



Prediksi Indeks BEI dengan *Ensemble Convolutional Neural Network* dan *Long Short-Term Memory*

Harya Widiputra¹, Adele Mailangkay², Elliana Gautama³

¹Fakultas Teknologi Informasi, Perbanas Institute

¹harya@perbanas.id, ²adele@perbanas.id, ³elliana@perbanas.id

Abstract

The Indonesian Stock Exchange (IDX) stock market index is one of the main indicators commonly used as a reference for national economic conditions. The value of the stock market index is often being used by investment companies and individual investors to help making investment decisions. Therefore, the ability to predict the stock market index value is a critical need. In the fields of statistics and probability theory as well as machine learning, various methods have been developed to predict the value of the stock market index with a good accuracy. However, previous research results have found that no one method is superior to other methods. This study proposes an ensemble model based on deep learning architecture, namely Convolutional Neural Network (CNN) and Long Short-Term Memory (LSTM), called the CNN-LSTM. To be able to predict financial time series data, CNN-LSTM takes feature from CNN for extraction of important features from time series data, which are then integrated with LSTM feature that is reliable in processing time series data. Results of experiments on the proposed CNN-LSTM model confirm that the hybrid model effectively provides better predictive accuracy than the stand-alone time series data forecasting models, such as CNN and LSTM.

Keywords: BEI indexes, ensemble, Convolutional Neural Network, Long Short-Term Memory, deep learning, time series data.

Abstrak

Indeks pasar saham Bursa Efek Indonesia (BEI) telah menjadi salah satu indikator utama yang umum digunakan sebagai referensi kondisi perekonomian nasional. Selain itu, nilai indeks pasar saham juga seringkali digunakan oleh perusahaan investasi maupun investor individual sebagai salah satu acuan dalam mengambil keputusan investasi. Oleh sebab itu, kemampuan untuk dapat memprediksi pergerakan nilai indeks pasar saham menjadi sebuah kebutuhan yang kritis. Dalam bidang ilmu statistika dan teori probabilitas maupun *machine learning* telah banyak dikembangkan berbagai metode untuk meramalkan nilai indeks pasar saham dengan tingkat akurasi yang baik. Namun demikian, berbagai hasil penelitian sebelumnya telah menemukan bahwa tidak ada satu metode yang paling unggul dibandingkan metode lainnya. Oleh sebab itu, penelitian ini mengusulkan sebuah model *ensemble* yang berbasis pada arsitektur *deep learning* yaitu *Convolutional Neural Network* (CNN) dan *Long Short-Term Memory* (LSTM), yang disebut *CNN-LSTM*. Untuk dapat melakukan prediksi data deret waktu keuangan, *CNN-LSTM* mengambil fitur unggulan dari *CNN* untuk ekstraksi fitur penting dari data deret waktu, yang kemudian diintegrasikan dengan fitur *LSTM* yang handal dalam memproses data deret waktu. Hasil uji coba terhadap model *CNN-LSTM* yang dibangun mengkonfirmasi bahwa model *hybrid* secara efektif memberikan akurasi prediksi yang lebih baik dibandingkan dengan model peramalan data deret waktu yang berdiri sendiri, seperti halnya *CNN* dan *LSTM*.

Kata kunci: indeks BEI, *ensemble*, *Convolutional Neural Network*, *Long Short-Term Memory*, *deep learning*, data deret waktu.

1. Pendahuluan

Pergerakan indeks pasar saham selalu identik sebagai salah satu indikator penting dalam menilai kondisi perekonomian sebuah negara, sehingga menjadi sebuah masalah yang sangat penting di bidang ekonomi [1]. Secara umum, pergerakan indeks pasar saham dipengaruhi oleh berbagai faktor internal dan eksternal, seperti lingkungan ekonomi dalam dan luar negeri, situasi internasional, prospek industri dan operasi pasar

saham, namun utamanya sangat terpengaruh oleh nilai historis dari indeks pasar saham itu sendiri. Berdasarkan karakteristik tersebut, maka berbagai metode peramalan pergerakan indeks pasar saham juga memiliki titik fokus yang berbeda dibandingkan dengan metode peramalan data deret waktu lainnya [2,3].

Metode analisis tradisional untuk meramal arah pergerakan indeks pasar saham umumnya berbasis pada analisa ekonomi dan keuangan yang utamanya

menggunakan teknik analisis fundamental dan juga analisis teknis. Dalam hal ini, teknik analisis fundamental lebih memperhatikan nilai intrinsik saham-saham yang diperdagangkan di bursa efek dan secara kualitatif menganalisis faktor-faktor eksternal yang mempengaruhinya, seperti halnya suku bunga, nilai tukar, inflasi, kebijakan industri, dan faktor ekonomi serta politik lainnya. Di sisi lain, teknik analisis teknikal fokus pada arah harga saham-saham yang diperdagangkan di bursa efek, volume perdagangan, dan ekspektasi psikologis investor. Teknik analisis teknikal juga fokus pada analisis lintasan historis dari indeks pasar guna menemukan pola-pola tertentu yang muncul secara berulang. Hingga saat ini, teknik analisis fundamental dan teknikal masih menjadi metode yang paling umum digunakan oleh banyak perusahaan investasi maupun dan investor individual dalam memperkirakan arah gerak indeks pasar saham untuk kemudian mengambil keputusan investasi [4,5].

Namun demikian, tingkat akurasi dari teknik analisis fundamental sulit untuk diyakini. Alasannya, faktor-faktor yang mempengaruhi tidak hanya dalam siklus jangka panjang, tetapi juga hasil prediksi yang lebih bergantung pada kualitas profesional analis atau bersifat subjektif. Indeks pasar saham masuk dalam kategori data deret waktu finansial, yang dalam hal ini memiliki karakteristik khusus, yaitu sifat *random walk* [6].

Dalam bidang ilmu statistika dan teori probabilitas, beberapa ahli menggunakan model peramalan deret waktu linier untuk memprediksi indeks pasar saham jangka pendek dengan sejumlah besar data jangka panjang, seperti *vector autoregression (VAR)* [7], *Bayesian vector autoregression model (BVAR)* [8], *autoregressive integrated moving average mode (ARIMA)* [9], dan *generalized autoregressive conditional heteroskedasticity model (GARCH)* [10], guna menangani data deret waktu dengan karakteristik *random walk*. Tetapi, tingkat akurasi peramalan yang dihasilkan oleh berbagai model deret waktu linier tersebut sering kali dipertanyakan. Hal ini karena tingginya tingkat ketidakpastian dan karakteristik derau yang tinggi dari data deret waktu keuangan serta terus terjadinya perubahan yang dinamis pada hubungan antara variabel independen dan variabel dependennya [11].

