

## Implementasi *Deep Learning* Menggunakan Metode *CNN* dan *LSTM* untuk Menentukan Berita Palsu dalam Bahasa Indonesia

Antonius Angga Kurniawan<sup>1</sup>, Metty Mustikasari<sup>2</sup>

Fakultas Teknologi Informasi, Universitas Gunadarma, Jl. Margonda Raya 100, Depok, Jawa Barat, Indonesia, 16424

e-mail: <sup>1</sup>anggaku@staff.gunadarma.ac.id, <sup>2</sup>metty@staff.gunadarma.ac.id

Submitted Date: October 06<sup>th</sup>, 2020

Reviewed Date: December 30<sup>th</sup>, 2020

Revised Date: December 31<sup>st</sup>, 2020

Accepted Date: January 05<sup>th</sup>, 2021

### Abstract

This research aims to implement deep learning techniques to determine fact and fake news in Indonesian language. The methods used are Convolutional Neural Network (CNN) and Long Short Term Memory (LSTM). The stages of the research consisted of collecting data, labeling data, preprocessing data, word embedding, splitting data, forming CNN and LSTM models, evaluating, testing new input data and comparing evaluations of the established CNN and LSTM models. The Data are collected from a fact and fake news provider site that is valid, namely TurnbackHoax.id. There are 1786 news used in this study, with 802 fact and 984 fake news. The results indicate that the CNN and LSTM methods were successfully applied to determine fact and fake news in Indonesian language properly. CNN has an accuracy test, precision and recall value of 0.88, while the LSTM model has an accuracy test and precision value of 0.84 and a recall of 0.83. In testing the new data input, all of the predictions obtained by CNN are correct, while the prediction results obtained by LSTM have 1 wrong prediction. Based on the evaluation results and the results of testing the new data input, the model produced by the CNN method is better than the model produced by the LSTM method.

Keywords: Fake News; Indonesian Language; Deep Learning; CNN; LSTM

### Abstrak

Tujuan dari penelitian ini adalah menerapkan teknik *deep learning* untuk menentukan berita fakta dan berita palsu dalam bahasa Indonesia. Metode yang digunakan adalah *Convolutional Neural Network* (CNN) dan *Long Short Term Memory* (LSTM). Tahapan penelitian terdiri dari pengumpulan data, *labeling data*, *preprocessing data*, *word embedding*, *splitting data*, proses pembentukan model CNN dan LSTM, evaluasi, pengujian data *input* baru dan perbandingan evaluasi dari model CNN dan LSTM yang sudah terbentuk. Pengumpulan data diambil dari situs penyedia berita-berita hoaks dan berita fakta yang sudah valid, yaitu TurnbackHoax.id. Data yang digunakan berjumlah 1786 berita dengan jumlah berita fakta sebanyak 802 dan berita palsu sebanyak 984. Hasil dari penelitian ini menunjukkan bahwa metode CNN dan LSTM berhasil diterapkan untuk menentukan berita fakta dan berita palsu dalam bahasa Indonesia dengan baik. CNN memiliki tingkat *accuracy test*, *precision* dan *recall* sebesar 0.88, sedangkan model LSTM memiliki tingkat *accuracy test*, *precision* sebesar 0.84 dan *recall* sebesar 0.83. Pada pengujian *input data* baru, hasil prediksi yang didapatkan CNN semuanya adalah benar, sedangkan hasil prediksi yang didapatkan LSTM terdapat 1 prediksi yang salah. Berdasarkan hasil evaluasi dan hasil pengujian *input data* baru, model yang dihasilkan metode CNN lebih baik daripada model yang dihasilkan metode LSTM.

Kata Kunci: Berita Palsu; Bahasa Indonesia; *Deep Learning*; *CNN*; *LSTM*

### 1. Pendahuluan

Perkembangan internet sebagai sarana informasi dapat dimanfaatkan oleh seseorang untuk melakukan kejahatan seperti penipuan informasi

kepada pengguna internet. Bentuk penipuan informasi lebih dikenal dengan berita palsu atau hoaks.

Salah satu contoh kasus yang terkenal adalah Saracen, sindikat penyebar hoaks yang telah ditangkap Agustus 2017 silam. Sindikat ini menawarkan jasa profesionalnya kepada para klien untuk memfitnah lawan politiknya dengan tarif puluhan juta rupiah (Mediani, 2017). Contoh berita palsu lainnya terjadi pada Oktober 2018 di Polewali Mandar, Sulawesi Barat. Warga panik dan berbondong-bondong mengungsi setelah mendengar berita akan terjadinya tsunami meskipun sudah diperingatkan untuk tidak percaya pada isu-isu yang belum jelas sumbernya (CNNIndonesia, 2018a). Hoaks di Toraja Utara mengenai virus berbahaya juga tersebar melalui salah satu aplikasi pesan singkat, pesan tersebut berisi bahwa di Tondon terdapat pasien yang terjangkit virus Corona yang sementara diisolasi di rumahnya dan akan disiapkan ruangan khusus RSUD LakiPadada. Namun, dilansir dari makassar.tribunnews.com, Direktur RSUD LakiPadada Dr Syafari mengklarifikasi bahwa informasi tersebut adalah hoaks (Kementerian Komunikasi Dan Informatika, 2020).

