Implementasi Metode *Bidirectional* LSTM-CRF untuk Ekstraksi Entitas Organisasi pada Berita yang Terafiliasi Telkom University

1st Andika Aroman
Fakultas Informatika
Universitas Telkom
Bandung, Indonesia
andikaaroman@student.telkomuniversity.ac.id

2nd Donni Richasdy
Fakultas Informatika
Universitas Telkom
Bandung, Indonesia
donnir@telkomuniversity.ac.id

3rd Siti Sa'adah
Fakultas Informatika
Universitas Telkom
Bandung, Indonesia
sitisaadah@telkomuniversity.ac.id

Abstrak— Dalam Natural Language Processing (NLP), Teknologi *Named Entity Recognition* (NER) merupakan salah satu bagian dari metode NLP dan banyak dipergunakan seperti ekstraksi informasi, pencarian informasi, terjemahan mesin dan sistem penjawab pertanyaan dan lain-lain, sehingga penelitian ini berfokus pada ekstraksi informasi. *Named Entity Recognition* (NER) memiliki tujuan utama mengidentifikasi nama entitas dengan makna khusus dalam teks, terutama nama pribadi, lokasi, organisasi, waktu dan entitas-entitas lainnya. Sumber data yang digunakan adalah teks berita berbahasa Indonesia yang dilabelin secara manual dengan menggunakan beberapa tag, yaitu nama pribadi, lokasi, organisasi dan waktu. Oleh karena itu, penelitian ini menggunakan metode *Bidirectional* LSTM-CRF. *Bidirectional* LSTM memanfaatkan pra-konteks(konteks sebelumnya) dan pasca-konteks(konteks sesudahnya) dengan memproses data dari dua arah yang kemudian diklasifikasikan menggunakan CRF. Pada penelitian ini, terdapat beberapa proses yang dilakukan, yaitu *preprocessing*(*case folding*, *filtering*, *tokenization*), *labeling*, word2vec, training, testing dan proses terakhir evaluasi. Hasil penelitian ini menunjukkan bahwa metode Bidirectional LSTM-CRF untuk sistem NER terhadap teks bahasa Indonesia memperoleh hasil f1-score untuk entitas organisasi sebesar 86%. Hasil ini didasarkan pada tiga skenario pengujian, yaitu mengatur *word embedding dimensions*, *units* dan *batch sizes*.

Kata kunci— *named entity recognition, natural language processing, bidirectiona*l LSTM-CRF

I. PENDAHULUAN

Pada artikel berita terdapat informasi penting berisi berbagai informasi seperti nama pribadi, nama tempat, nama institusi dan lain-lain. Penelitian ini menggunakan sumber data berita teks berbahasa Indonesia untuk dapat memperoleh entitas organisasi yang terafiliasi Telkom University, contoh teks berita terafiliasi Telkom University adalah "Universitas Telkom membuka beasiswa bagi siswa SMA/SMK/MA yang memiliki Kartu Indonesia Pintar (KIP) dengan nama Beasiswa KIP Kuliah Merdeka. Seleksi ini dilakukan dengan menggunakan nilai rapor. Proses pendaftaran tidak memungut biaya sepeserpun alias gratis. Adapun, jadwal pendaftaran dibuka sejak 10 Mei hingga 25 Juni 2021 mendatang". Sehingga dalam kasus ini yang dapat mempermudahkan untuk mendapatkan informasi dalam teks berita adalah dengan menggunakan Named Entity Recognition.

Named Entity Recognition (NER) merupakan salah satu bagian dari *Natural Language Processing (NLP*). *NER* banyak digunakan dalam aplikasi *NLP* dan kecerdasan buatan seperti ekstraksi informasi, terjemahan mesin, pencarian informasi, penjawab pertanyaan dan lain-lain[1]. Tujuan utama *NER* adalah mengidentifikasi entitas dengan makna khusus dalam teks, terutama termasuk nama pribadi, nama tempat, nama institusi, kata benda, dan lain-lain[2]. Berdasarkan uraian tersebut tentang *NER* maka penelitian ini mengimplementasikan metode *Bidirectional LSTM (Long Short Term Memory)-Conditional Random Field (CRF)* untuk kasus ekstraksi entitas organisasi pada berita yang terafiliasi Telkom University. Metode *bidirectional LSTM-CRF* menjadi pilihan dalam penelitian ini dikarenakan *bidirectional LSTM-CRF* tersebut memiliki hasil *f1-score* cukup baik diatas 80% untuk berbagai bahasa yang telah diuji. Terbukti pada beberapa penelitian seperti penelitian oleh *Guillaume Lampe, dkk*[3], membuktikan bahwa model *bidirectional LSTM-CRF* mendapatkan *f1-score* 90,33%, penelitian serupa lainnya oleh *L. T. Anh, dkk*[4], mendapatkan hasil *precision*, *recall* dan *f1-score* dari mengkombinasikan model *bidirectional LSTM* dengan *CRF* yaitu 96,49%, 97,19% dan 96,84%. Hasil dari beberapa penelitian serupa dapat disimpulkan bahwa model *bidirectional LSTM-CRF* merupakan model yang terbaik untuk sekarang. Oleh sebab itu, metode *bidirectional LSTM-CRF* terpilih untuk menyelesaikan kasus ekstraksi entitas organisasi yang terafiliasi Telkom University.