Selain beberapa model peramalan data deret waktu dari bidang ilmu statistika dan teori probabilitas di atas, dalam bidang ilmu machine learning khususnya jaringan saraf tiruan juga telah banyak dikembangkan banyak model pembelajaran *deep learning* untuk pemodelan dan prediksi data deret waktu [12]. Diantaranya adalah *Convolutional Neural Network* atau *CNN* yang merupakan salah satu model pembelajaran *deep learning* yang telah digunakan pada beberapa penelitian sebelumnya untuk melakukan prediksi data deret waktu [12], dimana ditemukan bahwa meskipun pada umumnya *CNN* banyak digunakan untuk proses

klasifikasi namun ternyata juga dapat memberikan hasil prediksi deret waktu yang baik. Selain itu, penelitian yang dipublikasikan oleh [6] juga mengkonfirmasi kinerja model pembelajaran *deep learning* lainnya yang lebih umum digunakan untuk prediksi data deret waktu, yaitu *Long Short-Term Memory* atau *LSTM* dalam peramalan harga saham.

Namun demikian, memprediksi arah pergerakan nilai deret waktu dengan hanya menggunakan sebuah model tunggal, baik yang berupa model peramalan deret waktu linier ataupun jaringan saraf tiruan, memiliki keterbatasan yang signifikan. Sehingga, menggabungkan berbagai algoritma terbaik untuk dapat memanfaatkan berbagai kelebihan yang dimiliki masing-masing algoritma dengan membangun sebuah model *hybrid* telah menjadi arah perkembangan dalam proses analisa dan pemodelan data deret waktu keuangan sekarang ini [12].

Sejalan dengan hal itu, model peramalan data deret waktu dalam penelitian ini dirancang dengan memanfaatkan karakteristik terbaik dari kedua model pembelajaran *deep learning* tersebut. Model *hybrid* atau *ensemble* yang dibangun dalam penelitian ini menggabungkan keunggulan *CNN*, yang dapat mengekstraksi berbagai fitur-fitur penting dari data, dengan kekuatan *LSTM* yang tidak hanya dapat menemukan saling ketergantungan antar data dalam data deret waktu, tetapi juga secara otomatis mampu mendeteksi mode terbaik serta relevan untuk data deret waktu yang diproses sehingga secara efektif dapat meningkatkan akurasi peramalan nilai indeks pasar saham. Pada arsitektur *CNN-LSTM* yang diusulkan, *layer CNN* digunakan untuk ekstraksi fitur dari data waktu masukan dan *layer LSTM* dimanfaatkan untuk memprediksi nilai penutupan dari indeks pasar saham pada hari berikutnya.

Untuk memverifikasi keefektifan model yang dirancang, penelitian ini menggunakan data pergerakan indeks pasar saham harian di Bursa Efek Indonesia, sejumlah 242 hari perdagangan dari 1 Januari 2020 hingga 31 Desember 2020.

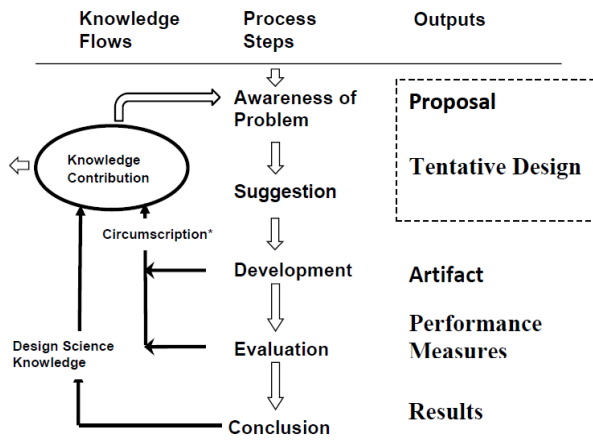
2. Metode Penelitian

Penelitian yang dilaksanakan adalah penelitian perancangan atau yang dikenal sebagai *Design Science Research* [13]. Secara umum penelitian *Design Science Research* bertujuan untuk memecahkan masalah yang bersifat nyata dengan pendekatan membangun sebuah solusi baik dalam bentuk model, algoritma, perangkat lunak, proto-tipe ataupun kerangka kerja.

Guna merealisasikan tujuan penelitian dan juga menjawab rumusan masalah yang telah ditetapkan, maka tahapan-tahapan yang akan dilakukan dalam penelitian ini dijabarkan pada Gambar 1. Adapun tahapan penelitian sebagaimana diperlihatkan pada Gambar 1 dijelaskan lebih lanjut sebagai berikut.

Awareness of Problem, melakukan kajian model *ensemble CNN-LSTM* untuk diterapkan guna membantu proses pemodelan dan prediksi pergerakan data deret waktu.

Suggestion, merancang struktur *ensemble CNN-LSTM* guna memodelkan dan melakukan prediksi pergerakan data deret waktu.



Gambar 1. Kerangka metode *Design Science Research* yang menjadi metodologi utama dalam pelaksanaan penelitian [13]

Development, implementasi struktur *ensemble CNN-LSTM* dan pengumpulan data nyata untuk uji coba kinerja dalam pemodelan dan prediksi pergerakan data deret waktu.

Evaluation, uji coba dan evaluasi struktur *ensemble CNN-LSTM* yang dikembangkan menggunakan data nyata yang telah dikumpulkan. Pada tahap ini apabila ditemukan masalah, maka akan kembali pada tahap perancangan serta implementasi untuk memperbaiki kinerjanya. Dilanjutkan dengan evaluasi serta analisa komparasi terhadap hasil uji coba dari *CNN-LSTM* dibandingkan dengan metode prediksi pergerakan data deret waktu lainnya.

Conclusion, penarikan kesimpulan yang akan memberikan kontribusi pada perkembangan *domain* ilmu, dalam hal ini analisa data deret waktu dan jaringan saraf tiruan.

2.1. Pengembangan Model *CNN-LSTM*

CNN dan *LSTM* dapat dimanfaatkan untuk membangun model *deep learning* yang memiliki kemampuan mempelajari secara mendalam pola-pola yang bersifat kompleks dan tersembunyi dalam tumpukan data yang beragam dan berukuran masif, termasuk didalamnya adalah data deret waktu khususnya dari bidang keuangan.

Memberdayakan keunggulan yang dimiliki oleh kedua model tersebut, untuk mencapai tujuan dari penelitian ini sebagaimana telah disampaikan pada bagian Pendahuluan, yaitu guna meningkatkan akurasi peramalan pergerakan indeks pasar saham, maka dalam penelitian ini dibangun sebuah model peramalan data

deret waktu dengan mengintegrasikan *CNN* dan *LSTM*, atau membangun model *ensemble* yang disebut sebagai *CNN-LSTM*. Dengan konsep *ensemble* ini diharapkan solusi yang diberikan akan menjadi lebih baik serta komprehensif karena tiap-tiap model memiliki peran dan kemampuan yang berbeda untuk dapat menangkap pola yang tersembunyi dalam data.

CNN adalah model jaringan yang diusulkan oleh Lecun dkk. pada tahun 1998 [14]. *CNN* adalah salah satu model *feedforward neural network*, yang memiliki kinerja yang baik dalam pemrosesan gambar dan juga *natural language processing* [15]. Meskipun pada umumnya *CNN* lebih sering diaplikasi pada data citra, namun model ini juga telah terbukti dapat diterapkan secara efektif pada peramalan data deret waktu [16].

Karakteristik utama dari *CNN* adalah kemampuan untuk dapat mengenali dan mengekstrak berbagai fitur yang tampak jelas dari garis pandang, sehingga model ini sangat umum digunakan dalam proses rekayasa fitur khususnya ekstraksi fitur dari data.