Pemerintah telah melakukan berbagai upaya untuk melawan penyebaran berita palsu, salah satunya Kemenkominfo mendirikan gerakan Masyarakat Anti Hoaks untuk menganjurkan seluruh elemen masyarakat agar waspada terhadap berita palsu (CNNIndonesia, 2018b), namun berita palsu saat ini masih banyak tersebar di internet melalui berbagai bidang seperti pesan singkat, artikel berita dan media sosial. 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).

Terdapat beberapa penelitian yang sudah dilakukan untuk menganalisis berita fakta dan berita palsu. Putri et al (2019) melakukan analisis dan deteksi berita palsu menggunakan berita dalam bahasa Indonesia. Data yang digunakan berjumlah 251 artikel berita yang terdiri dari 151 artikel berita fakta dan 100 artikel berita palsu. Dalam penelitiannya, Putri et al (2019) menggunakan *machine learning* dengan beberapa metode seperti *text preprocessing* dan *feature extraction*, kemudian membandingkan 5 model algoritma *machine learning*, yaitu *Multilayer Perceptron*, *SVM (Support Vector Machine)*, *Naïve Bayes*, *Random Forest* dan *Decision Tree*. Hasil menunjukkan bahwa *Random Forest* memiliki

tingkat akurasi yang lebih baik daripada 4 algoritma lainnya, yaitu dengan akurasi sebesar 76.47% (Putri et al., 2019). Kekurangan dalam penelitian ini adalah jumlah berita yang digunakan sebagai data belum cukup banyak, sehingga proses pelatihan dan pembentukan model yang dihasilkan belum begitu optimal. Selain itu, belum adanya pengujian menggunakan input data baru terhadap model yang sudah terbentuk. Hal ini diperlukan untuk menguji seberapa akurat model yang dihasilkan dari masing-masing algoritma.

Ananth et al (2019) melakukan penelitian untuk mendeteksi berita palsu menggunakan *deep learning*. Data yang diambil berasal dari situs Kaggle.com dengan *dataset fake news* dan *dataset real news* dalam bahasa Inggris. Dalam penelitiannya, dibandingkan hasil dari teknik *machine learning* dengan teknik *deep learning*. Metode yang digunakan adalah *K-Nearest Neighbor*, *Decision Tree*, *Naïve Bayes*, *Random Forest*, *CNN (Convolutional Neural Network)* dan *LSTM (Long Short Term Memory)*. Proses pembentukan vektor yang digunakan adalah teknik *Term Frequency – Inverse Document Frequency (TF-IDF)*. Hasil membuktikan bahwa teknik *deep learning* dengan model *CNN* dan *LSTM* memiliki hasil akurasi yang lebih baik dibandingkan teknik *machine learning* dengan model *K-Nearest Neighbor*, *Decision Tree*, *Naïve Bayes*, dan *Random Forest*. Ananth et al (2019) menyatakan, ketersediaan *dataset* dan literatur untuk mendeteksi berita palsu terbatas. Lingkup masa depan yang disarankan adalah membangun sebuah sistem berdasarkan model yang sudah terbentuk sehingga dapat dimanfaatkan langsung dalam menentukan berita palsu (Ananth et al., 2019).

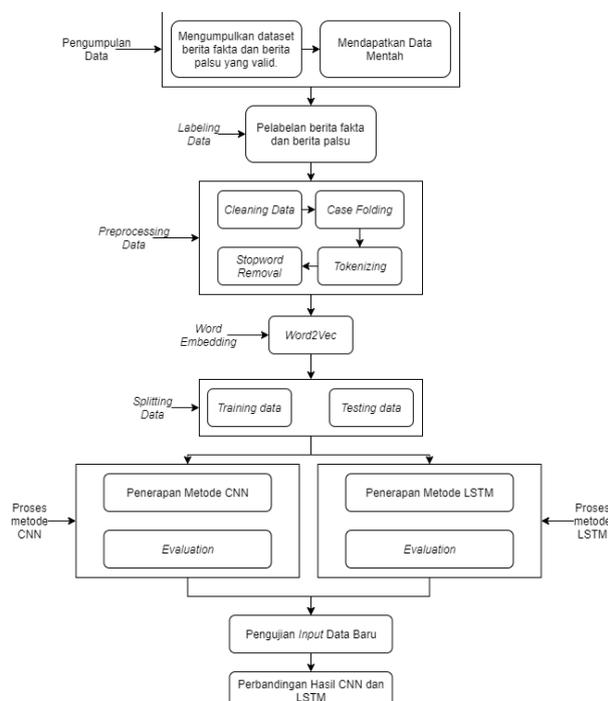
Rahutomo et al (2019) melakukan eksperimen menggunakan *Naïve Bayes* untuk mendeteksi hoaks berbahasa Indonesia. Jumlah *dataset* yang digunakan adalah 600 berita yang terdiri dari berita fakta dan berita palsu berbasis teks. Metode yang digunakan terdiri dari *preprocessing*, *manual voting tagging* untuk pelabelan berita fakta atau berita palsu, implementasi model *Naïve Bayes Classifier*, kemudian melakukan pengujian statis dan dinamis. 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) menyarankan untuk menggunakan data yang lebih banyak lagi sehingga proses pelatihan dan pembelajaran nantinya dapat

