

PERBANDINGAN *PRETRAINED MODEL TRANSFORMER* PADA DETEKSI ULASAN PALSU

Aisyah Awalina^{*1}, Fitra A. Bachtiar², Fitri Utaminingrum³

^{1,2,3}Universitas Brawijaya, Malang

Email: ¹awalinaa@gmail.com, ²fitra.bachtiar@ub.ac.id, ³f3_ningrum@ub.ac.id

^{*}Penulis Korespondensi

(Naskah masuk: 19 Oktober 2021, diterima untuk diterbitkan: 02 Juni 2022)

Abstrak

Kemudahan untuk memperoleh informasi saat ini, telah sedikit membantu hidup kita. Seperti mencari ulasan untuk menimbang tempat atau barang yang akan dipilih. Beberapa orang memanfaatkan hal tersebut dengan membuat ulasan palsu untuk kepentingan mereka sendiri. Sehingga deteksi ulasan palsu sangat dibutuhkan. Model *Transformer* saat ini banyak diterapkan pada pemrosesan bahasa alami karena kinerja yang diperolehnya sangat baik. Ada dua pendekatan yang dapat dilakukan dalam model *Transformer* yaitu *pre-training* dan *fine-tuning*. Penelitian sebelumnya telah banyak menggunakan *fine-tuning* dari model *Transformer* dikarenakan adanya kemudahan dalam pelatihan, waktu yang lebih sedikit, biaya dan kebutuhan lingkungan yang lebih rendah dibanding proses *pre-training*. Akan tetapi penelitian sebelumnya masih sedikit yang membandingkan model *deep learning* dengan *fine-tuning* yang khusus diterapkan pada deteksi ulasan palsu. Penelitian ini melakukan perbandingan model *Transformer* menggunakan pendekatan *fine-tuning* dengan metode *deep learning* yaitu CNN dengan berbagai *pretrained word embedding* untuk mengatasi deteksi ulasan palsu pada dataset Ott. Model RoBERTa mengungguli model *Transformer* dan *deep learning* dimana nilai akurasi 90,8%; *precision* 90%; *recall* 91,8% dan *f1-score* 90,8%. Namun dari segi waktu komputasi model pelatihan, DistilBERT memperoleh waktu komputasi terkecil yaitu dengan nilai 200,5 detik. Meskipun begitu, hasil yang diperoleh model *Transformer* maupun *deep learning* memiliki kinerja yang baik untuk deteksi ulasan palsu pada dataset Ott.

Kata kunci: ulasan palsu, transformer, deep learning

COMPARISON OF *PRETRAINED TRANSFORMER MODELS* ON SPAM REVIEW DETECTION

Abstract

The ease of obtaining information today has helped our lives, like looking for reviews to weigh the place or item to choose. Some people take advantage of this by creating spam reviews for their benefit. So the detection of spam reviews is needed. Transformer models are currently widely applied to natural language processing because they have outstanding performance. Two approaches in the Transformer model is pre-training and fine-tuning. Previous studies have used a lot of fine-tuning due to the ease of training, less time, costs, and lower environmental requirements than the pre-training process. However, a few previous studies compare deep learning models with fine-tuning applied explicitly for detecting spam reviews. This study compares the Transformer model using a fine-tuning approach with a deep learning method, namely CNN, which uses various pre-trained word embedding to overcome the detection of false reviews in the Ott dataset. The result is RoBERTa model outperforms between Transformer and deep learning models, where the accuracy is 90.8%, precision is 90%, recall is 91.8%, and f1-score is 90.8%. Afterward, DistilBERT models obtained the shortest computation time with 200.5 seconds. However, the results obtained by both Transformer and deep learning models perform well to detect spam reviews in the Ott dataset.

Keywords: spam review, transformer, deep learning

1. PENDAHULUAN

Kemajuan teknologi saat ini mempermudah hidup kita dalam berbagai cara. Seperti ketika kita bingung untuk mencari tempat untuk menginap,

makan ataupun berbelanja, kita dapat mencarinya di internet. Terkadang orang akan menimbang tempat atau barang yang dipilihnya, hal yang dapat dilakukan adalah melihat ulasan yang ada. Dari ulasan tersebut,

kita mendapatkan beberapa informasi mengenai produk, pelayanan yang sebelumnya kita belum mengetahui kualitas dan kinerja dari produk maupun pelayanan tersebut (Kim et al., 2021).