Salah satu keunggulan utama *CNN* adalah fitur *local perception* dan *weight sharing* yang dapat sangat mengurangi jumlah parameter, sehingga meningkatkan efisiensi pada proses pembelajaran. Dari sisi struktur, *CNN* utamanya terdiri dari dua bagian, yaitu *layer* konvolusi (*convolutional*) dan *layer* penyatuan (*pooling*). Dalam hal ini, setiap *layer* konvolusi berisi sejumlah *kernel* konvolusi, yang rumus penghitungannya ditunjukkan pada Rumus 1.

$$l_t = \tanh(x_t * k_t + b_t), \quad (1)$$

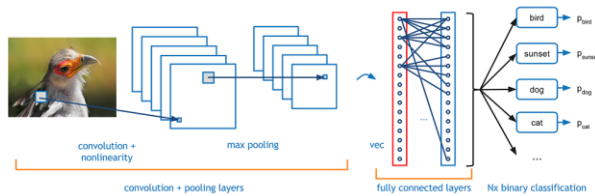
di mana l_t mewakili nilai keluaran setelah proses konvolusi, \tanh adalah fungsi aktivasi yang umum digunakan, x_t adalah vektor masukan, k_t adalah nilai bobot dari *kernel* konvolusi, b_t dan merupakan *bias* dari *kernel* konvolusi. Jika tidak menggunakan *bias*, berarti b_t adalah 0.

Setelah operasi konvolusi yang terjadi pada *layer* konvolusi, fitur penting dari data diekstraksi yang disertai dengan peningkatan dimensi fitur. Untuk mengatasi masalah ini dan mengurangi beban pada proses pelatihan, maka lapisan penyatuan ditambahkan dengan tujuan utama melakukan reduksi kembali terhadap jumlah fitur yang diekstraksi.

Jaringan konvolusional *CNN* diilhami oleh proses biologis [17,18,19] karena pola konektivitas antara *neuron* menyerupai organisasi korteks visual hewan. *CNN* menggunakan *pre-processing* yang relatif lebih sedikit dibandingkan dengan algoritma klasifikasi gambar lainnya. Cara kerja *CNN* diilustrasikan pada Gambar 2.

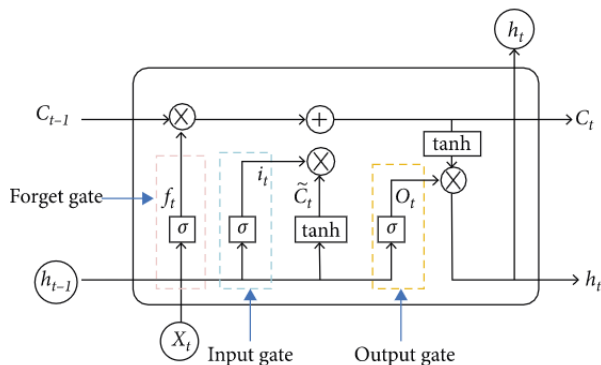
Di sisi lain, *LSTM* adalah sub-unit dari *recurrent neural network (RNN)*. *LSTM* sangat cocok untuk mengklasifikasi, memproses, dan membuat prediksi berdasarkan data deret waktu tunggal. Pada 2016,

perusahaan teknologi besar termasuk Google, Apple, dan Microsoft menggunakan LSTM sebagai komponen fundamental dalam produk barunya. Sebagai contoh, Google menggunakan LSTM untuk pengenalan suara pada *smartphone* [20], untuk asisten pintar Allo [21] dan untuk Google Translate [22,23]. Sedangkan Apple menggunakan LSTM untuk fungsi Quicktype pada iPhone [24,25] dan untuk Siri.



Gambar 2. Ilustrasi struktur CNN yang umum digunakan dalam proses pengenalan citra [17]

Selain itu, dalam beberapa tahun terakhir ini LSTM juga telah diadopsi di bidang peramalan pergerakan nilai di pasar saham [26,27,28], dimana struktur internal LSTM yang dibangun cukup sederhana dan hanya menggunakan satu modul *backpropagation* seperti halnya dalam struktur RNN biasa. Adapun, sel memori LSTM terdiri dari tiga bagian, yaitu: *forget gate*, *input gate*, dan *output gate* sebagaimana yang ditunjukkan pada Gambar 3.



Gambar 3. Struktur standar model LSTM yang terdiri dari komponen *forget gate*, *input gate* dan *output gate* dengan fitur *backpropagation* [29]

Proses komputasi yang terjadi dalam struktur LSTM guna melakukan penghitungan prediksi data deret waktu diawali dengan penghitungan nilai keluaran dari waktu sebelumnya dan nilai masukan dari waktu saat ini menjadi masukan ke *forget gate*, dan hasil pemrosesan dari *forget gate* diperoleh melalui komputasi menggunakan Rumus 2.

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f), \quad (2)$$

dimana rentang nilai dari f_t adalah (0,1), W_f adalah bobot dari *forget gate*, dan b_f adalah nilai *bias* yang diterapkan pada *forget gate*, x_t adalah nilai masukan untuk waktu saat ini, dan h_{t-1} adalah nilai keluaran dari pemrosesan waktu sebelumnya. Selanjutnya, nilai keluaran dari waktu sebelumnya dan nilai masukan dari waktu saat ini juga menjadi masukan ke *input gate*, dan nilai keluaran

serta kondisi sel kandidat pada *input gate* diperoleh setelah kalkulasi menggunakan Rumus 3 dan 4.

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i), \quad (3)$$

$$\tilde{C}_t = \tanh(W_c \cdot [h_{t-1}, x_t] + b_c), \quad (4)$$

dimana rentang nilai dari i_t adalah (0,1), W_i adalah bobot dari *input gate*, b_i adalah nilai *bias* dari *input gate*, W_c adalah bobot dari kandidat *input gate*, dan b_c adalah nilai *bias* dari kandidat *input gate*.

Tahap berikutnya pada model LSTM adalah proses penyesuaian nilai sel atau parameter model pada saat ini dilakukan berdasarkan Rumus 5.

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t, \quad (5)$$

dimana rentang nilai dari C_t adalah (0,1). Kemudian, pada pemrosesan waktu t , nilai keluaran h_{t-1} dan nilai masukan x_t menjadi masukan untuk *output gate*, dan keluaran dari *output gate* dihitung menggunakan Rumus 6.

$$o_t = \sigma(W_o [h_{t-1}, x_t] + b_o), \quad (6)$$

dimana, rentang nilai dari o_t adalah (0,1), W_o adalah bobot dari *output gate*, dan b_o adalah nilai *bias* dari *output gate*.

Pada akhirnya, nilai keluaran akhir dari LSTM dihasilkan oleh *output gate* dan merupakan hasil kalkulasi menggunakan Rumus 7.