Manfaat Teoritis dari tugas akhir ini adalah memberikan manfaat bagi setiap institusi yang ingin menggunakan tugas akhir ini untuk mencari informasi penting pada kumpulan artikel berita. Manfaat Praktis dari tugas akhir ini adalah bermanfaat bagi penulis, secara personal penulis akan mendapatkan manfaat berupa pengetahuan dan wawasan baru terkait implementasi metode *Bidirectional LSTM-CRF* pada artikel berita terkait Telkom University. Batasan masalah pada penelitian ini adalah dataset yang digunakan berupa kumpulan artikel berita berbahasa Indonesia yang membahas seputar Telkom University dengan jumlah dataset 20.061 kata, label informasi yang digunakan terbagi menjadi empat label, yaitu *Person (PER), Location (LOC), Organization (ORG), Time (TIM)* dan fokus pada penelitian ini hanya berfokus pada label *organization (ORG)*. Pengukuran performansi dari model *bidirectional LSTM-CRF* menggunakan *confusion matrix* seperti *accuracy, precision, recall* dan *f1-score.* Adapun tujuan dari penelitian ini adalah membangun model *NER* dengan metode *bidirectional LSTM-CRF* pada ekstraksi entitas organisasi

yang terafiliasi Telkom University, evaluasi performansi dan *f1-score* menggunakan metode *bidirectional LSTM-CRF* pada entitas organisasi yang terafiliasi Telkom University, dan juga mengukur pengaruh metode *bidirectional LSTM-CRF* pada entitas organisasi yang terafiliasi Telkom University. Output yang diharapkan dari penelitian ini adalah model yang digunakan dapat mengkategorikan entitas kata pada dataset dengan akurasi yang maksimal menggunakan *confusion matrix* seperti *accuracy, precision, recall* dan *f1-Score*. Berdasarkan tujuan tersebut pada penelitian ini berhasil membangun model *NER* dengan metode *bidirectional LSTM-CRF* untuk ekstraksi entitas organisasi yang terafiliasi Telkom University dengan mengolah dataset teks berita berbahasa Indonesia menghasilkan nilai *f1-score* untuk entitas organisasi diatas 80%. Pada penelitian ini telah dilakukan ketiga skenario pengujian yaitu dengan mengatur jumlah *embedding dimension*, jumlah *units* dan jumlah *batch size* pada model. Dapat disimpulkan bahwa dengan jumlah *embedding dimension* 100, jumlah *units* 50 dan jumlah *batch size* 16 merupakan hasil yang terbaik dengan hasil *precision* untuk B-ORG dan I-ORG mencapai 83% dan 89%, *recall* untuk B-ORG dan I-ORG mencapai 90% dan 74%, *f1-score* untuk B-ORG dan I-ORG mencapai 86% dan 81%. Dataset yang cukup memadai, sehingga dapat disimpulkan model *bidirectional LSTM-CRF* mampu menyelesaikan permasalahan *NER* untuk dataset teks berbahasa Indonesia dengan mendapatkan hasil yang cukup baik diatas 80%.

II. KAJIAN TEORI

Pada penelitian ini menggunakan beberapa teori, teori-teori tersebut mempermudah penjelasan mengenai alur dan skema permasalahan NER. Teori yang digunakan antara lain, *named entity recognition, word2vec, long short term memory (LSTM), bidirectional LSTM, conditional random field* dan *bidirectional LSTM-CRF*.

A. *Named Entity Recognition*

Named Entity Recognition (NER) merupakan bagian dari *Natural Language Processing (NLP). NER* banyak digunakan dalam aplikasi *NLP* dan kecerdasan buatan untuk mengklasifikasikan setiap kata seperti: orang, lokasi, organisasi, tanggal, waktu, persentase dan lain-lain[2]. Cara kerja *NER* adalah mengidentifikasi setiap entitas dan selanjutnya menetapkan kategori dari entitas nama tersebut.

B. *Word2Vec*

Word2Vec merupakan bagian dari metode word embedding yang digunakan untuk mengubah kata menjadi vektor. Word2Vec memiliki dua jenis arsitektur, yaitu *Continuous Bag-of-Words (CBOW)* dan *Skip-gram*. *Figure* 2 merupakan arsitektur *CBOW* dan *Skip-gram* sebagai berikut.

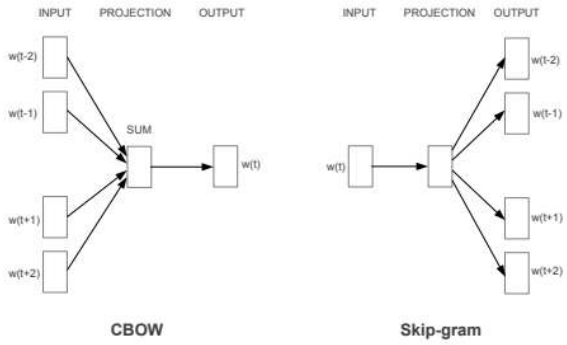

FIGURE 1
ARSITEKTUR CBOW DAN *SKIP-GRAM*[5]

Pada *figure* 1 arsitektur *CBOW* memanfaatkankonteks disekitar kata (*input*) untuk memprediksi kata dan diubah menjadi vektor (*output*) dan arsitektur *Skip-gram* memanfaatkan vektor kata *input*-an untuk memprediksi konteks disekitar kata tersebut. Pelatihan *CBOW* menggabungkan konteks-konteks kata disekitarnya untuk memprediksi kata yang berada di tengah. Pelatihan *Skip-gram* mempelajari vektor kata pada saat memprediksi konteks kata dalam kalimat yang sama. Kedua arsitektur tersebut memiliki kompleksitas yang terbilang rendah, sehingga arsitektur tersebut dapat di-*training* terhadap korpus yang berukuran besar dalam waktu yang singkat[5].

