Analisis Perbandingan Model CNN dan LSTM dengan Word2Vec dan BERT sebagai Representasi Teks untuk Identifikasi *Clickbait* Judul Berita Daring Berbahasa Indonesia

Dio Ahnaf Saputra - 1906352054
Widi Nugroho - 1906352161
Yudhistira Jinawi Agung - 1906354034
Venda Damianus Situmorang - 1906354192

Tugas Akhir Web Mining

Pendahuluan - Latar Belakang

- Berita atau surat kabar adalah cerita atau keterangan mengenai kejadian atau peristiwa yang hangat (KBBI, 2016).
- Seiring berkembangnya internet, informasi yang tersalurkan ke masyarakat menjadi lebih cepat. Namun ada kemungkinan informasi yang didapat merupakan informasi yang salah (*hoax*).
- Portal berita daring memperoleh penghasilan dari iklan-iklan yang ditampilkan pada halaman berita, semakin sering halaman berita dikunjungi maka semakin besar penghasilan yang didapat.
- Oleh karena itu, sering kali jurnalis menulis judul *clickbait* untuk meningkatkan *traffic*.
- Clickbait atau umpan klik adalah sesuatu yang dirancang untuk membuat pembaca ingin mengeklik suatu tautan.
- Menurut Rahadi (2017), *Clickbait* merupakan salah satu dari jenis informasi *hoax*.

Pendahuluan - *Related Works*

Penelitian tentang pendeteksian clickbait pada judul berita berbahasa Indonesia telah dilakukan oleh beberapa peneliti sebelumnya, yaitu:

- Model Convolutional Neural Network dengan metode embedding Word2Vec oleh Sinamo (2021)
- Model Long Short-Term Memory dengan metode embedding Word2Vec oleh Habibie (2018)

dengan hasil akurasi masing-masing metode 72,3% dan 82%. Convolutional Neural Network (CNN) dan Long Short-Term Memory (LSTM) merupakan metode Deep Learning.

Pendahuluan - Tujuan

- Mengidentifikasi *clickbait* pada berita daring berbahasa Indonesia menggunakan model CNN dan LSTM dengan BERT dan Word2Vec sebagai metode representasi teks
- Membandingkan kinerja model CNN dan LSTM dengan BERT dan Word2Vec sebagai metode representasi teks dalam mendeteksi *clickbait* pada judul berita daring berbahasa Indonesia berdasarkan akurasi, *F1 score*, *recall*, *precision*, dan *training time*

Pendahuluan - Batasan Masalah

- Data yang digunakan berbahasa Indonesia dengan label berupa *clickbait* atau bukan *clickbait*.
- Data yang digunakan adalah data judul berita yang bersumber dari media daring lokal. Data diperoleh dari *website* Mendeley Data.
- Kinerja model dilihat berdasarkan nilai metrik akurasi, *F1 score*, *recall*, *precision*, dan *training time*.

Metode Penelitian

1. Studi Literatur

   Studi literatur dilakukan dari hasil penelitian Sinamo (2021) tentang model CNN dengan metode *embedding Word2Vec* dan Habibie (2018) tentang model LSTM dengan metode *embedding Word2Vec*.

