

## MESIN CATUR BERBASIS *NEURAL NETWORK* MENGGUNAKAN LONG SHORT TERM MEMORY (LSTM)

Rafi Indra Fattah<sup>\*1</sup>, Putra Pandu Adikara<sup>2</sup>, Budi Darma Seriawan<sup>3</sup>

<sup>1,2,3</sup>Universitas Brawijaya, Malang

Email: <sup>1</sup>rafiplg@student.ub.ac.id, <sup>2</sup>adikara.putra@ub.ac.id, <sup>3</sup>s.budidarma@ub.ac.id

<sup>\*</sup>Penulis Korespondensi

(Naskah masuk: 13 Februari 2024, diterima untuk diterbitkan: 09 Mei 2025)

### Abstrak

Catur diperkirakan memiliki sekitar  $10^{43}$  kemungkinan posisi. Angka tersebut jauh melampaui kemampuan komputasi komputer yang ada saat ini, sehingga mengembangkan sebuah mesin catur dengan mempertimbangkan seluruh kemungkinan posisi dianggap tidak memungkinkan. Saat ini, penggunaan Neural Network pada pengembangan mesin catur sedang mengalami peningkatan dan telah membawa hasil yang menjanjikan sejak pertama kali diperkenalkan oleh AlphaZero milik Google DeepMind pada tahun 2017. Penelitian ini bertujuan untuk membawa potensi pendekatan baru pada ranah pengembangan mesin catur berbasis Neural Network dengan memperkenalkan mesin catur Deepfish yang melakukan gerakan berdasarkan keluaran model Long Short Term Memory (LSTM). Menggunakan lebih dari 57.000 pertandingan yang terbagi menjadi 1.200.000 posisi, model dilatih untuk memprediksi langkah berikutnya oleh putih untuk sebuah rangkaian gerakan yang diberikan. Model meraih *loss* sebesar 3,01 dan *Average Centipawn Loss* (ACPL) sebesar 219 pada data uji. Deepfish meraih 2 kemenangan, 72 kekalahan, dan 10 hasil seri pada tahap pengujian. Hasil yang tidak memuaskan ini dapat disebabkan oleh subjektivitas data terhadap cara berpikir pemain, menghasilkan kurangnya pola gerakan yang signifikan untuk dipelajari oleh model.

**Kata kunci:** Mesin Catur, Prediksi Gerakan Catur, long short term memory (LSTM), universal chess interface (UCI), average centipawn loss (ACPL)

## *NEURAL NETWORK BASED CHESS ENGINE USING LONG SHORT TERM MEMORY (LSTM)*

### Abstract

*Chess has been estimated to have around 1043 possible positions. This number surpasses the computing ability of any computer available, therefore, building a chess engine that considers every possible position is deemed impractical. Currently, the use of neural network in chess engine development is on the rise and has been delivering promising results since the introduction of Google DeepMind's AlphaZero in 2017. This research aims to bring a new potential approach to the field of neural network based chess engine development by introducing Deepfish chess engine that uses a Long Short Term Memory (LSTM) model as move generator. Trained on more than 57.000 games broken down into more than 1.200.000 positions, the model is trained to predict the next move played by white for a given sequence of moves. The model achieved a loss of 3.01 and an Average Centipawn Loss (ACPL) of 219 on the validation set. Deepfish achieved 2 wins, 72 losses, and 10 draws on the testing, showing a lack of board and contextual awareness. This unsatisfactory results are likely due to the subjectivity of the data to the player's way of thinking, resulting in lack of significant move pattern to be learned by the model.*

**Keywords:** chess engine, chess moves prediction, long short term memory (LSTM), universal chess interface (UCI), average centipawn loss (ACPL)

### 1. PENDAHULUAN

Upaya untuk membuat sebuah program komputer yang dapat bermain catur telah dimulai sejak tahun 1950-an, program ini sering juga disebut dengan *chess engine* yang berarti mesin catur, meski begitu, catur telah diperkirakan memiliki sekitar  $10^{43}$

kemungkinan posisi yang dapat terjadi (Shannon, 1950). Angka tersebut jauh melebihi kapasitas dan kemampuan komputasi komputer yang ada bahkan hingga saat ini, maka dari itu, membangun mesin catur dengan mempertimbangkan seluruh

kemungkinan posisi yang dapat terjadi dianggap tidak memungkinkan.