berjalan dengan baik dan mendapatkan model yang optimal. Pemilihan tahapan *preprocessing*, *feature extraction*, dan metode yang berbeda mungkin akan menghasilkan akurasi yang lebih tinggi (Rahutomo et al., 2019). Kekurangan dalam penelitian ini adalah metode yang diujikan hanya menggunakan 1 metode saja, yaitu Naïve Bayes. Pelabelan manual menggunakan *voting (manual voting tagging)* untuk berita fakta dan berita palsu perlu dikaji kembali karena dapat mempengaruhi proses pelatihan dan pembelajaran dalam pembentukan model.

Berdasarkan uraian masalah dan penelitian terdahulu, penelitian ini bertujuan untuk mengimplementasikan *deep learning* dalam menentukan berita fakta dan berita palsu menggunakan bahasa Indonesia. Model yang sudah dihasilkan kemudian diujikan kembali menggunakan data *input* berupa berita fakta dan berita palsu yang belum pernah dilatih.

## 2. Metodologi

Pada gambar 1 terdapat 8 tahapan yang dilakukan dalam penelitian ini. Delapan tahapan tersebut adalah tahapan pokok yang dilakukan di dalam penelitian ini.



Gambar 1. Tahapan Penelitian

Tahapan penelitian terdiri dari pengumpulan data, *labeling data*, *preprocessing data*, *word embedding*, *splitting data*, proses penerapan metode CNN dan LSTM, pengujian *input data*

baru, perbandingan hasil model CNN dan hasil model LSTM.

### 2.1 Pengumpulan Data

Data yang dikumpulkan berasal dari situs penyedia berita fakta dan berita palsu, yaitu Turnbackhoax.id. Situs ini dikelola oleh organisasi perkumpulan resmi yang berdiri sejak November 2016, yaitu Masyarakat Anti Hoax Indonesia (MAFINDO). Konten-konten yang diunggah pada situs ini berasal dari Forum Anti Fitnah Hasut dan Hoax (FAFH) (*Tentang Kami - TurnBackHoax*, 2016).

Setiap berita yang terdapat pada situs ini memiliki penanda yang menerangkan apakah berita tersebut merupakan berita fakta atau hoaks. Data yang dikumpulkan sebanyak 1786 berita yang terdiri dari 802 berita fakta dan 984 berita palsu. Data diambil berdasarkan berita dari periode Juni 2016 sampai dengan Desember 2018. Data sumber yang telah dikumpulkan disimpan ke dalam dokumen yang diberi nama dan format, yaitu *hoax-data.csv*.

### 2.2 Labeling Data

Proses *labeling* dilakukan untuk memberikan label pada berita yang sudah dikumpulkan dan disimpan pada *hoax-data.csv*. Tabel 1 menunjukkan contoh dari proses *labeling data*.

Tabel 1. *Labeling data*

Label	Text
0	Markas Besar Kepolisian Republik Indonesia melalui Divisi Teknologi Informasi (Div TI) Polri meluncurkan aplikasi berbasis Android dan iOS yang bernama "PolisiKu".
0	Badan Kepegawaian Negara (BKN) mengklarifikasi atas beredarnya surat pengangkatan Calon Pegawai Negeri Sipil (CPNS) di Palembang yang dilansir dari <i>bkn.go.id</i> dan <i>liputan6.com</i> merupakan surat ilegal atau palsu.
1	Saya mendukung Prabowo Subianto, karena Indonesia saat ini butuh pemimpin yang tegas dan ikhlas untuk mengangkat rakyat dari keterpurukan.

Label dibedakan menjadi 2, yaitu label dengan indeks "0" sebagai berita fakta dan label dengan indeks "1" sebagai berita palsu.

## 2.3 Preprocessing Data

Data yang sudah dimuat kemudian dipersiapkan agar lebih mudah dipelajari oleh mesin. Tahapan *preprocessing* terdiri dari 4 tahapan, yaitu *cleaning data*, *case folding*, *tokenizing*, dan *stopword removal*. Semua proses tersebut dijalankan secara berurutan.

### 2.3.1 Cleaning Data

Tujuan dari tahapan ini adalah untuk membersihkan elemen-elemen yang bisa saja mengurangi arti dari sebuah teks. Tanda baca, angka, spasi ganda, alamat URL, dan *mention* (@) adalah elemen-elemen yang dihilangkan. Tabel 2 menunjukkan contoh hasil dari proses *cleaning data*.