2. Pengumpulan Data

   Dataset diambil dari https://data.mendeley.com/datasets/k42j7x2kpn/1

3. Design dan Implementasi

Word2Vec/BERT → CNN/LSTM → Maxpooling → Dropout → Dense → Dropout → Dense

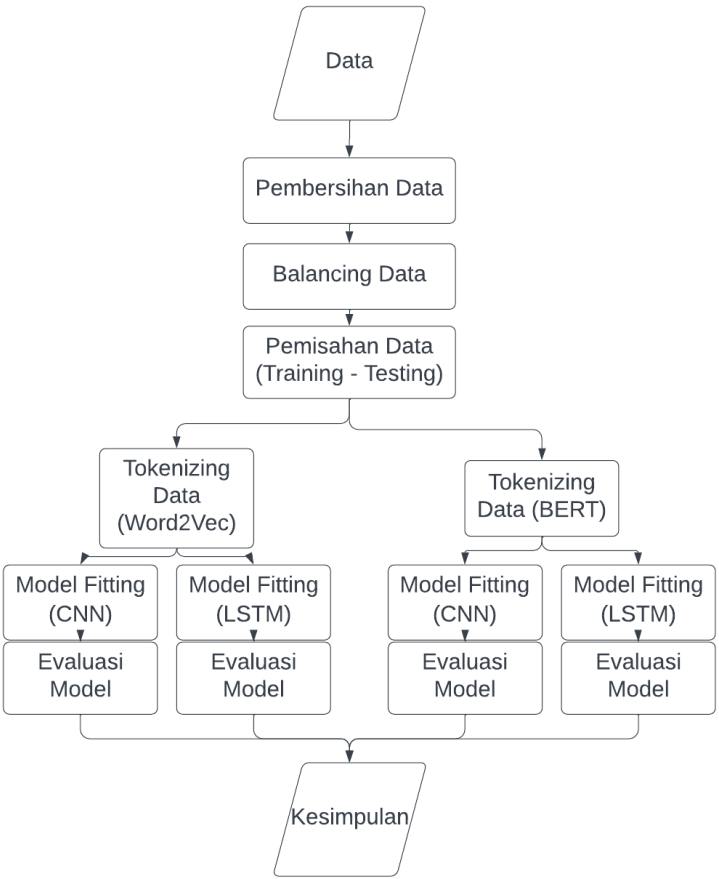
4. Analisis Hasil

   Dari hasil implementasi data ke model, maka akan didapatkan perbandingan kedua model CNN dan LSTM dengan menggunakan metode *embedding* BERT dan Word2Vec melalui metrik akurasi, *F1 score*, *recall*, *precision*, dan *training time*

Metode

- Metode Word2Vec
  - Word2Vec adalah model *shallow neural network* yang mengubah representasi kata yang merupakan kombinasi dari karakter *alphanumeric* menjadi vektor
  - Pada penelitian ini, digunakan idwiki_word2vec yang merupakan model word2vec berbahasa Indonesia yang dilatih dengan *dump* wikipedia
- Metode BERT
  - BERT adalah model NLP yang menggunakan masked language modeling yang dapat fokus dalam melihat dirinya sendiri sehingga kata-katanya memiliki makna independent.BERT kemudian mengidentifikasi kata yang sudah dimasking berdasarkan konteks
- Model CNN
  - CNN adalah model *Deep Learning* yang dikhususkan dalam pemrosesan data yang berbentuk *array*
  - CNN memiliki 3 tipe layer, yaitu *convolutional layer*, *pooling layer*, dan *fully connected layer*
- Model LSTM
  - LSTM adalah model RNN yang mampu mempelajari ketergantungan jangka panjang, terutama dalam masalah prediksi urutan.

Simulasi


Data
Pembersihan Data
Balancing Data
Pemisahan Data (Training - Testing)
Tokenizing Data (Word2Vec)
Tokenizing Data (BERT)
Model Fitting (CNN)
Model Fitting (LSTM)
Model Fitting (CNN)
Model Fitting (LSTM)
Evaluasi Model
Evaluasi Model
Evaluasi Model
Evaluasi Model
Kesimpulan

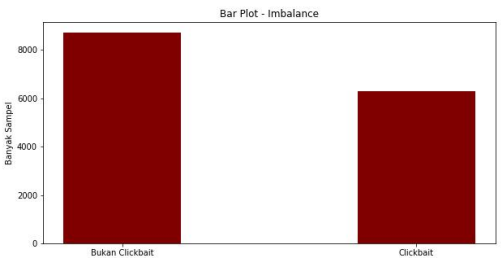
Simulasi - Data *Pre-processing*

Proses pembersihan data dengan urutannya sebagai berikut :

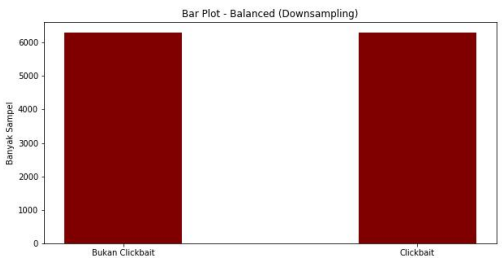
1. Mengkonversi huruf kapital ke huruf kecil
2. Menghapus spasi yang berlebih
3. Menghapus tanda baca
4. Menghapus *stopwords*

Simulasi - Data *Preprocessing*

Data akan di-*balancing* menggunakan teknik *downsampling*, dimana sebelumnya jumlah data *clickbait* sebanyak 6290 dan bukan *clickbait* sebanyak 8710



