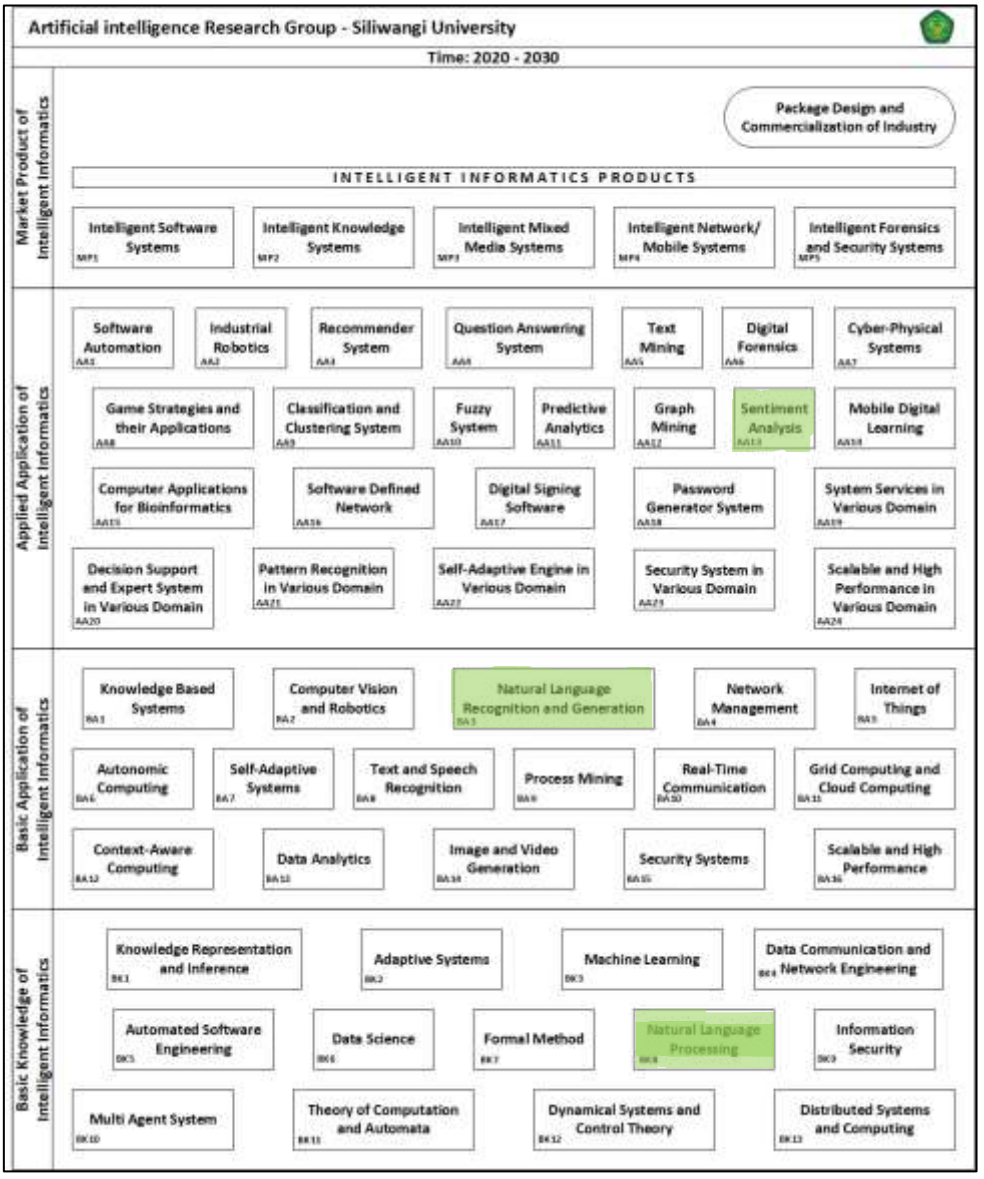
BAB III
METODE PENELITIAN

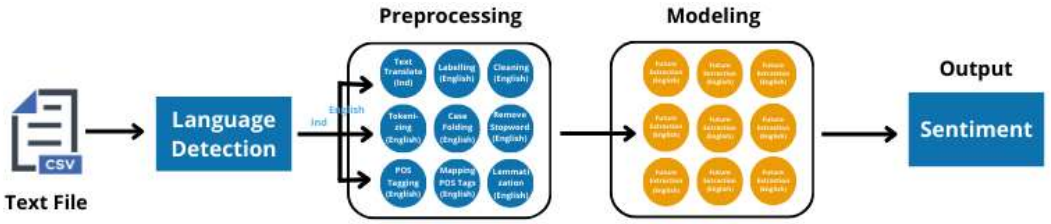
3.1 Peta Jalan Penelitian

Topik penelitian yang diangkat dalam penelitian ini sejalan dengan Peta Jalan Kelompok Keahlian Informatika dan Sistem Inteligen (ISI). Berikut merupakan peta jalan penelitian seperti yang telah digambarkan pada Gambar 3.1.



Gambar 3.1 Peta Jalan Penelitian (AIS Universitas Siliwangi, 2024)

Pada Gambar 3.1 dapat dilihat bahwa peta jalan pada penelitian tersebut dimulai dengan pemahaman tentang konsep dasar dalam bidang informasi cerdas, khususnya terkait dengan analisis sentimen. Langkah pertama adalah memperoleh pengetahuan dasar tentang *Natural Language Processing* (NLP). Berikut adalah gambaran proses NLP yang digunakan pada Gambar 3.2.