Tabel 2. *Cleaning data*

Data Asli	Hasil Cleaning Data
Markas Besar Kepolisian Republik Indonesia melalui Divisi Teknologi Informasi (Div TI) Polri meluncurkan aplikasi berbasis Android dan iOS yang bernama "PolisiKu".	Markas Besar Kepolisian Republik Indonesia melalui Divisi Teknologi Informasi Div TI Polri meluncurkan aplikasi berbasis Android dan iOS yang bernama PolisiKu
Badan Kepegawaian Negara (BKN) mengklarifikasi atas beredarnya surat pengangkatan Calon Pegawai Negeri Sipil (CPNS) di Palembang yang dilansir dari <a href="https://www.bkn.go.id/">https://www.bkn.go.id/</a> dan <a href="https://www.liputan6.com/">https://www.liputan6.com/</a> merupakan surat ilegal atau palsu.	Badan Kepegawaian Negara BKN mengklarifikasi atas beredarnya surat pengangkatan Calon Pegawai Negeri Sipil CPNS di Palembang yang dilansir dari dan merupakan surat ilegal atau palsu
Saya mendukung Prabowo Subianto, karena Indonesia saat ini butuh pemimpin yang tegas dan ikhlas untuk mengangkat rakyat dari keterpurukan.	Saya mendukung Prabowo Subianto karena Indonesia saat ini butuh pemimpin yang tegas dan ikhlas untuk mengangkat rakyat dari keterpurukan

Pada tabel 2 terlihat tanda baca seperti titik dan tanda garis dihilangkan. Kalimat yang mengandung alamat URL, dan angka juga ikut dihilangkan. Setelah proses *cleaning data* selesai dilakukan, berikutnya adalah melakukan tahapan *case folding*.

### 2.3.2 Case Folding

*Case folding* adalah proses di mana semua kalimat yang memiliki huruf besar akan diubah ke dalam huruf kecil. Tabel 3 menunjukkan contoh hasil dari proses *case folding*.

Tabel 3. *Case folding*

Hasil Cleaning Data	Hasil Case Folding
Markas Besar Kepolisian Republik Indonesia melalui Divisi Teknologi Informasi Div TI Polri meluncurkan aplikasi berbasis Android dan iOS yang bernama PolisiKu	markas besar kepolisian republik indonesia melalui divisi teknologi informasi div ti polri meluncurkan aplikasi berbasis android dan ios yang bernama polisiku
Badan Kepegawaian Negara BKN mengklarifikasi atas beredarnya surat pengangkatan Calon Pegawai Negeri Sipil CPNS di Palembang yang dilansir dari dan merupakan surat ilegal atau palsu	badan kepegawaian negara bkn mengklarifikasi atas beredarnya surat pengangkatan calon pegawai negeri sipil cpns di palembang yang dilansir dari dan merupakan surat ilegal atau palsu
Saya mendukung Prabowo Subianto karena Indonesia saat ini butuh pemimpin yang tegas dan ikhlas untuk mengangkat rakyat dari keterpurukan	saya mendukung prabowo subianto karena indonesia saat ini butuh pemimpin yang tegas dan ikhlas untuk mengangkat rakyat dari keterpurukan

Tidak semua teks atau kalimat konsisten dalam penggunaan huruf kecil dan kapital, oleh sebab itu perlu dilakukan pengubahan huruf kapital menjadi huruf kecil. Tahapan berikutnya adalah melakukan *tokenizing*.

### 2.3.3 Tokenizing

Proses *tokenizing* adalah proses pemisahan teks atau kalimat menjadi sebuah *token* kata. Tabel 4 menunjukkan contoh hasil dari proses *tokenizing*.

Tabel 4. *Tokenizing*

Hasil Case Folding	Hasil Tokenizing
markas besar kepolisian republik indonesia melalui divisi teknologi informasi div ti polri meluncurkan aplikasi berbasis android dan ios yang bernama polisiku	["markas", "besar", "kepolisian", "republik", "indonesia", "melalui", "divisi", "teknologi", "informasi", "div", "ti", "polri", "meluncurkan", "aplikasi", "berbasis", "android", "dan", "ios", "yang", "bernama", "polisiku"]

	"yang", "bernama", "polisiku"]
badan kepegawaian negara bkn mengklarifikasi atas beredarnya surat pengangkatan calon pegawai negeri sipil cpns di palembang yang dilansir dari dan merupakan surat ilegal atau palsu	["badan", "kepegawaian", "negara", "bkn", "mengklarifikasi", "atas", "beredarnya", "surat", "pengangkatan", "calon", "pegawai", "negeri", "sipil", "cpns", "di", "palembang", "yang", "dilansir", "dari", "dan", "merupakan", "surat", "ilegal", "atau", "palsu"]
saya mendukung prabowo subianto karena indonesia saat ini butuh pemimpin yang tegas dan ikhlas untuk mengangkat rakyat dari keterpurukan	["saya", "mendukung", "prabowo", "subianto", "karena", "indonesia", "saat", "ini", "butuh", "pemimpin", "yang", "tegas", "dan", "ikhlas", "untuk", "mengangkat", "rakyat", "dari", "keterpurukan"]