Mesin catur tradisional pada umumnya menggunakan sebuah skor untuk menentukan pihak mana yang unggul pada posisi tersebut, skor ini sering disebut sebagai *eval score*. Menentukan *eval score* pada mesin catur tradisional dilakukan dengan metode Alpha-Beta Search dan fungsi evaluasi yang dibuat secara eksplisit oleh pengembang mesin catur (Klein, 2022). Fungsi evaluasi pada umumnya mempertimbangkan berbagai atribut permainan seperti keamanan buah raja, struktur pion, dan ruang gerak buah untuk mengevaluasi sebuah posisi, semakin akurat pengembang mesin catur dalam merumuskan fungsi evaluasi, maka semakin baik pula mesin catur yang dihasilkan. (David, Netanyahu dan Wolf, 2016).

Meskipun demikian, belakangan ini penggunaan *deep learning* untuk menggantikan fungsi evaluasi tradisional dalam pengembangan mesin catur telah membuahkan hasil yang menjanjikan. Pada tahun 2017, mesin catur AlphaZero milik Google DeepMind yang mengimplementasikan Neural Network sebagai pengganti fungsi evaluasi tradisional, telah berhasil mengalahkan mesin catur Stockfish yang merupakan salah satu mesin catur terkuat pada saat itu. Kini, AlphaZero telah menjadi inspirasi bagi pengembangan-pengembangan mesin catur berbasis Neural Network terbaru seperti Leela Chess Zero dan Stockfish Neural Network Updatable Efficiently (Stockfish NNUE), keduanya dianggap sebagai salah satu mesin catur terkuat di dunia, mengindikasikan bahwa penggunaan Neural Network dalam pengembangan mesin catur akan terus mendominasi untuk beberapa waktu ke depan (Panchal, Mishra dan Shrivastava, 2021).

Long Short Term Memory atau LSTM adalah salah satu jenis Neural Network yang dapat digunakan pada data berurutan atau *sequential data* (Selle, Yudistira dan Dewi, 2022), karena kemampuannya dalam mengingat informasi yang penting dan melupakan informasi yang tidak penting dalam sebuah rangkaian informasi yang berurutan (Santosa, 2021). Kemampuan LSTM ini berpotensi untuk dapat diterapkan dalam pengembangan mesin catur. Berdasarkan uraian tersebut, penelitian ini memperkenalkan Deepfish, sebuah mesin catur berbasis Neural Network yang melakukan gerakan berdasarkan keluaran model LSTM.

## 2. LANDASAN KEPUSTAKAAN

### 2.1. Long Short Term Memory (LSTM)

Long Short Term Memory atau LSTM adalah salah satu jenis Neural Network yang memiliki kemampuan penyimpanan informasi untuk jangka waktu yang lama, sehingga dapat bekerja dengan baik pada data yang bersifat berurutan atau *sequential*. Perhitungan dari LSTM melibatkan 4 operasi yang

disebut sebagai *forget gate*, *input gate*, *candidate gate*, dan *output gate*.

$$f_t = \sigma(h_{t-1}W_{fh}^T + x_tW_{fx}^T + b_f) \quad (1)$$

$$\tilde{c}_t = \tanh(h_{t-1}W_{\tilde{c}h}^T + x_tW_{\tilde{c}x}^T + b_c) \quad (2)$$

$$i_t = \sigma(h_{t-1}W_{ih}^T + x_tW_{ix}^T + b_i) \quad (3)$$

$$c_t = f_t \cdot c_{t-1} + i_t \cdot \tilde{c}_t \quad (4)$$

$$o_t = \sigma(h_{t-1}W_{ox}^T + x_tW_{ox}^T + b_o) \quad (5)$$

$$h_t = o_t \cdot \tanh(c_t) \quad (6)$$

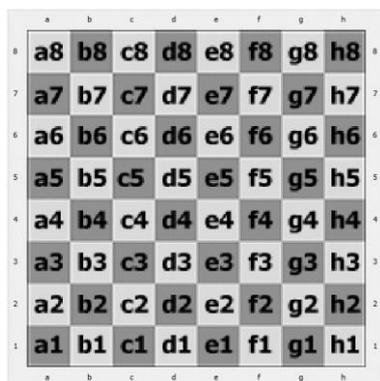
Pada Persamaan-persamaan tersebut,  $f_t$  adalah nilai *forget gate* ke-t,  $\tilde{c}_t$  adalah nilai *candidate gate* ke-t,  $i_t$  adalah nilai *input gate* ke-t,  $c_t$  adalah nilai *cell state* ke-t,  $o_t$  adalah nilai *output gate* ke-t,  $W_h$  adalah bobot *hidden state*,  $W_x$  adalah bobot masukan,  $b$  adalah bias,  $\sigma$  adalah fungsi aktivasi *sigmoid*,  $\tanh$  adalah fungsi aktivasi *tanh*.