C. *Long Short-Term Memory (LSTM)*

Model Jaringan *Long Short-Term Memory (LSTM)* merupakan bagian dari metode *Recurrent Neural Network (RNN)* yang telah ditingkatkan, dengan menggunakan lapisan tersembunyi sebagai unit memori, jaringan LSTM dapat mengatasi korelasi dalam deret waktu baik dalam jangka pendek maupun jangka Panjang. Pada *figure* 2 merupakan struktur memory cell dan memiliki tiga gate, yaitu *forget gate* ($\boldsymbol{f_t}$), *input gate* ($\boldsymbol{i_t}$) dan *output gate* ($\boldsymbol{o_t}$). Selain itu, status sel ditunjukkan oleh $\boldsymbol{C_t}$, input dari setiap *gate* adalah data yang telah diproses sebelumnya $\boldsymbol{X_t}$, status sel memori sebelumnya $\boldsymbol{C_{t-1}}$, dan tanda titik biru adalah pertemuan, yang berarti perkalian dan garis dari $\boldsymbol{C_t}$ ke $\boldsymbol{o_t}$, $\boldsymbol{i_t}$, $\boldsymbol{f_t}$ untuk fungsi keadaan sebelumnya. *Input*-nya adalah data yang diketahui, dan *output*-nya adalah hasil prediksi $\boldsymbol{h_t}$[6].

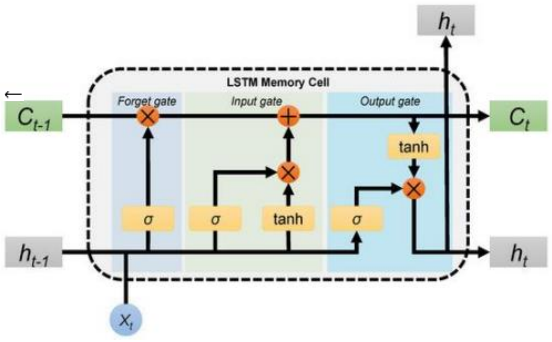

FIGURE 2
LSTM *MEMORY CALL*[7]

Proses komputasi dalam blok *LSTM* adalah sebagai berikut. Nilai *input* hanya dapat dipertahankan dalam keadaan sel jika *input gate* mengizinkannya. Nilai *input* $\boldsymbol{i_t}$ dan nilai kandidat sel memori $\boldsymbol{\tilde{C}_t}$, langkah waktu $\boldsymbol{t}$, dihitung *equation* 1 dan *equation* 2 di mana $\boldsymbol{W}$, $\boldsymbol{U}$, $\boldsymbol{b}$ masing-masing

mewakili matriks bobot dan bias.

$$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i), \quad (1)$$

$$\tilde{C}_t = \tanh(W_c x_t + U_c h_{t-1} + b_c), \quad (2)$$

Bobot unit keadaan diatur oleh *forget gate* dan nilai *forget gate* dihitung sebagai *equation* 3. Melalui proses ini, keadaan baru sel memori diperbarui sebagai *equation* 4. Dengan keadaan sel memori yang baru, nilai *output gate* dihitung sebagai *equation* 5. Nilai *output* akhir sel didefinisikan sebagai *equation* 6.

$$f_t = \sigma(W_f x_t + U_f h_{t-1} + b_f) \quad (3)$$

$$C_t = i_t \times \tilde{C}_t + f_t \times C_{t-1} \quad (4)$$

$$o_t = \sigma(W_o x_t + U_o h_{t-1} + V_o C_t + b_o) \quad (5)$$

$$h_t = o_t \times tanh(C_t) \quad (6)$$

Output sel dapat di blokir oleh *output gate*, semua *gate* menggunakan nonlinier sigmoidal dan unit keadaan dapat berfungsi sebagai masukan tambahan ke unit *gate* lainnya. Melalui proses ini, arsitektur *LSTM* dapat memecahkan masalah ketergantungan jangka panjang dengan biaya komputasi yang kecil[8].

D. *Bidirectional LSTM*

Pada *LSTM* dalam pelabelan sekuensial dapat bermanfaat jika memiliki akses kedua konteks dari sebelum dan sesudah. Namun, *hidden state* pada *LSTM* hanya mengambil konteks dari sebelumnya, sedangkan untuk konteks setelahnya tidak diketahui. Permasalahan seperti itu dapat diselesaikan dengan menggunakan *LSTM* dua arah (*bidirectional LSTM*)[9]. *Bidirectional LSTM* memanfaatkan konteks sebelumnya dan konteks sesudahnya dengan memproses data dari dua arah dengan *hidden layer* terpisah. Konteks sebelumnya direpresentasikan dengan *forward layer*, dan konteks sesudahnya direpresentasikan dengan *backward layer*[10]. Arsitektur *bidirectional LSTM* pada *figure* 3 menunjukkan adanya dua *layer LSTM* pada *output layer* yang sama sehingga juga terdapat dua *hidden layer*, yaitu $\vec{\boldsymbol{h}}$ dan $\boldsymbol{h}$. Urutan keluaran pada *forward layer*, $\vec{\boldsymbol{h}}$ dihitung secara berulang menggunakan masukan dalam urutan yang positif dari waktu ***t−1*** ke waktu ***t+1***, sedangkan pada *backward layer*, $\overleftarrow{\boldsymbol{h}}$ dihitung menggunakan masukan yang terbalik dari waktu ***t+1*** ke waktu ***t−1***. *Bidirectional LSTM* layer menghasilkan vektor keluaran, $\boldsymbol{Y_t}$, yang mana setiap elemen akan dihitung dengan menggunakan *equation* 7[11].

$$y_t = \sigma(\vec{h}, \overleftarrow{h}) \quad (7)$$

Pada *equation* 7, fungsi $\boldsymbol{\sigma}$ digunakan sebagai mengkombinasikan dua urutan output. Fungsi $\boldsymbol{\sigma}$ dapat berupa fungsi penggabungan, fungsi penjumlahan, fungsi rata-rata ataupun fungsi perkalian. Berikut ini *figure* 3 merupakan arsitektur *bidirectional LSTM*.