Ulasan-ulasan tersebut dapat dimanfaatkan untuk kepentingan yang kurang baik oleh pihak yang tidak bertanggung jawab. Sebagai contoh, para penjual atau pesaing dari sebuah usaha dapat memanfaatkan hal tersebut dengan membuat ulasan palsu. Pada penelitian He et al. (2021) menyebutkan bahwa ulasan palsu dapat berpengaruh dalam minggu pertama. Hal tersebut dipicu karena penjual ingin cepat mendapatkan keuntungan di awal dan produk yang dijual cepat laku. Sedangkan, di sisi lain pesaing dapat memberikan ulasan yang bersifat menjatuhkan sehingga para pelanggan tidak percaya terhadap para penjual. Dari survei yang dilakukan pada laman *BrightLocal* pada tahun 2020 (Murphy, 2020), ulasan yang bersifat positif dapat memicu pembeli untuk menggunakan produk atau layanan mereka sebesar 94%. Sedangkan, ulasan yang bersifat negatif dapat menyebabkan pelanggan tidak menggunakan produk atau layanan mereka sebesar 92%. Dari survei yang dilakukan *brightlocal* terlihat bahwa ulasan dapat menguntungkan para penjual untuk meningkatkan penjualannya, dan pesaingnya untuk dapat menjatuhkannya. Oleh karena itu, deteksi ulasan palsu ini sangat penting dilakukan agar tidak merugikan pihak manapun.

Penelitian-penelitian dalam deteksi ulasan palsu telah banyak dilakukan, diantaranya penelitian Ott et al. (2011) dimana ada 4 pendekatan yang dilakukan yaitu identifikasi genre, *psycholinguistic* dengan LIWC, kategorisasi teks dengan *N-grams* serta menggunakan *classifier* seperti Naïve-Bayes dan SVM. Hasil yang diperoleh pendekatan dengan menggunakan kategorisasi teks yaitu fitur *bigram* dan LIWC dengan *classifier* SVM memperoleh akurasi tertinggi yaitu dengan nilai 89,8%. Selanjutnya ada penelitian dari Mukherjee et al. (2011) melakukan deteksi ulasan palsu menggunakan dataset di dunia nyata yaitu dari laman Yelp. Ada dua pendekatan yang digunakan dalam penelitian yaitu menggunakan fitur *N-gram* dan *behavioral*. Hasil tertinggi didapatkan pada fitur gabungan *bigram* dan *behavioral* dimana mendapatkan akurasi tertinggi 84,8% pada dataset hotel dan 86,1% pada dataset restoran.

Pendekatan lain menggunakan *deep learning* untuk deteksi ulasan palsu juga telah banyak dilakukan. Penelitian Zhang et al. (2018) melakukan pendekatan dengan model *skip-gram* untuk proses *word embedding* nya dengan model pembelajarannya dengan *recurrent convolutional network* yang disebut model DRI-RCNN. DRI-RCNN mendapatkan hasil yang bagus dibandingkan dengan metode *state-of-the-art* lainnya. Hasil akurasi tertinggi didapatkan ketika data latih yang digunakan sebanyak 90% dengan nilai akurasi 86%. Penelitian dari Hajek et al. (2020) dimana dalam penelitian ini menggunakan

skip-gram model dengan *Word2Vec*, lalu *N-gram* dan fitur sentimen dengan *lexicon-based* dimana proses pembelajarannya menggunakan 2 model *deep learning* yaitu DFFNN (*Deep Feed Forward Neural Network*) dan CNN. Hasil yang diperoleh dengan menggabungkan semua fitur yang ada, hasil akurasi pada metode DFFNN maupun CNN mengalami peningkatan. Hasilnya lebih unggul jika dibandingkan dengan metode *baseline* dan *state-of-the-art* lainnya.

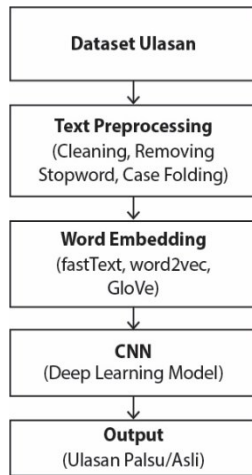
Pendekatan dengan model *Transformer* juga telah dilakukan pada deteksi ulasan palsu, namun masih jarang ditemui. Salah satu penelitian dengan model *Transformer* yaitu penelitian dari Kennedy (2019) yang menggunakan dua dataset yaitu dataset OpSpam (Ott et al., 2011) dan Yelp (Rayana dan Akoglu, 2015). Penelitian ini melakukan beberapa pendekatan model diantaranya dengan model *non-neural* (*Logistic Regression* dan *Support Vector Machine*), model *neural network* (*Feed Forward Neural Network*, *Convolutional Neural Network* dan *Long Short-Term Memory*) dan model *Transformer* (BERT). Hasilnya model *Transformer* dengan dataset OpSpam memperoleh hasil akurasi tertinggi yaitu dengan nilai 90,5%.

Model *Transformer* sendiri saat ini banyak diterapkan pada pemrosesan bahasa alami karena memiliki kinerja yang sangat baik untuk mengatasi masalah. Ada dua pendekatan yang dapat dilakukan dalam model *Transformer* yaitu *pre-training* dan *fine-tuning*. *Pre-training* sendiri merupakan sebuah proses inisialisasi bobot pada proses pelatihan data yang tidak berlabel. Sedangkan *fine-tuning* adalah proses pelatihan dengan model *pretrained* yang telah dibuat. *Fine-tuning* telah banyak dilakukan di beberapa penelitian, karena memiliki beberapa kelebihan seperti kemudahan pelatihan, waktu yang lebih sedikit, biaya dan kebutuhan lingkungan yang lebih rendah dibandingkan dengan proses *pre-training* (HuggingFace, 2021).

