PENGEMBANGAN DAN IMPLEMENTASI METODE *DEEP LEARNING* UNTUK MENENTUKAN BERITA PALSU DALAM BAHASA INDONESIA

SEMINAR BIDANG KAJIAN

ANTONIUS ANGGA KURNIAWAN

99219025

PROGRAM DOKTOR TEKNOLOGI INFORMASI

UNIVERSITAS GUNADARMA

AGUSTUS 2021

Abstrak

PENGEMBANGAN DAN IMPLEMENTASI METODE *DEEP LEARNING* UNTUK MENENTUKAN BERITA PALSU DALAM BAHASA INDONESIA.

Keyword: Berita Palsu, *Deep Learning*, Implementasi, Pengguna Internet.

Saat ini informasi sangat mudah didapatkan oleh pengguna internet melalui berbagai sumber seperti pesan singkat, artikel berita, dan juga media sosial. Hal ini dapat dimanfaatkan oleh pihak yang tidak bertanggung jawab untuk membuat dan menyebarkan berita palsu. Banyak pengguna internet yang belum dapat mengidentifikasi berita palsu dan berita fakta, khususnya di Indonesia. Hal ini disebabkan karena banyaknya berita palsu yang tersebar sehingga menyulitkan pengguna internet menentukan mana yang termasuk berita palsu dan mana yang termasuk berita fakta.

Penelitian terdahulu banyak yang menggunakan teknik *machine learning* dan *deep learning* dalam menentukan berita palsu. Dalam hal ini teknik *deep learning* memiliki hasil akurasi yang lebih tinggi dibandingkan dengan teknik *machine learning*. Pada penelitian sebelumnya, beberapa peneliti menyarankan untuk menggunakan jumlah data yang lebih banyak dan valid, mengintegrasikan model dengan hasil akurasi yang paling optimal ke dalam sistem siap pakai. Tujuan penelitian ini adalah untuk mengembangkan dan mengimplementasikan metode *deep learning* dengan model yang memiliki akurasi paling optimal ke dalam sistem otomatis untuk menentukan berita palsu dalam bahasa Indonesia.

Metodologi penelitian yang dilakukan terdiri dari pengumpulan data berita palsu dan fakta dari sumber yang valid, pelabelan data, *preprocessing*, *word embedding* dengan *word2vec*, membagi data menjadi data *training* dan data *testing*, menerapkan metode *deep learning* pertama kemudian dilakukan evaluasi, menerapkan metode *deep learning* yang kedua kemudian dilakukan evaluasi, dan terakhir merancang serta mengimplementasikan model yang paling optimal ke dalam sistem otomatis yang berguna untuk mengklasifikasikan berita palsu dengan cepat.

Daftar Isi

Daftar Gambar

Daftar Tabel

Bab 1

Pendahuluan

1.1 Latar Belakang

Perkembangan Internet tidak hanya mengubah penggunanya menjadi konsumen informasi, tetapi juga membawa kemudahan dalam produksi informasi. Semakin banyak pengguna Internet, semakin banyak informasi yang dihasilkan dan dikonsumsi. Kemp (2021) dalam artikelnya di Data Reportal menyebutkan bahwa pada Januari 2021, jumlah pengguna internet di Indonesia mencapai 202,6 juta. Jumlah penduduk Indonesia adalah 274,9 juta, dan tingkat penetrasi Internet adalah 73,7% (Kemp, 2021).

Data penggunaan internet di Indonesia secara tidak langsung menunjukkan besarnya potensi produsen dan konsumen informasi, termasuk produksi dan konsumsi berita palsu. Menurut survei terhadap 941 responden oleh Masyarakat Telematika Indonesia (Mastel) (2019), media sosial adalah media komunikasi yang paling banyak digunakan untuk menyebarkan berita palsu. Politik, ketertiban umum, bisnis, ilmu pengetahuan, kesehatan, bencana alam, dan sosial adalah beberapa bidang yang banyak digunakan untuk menyebarkan berita palsu (*Hasil Survey Wabah HOAX Nasional 2019 | Website Masyarakat Telematika Indonesia*, 2019). Ketika pengguna internet di Indonesia terpengaruh oleh penyebaran berita palsu, dampak kerugian dari berbagai bidang dapat terjadi.

Pemerintah telah melakukan banyak upaya untuk memerangi penyebaran berita palsu. Salah satunya bekerja sama dengan Facebook untuk memblokir dan menghapus berita palsu. Selain itu, gerakan-gerakan sosial melawan hoaks juga muncul dari masyarakat, yaitu dengan didirikannya situs turnbackhoax.id oleh MAFINDO (Masyarakat Anti Hoax Indonesia) untuk memeriksa kebenaran fakta dari suatu berita.

Metode identifikasi atau klasifikasi yang dilakukan pada situs turnbackhoax.id masih menggunakan proses manual (Panjaitan & Santoso, 2021). Oleh karena itu, ketika informasi tumbuh, semakin banyak informasi yang masuk, maka akan menjadi sulit. Selain itu, menurut survey yang dilakukan oleh Masyarakat Telematika Indonesia (Mastel), hanya 16.20% responden yang dapat langsung membedakan berita palsu. Selain itu, sekitar 21.80% responden merasa sulit untuk memeriksa kebenaran suatu

berita (*Hasil Survey Wabah HOAX Nasional 2019 | Website Masyarakat Telematika Indonesia*, 2019).

Oleh karena itu, dibutuhkan suatu metode untuk mengklasifikasikan berita dengan cepat, dan alat yang dapat digunakan pengguna Internet untuk secara otomatis membedakan antara berita yang benar dan yang palsu.

State of the art dalam penelitian ini mengenai deteksi berita palsu diantaranya adalah Kurniawan & Mustikasari (2021) melakukan penelitian dengan judul "Implementasi Deep Learning Menggunakan Metode CNN dan LSTM untuk Menentukan Berita Palsu dalam Bahasa Indonesia". Data yang digunakan sebanyak 1786 berita, terdiri dari 802 berita fakta dan 984 berita palsu. Metode yang digunakan adalah pelabelan data, *preprocessing* data, *word embedding, splitting data,* pembuatan model menggunakan CNN dan LSTM, kemudian menguji model dengan data baru, dan terakhir melakukan perbandingan dari kedua model. Hasil yang didapatkan adalah tingkat akurasi dari model CNN lebih baik daripada model LSTM, dengan akurasi tes CNN sebesar 0.88 dan LSTM sebesar 0.84. Peneliti menyarankan untuk menggunakan data yang lebih banyak lagi agar proses pembelajaran yang dilakukan semakin baik. Pengolahan data pada tahapan *preprocessing* dan *word embedding* dalam bahasa Indonesia lebih diperhatikan agar dapat menambah tingkat keakuratan dari sebuah model. Peneliti berpendapat bahwa ada kemungkinan jika menggabungkan kedua model CNN dan LSTM dapat meningkatkan hasil akurasi yang lebih optimal dalam menentukan berita palsu. Selain itu, peneliti juga menyarankan untuk membangun sebuah sistem otomatis menggunakan model yang paling optimal untuk dapat dimanfaatkan para pengguna internet dalam menentukan berita fakta dan berita palsu dengan cepat (Kurniawan & Mustikasari, 2021).

Putri et al (2019) melakukan penelitian dengan judul "*Analysis and Detection of Hoax Contents in Indonesian News Based on Machine Learning*". Data yang digunakan sebanyak 251 berita yang terdiri dari berita fakta sebanyak 151 dan berita palsu sebanyak 100. Metode yang digunakan adalah *text preprocessing* dan *feature extraction*, kemudian dibandingkan 5 model algoritma, yaitu *Mulitilayer Perceptron, Naïve Bayes, SVM, Random Forest* dan *Decision Tree. Random Forest* memiliki hasil akurasi yang lebih baik daripada algoritma lainnya, yaitu sebesar 76.47% (Putri et al., 2019). Kekurangan dari penelitian ini adalah jumlah berita yang digunakan sebagai data belum cukup banyak untuk proses pelatihan dan pembelajaran, sehingga hasil akurasi dari model yang dibentuk belum terlalu optimal.

