari *Google Maps*, lalu data tersebut di *preprocessing* melalui tahapan seperti pembersihan data (data *cleaning*), mengubah huruf menjadi kecil (*case folding*), memisahkan kata-kata (*tokenization*), hingga menghapus kata-kata yang tidak relevan (*stopword*). Setelah *preprocessing*, dilakukan normalisasi data, dengan merubah kata singkatan menjadi kata sebenarnya. Kemudian melakukan pembobotan data menggunakan metode TF-IDF agar data teks dapat dipahami oleh komputer dalam bentuk angka.

Data yang telah siap dibagi menjadi dua bagian, yaitu untuk melatih model (*training* data) dan menguji model (*testing* data) dengan perbandingan 80 : 20. Peneliti menggunakan metode *Long Short-Term Memory* (LSTM) untuk menganalisis data, yang kemudian dievaluasi hasilnya menggunakan metrik seperti akurasi, presisi, *recall* dan *f1-score*.

3.5 *Preprocessing*

Preprocessing merupakan metode yang dapat meningkatkan hasil sentimen (Haddi et al., 2013). Tahap *preprocessing* memastikan data berada dalam kondisi bersih dan terstruktur. *Preprocessing* sering dilakukan terhadap data mentah yang

tidak berguna yang masih berisi *noise*, *missing value*, atau ketidaksesuaian lain yang dapat mempengaruhi hasil analisis (Qaisar, 2020). Pada penelitian ini dilakukan beberapa tahapan *preprocessing* yaitu data *cleaning*, *case folding*, *tokenization*, *stopword removal*, dan *stemming*.

3.5.1 Data Cleaning

Tahap *preprocessing* pertama yang dilakukan yaitu cleaning, merupakan tahapan untuk membersihkan teks dari elemen yang tidak efisien dan inkonsistensi yang dapat mengganggu hasil analisis. Pada pembersihan data ini menggunakan *regular expression* (regex) melalui *library Python* re. Regex merupakan pola pencarian yang memungkinkan untuk melakukan pencarian, penggantian, atau pemrosesan teks secara efisien. Dengan regex dapat membersihkan teks dari karakter yang tidak berguna dan karakter yang berulang. Proses membersihkan teks dari elemen yang tidak relevan seperti angka, tanda baca seperti (()*&^%$#@!~<>?:;/{}[]_-+\|.), spasi ganda, atau karakter berulang. Selanjutnya setelah data dibersihkan, pada penelitian ini akan melakukan representasi emoji. Emoji yang muncul dalam teks diubah menjadi kata-kata yang mudah diproses oleh model analisis teks. Representasi emoji yang akan dilakukan yaitu pada *Unicode* emoji contoh emoji "😊", "❤️" dan sebagainya. Penulis akan melakukan representasi emoji dengan teknik substistusi menggunakan library re yaitu dengan memanipulasi teks sesuai kebutuhan, menggunakan fungsi sub() untuk menggantikan suatu pola dengan string lain. Referensi emoji *Unicode* diambil dari situs website https://unicode.org/emoji/charts-14.0/full-emoji-list.html. Data yang diambil hanya emoji *smiley & emotion* dan *body*.

Tabel 3.2 Proses Data *Cleaning*

Sebelum Data *Cleaning*	**Setelah Data *Cleaning***
Makanannya enk tempat nya nyaman harga terjangkau.. 💔	Makanannya enk tempat nya nyaman harga terjangkau cinta

3.5.2 *Case Folding*

Mengubah semua huruf dalam teks menjadi huruf kecil (*lowercase*) mulai dari awal kalimat sampai akhir dinamakan sebagai *case folding*. *Case Folding* dilakukan untuk menghindari perbedaan yang disebabkan oleh huruf besar dan kecil, untuk memastikan bahwa kata-kata yang sama dikenali sebagai entitas yang sama.

Tabel 3.3 Proses *Case Folding*

Sebelum *Case Folding*	**Setelah *Case Folding***
Makanannya enk tempat nya nyaman harga terjangkau.. 💔	makanannya enak tempat nya nyaman harga terjangkau cinta

3.5.3 *Tokenization*

Tokenization memecah sebuah kalimat menjadi kata (token). Yang dijadikan acuan pemecahan kata pada tahap ini adalah tanda baca dan spasi. Proses ini berguna untuk metode agar dapat memahami teks yang digunakan.

Tabel 3.4 Proses *Tokenization*

Sebelum *Tokenization*	**Setelah *Tokenization***
Makanannya enk tempat nya nyaman harga terjangkau.. 💔	'makanannya', 'enak', 'tempat', 'nya', 'nyaman', 'harga', 'terjangkau', ‘cinta’

3.5.4 *Stopword*

Stopword menghapus kata-kata yang tidak memiliki makna atau nilai untuk analisis biasanya tanda hubung, kata ganti, atau presposisi. *Stopword* bervariasi mengikut Bahasa yang digunakan, contoh *stopword* dalam Bahasa Indonesia "yang", "dan", "dari", "di", "ke", "adalah", "ini", "itu", "sebagai", "atau", dalam

Bahasa Inggris "*the*", "*is*", "*in*", "*at*", "*which*", "*on*", "*and*", "*but*", "*or*", "*he*", "*she*", "*they*".

Tabel 3.5 Proses *Stopword*

Sebelum *Stopword*	**Setelah *Stopword***
Makanannya enk tempat nya nyaman harga terjangkau.. 💔	'makanannya', 'enak', 'tempat', 'nyaman', 'harga', 'terjangkau', ‘cinta’

3.5.5 *Stemming*

Mengubah kata menjadi bentuk dasar dengan menghapus imbuhan. Imbuhan yang ditambahkan di depan kata dasar (*prefix*), imbuhan yang ditambahkan diakhir kata (*suffix*), imbuhan yang disisipkan di tengah kata (*infix*), dan imbuhan gabungan yang berada di awal dan akhir yang membentuk satu kesatuan makna (konfiks).

Tabel 3.6 Proses *Stemming*

Sebelum *Stemming*	**Setelah *Stemming***
Makanannya enak tempat nya nyaman harga terjangkau.. 💔	'makan', 'enak', 'tempat', 'nyaman', 'harga', 'jangkau', ‘cinta’

3.6 Normalisasi Data

Pada umumnya, teks yang diambil dari ulasan sering kali mengandung singkatan-singkatan atau kata-kata yang ditulis tidak lengkap. Contohnya, "enk" yang arti sebenarnya adalah enak. Pada tahapan normalisasi kata ini akan mengubah kata singkatan menjadi kata yang sebenarnya, seperti pada contoh sebelumnya kemudian akan diubah menjadi "enak" dengan menggunakan kamus kata yang sudah disiapkan. Kamus berisi daftar singkatan dan padanan kata yang lengkap, yang digunakan untuk menggantikan bentuk singkat dengan kata yang benar. Referensi kata yang benar berdasarkan KBBI https://www.kbbi.web.id/.

Tabel 3.7 Proses Normalisasi Kata

Sebelum Data Cleaning	Setelah Data Cleaning
Makanannya enk tempat nya nyaman harga terjangkau.. 💔	Makanannya enak tempat nya nyaman harga terjangkau cinta

3.7 *Term Frequency–Inverse Document Frequency* (TF-IDF)

Setelah melewati tahap *preproccesing*, selanjutnya data akan melalui proses pembobotan kata menggunakan *Term Frequency–Inverse Document Frequency* (TF-IDF), sebelum masuk ke proses klasifikasi sentimen dengan metode *Long short-term memory*. Untuk menghitung bobot dengan TF-IDF, dapat dihitung menggunakan persamaan 2.1 hingga 2.3. Menggunakan 3 data sebagai sampel yang ditunjukkan pada Tabel 3.1.

Tahapan pertama yang dilakukan yaitu perhitungan TF, nilai TF didapatkan dari hasil pembagian jumlah kata yang mucul dalam dokumen dengan total kata yang ada dalam dokumen. Contoh perhitungan TF ditunjukkan pada Tabel 3.8.

Tabel 3.8 Perhitungan TF

Dokumen	Kata	Frekuensi	TF
D1	makan	1	1/7 = 0.1428
D2		0	0/8 = 0
D3		1	1/10 = 0.1

Setelah mendapat nilai *Term Frequency*, selanjutnya dilakukan perhitungan IDF. Nilai IDF didapatkan dari hasil log jumlah dokumen dibagi dengan jumlah dokumen yang memiliki kata “x”. Contoh perhitungan IDF, ditunjukkan pada Tabel 3.9.

Tabel 3.9 Perhitungan IDF

Dokumen	Kata	Frekuensi	IDF
D1	makan	1	log3/2 = 0.1761
D2		0	
D3		1	

Setelah mendapatkan nilai TF dan IDF, dilanjutkan dengan perhitungan TF-IDF dengan mengalikan nilai TF dan nilai IDF. Contoh perhitungan TF-IDF ditunjukkan pada Tabel 3.10.

Tabel 3.10 Perhitungan TF-IDF

Kata	Dokumen	TF-IDF
makan	D1	0.1428 x 0.1761 = 0.0251
	D2	0 × 0.1761 = 0
	D3	0.1 × 0.1761 = 0.0176

Hasil perhitungan dari TF-ID masing masing kata atau *term*, ditunjukkan oleh Tabel 3.11.

Tabel 3.11 Hasil Perhitungan TF-IDF

Token	TF-IDF		
	D1	D2	D3
makan	0.0251	0	0.0176
enak	0	0	0
tempat	0.0681	0	0
nyaman	0.0251	0	0.0176
harga	0.0681	0	0
jangkau	0.0681	0	0
cinta	0.0681	0	0
rasa	0	0.0220	0.0176
kurang	0	0.0851	0
nasi	0	0.0851	0
pera	0	0.0851	0
telor	0	0.0851	0
balado	0	0.0851	0
asem	0	0.0851	0
layan	0	0	0.0702
perfect	0	0	0.0702
suasana	0	0	0.0702
sangat	0	0	0.0702
minum	0	0	0.0702
segar	0	0	0.0702

Tabel 3.11 merupakan hasil representasi kata dengan TF-IDF yang kemudian vector yang dihasilkan dapat digunakan untuk ke proses analisis sentimen dengan metode LSTM. Nilai bobot terendah dihasilkan oleh kata “makan”, “nyaman”, dan “rasa” yang muncul pada dokumen 3 dengan nilai 0.0176.

3.8 Pembagian Data

Pembagian daata dalam penelitian ini yaitu membagi data menjadi dua yaitu, data *training* dan data *testing*. Data *training* yang akan digunakan untuk melatih model LSTM untuk memahami pola dalam data dan membuat prediksi yang akurat. Kemudian data *testing* untuk menguji kinerja model yang sudah dilatih pada data baru yang sebelumnya tidak pernal dilihat. Menggunakan data *testing* dapat melihat seberapa baik model LSTM mampu melakukan analisis sentimen terhadap ulasan rumah makan pada payakumbuah. Dengan pembagian data yang tepat, peneliti dapat memastikan bahwa model LSTM yang dikembangkan memiliki performa yang stabil dan mampu bekerja dengan baik saat diterapkan pada data baru yang belum pernah digunakan sebelumnya. Berdasarkan penelitian sebelumnya peneliti membandingkan rasio untuk mendapatkan hasil performa terbaik dengan membandingkan 60% data *training* dan 40% data *testing*, 70% data *training* dan 30% data *testing*, 80% data *training* dan 20% data *testing*, serta 90% data *training* dan 10% data *testing* (Anam et al., 2025).

3.9 Alur Perhitungan *Long Short-Term Memory*

Setelah dilakukan pembobotan kata dengan menggunakan TF-IDF, selanjutnya melakukan proses perhitungan menggunakan metode *Long Short Term Memory*. Input yang akan dilakukan dalam perhitungan ini yaitu kalimat “makan enak tempat nyaman harga terjangkau”. Perhitungan dimulai dari menghitung *forget gate* pada persamaan 2.4 menggunakan *input* X_1 = 0.0294 dengan inisialisasi bobot (W) dan bias (b) yaitu W_f = 0.4 dan $b_f = 0.2$. Pada perhitungan ini inisialisasi

hidden state sebelumnya h_{t-1} = 0 dan *cell state* sebelumnya C_{t-1} = 0 maka didapatkan hasil 0.55274.

Forget gate :

f_t = σ (0.4 · [0, 0.0294] + 0.2)

f_t = σ ((0.4 · 0.0294) + 0.2)

f_t = σ (0.01176 + 0.2)

f_t = σ (0.21176)

f_t = 0.55274

Selanjutnya menghitung nilai *input gate* pada persamaan 2.5 menggunakan W_i = 0.3 dan b_i = 0.01, didapatkan hasil dengan nilai 0.52718.

Input gate :

i_t = σ (0.3 · [0, 0.0294] + 0.1)

i_t = σ ((0.3 · 0.0294) + 0.1)

i_t = σ (0.00882 + 0.1)

i_t = σ (0.10082)

i_t = 0.52718

Setelah melakukan perhitungan *input gate*, selanjutnya menghitung kandidat c*ell state*. Perhitungan kandidat *cell state* ditunjukkan pada persamaan 2.6 dengan menggunakan nilai W_c = 0.5 dan b_c = 0.05. Mendapat nilai sebesar 0.06461.

Kandidat *cell state* :

$C_t^{\sim}$ = tanh (0.5 · [0, 0.0294] + 0.05)

$C_t^{\sim}$ = tanh ((0.5 · 0.0294) + 0.05)

$C_t^{\sim}$ = tanh (0.0147 + 0.05)

$C_t^{\sim}$ = tanh (0.0647)

$C_t^{\sim}$ = 0.06461

Tahap selanjutnya yaitu perhitungan *cell state* yang ditunjukkan pada persamaan 2.7 menggunakan nilai *forget gate, input gate* dan kandidat *cell state* yang sudah dihitung sebelumnya.

Cell state :

C_t = (0.55274 · 0) + (0.52718 · 0.06461)

C_t = 0.03406

Pada langkah ini untuk menghitung nilai *output gate* ditunjukkan pada persamaan 2.8 dengan menggunakan W_o = 0.6 dan b_o = 0.15. Mendapatkan hasil sebesar 0.54181.

Output gate :

o_t = σ (0.6 · [0, 0.0294] + 0.15)

o_t = σ ((0.6· 0.0294) + 0.15)

o_t = σ (0.01764 + 0.15)

o_t = σ (0.16764)

o_t = 0.54181

Lalu yang terakhir adalah menghitung *hidden state* dengan hasil *output gate* yang sudah dihitung dikalikan dengan hasil *cell state.* Perhitungan *hidden state* ditunjukkan pada persamaan 2.9, dengan menghasilkan nilai sebesar 0.01845.