### 2.2. Notasi Gerak Catur

Setiap petak pada papan catur diidentifikasi menggunakan sebuah koordinat. Pada Gambar 1 dapat dilihat sistem koordinat petak-petak papan catur yang digunakan dan dikenali secara resmi oleh Internasional Chess Federation (FIDE). Sebuah gerakan dapat dituliskan sebagai koordinat petak buah sebelum dan sesudah digerakkan, sebagai contoh, *e2e4* menyatakan bahwa buah catur yang sebelumnya berada di petak *e2* kini digerakkan ke petak *e4*. Notasi ini disebut sebagai *Long Algebraic Notation*. Gerakan juga dapat dituliskan menggunakan inisial buah dalam bahasa inggris yang digabungkan dengan koordinat petak buah setelah digerakkan, sebagai contoh, gerakan *Qh5* dapat dibaca sebagai "*queen to h5*" atau "menteri ke h5". Notasi gerak ini disebut sebagai notasi *Short Algebraic Notation* (SAN)

### 2.3. Universal Chess Interface (UCI)

Universal Chess Interface atau UCI adalah sebuah standar protokol komunikasi yang paling umum digunakan dalam pengembangan mesin catur. UCI memungkinkan sebuah mesin catur untuk berkomunikasi dan dioperasikan melalui *stdin* dan *stdout*. Mesin catur yang dikembangkan menggunakan protokol UCI kemudian dapat dimuat ke sebuah antarmuka catur dan dioperasikan melalui antarmuka tersebut. Tabel 1 mengandung contoh komunikasi yang menerapkan protokol UCI.



Gambar 1. Sistem Koordinat Papan Catur

Tabel 1. Komunikasi Menggunakan UCI

masukan	nilai kembalian
uci	uciok
isready	readyok
ucinewgame	
position startpos moves e2e4	
isready	readyok
go	bestmove e7e5

### 2.4. Centipawn dan Average Centipawn Loss

Centipawn (cp) adalah satuan yang digunakan untuk mengukur keunggulan dalam permainan catur. Satu Centipawn setara dengan 0,01 pion, jika putih memiliki dua pion sementara hitam hanya memiliki satu pion, maka putih dikatakan unggul sebanyak satu pion, atau 100 cp. Keunggulan yang dinyatakan dalam satuan cp disebut Centipawn score. Skor ini dinyatakan dari sudut pandang putih; nilai Centipawn score sebesar +100 cp berarti putih unggul sebanyak 100 cp, sedangkan nilai -100 cp menunjukkan bahwa hitam unggul sebanyak 100 cp (Mehta et al., 2020).

Centipawn dan Centipawn Score menjadi dasar bagi berbagai metrik untuk mengukur kinerja pemain catur, seperti Centipawn loss yang mengukur perubahan Centipawn score untuk setiap gerakan yang dilakukan pemain, dan Average Centipawn Loss (ACPL) yang menghitung rata-rata Centipawn loss pemain selama sebuah pertandingan.

### 2.5. Stockfish

adalah proyek mesin catur open source sekaligus menjadi salah satu mesin catur terkuat saat ini. Stockfish telah memenangkan kejuaraan Top Chess Engine Championship atau TCEC sebanyak 14 kali, dan selalu merebut gelar juara pertama atau kedua.

Sampai dengan versi ke-11, Stockfish masih menggunakan fungsi evaluasi tradisional, setelah itu, Stockfish mulai menerapkan sistem Neural Network Updatable Efficiently (NNUE). Neural Network tersebut dilatih untuk melakukan evaluasi sebagaimana fungsi evaluasi tradisional, sehingga mempersingkat waktu evaluasi dan memungkinkan evaluasi yang lebih mendalam (Maharaj, Polson dan Turk, 2022).

Kinerja Stockfish yang sangat baik menjadikannya sebagai acuan utama dalam dunia

catur. Stockfish sering digunakan untuk menilai kualitas sebuah gerakan serta menghitung akurasi gerakan. Dua platform catur daring terbesar, chess.com dan lichess.org, keduanya menggunakan Stockfish untuk menganalisis gerakan pemain.