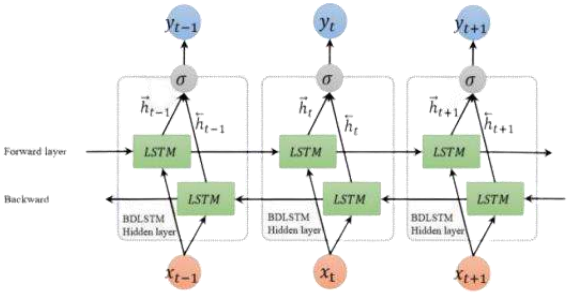

FIGURE 3
ARSITEKTUR *BIDIRECTIONAL* LSTM[11]

E. *Conditional Random Field*

Conditional Random Field (*CRF*) merupakan kerangka kerja yang digunakan dalam membangun model probabilistik yang digunakan untuk memprediksi data terstruktur. Untuk pekerjaan pelabelan urutan, *CRF* adalah salah satu model yang paling baik untuk memprediksi rantai label dari menganalisis hubungan kata[12]. Model *CRF* melakukan training untuk memprediksi sebuah vektor **y {y0,y1,y2,..,yT}** dari sebuah kalimat **x {x0, x1, x2,..xT}** dengan *equation* 8 sebagai berikut.

$$p(y|x) = \frac{e^{score(x,y)}}{\Sigma_{y'} e^{score(x,y')}} \quad (8)$$

$$score(x,y) = \sum_{i=0}^{T} A_{yi,yi+1} + \sum_{i=1}^{T} P_{i,yi} \quad (9)$$

Di mana untuk mencari ***score (x,y)*** dapat menggunakan *equation* 9 Di mana $\boldsymbol{A_{yi,yi+1}}$ merupakan probabilitas emisi yang mewakili dari tag ***i*** ke tag ***j***. $\boldsymbol{P_{i,yi}}$ merupakan probabilitas transisi yang mewakili skor transisi dari tag ***j*** ke kata ***i***.

F. *Bidirectional LSTM-CRF*

Bidirectional LSTM-CRF merupakan kombinasi antara *bidirectional LSTM* dengan *CRF*. *Bidirectional LSTM* menghitung vektor kata untuk menghasilkan *score* yang mewakili kemungkinan *tag* pada setiap kata didalam kalimat. Artinya nilai $\boldsymbol{P_{i,yi}}$ dari *equation* 9 dapat diganti dengan hasil dari *bidirectional LSTM*[10]. Sehingga *CRF* hanya menghitung nilai $\boldsymbol{A_{yi,yi+1}}$ pada *equation* 9. Berikut ini *figure* 4 merupakan arsitektur *bidirectional LSTM-CRF*.

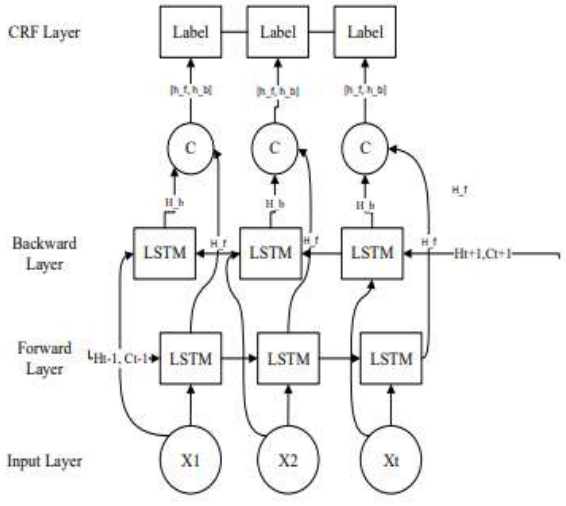

FIGURE 4
ARSITEKTUR *BIDIRECTIONAL* LSTM-CRF[13]

G. Performansi Sistem

Untuk mengevaluasi kinerja model *bidirectional LSTM-CRF* pada *NER*, pengukuran menggunakan *f1-score* lebih cocok jika dibandingkan dengan akurasi karena sebagian besar label pada data *NER* adalah label O, yang merujuk pada token yang tidak bernama entitas, dan sehingga akurasi yang tinggi dapat diperoleh. Oleh karena itu, penelitian ini akan menggunakan *f1-score* sebagai parameter untuk mengukur kinerja model[14]. Pengukuran dengan evaluasi *metrics* sebagai berikut: *Accuracy* menggambarkan seberapa akurat model dalam mengklasifikasikan dengan benar, satuan *accuracy* menggunakan satuan persen (%). Nilai *accuracy* dapat dihitung menggunakan *equation* 10 berikut ini[15].

$$A = \frac{TP+FN}{TP+FP+FN+TN} \quad (10)$$

Precision menggambarkan akurasi antara data yang diminta dengan hasil prediksi yang diberikan model, satuan *precision* menggunakan satuan persen (%). Nilai *precision* dapat dihitung menggunakan *equation* 11 berikut ini[15].

$$P = \frac{TP}{TP+FP} \quad (11)$$

Recall atau *sensitivity* menggambarkan keberhasilan model dalam menemukan kembali sebuah informasi, satuan *recall* menggunakan satuan persen (%). Nilai *recall* dapat dihitung menggunakan *equation* 12 berikut ini[15].

$$R = \frac{TP}{TP+FN} \quad (12)$$

F1-Measure menggambarkan perbandingan rata-rata *precision* dan *recall* yang di bobotkan. *Accuracy* tepat kita gunakan sebagai acuan performansi algoritma jika dataset kita memiliki jumlah data *False Negative* dan *False Positive* yang sangat mendekati *(symmetric)*. Namun jika jumlahnya tidak mendekati, maka sebaiknya kita menggunakan *f1-score* sebagai acuan. Satuan *f1-score* menggunakan satuan persen (%). Nilai *f1-score* dapat dihitung menggunakan *equation* 13 berikut ini[15].

$$F1 - Measure = \frac{2 \times Precission \times Recall}{Precission+Recall} \quad (13)$$

III. METODE

Secara garis besar, sistem perancangan *NER* dibangun pada penelitian ini terdiri dari beberapa tahapan antara lain *input* dataset, *preprocessing data (case folding, filtering, tokenization), labeling, word2vec, data split, bidirectional LSTM-CRF* dan evaluasi. Berikut adalah alur kerja sistem yang dibangun pada penelitian ini seperti pada *figure* 5.