Hidden state :

h_1 = 0.54181 · tanh(0.03406)

h_1 = 0.01845

Sudah didapatkan hasil nilai *hidden state* (h_1) dengan nilai 0.01845 setelah menggunakan nilai X_1 = 0.0294 sebagai *input* proses. Perhitungan ini diulangin sampai *input* X_6, pada Tabel 3.12 merupakan hasil dari *forget gate*, *input gate*, kandidat *cell state*, *cell state*, *output gate*, dan *hidden state.*

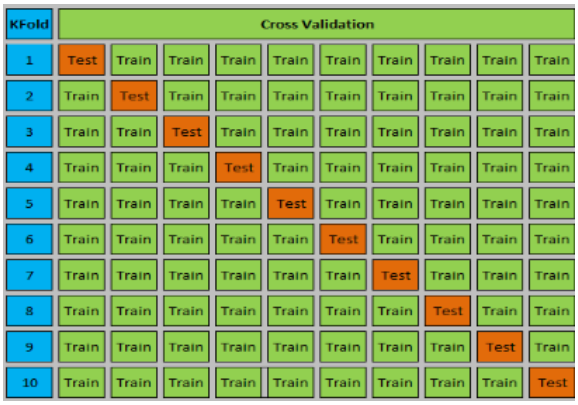
Tabel 3.12 Hasil Perhitungan LSTM

Input	**f**	**i**	**C'**	**C**	**o**	**H**
0.0294	0.5527	0.5272	0.0646	0.0341	0.5418	0.0184
0.0000	0.5498	0.5250	0.0500	0.0450	0.5374	0.0241
0.0795	0.5577	0.5309	0.0895	0.0726	0.5493	0.0398
0.0294	0.5527	0.5272	0.0646	0.0742	0.5418	0.0401
0.0795	0.5577	0.5309	0.0895	0.0889	0.5493	0.0487
0.0795	0.5577	0.5309	0.0895	0.1015	0.5493	0.0556

3.10 Skenario Pengujian

Uji coba yang dilakukan pada penelitian ini untuk melatih model data *training* dengan algoritma LSTM, kemudian mengujinya dengan data *testing*. Pada skenario pengujian pertama yaitu dengan membagi data ke dalam berbagai rasio. Rasio ini mencakup 60:40 (60% data *training* dan 40% data *testing*), 70:30, 80:20, dan 90:10. Dengan bertujuan untuk menentukan hasil performa terbaik. Kemudian pada pengujian kedua menggunakan *K-fold cross validation* untuk menilai lebih lanjut akurasi model. *K-Fold Cross Validation* merupakan teknik evaluasi model yang sering digunakan untuk menilai kinerja model. Metode ini membagi data pelatihan menjadi K bagian yang memiliki ukuran sama, disebut "*fold*" (Dewi & Putra, 2024). Proses ini dilakukan sebanyak K kali, dengan bergantian antara data *training* dan dana *testing* setiap iterasinya (Irawan et al., 2024). Dalam penelitian ini pengujian dilakukan dengan menggunakan K=10 ke dalam *split* data untuk menganalisis bagaimana pengaruh jumlah lipatan terhadap akurasi yang diperoleh.

K-fold cross validation pada Gambar 3.3 merupakan teknik untuk menilai akurasi dari model yang akan dibangun. Kelebihan menggunakan *k-fold cross validation* yaitu memberikan estimasi kinerja yang lebih konsisten dan tepat dengan cara mengkalkulasi rata-rata hasil dari "k" putaran, serta menghasilkan risiko empiris yang lebih rendah, dibandingkan hanya membagi data menjadi data pelatihan dan data pengujian (Wijiyanto et al., 2024). *Dataset* yang digunakan akan dibagi menggunakan rasio pembagian data yang berbeda, yaitu 60:40, 70:30, 80:20, dan 90:10, dengan masing-masing rasio menggunakan metode 10-fold *cross validation*. Pada setiap iterasi, data *testing* akan divalidasi silang menggunakan 10 *fold*, dengan sisa data digunakan untuk pelatihan. Pembagian ini dilakukan dengan tujuan untuk mengevaluasi model secara lebih akurat dengan pengulangan sebanyak 10 kali pada setiap rasio pembagian data.



Gambar 3.3 Skema *K-fold Cross Validation*. Sumber: (Wijiyanto et al., 2024)

Dengan metode *k-fold cross validation* setiap subset data akan secara bergantian digunakan sebagian data *training* dan data *testing*. Untuk memastikan bahwa setiap *point* dilatih dan diuji, guna memberi hasil akurasi yang baik. Hal ini dapat menjadi evaluasi menyeluruh terhadap performa model dan membantu dalam

menentukan pengaturan optimal untuk mencapat akurasi yang terbaik dalam analisis sentimen ulasan rumah makan padang Payakumbuah.

3.11 Evaluasi Performa

Tahap terakhir dari proses ini adalah menghitung metrik kinerja untuk menghitung kinerja klasifikasi, kita menggunakan *confusion matrix***.** *Confusion matrix* berbentuk tabel untuk mengevaluasi kinerja suatu model klasifikasi dengan membandingkan kelas prediksi terhadap label asli. *Confusion matrix* dapat dilihat pada Tabel 3.13 akan digunakan menghitung seperti akurasi, *precision*, *recall* dan *f1-score* untuk mengevaluasi seberapa baik model mampu mengklasifikasi sentimen pada data *testing*.

Tabel 3.13 Tabel *Confusion Matrix*

Actual Class	***Predicted Class***	
	Negatif	**Positif**
Negatif	*True Negative* (TN)	*False Positive* (FP)
Positif	*False Negative* (FN)	*True Positive* (TP)

1. *True Negative* (TN): Data negatif di kelas sebenarnya di prediksi negatif di kelas prediksi.
2. *False Positive* (FP): Data negatif di kelas sebenarnya di prediksi positif di kelas prediksi.
3. *False Negative* (FN): Data positif di kelas sebenarnya di prediksi negatif di kelas prediksi.
4. *True Positive* (TP): Data positif di kelas sebenarnya di prediksi positif di kelas prediksi.

Akurasi menunjukkan seberapa tepat model dalam mengklasifikasikan sentimen secara keseluruhan. Sementara itu, presisi menggambarkan proporsi data

yang diprediksi sebagai positif yang benar-benar bersentimen positif. *Recall* menunjukkan sejauh mana model mampu mengenali data yang memang positif dan *f1-score* merupakan metrik yang menggabungkan presisi dan *recall*, memberikan gambaran yang lebih lengkap mengenai kinerja model. Evaluasi yang menyeluruh ini membantu peneliti memahami kelebihan dan kekurangan model, serta mengidentifikasi aspek yang perlu diperbaiki atau ditingkatkan pada iterasi berikutnya. Persamaan seperti berikut (Romadhoni & Holle, 2022).

Positive prediction value merupakan sampel milik kelas positif yang diprediksi benar-benar positif. Sedangkan *negative prediction value* adalah sampel milik kelas negatif yang diprediksi benar-benar negatif. *True positive rate* adalah sampel milik kelas positif yang diklasifikasi dengan benar. *True negative rate* adalah sampel milik kelas negatif yang diklasifikasi dengan benar (Tharwat, 2021).

$$\text{Akurasi} = \frac{TP + TN}{TP + TN + FP + FN} \times 100\% \tag{3.7}$$

$$\text{Presisi} = \frac{TP}{TP + FP} \times 100\% \tag{3.8}$$

$$\text{Recall} = \frac{TP}{TP + FN} \times 100\% \tag{3.9}$$

$$\text{F1} - \text{Score} = 2 \times \frac{Presisi \times Recall}{Presisi + Recall} \times 100\% \tag{3.10}$$

$$Positive\ Prediction\ Value = \frac{TP}{FP + TP} \times 100\% \tag{3.11}$$

$$Negative\ Prediction\ Value = \frac{TN}{FN + TN} \times 100\% \tag{3.12}$$

$$True\ Positive\ Rate = \frac{TP}{TP + FN} \times 100\% \tag{3.13}$$

$$True\ Negative\ Rate = \frac{TN}{TN + FP} \times 100\% \tag{3.14}$$

BAB IV

HASIL DAN PEMBAHASAN

Bab ini menyajikan hasil penelitian dan analisis yang dilakukan selama penelitian analisis sentimen menggunakan *Long Short-Term Memory* terhadap ulasan rumah makan padang payakumbuah oleh peneliti.

4.1 Implementasi Sistem

Penelitian ini dilakukan dengan menggunakan bahasa pemrograman *Python* dan berbagai *library* yang mendukung proses penelitian yang dilakukan. Beberapa library yang digunakan antara lain spaCy untuk pemrosesan teks dan TensorFlow untuk penerapan metode LSTM. Pengujian akan diimplementasikan menggunakan tabel *Confusion Matrix*, yang kemudian akan dihitung nilai akurasi, presisi, *recall* dan *f1-score* dalam analisis sentimen ulasan rumah makan padang. Sistem ini mengandalkan data yang diperoleh melalui proses *crawling* ulasan dari platform *Google Maps* Rumah Makan Padang Payakumbuah yang ada di daerah Jakarta. Data yang terkumpul berjumlah 976 ulasan, yang kemudian diberi label positif dan negatif secara manual. Proses ini dilakukan dengan menggunakan API *Google Maps* yang diakses melalui layanan SerpAPI.

4.1.1 Data

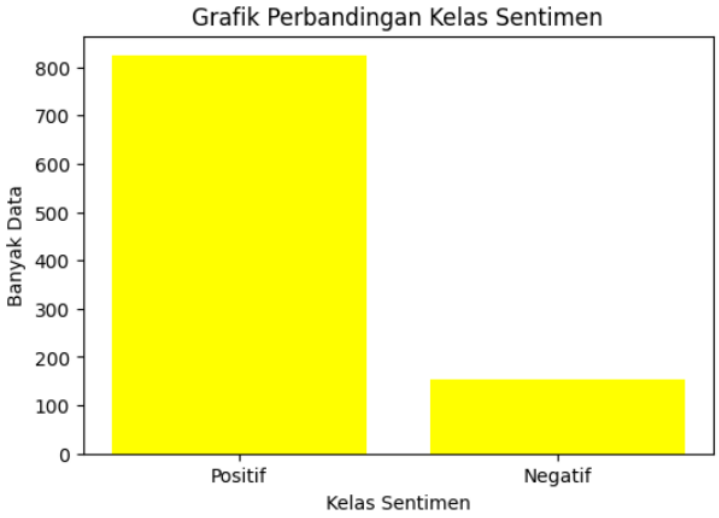
Data ulasan yang digunakan dalam penelitian ini diperoleh melalui proses *crawling* menggunakan *Google Maps* API yang dilakukan dengan bantuan SerpAPI. Dengan menggunakan kata kunci “Rumah Makan Padang Payakumbuah

Jakarta", sebanyak 976 ulasan berhasil dikumpulkan dari beberapa cabang Rumah Makan Padang Payakumbuah yang ada di Jakarta. Ulasan-ulasan ini kemudian diberikan label sentimen secara manual dan di validasi oleh ibu Dra. Hj. Maftukhah sebagai seorang guru Bahasa Indonesia Sekolah Menengah Atas Islam Kepanjen. Dengan kriterian sentimen telah ditentukan menjadi 2 yaitu positif dan negatif. Tabel 4.1 menampilkan contoh *dataset* yang telah diperoleh untuk penelitian.

Tabel 4.1 Hasil *Crawling* Data Ulasan dan Label

Ulasan	Label
First time ke payakumbuah, service nya mantap banget. Dapet free kopi milo. dendeng batokok lama banak 👌😍	Positif
makanannya enak bgt apalagi ikan salainya. Mantab 👍👍	Positif
Telor barendo dan ikan salai the best.. nasinya beda dari rm padang lain	Positif
HARUS TAMBAH ATAU GANTI MANPOWER, PESAN AYAM POP TAPI ENGGA DATENG2, UDAH 3 KALI DIINGETIN TAPI AYAM TETAP GA DATANG, CUMA IYA2 AJA, AKHIRNYA CUMA MAKAN TELUR DADAR DAN KERUPUK KULIT PAKAI NASI, REALLY HATE THE SERVICES.	Negatif
Tempat nyaman dan makanannya enak, pas banget buat makan ramean. Service staffnya jg ramah sekali, mas Alfarabi.	Positif
1. Ayam goreng, 28rb (B aja) 2. Rendang, 35rb (Enak, bumbunya wangi banget) 3. Dendeng batokok lado hijau, 35rb (Enak, cabe nya wangi krn pakai bawang merah) 4. Sayur kapau, 17rb 5. Sambal hijau, 10rb (kurang, tipe cabe yg sdh halus dan rada encer) 6. Nasi putih, 12.500 7. Teh tawar, 10rb 8. Es teh tawar, 12rb 9. Pelayanan bagus. 10. Enjoy 👌	Positif

Dataset tersebut dapat diilustrasikan melalui grafik yang membandingkan antara data positif dan negatif, terlihat pada Gambar 4.1. Digambarkan kelas sentimen positif terdapat 823 data dan negatif terdapat 153 data.



Gambar 4.1 Grafik Perbandingan Kelas Sentimen

4.2 *Preprocessing* Data

Tahap *preprocessing* data bertujuan untuk membersihkan teks dari segala jenis noise, terdapat beberapa teknik *preprocessing* dilakukan untuk memastikan bahwa data teks bersih, konsisten, dan siap diproses lebih lanjut. Proses *preprocessing* yang dilakukan mencakup data *cleaning*, representasi emoji, *case folding*, *tokenization*, *stopword removal*, *stemming*.

4.2.1 Data Cleaning

Pada tahap ini, data dibersihkan dari karakter yang tidak diperlukan seperti tanda baca, spasi berlebih, dan pengulangan huruf untuk memastikan teks lebih terstruktur dan mudah dianalisis. Proses ini menggunakan *library Regular Expressions* (re) untuk melakukan berbagai operasi pembersihan. *Method* re.sub(r'[^a-zA-Z0-9\s]', '', text) digunakan untuk menghapus semua karakter selain huruf dan angka, sedangkan *method* re.sub(r'\s+', ' ', text).strip() berfungsi untuk menghilangkan spasi berlebih yang dapat mengganggu analisis teks. Selain itu,

terdapat fungsi remove_repeated_letters(text), yang menggunakan *method* ekspresi reguler re.sub(r'(.)\1+', r'\1', text) untuk menghapus pengulangan huruf berlebihan, misalnya kata "baaaaagus" akan dikonversi menjadi "bagus". Selain pembersihan teks, tahap ini juga mencakup konversi emoji menjadi representasi kata agar dapat diproses oleh model menggunakan *dictionary* emoji_dict. Misalnya, emoji "👍" akan digantikan dengan kata "bagus" agar lebih mudah dipahami oleh model analisis teks. Dengan langkah-langkah ini, data menjadi lebih bersih dan siap untuk tahap pemrosesan selanjutnya. Hasil data *cleaning* dapat ditunjukkan pada Tabel 4.2.

Tabel 4.2 Hasil Data *Cleaning* dan Representasi Emoji

Ulasan Sebelum *Preprocessing*	**Data *Cleaning***
First time ke payakumbuah, service nya mantap banget. Dapet free kopi milo. dendeng batokok lama banak 👌 😍	First time ke payakumbuah service nya mantap banget Dapet free kopi milo. dendeng batokok lama banak oke suka
makanannya enak bgt apalagi ikan salainya. Mantab 👍 👍	makanannya enak bgt apalagi ikan salainya Mantab bagus bagus
Telor barendo dan ikan salai the best.. nasinya beda dari rm padang lain	Telor barendo dan ikan salai the best nasinya beda dari rm padang lain
HARUS TAMBAH ATAU GANTI MANPOWER, PESAN AYAM POP TAPI ENGGA DATENG2, UDAH 3 KALI DIINGETIN TAPI AYAM TETAP GA DATANG, CUMA IYA2 AJA, AKHIRNYA CUMA MAKAN TELUR DADAR DAN KERUPUK KULIT PAKAI NASI, REALLY HATE THE SERVICES.	HARUS TAMBAH ATAU GANTI MANPOWER PESAN AYAM POP TAPI ENGGA DATENG UDAH KALI DIINGETIN TAPI AYAM TETAP GA DATANG CUMA IYA AJA AKHIRNYA CUMA MAKAN TELUR DADAR DAN KERUPUK KULIT PAKAI NASI REALLY HATE THE SERVICES
Tempat nyaman dan makanannya enak, pas banget buat makan ramean. Service staffnya jg ramah sekali, mas Alfarabi.	Tempat nyaman dan makanannya enak pas banget buat makan ramean Service staffnya jg ramah sekali mas Alfarabi
1. Ayam goreng, 28rb (B aja) 2. Rendang, 35rb (Enak, bumbunya wangi banget) 3. Dendeng batokok lado hijau, 35rb (Enak, cabe nya wangi krn pakai bawang merah) 4. Sayur kapau, 17rb 5. Sambal hijau, 10rb (kurang, tipe cabe yg sdh halus dan rada encer) 6. Nasi putih, 12.500 7. Teh tawar, 10rb 8. Es teh tawar, 12rb 9. Pelayanan bagus.	Ayam goreng aja Rendang Enak, bumbunya wangi banget Dendeng batokok lado hijau, Enak, cabe nya wangi krn pakai bawang merah Sayur kapau Sambal hijau, kurang tipe cabe yg sdh halus dan rada encer Nasi putih Teh tawar Es teh tawar Pelayanan bagus Enjoy oke

Ulasan Sebelum *Preprocessing*	Data *Cleaning*
10. Enjoy 👌	

4.2.2 *Case Folding*

Pada tahap ini, seluruh teks dikonversi menjadi huruf kecil untuk menjaga konsistensi data dan menghindari perbedaan akibat penggunaan huruf besar dan kecil. Proses ini dilakukan dengan menggunakan *method lower()*, yang secara otomatis mengubah semua huruf dalam teks menjadi huruf kecil. Dengan langkah ini, kata-kata seperti "Bagus" dan "bagus" akan dianggap sama, sehingga analisis teks menjadi lebih akurat dan terstandarisasi.