Rahutomo et al (2019) melakukan penelitian yang berjudul "Eksperimen Naïve Bayes Pada Deteksi Berita *Hoax* Berbahasa Indonesia". Jumlah *dataset* yang digunakan sebanyak 600 berita yang terdiri dari berita fakta dan berita palsu. Metode yang digunakan terdiri dari *preprocessing, manual voting tagging* untuk pelabelan berita fakta atau berita palsu, implementasi model *Naïve Bayes Classifier*, kemudian melakukan pengujian statis dan dinamis. Pengujian dinamis dilakukan dengan membuat web sederhana yang digunakan untuk menguji dan mendeteksi berita fakta atau berita palsu. Hasil evaluasi dilakukan menggunakan 3 parameter yaitu *accuracy, precision* dan *recall*. Akurasi yang dihasilkan dengan pengujian statis sebesar 82.6%, sedangkan akurasi yang dihasilkan dengan pengujian dinamis sebesar 68.33% (Rahutomo et al., 2019). Kekurangan dalam penelitian ini adalah metode yang diujikan hanya menggunakan 1 metode saja, yaitu *Naïve Bayes*. Pada pengujian dinamis hasil akurasi yang dihasilkan masih kurang dari 70%.

Ananth et al (2019) melakukan penelitian yang berjudul "*Fake News Detection Using Convolution Neural Network in Deep Learning*". Penelitian ini bertujuan untuk membandingkan teknik *machine learning* dan teknik *deep learning* yang terdiri dari *Naïve Bayes, Decision Tree, Random Forest, K-Nearest Neighbor,* CNN dan LSTM. Hasil dari penelitian ini adalah teknik *deep learning* menggunakan metode CNN dan LSTM memiliki hasil akurasi yang lebih baik daripada teknik *machine learning* dengan metode *Naïve Bayes, Decision Tree, Random Forest, K-Nearest Neighbor*. Ananth et al (2019) merasa ketersediaan *dataset* dan literatur yang digunakan untuk menentukan berita fakta dan berita palsu masih terbatas. Ananth et al (2019) juga mengungkapkan bahwa untuk penelitian selanjutnya diharapkan melakukan implementasi model untuk deteksi berita palsu ke dalam sebuah sistem atau aplikasi yang dapat memudahkan pengguna internet untuk menentukan berita palsu secara cepat (Ananth et al., 2019).

Rodriguez & Iglesias (2019) melakukan penelitian yang berjudul "*Fake News Detection Using Deep Learning*". Data yang digunakan sebanyak 20015 berita yang terdiri dari berita fakta dan berita palsu dalam bahasa Inggris. Metode yang digunakan terdiri dari pengumpulan data, data transformasi menggunakan *Word2Vec,* kemudian mengimplementasikan LSTM, CNN, dan Bert model. Penelitian ini membuktikan bahwa teknik *deep learning* layak digunakan untuk klasifikasi *fake news*, karena akurasi dari ketiga model cukup tinggi, yaitu di atas 90%. Rodriguez & Iglesias (2019) menyarankan untuk penelitian berikutnya, data yang digunakan harus lebih banyak lagi guna melatih model supaya lebih baik lagi dalam menentukan berita fakta atau berita

palsu. Selain itu, disarankan untuk mengintegrasikan model dengan media sosial, web atau aplikasi agar dapat dipakai dalam kehidupan nyata untuk menentukan berita palsu (Rodriguez & Iglesias, 2019).

Berdasarkan penelitian terdahulu, tujuan dari penelitian ini adalah mengembangkan dan mengimplementasikan metode dari *deep learning* ke dalam sistem otomatis yang dapat digunakan oleh pengguna internet dalam menentukan berita fakta dan berita palsu dalam bahasa Indonesia dengan cepat.

1.2 Identifikasi Masalah

Berdasarkan uraian dari latar belakang, diidentifikasi beberapa masalah yang ada, yaitu:

1. Banyaknya berita palsu atau hoaks yang tersebar melalui internet.
2. Dampak buruk yang dialami para pengguna internet akibat berita palsu yang tersebar luas.
3. Proses identifikasi dan klasifikasi berita palsu yang dilakukan oleh gerakan-gerakan sosial melawan hoaks masih menggunakan proses manual.
4. Belum semua pengguna internet mampu menyaring berita-berita yang diterima adalah berita palsu atau berita fakta, sehingga dibutuhkan sebuah sistem yang dapat menentukan berita palsu atau berita fakta secara otomatis.

1.3 Batasan Masalah

Dalam penelitian ini terdapat batasan dan tujuan yang terdiri dari:

1. Data yang digunakan menggunakan berita dalam bahasa Indonesia.
2. Metode yang dipilih menggunakan metode *deep learning.*
3. Membantu para pengguna internet dalam mengidentifikasi berita palsu atau hoaks dalam bahasa Indonesia menggunakan sebuah sistem otomatis.

1.3 Rumusan Masalah

Berdasarkan uraian latar belakang sebelumnya, masalah yang dapat dirumuskan adalah:

1. Bagaimana cara mendapatkan data sumber yang banyak dan valid terkait berita palsu dan berita fakta dalam bahasa Indonesia?
2. Bagaimana cara mendapatkan model yang optimal untuk menentukan berita palsu atau berita fakta dalam bahasa Indonesia?

3. Bagaimana cara mengimplementasikan model yang optimal ke dalam sebuah sistem otomatis yang bisa digunakan oleh pengguna internet dalam menentukan berita palsu atau berita fakta?

1.4 Tujuan Penelitian

Penelitian ini bertujuan untuk mengembangkan dan mengimplementasikan metode dari *deep learning* ke dalam sistem otomatis yang dapat digunakan oleh pengguna internet dalam menentukan berita fakta dan berita palsu dalam bahasa Indonesia.

1.5 Kontribusi Penelitian

Penelitian ini memiliki 2 manfaat yang terdiri dari:

1. Untuk Pengetahuan

   Penelitian ini dapat memberikan kontribusi dalam ilmu pengetahuan, terutama dalam bidang *data science* seperti *machine learning* dan *deep learning*. Dengan penelitian ini, diharapkan agar pengetahuan pada bidang *machine learning* dan *deep learning*, terutama dalam hal klasifikasi dapat terus berkembang dan lebih banyak lagi peneliti atau orang lain yang tertarik untuk mengembangkan ilmu pada bidang ini.
2. Untuk Pengguna Internet

   Hasil yang diperoleh dari penelitian ini dapat membantu pengguna internet dalam mengidentifikasi informasi atau berita yang diterima melalui internet. Sehingga pengguna internet dapat lebih kritis, terbuka, dan teliti lagi dalam menerima suatu informasi atau berita.

1.6 Usulan Kebaruan

Berdasarkan *state of the art* yang sudah dijelaskan pada latar belakang, untuk itu peneliti mengusulkan beberapa usulan yang kiranya dapat menjadi kebaruan, diantaranya adalah:

1. Melakukan pengembangan metode *deep learning* untuk menentukan berita palsu dan berita fakta dalam bahasa Indonesia menggunakan 2 metode *deep learning.* Dalam hal ini rencananya akan menggunakan metode LSTM dan CNN namun masih perlu mempelajarinya kembali dan membaca studi literatur terkait metode yang diajukan.

2. Setelah mendapatkan model yang optimal untuk menentukan berita palsu dan berita fakta, dalam penelitian ini mengusulkan untuk mengimplementasikan model yang optimal tersebut ke dalam sistem otomatis seperti situs web yang bisa diakses dan digunakan untuk menentukan berita palsu dengan cepat.