Meskipun sudah diketahui kelebihan dibandingkan dengan metode yang lain, penelitian yang spesifik membandingkan penggunaan model *deep learning* dengan *fine-tuning pretrained* model *Transformer* pada deteksi ulasan palsu belum banyak dilakukan. Oleh karena itu, jarang penelitian deteksi ulasan palsu pada model *Transformer*, sehingga penelitian ini melakukan perbandingan *pretrained* antara model *Transformer* dan *deep learning* untuk mengetahui kinerja dari masing-masing *pretrained* model dalam deteksi ulasan palsu. Pendekatan *fine-tuning* digunakan pada model *Transformer*. Adapun *pretrained* model *Transformer* yang digunakan masih tergolong cukup baru, diantaranya yaitu BERT, XLNet, ALBERT, RoBERTa, dan DistilBERT. Sedangkan model *deep learning* nya menggunakan model CNN dengan *pretrained word embedding* yang telah disediakan secara publik.

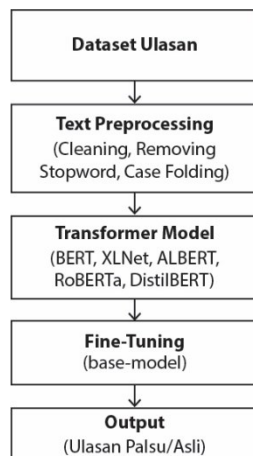
2. METODE PENELITIAN

Adapun gambaran umum yang akan dilakukan penelitian ini, dapat dilihat pada Gambar 1 dan Gambar 2.



Gambar 1 Gambaran Umum Sistem untuk Model Deep Learning

Berdasarkan Gambar 1 dapat dilihat pendekatan yang dilakukan pada model *Deep*. Tahap pertama yang dilakukan adalah *text preprocessing*, dimana proses yang dilakukan adalah *cleaning*, *removing stopwords* dan *case folding*. Selanjutnya adalah *word embedding*, dimana model *deep learning* menggunakan 3 *pretrained word embedding* yaitu *fastText*, *Word2Vec*, dan *GloVe*. Setelah *word embedding*, selanjutnya pada model *deep learning* menggunakan model CNN untuk proses pelatihan dan klasifikasi.



Gambar 2 Gambaran Umum Sistem untuk Model Transformer

Berdasarkan Gambar 2 dapat dilihat pendekatan yang dilakukan pada model *Transformer*. Tahap pertama yang dilakukan adalah *text preprocessing*, detailnya sama seperti model *deep learning*. Selanjutnya proses *fine-tuning* dengan *pretrained base-model* yang dimiliki oleh BERT, XLNet, ALBERT, RoBERTa, DistilBERT. Setelah itu

dilanjutkan dengan *dense layer* yang dibantu dengan *library* dari *Hugging Face* yaitu *Sequence Classification*.

2.1. Data yang digunakan

Dataset yang digunakan berasal dari penelitian Ott et al. (2011; 2013) dimana memiliki 1600 data yang berasal dari ulasan 20 hotel di Chicago. Adapun detail dataset tersebut 400 data kelas ulasan asli dengan sentimen positif, 400 data kelas ulasan palsu dengan sentimen positif, 400 data kelas ulasan asli dengan sentimen negatif dan 400 data kelas ulasan palsu dengan sentimen negatif. Dataset tersebut dapat diakses pada laman <https://myleott.com/op-spam.html>. Dataset yang telah dilakukan *text-preprocessing* selanjutnya akan dibagi menjadi data latih dan uji dengan proporsi 80%:20%.

2.2. Text-preprocessing

Adapun tahap *text-preprocessing* yang dilakukan dalam penelitian ini:

a. Cleaning

Dalam proses *cleaning* menghapus kata-kata yang bukan termasuk dalam kata alfabet maupun angka seperti '!', '#', '%', '&', '\$' dan lain-lain. Contoh proses *cleaning* dapat dilihat pada Tabel 1.

Tabel 1. Proses *Cleaning*

Teks:
Highly recommended! Best service!
Teks pasca proses:
Highly recommended Best service

b. Removing Stopwords

Stopwords yang digunakan berasal dari *library nltk* yang berbahasa inggris. Proses ini menghapus beberapa kata yang dianggap kurang penting dalam deteksi. Beberapa contoh kata pada *stopwords* adalah "a", "an", "the", "of", "in", dan lain-lain. Contoh proses *removing stopwords* dapat dilihat pada Tabel 2.