Tabel 4.3 Hasil *Case Folding*

Data *Cleaning*	*Case Folding*
First time ke payakumbuah service nya mantap banget Dapet free kopi milo. dendeng batokok lama banak oke suka	first time ke payakumbuah service nya mantap banget dapet free kopi milo dendeng batokok lama banak oke suka
makanannya enak bgt apalagi ikan salainya Mantab bagus bagus	makanannya enak bgt apalagi ikan salainya mantab bagus bagus
Telor barendo dan ikan salai the best nasinya beda dari rm padang lain	telor barendo dan ikan salai the best nasinya beda dari rm padang lain
HARUS TAMBAH ATAU GANTI MANPOWER PESAN AYAM POP TAPI ENGGA DATENG UDAH KALI DIINGETIN TAPI AYAM TETAP GA DATANG CUMA IYA AJA AKHIRNYA CUMA MAKAN TELUR DADAR DAN KERUPUK KULIT PAKAI NASI REALLY HATE THE SERVICES	harus tambah atau ganti manpower pesan ayam pop tapi engga dateng udah kali diingetin tapi ayam tetap ga datang cuma iya aja akhirnya cuma makan telur dadar dan kerupuk kulit pakai nasi really hate the services
Tempat nyaman dan makanannya enak pas banget buat makan ramean Service staffnya jg ramah sekali mas Alfarabi	tempat nyaman dan makanannya enak pas banget buat makan ramean service staffnya jg ramah sekali mas alfarabi
Ayam goreng aja Rendang Enak, bumbunya wangi banget Dendeng batokok lado hijau, Enak, cabe nya wangi krn pakai bawang merah Sayur kapau Sambal hijau, kurang tipe cabe yg sdh halus dan rada encer Nasi putih Teh tawar Es teh tawar Pelayanan bagus Enjoy oke	ayam goreng aja rendang enak bumbunya wangi banget dendeng batokok lado hijau enak cabe nya wangi krn pakai bawang merah sayur kapau sambal hijau kurang tipe cabe yg sdh halus dan rada encer nasi putih teh tawar es teh tawar pelayanan bagus enjoy oke

4.2.3 *Tokenization*

Tokenization merupakan proses memecah teks menjadi kata-kata yang lebih kecil atau *token* untuk mempermudah analisis lebih lanjut. Dalam proses ini, *library spaCy* digunakan untuk melakukan tokenisasi secara otomatis. Model NLP *nlp = spacy.load("xx_ent_wiki_sm")* dimanfaatkan untuk memproses teks, sehingga sistem dapat memahami struktur kalimat dengan lebih baik. Setelah teks diproses oleh model tersebut, fungsi *doc = nlp(text_after_abbreviation)* digunakan untuk membagi teks menjadi unit-unit kata atau token. Selanjutnya, daftar token yang telah dihasilkan oleh *spaCy* diekstraksi menggunakan perintah *tokens = [token.text for token in doc]*, yang mengumpulkan semua kata hasil tokenisasi dalam bentuk list. Dengan pendekatan ini, teks dapat diuraikan secara sistematis, memungkinkan analisis yang lebih mendalam dalam pemrosesan bahasa alami (NLP).

Tabel 4.4 Hasil *Tokenization*

Case Folding	***Tokenization***
first time ke payakumbuah service nya mantap banget dapet free kopi milo dendeng batokok lama banak oke suka	'first' 'time' 'ke' 'payakumbuah' 'service' 'nya' 'mantap' 'banget' 'dapet' 'free' 'kopi' 'milo' 'dendeng' 'batokok' 'lama' 'banak' 'oke' 'suka'
makanannya enak bgt apalagi ikan salainya mantab bagus bagus	'makanannya' 'enak' 'bgt' 'apalagi' 'ikan' 'salainya' 'mantab' 'bagus' 'bagus'
telor barendo dan ikan salai the best nasinya beda dari rm padang lain	'telor' 'barendo' 'dan' 'ikan' 'salai' 'the' 'best' 'nasinya' 'beda' 'dari' 'rm' 'padang' 'lain'
harus tambah atau ganti manpower pesan ayam pop tapi engga dateng udah kali diingetin tapi ayam tetap ga datang cuma iya aja akhirnya cuma makan telur dadar dan kerupuk kulit pakai nasi really hate the services	'harus' 'tambah' 'atau' 'ganti' 'manpower' 'pesan' 'ayam' 'pop' 'tapi' 'engga' 'dateng' 'udah' 'kali' 'diingetin' 'tapi' 'ayam' 'tetap' 'ga' 'datang' 'cuma' 'iya' 'aja' 'akhirnya' 'cuma' 'makan' 'telur' 'dadar' 'dan' 'kerupuk' 'kulit' 'pakai' 'nasi' 'really' 'hate' 'the' 'services'
tempat nyaman dan makanannya enak pas banget buat makan ramean service staffnya jg ramah sekali mas alfarabi	'tempat' 'nyaman' 'dan' 'makanannya' 'enak' 'pas' 'banget' 'buat' 'makan' 'ramean' 'service' 'staffnya' 'jg' 'ramah' 'sekali' 'mas' 'alfarabi'
ayam goreng aja rendang enak bumbunya wangi banget dendeng batokok lado hijau enak cabe nya wangi krn pakai bawang merah sayur kapau sambal hijau kurang tipe cabe yg sdh halus dan rada encer nasi putih teh tawar es teh tawar pelayanan bagus enjoy oke	'ayam' 'goreng' 'aja' 'rendang' 'enak' 'bumbunya' 'wangi' 'banget' 'dendeng' 'batokok' 'lado' 'hijau' 'enak' 'cabe' 'nya' 'wangi' 'krn' 'pakai' 'bawang' 'merah' 'sayur' 'kapau' 'sambal' 'hijau' 'kurang' 'tipe' 'cabe' 'yg' 'sdh' 'halus' 'dan' 'rada' 'encer' 'nasi' 'putih' 'teh' 'tawar' 'es' 'teh' 'tawar' 'pelayanan' 'bagus' 'enjoy' 'oke'

4.2.4 *Stopword*

Pada tahap *stopword*, proses untuk menghapus kata yang tidak memberikan makna atau informasi signifikan untuk analisis sentimen, seperti kata sambung, kata ganti, atau preposisi. Dalam proses ini, library *spaCy* digunakan kembali untuk secara otomatis mendeteksi dan menghapus *stopwords* dari teks. Proses ini dilakukan dengan perintah *tokens_after_stopwords = [word for word in tokens if word not in nlp.Defaults.stop_words]*, yang akan menyaring token-token yang masih memiliki makna dan mengabaikan kata-kata yang sudah teridentifikasi sebagai *stopwords*.

Tabel 4.5 Hasil *Stopword*

Tokenization	***Stopword***
'first' 'time' 'ke' 'payakumbuah' 'service' 'nya' 'mantap' 'banget' 'dapet' 'free' 'kopi' 'milo' 'dendeng' 'batokok' 'lama' 'banak 'oke' 'suka'	first time ke payakumbuah service nya mantap banget dapet free kopi milo dendeng batokok lama banak oke suka
'makanannya' 'enak' 'bgt' 'apalagi' 'ikan' 'salainya' 'mantab' 'bagus' 'bagus'	makananya enak banget apalagi ikan salainya mantab bagus bagus
'telor' 'barendo' 'dan' 'ikan' 'salai' 'the' 'best' 'nasinya' 'beda' 'dari' 'padang' 'lain'	telor barendo dan ikan salai the best nasinya beda dari padang lain
'harus' 'tambah' 'atau' 'ganti' 'manpower' 'pesan' 'ayam' 'pop' 'tapi' 'engga' 'dateng' 'udah' 'kali' 'diingetin' 'tapi' 'ayam' 'tetap' 'ga' 'datang' 'cuma' 'iya' 'aja' 'akhirnya' 'cuma' 'makan' 'telur' 'dadar' 'dan' 'kerupuk' 'kulit' 'pakai' 'nasi' 'really' 'hate' 'the' 'services'	harus tambah atau ganti manpower pesan ayam pop tapi enga dateng udah kali dingetin tapi ayam tetap ga datang cuma iya aja akhirnya cuma makan telur dadar dan kerupuk kulit pakai nasi really hate the services
'tempat' 'nyaman' 'dan' 'makanannya' 'enak' 'pas' 'banget' 'buat' 'makan' 'ramean' 'service' 'staffnya' 'jg' 'ramah' 'sekali' 'mas' 'alfarabi'	tempat nyaman dan makananya enak pas banget buat makan ramean service stafnya jg ramah sekali mas alfarabi
'ayam' 'goreng' 'aja' 'rendang' 'enak' 'bumbunya' 'wangi' 'banget' 'dendeng' 'batokok' 'lado' 'hijau' 'enak' 'cabe' 'nya' 'wangi' 'krn' 'pakai' 'bawang' 'merah' 'sayur' 'kapau' 'sambal' 'hijau' 'kurang' 'tipe' 'cabe' 'yg' 'sdh' 'halus' 'dan' 'rada' 'encer' 'nasi' 'putih' 'teh' 'tawar' 'es' 'teh' 'tawar' 'pelayanan' 'bagus' 'enjoy' 'oke'	ayam goreng aja rendang enak bumbunya wangi banget dendeng batokok lado hijau enak cabe nya wangi karena pakai bawang merah sayur kapau sambal hijau kurang tipe cabe yang sdh halus dan rada encer nasi putih teh tawar es teh tawar pelayanan bagus enjoy oke

4.2.5 *Stemming*

Dalam proses ini, *library* Sastrawi digunakan untuk melakukan *stemming* dalam bahasa Indonesia. Langkah pertama adalah membuat objek stemmer dengan *factory = StemmerFactory(),* kemudian *stemmer = factory.create_stemmer()* digunakan untuk menginisialisasi *stemmer*. Setelah itu, kata-kata dalam teks diproses menggunakan fungsi *stemmed_word = stemmer.stem(word),* yang secara otomatis mengubah kata menjadi bentuk dasarnya.

Tabel 4.6 Hasil *Stemming*

Stopword	***Stemming***
first time ke payakumbuah service nya mantap banget dapet free kopi milo dendeng batokok lama banak oke suka	first time ke payakumbuah service mantap banget dapet free kopi milo dendeng batokok lama banak oke suka
makananya enak banget apalagi ikan salainya mantab bagus bagus	makan enak banget apalagi ikan salai mantab bagus bagus
telor barendo dan ikan salai the best nasinya beda dari padang lain	telor barendo dan ikan salai the best nasi beda dari padang lain
harus tambah atau ganti manpower pesan ayam pop tapi enga dateng udah kali dingetin tapi ayam tetap ga datang cuma iya aja akhirnya cuma makan telur dadar dan kerupuk kulit pakai nasi really hate the services	harus tambah atau ganti manpower pesan ayam pop tapi enga dateng udah kali inget tapi ayam tetap ga datang cuma iya aja akhir cuma makan telur dadar dan kerupuk kulit pakai nasi really hate the services
tempat nyaman dan makananya enak pas banget buat makan ramean service stafnya jg ramah sekali mas alfarabi	tempat nyaman dan makan enak pas banget buat makan ramean service staf jg ramah sekali mas alfarabi
ayam goreng aja rendang enak bumbunya wangi banget dendeng batokok lado hijau enak cabe nya wangi karena pakai bawang merah sayur kapau sambal hijau kurang tipe cabe yang sdh halus dan rada encer nasi putih teh tawar es teh tawar pelayanan bagus enjoy oke	ayam goreng aja rendang enak bumbu wangi banget dendeng batokok lado hijau enak cabe nya wangi karena pakai bawang merah sayur kapau sambal hijau kurang tipe cabe yang sdh halus dan rada encer nasi putih teh tawar es teh tawar layan bagus enjoy oke

4.3 Normalisasi

Normalisasi bertujuan untuk mengganti kata-kata singkatan atau slang menjadi bentuk yang lebih baku. Sebagai contoh, kata singkatan seperti "enk" diubah menjadi "enak", atau "krn" diubah menjadi "karena". Peneliti telah menyiapkan kamus kata yang berisi daftar singkatan dan padanan kata yang benar untuk mempermudah proses ini. Contoh kamus kata:

"dtng": "datang", "smua": "semua", "lg": "lagi", "tp": "tapi", "utk": "untuk", "yg": "yang", "bgt": "banget", "bngt": "banget", "fav": "favorit", "dr": "dari", "mknan": "makanan", "Jkt": "Jakarta", "bbrp": "beberapa", "dgn": "dengan", "sm": "sama", "jdi": "jadi", "jd": "jadi", "sy": "saya", "sya": "saya", "lbh": "lebih", "msh": "masih", "mkn": "makan", "hrs": "harus", "ksni": "kesini", "ksna": "kesana", "ydh": "yaudah", "pd": "pada", "trs": "terus", "gt": "gitu", "krn": "karena", "pdhl": "padahal", "kya": "kayak", "udh": "udah", "drpd": "daripada", "klo": "kalo", "tdk": "tidak", "brg": "bareng", "cm": "cuma", "lt": "lantai", "emg": "emang", "ttp": "tetap", "pgi": "pagi", "sgtu": "segitu", "bnr": "benar", "bs": "bisa", "jg": "juga", "blg": "bilang"

Normalisasi kata dilakukan menggunakan kamus kata (*dictionary lookup*) untuk menggantikan kata-kata singkatan dengan bentuk yang lebih baku. Proses ini menggunakan *method .get()* pada *dictionary* kamus_kata**,** di mana setiap kata dalam teks dicek apakah terdapat dalam kamus; jika ada, kata tersebut diganti dengan bentuk normalnya, jika tidak, tetap digunakan.

Tabel 4.7 Sampel Hasil Normalisasi

Stemming	**Normalisasi**
first time ke payakumbuah service nya mantap banget dapet free kopi milo dendeng batokok lama banak oke suka	first time ke payakumbuah service nya mantap banget dapet free kopi milo dendeng batokok lama banak oke suka
makan enak banget apalagi ikan salai mantab bagus bagus	makan enak banget apalagi ikan salai mantab bagus bagus
telor barendo dan ikan salai the best nasi beda dari padang lain	telor barendo dan ikan salai the best nasi beda dari padang lain
harus tambah atau ganti manpower pesan ayam pop tapi enga dateng udah kali inget tapi ayam tetap ga datang cuma iya aja akhir cuma makan telur dadar dan kerupuk kulit pakai nasi really hate the services	harus tambah atau ganti manpower pesan ayam pop tapi ga dateng udah kali inget tapi ayam tetap ga datang cuma iya aja akhir cuma makan telur dadar dan kerupuk kulit pakai nasi really hate the services
tempat nyaman dan makan enak pas banget buat makan ramean service staf jg ramah sekali mas alfarabi	tempat nyaman dan makan enak pas banget buat makan ramean service staf juga ramah sekali mas alfarabi

Stemming	**Normalisasi**
ayam goreng rb aja rendang rb enak bumbu wangi banget dendeng batokok lado hijau rb enak cabe nya wangi karena pakai bawang merah sayur kapau rb sambal hijau rb kurang tipe cabe yang sdh halus dan rada encer nasi putih teh tawar rb es teh tawar rb layan bagus enjoy oke	ayam goreng ribu aja rendang ribu enak bumbu wangi banget dendeng batokok lado hijau ribu enak cabe nya wangi karena pakai bawang merah sayur kapau ribu sambal hijau ribu kurang tipe cabe yang sdh halus dan rada encer nasi putih teh tawar ribu es teh tawar ribu layan bagus enjoy oke

4.4 Hasil Pembobotan Kata

Dari ulasan sudah melewati tahapan text processing, maka tahap selanjutnya yaitu representasi kata dengan menggunakan metode TF-IDF. Metode TF-IDF merupakan *upgrade* dari berbagai metode sebelumnya, yaitu menggabungkan dari dua metode yaitu metode TF dan IDF. Maka dari itu nilai TF-IDF di dapat dari perhitungan perkalian nilai TF dan IDF. Representasi kata dengan TF-IDF menggunakan *library TfidfVectorizer*. *Library TfidfVectorizer* berfungsi untuk merubah teks menjadi numerik secara langsung, menghitung frekuensi kata (TF) dan memberi bobot berdasarkan seberapa penting kata tersebut dalam keseluruhan korpus (IDF). Dengan cara ini, matriks TF-IDF yang menggambarkan bobot kata-kata dalam setiap ulasan dapat diperoleh tanpa perlu menggunakan proses terpisah untuk menghitung TF dan IDF.