1.7 Hipotesis

Berdasarkan usulan kebaruan yang dicantumkan muncul beberapa hipotesis diantaranya adalah:

1. Dengan menerapkan 2 metode *deep learning* untuk menentukan berita palsu, diharapkan model yang dihasilkan memiliki tingkat keakuratan yang lebih tinggi daripada penelitian terdahulu.
2. Dengan pembuatan situs web menggunakan model yang sudah optimal dalam menentukan berita palsu, diharapkan banyak pengguna internet yang dapat dengan mudah mengidentifikasi validitas sebuah berita atau informasi yang diterima palsu atau fakta.

Bab 2

Tinjauan Pustaka

2.1 Penelitian Terdahulu

Terdapat beberapa penelitian terdahulu yang menjadi referensi dalam melakukan penelitian ini. Penelitian-penelitian tersebut mengenai identifikasi berita palsu atau hoaks. Penelitian-penelitian terdahulu ditunjukan pada tabel 2.1.

Tabel 2.1. Penelitian Terdahulu

Nama Peneliti	Judul	Metode	Hasil Penelitian
(Kurniawan & Mustikasari, 2021)	Implementasi Deep Learning Menggunakan Metode CNN dan LSTM untuk Menentukan Berita Palsu dalam Bahasa Indonesia	Jumlah data: 1786 berita, terdiri dari 802 berita fakta dan 984 berita palsu. Metode yang digunakan adalah pelabelan data, *preprocessing data*, *word embedding*, *splitting data*, pemodelan dengan CNN dan LSTM, kemudian menguji model dengan data baru.	Hasilnya adalah tingkat akurasi dari model CNN lebih baik daripada model LSTM, dengan akurasi tes CNN sebesar 0.88 dan LSTM sebesar 0.84. Pengujian dengan data baru, dari 4 berita hanya 1 berita yang salah hasil prediksinya menggunakan LSTM.
(Putri et al., 2019)	Analysis and Detection of Hoax Contents in Indonesian News Based on Machine Learning	Machine Learning Text Classification (Text Preprocessing and Feature Extraction), Membandingkan Algoritma Multilayer Perceptron, Naïve Bayes, SVM, Random Forest, Decision Tree	Algoritma Random Forest memiliki hasil akurasi yang lebih baik dibanding algoritma lainnya.
(Rahutomo et al., 2019)	Eksperimen Naïve Bayes Pada Deteksi Berita Hoax Berbahasa Indonesia	Klasifikasi teks dengan *preprocessing*, *manual voting tagging* untuk pelabelan valid dan hoax, implementasi Naïve Bayes Classifier, pengujuan statis dan dinamis	Hasil evaluasi matriks dengan 3 parameter, yaitu Accuracy, Precision, Recall. Pengujian statis akurasinya lebih tinggi dari dinamis.

(Rodriguez & Iglesias, 2019)	Fake News Detection Using Deep Learning	Pengumpulan Data, Data Transformasi (Word2Vec), Implementasi LSTM, CNN, dan Bert.	Deep learning layak digunakan untuk klasifikasi *fake news*. Akurasi tinggi sekitar 93%.
(Ananth et al., 2019)	Fake News Detection using Convolution Neural Network in Deep Learning	Menggunakan NLP, Machine Learning, dan Deep Learning Teknik. Model: CNN, LSTM, Naïve Bayes, Decision Tree, Random Forest, K-Nearest Neighbor	Deep Learning model CNN & LSTM memiliki hasil yang lebih baik dibandingkan Machine Learning model.
(Prasetyo et al., 2018)	Klasifikasi Hoax Pada Berita Kesehatan Berbahasa Indonesia Dengan Menggunakan Metode Modified K-Nearest Neighbor	*Preprocessing* (*parsing*, tokenisasi, *filtering/stopword removal*, *stemming*), pembobotan, implementasi metode K-Nearest Neighbor (KNN)	Implementasi dan pengujian menghasilkan nilai k terbaik berjumlah 4, precision sebesar 0,83, recall sebesar 0,75, f-measure sebesar 0,79 dan akurasi sebesar 75%.
(Afriza & Adisantoso, 2018)	Metode Klasifikasi Rocchio untuk Analisis Hoax Rocchio Classification Method for Hoax Analysis	Pengumpulan dokumen, praproses, seleksi fitur (tf-idf), pembagian data, klasifikasi Rocchio dan Naïve Bayes, kemudian melakukan evaluasi menggunakan *confusion matrix*.	Akurasi Rocchio sebesar 83.501% sedangkan Multinomial Naive Bayes sebesar 65.835%.
(Utami & Sari, 2018)	Filtering Hoax Menggunakan Naive Bayes Classifier	*Pre-processing (Case folding -Tokenize - Stopwords*) -> Naïve Bayes-> Evaluasi (*accuracy)*	Hasil akurasi Naïve Bayes *Classifier* adalah 88%, diambil dari 50 data testing komentar pada forum Female Daily.
(Maulina & Sagara, 2018)	Klasifikasi Artikel Hoax Menggunakan Support Vector Machine Linear Dengan Pembobotan Term	*Pre-processing* (*Tokenize - Stopwords - Stemming*) -> TF-IDF -> SVM -> Evaluasi (Akurasi dan Kecepatan).	Dari 108 artikel hoax dan 132 artikel tidak hoax, tingkat akurasi dengan *Cross Validation* 10Fold adalah 95,8333%,

	Frequency – Inverse Document Frequency		dukungan vektor yang dimiliki oleh model adalah 14 vektor.
(Radhika et al., 2018)	A Text Classification Model Using Convolution Neural Network and Recurrent Neural Network	Melakukan training dan test terhadap CNN dan RNN model	RNN model memiliki hasil akurasi yang lebih baik dibandingkan dengan CNN model.
(Nowak et al., 2017)	LSTM Recurrent Neural Networks for Short Text and Sentiment Classication	Menggunakan beberapa LSTM Network seperti LSTM, Bi-LSTM, dan Gated Recurrent Unit. Dataset menggunakan 3 sumber dataset: *Spam Collection Dataset*, *Farms Ads Dataset*, dan *Amazon Books Review*.	Bi-LSTM memiliki hasil akurasi yang paling baik dibandingkan dengan GRU dan LSTM.
(Rasywir & Purwarianti, 2015)	Eksperimen pada Sistem Klasifikasi Berita Hoax Berbahasa Indonesia Berbasis Pembelajaran Mesin	Praproses, ekstraksi fitur, seleksi fitur dan pengeksekusian model klasifikasi Naïve Bayes, SVM, dan C4.5	Naïve bayes menunjukkan hasil akurasi yang terbaik dibandingkan dengan SVM dan C4.5 dengan nilai akurasi 91.36%.

2.2 Berita Palsu (Hoaks)

Berita palsu atau hoaks merupakan serangkaian informasi yang sesungguhnya tidak benar, tetapi sengaja dibuat seolah-olah benar adanya. Hoaks juga dapat didefinisikan sebagai isu-isu terkini yang digunakan sebagai senjata politik, kebenaran yang tidak relevan, atau kabar bohong yang disebarkan secara sengaja (Berghel, 2017).