Tabel 2. Proses *Removing Stopwords*

Teks:
Very nice place
Teks pasca proses:
nice place

c. Case Folding

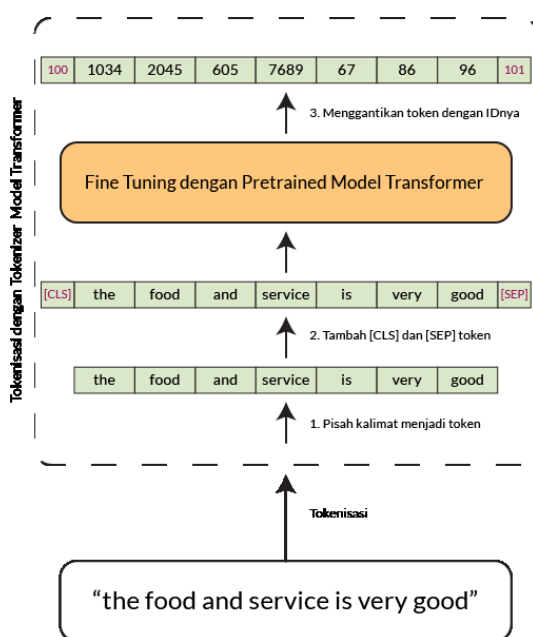
Dalam proses *case folding* dilakukan pengubahan seluruh karakter huruf menjadi huruf kecil (*lower case*). Contoh proses *case folding* dapat dilihat pada Tabel 3.

Tabel 3. Proses *Case Folding*

Teks:
<i>The breakfast food is delicious</i>
Teks pasca proses:
<i>the breakfast food is delicious</i>

2.3. Model Transformer dan Deep Learning

Pada Model Transformer terdapat dua pendekatan yaitu *pre-training* dan *fine-tuning*. Dalam *pre-training* dilakukan dengan melatih model dengan data yang tidak berlabel sedangkan *fine-tuning* dilakukan dengan inisialisasi dari parameter dari *pre-trained* dan *fine-tuning* dari data yang telah berlabel. Pendekatan *fine-tuning* yang terpilih pada penelitian ini. Ilustrasi proses *fine-tuning* dapat dilihat pada Gambar 3.

Gambar 3 Ilustrasi pada Model Transformer *Tokenizer*

Berdasarkan Gambar 3, kalimat ulasan yang dicontohkan tersebut dilakukan tokenisasi dengan bantuan *tokenizer* masing-masing model Transformer dimana dalam penelitian ini menggunakan *library* dari *Hugging Face*. Proses pertama yaitu dilakukan proses kalimat menjadi token. Setelah proses tersebut, dilakukan penambahan token [CLS] di awal dan token [SEP] di akhir. Setelah itu dilakukan proses *fine-tuning* pada *pretrained* yang dipilih masing-masing Model Transformer. Setelah proses tersebut, didapatkanlah ID yang nantinya diteruskan kepada *fully connected layer*. Adapun *pretrained* masing-masing Model Transformer akan dijelaskan lebih lanjut pada masing-masing Subbab.

BERT

BERT (Bidirectional Encoder Representations from Transformers) merupakan model *Transformer* dari penelitian Devlin et al. (2019). BERT sendiri memberikan hasil yang sangat baik dalam mengatasi

masalah pemrosesan bahasa alami. Namun BERT juga tetap memiliki kekurangan, sehingga penelitian ini juga melakukan perbandingan pada model yang dibangun berdasarkan model BERT. *Pre-trained* dari model BERT telah banyak disediakan secara publik. *Pre-trained* yang akan digunakan pada penelitian ini adalah *BERT-base-cased* yang dapat diakses pada laman *huggingface*. *BERT-base-cased* sendiri memiliki 12 *layer*, 768 *hidden layer*, 12 *head layer* serta 109 juta parameter dimana merupakan model BERT untuk teks berbahasa Inggris.

XLNet

Penelitian Yang et al. (2019) mengusulkan model XLNet dimana merupakan sebuah metode *generalized autoregressive* yang mengambil kelebihan dari bahasa pemodelan *autoregressive* dan *autoencoding* dengan menghindari kekurangan yang dimiliki keduanya. Model XLNet sendiri telah banyak mengungguli BERT di berbagai masalah, seperti *question answering*, analisis sentimen, dan lain-lain. Model XLNet menyebutkan bahwa kekurangan BERT adalah mengabaikan ketergantungan dari posisi *masked* dan kerumitan dari perbedaan *pretrain-finetune*. Sehingga XLNet menggunakan pendekatan *autoregressive* untuk mengatasi masalah BERT. Dalam penelitian ini, menggunakan model *pretrained* yang telah tersedia yaitu XLNet-base-cased yang memiliki 12 *layer*, 768 *hidden layer*, 12 *head layer* serta 110 juta parameter dimana proses *pre-training* dilakukan pada teks berbahasa Inggris.

ALBERT

ALBERT (A Lite BERT) merupakan model *Transformer* yang memiliki parameter lebih sedikit dibandingkan dari BERT. Model yang diusulkan dari penelitian Lan et al. (2019) ini memiliki konfigurasi yang hampir sama dengan BERT-large dengan parameter 18 kali lebih kecil dan melatih 1,7 kali lebih cepat. Model ALBERT sendiri bertujuan untuk menurunkan konsumsi memori dan meningkatkan kecepatan pelatihan dari BERT. Penelitian deteksi ulasan palsu dengan ALBERT juga masih sangat sedikit. Dalam penelitian ini menggunakan model *pretrained* yang telah tersedia yaitu ALBERT-Base-v1 dimana memiliki 12 *layer* yang berulang, 128 *embedding layer*, 768 *hidden layer*, 12 *head layer*, 11 juta parameter.