"divisi", "teknologi", "informasi", "div", "ti", "polri", "meluncurkan", "aplikasi", "berbasis", "android", "dan", "ios", "yang", "bernama", "polisiku"]	"teknologi", "informasi", "div", "ti", "polri", "meluncurkan", "aplikasi", "berbasis", "android", "ios", "bernama", "polisiku"]
["badan", "kepegawaian", "negara", "bkn", "mengklarifikasi", "atas", "beredarnya", "surat", "pengangkatan", "calon", "pegawai", "negeri", "sipil", "cpns", "di", "palembang", "yang", "dilansir", "dari", "dan", "merupakan", "surat", "ilegal", "atau", "palsu"]	["badan", "kepegawaian", "negara", "bkn", "mengklarifikasi", "beredarnya", "surat", "pengangkatan", "calon", "pegawai", "negeri", "sipil", "cpns", "palembang", "dilansir", "surat", "ilegal", "palsu"]
["saya", "mendukung", "prabowo", "karena", "indonesia", "saat", "ini", "butuh", "pemimpin", "yang", "tegas", "dan", "ikhlas", "untuk", "mengangkat", "rakyat", "dari", "keterpurukan"]	["mendukung", "prabowo", "subianto", "indonesia", "butuh", "pemimpin", "ikhlas", "mengangkat", "rakyat", "keterpurukan"]

Proses *tokenizing* merubah kalimat menjadi satuan kata yang dibentuk ke dalam sebuah *array* seperti yang ditunjukkan pada tabel 4. Kalimat dipisahkan ke dalam kata satu per satu. Setelah proses *tokenizing* selesai dilakukan, berikutnya adalah melakukan tahapan *stopword removal*.

### 2.3.4 Stopword Removal

Dalam suatu teks, sering kali terdapat kata yang sering muncul namun tidak relevan dengan subjek data pada penelitian. Oleh karena itu, pada tahap ini, kata-kata yang menjadi *stopword* dihilangkan. *Stopword* yang akan dihilangkan terdiri dari kata depan, kata penghubung, dan kata lain yang sejenis. Selain kata-kata tersebut, daftar *Stopword* juga bisa ditambahkan sesuai dengan subjek utama datanya. Pada tahap ini digunakan daftar *stopword* dalam bahasa Indonesia yang merupakan *dataset* publik dari Devid Haryalesmana yang diunggah pada repositori Github (Haryalesmana, 2016). Contoh hasil proses *stopword removal* pada penelitian ini terlihat seperti pada tabel 5.

Tabel 5. *Stopword removal*

Hasil <i>Tokenizing</i>	Hasil <i>Stopword Removal</i>
["markas", "besar", "kepolisian", "republik", "indonesia", "melalui"]	["markas", "kepolisian", "republik", "indonesia", "divisi"]

Kata-kata seperti “yang”, “dan”, “di”, “untuk”, “saya” dan sebagainya dihilangkan. 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*.

### 2.4 Word Embedding

Tahapan *word embedding* pada penelitian ini menggunakan *Word2Vec*. *Corpus* yang digunakan adalah *corpus* yang disediakan oleh Wiki untuk *word embedding* dalam bahasa Indonesia. Proses ini mengubah kata-kata di dalam teks berita pada *dataset* menjadi kumpulan vektor.

Tujuan dari *word embedding* adalah untuk menangkap dan memetakan kata-kata dalam data

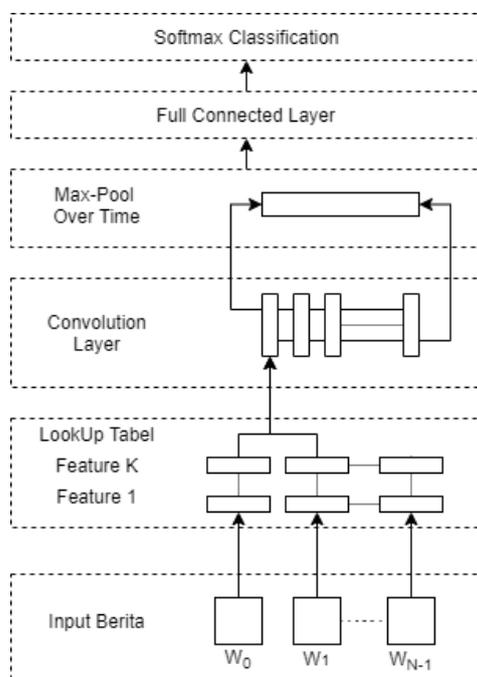
teks yang tidak berlabel ke ruang dimensi rendah yang bernilai terus menerus, untuk menangkap informasi semantik dan sintaksis internal (Li & Yang, 2017). Dalam perspektif komputasi, pada dasarnya adalah vektor yang menyimpan semua hubungan kontekstual, semantik dan sintaksis dari kata itu (Verma & Khandelwal, 2019).

## 2.5 Splitting Data

Pada tahapan ini dilakukan pemisahan data menjadi dua bagian, yaitu *training data* dan *testing data* dengan rasio 0.7:0.3. Rasio dari *training data* harus lebih banyak dibandingkan dengan rasio *testing data*. Hal ini dikarenakan agar dalam proses pelatihan dan pembelajaran data, model dapat menghasilkan akurasi yang semakin baik. Setiap proses pembelajaran atau pelatihan yang dibentuk akan diuji validitas hasilnya dan setelah itu diujikan kembali menggunakan *testing data* untuk memeriksa keakuratan dari model yang sudah dihasilkan.