Gambar 3.2 Classical NLP

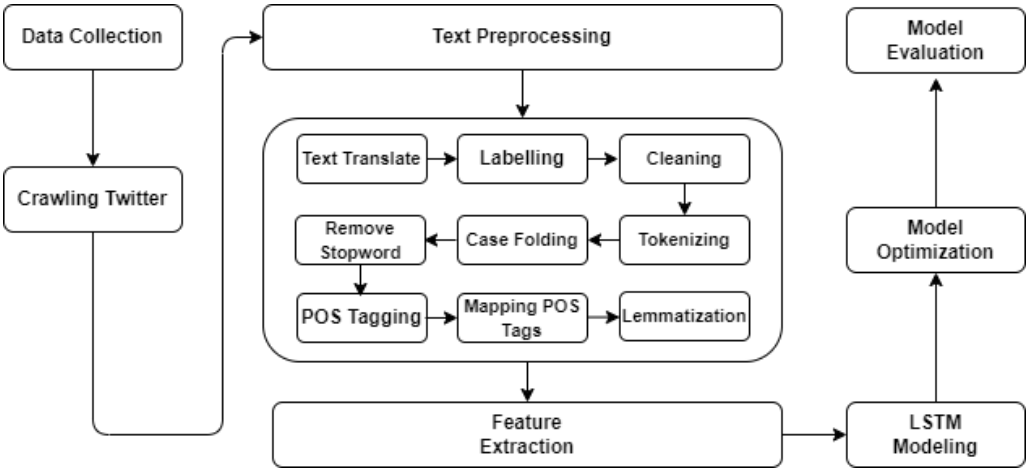
Gambar 3.2 menjelaskan proses analisis sentimen yang berkaitan dengan NLP menggunakan data teks. Proses ini dimulai dengan file teks dalam format CSV yang kemudian melalui tahap deteksi bahasa untuk menentukan bahasa teks tersebut. Setelah itu, data teks diproses melalui beberapa langkah prapemrosesan, termasuk tokenisasi, *case folding*, *remove stopword* dan *lemmatization*. Setelah tahap pra-pemrosesan selesai, data teks kemudian dimodelkan menggunakan algoritma yang telah ditentukan. Hasil akhir dari proses ini adalah keluaran berupa sentimen yang dianalisis dari data teks tersebut. Diagram tersebut memberikan gambaran yang jelas dan terstruktur tentang langkah-langkah yang terlibat dalam analisis sentimen teks.