4.5 Pembagian Data

Setelah tahap *prepocessing* data dilakukan, berikutnya yaitu membagi data menjadi dua, yaitu data *training* dan data *testing*. Data *training* digunakan untuk data latih model LSTM agar dapat mempelajari pola-pola sentimen dari ulasan rumah makan padang payakumbuah yang telah dikumpulkan. Sedangkan data *testing* digunakan untuk menguji performa model LSTM yang telah dilatih pada

data *training*. Pembagian data ini dilakukan secara proporsional, oleh karena itu porsi data *training* lebih besar dibandingkan data *testing*. Jika data *training* lebih sedikit, model mungkin tidak cukup belajar atau tidak mampu mengenali pola dengan baik.

Proses ini penting untuk memastikan bahwa model LSTM yang dibangun memiliki kemampuan generalisasi yang baik dan dapat digunakan untuk memprediksi sentimen pada ulasan rumah makan padang payakumbuah secara akurat. Pembagian data dilakukan dengan membandingkan berbagai rasio seperti pada Tabel 4.8, yaitu 60% data *training* dan 40% data *testing*, 70% data *training* dan 30% data *testing*, 80% data *training* dan 20% data *testing*, serta 90% data *training* dan 10% data *testing*.

Tabel 4.8 Hasil Pembagian Data

Rasio	**Data *Training***	**Data *Testing***
60% : 40%	586	390
70% : 30%	683	293
80% : 20%	781	195
90% : 10%	878	98

4.6 Pemodelan LSTM

Pemodelan LSTM dilakukan menggunakan *TensorFlow* dan *Sklearn* untuk menangani analisis sentimen pada data teks. Proses ini dilakukan dengan memanfaatkan kemampuan LSTM dalam menangani data urutan, seperti teks, yang memiliki dependensi temporal (misalnya, kata yang muncul lebih awal dalam kalimat bisa berhubungan dengan kata yang muncul kemudian). Arsitektur model LSTM yang diterapkan terdiri dari beberapa lapisan LSTM yang diikuti oleh lapisan Dense untuk klasifikasi output menjadi dua kategori positif dan negatif.

Model ini dilatih menggunakan data yang telah diproses sebelumnya dengan TF-IDF.

4.7 Pelatihan Model

Proses pelatihan model menggunakan pembagian rasio data yang ditetapkan untuk mengevaluasi performa model. Proses pelatihan pertama *split* data menggunakan empat rasio yaitu 60:40 (60% data *training* dan 40% data *testing*), 70:30 (70% data *training* dan 30% data *testing*), 80:20 (80% data *training* dan 20% data *testing*), dan 90:10 (90% data *training* dan 10% data *testing*). Pelatihan kedua yaitu melatih *split* data menggunakan *k-fold cross validation* dengan K yaitu 10. Pengujian ini dilakukan bertujuan untuk menentukan evaluasi model untuk mencapai akurasi yang terbaik dalam analisis sentimen ulasan rumah makan padang Payakumbuah.

4.7.1 *Split* Data

Uji coba pelatihan model dengan *split* data terdapat empat percobaan yaitu dengan skenario rasio 60:40, 70:30, 80:20 dan 90:10.

4.7.1.1 Uji Coba Rasio 60:40

Pengujian pertama dilakukan dengan menggunakan rasio 60:40. Dari hasil pengujian ini didapatkan hasil *confusion matrix* pada Tabel 4.9, hasil *loss training* dan *test accuracy* pada Tabel 4.10 dan perhitungan masing-masing parameter setiap kelas pada Tabel 4.11 dengan menggunakan persamaan 3.7 sampai 3.14 sebagai berikut.

Tabel 4.9 Hasil *Confusion Matrix* Rasio 60:40

Aktual	**Prediksi**	
	Negatif	**Positif**
Negatif	0	61
Positif	0	330

Tabel 4.10 Hasil *Training loss* dan *Test accuracy* 60:40

Epoch	***Training loss***	***Test accuracy***
1	65,17%	84,39%
2	45,53%	84,39%
3	43,85%	84,39%
4	43,92%	84,39%
5	43,63%	84,39%
6	43,25%	84,39%
7	44,02%	84,39%
8	43,21%	84,39%
9	43,77%	84,39%
10	44,61%	84,39%

Tabel 4.11 Hasil Perhitungan Masing-Masing Parameter setiap Kelas

	Precision	***Recall***	***F1-score***	**Akurasi**
Negatif	0%	0%	0%	84,39%
Positif	84,39%	100%	91,54%	

$$\text{Akurasi} = \frac{330 + 0}{330 + 0 + 61 + 0} \times 100\% = 84{,}39\%$$

$$\textit{Precision}\ \text{Positif} = \frac{330}{330 + 61} \times 100\% = 84{,}39\%$$

$$\textit{Precision}\ \text{Negatif} = \frac{0}{0 + 61} \times 100\% = 0\%$$

$$\textit{Recall}\ \text{Positif} = \frac{330}{330 + 0} \times 100\% = 100\%$$

$$\textit{Recall}\ \text{Negatif} = \frac{0}{0 + 0} \times 100\% = 0\%$$

$$\textit{F1-score}\ \text{Positif} = 2 \times \frac{0{,}8439 \times 1.0}{0{,}8439 + 1.0} \times 100\% = 91{,}54\%$$

$$\textit{F1-score}\ \text{Negatif} = 2 \times \frac{0 \times 0}{0 + 0} \times 100\% = 0\%$$

Pada skenario 60:40, hasil perhitungan menunjukkan nilai akurasi sebesar 84,39%, untuk kelas positif nilai *precision* adalah 84,39%, *recall* 100% dan *f1-score* 91,54%. Untuk kelas negatif *precision*, *recall* dan *f1-score* mendapat nilai

yang sama yaitu 0%. Berdasarkan Tabel 4.10, model menunjukkan penurunan *training loss* yang signifikan dari 65,17% pada *epoch* pertama menjadi 44,61% pada *epoch* ke-10, namun *test accuracy* tetap konstan di angka 84,39% sepanjang pelatihan.

4.7.1.2 Uji Coba Rasio 70:30

Pengujian pertama dilakukan dengan menggunakan rasio 70:30. Dari hasil pengujian ini didapatkan hasil *confusion matrix* pada Tabel 4.12, hasil *loss training* dan *test accuracy* pada Tabel 4.13 dan perhitungan masing-masing parameter setiap kelas pada Tabel 4.14 dengan menggunakan persamaan 3.7 sampai 3.14 sebagai berikut.

Tabel 4.12 Hasil *Confusion Matrix* Rasio 70:30

Aktual	**Prediksi**	
	Negatif	**Positif**
Negatif	0	46
Positif	0	247

Tabel 4.13 Hasil *Training loss* dan *Test accuracy* 70:30

Epoch	***Training loss***	***Test accuracy***
1	60,83%	84,30%
2	44,67%	84,30%
3	44,00%	84,30%
4	43,59%	84,30%
5	43,94%	84,30%
6	43,90%	84,30%
7	43,21%	84,30%
8	44,14%	84,30%
9	43,47%	84,30%
10	43,96%	84,30%

Tabel 4.14 Hasil Perhitungan Masing-Masing Parameter setiap Kelas

	Precision	***Recall***	***F1-score***	**Akurasi**
Negatif	0%	0%	0%	84,30%
Positif	84,30%	100%	91,48%	

Akurasi	$= \frac{247 + 0}{247 + 0 + 46 + 0} \times 100\%$	= 84,30%
Precision Positif	$= \frac{247}{247 + 46} \times 100\%$	= 84,30%
Precision Negatif	$= \frac{0}{0 + 46} \times 100\%$	= 0%
Recall Positif	$= \frac{247}{247 + 0} \times 100\%$	= 100%
Recall Negatif	$= \frac{0}{0 + 0} \times 100\%$	= 0%
F1-score Positif	$= 2 \times \frac{0{,}843 \times 1.0}{0{,}843 + 1.0} \times 100\%$	= 91,48%
F1-score Negatif	$= 2 \times \frac{0 \times 0}{0 + 0} \times 100\%$	= 0%

Pada skenario 70:30, hasil perhitungan menunjukkan nilai akurasi sebesar 84,30%, untuk kelas positif nilai *precision* adalah 84,30%, *recall* 100% dan *f1-score* 91,48%. Untuk kelas negatif *precision*, *recall* dan *f1-score* mendapat nilai yang sama yaitu 0%. Berdasarkan Tabel 4.13, model menunjukkan penurunan *training loss* yang signifikan dari 60,83% pada *epoch* pertama menjadi 43,96% pada *epoch* ke-10, namun *test accuracy* tetap konstan di angka 84,30% sepanjang pelatihan.

4.7.1.3 Uji Coba Rasio 80:20

Pengujian pertama dilakukan dengan menggunakan rasio 80:20. Dari hasil pengujian ini didapatkan hasil *confusion matrix* pada Tabel 4.15, hasil *loss training* dan *test accuracy* pada Tabel 4.16 dan perhitungan masing-masing parameter setiap kelas pada Tabel 4.17 dengan menggunakan persamaan 3.7 sampai 3.14 sebagai berikut.

Tabel 4.15 Hasil *Confusion Matrix* Rasio 80:20

Aktual	**Prediksi**	
	Negatif	**Positif**
Negatif	0	31
Positif	0	165

Tabel 4.16 Hasil *Training loss* dan *Test accuracy* 80:20

Epoch	***Training loss***	***Test accuracy***
1	56,80%	84,18%
2	43,69%	84,18%
3	44,46%	84,18%
4	43,33%	84,18%
5	43,78%	84,18%
6	43,68%	84,18%
7	44,03%	84,18%
8	43,80%	84,18%
9	43,79%	84,18%
10	43,94%	84,18%

Tabel 4.17 Hasil Perhitungan Masing-Masing Parameter setiap Kelas

	Precision	***Recall***	***F1-score***	**Akurasi**
Negatif	0%	0%	0%	84,18%
Positif	84,18%	100%	91,41%	

$$\text{Akurasi} = \frac{165 + 0}{165 + 0 + 31 + 0} \times 100\% = 84{,}18\%$$

$$\textit{Precision}\ \text{Positif} = \frac{165}{165 + 31} \times 100\% = 84{,}18\%$$

$$\textit{Precision}\ \text{Negatif} = \frac{0}{0 + 31} \times 100\% = 0\%$$

$$\textit{Recall}\ \text{Positif} = \frac{165}{165 + 0} \times 100\% = 100\%$$

$$\textit{Recall}\ \text{Negatif} = \frac{0}{0 + 0} \times 100\% = 0\%$$

$$\textit{F1-score}\ \text{Positif} = 2 \times \frac{0{,}843 \times 1.0}{0{,}843 + 1.0} \times 100\% = 91{,}41\%$$

$$\textit{F1-score}\ \text{Negatif} = 2 \times \frac{0 \times 0}{0 + 0} \times 100\% = 0\%$$

Pada skenario 80:20, hasil perhitungan menunjukkan nilai akurasi sebesar 84,18%, untuk kelas positif nilai *precision* adalah 84,18%, *recall* 100% dan *f1-score* 91,41%. Untuk kelas negatif *precision*, *recall* dan *f1-score* mendapat nilai

yang sama yaitu 0%. Berdasarkan Tabel 4.16, model menunjukkan penurunan *training loss* yang signifikan dari 56,80% pada *epoch* pertama menjadi 43,94% pada *epoch* ke-10, namun *test accuracy* tetap konstan di angka 84,18% sepanjang pelatihan.

4.7.1.4 Uji Coba Rasio 90:10

Pengujian pertama dilakukan dengan menggunakan rasio 90:10. Dari hasil pengujian ini didapatkan hasil *confusion matrix* pada Tabel 4.18, hasil *loss training* dan *test accuracy* pada Tabel 4.19 dan perhitungan masing-masing parameter setiap kelas pada Tabel 4.20 dengan menggunakan persamaan 3.7 sampai 3.14 sebagai berikut.

Tabel 4.18 Hasil *Confusion Matrix* Rasio 90:10

Aktual	**Prediksi**	
	Negatif	**Positif**
Negatif	0	15
Positif	0	83

Tabel 4.19 Hasil *Training loss* dan *Test accuracy* 90:10

Epoch	***Training loss***	***Test accuracy***
1	57,54%	84,69%
2	43,93%	84,69%
3	43,55%	84,69%
4	43,55%	84,69%
5	43,80%	84,69%
6	44,15%	84,69%
7	43,50%	84,69%
8	43,87%	84,69%
9	43,71%	84,69%
10	43,73%	84,69%

Tabel 4.20 Hasil Perhitungan Masing-Masing Parameter setiap Kelas

	Precision	***Recall***	***F1-score***	**Akurasi**
Negatif	0%	0%	0%	84,69%
Positif	84,69%	100%	91,71%	

Akurasi	$= \frac{83+0}{83+0+15+0} \times 100\%$	= 84,69%
Precision Positif	$= \frac{83}{83+15} \times 100\%$	= 84,69%
Precision Negatif	$= \frac{0}{0+15} \times 100\%$	= 0%
Recall Positif	$= \frac{83}{83+0} \times 100\%$	= 100%
Recall Negatif	$= \frac{0}{0+0} \times 100\%$	= 0%
F1-score Positif	$= 2 \times \frac{0{,}8469 \times 1.0}{0{,}8469+1.0} \times 100\%$	= 91,71%
F1-score Negatif	$= 2 \times \frac{0 \times 0}{0+0} \times 100\%$	= 0%

Pada skenario 90:10, hasil perhitungan menunjukkan nilai akurasi sebesar 84,69%, untuk kelas positif nilai *precision* adalah 84,69%, *recall* 100% dan *f1-score* 91,71%. Untuk kelas negatif *precision*, *recall* dan *f1-score* mendapat nilai yang sama yaitu 0%. Berdasarkan Tabel 4.19, model menunjukkan penurunan *training loss* yang signifikan dari 57,54% pada *epoch* pertama menjadi 43,73% pada *epoch* ke-10, namun *test accuracy* tetap konstan di angka 84,69% sepanjang pelatihan.

4.7.2 *Split* Data dengan 10-*Fold Cross Validation*

Uji coba pelatihan model kedua *split* data dengan k-*fold cross validation* terdapat empat skenario rasio yaitu 60:40, 70:30, 80:20 dan 90:10 dengan menggunakan 10-*fold cross validation*.