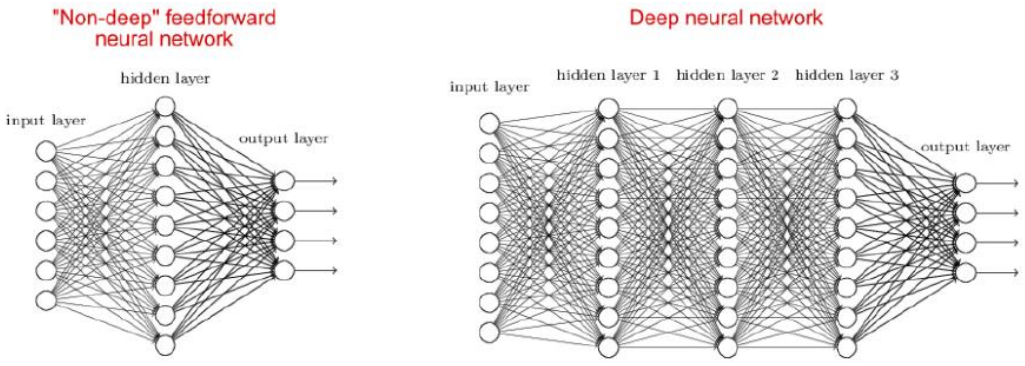
Pembuat hoaks akan berusaha membuat berita yang meyakinkan sehingga membuat pengguna internet, pendengar dan pembaca berita atau informasi mempercayai berita palsu yang disebarkan. Dampak negatif yang sering ditimbulkan akibat adanya berita palsu atau hoaks diantaranya adalah buang-buang waktu dan uang, pengalihan isu, penipuan publik dan pemicu kepanikan publik.

2.3 *Deep Learning*

Deep learning merupakan disiplin ilmu subset dari *machine learning*, tidak seperti teknik *machine learning* konvensional yang dibatasi oleh kemampuannya untuk memproses data mentah dan tergantung pada keahlian domain yang besar. Jika *machine learning* membutuhkan rekayasa yang cermat untuk merancang ekstrasi fitur, *deep learning* dapat mempelajari fitur dan memproses data secara langsung dalam bentuk mentah (LeCun, Y., Bengio, Y., Hinton, 2015).

Goodfellow et al (2016) mendefinisikan *deep learning* sebagai berikut: "*Deep Learning* adalah jenis *machine learning* yang memiliki kekuatan besar dan fleksibilitas yang tinggi dengan belajar untuk mewakili dunia sebagai hierarki konsep bersarang, dengan masing-masing konsep didefinisikan dalam kaitannya dengan konsep yang lebih sederhana, dan lebih banyak representasi abstrak yang dihitung dalam hal yang kurang abstrak" (Goodfellow et al., 2016).

Sebagai contoh jika ada tugas untuk mengklasifikasikan gambar yang diberikan, jika itu mewakili kucing atau anjing, teknik *machine learning* konvensional harus mendefinisikan fitur wajah seperti telinga, mata, kumis, mulut dan sebagainya, maka perlu menulis metode untuk menentukan fitur mana yang lebih penting ketika mengklasifikasikan hewan tertentu, sedangkan *deep learning* tidak perlu menyediakan fitur secara manual, dengan *deep learning* fitur yang paling penting akan diekstraksi secara otomatis, setelah menentukan fitur mana yang paling penting untuk mengklasifikasi foto (Zaccone & Karim, 2018).



Gambar 2.1. Perbandingan Standar *Neural Network* dan *Deep Neural Network* (Arbones, 2017)

Implementasi dari *deep learning* biasanya menggunakan arsitektur *neural network* seperti yang ditunjukkan pada gambar 2.1, tetapi tidak seperti *neural network*

tradisional yang biasanya hanya terdiri dari beberapa layer, *deep learning* biasanya terdiri dari ratusan bahkan ribuan layer untuk jaringan tersebut (Huang et al., 2016).

2.4 Data *Preprocessing*

Proses *preprocessing* adalah langkah-langkah dalam memproses data yang bertujuan untuk meningkatkan urutan kata atau kalimat agar lebih mudah dibaca oleh mesin. Dalam hal ini, pada saat yang sama, dapat mengurangi nilai ambiguitas saat mengekstraksi fitur. Tahapan *data preprocessing* yang akan dilakukan adalah *data cleaning, case folding, tokenizing,* dan *filtering (stopword removal)*.

2.4.1 Data *Cleaning*

Data *cleaning* atau biasa disebut dengan data *cleansing* merupakan suatu proses analisa kualitas dari suatu data. Cara yang biasa dilakukan adalah dengan mengubah, mengoreksi, atau menghapus data-data yang salah, tidak lengkap, tidak akurat, atau memiliki format yang salah, sehingga data *cleaning* dapat menghasilkan data berkualitas tinggi.

2.4.2 *Case Folding*

Case folding merupakan sebuah teknik yang bertujuan untuk mengubah semua karakter huruf dalam dokumen atau kalimat menjadi huruf kecil (*lowercase*). *Case folding* adalah salah satu bentuk dari text *preprocessing* yang paling sederhana dan efektif meskipun sering diabaikan. Terdapat beberapa cara yang dapat dilakukan, yaitu merubah teks menjadi huruf kecil, menghapus atau menghilangkan karakter yang dianggap tidak valid seperti angka, tanda baca, dan uniform resource locator (url), serta menghapus karakter kosong (whitespace) (Indraloka & Santosa, 2017).

2.4.3 *Tokenizing*

Tokenizing adalah proses pemisahan teks menjadi potongan-potongan yang disebut sebagai token untuk kemudian di analisa. Kata, angka, simbol, tanda baca dan entitas penting lainnya dapat dianggap sebagai *token*. Pada NLP token diartikan sebagai "kata" meskipun *tokenize* juga dapat dilakukan pada paragraf maupun kalimat.

2.4.4 *Stopword Removal*

Stopword removal adalah proses mengambil kata-kata penting dari hasil *token* dengan menggunakan algoritma *stoplist* (membuang kata kurang penting) atau wordlist (menyimpan kata penting) (Nugroho, 2019).

Stopword adalah kata umum yang biasanya muncul dalam jumlah besar dan dianggap tidak memiliki makna. Contoh *stopword* dalam bahasa Indonesia adalah "yang", "dan", "di", "dari", dll. Makna di balik penggunaan *stopword* yaitu dengan menghapus kata-kata yang memiliki informasi rendah dari sebuah teks, kita dapat fokus pada kata-kata penting sebagai gantinya.

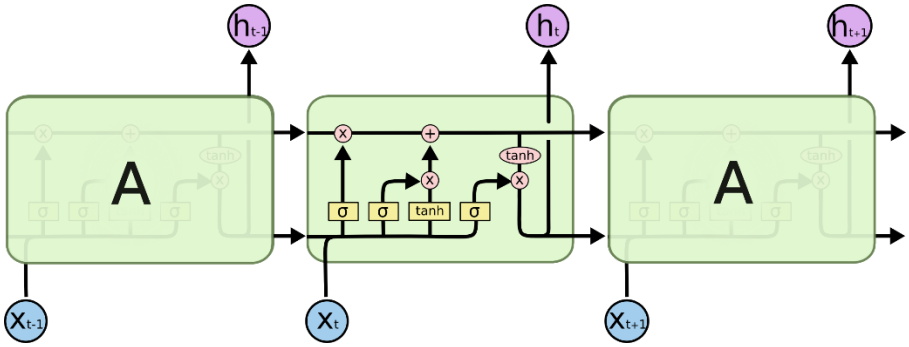
2.5 *Word2Vec*

Word2Vec merupakan algoritma *word embedding*, yaitu pemetaan dari kata menjadi vektor. *Word2Vec* juga dapat diartikan suatu metode untuk merepresentasikan setiap kata di dalam konteks sebagai vektor dengan N dimensi (*Word2Vec - Arif R - Medium*, 2018). Dalam mempresentasikan suatu kata, *Word2Vec* mengimplementasi *neural network* untuk menghitung *contextual* dan *semantic similarity* (kesamaan kontekstual dan semantik) dari setiap kata (*inputan*) yang berbentuk *one-hot encoded vectors*. Hasil dari *contextual* dan *semantic similarity* ini dapat merepresentasikan relasi suatu kata dengan kata lainnya, misalnya relasi antara 'Laki-laki — Perempuan', relasi pada '*Verb tense*', dan bahkan relasi pada 'Marah — Mengamuk'.