Langkah berikutnya adalah menerapkan konsep-konsep dasar tersebut dalam konteks aplikasi nyata. Fokusnya adalah pada pengenalan dan pembangkitan bahasa alami (*Natural Language Recognition and Generation*). Dalam penelitian ini, perlu dikembangkan model LSTM yang mampu mengenali dan menghasilkan teks dalam bahasa alami dari data yang diperoleh dari *Twitter*.

Langkah selanjutnya adalah menerapkan aplikasi terapan dari informasi cerdas, khususnya dalam analisis sentimen. Dalam penelitian ini melibatkan penggunaan model LSTM yang dikembangkan untuk menganalisis sentimen terkait dengan pemilihan presiden Indonesia 2024 berdasarkan data dari *Twitter*.

3.2 Tahapan Penelitian

Capaian pada penelitian ini diharapkan dapat menciptakan suatu model LSTM yang mampu bekerja lebih efesien dengan nilai akurasi yang lebih tinggi pada kasus analisis sentimen pemilu presiden 2024. Dengan demikian dilakukan penggunaan *feature extraction* dengan *Word2Vec* dan juga penggunaan *model optimization* agar didapatkan hasil yang diinginkan. Sehingga diciptakan tahapan penelitian yang mengutip pengembangan metode LSTM pada tahapan yang telah dilakukan oleh penelitian sebelumnya, penelitian tersebut antara lain oleh (Alfauzi & Maharani, 2023) dan (Puad et al., 2023). Maka dari itu, berikut merupakan rangkaian tahapan penelitian yang telah dikembangkan dan akan dilakukan oleh peneliti seperti pada Gambar 3.3.



Gambar 3.3 Tahapan Penelitian

Berdasarkan tahapan penelitan pada Gambar 3.3 berikut adalah penjelasan lebih lengkapnya :

3.2.1 Data Collection

Proses Pengumpulan Data merupakan tahap krusial yang memungkinkan peneliti untuk memperoleh dataset yang relevan dan representatif untuk analisis sentimen terhadap Pemilu Presiden Indonesia 2024 melalui Aplikasi X (*Twitter*). Untuk mengakses data publik di *Twitter*, diperlukan token otentikasi Twitter, yang diperoleh melalui proses otentikasi dengan *Twitter API*. Setelah token diperoleh, kemudian melakukan *crawling* data menggunakan *tweet-harvest*, hal tersebut

memungkinkan penentuan kata kunci pencarian dan jumlah data yang akan diambil. Proses pengambilan data dilakukan melalui permintaan *API Twitter* menggunakan token otentikasi yang telah diberikan, dan hasilnya disimpan dalam format file CSV untuk memudahkan pengolahan dan analisis lebih lanjut. Nantinya setelah melewati beberapa proses akan dilakukan *splitting* data menjadi dua bagian, yaitu data *training* dan data *testing*. Data *training* akan mencakup 80% dari total dataset, sedangkan data *testing* akan mencakup 20% sisanya. Hal ini bertujuan untuk memastikan bahwa model yang dikembangkan dapat mengeneralisasi dengan baik pada data yang belum pernah dilihat sebelumnya. Tahapan *data collection* ini menjadi landasan utama dalam penelitian selain untuk mengumpulkan dataset pengumpulan studi literatur juga dilakukan dalam tahap ini.

3.2.2 Text Preprocessing

Text Preprocessing merupakan salah satu langkah penting dalam mempersiapkan data teks untuk analisis lebih lanjut. *Text Preprocessing* bertujuan untuk membersihkan dan mempersiapkan teks sehingga dapat diolah dengan lebih baik dalam analisis sentiment (Mantik et al., 2021). Tahapan *text preprocessing* yang akan dilakukan antara lain :

3.2.2.1 Text Translate

Text translate diperlukan untuk mengubah data dari bahasa Indonesia ke dalam bahasa Inggris menggunakan library *googletrans*. Hal ini dilakukan agar pelabelan menggunakan *vader lexicon* lebih maksimal.