RoBERTa

RoBERTa (Robustly Optimized BERT) merupakan studi replikasi dari BERT (2019) dimana memperhatikan dengan hati-hati efek dari *hyperparameter* dan ukuran dari data latih. RoBERTa memilih beberapa set yang dianggap penting dalam desain dan strategi pelatihan yang dapat mengungguli dari segi kinerja. RoBERTa sendiri mengungguli BERT pada dataset GLUE, RACE, dan SQuAD.

Dalam penelitian ini menggunakan model *pretrained* yang telah tersedia yaitu Roberta-base dimana memiliki 12 *layer*, 768 *hidden layer*, 12 *head layer* dan 125 juta parameter.

DistilBERT

DistilBERT merupakan versi *distilled* BERT dari penelitian Sanh et al. (2019). Metode ini menjanjikan *pre-train* yang lebih kecil dari model representasi bahasa pada umumnya. Metode ini dapat melakukan *fine-tune* dengan kinerja yang baik di tugas-tugas yang besar. Penelitian DistilBERT telah banyak diterapkan, namun pada deteksi ulasan palsu khususnya dataset Ott et al. (2011; 2013) belum ditemui. Dalam penelitian ini menggunakan model *pretrained* DistilBERT yang telah tersedia yaitu DistilBERT-base-cased dimana memiliki 6 *layer*, 768 *hidden layer*, 12 *head layer* dan 65 juta parameter.

CNN

CNN sendiri biasa digunakan dalam permasalahan di visi komputer, seperti penelitian Puteri et. al (2020) dan Hakim et. al (2021) yang menggunakan CNN untuk mengatasi masalahnya. Namun CNN telah terbukti keefektifannya dalam permasalahan *natural language processing* (Kim, 2014). Penelitian sebelumnya dengan metode CNN pada pemrosesan bahasa alami telah banyak ditemui, dan memiliki hasil kinerja yang baik. Adapun detail dari arsitektur CNN yang bangun dalam penelitian dapat dilihat pada Tabel 4 dan Gambar 4.

Berdasarkan Gambar 4 dapat dilihat bahwa *dataset* setelah *preprocessing* dilanjutkan kepada *layer* pertama yaitu *embedding layer* dimana dalam penelitian ini menggunakan ukuran 300d. Adapun beberapa *pretrained word embedding* yang digunakan dalam penelitian ini diantaranya:

1. 300d FastText.

Pretrained ini memiliki 1 juta *word vector* yang dilatih pada Wikipedia 2017, UMBC dan statmt.org dataset berita (Mikolov et al., 2017). *Pretrained* ini dapat diakses pada laman <https://fasttext.cc/>.

2. 300d Word2Vec.

Pretrained ini merupakan kumpulan *word vector* 300d dari wikipedia2vec (Yamada et al., 2016) dimana *word vector* tersebut dilakukan dengan metode Word2vec (Mikolov et al., 2013; Le and Mikolov, 2014). *Pretrained* ini dapat diakses pada laman <https://wikipedia2vec.github.io/wikipedia2vec>.

3. 300d GloVe.

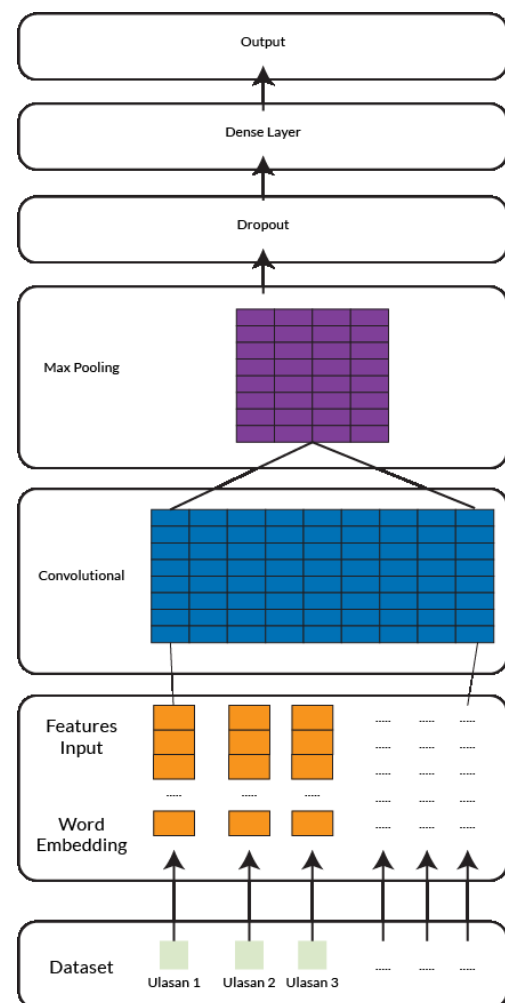
Pretrained ini merupakan kumpulan *word vector* 300d yang dilakukan dengan metode GloVe (Pennington, Socher and Manning, 2014) dimana *word vector* tersebut diambil dari Wikipedia dan

Gigaword. *Pretrained* ini dapat diakses pada laman <https://nlp.stanford.edu/projects/glove/>.