2.6 *Long Short-Term Memory* (LSTM)

Long Short-Term Memory (LSTM) adalah jenis khusus dari RNN yang mampu mempelajari ketergantungan jangka panjang. LSTM diciptakan untuk menyelesaikan permasalahan RNN sebelumnya, yaitu gradien yang menghilang (*vanishing gradient*). LSTM dirancang secara eksplisit untuk menghindari masalah ketergantungan jangka panjang. Mengingat informasi untuk jangka waktu yang lama adalah perilaku standar dari LSTM.

LSTM juga memiliki struktur berulang seperti RNN, namun LSTM memiliki struktur yang berbeda dalam melakukan pemrosesan. Biasanya RNN hanya memiliki 1 lapisan jaringan saraf, tetapi LSTM memiliki 4 lapisan dan berinteraksi dengan cara yang sangat istimewa.



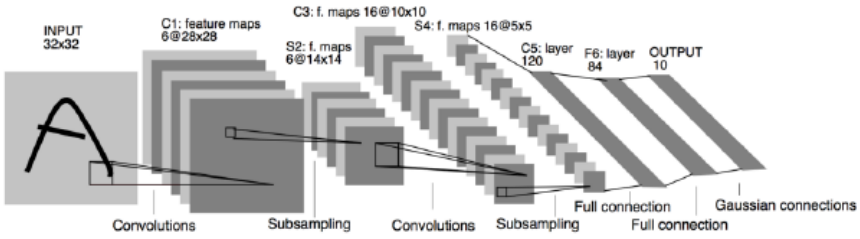
Gambar 2.2 Modul berulang dalam LSTM berisi empat lapisan yang saling berinteraksi [27].

Gambar 2.3 Notasi modul berulang LSTM (Colah, 2015).

Pada gambar 2.2 menunjukkan 4 lapisan yang dimaksud dan gambar 2.3 adalah keterangan dari notasi berulang LSTM, setiap garis membawa seluruh vektor, dari output satu simpul (*node*) ke input yang lain. Lingkaran merah muda mewakili operasi elemen, seperti penambahan atau perkalian elemen vektor, sedangkan kotak kuning adalah lapis jaringan saraf (mengandung parameter dan bias) yang bisa belajar. Dua garis yang bergabung menandakan penggabungan dua matriks atau vektor, sementara garis berpisah menandakan kontennya disalin dan salinannya pergi ke simpul yang berbeda (Colah, 2015).

2.7 *Convolutional Neural Network* (CNN)

Convolutional Neural Network adalah salah satu metode *machine learning* dari pengembangan *Multi Layer Perceptron* (MLP) yang didesain untuk mengolah data dua dimensi. CNN termasuk dalam jenis *Deep Neural Network* karena dalamnya tingkat jaringan dan banyak diimplementasikan untuk masalah klasifikasi dan juga data citra. CNN memiliki dua metode, yaitu klasifikasi menggunakan *feedforward* dan tahap pembelajaran menggunakan *backpropagation*. Gambar 2.4 menunjukkan bentuk LeNet arsitektur dari metode CNN.



Gambar 2.4. LeNet Arsitektur (Goodfellow et al., 2016)

Cara kerja CNN memiliki kesamaan pada MLP (*Multi Layer Perceptron*), namun dalam CNN setiap *neuron* dipresentasikan dalam bentuk dua dimensi, tidak seperti MLP yang setiap *neuron* hanya berukuran satu dimensi. Sama halnya dengan *Neural Network* pada umumnya, CNN memiliki beberapa lapisan tersembunyi (*hidden layers*) dari sebuah *input* berupa vektor tunggal (Goodfellow et al., 2016).

2.8 *Evaluation Metrics*

Evaluasi dapat menunjukkan bahwa model yang digunakan memiliki hasil yang memuaskan atau tidak. Misalnya, metode yang digunakan memiliki hasil model yang memuaskan pada metrik evaluasi *precision*, tetapi metrik evaluasi lainnya memiliki hasil yang buruk. Jadi, perlu untuk mengevaluasi beberapa jenis metrik evaluasi yang ada, untuk mendapatkan kesimpulan apakah metode yang digunakan memiliki model yang baik atau buruk. Ada beberapa evaluasi yang dapat digunakan untuk mengevaluasi kinerja algoritma klasifikasi untuk prediksi, seperti *confusion matrix*, *accuracy*, *recall*, *precision*, dan lain-lain.

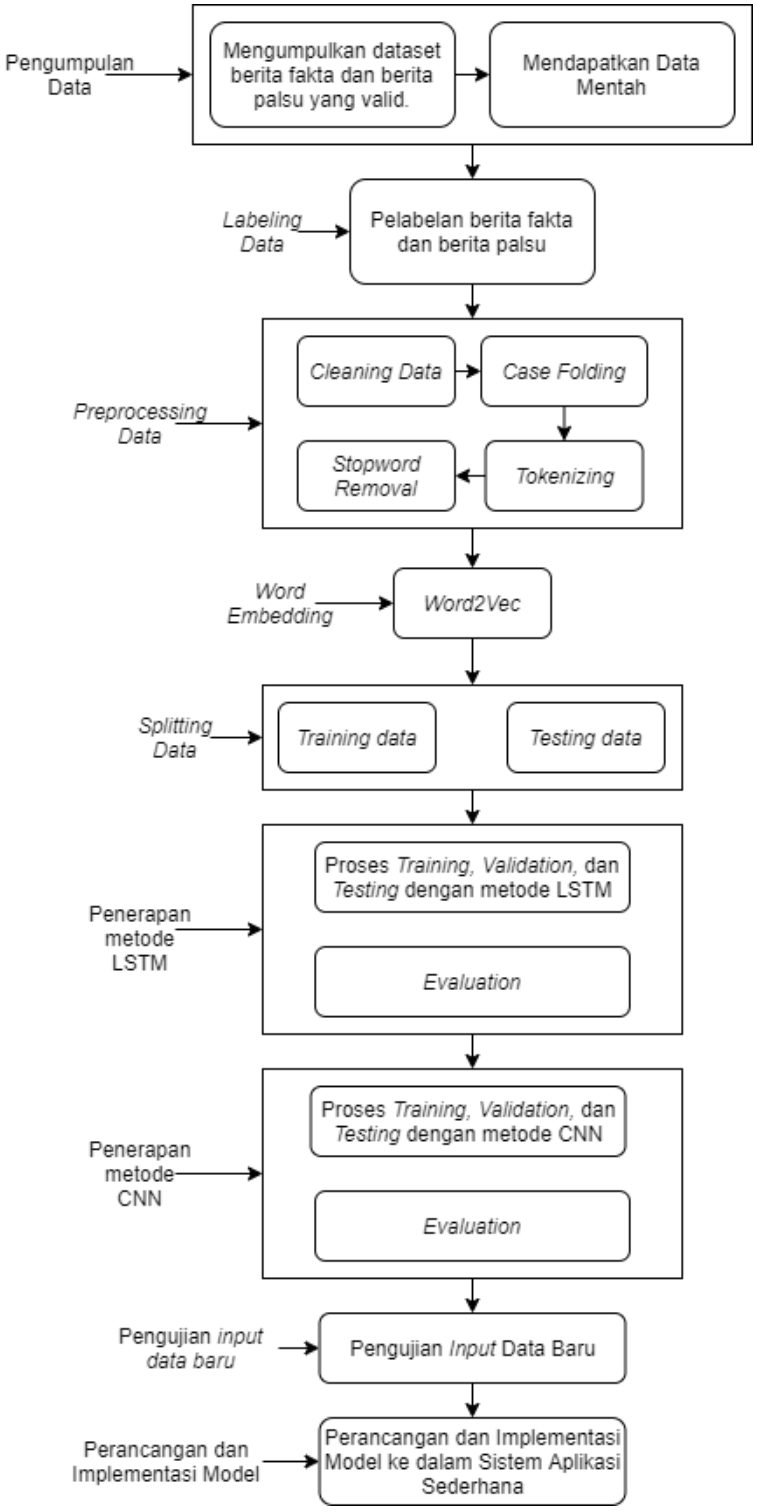
2.9 *Website*

Website adalah kumpulan informasi atau kumpulan halaman yang biasa diakses lewat jalur internet. Setiap orang di berbagai tempat dan segala waktu bisa menggunakannya selama terhubung secara *online* di jaringan internet. Secara teknis, *website* adalah kumpulan dari halaman, yang tergabung kedalam suatu domain atau subdomain tertentu. *Website-website* yang ada berada di dalam *World Wide Web* (WWW) Internet.