3.2.2.2 Labelling

Dalam penelitian ini Proses *labelling* menggunakan *Vader Lexicon* dimulai dengan memuat data teks yang akan dianalisis. Setelah itu, dilakukan analisis sentimen pada setiap teks menggunakan *SentimentIntensityAnalyzer* dari *NLTK's Vader lexicon* untuk menghasilkan nilai positif, negatif, netral, dan komposit. Selanjutnya, berdasarkan nilai komposit yang dihasilkan, setiap teks diberi label sentimen positif jika nilai compoundnya lebih besar dari atau sama dengan 0.05, label sentimen negatif jika nilainya kurang dari atau sama dengan -0.05, dan label netral jika nilainya berada di antara kisaran tersebut. Hasil *labelling* sentimen tersebut kemudian dimasukkan ke dalam *DataFrame* untuk analisis lebih lanjut.

3.2.2.3 Cleaning

Pada tahap ini, teks mentah dibersihkan dari informasi yang tidak relevan atau mengganggu. Termasuk menghapus URL, HTML, emoji dan simbol yang tidak diperlukan. Tujuannya adalah untuk mempersiapkan teks agar lebih mudah diproses tanpa gangguan dari informasi yang tidak relevan.

3.2.2.4 Tokenizing

Tokenizing adalah proses membagi teks menjadi bagian-bagian yang lebih kecil yang disebut token. Token ini bisa berupa kata, frasa, atau karakter terpisah. Tujuannya adalah untuk memecah teks menjadi unit-unit yang lebih mudah diolah. Misalnya, memisahkan kalimat menjadi kata-kata atau memisahkan kata- kata dari tanda baca.

3.2.2.5 Case Folding

Case folding melibatkan mengubah semua huruf dalam teks menjadi huruf kecil atau besar agar perbedaan antara huruf besar dan kecil tidak lagi relevan dalam pemrosesan teks. Ini membantu memastikan konsistensi dalam analisis teks. Misalnya, mengubah "*Hello*" menjadi "*hello*" atau "*WORLD*" menjadi "*world*". Dalam penelitian ini dilakukan perubahan menjadi huruf kecil secara keseluruhan atau *lowercasing*.

3.2.2.6 Remove Stopword

Stopwords merupakan kata-kata yang umumnya tidak memberikan kontribusi signifikan pada pemahaman teks dan sering dihapus dari teks. Ini termasuk kata-kata seperti "dan", "atau", "sebuah", dll. Menghapus *stopwords* membantu fokus pada kata-kata kunci yang lebih penting dalam analisis teks.

3.2.2.7 POS Tagging

Pos tagging dalam analisis sentimen membantu mengidentifikasi peran gramatikal kata-kata dalam kalimat. Hal ini memungkinkan untuk mengenali subjek, objek, dan tindakan yang diekspresikan dalam teks, meningkatkan akurasi analisis sentimen dengan memperhatikan konteks kata-kata. Dengan demikian, pos tagging memberikan pemahaman yang lebih baik tentang struktur kalimat dan membantu mempersiapkan data teks untuk analisis sentimen yang lebih akurat.

3.2.2.8 Mapping POS Tags

Mapping POS Tags bertujuan untuk mengonversi *tag POS* yang dihasilkan oleh *POS tagger* ke format yang dikenali oleh *WordNet*. Dalam konteks analisis sentimen, *mapping* ini memungkinkan penggunaan informasi *POS tag* dalam operasi selanjutnya, seperti *lemmatization* menggunakan *WordNet*. Dengan menggunakan informasi *POS tag* yang telah di-mapping, peneliti dapat memperoleh kata-kata dalam teks dalam bentuk yang lebih bermakna dan akurat sesuai dengan peran gramatikal, sehingga memperbaiki kualitas dan kedalaman analisis sentimen yang dilakukan.