(a) Sebelum *Downsampling*

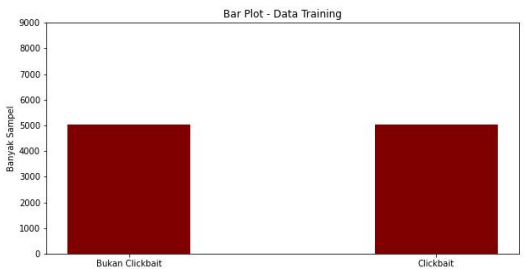


(b) Setelah *Downsampling*

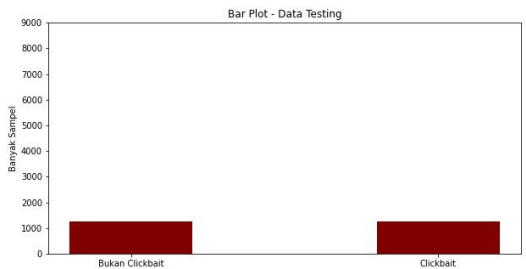
Terlihat bahwa jumlah data telah sama untuk *clickbait* dan bukan *clickbait* (sejumlah 6290 data untuk kedua jenis)

Simulasi - Data *Preprocessing*

- Setelah data dibagi menjadi dua sama banyak, selanjutnya akan dibagi dengan data *splitting*
- Pembagian data *training* 80%, data *testing* 20%



(a) Data *Training*



(b) Data *Testing*

- Untuk data *training* : *clickbait* = 5032, bukan *clickbait* = 5032, total data = 10064
- Untuk data *testing* : bukan *clickbait* = 1258, *clickbait* = 1258, total data = 2516

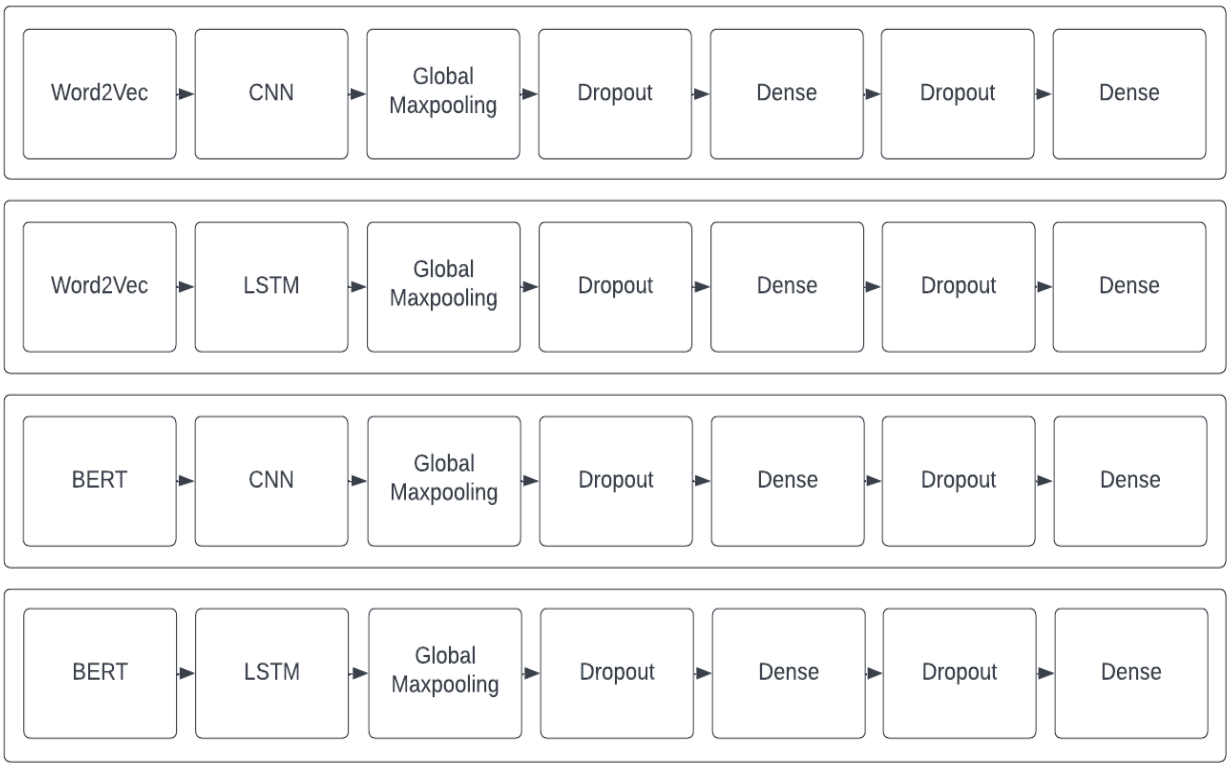
Simulasi - *Tokenizing*

- BERT
  - Tokenizer menggunakan AutoTokenizer.from_pretrained dari *library* Transformers
  - Dengan konfigurasi sebagai berikut

```
tokenizer(teks,
          add_special_tokens = True,
          max_length = max_length,
          padding = 'max_length',
          truncation = True,
          return_attention_mask = True,
          return_tensors = 'tf')
```