Bab 3

Metodologi

Tahapan penelitian terbagi menjadi 9 tahap seperti ditunjukkan pada gambar 3.1.



Gambar 3.1. Tahapan penelitian

3.1 Pengumpulan Data

Pada penelitian ini data yang dikumpulkan berbentuk teks berita yang terbagi menjadi dua, yaitu berita fakta dan berita palsu. Data yang dikumpulkan diambil secara manual melalui situs penyedia berita fakta dan berita palsu. Indonesia memiliki dua buah situs penyedia berita palsu dan berita fakta yang sudah diakui validitasnya, yaitu situs Kemenkominfo dan situs Turnbackhoax.id.

Data yang dikumpulkan rencananya akan mengambil data dengan rentan waktu 1 sampai 2 tahun terakhir supaya data yang diolah adalah data terbaru dan jumlahnya banyak. Semakin banyak jumlah data yang digunakan, maka suatu metode dapat melakukan pelatihan dan pembelajaran yang semakin baik. Data yang diperoleh masih berbentuk data mentah yang nantinya akan disimpan ke dalam format CSV *file*.

3.2 *Labeling Data*

Tahap *labeling data* adalah tahap memberikan sebuah label berita fakta dan berita palsu dari data berita yang dikumpulkan. Pemberian label akan dibagi menjadi 2 kategori dengan "0" sebagai berita fakta dan "1" sebagai berita palsu.

3.3 *Preprocessing Data*

Pada tahap ini, data yang sudah dimuat dan diberikan label kemudian dipersiapkan agar lebih mudah dibaca oleh mesin. Pada penelitian ini tahap *preprocessing* terdiri menjadi 4 tahap, yaitu proses *cleaning data*, proses *case folding*, proses *tokenizing*, dan proses *stopword removal*.

3.3.1 Cleaning Data

Proses ini bertujuan untuk membersihkan elemen-elemen yang dapat mengurangi makna sebenarnya dari suatu teks. Elemen yang dihilangkan adalah tanda baca, angka, spasi ganda, link URL, dan mention (@). Elemen tersebut ada di dalam data teks, namun tidak relevan dengan topik dari berita palsu. Contoh hasil dari proses cleaning ditunjukkan pada tabel 3.1.

Tabel 3.1 Contoh hasil *data cleaning*

Data Asli	Hasil Cleaning
Mendiang Abdullah Fithri Setiawan – rahiimahulloh – adalah karyawan TV Muhammadiyah, yang meliput dalam	Mendiang Abdullah Fithri Setiawan rahiimahulloh adalah karyawan TV Muhammadiyah yang meliput dalam

tentang mobil Esemka. Dan kini dia – rahiimahulloh – dibunuh dengan sadis.	tentang mobil Esemka Dan kini dia rahiimahulloh dibunuh dengan sadis
Beredar pesan siaran atau broadcast … dilakukan lewat situs https://sidalih3.kpu.go.id hingga hari pencoblosan. ...	Beredar pesan siaran atau broadcast … dilakukan lewat situs hingga hari pencoblosan ...
Penonton sepi teman Ahok dikasih tiket gratis dan uang Rp 50 ribu supaya mau nonton film si Ahok	Penonton sepi teman Ahok dikasih tiket gratis dan uang Rp ribu supaya mau nonton film si Ahok

Tabel 3.1 menunjukkan hasil dari contoh proses *cleaning data*. Dalam tabel terlihat tanda baca seperti titik dan setrip dihilangkan. Kalimat yang mengandung alamat URL dan angka juga ikut dihilangkan. Setelah proses *cleaning data* selesai dilakukan, berikutnya adalah melakukan tahapan *case folding.*

3.3.2 *Case Folding*

Seluruh teks yang telah dibersihkan kemudian diubah menjadi menjadi huruf kecil. Hal ini dilakukan karena tidak semua data teks konsisten dalam penggunaan huruf kapital. Contoh hasil dari proses case folding ditunjukkan pada tabel 3.2.

Tabel 3.2 Contoh hasil *case folding*

Hasil *Data Cleaning*	**Hasil *Case Folding***
Mendiang Abdullah Fithri Setiawan rahiimahulloh adalah karyawan TV Muhammadiyah yang meliput dalam tentang mobil Esemka Dan kini dia rahiimahulloh dibunuh dengan sadis	mendiang abdullah fithri setiawan rahiimahulloh adalah karyawan tv muhammadiyah yang meliput dalam tentang mobil esemka dan kini dia rahiimahulloh dibunuh dengan sadis
Beredar pesan siaran atau broadcast … dilakukan lewat situs hingga hari pencoblosan ...	beredar pesan siaran atau broadcast … dilakukan lewat situs hingga hari pencoblosan ...
Penonton sepi teman Ahok dikasih tiket gratis dan uang Rp ribu supaya mau nonton film si Ahok	penonton sepi teman ahok dikasih tiket gratis dan uang rp ribu supaya mau nonton film si ahok

Case folding adalah proses di mana semua kalimat yang memiliki huruf besar akan diubah ke dalam huruf kecil seperti ditunjukkan pada tabel 3.2. Setelah proses case folding selesai dilakukan, berikutnya adalah melakukan tahapan tokenization.

3.3.3 Tokenizing

Secara umum, tahap *tokenizing* adalah proses pemisahan kata di dalam teks ke dalam unit kata atau token. Tujuan dari proses ini adalah untuk membuat token individual yang mewakili setiap kata dalam pernyataan. Contoh hasil dari proses *tokenizing* ditunjukkan pada table 3.3.

Tabel 3.3 Contoh hasil *tokenizing*

Hasil *Case Folding*	**Hasil *Tokenizing***
mendiang abdullah fithri setiawan rahiimahulloh adalah karyawan tv muhammadiyah yang meliput dalam tentang mobil esemka dan kini dia rahiimahulloh dibunuh dengan sadis	[“mendiang”, ”abdullah”, ”fithri”, ”setiawan”, ”rahiimahulloh”, ”adalah”, ”karyawan”, ”tv”, ”muhammadiyah”, ”yang”, ”meliput”, ”dalam”, ”tentang”, ”mobil”, ”esemka”, ”dan”, ”kini”, ”dia”, ”rahiimahulloh”, ”dibunuh”, ”dengan”, ”sadis”]
beredar pesan siaran atau broadcast … dilakukan lewat situs hingga hari pencoblosan ...	[”beredar”, ”pesan”, ”siaran”, ”atau”, ”broadcast”, … ”dilakukan”, ”lewat”, ”situs”, ”hingga”, ”hari”, ”pencoblosan” ...]
penonton sepi teman ahok dikasih tiket gratis dan uang rp ribu supaya mau nonton film si ahok	[”penonton”, ”sepi”, ”teman”, ”ahok”, ”dikasih”, ”tiket”, ”gratis”, ”dan”, ”uang”, ”rp”, ”ribu”, ”supaya”, ”mau”, ”nonton”, ”film”, ”si”, ”ahok”]