3.2.2.9 Lemmatization

Dalam penelitian ini, *lemmatization* berperan penting dalammempersiapkan data teks untuk proses analisis. *Lemmatization* merupakan proses mengubah kata-kata dalam sebuah teks ke bentuk dasarnya atau lemma. Proses ini berbeda dari *stemming*, yang memotong akhiran kata untuk mengurangi kata ke akar katanya. *Lemmatization* menggunakan analisis morfologis kata untukmengembalikannya ke bentuk dasar yang lebih bermakna. Misalnya, kata-kata seperti "*running*", "*ran*", dan "*runs*" akan dikembalikan ke bentuk dasar "*run*". Dan pada penelitian ini digunakan proses lematisasi agar analisis teks yang dilakukan lebih akurat dan bermakna.

3.2.3 Feature Extraction

Feature Extraction dalam penelitian ini berkaitan dengan mengubah teks dari dataset *Twitter* yang telah dipreproses menjadi representasi vektor numerik yang dapat dimengerti oleh model yang akan digunakan, yaitu model LSTM. Proses ini dilakukan dengan menggunakan model *Word2Vec*, yang merupakan salah satu metode populer dalam representasi kata-kata dalam bentuk vektor. Dalam konteks *Word2Vec*, setiap kata dalam teks direpresentasikan sebagai vektor numerik dalam ruang berdimensi tinggi. Representasi vektor ini memperhitungkan hubungan semantik antara kata-kata tersebut, sehingga kata- kata yang sering muncul bersama atau memiliki konteks yang mirip akan memiliki representasi vektor yang mendekat satu sama lain dalam ruang vektor.

Dengan demikian, *Word2Vec* dapat menangkap makna dan hubungan antar kata dalam teks. Dengan menggunakan fitur yang diekstraksi dari model *Word2Vec*, model LSTM dapat belajar pola-pola yang terdapat dalam teks, termasuk pola sentimen yang terkandung di dalamnya (Siti Khomsah et al., 2022). Dengan demikian, tahapan *Feature Extraction* ini bertujuan untuk mengubah teks menjadi representasi vektor yang lebih terstruktur dan informatif, sehingga memungkinkan model LSTM untuk melakukan analisis sentimen dengan lebih akurat dan efektif.

3.2.4 LSTM Modeling

Dalam penelitian ini, model LSTM digunakan untuk analisis sentimen dengan melalui beberapa langkah utama. Pertama, data teks di-tokenisasi dan kosakata dibentuk menggunakan *Tokenizer* dari Keras, yang kemudian mengubah setiap kata menjadi angka berdasarkan indeks kosakata. Setelah itu, urutan angka ini dipad (*padding*) agar semua input memiliki panjang yang konsisten, memastikan keseragaman input ke dalam model. Label kategori yang ada diubah menjadi representasi numerik menggunakan *LabelEncoder*, dan bentuk *array* label diatur menjadi dua dimensi untuk memenuhi format yang diharapkan oleh model. Langkah selanjutnya adalah membuat *layer embedding*. *Matriks embedding* dibentuk dengan memetakan setiap kata dalam kosakata ke *vektor embedding* yang telah dipelajari sebelumnya dari model *Word2Vec*. *Layer embedding* ini kemudian dibuat dengan menggunakan *matriks embedding* tersebut dan diatur agar tidak dapat dilatih, sehingga hanya memanfaatkan *vektor embedding* yang sudah ada tanpa memperbaruinya selama pelatihan model. Setelah itu, model LSTM dibangun menggunakan Keras. Model ini terdiri dari beberapa layer diantaranya layer embedding untuk mengubah token menjadi *vektor embedding*, *layer dropout* untuk mengurangi *overfitting*, layer LSTM untuk menangani pemahaman konteks urutan kata, dan *layer dense* untuk menghasilkan output akhir dengan fungsi aktivasi sigmoid yang cocok untuk klasifikasi biner.