- Word2Vec
  - Tokenizer menggunakan wv.key_to_index pada *class model library* gensim

Simulasi - Modelling

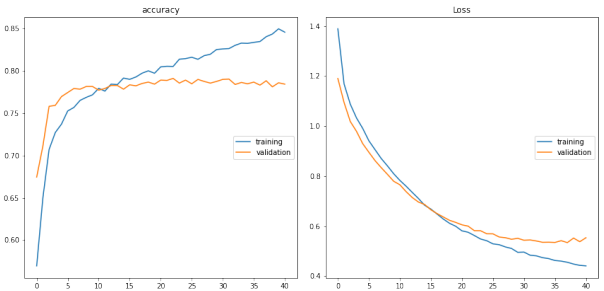


Simulasi - *Modelling*

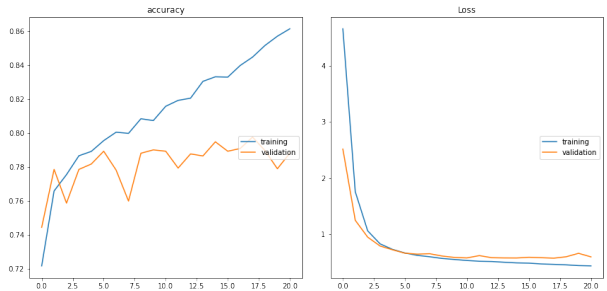
- Teknik optimasi pembaruan bobot menggunakan *optimizer* Adam
- Teknik mencari *hyperparameter* terbaik menggunakan *Bayesian Optimization* dengan jumlah percobaan sebanyak 8 kali dan *epoch* sebanyak 200.
- *Early stopping* menggunakan kriteria *validation loss* dengan jumlah *patience* = 3
- Jumlah *batch size* = 256

	Kandidat		Optimal Hyperparameter			
	Range/Value	Step	BERT - CNN	BERT - LSTM	Word2Vec - CNN	Word2Vec - LSTM
Learning Rate	5e-4, 5e-5, 3e-5, 2e-5		0.0005	5.00E-04	5.00E-04	5.00E-04
Weight Decay	1e-1, 1e-2, 1e-3, 1e-4		0.0001	0.0001	0.001	0.0001
Units Dense	(16, 256)	32	16	240	112	16
Kernel Dense	0.1, 0.01, 0.001		0.1	0.001	0.001	0.1
Filters CNN	(50, 200)	10	200		70	
Kernel CNN	(1, 3)	1	1		2	
Kernel Regularizer CNN	0.01, 0.001		0.001		0.01	
Units LSTM	(300, 600)	50		600		350
Kernel Regularizer LSTM	0.001, 0.0001, 0.00001			0.001		1.00E-04
Reccurent Regularizer LSTM	0.01, 0.001			0.01		0.01