Proses *tokenizing* merubah kalimat menjadi satuan kata yang dibentuk ke dalam sebuah *array* seperti yang ditunjukkan pada tabel 3.3. Kalimat dipisahkan ke dalam kata satu per satu. Setelah proses *tokenization* selesai dilakukan, berikutnya adalah melakukan tahapan *stopword removal.*

3.3.4 Stopword Removal

Pada tahap ini, istilah-istilah yang tidak relevan dengan subjek utama dari data yang digunakan akan dihilangkan, meskipun kata-kata tersebut sering muncul dalam teks yang digunakan. Kata-kata tersebut terdiri dari kata penghubung, kata depan, kata penentu, dan kata-kata lain yang sejenis. *Stopword* juga bisa dikostumisasi sesuai dengan kebutuhan datanya (Cios et al., 2007). Daftar *stopwords* dalam bahasa Indonesia yang mungkin akan digunakan adalah *dataset* publik yang didapatkan dari Devid Haryalesmana yang diunggah pada repositori Github (ID-Stopwords/id.stopwords.02.01.2016.txt at master · masdevid/ID-Stopwords · GitHub,

no date). Selain itu *dataset stopword* khusus bahasa Indonesia juga sudah disediakan di dalam *library* NLTK Python. Contoh hasil dari proses *stopword removal* ditunjukkan pada table 3.4.

Tabel 3.4 Contoh hasil *stopword removal*

Hasil *Tokenizing*	**Hasil *Stopword Removal***
[“mendiang”, ”abdullah”, ”fithri”, ”setiawan”, ”rahiimahulloh”, ”adalah”, ”karyawan”, ”tv”, ”muhammadiyah”, ”yang”, ”meliput”, ”dalam”, ”tentang”, ”mobil”, ”esemka”, ”dan”, ”kini”, ”dia”, ”rahiimahulloh”, ”dibunuh”, ”dengan”, ”sadis”]	[“mendiang”, ”abdullah”, ”fithri”, ”setiawan”, ”rahiimahulloh”, ”karyawan”, ”tv”, ”muhammadiyah”, ”meliput”, ”mobil”, ”esemka”, ”rahiimahulloh”, ”dibunuh”, ”sadis”]
[”beredar”, ”pesan”, ”siaran”, ”atau”, ”broadcast”, … ”dilakukan”, ”lewat”, ”situs”, ”hingga”, ”hari”, ”pencoblosan” ...]	[”beredar”, ”pesan”, ”siaran”, ”broadcast”, … ”situs”, ”pencoblosan” ...]
[”penonton”, ”sepi”, ”teman”, ”ahok”, ”dikasih”, ”tiket”, ”gratis”, ”dan”, ”uang”, ”rp”, ”ribu”, ”supaya”, ”mau”, ”nonton”, ”film”, ”si”, ”ahok”]	[”penonton”, ”sepi”, ”teman”, ”ahok”, ”dikasih”, ”tiket”, ”gratis”, ”uang”, ”rp”, ”ribu”, ”nonton”, ”film”, ”si”, ”ahok”]

Proses *stopword* seperti terlihat pada tabel 3.4 akan menghilangkan kata-kata seperti “yang”, “dan”, “di”, “dengan”, “saya” dan sebagainya. Dalam hal ini perlu disempurnakan atau dikostumisasi lagi isi dari *corpus* yang digunakan supaya memperoleh hasil yang lebih akurat dalam melakukan *stopword removal*. Setelah proses *stopword removal* selesai dilakukan, berikutnya adalah melakukan tahapan proses *word embedding*.

3.4 *Word Embedding*

Tahapan *word embedding* yang akan digunakan adalah *Word2Vec* khusus bahasa Indonesia yang saat ini memungkinkan adalah menggunakan corpus yang disediakan oleh FastText atau *corpus* untuk *word embedding* yang disediakan oleh Wiki. Proses ini akan merubah teks berita pada *dataset* yang dikumpulkan diubah menjadi kumpulan vektor. Tahapan *word embedding* memungkinkan kata diubah menjadi sebuah vektor yang berisi angka-angka berukuran cukup kecil untuk mengandung informasi yang lebih banyak. Informasi yang diperoleh akan cukup banyak sampai-sampai vektor yang

terbentuk akan dapat mendeteksi makna, seperti kata “marah” dan “mengamuk” itu lebih memiliki kedekatan nilai ketimbang kata “marah” dengan “bahagia”.

3.5 *Splitting Data*

Splitting Data dilakukan untuk membagi *dataset* yang sudah diubah ke dalam bentuk vektor kemudian dibagi menjadi 2 bagian, yaitu *training data* dan *testing data*. *Training Data* harus lebih banyak daripada *testing data*, hal ini disebabkan agar dalam proses pelatihan dan pembelajaran data menghasilkan akurasi dan model yang semakin baik. *Testing data* akan diujikan dengan model yang sudah dilatih untuk memeriksa keakuratan dari proses pelatihan dan pembelajaran yang sudah dilakukan.

3.6 Penerapan Metode LSTM

Tahap selanjutnya adalah menerapkan metode LSTM dari *deep learning* untuk membentuk model dalam mengidentifikasi berita palsu atau berita fakta. Hasil pemodelan yang didapat kemudian dievaluasi menggunakan beberapa teknik seperti *accuracy, precision, recall,* dan *confusion matrix*.

3.7 Penerapan Metode CNN

Setelah proses pemodelan menggunakan metode LSTM selesai dilakukan dan selesai dievaluasi, kemudian hasil pemodelan dari metode LSTM diteruskan kembali menggunakan metode lain, yaitu metode CNN sampai didapatkan hasil pemodelannya kembali. Hasil pemodelan yang didapat kemudian dievaluasi menggunakan beberapa teknik seperti *accuracy, precision, recall*, dan *confusion matrix*.

3.8 Pengujian *Input* Data Baru

Setelah melakukan evaluasi terhadap model terakhir, dilakukan pengujian dengan cara memasukkan data baru berupa berita fakta dan berita palsu yang belum pernah dilatih dan dites. Hal ini berfungsi untuk menguji keakuratan dari sebuah model dalam menentukan berita palsu dan berita fakta bahwa hasil prediksinya sudah sesuai atau belum.

3.9 Perancangan dan Implementasi Model ke dalam Sistem

Setelah metode LSTM dan metode CNN dari *deep learning* sudah diterapkan dan sudah dilakukan evaluasi, jika hasil evaluasi terakhir menunjukkan tingkat akurasi yang

tinggi, maka model dari *deep learning* yang didapat sudah optimal dan siap diimplementasikan ke dalam sebuah sistem aplikasi.

Rencana penelitian ini akan mengimplementasikan model yang sudah optimal ke dalam sebuah sistem berbasis web sederhana. Maka dari itu, tahapan berikutnya adalah membuat rancangan web yang akan dibuat. Kemudian mencoba mengimplementasikan model ke dalam web tersebut. Web tersebut rencananya dibuat agar dapat digunakan oleh masyarakat untuk mengidentifikasi berita palsu. Ide awalnya mungkin terdapat sebuah *Text Field* untuk menginput berita yang akan diidentifikasi, kemudian terdapat tombol "*Check*" yang apabila ditekan akan menghasilkan output apakah berita tersebut termasuk berita palsu atau berita fakta.

Perkembangan Penelitian

Kemajuan atau perkembangan dari peneliti terkait penelitian pada topik ini, yaitu peneliti sudah mulai mencoba untuk melakukan implementasi *deep learning* menggunakan metode CNN dan LSTM untuk menentukan berita fakta dan berita palsu dalam bahasa Indonesia. Peneliti juga sudah melakukan publikasi ilmiah pada jurnal terakreditasi nasional Sinta 4 di Jurnal Informatika Universitas Pamulang dengan status diterima (http://openjournal.unpam.ac.id/index.php/informatika/article/view/6760).