4.7.2.1 Uji Coba Rasio 60:40

Pada pengujian rasio 60:40 hasil pada Tabel 4.21 menunjukkan penurunan *training loss* di setiap *n-fold*, dengan variasi yang berbeda. Pada *n-fold* 1, *training loss* turun dari 66,14% menjadi 45,21%, dengan *test accuracy* stabil di 89,83%. Di *n-fold* 2, meskipun *training loss* turun dari 65,01% menjadi 44,31%, *test accuracy* tetap stabil di 88,14%. Pada *n-fold* 3, *training loss* berkurang dari 64,12% menjadi 43,63%, sementara *test accuracy* tetap di 84,75%. Di *n-fold* 4, *training loss* menurun dari 65,07% menjadi 42,42%, namun *test accuracy* lebih rendah, yakni 77,97%. Pada *n-fold* 5, meskipun *training loss* berkurang dari 63,56% menjadi 41,90%, *test accuracy* tetap di 76,27%. Pada *n-fold* 6, *training loss* turun dari 62,99% menjadi 42,71%, dengan *test accuracy* stabil di 79,31%. Di *n-fold* 7, *training loss* menurun dari 66,77% menjadi 44,59%, namun *test accuracy* tetap stabil di 87,93%. Pada *n-fold* 8, meskipun *training loss* turun dari 65,80% menjadi 44,09%, *test accuracy* lebih rendah, yaitu 86,21%. Di *n-fold* 9 dan *n-fold* 10, *training loss* masing-masing berkurang, dengan *test accuracy* yang stabil pada 85,42% dan 79,17%. Secara keseluruhan, meskipun *training loss* menurun di setiap *n-fold*, *test accuracy* menunjukkan hasil yang bervariasi antar *n-fold*.

Tabel 4.21 Hasil *Training loss* dan *Test accuracy* 60:40 dengan 10-*Fold*

n-fold	***Epoch***	***Training loss***	***Test accuracy***
1	1	66,14%	89,83%
	2	48,17%	89,83%
	…	…	…
	9	44,95%	89,83%
	10	45,21%	89,83%
2	1	65,01%	88,14%
	2	45,49%	88,14%
	…	…	…
	9	45,06%	88,14%
	10	44,31%	88,14%
3	1	64,12%	84,75%
	2	44,59%	84,75%

n-fold	*Epoch*	*Training loss*	*Test accuracy*
	…	…	…
	9	43,89%	84,75%
	10	43,63%	84,75%
4	1	65,07%	77,97%
	2	43,73%	77,97%
	…	…	…
	9	42,76%	77,97%
	10	42,42%	77,97%
5	1	63,56%	76,27%
	2	44,70%	76,27%
	…	…	…
	9	42,49%	76,27%
	10	41,90%	76,27%
6	1	62,99%	79,31%
	2	45,87%	79,31%
	…	…	...
	9	42,53%	79,31%
	10	42,71%	79,31%
7	1	66,77%	87,93%
	2	48,31%	87,93%
	…	…	…
	9	44,88%	87,93%
	10	44,59%	87,93%
8	1	65,80%	86,21%
	2	46,10%	86,21%
	…	…	…
	9	44,17%	86,21%
	10	44,09%	86,21%
9	1	62,99%	85,42%
	2	46,93%	85,42%
	…	…	…
	9	44,05%	85,42%
	10	44,28%	85,42%
10	1	63,90%	79,17%
	2	45,37%	79,17%
	…	…	…
	9	44,15%	79,17%
	10	44,15%	79,17%

Hasil rasio 60:40 10-*fold cross validation* mendapatkan hasil perhitungan seperti pada Tabel 4.22. Nilai akurasi yang dihasilkan setiap *fold* tidak stabil di awal dan mulai stabil dari 3 *fold* terakhir. Nilai akurasi mencapai nilai tertinggi yaitu 89,83% pada *fold* ke-1. Sedangkan untuk nilai akurasi terendah berada pada *fold*

ke-5 dengan nilai 76,27%. Rata-rata keseluruhan akurasi dari 10-*fold* rasio 60:40 yaitu sebesar 84,28%.

Tabel 4.22 Hasil 10-*fold Cross Validation* Rasio 60:40

n-fold	***Precision* Positif**	***Precision* Negatif**	***Recall* Positif**	***Recall* Negatif**	***F1-score* Positif**	***F1-score* Negatif**	**Akurasi**	**Rata-Rata Akurasi**
1	89,83%	0%	100%	0%	94,64%	0%	89,83%	84,28%
2	88,14%	0%	100%	0%	93,69%	0%	88,14%	
3	84,75%	0%	100%	0%	91,74%	0%	84,75%	
4	77,97%	0%	100%	0%	87,62%	0%	77,97%	
5	76,27%	0%	100%	0%	86,54%	0%	76,27%	
6	79,31%	0%	100%	0%	88,46%	0%	79,31%	
7	87,93%	0%	100%	0%	93,58%	0%	87,93%	
8	86,21%	0%	100%	0%	92,59%	0%	86,21%	
9	86,21%	0%	100%	0%	92,59%	0%	85,42%	
10	86,21%	0%	100%	0%	92,59%	0%	79,17%	

4.7.2.2 Uji Coba Rasio 70:30

Pada pengujian rasio 70:30 hasil pada Tabel 4.23 menunjukkan penurunan *training loss* di setiap *n-fold*, dengan variasi yang berbeda. Pada *n-fold* 1, *training loss* turun dari 59,66% menjadi 42,86%, dengan *test accuracy* stabil di 78,26%. Di *n-fold* 2, meskipun *training loss* turun dari 60,85% menjadi 43,94%, *test accuracy* meningkat signifikan ke 86,96%. Sementara itu, pada *n-fold* 3 hingga *n-fold* 7, *test accuracy* stabil di kisaran 82,35%, meski *training loss* menurun. Pada *n-fold* 8, *test accuracy* lebih tinggi di 89,71%, meskipun *training loss* turun dari 60,67% menjadi 44,76%, *n-fold* 9 dan *n-fold* 10 menunjukkan *test accuracy* yang stabil pada 82,35%, dengan *training loss* masing-masing berkurang.

Tabel 4.23 Hasil *Training loss* dan *Test accuracy* 70:30 dengan 10-*Fold*

n-fold	***Epoch***	***Training loss***	***Test accuracy***
1	1	59,66%	78,26%
	2	45,09%	78,26%
	…	…	…
	9	42,66%	78,26%
	10	42,86%	78,26%
2	1	60,85%	86,96%
	2	47,81%	86,96%

n-fold	*Epoch*	*Training loss*	*Test accuracy*
	…	…	…
	9	43,96%	86,96%
	10	43,94%	86,96%
3	1	63,99%	88,41%
	2	47,84%	88,41%
	…	…	…
	9	44,63%	88,41%
	10	44,65%	88,41%
4	1	61,44%	82,35%
	2	44,26%	82,35%
	…	…	…
	9	43,33%	82,35%
	10	43,49%	82,35%
5	1	61,91%	85,29%
	2	49,61%	85,29%
	…	…	…
	9	43,48%	85,29%
	10	43,71%	85,29%
6	1	62,95%	85,29%
	2	44,39%	85,29%
	…	…	…
	9	43,52%	85,29%
	10	43,67%	85,29%
7	1	60,29%	82,35%
	2	46,78%	82,35%
	…	…	…
	9	43,44%	82,35%
	10	43,26%	82,35%
8	1	60,67%	89,71%
	2	47,79%	89,71%
	…	…	…
	9	44,53%	89,71%
	10	44,76%	89,71%
9	1	64,65%	82,35%
	2	43,55%	82,35%
	…	…	…
	9	43,52%	82,35%
	10	43,40%	82,35%
10	1	63,31%	82,35%
	2	44,29%	82,35%
	…	…	…
	9	43,50%	82,35%
	10	43,43%	82,35%

Hasil rasio 70:30 10-*fold cross validation* mendapatkan hasil perhitungan seperti pada Tabel 4.24. Nilai akurasi yang dihasilkan setiap *fold* tidak stabil. Nilai akurasi mencapai nilai tertinggi yaitu 89,71% pada *fold* ke-8. Sedangkan untuk nilai

akurasi terendah berada pada *fold* ke-1 dengan nilai 78,26%. Rata-rata keseluruhan akurasi dari 10-*fold* rasio 70:30 yaitu sebesar 84,33%.

Tabel 4.24 Hasil 10-*fold Cross Validation* Rasio 70:30

n-fold	***Precision* Positif**	***Precision* Negatif**	***Recall* Positif**	***Recall* Negatif**	***F1-score* Positif**	***F1-score* Negatif**	**Akurasi**	**Rata-Rata Akurasi**
1	78,26%	0%	100%	0%	87,80%	0%	78,26%	84,33%
2	86,96%	0%	100%	0%	93,02%	0%	86,96%	
3	88,41%	0%	100%	0%	93,85%	0%	88,41%	
4	82,35%	0%	100%	0%	90,32%	0%	82,35%	
5	85,29%	0%	100%	0%	92,06%	0%	85,29%	
6	85,29%	0%	100%	0%	92,06%	0%	85,29%	
7	82,35%	0%	100%	0%	90,32%	0%	82,35%	
8	89,71%	0%	100%	0%	94,57%	0%	89,71%	
9	82,35%	0%	100%	0%	90,32%	0%	82,35%	
10	82,35%	0%	100%	0%	90,32%	0%	82,35%	

4.7.2.3 Uji Coba Rasio 80:20

Pada pengujian rasio 80:20 hasil pada Tabel 4.25 menunjukkan penurunan *training loss* di setiap *n-fold*, dengan variasi yang berbeda. Pada *n-fold* 1, *training loss* turun dari 61,53% menjadi 43,65%, dengan *test accuracy* stabil di 83,33%. Di *n-fold* 2, meskipun *training loss* turun dari 60,65% menjadi 43,75%, *test accuracy* tetap stabil di 84,62%. Pada *n-fold* 3, *training loss* berkurang dari 60,19% menjadi 43,96%, sementara *test accuracy* tetap stabil di 87,18%. Di *n-fold* 4, *training loss* turun dari 61,48% menjadi 44,38%, dengan *test accuracy* tetap stabil di 87,18%. Pada *n-fold* 5, *training loss* berkurang dari 58,00% menjadi 43,32%, namun *test accuracy* tetap pada 80,77%. Pada *n-fold* 6, *training loss* turun dari 60,12% menjadi 43,80%, dengan *test accuracy* stabil di 85,90%. Di *n-fold* 7, *training loss* menurun dari 60,25% menjadi 43,40%, dan *test accuracy* tetap stabil di 80,77%. Pada *n-fold* 8, *training loss* turun dari 60,55% menjadi 43,73%, dengan *test accuracy* stabil di 83,33%. Di *n-fold* 9, *training loss* berkurang dari 62,08% menjadi 43,46%,

sementara *test accuracy* tetap di 83,33%. Terakhir, pada *n-fold* 10, *training loss* menurun dari 59,73% menjadi 44,12%, dan *test accuracy* tetap di 87,18%.

Tabel 4.25 Hasil *Training loss* dan *Test accuracy* 80:20 dengan 10-*Fold*

n-fold	***Epoch***	***Training loss***	***Test accuracy***
1	1	61,53%	83,33%
	2	44,69%	83,33%
	…	…	…
	9	43,47%	83,33%
	10	43,65%	83,33%
2	1	60,65%	84,62%
	2	43,56%	84,62%
	…	…	…
	9	43,64%	84,62%
	10	43,75%	84,62%
3	1	60,19%	87,18%
	2	47,05%	87,18%
	…	…	…
	9	44,65%	87,18%
	10	43,96%	87,18%
4	1	61,48%	87,18%
	2	46,28%	87,18%
	…	…	…
	9	43,94%	87,18%
	10	44,38%	87,18%
5	1	58,00%	80,77%
	2	44,59%	80,77%
	…	…	…
	9	43,22%	80,77%
	10	43,32%	80,77%
6	1	60,12%	85,90%
	2	45,54%	85,90%
	…	…	…
	9	44,14%	85,90%
	10	43,80%	85,90%
7	1	60,25%	80,77%
	2	43,46%	80,77%
	…	…	…
	9	42,94%	80,77%
	10	43,40%	80,77%
8	1	60,55%	83,33%
	2	43,95%	83,33%
	…	…	…
	9	43,45%	83,33%
	10	43,73%	83,33%
9	1	62,08%	83,33%
	2	44,77%	83,33%
	…	…	…
	9	43,21%	83,33%
	10	43,46%	83,33%
10	1	59,73%	87,18%
	2	44,45%	87,18%

n-fold	*Epoch*	*Training loss*	*Test accuracy*
	…	…	…
	9	44,22%	87,18%
	10	44,12%	87,18%

Hasil rasio 80:20 10-*fold cross validation* mendapatkan hasil perhitungan seperti pada Tabel 4.26. Nilai akurasi yang dihasilkan setiap *fold* tidak stabil. Nilai akurasi mencapai nilai tertinggi yaitu 87,18% pada *fold* ke 3, 4 dan 10. Sedangkan untuk nilai akurasi terendah berada pada *fold* ke 5 dan 7 dengan nilai 80,77%. Rata-rata keseluruhan akurasi dari 10-*fold* rasio 80:20 yaitu sebesar 84,35%.

Tabel 4.26 Hasil 10-*fold Cross Validation* Rasio 80:20

n-fold	***Precision* Positif**	***Precision* Negatif**	***Recall* Positif**	***Recall* Negatif**	***F1-score* Positif**	***F1-score* Negatif**	**Akurasi**	**Rata-Rata Akurasi**
1	83,33%	0%	100%	0%	90,91%	0%	83,33%	84,35%
2	84,62%	0%	100%	0%	91,67%	0%	84,62%	
3	87,18%	0%	100%	0%	93,15%	0%	87,18%	
4	87,18%	0%	100%	0%	93,15%	0%	87,18%	
5	80,77%	0%	100%	0%	89,36%	0%	80,77%	
6	85,90%	0%	100%	0%	92,41%	0%	85,90%	
7	80,77%	0%	100%	0%	89,36%	0%	80,77%	
8	83,33%	0%	100%	0%	90,91%	0%	83,33%	
9	83,33%	0%	100%	0%	90,91%	0%	83,33%	
10	87,18%	0%	100%	0%	93,15%	0%	87,18%	

4.7.2.4 Uji Coba Rasio 90:10

Pada pertama rasio 90:10 hasil pada Tabel 4.27 menunjukkan penurunan *training loss* di setiap *n-fold*, dengan variasi yang berbeda. Pada *n-fold* 1, *training loss* turun dari 58,41% menjadi 42,51%, dengan *test accuracy* stabil di 77,27%. Di *n-fold* 2, *training loss* menurun dari 58,55% menjadi 43,76%, sementara *test accuracy* tetap stabil di 82,95%. Pada *n-fold* 3, *training loss* berkurang dari 60,97% menjadi 44,03%, dengan *test accuracy* tetap stabil di 87,50%. Di *n-fold* 4, *training loss* turun dari 62,18% menjadi 44,73%, dan *test accuracy* tetap stabil di 89,77%.

Pada *n-fold* 5, *training loss* berkurang dari 60,81% menjadi 44,28%, sementara *test accuracy* tetap di 86,36%. Di *n-fold* 6, *training loss* menurun dari 59,12% menjadi 43,45%, dengan *test accuracy* stabil di 82,95%. Pada *n-fold* 7, *training loss* turun dari 55,68% menjadi 42,67%, dan *test accuracy* tetap stabil di 80,68%. Di *n-fold* 8, *training loss* turun dari 55,94% menjadi 44,64%, dengan *test accuracy* stabil di 89,77%. Pada *n-fold* 9, *training loss* berkurang dari 58,23% menjadi 44,93%, dengan *test accuracy* tetap di 86,21%. Terakhir, pada *n-fold* 10, *training loss* menurun dari 61,12% menjadi 42,74%, dan *test accuracy* tetap di 79,31%.