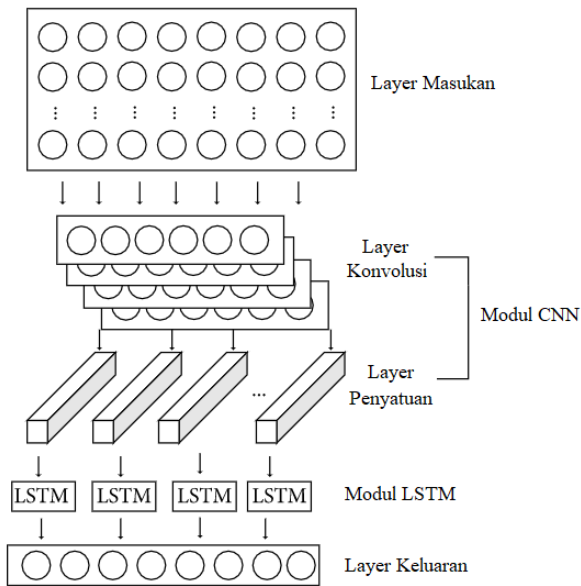
$$h_t = o_t * \tanh(C_t). \quad (7)$$

dalam hal ini tanh adalah fungsi aktivasi yang dapat disesuaikan dengan kebutuhan maupun karakteristik dari permasalahan yang akan diselesaikan.

LSTM memiliki kemampuan untuk memproses data yang direkam secara urutan waktu tertentu, dan oleh karenanya telah digunakan secara luas dalam proses analisa dan pemodelan data deret waktu.

Struktur dari model *ensemble CNN* dan LSTM yang dibangun pada penelitian ini mengacu pada struktur yang digunakan oleh Lu, dkk. pada penelitian mereka yang memprediksi pergerakan harga saham di bursa efek Shanghai [29]. Namun demikian, struktur data masukan, konfigurasi parameter pada tiap *layer* dan juga parameter proses pelatihan dalam penelitian ini disesuaikan dengan karakteristik data indeks pasar saham BEI.

Diagram struktur dari model CNN-LSTM diperlihatkan pada Gambar 4, dimana CNN dan LSTM menjadi komponen utama disertai dengan *layer* masukan (input), *layer* konvolusi 1 dimensi (1D *convolutional*), *layer* penyatuan (*pooling*), *layer* tersembunyi LSTM (*hidden*), dan *layer* koneksi penuh (*full connection*) yang akan mengeluarkan hasil akhir prediksi.



Gambar 4. Diagram struktur model *CNN-LSTM* untuk prediksi nilai data deret waktu [29]

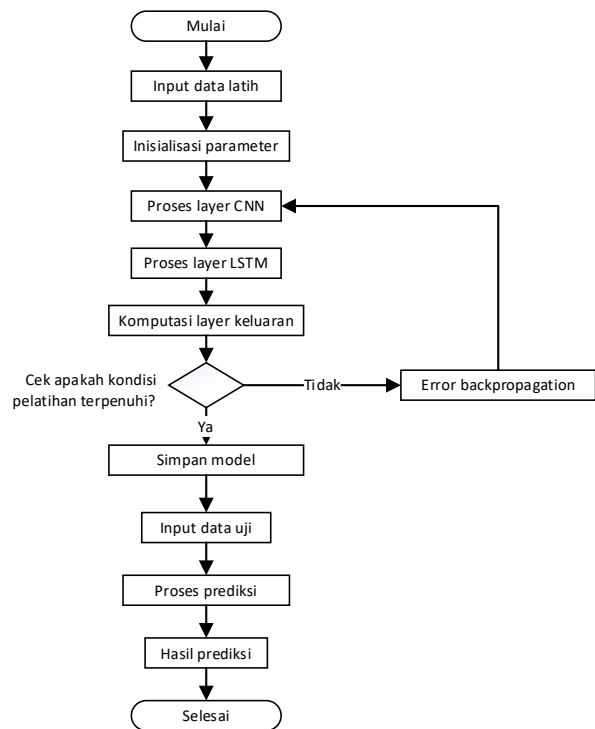
2.2. Proses Pelatihan dan Prediksi *CNN-LSTM*

Tahapan proses pelatihan dan prediksi *CNN-LSTM* pada data deret waktu ditampilkan pada Gambar 5. Secara rinci tahapan pelatihan dimulai dengan proses input data latih (*training*), pada tahap ini terjadi proses memasukan data yang digunakan untuk pelatihan *CNN-LSTM*. Tahap selanjutnya adalah inialisasi parameter jaringan guna menetapkan bobot dan nilai *bias* (jika ada) pada awal setiap *layer CNN-LSTM*. Kemudian, dilanjutkan dengan proses pada *layer CNN* dimana data masukan secara berurutan melewati *layer konvolusi* dan *layer penyatuan* (*pooling*) pada *layer CNN*, dilanjutkan dengan proses ekstraksi fitur data masukan, serta menghasilkan nilai keluaran yang akan menjadi masukan untuk *layer LSTM*. Pada *layer CNN* ini utamanya terjadi proses ekstraksi fitur dari data deret waktu yang menjadi masukan bagi *layer LSTM*.

Selanjutnya nilai keluaran dari *layer CNN* akan memasuki *layer LSTM*. Pada *layer LSTM* ini utamanya terjadi proses prediksi nilai deret waktu yang diamati, dimana nilai keluaran dari *layer LSTM* menjadi masukan untuk *layer koneksi penuh* (*full connection*) yang kemudian menghasilkan nilai akhir prediksi. Pada tahap ini proses pelatihan prediksi selesai dan kemudian dilanjutkan dengan proses evaluasi hasil pelatihan dimana dilakukan penghitungan kesalahan hasil prediksi. Nilai keluaran berupa hasil prediksi yang dihitung oleh *layer keluaran* dibandingkan dengan nilai sebenarnya dari kelompok data yang diproses, untuk kemudian dihitung nilai kesalahannya.

Hasil evaluasi menjadi acuan untuk menentukan apakah kondisi berhenti untuk pelatihan terpenuhi. Dalam hal ini, kondisi berhenti pelatihan adalah tercapainya jumlah siklus pelatihan yang telah ditentukan (*epochs*), dan nilai

kesalahan prediksi lebih rendah dari ambang tertentu yang telah ditentukan.



Gambar 5. Tahapan proses pelatihan dan pengujian pada model *CNN-LSTM* yang diusulkan untuk prediksi nilai indeks pasar saham BEI

Jika berdasarkan hasil evaluasi ditetapkan kondisi berhenti untuk pelatihan belum dicapai, maka nilai kesalahan yang dihitung dipropagasi kembali ke *layer* sebelumnya dan kemudian dilakukan penyesuaian bobot serta nilai *bias* pada tiap *layer* (*error backpropagation*), dan proses pelatihan Kembali dilakukan hingga kondisi berhenti tercapai. Namun jika salah satu syarat dari kondisi berhenti untuk pelatihan terpenuhi, maka pelatihan selesai dan konfigurasi seluruh jaringan *CNN-LSTM* disimpan.

Tahap selanjutnya adalah pengujian model *CNN-LSTM* yang telah selesai dilatih menggunakan data uji. Proses ini dimulai dengan memasukan data uji yang digunakan untuk prediksi atau input data uji (*testing*) ke model *CNN-LSTM* yang telah disimpan untuk kemudian mendapatkan nilai keluaran (hasil prediksi) sebagai keluaran akhir dari proses pelatihan dan prediksi *CNN-LSTM*.