### 2.6. Akurasi Gerakan Catur

Persentase akurasi menunjukkan kedekatan terhadap gerakan terbaik. Akurasi 100% menyatakan gerakan terbaik, sedangkan akurasi 0% menyatakan gerakan terburuk. Perhitungan akurasi gerakan pada penelitian ini dilakukan menggunakan persamaan akurasi milik lichess.org, yang juga menggunakan Stockfish untuk menghitung Centipawn score. Persamaan akurasi gerakan dapat dilihat pada Persamaan 7 dan Persamaan 8.

$$W_t = 50 + 50 \left( \frac{2}{1 + e^{-0,00368208 * cp_t}} - 1 \right) \quad (7)$$

$$A_t = 103,1668 * e^{-0,04354(W_{t-1} - W_t)} - 3,1169 \quad (8)$$

Pada Persamaan-persamaan tersebut,  $W_t$  adalah peluang menang pada gerakan ke-t,  $cp_t$  adalah nilai Centipawn score pada gerakan ke-t, dan  $A_t$  adalah akurasi gerakan ke-t.

## 3. METODE PENELITIAN

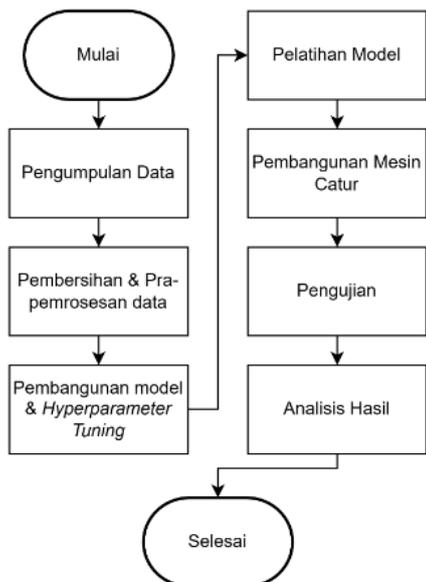
### 3.1. Diagram Alir Penelitian

Tahapan pertama dari penelitian ini adalah pengumpulan data dari basis data milik lichess.org (<https://database.lichess.org>). Tahapan kedua adalah pembersihan dan pra-pemrosesan data dengan mengambil data pertandingan yang dimenangkan oleh putih dalam kurang dari 100 gerakan, membuat kamus gerakan, mengkodekan gerakan menjadi numerik, membentuk N-Grams, memberikan pre-padding, dan memisahkan fitur dari label. Tahapan ketiga adalah pembangunan model menggunakan Pytorch, dan hyperparameter tuning menggunakan metode grid search. Tahapan keempat adalah pelatihan model menggunakan konfigurasi hyperparameter terbaik berdasarkan hasil hyperparameter tuning. Tahapan kelima adalah penulisan kode program menggunakan bahasa pemrograman Python dan menerapkan protokol UCI. Tahapan keenam adalah pengujian dengan menandingkan Deepfish melawan mesin catur Komodo level 1, Maia Chess 1, dan Stockfish 16.

## 4. HASIL DAN PEMBAHASAN

### 4.1. Hasil Pelatihan Model

Pelatihan model dilakukan sebanyak 10 epoch sebelum dihentikan oleh callback.



Gambar 2. Diagram Alir Penelitian

Selama 10 *epoch* pelatihan, nilai *loss* data latih mengalami penurunan dari angka 4,04 hingga 3,01, nilai *loss* data uji mengalami penurunan dari angka 3,68 hingga 3,12, nilai ACPL data uji mengalami penurunan dari angka 245,74 hingga 219,51, sedangkan jumlah *illegal move* dari prediksi data uji mengalami penurunan dari angka 85.436 hingga 48.840.

#### 4.2. Hasil Penguujian Melawan Mesin Catur Lain

Penguujian melawan mesin catur lain dilakukan dengan melawan mesin catur Komodo Level 1, Maia Chess 1, dan Stockfish 16, dengan jumlah pertandingan sebanyak 50 pertandingan untuk masing-masing lawan. Pemeriksaan lebih lanjut menunjukkan terdapat pertandingan duplikat, yang kemudian dihapus dari data penguujian. Hasil yang didapatkan oleh Deepfish kurang memuaskan, dengan total 2 kemenangan, 72 kekalahan, dan 10 hasil seri.