Tabel 4.27 Hasil *Training loss* dan *Test accuracy* 90:10 dengan 10-*Fold*

n-fold	***Epoch***	***Training loss***	***Test accuracy***
1	1	58,41%	77,27%
	2	43,59%	77,27%
	…	…	…
	9	42,21%	77,27%
	10	42,51%	77,27%
2	1	58,55%	82,95%
	2	44,97%	82,95%
	…	…	…
	9	43,86%	82,95%
	10	43,76%	82,95%
3	1	60,97%	87,50%
	2	45,88%	87,50%
	…	…	…
	9	44,20%	87,50%
	10	44,03%	87,50%
4	1	62,18%	89,77%
	2	46,30%	89,77%
	…	…	…
	9	44,99%	89,77%
	10	44,73%	89,77%
5	1	60,81%	86,36%
	2	44,22%	86,36%
	…	…	…
	9	44,03%	86,36%
	10	44,28%	86,36%
6	1	59,12%	82,95%
	2	45,51%	82,95%
	…	…	…
	9	43,59%	82,95%
	10	43,45%	82,95%
7	1	55,68%	80,68%
	2	43,12%	80,68%
	…	…	…

n-fold	*Epoch*	*Training loss*	*Test accuracy*
	9	43,14%	80,68%
	10	42,67%	80,68%
8	1	55,94%	89,77%
	2	44,71%	89,77%
	…	…	…
	9	45,08%	89,77%
	10	44,64%	89,77%
9	1	58,23%	86,21%
	2	44,72%	86,21%
	…	…	…
	9	43,92%	86,21%
	10	43,93%	86,21%
10	1	61,12%	79,31%
	2	42,90%	79,31%
	…	…	…
	9	42,52%	79,31%
	10	42,74%	79,31%

Hasil rasio 90:10 10-*fold cross validation* mendapatkan hasil perhitungan seperti pada Tabel 4.28. Nilai akurasi yang dihasilkan setiap *fold* tidak stabil. Nilai akurasi mencapai nilai tertinggi yaitu 89,77% pada *fold* ke 4 dan 8. Sedangkan untuk nilai akurasi terendah berada pada *fold* ke-1 dengan nilai 77,27%. Rata-rata keseluruhan akurasi dari 10-*fold* rasio 90:10 yaitu sebesar 84,27%.

Tabel 4.28 Hasil 10-*fold Cross Validation* Rasio 90:10

n-fold	***Precision* Positif**	***Precision* Negatif**	***Recall* Positif**	***Recall* Negatif**	***F1-score* Positif**	***F1-score* Negatif**	**Akurasi**	**Rata-Rata Akurasi**
1	77,27%	0%	100%	0%	87,18%	0%	77,27%	84,27%
2	82,95%	0%	100%	0%	90,68%	0%	82,95%	
3	87,50%	0%	100%	0%	93,33%	0%	87,50%	
4	89,77%	0%	100%	0%	94,61%	0%	89,77%	
5	86,36%	0%	100%	0%	92,68%	0%	86,36%	
6	82,95%	0%	100%	0%	90,68%	0%	82,95%	
7	80,68%	0%	100%	0%	89,31%	0%	80,68%	
8	89,77%	0%	100%	0%	94,61%	0%	89,77%	
9	86,21%	0%	100%	0%	92,59%	0%	86,21%	
10	79,31%	0%	100%	0%	88,46%	0%	79,31%	

4.8 Hasil Evaluasi Model

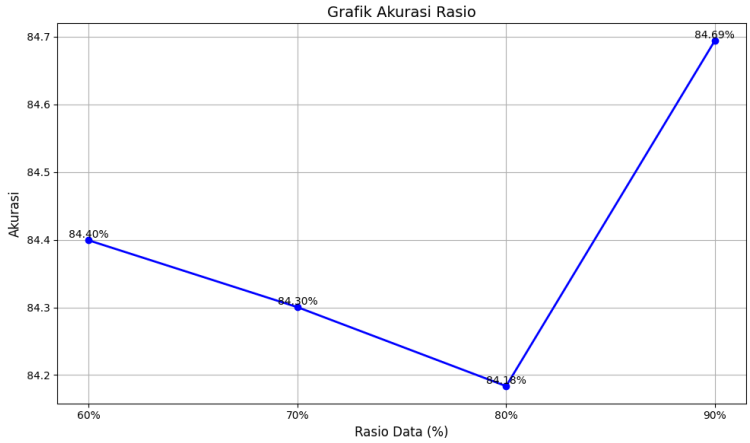
4.8.1 Hasil Evaluasi *Split* Data

Sistem menggunakan metode LSTM dan di evaluasi kinerjanya dengan menggunakan *confusion matrix* untuk mendapatkan akurasi, *precision*, *recall* dan *f1-score*. Dengan membagi data menjadi data *training* dan data *testing* dan pengujian dengan rasio yang berbeda, yaitu 60:40, 70:30, 80:20, dan 90:10. Hasil perbandingan akurasi semua rasio terlihat pada Tabel 4.29.

Tabel 4.29 Perbandingan Akurasi *Split* Data

Rasio	**Akurasi**
60:40	84,40%
70:30	84,30%
80:20	84,18%
90:10	84,69%

Berdasarkan perbandingan pada Tabel 4.29 dan grafik pada Gambar 4.2 menunjukkan bahwa hasil akurasi tertinggi yaitu sebesar 84,69% pada rasio 90:10 dan akurasi terendah pada rasio 80:20 dengan nilai sebesar 84,18%. Dengan ke empat model tersebut menunjukkan bahwa dengan rasio 90:10 yaitu 90% data *training* dan 10% data *testing* mendapatan hasil akurasi yang baik dibanding dengan rasio lainnya. Hasil dari nilai performa *confusion matrix* 90:10 ditunjukkan pada Tabel 4.19.



Gambar 4.2 Grafik Akurasi Rasio

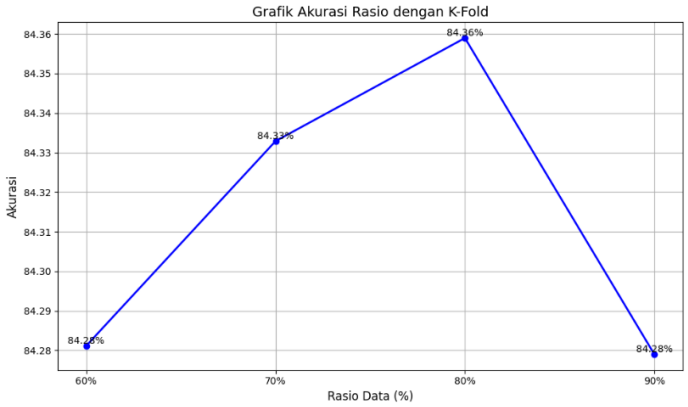
4.8.2 Hasil Evaluasi *Split* Data dengan *K-Fold*

Sistem menggunakan metode LSTM dan di evaluasi kinerjanya dengan menggunakan *confusion matrix* untuk mendapatkan akurasi, *precision*, *recall* dan *f1-score*. Dengan membagi data menjadi data *training* dan data *testing* dan pengujian dengan rasio yang berbeda, yaitu 60:40, 70:30, 80:20, dan 90:10. Menggunakan metode *k-fold cross validation* dengan nilai k 10. Hasil perbandingan akurasi rasio terlihat pada Tabel 4.30.

Tabel 4.30 Hasil Perbandingan Akurasi *Split* Data dengan 10-*Fold*

Rasio	Akurasi
60:40	84,28%
70:30	84,33%
80:20	84,36%
90:10	84,28%

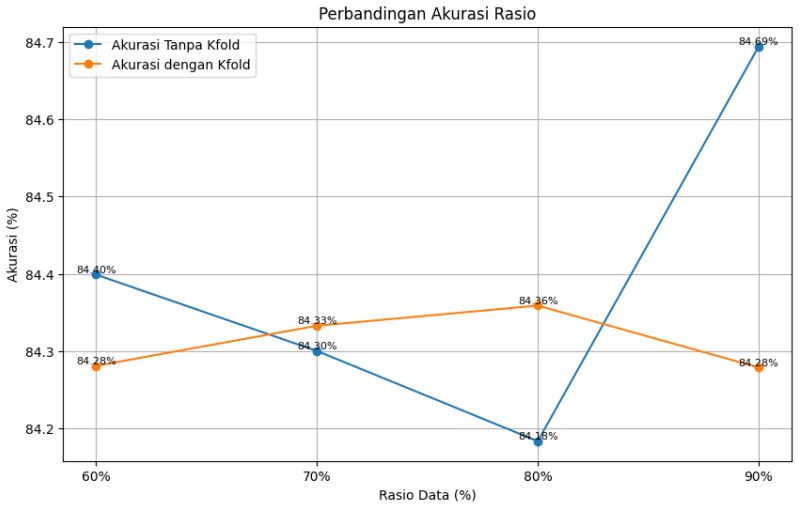
Berdasarkan hasil perbandingan pada Tabel 4.30 dan grafik pada Gambar 4.3 menunjukkan bahwa hasil akurasi tertinggi didapatkan nilai sebesar 84,36% pada rasio 80:20 dan nilai akurasi terendah berada pada rasio 60:40 dan 90:10 dengan nilai 84,28%. Dari hasil ini dengan menggunakan 10-*fold cross validation* dan rasio dengan data *testing* dan data *training* seimbang mendapatkan hasil terbaik. Hasil performa nilai *confusion matrix* 80:20 terlihat pada Tabel 4.26.



Gambar 4.3 Grafik Akurasi Rasio dengan 10-*Fold*

4.9 Pembahasan

Dari hasil penelitian ini menggunakan metode LSTM untuk analisis sentimen ulasan rumah makan Padang Payakumbuah berdasarkan data *review* dari *Google Maps*. Dengan jumlah data sebanyak 977 *dataset*. Yang diberi label positif dan negatif, terdapat 823 data positif dan 153 data negatif yang menunjukkan adanya ketidakseimbangan dalam distribusi sentimen. Kemudian dilakukan *preprocessing text* untuk pembersihan data. Terdapat representasi emoji menjadi teks dan normalisasi kata singkatan menjadi kata dasar atau kata sebenarnya. Selanjutnya pembobotan kata dengan TF-IDF. Kemudian membuat dua model pelatihan menggunakan *split* data dengan model pertama *split* data tanpa *k-fold cross validation* dan model kedua menggunakan *k-fold cross validation*. Terakhir yaitu mengevaluasi dengan menggunakan *confusion matrix* dan menghitung matrix seperti akurasi, *precision*, *recall* dan *f1-score*.



Gambar 4.4 Perbandingan Seluruh Akurasi

Tabel 4.31 Perbandingan Hasil Akurasi Rasio

Rasio	**Akurasi Tanpa *K-fold***	**Akurasi Rata-Rata *K-fold***
60:40	84,40%	84,28%

Rasio	Akurasi Tanpa *K-fold*	Akurasi Rata-Rata *K-fold*
70:30	84,30%	84,33%
80:20	84,18%	84,36%
90:10	84,69%	84,28%

Pada Tabel 4.31 *split* data yang digunakan terdapat empat rasio yaitu 60:40, 70:30, 80:20, dan 90:10. Hasil didapatkan rasio 60:40 tanpa *k-fold* mendapat nilai akurasi sebesar 84,40%, sedangkan menggunakan *k-fold* mendapat nilai lebih rendah sebesar 84,28%. Untuk rasio 70:30, tanpa *k-fold* mendapat nilai akurasi 84,30%, mengalami kenaikan akurasi dengan menggunakan *k-fold* menjadi 84,33%. Rasio 80:20, tanpa *k-fold* mendapat nilai 84,18%, mengalami kenaikan signifikan dengan menggunakan *k-fold* menjadi 84,36%. Terakhir pada rasio 90:10, tanpa *k-fold* mendapatkan nilai sebesar 84,69%, sedangkan menggunakan *k-fold* mengalami penurunan akurasi dengan hanya mendapat nilai sebesar 84,28%.

Grafik pada Gambar 4.4 menunjukkan bahwa nilai akurasi rasio tanpa atau menggunakan k-fold memiliki nilai yang tidak jauh berbeda, bernilai akurasi antara 84,1% sampai 84,7%. Dari hasil pelatihan model, *split* data tanpa *k-fold* yang mendapatkan akurasi terbaik yaitu rasio 90:10 dengan 90% data *training* sebesar 878 data dan 10% data *testing* sebesar 98 data, dengan hasil akurasi 84,69%. Hasil pelatihan model *split* data dengan menggunakan *k-fold* dengan k=10, mendapatkan hasil dengan akurasi terbaik berada pada rasio 80:20 dengan nilai akurasi sebesar 84,36%. Secara keseluruhan hasil akurasi terbaik berada pada *split* data tanpa *k-fold cross validation* rasio 90:10 dengan mencapai nilai akurasi 84,69% dalam analisis sentimen ulasan rumah makan padang Payakumbuah menggunakan metode LSTM. Pada penelitian ini menggunakan *k-fold cross validation* tidak meningkatan akurasi secara signifikan. Hal ini menunjukkan bahwa metode *split* data tanpa *k-fold*

mendapatkan performa yang lebih baik dan signifikan untuk melakukan analisis sentimen.

4.10 Integrasi Sains dan Islam

Penelitian ini mengimplementasikan analisis sentimen berbasis *Long Short-Term Memory* (LSTM) untuk menilai ulasan pelanggan rumah makan Padang Payakumbuah yang diperoleh dari *Google Maps*. Dalam Islam, kejujuran merupakan prinsip yang sangat ditekankan dalam kehidupan sehari-hari, termasuk dalam memberikan ulasan dan melakukan analisis terhadap data. Integrasi antara sains dan Islam dalam penelitian ini dapat dikaitkan dengan ajaran Islam tentang kejujuran dalam berbicara dan menilai sesuatu secara objektif.

Dalam proses analisis sentimen, data yang digunakan harus mencerminkan opini pelanggan secara jujur dan objektif. Ini sesuai dengan QS. *Al-Maidah* ayat 8:

يٰٓاَيُّهَا الَّذِيْنَ اٰمَنُوْا كُوْنُوْا قَوّٰمِيْنَ لِلّٰهِ شُهَدَاۤءَ بِالْقِسْطِۖ وَلَا يَجْرِمَنَّكُمْ شَنَاٰنُ قَوْمٍ عَلٰٓى اَلَّا تَعْدِلُوْا اِعْدِلُوْا هُوَ اَقْرَبُ لِلتَّقْوٰىۖ وَاتَّقُوا اللّٰهَ ۗ اِنَّ اللّٰهَ خَبِيْرٌۢ بِمَا تَعْمَلُوْنَ

"Hai orang-orang yang beriman hendaklah kamu jadi orang-orang yang selalu menegakkan (bersaksi atau jujur tentang kebenaran) karena Allah, menjadi saksi dengan adil. Dan janganlah sekali-kali kebencianmu terhadap suatu kaum, mendorong kamu untuk berlaku tidak adil. Berlaku adillah, karena adil itu lebih dekat kepada takwa. Dan bertakwalah kepada Allah, sesungguhnya Allah Maha Mengetahui apa yang kamu kerjakan," (QS. Al-Maidah: 8).

Ayat ini mengajarkan bahwa dalam menyampaikan informasi, termasuk dalam ulasan pelanggan, tidak boleh ada unsur pemalsuan atau manipulasi data. Dalam penelitian ini, teknik data *preprocessing* seperti pembersihan data, *stopword removal*, dan *stemming* dilakukan untuk memastikan bahwa teks yang dianalisis

adalah murni opini pelanggan tanpa distorsi yang dapat mempengaruhi hasil klasifikasi sentimen.

Kejujuran dalam menilai suatu hal sangat penting agar hasil penelitian tidak dipengaruhi oleh bias atau kepentingan tertentu. Hal ini sejalan dengan QS. *At-Taubah* ayat 119:

يَا أَيُّهَا الَّذِينَ آمَنُوا اتَّقُوا اللَّهَ وَكُونُوا مَعَ الصَّادِقِينَ

"Hai orang-orang yang beriman bertakwalah kepada Allah, dan hendaklah kamu bersama orang-orang yang benar" (Q.S. At-Taubah: 119).