Pengukuran tingkat akurasi akan dilakukan dengan menerapkan penghitungan nilai *Root Mean Square Error (RMSE)*, guna melihat besaran simpangan antara nilai sebenarnya terhadap nilai prediksi yang dihasilkan.

2.3. Implementasi Model *CNN-LSTM*

Sebagaimana telah dijelaskan sebelumnya struktur model *CNN-LSTM* yang dibangun terdiri dari 2 (dua) *layer* komponen utama, yaitu *layer CNN* dan *layer LSTM*. Adapun, *layer CNN* terdiri dari: *layer konvolusi*

1 dimensi, *layer* penyatuan (*pooling*), dan *layer* perataan (*flatten*). Sedangkan *layer LSTM* terdiri dari: *layer* tersembunyi *LSTM (hidden)*, dan *layer* pemadatan (*dense*). Pada penelitian ini konfigurasi dari parameter *CNN-LSTM* menggunakan Keras di Python diberikan pada Tabel 1, dimana jumlah *kernel* diset 1 karena prediksi bersifat *univariate* atau hanya melibatkan 1 variabel masukan, dan struktur rinci dari model *CNN-LSTM* diilustrasikan pada Gambar 6.

Tabel 1. Konfigurasi parameter pada tiap *layer* struktur *CNN-LSTM* yang diusulkan untuk prediksi nilai indeks pasar saham BEI. Konfigurasi parameter mengacu pada konfigurasi yang digunakan oleh [29] dan [30] pada penelitiannya mengenai pemanfaatan *CNN-LSTM* untuk prediksi data deret waktu

Parameter	Nilai	Layer
Filter layer konvolusi	64	CNN
Jumlah kernel pada filter	1	CNN
Fungsi aktivasi layer konvolusi	Relu	CNN
Padding layer penyatuan	Valid	CNN
Pool_size layer penyatuan	2	CNN
Jumlah hidden unit pada layer LSTM	50	LSTM
Fungsi aktivasi layer LSTM	Relu	LSTM
Time step	4	LSTM
Optimizer	Adam	LSTM
Loss function	MAE	LSTM
Epochs	500	Model Fit

2.4. Data, Alat Penelitian dan Evaluasi

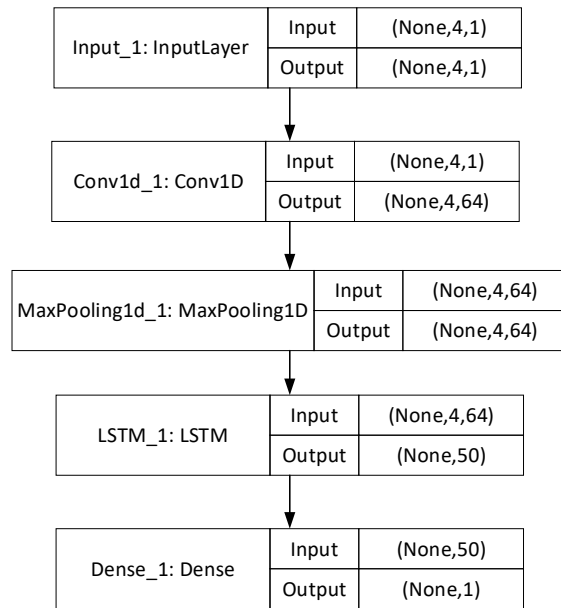
Data lapangan yang digunakan dalam hal ini adalah data harian pergerakan indeks pasar saham atau indeks harga saham gabungan Bursa Efek Indonesia (BEI). Data dapat diunduh dari Yahoo Finance pada tautan <https://finance.yahoo.com/lookup>. Jangka waktu pengumpulan data adalah pada periode 1 Januari 2020 s.d. 31 Desember 2020 atau selama periode pandemi Covid-19 dimana pergerakan indeks pasar saham menjadi lebih tidak pasti.

Selanjutnya, alat dukung yang digunakan dalam penelitian ini adalah komputer untuk melakukan perancangan dan implementasi struktur *CNN-LSTM* serta uji coba dan evaluasi kinerja dari model peremalan data deret waktu yang dibuat. Dalam hal ini, komputer juga digunakan untuk melakukan implementasi struktur menggunakan lingkungan pemrograman Python 2.7 dan *library* Keras.

Sebagaimana telah disampaikan pada bagian Pendahuluan, tujuan utama dari penelitian ini adalah untuk dapat membangun sebuah struktur jaringan saraf tiruan yang dapat mempelajari pola gerak data deret waktu dan memprediksi nilainya di waktu mendatang sert dan menguji kinerja dari model tersebut khususnya untuk memprediksi pergerakan nilai indeks BEI. Oleh sebab itu, analisa hasil yang dilakukan pada penelitian ini akan fokus pada tingkat akurasi dari stuktur *CNN-LSTM* dalam memprediksi pergerakan nilai indeks pasar saham BEI.

Selanjutnya, untuk melakukan evaluasi kinerja dari *CNN-LSTM* dalam memprediksi pergerakan indeks pasar saham, dalam penelitian ini digunakan nilai *RMSE*

sebagai kriteria utama metode evaluasi. Adapun nilai *RMSE* dihitung menggunakan Rumus 8.



Gambar 6. Struktur rinci model *CNN-LSTM* yang diusulkan disertai dengan perubahan dimensi data *input* dan *output* pada tiap *layer*-nya

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (\hat{y}_i - y_i)^2}, \quad (8)$$

dimana \hat{y}_i adalah hasil prediksi untuk indeks pasar saham pada waktu i dan y_i adalah nilai nyata atau sebenarnya dari indeks pasar saham pada waktu i .

3. Hasil dan Pembahasan

3.1. Analisa Grafik Perbandingan Hasil Prediksi

Sebagaimana telah disampaikan implementasi model *CNN-LSTM* yang dibangun dalam penelitian ini dilakukan menggunakan lingkungan pemrograman Python 2.7 dan *library* Keras. Modul utama yang dibuat terdiri dari modul pembangunan model *CNN-LSTM*, modul prediksi, dan modul penghitungan nilai *error* untuk evaluasi kinerja model *CNN-LSTM* dalam memprediksi nilai indeks pasar saham BEI.

Implementasi dari modul pembangunan model *CNN-LSTM* diberikan sebagai berikut:

```

Modul Build_Model
Input: X, y, n_timesteps
Output: model
cnn = Sequential()
cnn.add(Conv1D(64, 1, activation='relu', input_shape=(n_timesteps, 1)))
cnn.add(MaxPooling1D())
cnn.add(Flatten())
model = Sequential()
model.add(TimeDistributed(cnn))
model.add(LSTM(50, activation='relu'))
model.add(Dense(1))
model.compile(optimizer='adam', loss='mse')
# fit model
model.fit(X, y, epochs=500, verbose=0)
return(model)
    
```

Masukan pada modul `Build_Model` adalah X , yaitu kumpulan data yang akan menjadi masukan prediksi; y , yaitu target nilai prediksi; dan $n_timesteps$ yaitu jumlah langkah data ke belakang yang digunakan untuk memprediksi data pada waktu berikutnya dan menghasilkan keluaran berupa model *CNN-LSTM* yang telah dilatih. Selanjutnya, implementasi untuk modul prediksi diberikan sebagai berikut:

Modul Prediksi

Input: $X, model, n_timesteps$

Output: y_hat

```
x_input = X
numrows = len(x_input)
numcols = len(x_input[0])
n_features = 1
n_seq = 2
X = X.reshape((X.shape[0], n_seq,
               n_timesteps, n_features))
# demonstrate prediction
y_hat = np.array([])
for i in range(0, numRows, 1):
    x = np.array(x_input[i, :])
    x = x.reshape((1, n_seq, n_timesteps,
                  n_features))
    y_hat = np.append(y_hat, model.predict(x,
                                           verbose=0))
return(y_hat)
```

Pada modul `Prediksi`, masukannya adalah X , yaitu kumpulan data yang akan menjadi masukan proses prediksi; $n_timesteps$ yaitu jumlah langkah data ke belakang yang digunakan untuk memprediksi data pada waktu berikutnya; dan juga $model$, yaitu model *CNN-LSTM* yang telah dilatih. Keluaran dari model ini adalah y_hat , yaitu hasil prediksi. Kemudian, implementasi modul penghitungan kesalahan pada tiap titik waktu dan juga *RMSE* diberikan sebagai berikut:

Modul RMSE

Input: $x, y, blim, ulim$

Output: $rmse, errate$

```
mse = np.square(np.subtract(x[blim-1:ulim-1],
                             y[blim-steps:ulim-steps])).mean(axis=0)
rmse = np.sqrt(mse)
errate = abs(x[blim-1:ulim-1] - y[blim-steps:ulim-steps])
return(rmse, errate)
```

Pada modul *RMSE*, yang menjadi masukan adalah x , yaitu nilai asli dari indeks yang diprediksi; y , nilai hasil prediksi; $blim$ adalah batas bawah data deret waktu yang diproses; dan $ulim$ adalah batas atas data yang diproses. Keluaran dari modul ini adalah nilai *RMSE* dan nilai kesalahan prediksi pada tiap titik waktu (*errate*).

Perbandingan hasil prediksi pada tahap pelatihan dan juga tahap uji ditampilkan pada Gambar 7 dan 8 baik untuk model *CNN* dan *LSTM* secara individu dan juga model *ensemble CNN-LSTM* yang diusulkan dalam penelitian ini. Gambar 7 menampilkan grafik perbandingan antara nilai indeks pasar saham BEI hasil prediksi terhadap nilai aslinya pada tahap pelatihan, dan Gambar 8 menampilkan grafik perbandingan antara nilai indeks pasar saham BEI hasil prediksi terhadap nilai aslinya pada tahap pengujian.

Evaluasi kinerja terhadap model *CNN-LSTM* yang dibangun dilakukan menggunakan data uji setelah model *CNN-LSTM* dilatih menggunakan data latih. Adapun data latih yang digunakan adalah sejumlah 170 sampel data harian nilai indeks pasar saham BEI yang merupakan 70% bagian dari keseluruhan 242 data harian nilai indeks pasar saham pada periode 1 Januari 2020 s.d. 31 Desember 2020, dan sisanya digunakan sebagai data uji. Data harian nilai indeks pasar saham BEI tersebut diunduh dari Yahoo Finance pada tautan <https://finance.yahoo.com/lookup>.

Secara umum dapat dilihat bahwa grafik nilai indeks pasar saham BEI yang merupakan hasil prediksi dari *CNN-LSTM* memiliki pola gerak yang mirip dan hampir berdempetan dengan laju grafik nilai indeks pasar saham BEI yang asli. Hal ini terjadi baik pada grafik perbandingan pada tahap pelatihan maupun pengujian. Fakta ini memberikan gambaran yang jelas bahwa model *CNN-LSTM* dengan konfigurasi parameter yang dibuat pada penelitian ini terbukti efektif untuk memprediksi nilai harian indeks pasar saham BEI.

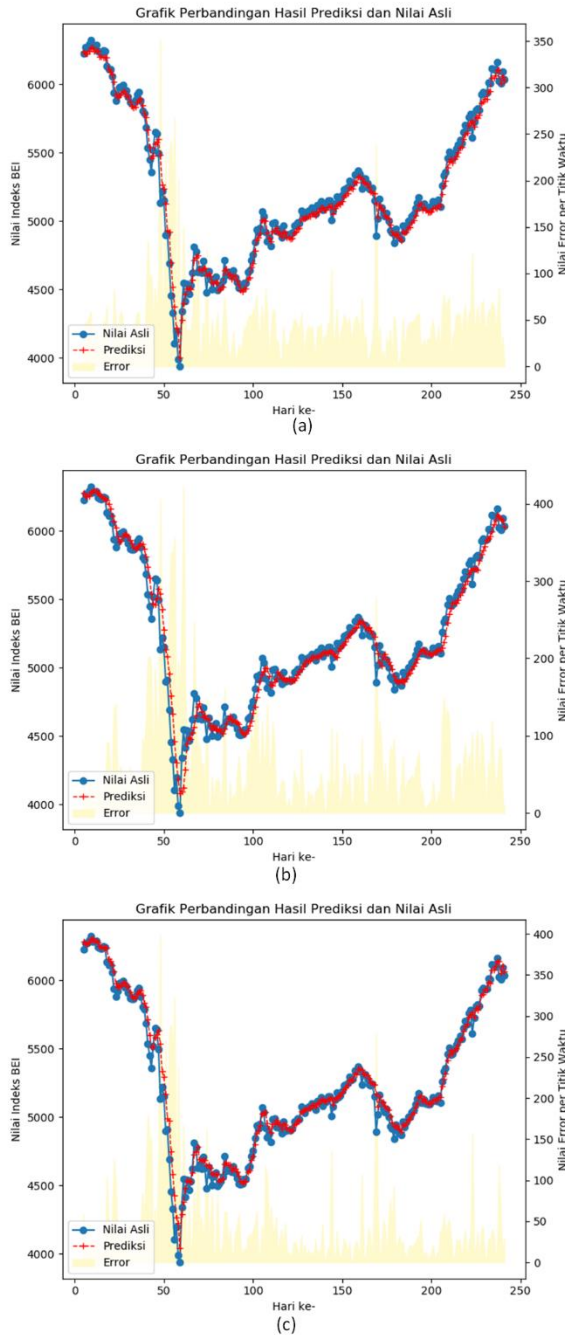
Gambar 7 dan 8 juga menampilkan informasi mengenai *error rate* hasil prediksi, yang dihitung berdasarkan nilai mutlak dari selisih hasil prediksi dan nilai aslinya pada tiap titik waktu baik pada tahap pelatihan maupun pengujian. Grafik *error rate* juga mengindikasikan bahwa rata-rata *error rate* hasil prediksi berada pada kisaran yang dan relatif kecil dan dapat diterima.

Selain itu, mengingat bahwa data deret waktu yang diproses berada pada cakupan periode pandemi Covid-19, dimana pergerakan berbagai indikator ekonomi menjadi lebih tidak pasti, model *CNN-LSTM* juga terkonfirmasi tetap memiliki kemampuan untuk memprediksi nilai dari data deret waktu keuangan dengan kinerja yang baik. Selisih terbesar antara prediksi dengan nilai asli terjadi pada kisaran hari perdagangan ke-55, yang jika dipetakan ke hari kalender jatuh pada awal Maret 2020 dimana pandemi Covid-19 mulai mempengaruhi kondisi perekonomian nasional. Namun setelah itu, tampak bahwa nilai *error rate* terus menurun yang mana mengindikasikan kemampuan model *CNN-LSTM* untuk beradaptasi dengan perubahan pola gerak nilai indeks pasar saham.