3.2.5 Model Optimization

Dalam proses optimasi model, langkah-langkah yang dilakukan adalah membentuk kerangka kerja untuk mengoptimalkan kinerja model LSTM dalam analisis sentimen. Tahapan *hyperparameter tuning* tersebut dimulai dengan pemilihan *loss function* yang sesuai dengan jenis masalah yang dihadapi, yaitu *binary_crossentropy* untuk masalah klasifikasi biner pada analisis sentimen. *Loss function* ini memberikan pengukuran yang tepat untuk meminimalkan kesalahan prediksi model terhadap probabilitas kelas yang dihasilkan. *Optimizer* Adam dipilih sebagai algoritma optimasi utama karena adaptabilitasnya yang kuat dalam menyesuaikan laju pembelajaran secara adaptif berdasarkan momen gradien. Hal ini membantu dalam mempercepat konvergensi model menuju solusi yang optimal, serta mengatasi masalah laju pembelajaran yang lambat atau terlalu cepat. Penyetelan *hyperparameter* dilakukan melalui penggunaan *callback EarlyStopping* dan *ReduceLROnPlateau*. *Callback EarlyStopping* memonitor akurasi validasi (*val_accuracy*) dan menghentikan pelatihan jika tidak terjadi peningkatan setelah sejumlah *epoch* tertentu. Sedangkan, *callback ReduceLROnPlateau* mengurangi laju pembelajaran jika tidak ada peningkatan pada *loss function* selama beberapa *epoch*. Kedua *callback* ini membantu dalam mencegah *overfitting* dan memastikan bahwa model tidak hanya mempelajari data pelatihan dengan baik, tetapi juga mampu menggeneralisasi ke data baru dengan baik (Yu & Zhu, 2020).

Selama proses pelatihan, model dilatih dengan memanggil fungsi *model.fit()* dengan menggunakan *callback early_stopping* yang telah diatur sebelumnya. Model dilatih dengan *epoch* yang telah ditentukan, namun pelatihan dapat berhenti lebih awal jika terjadi kondisi yang ditentukan oleh *callback EarlyStopping*. Dengan langkah-langkah ini, proses *tuning* model dapat dilakukan secara efisien dan memastikan bahwa model yang dihasilkan memiliki kinerja yang baik dalam melakukan analisis sentimen pada data teks.

3.2.6 Model Evaluation

Tahapan *Model Evaluation* memegang peranan krusial dalam mengevaluasi kualitas dan kinerja model yang telah dikembangkan. Evaluasi model akan dilakukan dengan pembuatan *classification report* memberikan ringkasan kinerja

model dengan lebih lengkap, dengan menampilkan metrik evaluasi seperti *accuracy*, *precision, recall* dan *f1-score* untuk setiap kelas. Selain itu akan digunakan AUC-ROC (*Area Under the Receiver Operating Characteristic Curve*) dan AUC-PR (*Area Under the Precision-Recall Curve*) untuk mengukur kinerja model LSTM. ROC *Curve* merupakan grafik yang menunjukkan kemampuan model klasifikasi dalam membedakan antara kelas positif dan negatif pada berbagai threshold. Sumbu y (*True Positive Rate* atau *Recall*) menunjukkan tingkat deteksi positif yang benar, sedangkan sumbu x (*False Positive Rate*) menunjukkan tingkat kesalahan positif, AUC-ROC merupakan nilai area di bawah kurva ROC, yang mengukur kemampuan keseluruhan model dalam membedakan antara kelas positifdan negatif. Sedangkan *Precision-Recall Curve* adalah grafik yang menunjukkan *trade-off* antara *precision* (akurasi prediksi positif) dan *recall* (tingkat deteksi positif) pada berbagai *threshold*. *Precision* ditampilkan pada sumbu y, sementara *recall* ditampilkan pada sumbu x, AUC-PR merupakan nilai area di bawah kurva *precision-recall*. Metrik ini sangat berguna ketika bekerja dengan dataset yang tidak seimbang, karena memberikan informasi yang lebih baik tentang performa model pada kelas minoritas. Tahapan *Model Evaluation* ini penting untuk memastikan bahwa model yang dikembangkan dapat memberikan hasil yang dapat diandalkan dalam melakukan analisis sentimen pada data teks (Admojo & Sulistya, 2022).