Dalam penelitian ini, kejujuran diterapkan melalui penggunaan metode LSTM, yang bertujuan untuk menilai ulasan pelanggan secara objektif dan adil tanpa pengaruh subjektivitas manusia. Model ini tidak memihak, sehingga hasil analisis sentimen benar-benar mencerminkan opini pelanggan berdasarkan pola dan makna dalam teks ulasan. Dalam Islam, seseorang yang memberikan ulasan atau pendapat harus jujur dan tidak menyesatkan. Dari Abdullah Ibnu Mas'ud, Rasulullah SAW bersabda:

عَلَيْكُمْ بِالصِّدْقِ فَاِنَّ الصِّدْقَ يَهْدِيْ اِلَى الْبِرِّ اِنَّ الْبِرِّيَهْدِيْ اِلَى الْجَنَّةِ (رواه البخارى ومسل)

"Hendaklah kalian berkata jujur, karena kejujuran itu membawa kepada kebaikan, dan kebaikan itu akan membawa kepada surga." (HR. Bukhari & Muslim)

Penelitian ini memastikan bahwa kejujuran dalam ulasan pelanggan dapat diidentifikasi dengan baik, sehingga rumah makan dapat mengambil langkah-langkah perbaikan yang benar-benar berdasarkan masukan pelanggan yang jujur. Dengan demikian, penelitian ini bukan hanya sekadar penerapan teknologi, tetapi juga mendukung nilai kejujuran yang diajarkan dalam Islam.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Penelitian ini mengkaji penggunaan metode *Long Short-Term Memory* (LSTM) dalam analisis sentimen terhadap ulasan pelanggan rumah makan padang Payakumbuah yang diambil melalui platform *Google Maps*. Ulasan tersebut kemudian dikategorikan dalam dua kelas sentimen, yaitu positif dan negatif, dan dianalisis menggunakan teknik pemrosesan bahasa alami yang meliputi tahapan data *cleaning*, *case folding*, *tokenization*, *stopword removal*, *stemming*, normalisasi, dan representasi kata dengan metode *Term Frequency-Inverse Document Frequency* (TF-IDF).

Berdasarkan hasil eksperimen, model LSTM menunjukkan performa yang baik dalam mengklasifikasikan sentimen dengan tingkat akurasi yang optimal pada rasio 90:10 (90% data pelatihan dan 10% data pengujian), dengan hasil akurasi sebesar 84,69%. Hasil ini menunjukkan bahwa semakin banyak data yang digunakan dalam pelatihan, semakin baik performa model dalam menganalisis sentimen. Sementara itu, penggunaan teknik *k-fold cross validation* dengan k=10 tidak memberikan peningkatan yang signifikan pada akurasi dibandingkan dengan model yang hanya menggunakan pembagian data langsung. Hal ini menunjukkan bahwa pembagian data yang lebih besar untuk pelatihan memiliki dampak yang lebih besar dalam meningkatkan kinerja model. Selain itu, penelitian ini juga menunjukkan bahwa proses *preprocessing* yang tepat sangat berpengaruh pada

performa model. Dengan pembersihan data yang teliti dan representasi emoji serta normalisasi kata yang baik, model LSTM mampu memproses teks dengan lebih akurat. Peneliti juga menggunakan evaluasi model dengan metrik-metrik seperti akurasi, *precision*, *recall*, dan *F1-Score* untuk mengukur performa model, dan hasilnya menunjukkan bahwa model LSTM efektif dalam mengklasifikasikan sentimen ulasan.

5.2 Saran

Berdasarkan hasil penelitian, beberapa saran dapat diajukan untuk pengembangan lebih lanjut:

1. Mengingat adanya ketidakseimbangan data antara ulasan positif dan negatif, penelitian selanjutnya dapat mencoba teknik untuk menangani masalah ini, seperti *oversampling* atau *undersampling*, guna memperoleh model yang lebih seimbang dalam pengklasifikasian sentimen.
2. Pengembangan analisis sentimen dengan berbasis aspek dapat menambah hasil analisis sentimen untuk ulasan rumah makan padang payakumbuah lebih terperinci.
3. Meskipun model LSTM memberikan akurasi yang baik, namun penggunaan model *deep learning* lainnya dapat dipertimbangkan untuk meningkatkan akurasi dan kemampuan model dalam menangkap sentimen yang lebih kompleks.

DAFTAR PUSTAKA

Afandi, M. D., Homaidi, A., Ghofur, A., & Zubairi, A. (2024). Penerapan Information Retrieval dalam Sistem Analisis Kemiripan Proposal Skripsi menggunakan Cosine Similarity. *Swabumi*, *12*(1), 39–46. https://doi.org/10.31294/swabumi.v12i1.17087

Ahmad, S., Ridwan, A. M., & Setyawan, G. D. (2023). Analisis Sentimen Product Tools & Home Menggunakan Metode CNN dan LSTM. *Jurnal Teknologi dan Rekayasa Sistem Komputer*, *6*(2), 133–140. https://doi.org/10.31943/teknokom.v6i2.154

Al-Sheikh, bdullah bin M. bin A. (2004). *Tafsir Ibnu Katsir Jilid 6*.

Amalsyah, M. R., Kurniawan, D., Rifai, A., & Sari, P. (2025). Sentiment Analysis of Fintech Application User Reviews Using the CRISP-DM Framework for Product Development Prioritization. *Jurnal Sistem Informasi*, *14*(2), 813. https://doi.org/10.32520/stmsi.v14i2.5064

Amrustian, M. A., Widayat, W., & Wirawan, A. M. (2022). Analisis Sentimen Evaluasi terhadap Pengajaran Dosen di Perguruan Tinggi menggunakan Metode LSTM. *Jurnal Media Informatika Budidarma*, *6*(1), 535. https://doi.org/10.30865/mib.v6i1.3527

Anam, M. K., Putra, P. P., Malik, R. A., Karfindo, Putra, T. A., Elva, Y., Mahessya, R. A., Firdaus, uhammad B., Ikhsan, & Gunawan, C. R. (2025). Enhancing the Performance of Machine Learning Algorithm for Intent Sentiment Analysis on Village Fund Topic. *Journal of Applied Data Sciences*, *6*(2), 1102–1115. https://doi.org/10.47738/jads.v6i2.637

Cahyani, J., Mujahidin, S., & Fiqar, T. P. (2023). Implementasi Metode Long Short Term Memory (LSTM) untuk Memprediksi Harga Bahan Pokok Nasional. *Jurnal Sistem dan Teknologi Informasi*, *11*(2), 346. https://doi.org/10.26418/justin.v11i2.57395

Deviyanto, A., & Wahyudi, M. D. R. (2018). Penerapan Analisis Sentimen pada Pengguna Twitter menggunakan Metode K-Nearest Neighbor. *Jurnal Informatika Sunan Kalijaga*, *3*(1), 1. https://doi.org/10.14421/jiska.2018.31-01

Dewi, M. N., & Putra, R. E. (2024). Pengembangan Sistem Analisis Sentimen Masyarakat terhadap Chatgpt pada Twitter dengan Perbandingan Metode Naive Bayes Classifier dan K-Nearest Neighbors. *Journal of Informatics and Computer Science*, *5*(03), 344–357. https://doi.org/10.26740/jinacs.v5n03.p344-357

Fitrana, L. A., Linawati, S., Herlinawati, N., Sa'adah, R., & Seimahuria, S. (2024). Analisis Sentimen Pengguna Twitter terhadap Brand Indosat Menggunakan Metode Naïve Bayes Classifier. *Jurnal Mahasiswa Teknik Informatika*, *8*(3), 4291–4297. https://doi.org/10.36040/jati.v8i3.9866

Gunawan, N. D., Anggraeni, F., & Citradewi, A. (2023). Implementasi Rasio Profitabilitas sebagai Media Analisis Perubahan Laba pada PT Alam Sutera Realty Tbk. *Jurnal Ekonomi Dan Bisnis*, *15*(2), 70–77. https://doi.org/10.55049/jeb.v15i2.225

Haddi, E., Liu, X., & Shi, Y. (2013). The Role of Text Pre-processing in Sentiment Analysis. *Procedia Computer Science*, *17*, 26–32. https://doi.org/10.1016/j.procs.2013.05.005

Hochreiter, S., & Schmidhuber, J. (1997). Long Short-Term Memory. *Neural Computation*, *9*(8), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735

Irawan, R. N., Hindrayani, K. M., & Idhom, M. (2024). Penerapan Cross Validation sebagai Analisis Sentimen Pelayanan Publik Kereta Api Lokal Daop 8 menggunakan Metode Multinomial Naïve Bayes. *G-Tech: Jurnal Teknologi Terapan*, *8*(2), 954–963. https://doi.org/10.33379/gtech.v8i2.4117

Isnain, A. R., Sulistiani, H., Hurohman, B. M., Nurkholis, A., & Styawati, S. (2022). Analisis Perbandingan Algoritma LSTM dan Naive Bayes untuk Analisis Sentimen. *Jurnal Edukasi dan Penelitian Informatika*, *8*(2), 299. https://doi.org/10.26418/jp.v8i2.54704

Jurianto, I. P. (2023). *Padang Payakumbuah: Rumah Makan Padang Hits Milik Influencer Beken Arief Muhammad*. Wisata App. https://wisata.app/diary/lokasi-menu-restoran-padang-payakumbuah-arief-muhammad

Kosasih, R., & Alberto, A. (2021). Analisis Sentimen Produk Permainan menggunakan Metode TF-IDF dan Algoritma K-Nearest Neighbor. *Jurnal Nasional Informatika dan Teknologi Jaringan*, *6*(1). https://doi.org/10.30743/infotekjar.v6i1.3893

Le, X.-H., Ho, H. V., Lee, G., & Jung, S. (2019). Application of Long Short-Term Memory (LSTM) Neural Network for Flood Forecasting. *Water*, *11*(7), 1387. https://doi.org/10.3390/w11071387

Ma'ady, M. N. P., Rizaldy, D. D., Satria, R. F., & Anaking, P. (2023). Sparring: Sistem Rekomendasi Peneliti Terintegrasi Google Scholar Via Serpapi dan Latent Dirichlet Allocation pada Konteks Perguruan Tinggi. *Jurnal Teknologi dan Manajemen Informatika*, *9*(2), 161–171. https://doi.org/10.26905/jtmi.v9i2.11111

Maharani, C. A., Warsito, B., & Santoso, R. (2024). Analisis Sentimen Vaksin Covid-19 pada Twitter menggunakan Recurrent Neural Network (RNN) dengan Algoritma Long Short-Term Memory (LSTM). *Jurnal Gaussian*, *12*(3), 403–413. https://doi.org/10.14710/j.gauss.12.3.403-413

Meriani, A. P., & Rahmatulloh, A. (2024). Perbandingan Gated Recurrent Unit (GRU) dan Algoritma Long Short Term Memory (LSTM) Linear Refression dalam Prediksi Harga Emas menggunakan Model Time Series. *Jurnal Informatika dan Teknik Elektro Terapan*, *12*(1). https://doi.org/10.23960/jitet.v12i1.3808

Muthia, D. A. (2018). Komparasi Algoritma Klasifikasi Text Mining untuk Analisis Sentimen pada Review Restoran. *Jurnal Pilar Nusa Mandiri*, *14*(1), 69–74. https://doi.org/10.33480/pilar.v14i1.92

Mutmatinah, S., Khairunnas, & Khairunnisa. (2024). Metode Deep Learning LSTM dalam Analisis Sentimen Aplikasi PeduliLindungi. *Scientific Journal of Computers Sciences and Informatics*, *1*(1). https://doi.org/10.34304/scientific.v1i1.231

Novaković, J. D., Veljovic, A., Ilić, S., Papic, Ž. M., & Milica, T. (2017). Evaluation of Classification Models in Machine Learning. *Theory and Applications of Mathematics & Computer Science*, *7*(1), 39–46.

Nurpandi, F., Sulaeman, F. S., & Hermawan, A. (2024). Analisis Sentimen Terhadap Kinerja Kepolisian Indonesia Menggunakan Metode Multinomial Naive Bayes, Long Short-Term Memory, dan Lexicon-Based. *Media Jurnal Informatika*, *16*(1), 1. https://doi.org/10.35194/mji.v16i1.4165

Qaisar, S. M. (2020). Sentiment Analysis of IMDb Movie Reviews Using Long Short Term Memory. *2020 2nd International Conference on Computer and Information Sciences (ICCIS)*, 1–4. https://doi.org/10.1109/ICCIS49240.2020.9257657

Riyani, A., Naf'an, M. Z., & Burhanuddin, A. (2019). Penerapan Cosine Similarity dan Pembobotan TF-IDF untuk Mendeteksi Kemiripan Dokumen. *Indonesia Association of Computational Linguistics*, *2*(1), 23–27. https://doi.org/10.26418/jlk.v2i1.17

Rizqilla, M., Nurani, D., Rahmi, A. N., & Maemunah, M. (2025). Analisis Sentimen Kinerja Jokowi menggunakan Motode Naïve Bayes. *Riset dan E-Jurnal Manajemen Informatika Komputer*, *9*(2), 550–561. https://doi.org/10.33395/remik.v9i2.14670

Romadhoni, Y., & Holle, K. F. H. (2022). Analisis Sentimen terhadap PERMENDIKBUD No.30 pada Media Sosial Twitter menggunakan

Metode Naive Bayes dan LSTM. *Jurnal Informatika: Jurnal Pengembangan IT*, *7*(2), 118–124. https://doi.org/10.30591/jpit.v7i2.3191

Sartika, Kamaruzzaman, Putri, R. Z., Khairunnisa, Yunita, I., Dariatunfuaddah, D. A., Decky, Ardiansyah, T., Maulana, M. F., & Dharmasetiawan. (2024). Pengaruh Harga Nasi Padang Bungo Lawang terhadap Daya Tarik Pembeli. *Jurnal Analisis Manajemen*, *10*(1). https://doi.org/10.32520/jam.v10i1.3683

Sugiyono. (2013). *Metode Penelitian Kuantitatif, Kualitatif dan R&D* (19th ed.). Alfabeta.