Simulasi - Plot Metrik BERT

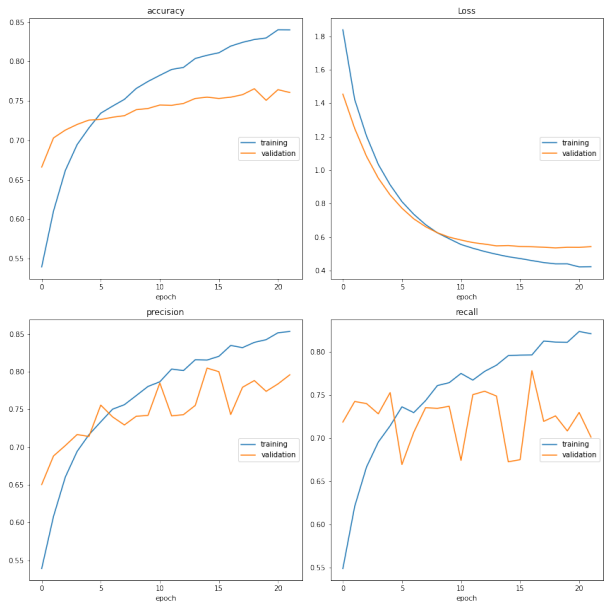


(a) Plot Metrik BERT-CNN

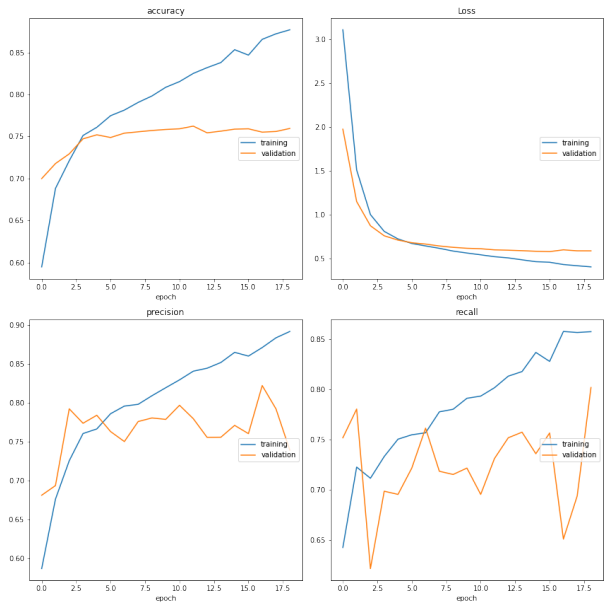


(b) Plot Metrik BERT-LSTM

Simulasi - Plot Metrik Word2Vec



(a) Plot Metrik Word2Vec-CNN



(b) Plot Metrik Word2Vec-LSTM

Simulasi - Hasil

	BERT - CNN	BERT - LSTM	Word2Vec - CNN	Word2Vec - LSTM
Accuracy	0.788155803	0.797694754	0.765500795	0.759141494
F1 score	0.787544136	0.797194167	0.765129767	0.759140125
Recall	0.788155803	0.797694754	0.765500795	0.759141494
Precision	0.791512901	0.800663268	0.767189126	0.759147389
Training Time (mm:ss)	37:22	20:43	00:57	01:05

- Terlihat bahwa model Word2Vec-LSTM paling buruk pada kriteria *accuracy*, F1 *score*, *recall*, dan *precision*
- *Training time* terlama adalah untuk model BERT-CNN yakni 37 menit 22 detik, sedangkan *training time* tercepat adalah Word2Vec-CNN yakni 57 detik. Walau model ini memiliki *training time* tercepat, tetapi model ini bukanlah model yang terbaik apabila dibandingkan dari kriteria metrik yang digunakan

Kesimpulan dan Saran

- Kesimpulan
  - Dari hasil simulasi didapat model BERT-LSTM memiliki performa terbaik, dengan akurasi sebesar 79,77% yang dicapai setelah *training* selama 20 menit 43 detik
  - Selain itu, untuk tiga kategori akurasi, presisi, serta *recall* terbaik secara berurutan adalah model BERT-CNN, Word2Vec - CNN, dan Word2Vec-LSTM
- Saran
  - Menggunakan model *hybrid*
  - Menggunakan metode *embedding* RoBERTa
- Link Notebook
  Notebook tugas dapat diakses pada link berikut:
  `https://bit.ly/NotebookKelompok6`

Daftar Pustaka

- Contextual and Non-Contextual Word Embeddings: an in-depth Linguistic Investigation (Miaschi Dell'Orletta, RepL4NLP 2020)
- Merriam-Webster. (n.d.). Clickbait. Pada Merriam-Webster.com dictionary. Diambil 2 April 2022, dari https://www.merriam-webster.com/dictionary/clickbait
- berita. 2016. Pada KBBI Daring. Diambil 2 April 2022, dari https://kbbi.kemdikbud.go.id/entri/berita
- Rahadi, Dedi Rianto. 2017. Perilaku Pengguna dan Informasi Hoax di Media Sosial. Fakultas Ekonomi dan Bisnis, Universitas Presiden.
- Sinamo, Pratiwi Rohnola Restu. 2021. Deteksi Judul Berita Clickbait Bahasa Indonesia Menggunakan Convolutional Neural Network (CNN). Fakultas Ilmu Komputer dan Teknologi Informasi, Universitas Sumatera Utara.

Daftar Pustaka

- Habibie, Ibnu. 2018. Identifikasi Judul Berita Clickbait Berbahasa Indonesia Dengan Algoritma Long Short Term Memory (LSTM) Recurrent Neural Network. Fakultas Ilmu Komputer dan Teknologi Informasi, Universitas Sumatera Utara.
- William, Andika; Sari, Yunita (2020), "CLICK-ID: A Novel Dataset for Indonesian Clickbait Headlines", Mendeley Data, V1, doi: 10.17632/k42j7x2kpn.1
- https://mti.binus.ac.id/2020/11/17/word-embedding-dengan-word2vec/