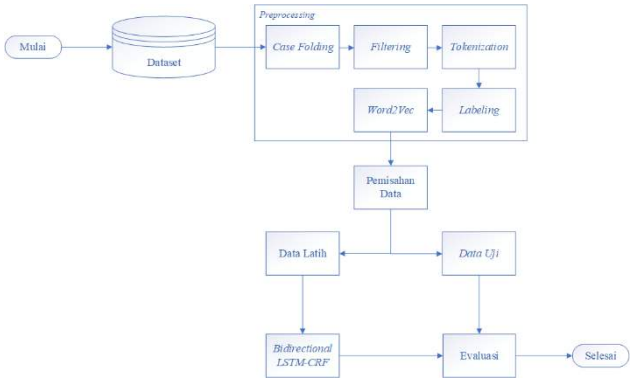

FIGURE 5
ARSITEKTUR SISTEM *BIDIRECTIONAL* LSTM-CRF

A. Dataset

Dataset yang digunakan dalam penelitian ini adalah kumpulan teks berita berbahasa Indonesia yang berhubungan dengan Telkom University. Dataset diperoleh dengan menggunakan pustaka *python "scrapy"* dengan sumber artikel berita terkait. Dataset yang diperoleh sebanyak 424 artikel, selanjutnya dilakukan *pre-processing* dengan melakukan pelabelan secara manual sehingga dataset yang terkumpul dan digunakan hanya berjumlah 51 artikel, 20.061 kata dan 1286 kalimat.

Pelabelan menggunakan Notasi *IOB*. Notasi *IOB* menunjukkan setiap *token* dengan salah satu dari tiga *tag*. Tiga *tag* yang dimaksud adalah I (*Inside*), O (*Other*), dan B (*Beginning*)[16] dan dibagi menjadi empat kategori, yaitu Orang, Organisasi, Lokasi dan Waktu. Pelabelan dilakukan untuk keempat kategori tersebut, berikut hasil pelabelan pada *table* 1.

TABLE 1
INFORMASI LABEL

Label Categories	Label
Person	B-PER
	I-PER
Organization	B-ORG
	I-ORG
Location	B-LOC
	I-LOC
Time	B-TIM
	I-TIM

1. Kata pertama dalam kalimat hanya bisa dilakukan *labeling* menggunakan *label* 'O', atau variasi dari label 'B-PER', 'B-LOC', 'B-ORG','B-TIM'.
2. *Labeling* kata yang bukan awalan dari kalimat atau kata yang berada di akhir kalimat dilabelkan berdasarkan kata sebelumnya.
3. Jika kata sebelumnya berlabel 'O' maka kata selanjutnya dapat berlabel 'O' atau variasi dari 'B-PER', 'B-LOC', 'B-ORG', 'B-TIM'.
4. Jika kata sebelumnya berlabel 'B-PER', 'B-LOC', 'B-ORG', 'B-TIM', maka kata selanjutnya dapat berlabel 'B-PER', 'B-LOC', 'B-ORG', 'B-TIM' dengan aspek berbeda, label 'I-PER', 'I-LOC', 'I-ORG', 'I-TIM'dengan aspek yang sama, atau berlabel 'O'
5. Jika kata sebelumnya berlabel 'I-PER', 'I-LOC', 'I-ORG', 'I-TIM' maka kata selanjutnya dapat dilabelkan oleh label 'B-PER', 'B-LOC', 'B-ORG', 'B-TIM' dengan aspek yang berbeda. Dan label 'I-PER', 'I-LOC', 'I-ORG', 'I-TIM' dengan aspek yang sama atau label 'O'

B. Preprocessing Data

Preprocessing merupakan tahapan penting dalam sebuah sistem yang dibangun untuk menghasilkan data yang berkualitas[17]. *Preprocessing* adalah proses yang dilakukan sebelum data diproses oleh model. *Preprocessing* juga merupakan langkah pertama dalam menyelesaikan masalah APM. Proses ini menghasilkan kata per kata yang dapat diolah dengan model seperti 'universitas', 'telkom', 'terbuka', 'beasiswa', 'berbagi', 'mahasiswa' dan lain-lain. Kata demi kata diperoleh dari tahapan-tahapan yang terjadi pada preprocessing, antara lain:

1. *Case Folding*, adalah tahapan mengubah huruf kapital menjadi huruf kecil (*lowercase*). Pelipatan huruf hanya berfungsi untuk karakter alfabet, seperti 'Universitas Telkom' hingga 'universitas telkom'.
2. *Filtering*, adalah tahapan menghilangkan simbol dari kalimat, tanda baca dan lain-lain yang non-abjad, seperti 'sejak 10 mei' hingga 'sejak mei'.
3. *Tokenizing*, adalah tahapan memisahkan kalimat menjadi kata per kata pada dataset, seperti 'universitas telkom' hingga 'universitas', 'telkom'.

C. Word2Vec

Pada tahapan ini dataset yang sudah di-*labeling* secara manual berupa *token* (kata per kata) selanjutnya diolah pada tahapan *word2vec* menjadi vektor. Tujuan dari tahapan *word2vec* agar model *bidirectional LSTM-CRF* dapat mengolah data menjadi prediksi entitas. Model *bidirectional LSTM-CRF* tidak dapat memproses langsung data yang masih berupa *token* maka tahapan ini mengubah kata menjadi vektor yang kemudian bisa diolah oleh model *bidirectional LSTM-CRF*. Berikut merupakan kata-kata yang sudah diubah menjadi vektor pada *table 2*.