Setelah *embedding layer*, akan dilanjutkan ke *convolution layer* dengan filter 64 dan ukuran *kernel* 7x7. Kemudian dilakukan *MaxPolling*, setelah itu dilakukan *dropout* dengan nilai 0,5. Setelah dilanjutkan ke *dense layer* dengan aktivasi *sigmoid*.

Tabel 4. Model CNN

Layer	Parameter
Embedding	300000
Conv	134464
Global Max Pooling	0
Dropout	0
Dense	130



Gambar 4 Detail Arsitektur CNN

2.4. Rancangan Pengujian

Setelah data latih dan uji dibagi, selanjutnya dilakukan proses pencarian model terbaik untuk diuji pada data uji. Adapun *hyperparameter* yang digunakan yaitu sama dalam semua pengujian baik *fine-tuning* menggunakan model *Transformer* dan CNN. *Hyperparameter* tersebut dapat dilihat pada Tabel 5. Dalam pengujian ini menggunakan *check point* dari *library python* untuk mengambil model

terbaik yang dapat digunakan pada data uji. Adapun pembagian untuk data validasi di pengujian yaitu 20% dari data latih. Setelah model terbaik didapatkan maka dilakukan proses klasifikasi pada data uji.

Tabel 5. Hyperparameter

Hyperparameter	Nilai
Fungsi Loss	<i>Categorical Cross-entropy</i>
Optimisasi	Adam
<i>Learning rate</i>	2e-5
<i>Epsilon</i>	1e-08
<i>Batch size</i>	16

2.5. Metode Evaluasi

Adapun penilaian evaluasi yang digunakan pada penelitian ini yaitu akurasi, *precision*, *recall* dan *f1-measure*. Penilaian evaluasi tersebut dapat dicari dari *confusion matrix*. Adapun contoh dari *confusion matrix* dapat dilihat pada Tabel 6.

Tabel 6. Confusion Matrix

		Aktual	
		True	False
Prediksi	True	True Positive (TP)	False Positive (FP)
	False	False Negative (FN)	True Negative (TN)

Adapun persamaan dari tiap-tiap penilaian evaluasi dapat dilihat pada Persamaan (1), (2), (3) dan (4).

$$Akurasi = \frac{TP+TN}{(TP+TN+FP+FN)} \quad (1)$$

$$Precision = \frac{TP}{(FP+TP)} \quad (2)$$

$$Recall = \frac{TP}{(FN+TP)} \quad (3)$$

$$F - Score = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (4)$$

3. HASIL DAN PEMBAHASAN

Pengujian dijalankan sebanyak 5 kali dengan tahapan yang dijelaskan pada Gambar 1. Hasil dari pengujian dalam penelitian ini dapat dilihat pada Tabel 7.

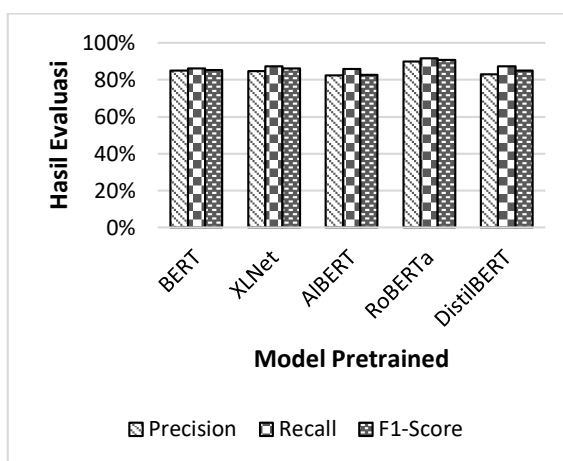
Berdasarkan Tabel 7 dapat dilihat hasil dari model *Transformer* dan CNN. Adapun urutan hasil akurasi model *Transformer* secara *ascending* yaitu ALBERT, DistilBERT, BERT, XLNet, dan RoBERTa dengan nilai akurasi 83%; 84,4%; 85%; 85,6% dan 90,8%. Dari hasil ini terlihat bahwa keseluruhan model *Transformer* dapat bekerja dengan baik dalam deteksi ulasan palsu menggunakan dataset Ott et al. (2011; 2013).

Tabel 7. Hasil dengan Model Transformer dan Deep Learning

Model		Akurasi	Precision	Recall	F1-Score
BERT	Asli	85%	85,4%	84,2%	84,4%
	Palsu		85%	86,2%	85,2%
XLNet	Asli	85,6%	87%	83,8%	85,2%
	Palsu		84,8%	87,4%	86,2%

Model		Akurasi	Precision	Recall	F1-Score
ALBERT	Asli	83%	84,8%	80,8%	83,4%
	Palsu		82,4%	85,8%	82,8%
RoBERTa	Asli	90,8%	91,6%	89,2%	90,2%
	Palsu		90%	91,8%	90,8%
DistilBERT	Asli	84,4%	86,2%	81,6%	83,6%
	Palsu		83%	87,4%	85%
CNN	300d-Asli	74,2%	74%	73,2%	73,4%
	FastText		74,8%	75,2%	74,8%
	300d-Word2vec	78,2%	78,2%	77,8%	78%
	Palsu		78,4%	78,4%	78,4%
300d-GloVe	Asli	79,8%	78,6%	82%	80,2%
	Palsu		81,6%	78%	79,6%

Grafik dari dari *precision*, *recall* dan *f1-score* model *Transformer* dapat dilihat pada Gambar 2.