## 2.6 Penerapan Metode CNN dan LSTM

Metode pertama yang diterapkan adalah metode CNN. Gambar 2 menunjukkan arsitektur CNN yang digunakan untuk memproses sebuah teks.



Gambar 2. Arsitektur CNN untuk pemrosesan teks

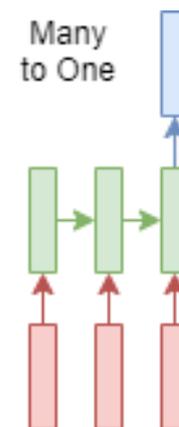
Pertama dimulai dengan berita *input* yang dipecah menjadi kata yang sudah dilakukan dengan *word2vec* pada proses *word embedding* sebelumnya. Kata-kata dipecah menjadi fitur dan

dimasukkan ke dalam lapisan *convolutional*. Hasil dari *convolution* disatukan atau dikumpulkan ke dalam angka representatif. Angka ini dimasukkan ke struktur saraf yang terhubung penuh (*full connected layer*), kemudian keputusan klasifikasi dihasilkan berdasarkan bobot yang ditetapkan untuk setiap fitur dalam teks.

Dalam CNN, teks disusun dalam sebuah matriks, dengan setiap baris mewakili *word embedding*, sebuah kata, atau karakter. *Convolution layer* CNN "memindai" teks seperti gambar, memecahnya menjadi fitur, dan menilai apakah setiap fitur cocok dengan label yang relevan atau tidak.

Pada tahap akhir, skor atau bobot yang dihasilkan adalah *input* ke dalam *full connected layer*. Bagian "*full connected*" dari jaringan CNN melewati proses *backpropagation* sendiri, untuk menentukan bobot yang paling akurat (Goodfellow et al., 2016). Setiap *neuron* menerima bobot yang memprioritaskan label yang paling tepat, dalam hal ini adalah "berita fakta" atau "berita palsu". Akhirnya, *neuron* menentukan nilai masing-masing label, dan menentukan keputusan atau hasil klasifikasi.

Metode kedua yang diterapkan adalah LSTM (*Long Short Term Memory*). LSTM dibentuk untuk memperbaiki kekurangan sebelumnya yang ada pada metode pendahulunya, yaitu RNN (Goodfellow et al., 2016). Gambar 3 menunjukkan ilustrasi sederhana dari arsitektur LSTM untuk pemrosesan teks.



Gambar 3. Arsitektur LSTM untuk pemrosesan teks

Dalam klasifikasi dokumen seperti artikel berita, arsitektur LSTM yang digunakan adalah hubungan *many-to-one*, di mana *input* adalah urutan dari kata-kata dan *output* adalah satu kelas atau label. Setiap kata-kata yang dimasukkan saling memiliki keterhubungan satu sama lain melalui

beberapa cara. Prediksi dihasilkan ketika semua kata dalam artikel yang dimasukkan sudah diproses. Cara kerja dari LSTM yaitu menggunakan *input* dari *output* terakhir yang dimasukkan sebelumnya, dengan hal tersebut LSTM dapat menyimpan informasi dengan baik, sehingga LSTM dapat memanfaatkan semua informasi yang diproses hingga membuat sebuah prediksi yang relevan, yaitu berita fakta atau berita palsu.

Proses *training* dilakukan dengan 10 *epoch*. Hal ini berarti saat semua *dataset* sudah dilakukan *training* sampai kembali ke awal sebanyak 10 putaran. Jumlah *epoch* adalah jumlah yang akan dipelajari oleh suatu model.

Setelah proses penerapan metode selesai, dilakukan evaluasi terhadap model yang dihasilkan. Evaluasi model dapat menunjukkan bahwa model yang digunakan memiliki hasil yang memuaskan atau tidak. Evaluasi yang digunakan di dalam penelitian ini adalah *accuracy train*, *accuracy test*, *precision*, *recall* dan *confusion matrix*.

*Accuracy train* digunakan untuk mengukur keakuratan model pada data yang dibuat atau data yang sudah diketahui sebelumnya. *Accuracy test* digunakan untuk mengukur akurasi model pada data yang belum pernah dilihat sebelumnya atau terpisah dari *training data*. *Precision* digunakan untuk mengukur mode prediksi positif yang benar di antara semua mode prediksi dalam kategori positif. *Recall* digunakan untuk mengukur proporsi pola positif yang diklasifikasikan dengan benar untuk semua pengamatan dalam kategori positif aktual (Hossin & Sulaiman, 2015). *Confusion matrix* menunjukkan detail yang lebih terperinci mengenai klasifikasi yang benar dan salah untuk setiap kelas (Zheng, 2015).