TABLE 2
HASL WORD2VEC

Kata	Vektor
tentang	[-0.72703886]
telkom	[-0.7116809]
universty	[0.64588714]
universitas	[0.9009273]
swasta	[0.84986186]
indhome	[0.064022064]

D. *Bidirectional LSTM-CRF*

Pada tahapan ini model *Bidirectional LSTM-CRF* mengolah data berupa vektor. Vektor tersebut dihitung dengan model *bidirectional LSTM* selanjutnya hasil berupa *score*. *Score* tersebut mewakili *tag* pada setiap kata yang selanjutnya diklasifikasi oleh model *CRF*. Tahapan ini mengkombinasikan *Bidirectional LSTM* dan *CRF*. Berikut ini *figure 6* merupakan mendefinisikan model sebagai berikut.

```
Model: "model_1"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
input_1 (InputLayer)         (None, 213)               0
_________________________________________________________________
embedding_1 (Embedding)      (None, 213, 100)          416100
_________________________________________________________________
bidirectional_1 (Bidirection (None, 213, 100)          60400
_________________________________________________________________
time_distributed_1 (TimeDist (None, 213, 50)           5050
_________________________________________________________________
crf_1 (CRF)                  (None, 213, 10)           630
=================================================================
Total params: 482,180
Trainable params: 482,180
Non-trainable params: 0
_________________________________________________________________
```

FIGURE **6**
DEFINE MODEL BIDIRECTIONAL LSTM-CRF

IV. HASIL DAN PEMBAHASAN

Pengujian yang dilakukan pada penelitian ini menggunakan tiga skenario, yaitu dengan mengatur jumlah dimensi *word embedding*, jumlah *units* dan jumlah *batch size*. Dengan tiga skenario tersebut dapat menghasilkan hasil akurasi dan performansi maksimal dari model. Berikut *table 3* merupakan struktur model *default bidirectional LSTM-CRF* sebagai berikut.

TABLE 3
MODEL DEFAULT BIDIRECTIONAL LSTM-CRF

Parameter default	Value
Max_length	213
Embedding Dimension	213
Dense	50
Units	50
Batch Size	16
Epoch	10

Pada ketiga skenario dengan mengatur jumlah *word embedding*, *units* dan *batch size* bertujuan untuk mengetahui apakah model *default* sudah termasuk parameter terbaik. Pada *word embedding* termasuk parameter terpenting agar setiap kata dapat diproses secara efisien sesuai dengan jumlah dimensi yang diperlukan, pada *units* termasuk juga penting karena jika units berlebihan dapat menyebabkan model tidak dapat menentukan pola yang baik dan akhir terjasi *overvitting*, pada *batch size* termasuk juga penting karena *batch size* merupakan berapa data sampel yang harus diperlukan untuk model mengelolah data.

A. Skenario Pertama (*Embedding Dimension*)

Pada Skenario pertama dilakukan pengujian dengan mengatur dimensi pada *word embedding* untuk mengklasifikasikan kata pada kata yang serupa, dan mampu memahami konteks suatu kata sehingga kata-kata serupa memiliki penyematan kata rinci. Berikut *table 4* merupakan hasil pengujian yang dilakukan.

TABLE 4
HASIL EMBEDDING DIMENSION B-ORG DAN I-ORG

Embedding Dimension	B-ORG			I-ORG		
	P	R	F	P	R	F
50	79	61	61	75	74	75
100	80	84	84	66	87	75
213	82	77	77	67	67	67

300	81	80	80	83	68	75

Pada hasil pengujian di atas didapatkan dengan mengurangi jumlah *embedding dimension* mempengaruhi performansi dari *bidirectional LSTM-CRF* tersebut. Dengan mengurangi jumlah *embedding dimension* menjadi 100 membuat klasifikasi pada kata dengan dimensi menjadi efisien sehingga mempengaruhi hasil yang didapatkan, sehingga model dapat menempatkan kata sesuai dengan dimensi *word embedding* yang diperlukan, dan didapatkan hasil *f1-score* yang sangat bagus untuk entitas B-ORG dan I-ORG adalah 82% dan 75%, tetapi jika dikecilkan menjadi 50 membuat klasifikasi pada kata dengan dimensi menjadi terbatas, sehingga model menempatkan kata yang tidak sesuai dengan jumlah dimensi dan didapatkan hasil *f1-score* yang cukup rendah untuk entitas B-ORG dan I-ORG adalah 69% dan 75%. Selanjutnya, untuk jumlah dimensi *embedding* 213 dan 300 mendapatkan hasil *f1-score* yang menurun untuk entitas B-ORG adalah 79% dan 80% dan untuk entitas I-ORG adalah 67% dan 75%. Dikarenakan kinerja dari model semakin besar sehingga model tidak efisien untuk mengolah kata dan mengakibatkan penurunan hasil *f1-score*.

B. Skenario Kedua (*Units)*

Pada skenario pengujian pertama mendapatkan hasil yang terbaik dengan jumlah *embedding dimension* 100, maka pada skenario pengujian kedua ini menggunakan *embedding dimension* yang sama. Pada skenario kedua ini mangatur jumlah units sehingga menghasilkan hasil pada *table 5* sebagai berikut.

TABLE 5
HASIL UNITS B-ORG DAN I-ORG

Units	B-ORG			I-ORG		
	P	R	F	P	R	F
50	80	84	82	66	87	75
100	73	61	67	75	47	57
150	83	57	68	67	77	72
200	75	81	78	56	88	68

Pada *table* 5, berdasarkan model *default bidirectional LSTM-CRF* untuk jumlah *units* 50 mendapatkan nilai *f1-score* yang cukup baik untuk B-ORG dan I-ORG dengan hasil 82% dan 75%, dibandingkan dengan jumlah *units* 100, 150 dan 200 mengalami penurunan *f1-score* untuk B-ORG adalah 67%, 68% dan 78% sedangkan, untuk I-ORG adalah 75%, 57% dan 68%. Hal ini terjadi karena jika penambahan jumlah *units* dapat membuat model lebih rumit menentukan pola pada dataset yang dapat beresiko terjadinya *overfitting*, sehingga mempengaruhi hasil dari *f1-score*.