3.2. Analisa RMSE

Sebagaimana telah disampaikan sebelumnya, dalam penelitian ini kinerja dari model *CNN-LSTM* juga dievaluasi dengan melakukan penghitungan nilai *RMSE*. Penghitungan *RMSE* dilakukan baik pada tahap pelatihan dan juga pada tahap pengujian. Selain itu, untuk membuktikan bahwa membangun sebuah struktur *ensemble CNN-LSTM* yang merupakan integrasi dari model *CNN* dan *LSTM*, pada penelitian ini juga dilakukan uji coba prediksi nilai indeks pasar saham BEI dengan jumlah data dan konfigurasi data latih serta data uji yang sama menggunakan model *CNN* dan model *LSTM* yang berdiri sendiri-sendiri. Perbandingan nilai

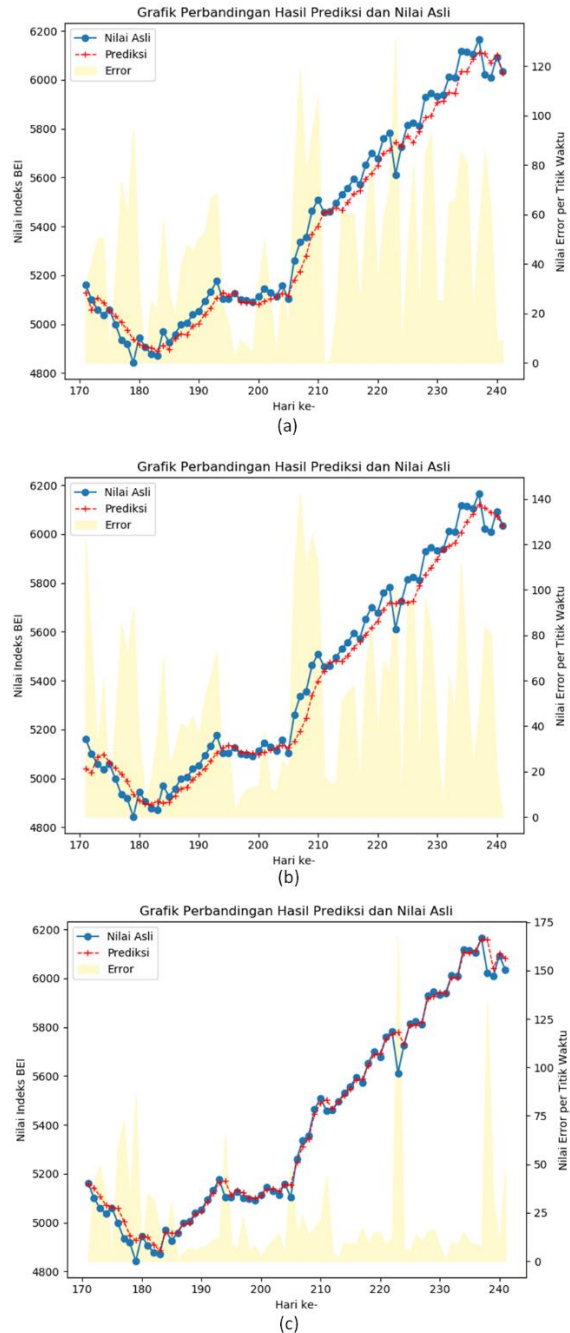
RMSE untuk masing-masing model dan tiap tahapannya diberikan pada Tabel 2 dan diilustrasikan pada Gambar 9.



Gambar 7. Perbandingan hasil prediksi terhadap nilai asli indeks pasar saham pada periode pelatihan (170 hari pertama pada periode 1 Januari 2020 s.d. 31 Desember 2020), (a) pada model CNN, (b) pada model LSTM dan (c) pada model CNN-LSTM.

Tabel 2. Perbandingan nilai RMSE pada model CNN, LSTM dan CNN-LSTM pada prediksi indeks pasar saham BEI

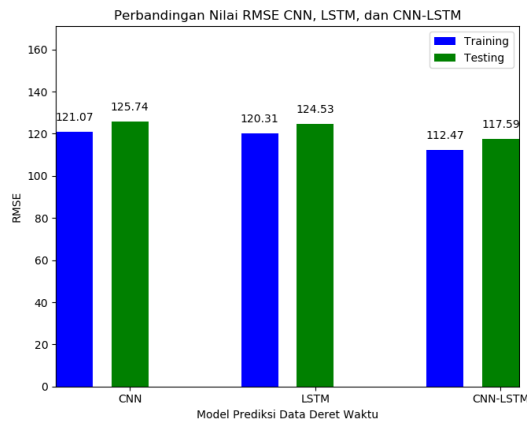
Model	RMSE	
	Pelatihan	Pengujian
CNN	121.07	125.74
LSTM	120.31	124.53
CNN-LSTM	112.47	117.59



Gambar 8. Perbandingan hasil prediksi terhadap nilai asli indeks pasar saham pada periode pengujian (72 hari terakhir pada periode 1 Januari 2020 s.d. 31 Desember 2020), (a) pada model CNN, (b) pada model LSTM dan (c) pada model CNN-LSTM.

Berdasarkan nilai RMSE yang ditampilkan pada Tabel 2, maka dapat diambil kesimpulan bahwa akurasi terbaik diberikan oleh model CNN-LSTM, dengan peningkatan akurasi pada model CNN-LSTM sebesar 7.6% (CNN) dan 8.7% (LSTM) pada tahap *training* serta 2.7% (CNN) dan 1.6% (LSTM) pada tahap *testing*. Dengan demikian, fakta ini mengkonfirmasi teori yang menyatakan bahwa dengan menggabungkan berbagai fitur serta keunggulan yang dimiliki oleh model atau algoritma yang berbeda menjadi sebuah *ensemble* dapat memberikan hasil yang

lebih baik dalam penyelesaian sebuah masalah, yang dalam hal ini adalah untuk memprediksi nilai indeks pasar saham BEI.



Gambar 9. Grafik perbandingan nilai *RMSE* model *CNN*, *LSTM* dan *CNN-LSTM* pada periode pelatihan dan pegujian prediksi nilai indeks pasar saham BEI

4. Kesimpulan

Penelitian ini mengusulkan sebuah model *CNN-LSTM* untuk memprediksi nilai indeks pasar saham satu tahapan waktu ke depan (keesokan harinya). Metode yang digunakan adalah peramalan data deret waktu variabel tunggal (*univariate*), dimana prediksi indeks pasar saham dilakukan dengan hanya mempelajari pola gerak indeks itu sendiri pada masa yang lampau. Pada model yang dibangun, *CNN* digunakan untuk ekstraksi fitur dari data masukan sedangkan *LSTM* memiliki peran untuk mempelajari data fitur yang diekstraksi serta melakukan proses akhir prediksi nilai indeks pasar saham keesokan harinya. Dalam uji coba digunakan data yang relevan dari Bursa Efek Indonesia sebagai data latih dan data uji untuk memverifikasi hasil eksperimen. Hasil percobaan menunjukkan bahwa *CNN-LSTM* memiliki akurasi prediksi tertinggi dan kinerja terbaik (memiliki nilai *RMSE* terkecil) dibandingkan dengan model *CNN* dan *LSTM* secara individu.