## 2.7 Pengujian Input Data Baru

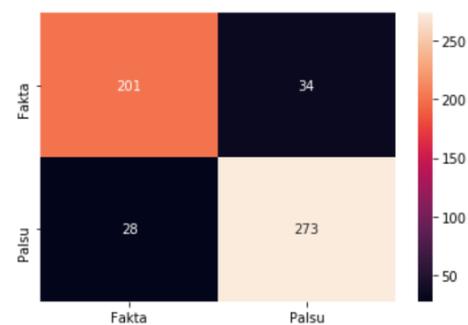
Tahapan ini dilakukan untuk menguji sebuah model dengan cara memberikan *input data* berupa teks berita yang belum pernah dilatih atau dites sebelumnya. Tujuannya adalah menguji apakah model CNN dan LSTM yang sudah terbentuk memiliki hasil yang baik atau tidak dalam mempelajari dan memprediksi data yang baru untuk menentukan berita fakta dan berita palsu. Pada penelitian ini data baru yang diujikan sebanyak 2 berita fakta dan 2 berita palsu. Setiap data baru ini disimpan ke dalam sebuah variabel di dalam program yang dibuat.

## 2.8 Perbandingan Hasil CNN dan LSTM

Model yang sudah terbentuk menggunakan CNN dan LSTM dibandingkan berdasarkan hasil evaluasinya, yaitu *accuracy train*, *accuracy test*, *precision*, *recall* dan *confusion matrix*. Hal ini dilakukan agar dapat mengetahui model terbaik untuk menentukan berita fakta dan berita palsu dalam bahasa Indonesia diantara model CNN dan model LSTM.

## 3. Hasil dan Pembahasan

Hasil evaluasi *confusion matrix* dari model CNN ditunjukkan pada gambar 4.



Gambar 4. *Confusion Matrix* dari Model CNN

*Confusion matrix* menunjukkan jumlah berita fakta yang diprediksi benar (*True Positive*) sebagai berita fakta sebanyak 201, jumlah berita fakta yang diprediksi salah (*False Positive*) sebagai berita palsu sebanyak 34, jumlah berita palsu yang diprediksi salah (*False Negative*) sebagai berita fakta sebanyak 28, kemudian jumlah berita palsu yang diprediksi benar (*True Negative*) sebagai berita palsu sebanyak 273.

Hasil evaluasi *confusion matrix* dari model LSTM ditunjukkan pada gambar 5.



Gambar 5. *Confusion Matrix* dari Model LSTM

*Confusion matrix* menunjukkan jumlah berita fakta yang diprediksi benar (*True Positive*) sebagai berita fakta sebanyak 202, jumlah berita fakta yang diprediksi salah (*False Positive*) sebagai

berita palsu sebanyak 33, jumlah berita palsu yang diprediksi salah (*False Negative*) sebagai berita fakta sebanyak 56, kemudian jumlah berita palsu yang diprediksi benar (*True Negative*) sebagai berita palsu sebanyak 245.

Tabel 6. Hasil Evaluasi Model

	<i>Acc. Train</i>	<i>Acc. Test</i>	<i>Precision</i>	<i>Recall</i>
<b>CNN</b>	0.93	0.88	0.88	0.88
<b>LSTM</b>	0.99	0.84	0.84	0.83

Tabel 6 menunjukkan hasil evaluasi model dari CNN dan LSTM. Evaluasi pada model CNN memiliki nilai *accuracy train* 0.93, *accuracy test* 0.88, *precision* 0.88, *recall* 0.88. Evaluasi pada model LSTM memiliki nilai *accuracy train* 0.99, *accuracy test* 0.84, *precision* 0.84, *recall* 0.83.

Setelah melakukan evaluasi, dilakukan pengujian dengan cara memasukkan data baru berupa berita fakta dan berita palsu yang belum pernah dilatih dan dites. Pada penelitian ini menggunakan 2 data berita fakta dan 2 data berita palsu. Tabel 7 menunjukkan hasil pengujian terhadap input data baru.

Tabel 7. Hasil Pengujian Input Data Baru

Data	CNN	LSTM
Berita Fakta 1 [0]	Berita Fakta [0]	Berita Fakta [0]
Berita Fakta 2 [0]	Berita Fakta [0]	Berita Palsu [1]
Berita Palsu 1 [1]	Berita Palsu [1]	Berita Palsu [1]
Berita Palsu 2 [1]	Berita Palsu [1]	Berita Palsu [1]

Hasil pengujian dengan model CNN memiliki hasil yang tepat, yaitu berita fakta 1 diprediksi dengan tepat sebagai berita fakta, berita fakta 2 diprediksi dengan tepat sebagai berita fakta, berita palsu 1 diprediksi tepat sebagai berita palsu dan berita palsu 2 diprediksi tepat sebagai berita palsu.

Hasil pengujian dengan model LSTM menunjukkan berita fakta 1 diprediksi tepat sebagai berita fakta, berita fakta 2 diprediksi tidak tepat sebagai berita palsu, berita palsu 1 diprediksi tepat sebagai berita palsu dan berita palsu 2 diprediksi tepat sebagai berita palsu.