C. Skenario Ketiga (*Batch Size)*

Pada skenario pengujian pertama didapatkan hasil terbaik untuk *embedding dimension* yaitu 100, kemudian pada skenario kedua didapatkan hasil terbaik untuk jumlah *units* yaitu 50. sehingga pada skenario ketiga akan menggunakan hasil dari skenario pertama dan kedua. Skenario ketiga ini akan mengatur jumlah *batch size*, berikut *table* 6 merupakan hasil dari skenario ketiga.

TABLE 6
HASL BATCH SIZE B-ORG DAN I-ORG

Batch Size	B-ORG			I-ORG		
	P	R	F	P	R	F
16	83	90	86	89	74	81
32	77	76	77	61	61	58
64	84	66	74	58	58	61
128	68	29	40	0	0	0

Pada skenario pengujian terakhir semakin besar jumlah *batch size* dapat mempengaruhi performa dari model tersebut sehingga membuat hasil *f1-score* menurun drastis, *f1-score* pada *batch size* 32 dan 64 untuk B-ORG adalah 77% dan 74%, dan I-ORG adalah 58% dan 61%. Selanjutnya ketika pada *batch size* 128 terjadi loss pada I-ORG dengan hasil *f1-score* 0 dan untuk B-ORG hanya mendapatkan 40%. Dikarenakan model terlalu banyak mengolah data sampel (*batch size*) dan semakin besar jumlah *batch* size akan membuat jumlah *batch size* dengan data latih tidak cukup untuk melakukan *training* pada data yang membuat pelatihan menjadi singkat, sehingga didapatkan hasil terbaik dari jumlah *batch size* 16 dengan hasil *f1-score* untuk B-ORG dan I-ORG adalah 86% dan 81%.

D. Perbandingan dengan skenario terbaik

Telah dilakukan beberapa pengujian dan didapatkan skenario terbaik dengan menggunakan skenario pertama (mengatur jumlah word embedding dimension), skenario kedua (mengatur jumlah units) dan skenario ketiga (mengatur jumlah batch size). Berikut *figure* 7 merupakan hasil dari model *bidirectional LSTM-CRF* parameter *default*.

	precision	recall	f1-score	support
B-LOC	0.87	0.94	0.90	170
B-ORG	0.82	0.77	0.79	145
B-PER	0.97	0.96	0.96	70
B-TIM	0.96	0.94	0.95	50
I-LOC	0.96	0.92	0.94	169
I-ORG	0.67	0.67	0.67	92
I-PER	1.00	0.95	0.98	43
I-TIM	1.00	0.50	0.67	6
O	0.98	0.98	0.98	2506
PAD	1.00	1.00	1.00	44674
accuracy			1.00	47925
macro avg	0.92	0.86	0.88	47925
weighted avg	1.00	1.00	1.00	47925

FIGURE 7
HASIL MODEL *BIDIRECTIONAL* LSTM-CRF PARAMETER *DEFAULT*

Pada *figure* 7 merupakan hasil dari model *bidirectional LSTM-CRF*, dimana parameter yang digunakan yaitu *max length* 213, *embedding dimension* 213, *dense* 50, *units* 50, *batch size* 16 dan *epoch* 10. *Model default* mendapatkan hasil *f1-score* untuk entitas organisasi B-ORG sebesar 79% dan I-ORG sebesar 67%. Berbeda dengan *figure* 8 dengan skenario terbaik dengan mengubah jumlah word embedding menjadi 100 bisa mendapatkan hasil *f1-score* pada B-ORG sebesar 86% dan I-ORG sebesar 81%, dikarenakan Dengan mengurangi jumlah *embedding dimension* menjadi 100 membuat klasifikasi pada kata dengan dimensi menjadi efisien sehingga mempengaruhi hasil yang didapatkan, sehingga model dapat menempatkan kata sesuai dengan dimensi *word embedding* yang diperlukan. Berikut *figure* 8 merupakan hasil dari skenario pengujian terbaik.

	precision	recall	f1-score	support
B-LOC	0.94	0.95	0.95	203
B-ORG	0.83	0.90	0.86	122
B-PER	0.97	0.97	0.97	78
B-TIM	0.89	0.95	0.92	43
I-LOC	0.94	0.95	0.94	172
I-ORG	0.89	0.74	0.81	69
I-PER	0.95	0.96	0.96	57
I-TIM	0.80	0.57	0.67	7
O	0.99	0.99	0.99	2672
PAD	1.00	1.00	1.00	44502
accuracy			1.00	47925
macro avg	0.92	0.90	0.91	47925
weighted avg	1.00	1.00	1.00	47925

FIGURE 8
HASIL SKENARIO TERBAIK

V. KESIMPULAN

Telah dibangun model Bidirectional LSTM-CRF untuk ekstraksi entitas organisasi pada berita seputar Telkom University. Ketiga skenario pengujian yaitu dengan mengatur jumlah *embedding dimension*, jumlah *units* dan jumlah *batch size* pada model. Dapat disimpulkan bahwa dengan jumlah *embedding dimension* 100, jumlah *units* 50 dan jumlah *batch size* 16 merupakan hasil yang terbaik dengan hasil *f1-score* untuk B-ORG dan I-ORG mencapai 86% dan 81%. Dataset yang cukup memadai, model *bidirectional LSTM-CRF* mampu mendapatkan hasil yang cukup baik diatas 80%. Dapat dilihat bahwa tiga skenario pengujian tersebut diantaranya skenario pertama mengatur jumlah *word embedding dimension* mempengaruhi kinerja dari model karena mengurangi atau menambahkan jumlah *embedding dimension* membuat model terbatas atau berlebihan untuk menempatkan kata sesuai dengan jumlah dimensi, sehingga didapatkan hasil terbaik dari jumlah *embedding dimension* adalah 100, skenario kedua mengatur jumlah *units* mempengaruhi kinerja model, hal ini terjadi karena jika penambahan jumlah *units* dapat membuat model lebih rumit menentukan pola pada dataset yang dapat beresiko terjadinya *overfitting*, sehingga didapatkan hasil terbaik dari jumlah *units* adalah 50 dan skenario ketiga mengatur jumlah *batch size* mempengaruhi performa model, semakin besar jumlah *batch size* model mengalami penurunan *f1-score* dikarenakan jumlah *batch size* dengan data latih tidak cukup untuk melakukan *training* pada data yang membuat pelatihan menjadi singkat, sehingga didapatkan hasil terbaik dari jumlah *batch size* adalah 16.