Tharwat, A. (2021). Classification Assessment Methods. *Applied Computing and Informatics*, *17*(1), 168–192. https://doi.org/10.1016/j.aci.2018.08.003

Tul, Q., Ali, M., Riaz, A., Noureen, A., Kamranz, M., Hayat, B., & Rehman, A. (2017). Sentiment Analysis Using Deep Learning Techniques: A Review. *International Journal of Advanced Computer Science and Applications*, *8*(6). https://doi.org/10.14569/IJACSA.2017.080657

Vidya Chandradev, I Made Agus Dwi Suarjaya, & I Putu Agung Bayupati. (2023). Analisis Sentimen Review Hotel menggunakan Metode Deep Learning BERT: Analisis Sentimen Review Hotel menggunakan Metode Deep Learning BERT. *Jurnal Buana Informatika*, *14*(02), 107–116. https://doi.org/10.24002/jbi.v14i02.7244

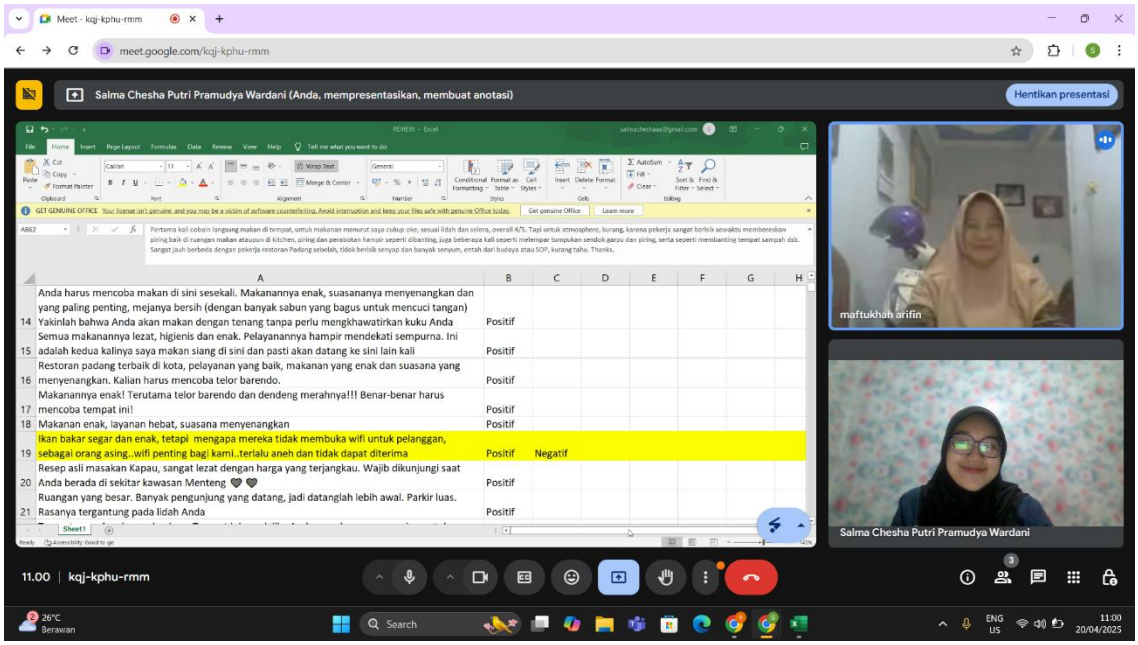
Widayat, W. (2021). Analisis Sentimen Movie Review menggunakan Word2Vec dan Metode LSTM Deep Learning. *Jurnal Media Informatika Budidarma*, *5*(3), 1018. https://doi.org/10.30865/mib.v5i3.3111

Wijayanto, H., Nugroho, D., & Santoso, B. A. (2020). Sistem Informasi Geografis Penentuan Rute Terpendek Lokasi Villa menggunakan Algoritma Floyd Warshall. *Jurnal Teknologi Informasi dan Komunikasi*, *8*(1), 29–36. https://doi.org/10.30646/tikomsin.v8i1.474

Wijiyanto, W., Pradana, A. I., Sopingi, S., & Atina, V. (2024). Teknik K-Fold Cross Validation untuk Mengevaluasi Kinerja Mahasiswa. *Jurnal Algoritma*, *21*(1). https://doi.org/10.33364/algoritma/v.21-1.1618

LAMPIRAN

Lampiran 1 – Bukti Validasi Data Bersama Validator



Lampiran 2 - Tabel *Training loss* dan *Test accuracy* 60:40

n-fold	***Epoch***	***Training loss***	***Test accuracy***
1	1	66,41%	89,83%
	2	48,17%	89,83%
	3	44,98%	89,83%
	4	44,95%	89,83%
	5	44,98%	89,83%
	6	44,46%	89,83%
	7	44,49%	89,83%
	8	45,31%	89,83%
	9	44,95%	89,83%
	10	45,21%	89,83%
2	1	65,01%	88,14%
	2	45,49%	88,14%
	3	44,86%	88,14%
	4	44,62%	88,14%
	5	44,45%	88,14%
	6	45,07%	88,14%
	7	44,42%	88,14%
	8	44,50%	88,14%
	9	45,06%	88,14%
	10	44,31%	88,14%
3	1	64,12%	84,75%
	2	44,59%	84,75%
	3	45,63%	84,75%
	4	44,77%	84,75%
	5	44,07%	84,75%
	6	44,68%	84,75%

n-fold	*Epoch*	*Training loss*	*Test accuracy*
	7	43,38%	84,75%
	8	43,76%	84,75%
	9	43,89%	84,75%
	10	43,63%	84,75%
4	1	65,07%	77,97%
	2	43,73%	77,97%
	3	43,09%	77,97%
	4	42,65%	77,97%
	5	42,26%	77,97%
	6	42,73%	77,97%
	7	42,25%	77,97%
	8	42,58%	77,97%
	9	42,76%	77,97%
	10	42,42%	77,97%
5	1	63,56%	76,27%
	2	44,70%	76,27%
	3	42,21%	76,27%
	4	42,39%	76,27%
	5	42,66%	76,27%
	6	42,08%	76,27%
	7	41,62%	76,27%
	8	42,00%	76,27%
	9	42,49%	76,27%
	10	91,90%	76,27%
6	1	62,99%	79,31%
	2	45,87%	79,31%
	3	43,70%	79,31%
	4	42,91%	79,31%
	5	42,38%	79,31%
	6	42,97%	79,31%
	7	42,78%	79,31%
	8	42,41%	79,31%
	9	42,53%	79,31%
	10	42,71%	79,31%
7	1	66,77%	87,93%
	2	48,31%	87,93%
	3	44,77%	87,93%
	4	45,11%	87,93%
	5	44,74%	87,93%
	6	44,53%	87,93%
	7	44,23%	87,93%
	8	44,18%	87,93%
	9	44,88%	87,93%
	10	44,59%	87,93%
8	1	65,80%	86,21%
	2	46,10%	86,21%
	3	44,83%	86,21%
	4	43,87%	86,21%
	5	44,46%	86,21%
	6	43,65%	86,21%
	7	44,22%	86,21%
	8	44,50%	86,21%

n-fold	*Epoch*	*Training loss*	*Test accuracy*
	9	44,17%	86,21%
	10	44,09%	86,21%
9	1	62,99%	85,42%
	2	46,93%	85,42%
	3	44,24%	85,42%
	4	44,70%	85,42%
	5	44,18%	85,42%
	6	44,35%	85,42%
	7	43,945	85,42%
	8	44,03%	85,42%
	9	44,05%	85,42%
	10	44,28%	85,42%
10	1	63,90%	79,17%
	2	45,37%	79,17%
	3	45,06%	79,17%
	4	44,55%	79,17%
	5	44,32%	79,17%
	6	43,80%	79,17%
	7	44,10%	79,17%
	8	43,94%	79,17%
	9	44,15%	79,17%
	10	44,15%	79,17%

Lampiran 3 - Tabel *Training loss* dan *Test accuracy* 70:30

n-fold	*Epoch*	*Training loss*	*Test accuracy*
1	1	59,66%	78,26%
	2	45,09%	78,26%
	3	42,54%	78,26%
	4	42,79%	78,26%
	5	43,06%	78,26%
	6	42,72%	78,26%
	7	43,30%	78,26%
	8	42,88%	78,26%
	9	42,66%	78,26%
	10	42,86%	78,26%
2	1	60,85%	86,96%
	2	47,81%	86,96%
	3	44,70%	86,96%
	4	44,33%	86,96%
	5	44,69%	86,96%
	6	44,35%	86,96%
	7	44,47%	86,96%
	8	44,24%	86,96%
	9	43,96%	86,96%
	10	43,94%	86,96%
3	1	63,99%	88,41%
	2	47,84%	88,41%
	3	45,14%	88,41%
	4	44,96%	88,41%

n-fold	*Epoch*	*Training loss*	*Test accuracy*
	5	44,31%	88,41%
	6	44,81%	88,41%
	7	44,48%	88,41%
	8	44,24%	88,41%
	9	44,63%	88,41%
	10	44,65%	88,41%
4	1	61,44%	82,35%
	2	44,26%	82,35%
	3	43,25%	82,35%
	4	43,41%	82,35%
	5	43,71%	82,35%
	6	43,31%	82,35%
	7	43,05%	82,35%
	8	42,98%	82,35%
	9	43,33%	82,35%
	10	43,49%	82,35%
5	1	61,91%	85,29%
	2	49,61%	85,29%
	3	43,48%	85,29%
	4	44,28%	85,29%
	5	43,84%	85,29%
	6	44,16%	85,29%
	7	44,18%	85,29%
	8	43,86%	85,29%
	9	43,48%	85,29%
	10	43,71%	85,29%
6	1	62,95%	85,29%
	2	44,39%	85,29%
	3	45,03%	85,29%
	4	43,85%	85,29%
	5	44,21%	85,29%
	6	43,82%	85,29%
	7	43,92%	85,29%
	8	43,82%	85,29%
	9	43,52%	85,29%
	10	43,67%	85,29%
7	1	60,29%	82,35%
	2	46,78%	82,35%
	3	43,47%	82,35%
	4	43,50%	82,35%
	5	43,06%	82,35%
	6	42,78%	82,35%
	7	43,43%	82,35%
	8	43,02%	82,35%
	9	43,44%	82,35%
	10	43,26%	82,35%
8	1	60,67%	89,71%
	2	47,79%	89,71%
	3	45,32%	89,71%
	4	44,77%	89,71%
	5	45,18%	89,71%
	6	44,98%	89,71%

n-fold	*Epoch*	*Training loss*	*Test accuracy*
	7	44,77%	89,71%
	8	44,71%	89,71%
	9	44,53%	89,71%
	10	44,76%	89,71%
9	1	64,65%	82,35%
	2	43,55%	82,35%
	3	43,58%	82,35%
	4	43,82%	82,35%
	5	43,42%	82,35%
	6	43,36%	82,35%
	7	43,29%	82,35%
	8	43,53%	82,35%
	9	43,52%	82,35%
	10	43,40%	82,35%
10	1	63,31%	82,35%
	2	44,29%	82,35%
	3	43,69%	82,35%
	4	43,54%	82,35%
	5	43,74%	82,35%
	6	43,33%	82,35%
	7	43,45%	82,35%
	8	43,17%	82,35%
	9	43,50%	82,35%
	10	43,43%	82,35%

Lampiran 4 - Tabel *Training loss* dan *Test accuracy* 80:20

n-fold	*Epoch*	*Training loss*	*Test accuracy*
1	1	61,53%	83,33%
	2	44,69%	83,33%
	3	44,13%	83,33%
	4	43,76%	83,33%
	5	43,59%	83,33%
	6	43,64%	83,33%
	7	43,575	83,33%
	8	42,64%	83,33%
	9	43,47%	83,33%
	10	43,65%	83,33%
2	1	60,65%	84,62%
	2	43,56%	84,62%
	3	43,53%	84,62%
	4	43,96%	84,62%
	5	44,07%	84,62%
	6	43,81%	84,62%
	7	43,66%	84,62%
	8	43,51%	84,62%
	9	43,64%	84,62%
	10	43,75%	84,62%
3	1	60,19%	87,18%
	2	47,05%	87,18%
	3	44,34%	87,18%

n-fold	*Epoch*	*Training loss*	*Test accuracy*
	4	44,84%	87,18%
	5	44,33%	87,18%
	6	43,89%	87,18%
	7	44,24%	87,18%
	8	44,26%	87,18%
	9	44,65%	87,18%
	10	43,96%	87,18%
4	1	61,48%	87,18%
	2	46,28%	87,18%
	3	44,26%	87,18%
	4	44,07%	87,18%
	5	44,55%	87,18%
	6	44,56%	87,18%
	7	43,92%	87,18%
	8	44,21%	87,18%
	9	43,94%	87,18%
	10	44,38%	87,18%
5	1	58,00%	80,77%
	2	44,59%	80,77%
	3	43,99%	80,77%
	4	42,90%	80,77%
	5	42,73%	80,77%
	6	43,71%	80,77%
	7	43,02%	80,77%
	8	43,05%	80,77%
	9	43,22%	80,77%
	10	43,32%	80,77%
6	1	60,12%	85,90%
	2	45,54%	85,90%
	3	43,88%	85,90%
	4	44,29%	85,90%
	5	43,94%	85,90%
	6	43,87%	85,90%
	7	43,82%	85,90%
	8	43,60%	85,90%
	9	44,14%	85,90%
	10	43,80%	85,90%
7	1	60,25%	80,77%
	2	43,46%	80,77%
	3	43,11%	80,77%
	4	42,76%	80,77%
	5	42,81%	80,77%
	6	42,88%	80,77%
	7	42,82%	80,77%
	8	43,09%	80,77%
	9	42,94%	80,77%
	10	43,40%	80,77%
8	1	60,55%	83,33%
	2	43,95%	83,33%
	3	43,95%	83,33%
	4	43,23%	83,33%
	5	43,31%	83,33%

n-fold	*Epoch*	*Training loss*	*Test accuracy*
	6	43,07%	83,33%
	7	43,04%	83,33%
	8	43,58%	83,33%
	9	43,45%	83,33%
	10	43,73%	83,33%
9	1	62,08%	83,33%
	2	44,77%	83,33%
	3	44,06%	83,33%
	4	43,39%	83,33%
	5	43,38%	83,33%
	6	43,43%	83,33%
	7	43,18%	83,33%
	8	43,57%	83,33%
	9	43,21%	83,33%
	10	43,46%	83,33%
10	1	59,73%	87,18%
	2	44,45%	87,18%
	3	45,21%	87,18%
	4	44,47%	87,18%
	5	44,06%	87,18%
	6	43,95%	87,18%
	7	44,26%	87,18%
	8	44,32%	87,18%
	9	44,22%	87,18%
	10	44,12%	87,18%

Lampiran 5 - Tabel *Training loss* dan *Test accuracy* 90:10

n-fold	*Epoch*	*Training loss*	*Test accuracy*
1	1	58,41%	77,27%
	2	43,59%	77,27%
	3	42,54%	77,27%
	4	42,54%	77,27%
	5	42,35%	77,27%
	6	42,58%	77,27%
	7	42,53%	77,27%
	8	42,28%	77,27%
	9	42,21%	77,27%
	10	42,51%	77,27%
2	1	58,55%	82,95%
	2	44,97%	82,95%
	3	43,89%	82,95%
	4	43,18%	82,95%
	5	43,63%	82,95%
	6	43,51%	82,95%
	7	43,49%	82,95%
	8	43,20%	82,95%
	9	43,86%	82,95%
	10	43,76%	82,95%
3	1	60,97%	87,50%
	2	45,88%	87,50%
	3	44,27%	87,50%

	4	44,53%	87,50%
	5	44,64%	87,50%
	6	44,60%	87,50%
	7	44,66%	87,50%
	8	44,53%	87,50%
	9	44,20%	87,50%
	10	44,03%	87,50%
4	1	62,18%	89,77%
	2	46,30%	89,77%
	3	45,31%	89,77%
	4	45,35%	89,77%
	5	44,68%	89,77%
	6	44,86%	89,77%
	7	44,91%	89,77%
	8	45,20%	89,77%
	9	44,99%	89,77%
	10	44,73%	89,77%
5	1	60,81%	86,36%
	2	44,22%	86,36%
	3	44,39%	86,36%
	4	43,81%	86,36%
	5	44,54%	86,36%
	6	43,91%	86,36%
	7	44,59%	86,36%
	8	44,04%	86,36%
	9	44,03%	86,36%
	10	44,28%	86,36%
6	1	59,12%	82,95%
	2	45,51%	82,95%
	3	43,63%	82,95%
	4	43,71%	82,95%
	5	43,61%	82,95%
	6	44,29%	82,95%
	7	43,95%	82,95%
	8	43,69%	82,95%
	9	43,59%	82,95%
	10	43,45%	82,95%
7	1	55,68%	80,68%
	2	43,12%	80,68%
	3	43,48%	80,68%
	4	43,33%	80,68%
	5	42,74%	80,68%
	6	42,85%	80,68%
	7	43,19%	80,68%
	8	42,59%	80,68%
	9	43,14%	80,68%
	10	42,67%	80,68%
8	1	55,94%	89,77%
	2	44,71%	89,77%
	3	45,36%	89,77%
	4	44,95%	89,77%
	5	45,13%	89,77%
	6	44,88%	89,77%

	7	44,90%	89,77%
	8	44,71R	89,77%
	9	45,08%	89,77%
	10	44,64%	89,77%
9	1	58,23%	86,21%
	2	44,72%	86,21%
	3	44,38%	86,21%
	4	43,99%	86,21%
	5	44,05%	86,21%
	6	44,04%	86,21%
	7	44,14%	86,21%
	8	44,07%	86,21%
	9	43,92%	86,21%
	10	43,93%	86,21%
10	1	61,12%	79,31%
	2	42,90%	79,31%
	3	42,88%	79,31%
	4	42,95%	79,31%
	5	42,86%	79,31%
	6	43,02%	79,31%
	7	42,83%	79,31%
	8	43,26%	79,31%
	9	42,52%	79,31%
	10	42,74%	79,31%