Gambar 5 Evaluasi ulasan palsu pada model *Transformer*

Berdasarkan Gambar 2 terlihat bahwa hasil dengan model *pretrained* RoBERTa mengungguli hasil model *Transformer* yang lain dari segi akurasi, *precision*, *recall* dan *f1-score*. Sedangkan model ALBERT merupakan model yang memiliki hasil terendah dibandingkan model *Transformer* yang lain. Meskipun terendah dalam model *Transformer* yang lain, namun masih mengungguli hasil dari model CNN.

Pretrained dari *word embedding GloVe* mendapatkan hasil akurasi tertinggi pada model CNN yaitu dengan akurasi 79,8%. Sedangkan hasil terendahnya dengan *pretrained* dari *word embedding* FastText yaitu dengan akurasi 74,2%. Perbedaan dari hasil dengan *pretrained word embedding* ini dapat disebabkan dari ukuran token, sumber data *word embedding* dan bagaimana *pretrained* itu dilatih (Maslej-Krešňáková et al., 2020).

Penelitian ini juga melakukan pengujian terhadap waktu komputasi yang dibutuhkan pada pelatihan model *Transformer*. Adapun hasil rata-rata waktu komputasi pelatihan model *Transformer* yang dilakukan sebanyak dua kali pengujian dapat dilihat pada Tabel 8.

Tabel 8. Hasil Waktu Komputasi

Model	Waktu Komputasi (detik)
BERT	7139,5
XLNet	348
ALBERT	473,5
RoBERTa	263,5
DistilBERT	200,5

Berdasarkan Tabel 8 dapat dilihat bahwa DistilBERT menghasilkan waktu komputasi tercepat dibandingkan model *Transformer* yang lain yaitu 200,5 detik. Sedangkan model dengan waktu komputasi terlama adalah BERT yaitu 7139,5 detik. Hasil waktu komputasi dari BERT juga tergolong sangat lama apabila dibandingkan dengan model *Transformer* lainnya dengan perbedaan $\pm 6818,125$ detik.

Secara keseluruhan model *Transformer* mendapatkan hasil yang lebih baik dibandingkan dengan CNN. Meski begitu, secara keseluruhan kedua model *Transformer* dan CNN memperoleh hasil yang bagus pada dataset Ott et al. (2011; 2013).

4. KESIMPULAN

Penelitian ini melakukan perbandingan model *transformer* dan *deep learning* pada deteksi ulasan palsu dengan dataset Ott et al. (2011; 2013). *Pretrained* dengan Model RoBERTa memberikan hasil yang tertinggi dibandingkan dengan model *Transformer* lainnya dimana nilai akurasi 90,8%; *precision* 90%; *recall* 91,8% dan *f1-score* 90,8%. Namun, dari segi waktu komputasi pelatihan model, model DistilBERT adalah model tercepat dalam waktu komputasi yaitu dengan waktu rata-rata 200,5%.

Hasil dari penelitian ini juga memperlihatkan bahwa model *Transformer* lebih unggul dibandingkan dengan *deep learning* untuk menyelesaikan masalah deteksi ulasan palsu pada dataset Ott. Meskipun begitu, hasil dari *deep learning* masih cukup baik.

Saran untuk penelitian selanjutnya yaitu menggunakan dataset ulasan palsu yang diambil dari *website* sesungguhnya seperti dataset dari penelitian Rayana et al. (2015).