Dengan demikian, dapat disimpulkan bahwa *CNN-LSTM* efektif untuk peramalan nilai indeks pasar saham dan dapat memberikan referensi yang relevan bagi investor untuk mengambil keputusan investasinya. Selain itu, *CNN-LSTM* juga memberikan alternatif praktis bagi penelitian yang berhubungan dengan pengembangan model untuk analisa data deret waktu keuangan.

Ucapan Terimakasih

Tim peneliti mengucapkan terima kasih kepada Direktorat Riset dan Pengabdian Masyarakat, Kemristek/BRIN Republik Indonesia atas pembiayaan yang diberikan melalui skema Hibah Penelitian Dasar tahun anggaran 2021.

Daftar Rujukan

- [1] R. Vanaga and B. Sloka, "Financial and capital market commission financing: aspects and challenges," *Journal of Logistics, Informatics and Service Science*, vol. 7, no. 1, pp. 17–30, 2020.
- [2] L. Zhang and H. Kim, "The influence of financial service characteristics on use intention through customer satisfaction with mobile fintech," *Journal of System and Management Sciences*, vol. 10, no. 2, pp. 82–94, 2020.
- [3] L. Badea, V. Ionescu, and A.-A. Guzun, "What is the causal relationship between stoxx europe 600 sectors? But between large firms and small firms?" *Economic Computation And Economic Cybernetics Studies And Research*, vol. 53, no. 3, pp. 5–20, 2019.
- [4] J. Sousa, J. Montevechi, and R. Miranda, "Economic lot-size using machine learning, parallelism, metaheuristic and simulation," *Journal of Logistics, Informatics and Service Science*, vol. 18, no. 2, pp. 205–216, 2019.
- [5] A. Coser, M. M. Maer-Matei, and C. Albu, "Predictive models for loan default risk assessment," *Economic Computation And Economic Cybernetics Studies And Research*, vol. 53, no. 2, pp. 149–165, 2019.
- [6] R. Qiao, "Stock prediction model based on neural network," *Operations Research and Management Science*, vol. 28, no. 10, pp. 132–140, 2019.
- [7] C. Jung and R. Boyd, "Forecasting UK stock prices," *Applied Financial Economics*, vol. 6, no. 3, pp. 279–286, 1996.
- [8] W. Bleessner and P. Liicoff, "Predicting stock returns with bayesian vector autoregressive," *Data Analysis, Machine Learning and Applications*, vol. 1, pp. 499–506, 2005.
- [9] A. Adebisi, A. Adewumi, and C. Ayo, "Stock price prediction using the ARIMA model," in *Proceedings of the 2014 UKSim-AMSS 16th International Conference on Computer Modelling and Simulation*, IEEE, Cambridge, UK, March 2014.
- [10] C. Zhang, X. Cheng, and M. Wang, "An empirical research in the stock market of Shanghai by GARCH model," *Operations Research and Management Science*, vol. 4, pp. 144–146, 2005.
- [11] Q. Yang and C. Wang, "A study on forecast of global stock indices based on deep LSTM neural network," *Statistical Research*, vol. 36, no. 6, pp. 65–77, 2019.
- [12] K.-S. Moon and H. Kim, "Performance of deep learning in prediction of stock market volatility," *Economic Computation And Economic Cybernetics Studies And Research*, vol. 53, no. 2, pp. 77–92, 2019.
- [13] B. Kuechler, V. Vaishnavi, "Design Science Research in Information Systems," 2004.
- [14] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.
- [15] B. S. Kim and T. G. Kim, "Cooperation of simulation and data model for performance analysis of complex systems," *International Journal of Simulation Modelling*, vol. 18, no. 4, pp. 608–619, 2019.
- [16] Y. Sun, Y. Liang, and W. Zhang, "Optimal partition algorithm of the RBF neural network and its application to financial time series forecasting," *Neural Computing and Applications*, vol. 14, pp. 1441–1449, 2005.
- [17] D. Ciresan, U. Meier, and J. Schmidhuber, "Multi-column deep neural networks for image classification," *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3642–3649, 2012.
- [18] M.L. Brocardo, I. Traore, I. Woungang, and M.S. Obaidat, "Authorship verification using deep belief network systems," *Int J Commun Syst.*, pp. 1013-1021, 2017.
- [19] D. Silver, A. Huang, C. Maddison, A. Guez, L. Sifre, G. Driessche, J. Schrittwieser, I. Antonoglou, and V. Panneershelvam, "Mastering the game of Go with deep neural networks and tree search," *Nature*, vol. 529, no. 7587, pp. 484–489, 2016.
- [20] A. Marblestone, G. Wayne, and K. Kording, "Toward an Integration of Deep Learning and Neuroscience," *Frontiers in Computational Neuroscience*, vol. 21, no. 3, pp. 213-230, 2016.

- [21] Y. Bengio, A. Courville, and P. Vincent, "Representation Learning: A Review and New Perspectives," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 8, pp. 1798–1828, 2013.
- [22] L. Deng, and D. Yu, "Deep Learning: Methods and Applications," *Foundations and Trends in Signal Processing*, vol. 7, no. 3–4, pp. 1–199, 2014.
- [23] Y. Bengio, "Learning Deep Architectures for AI," *Foundations and Trends in Machine Learning*, vol. 2, no. 1, pp. 1–127, 2009.
- [24] Y. LeCun, Y. Bengio, and G. Hinton, "Deep Learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.
- [25] J. Schmidhuber, "Deep Learning in Neural Networks: An Overview," *Neural Networks*, vol. 61, pp. 85–117, 2015.
- [26] A. Yadav, C. K. Jha, and A. Sharan, "Optimizing LSTM for time series prediction in Indian stock market," *Procedia Computer Science*, vol. 167, pp. 2091–2100, 2020.
- [27] H. Y. Kim and C. H. Won, "Forecasting the volatility of stock price index: a hybrid model integrating LSTM with multiple GARCH-type models," *Expert Systems with Applications*, vol. 103, pp. 25–37, 2018.
- [28] N. C. Petersen, R. Christoffer, F. Rodrigues, and F. C. Pereira, "Multi-output bus travel time prediction with convolutional LSTM neural network," *Expert Systems with Applications*, vol. 120, pp. 426–435, 2019.
- [29] W. Lu, J. Li, Y. Li, A. Sun, and J. Wang, "A CNN-LSTM-Based Model to Forecast Stock Prices," *Complexity*, <https://doi.org/10.1155/2020/6622927>, 2020.
- [30] J. Brownlee, "Deep Learning for Time Series Forecasting Predict the Future with MLPs, CNNs and LSTMs in Python," Edition v.1.7, *Machine Learning Mastery*, San Juan, PR 00901, 2020.