Masalah yang terdapat pada penelitian ini adalah pengolahan dataset yang masih manual atau pelabelan pada kata atau entitas masih dilakukan secara manual sehingga memerlukan waktu yang cukup lama dan sulit memberikan penamaan entitas yang tepat agar sesuai dengan arti kata yang sebenarnya. Untuk penelitian selanjutnya diharapkan menggunakan dataset yang dapat diolah otomatis (tidak secara manual), dataset yang lebih banyak untuk mendapatkan hasil yang lebih baik dan memperbanyak kategori entitas dan penambahan parameter model untuk mendapatkan hasil yang lebih baik.

REFERENSI

[1] H. L. Chieu and H. T. Ng, "Named Entity Recognition: A Maximum Entropy Approach Using Global Information," *Proc. 19th Int. Conf. Comput. Linguist.*, pp. 1–7, 2002.

[2] Q. Guo, S. Wang, and F. Wan, "Research on named entity recognition for information extraction," *Proc. - 2020 2nd Int. Conf. Artif. Intell. Adv. Manuf. AIAM 2020*, pp. 121–124, 2020, doi: 10.1109/AIAM50918.2020.00030.

[3] G. Lample, M. Ballesteros, S. Subramanian, K. Kawakami, and C. Dyer, "Neural architectures for named entity recognition," *2016 Conf. North Am. Chapter Assoc. Comput. Linguist. Hum. Lang. Technol. NAACL HLT 2016 - Proc. Conf.*, no. July, pp. 260–270, 2016, doi: 10.18653/v1/n16-1030.

[4] T. A. Le, M. Y. Arkhipov, and M. S. Burtsev, "Application of a hybrid Bi-LSTM-CRF Model to the task of Russian named entity recognition," *Commun. Comput. Inf. Sci.*, vol. 789, pp. 91–103, 2018, doi: 10.1007/978-3-319-71746-3_8.

[5] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *1st Int. Conf. Learn. Represent. ICLR 2013 - Work. Track Proc.*, pp. 1–12, 2013.

[6] H. Chung and K. S. Shin, "Genetic algorithm-optimized long short-term memory network for stock market prediction," *Sustain.*, vol. 10, no. 10, 2018, doi: 10.3390/su10103765.

[7] H. Fan, M. Jiang, L. Xu, H. Zhu, J. Cheng, and J. Jiang, "Comparison of long short term memory networks and the hydrological model in runoff simulation," *Water (Switzerland)*, vol. 12, no. 1, pp. 1–15, 2020, doi: 10.3390/w12010175.

[8] Z. Zhao, W. Chen, X. Wu, P. C. Y. Chen, and J. Liu, "LSTM network: A deep learning approach for Short-term traffic forecast," *IET Intell. Transp. Syst.*, vol. 11, no. 2, pp. 68–75, 2017, doi: 10.1049/iet-its.2016.0208.

[9] X. Ma and E. Hovy, "End-to-end sequence labeling via bi-directional LSTM-CNNs-CRF," *54th Annu. Meet. Assoc. Comput. Linguist. ACL 2016 - Long Pap.*, vol. 2, pp. 1064–1074, 2016, doi: 10.18653/v1/p16-1101.

[10] M. Maimaiti, A. Wumaier, K. Abiderexiti, and T. Yibulayin, "Bidirectional long short-term memory network with a conditional random field layer for Uyghur part-of-speech tagging," *Inf.*, vol. 8, no. 4, 2017, doi: 10.3390/info8040157.

[11] Z. Cui, R. Ke, Z. Pu, and Y. Wang, "Deep Bidirectional and Unidirectional LSTM Recurrent Neural Network for Network-wide Traffic Speed Prediction," pp. 1–11, 2018, [Online]. Available: http://arxiv.org/abs/1801.02143

[12] J. Lafferty, A. Mccallum, and F. Pereira, "Conditional Random Fields : Probabilistic Models for Segmenting and Labeling Sequence Data Abstract," vol. 2001, no. June, pp. 282–289, 1999.

[13] H. Permana, "Named Entity Recognition Menggunakan Metode Bidirectional Lstm-Crf Pada Teks Bahasa Indonesia," *Univ. Komput. Indones.*, no. 112, 2019.

[14] Y. Luo, H. Zhao, and J. Zhan, "Named entity recognition only from word embeddings," *EMNLP 2020 - 2020 Conf. Empir. Methods Nat. Lang. Process. Proc. Conf.*, pp. 8995–9005, 2020, doi:

10.18653/v1/2020.emnlp-main.723.
[15] R. Rifani, M. A. Bijaksana, and I. Asror, "Named Entity Recognition for an Indonesian Based Language Tweet using Multinomial Naive Bayes Classifier," *Indones. J. Comput.*, vol. 4, no. 2, pp. 119–126, 2019, doi: 10.21108/indojc.2019.4.2.330.
[16] T. Yang *et al.*, "Chinese Data Extraction and Named Entity Recognition," *2020 5th IEEE Int. Conf. Big Data Anal. ICBDA 2020*, vol. 5, no. 2, pp. 105–109, 2020, doi: 10.1109/ICBDA49040.2020.9101204.
[17] J. Cheng and R. Greiner, "Comparing Bayesian Network Classifiers," pp. 101–108, 2013, [Online]. Available: http://arxiv.org/abs/1301.6684