DAFTAR PUSTAKA

- BAHY HAKIM, H., UTAMININGRUM, F., & SETIA BUDI, A., 2021. Early Detection of COVID-19 Patient's Survivability Based On The Image Of Lung X-Ray Image Using Deep Neural Networks. *Kinetik: Game Technology, Information System, Computer Network, Computing, Electronics, and Control*, 4(3). <https://doi.org/10.22219/kinetik.v6i3.1265>.
- DEVLIN, J., CHANG, M.W., LEE, K., & TOUTANOVA, K., 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. *CoRR*, [online] 1, pp.4171–4186. Available at: <<http://arxiv.org/abs/1810.04805>>.
- HAJEK, P., BARUSHKA, A., & MUNK, M., 2020. Fake consumer review detection using deep neural networks integrating word embeddings and emotion mining. *Neural Computing and Applications*, [online] 32(23), pp.17259–17274. <https://doi.org/10.1007/s00521-020-04757-2>.
- HE, S., HOLLENBECK, B., & PROSERPIO, D., 2021. The Market for Fake Reviews. *SSRN Electronic Journal*.
- HUGGINGFACE, 2021. *Transformer models - Hugging Face Course*. [online] Available at: <<https://huggingface.co/course/chapter1/4?fw=pt>> [Accessed 4 Aug. 2021].
- KENNEDY, S., WALSH, N., SLOKA, K., MCCARREN, A., & FOSTER, J., 2019. Fact or factitious? Contextualized opinion spam detection. In: *ACL 2019 - 57th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Student Research Workshop*. pp.344–350.
- KIM, M., LEE, S.M., CHOI, S., & KIM, S.Y., 2021. Impact of visual information on online consumer review behavior: Evidence from a hotel booking website. *Journal of Retailing and Consumer Services*, [online] 60, p.102494. <https://doi.org/10.1016/j.jretconser.2021.102494>.
- KIM, Y., 2014. Convolutional neural networks for sentence classification. *EMNLP 2014 - 2014 Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference*, pp.1746–1751. <https://doi.org/10.3115/v1/d14-1181>.
- LAN, Z., CHEN, M., GOODMAN, S., GIMPEL, K., SHARMA, P., & SORICUT, R., 2019. ALBERT: A Lite BERT for Self-supervised Learning of Language Representations. [online] pp.1–17. Available at: <<http://arxiv.org/abs/1909.11942>>.
- LE, Q., & MIKOLOV, T., 2014. Distributed representations of sentences and documents. *31st International Conference on Machine Learning, ICML 2014*, 4, pp.2931–2939.
- MASLEJ-KREŠŇÁKOVÁ, V., SARNOVSKÝ, M., BUTKA, P., & MACHOVÁ, K., 2020. Comparison of deep learning models and various text pre-processing techniques for the toxic comments classification. *Applied Sciences (Switzerland)*, 10(23), pp.1–26. <https://doi.org/10.3390/app10238631>.
- MIKOLOV, T., GRAVE, E., BOJANOWSKI, P., PUHRSCHE, C., & JOULIN, A., 2017. Advances in pre-training distributed word

representations.

- MIKOLOV, T., SUTSKEVER, I., CHEN, K., CORRADO, G., & DEAN, J., 2013. Distributed Representations of Words and Phrases and their Compositionality. In: *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2*. [online] Lake Tahoe, Nevada: Curran Associates Inc.pp.3111–3119. Available at: <<http://arxiv.org/abs/1310.4546>>.
- MUKHERJEE, A., VENKATARAMAN, V., LIU, B., & GLANCE, N., 2011. What yelp fake review filter might be doing? *Proceedings of the Seventh International AAAI Conference on Weblogs and Social Media*, [online] pp.409–418. Available at: <<https://www.aaai.org/ocs/index.php/ICWSM/ICWSM13/paper/viewPaper/6006>>.
- MURPHY, R., 2020. *Local Consumer Review Survey: How Customer Reviews Affect Behavior*. [online] Available at: <<https://www.brightlocal.com/research/local-consumer-review-survey/>> [Accessed 4 Aug. 2021].
- OTT, M., CARDIE, C., & HANCOCK, J.T., 2013. Negative deceptive opinion spam. In: *NAACL HLT 2013 - 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Proceedings of the Main Conference*. Atlanta, Georgia.pp.497–501.
- OTT, M., CHOI, Y., CARDIE, C., & HANCOCK, J.T., 2011. Finding deceptive opinion spam by any stretch of the imagination. In: *ACL-HLT 2011 - Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*. Portland, Oregon.pp.309–319.
- PENNINGTON, J., SOCHER, R., & MANNING, C., 2014. GloVe: Global Vectors for Word Representation. In: *EMNLP*. [online] Doha, Qatar: Association for Computational Linguistics.pp.1532–1543. <https://doi.org/10.3115/v1/D14-1162>.
- PUTERI, R.T., & UTAMININGRUM, F., 2020. Micro-sleep detection using combination of haar cascade and convolutional neural network. *ACM International Conference Proceeding Series*, pp.130–135. <https://doi.org/10.1145/3427423.3427433>.
- RAYANA, S., & AKOGLU, L., 2015. *Collective opinion spam detection: Bridging review networks and metadata*. *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. New York, NY, USA: Association for Computing Machinery. <https://doi.org/10.1145/2783258.2783370>.
- SANH, V., DEBUT, L., CHAUMOND, J., & WOLF, T., 2019. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. *CoRR*. [online] Available at: <<http://arxiv.org/abs/1910.01108>>.
- YAMADA, I., SHINDO, H., TAKEDA, H., & TAKEFUJI, Y., 2016. Joint learning of the embedding of words and entities for named entity disambiguation. *CoRR*, pp.250–259. <https://doi.org/10.18653/v1/k16-1025>.
- YANG, Z., DAI, Z., YANG, Y., CARBONELL, J., SALAKHUTDINOV, R., & LE, Q. V., 2019. XLNet: Generalized autoregressive pretraining for language understanding. [online] Available at: <<http://arxiv.org/abs/1906.08237>>.
- ZHANG, W., DU, Y., YOSHIDA, T., & WANG, Q., 2018. DRI-RCNN: An approach to deceptive review identification using recurrent convolutional neural network. *Information Processing and Management*, 54, pp.576–592. <https://doi.org/10.1016/j.ipm.2018.03.007>.