Berdasarkan uraian hasil evaluasi dan pengujian pada masing-masing model menunjukkan bahwa model CNN memiliki tingkat akurasi yang lebih baik daripada model LSTM dalam menentukan fakta atau berita palsu. Hal ini dibuktikan dengan skor dari model CNN pada *accuracy test*, *precision* dan *recall* sebesar 0.88, sedangkan skor dari model LSTM pada *accuracy*

*test*, *precision* sebesar 0.84 dan *recall* sebesar 0.83. Hasil pengujian dengan data *input* baru juga menunjukkan bahwa model CNN lebih baik daripada model LSTM. Hal ini dibuktikan dari data *input* baru yang diujikan dengan model CNN sama dengan prediksi yang dihasilkan, sedangkan dengan model LSTM terdapat 1 prediksi yang tidak tepat atau salah.

#### 4. Kesimpulan

Pada penelitian ini penerapan teknik *deep learning* menggunakan metode CNN dan metode LSTM telah berhasil dilakukan dan layak diterapkan untuk menentukan berita fakta dan berita palsu dalam bahasa Indonesia. Hasil evaluasi model yang didapatkan juga sudah cukup baik untuk kedua metode, di mana nilai *accuracy*, *precision* dan *recall* dari CNN dan LSTM sudah di atas 0.8 atau sudah hampir mendekati 1. CNN memiliki hasil akurasi yang lebih baik sebesar 0.88, sedangkan LSTM sebesar 0.84.

Pengujian menggunakan data *input* baru juga berhasil dilakukan dan hasil yang di dapat cukup baik, dari 4 berita yang diujikan hanya 1 prediksi yang salah pada saat menggunakan model LSTM.

#### 5. Penelitian Masa Depan

Penelitian ini akan dikembangkan kembali menggunakan data yang lebih banyak dari sebelumnya. Pengolahan data pada tahap *preprocessing* dan *word embedding* dalam bahasa Indonesia akan lebih diperhatikan kembali agar dapat meningkatkan akurasi dari model. Pengembangan selanjutnya akan menggunakan metode CNN dan metode LSTM sekaligus, sehingga diharapkan model yang terbentuk mendapatkan hasil yang lebih optimal. Selain itu, penelitian ini akan dikembangkan kembali untuk membangun sebuah sistem web sederhana menggunakan model yang sudah optimal untuk dapat dimanfaatkan oleh pengguna internet dalam menentukan berita fakta dan berita palsu secara langsung.

#### Daftar Pustaka

- 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).
- CNNIndonesia. (2018a, October 1). *VIDEO: Warga Panik Akibat Hoaks Gempa-Tsunami di Sulbar*. <https://www.cnnindonesia.com/nasional/20181001165424-24-334695/video-warga-panik-akibat-hoaks-gempa-tsunami-di-sulbar>

- CNNIndonesia, R. (2018b, June 23). *Upaya Negara Perangi Penyebaran Berita Palsu*. <https://www.cnnindonesia.com/teknologi/20180623085115-185-308291/upaya-negara-perangi-penyebaran-berita-palsu>
- Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep Learning. In *MIT Press*. MIT Press. <https://www.deeplearningbook.org/>
- Haryalesmana, D. (2016). *ID-Stopwords/id.stopwords.02.01.2016.txt at master · masdevid/ID-Stopwords · GitHub*. <https://github.com/masdevid/ID-stopwords/blob/master/id.stopwords.02.01.2016.txt>
- Hasil Survey Wabah HOAX Nasional 2019 | Website Masyarakat Telematika Indonesia*. (2019). <https://mastel.id/hasil-survey-wabah-hoax-nasional-2019/>
- Hossin, M., & Sulaiman, M. N. (2015). A Review on Evaluation Metrics for Data Classification Evaluations. *International Journal of Data Mining & Knowledge Management Process*. <https://doi.org/10.5121/ijdkp.2015.5201>
- Kementerian Komunikasi dan Informatika*. (2020, February 15). [https://www.kominfo.go.id/content/detail/24415/hoaks-satu-warga-di-toraja-utara-terjangkit-virus-corona/0/laporan\\_isu\\_hoaks](https://www.kominfo.go.id/content/detail/24415/hoaks-satu-warga-di-toraja-utara-terjangkit-virus-corona/0/laporan_isu_hoaks)
- Li, Y., & Yang, T. (2017). *Word Embedding for Understanding Natural Language: A Survey* (Vol. 26). <https://doi.org/10.1007/978-3-319-53817-4>
- Mediani, M. (2017, August 28). *Saracen: Bisnis Hoax Hancurkan Lawan Politik*. <https://www.cnnindonesia.com/nasional/20170828063335-32-237676/saracen-bisnis-hoax-hancurkan-lawan-politik>
- 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*.
- 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>
- Tentang Kami - TurnBackHoax*. (2016). <https://turnbackhoax.id/tentang-kami/>
- Verma, P., & Khandelwal, B. (2019). Word embeddings and its application in deep learning. *International Journal of Innovative Technology and Exploring Engineering*. <https://doi.org/10.35940/ijitee.K1343.0981119>
- Zheng, A. (2015). Evaluating Machine Learning Models - O'Reilly Media. In *Oreilly*.