Jumlah data yang sudah dikumpulkan sebanyak 1786 berita yang terdiri dari 802 berita palsu dan 984 berita fakta. Jumlah data tersebut akan terus ditambahkan kembali. Data yang dikumpulkan berasal dari situs TurnbackHoax.id.

Pada pengujian yang sudah dilakukan, peneliti membandingkan hasil evaluasi dari model pada masing-masing metode menggunakan *accuracy train, accuracy test, precision, recall* dan *confusion matrix*. Hasil menunjukkan bahwa CNN memiliki model yang lebih baik daripada LSTM. Hasil yang diperoleh pada metode CNN adalah 0.93 untuk *accuracy train,* 0.88 untuk *accuracy test,* 0.88 untuk *precision,* 0.88 untuk *recall*, sedangkan hasil pada metode LSTM adalah 0.99 untuk *accuracy train,* 0.84 untuk *accuracy test,* 0.84 untuk *precision,* 0.83 untuk *recall*. Metode CNN dan LSTM memiliki hasil *confusion matrix* yang cukup baik karena jumlah data yang diprediksi benar jauh lebih banyak daripada jumlah data yang diprediksi salah.

Pengujian juga dilakukan dengan melakukan *input* data baru, namun data baru yang digunakan masih sedikit, yaitu hanya menggunakan 4 berita (2 berita fakta dan 2 berita palsu). Dari 4 berita, hanya 1 berita yang diprediksi salah, yaitu pada saat

pengujian data baru menggunakan metode LSTM. Oleh sebab itu, peneliti akan mencoba untuk mendapatkan model yang paling optimal menggunakan kedua metode tersebut dan kemudian diimplementasikan ke dalam sistem aplikasi sederhana.

Jadwal Penelitian

Waktu	**Kegiatan**
Bulan ke 1 – 3	Studi Literatur dan Pengumpulan Data
Bulan ke 4 – 6	*Preprocessing Data*
Bulan ke 7 – 8	*Word Embedding*
Bulan ke 9-10	*Splitting Data*
Bulan ke 11-17	Pengembangan Metode *Deep Learning* Termasuk *Training Process*, *Validation*, *Testing* dan *Evaluation*.
Bulan ke 18-24	Perancangan sistem dan implementasi model ke dalam sistem aplikasi sederhana.

Daftar Pustaka

Afriza, A., & Adisantoso, J. (2018). Metode Klasifikasi Rocchio untuk Analisis Hoax. *Jurnal Ilmu Komputer Dan Agri-Informatika*. https://doi.org/10.29244/jika.5.1.1-10

Ananth, S., Radha, D. K., Prema, S., D., & Nirajan, K. (2019). Fake News Detection using Convolution Neural Network in Deep Learning. *International Journal Of Innovative Research In Computer And Communication Engineering*, *7*(1).

Arbones, M. (2017). *Deep learning: Creating bridges between dmps in auto encoders and recurrent neural networks*. Escola TÃšcnica Superior d'Enginyeria Industrial de Barcelona.

Berghel, H. (2017). Alt-News and Post-Truths in the "Fake News" Era. *Computer*. https://doi.org/10.1109/MC.2017.104

Colah. (2015, August 27). *Understanding LSTM Networks -- colah's blog*. http://colah.github.io/posts/2015-08-Understanding-LSTMs/

Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep Learning. In *MIT Press*. MIT Press. https://www.deeplearningbook.org/

Hasil Survey Wabah HOAX Nasional 2019 | Website Masyarakat Telematika Indonesia. (2019). https://mastel.id/hasil-survey-wabah-hoax-nasional-2019/

Huang, G., Sun, Y., Liu, Z., Sedra, D., & Weinberger, K. Q. (2016). Deep networks with stochastic depth. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. https://doi.org/10.1007/978-3-319-46493-0_39

Indraloka, D. S., & Santosa, B. (2017). Penerapan Text Mining untuk Melakukan Clustering Data Tweet Shopee Indonesia. *Jurnal Sains Dan Seni ITS*. https://doi.org/10.12962/j23373520.v6i2.24419

Kemp, S. (2021). *Digital in Indonesia: All the Statistics You Need in 2021 — DataReportal – Global Digital Insights*. https://datareportal.com/reports/digital-2021-indonesia

Kurniawan, A. A., & Mustikasari, M. (2021). Implementasi Deep Learning Menggunakan Metode CNN dan LSTM untuk Menentukan Berita Palsu dalam Bahasa Indonesia. *Jurnal Informatika Universitas Pamulang*, *5*(4), 544. https://doi.org/10.32493/informatika.v5i4.6760

LeCun, Y., Bengio, Y., Hinton, G. (2015). Deep learning. nature 521 (7553): 436. *Nature*, *521*, 436–444.

Maulina, D., & Sagara, R. (2018). Klasifikasi Artikel Hoax Menggunakan Support Vector Machine Linear Dengan Pembobotan Term Frequency – Inverse Document Frequency. *Jurnal Mantik Penusa*, *2*(1).

Nowak, J., Taspinar, A., & Scherer, R. (2017). LSTM recurrent neural networks for short text and sentiment classification. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. https://doi.org/10.1007/978-3-319-59060-8_50

Nugroho, K. S. (2019, January 18). *Dasar Text Preprocessing dengan Python | by Kuncahyo Setyo Nugroho | Medium*. https://medium.com/@ksnugroho/dasar-text-preprocessing-dengan-python-a4fa52608ffe

Panjaitan, A. T. B., & Santoso, I. (2021). Deteksi Hoaks Pada Berita Berbahasa Indonesia Seputar COVID-19. *Format : Jurnal Ilmiah Teknik Informatika*, *10*(1), 76. https://doi.org/10.22441/format.2021.v10.i1.007

Prasetyo, A. R., Indriati, & Adikara, P. P. (2018). Klasifikasi Hoax Pada Berita Kesehatan Berbahasa Indonesia Dengan Menggunakan Metode Modified K-Nearest Neighbor. *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer*, *2*(12).

Putri, T. T. A., S, H. W., Sitepu, I. Y., Sihombing, M., & Silvi. (2019). Analysis and Detection of Hoax Contents in Indonesian News Based on Machine Learning. *Journal Of Informatics Pelita Nusantara*.

Radhika, K., Bindu, K. R., & Parameswaran, L. (2018). A Text Classification Model Using Convolution Neural Network and Recurrent Neural Network. *International Journal of Pure and Applied Mathematics*.

Rahutomo, F., Pratiwi, I. Y. R., & Ramadhani, D. M. (2019). Eksperimen Naïve Bayes Pada Deteksi Berita Hoax Berbahasa Indonesia. *JURNAL PENELITIAN KOMUNIKASI DAN OPINI PUBLIK*. https://doi.org/10.33299/jpkop.23.1.1805

Rasywir, E., & Purwarianti, A. (2015). Import citahttp://Eksperimen pada Sistem Klasifikasi Berita Hoax Berbahasa Indonesia Berbasis Pembelajaran Mesin. *Jurnal Cybermatika*.

Rodriguez, A. I., & Iglesias, L. L. (2019). *Fake news detection using Deep Learning*.

Utami, P. D., & Sari, R. (2018). Filtering Hoax Menggunakan Naive Bayes Classifier. *MULTINETICS*. https://doi.org/10.32722/vol4.no1.2018.pp57-61

Word2Vec - Arif R - Medium. (2018, November 29). https://medium.com/@arifromadhan19/word2vec-95c5df46e045

Zaccone, G., & Karim, M. R. (2018). *Deep Learning with TensorFlow: Explore neural networks and build intelligent systems with Python* (2nd ed.). Packt Publishing.