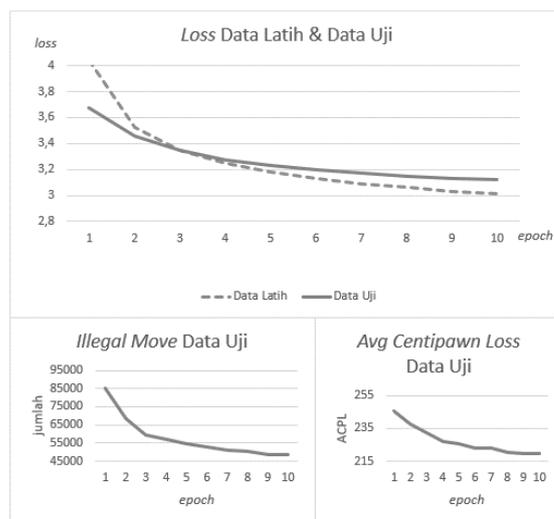
Tabel 2. Hasil Penguujian Melawan Mesin Catur Lain

bertanding melawan	rating lawan	skor*	rata-rata akurasi
Komodo level 1	~100	1-14-9	45%
Maia Chess 1	~1100	1-22-1	72%
Stockfish 16	~3000	0-36-0	80,2%

\*keterangan: skor dalam format menang-kalah-seri

#### 4.3. Analisis Hasil

Penguujian melawan mesin catur lain menunjukkan kinerja Deepfish yang kurang memuaskan. Namun, analisis lebih lanjut menunjukkan bahwa Deepfish bermain dengan cukup baik pada fase pembukaan atau *opening*, dengan melakukan gerakan-gerakan yang bermakna dan masuk akal.



Gambar 3. Grafik Pelatihan Model

Satu kemenangan yang didapat saat melawan Maia Chess 1 dan Komodo Level 1 juga menunjukkan bahwa Deepfish mampu mengidentifikasi pola-pola sekakmat sederhana seperti pola *scholar's mate*. Namun, memasuki fase pertengahan pertandingan, Deepfish seolah kehilangan konteks dan situasi dari papan catur, sehingga mulai melakukan gerakan-gerakan yang tidak akurat. Secara keseluruhan, gerakan-gerakan yang dilakukan Deepfish pada fase pertengahan ini masih memiliki maksud dan tujuan yang dapat dipahami, namun Deepfish gagal dalam memperhitungkan respons apa yang dapat dilakukan oleh lawan, serta dampak yang dapat ditimbulkan oleh gerakan tersebut. Pada fase akhir pertandingan, Deepfish telah kehilangan sepenuhnya konteks dari papan, dan hanya menggerakkan buah raja, melakukan gerakan yang sama berulang kali, atau mendorong pion tanpa tujuan yang jelas.

Hilangnya pemahaman terhadap konteks papan tersebut dapat disebabkan oleh data yang inkonsisten. Data pada penelitian ini diambil dari pertandingan-pertandingan yang dimainkan oleh lebih dari 16.000 pemain di platform lichess.com. Masing-masing pemain pasti memiliki strategi bermain dan cara berpikir yang berbeda-beda, membuat data menjadi inkonsisten dan subjektif terhadap pemain, khususnya pada fase pertengahan dan akhir pertandingan yang tidak memiliki pola gerakan yang signifikan untuk dipelajari oleh model, serta sangat sensitif terhadap posisi dan cara berpikir pemain.

Tabel 3 menunjukkan bahwa sebuah rangkaian gerakan dapat memiliki hingga lima label yang berbeda, menunjukkan data yang ambigu atau inkonsisten. Menurut (Metzner et al., 2022), klasifikasi yang dilakukan terhadap data yang ambigu atau inkonsisten memiliki batas akurasi yang dapat dicapai, yang berlaku untuk semua jenis model klasifikasi dengan mekanisme yang berbeda-beda, seluruhnya tidak dapat mencapai kinerja yang lebih baik dari batas tersebut.

Tabel 3. Contoh Inkonsistensi Data

no.	fitur	label
1.	["e4", "e5", "nc6", "d4", "ed4"]	Nd4
2.	["e4", "e5", "nc6", "d4", "ed4"]	Bc4
3.	["e4", "e5", "nc6", "d4", "ed4"]	c3
4.	["e4", "e5", "nc6", "d4", "ed4"]	Bb5
5.	["e4", "e5", "nc6", "d4", "ed4"]	Ng5

Deeplefish juga seolah bermain dengan akurasi yang lebih tinggi saat bermain melawan mesin catur yang lebih kuat. Saat melawan Komodo level 1 yang memiliki estimasi *rating* sebesar 100, Deeplefish mendapatkan rata-rata akurasi sebesar 45%, sedangkan saat melawan Stockfish 16 yang memiliki estimasi *rating* sebesar 3000, Deeplefish mendapatkan rata-rata akurasi sebesar 80,2%. Akurasi yang seolah meningkat ini disebabkan karena lawan yang kuat lebih efektif dalam memanfaatkan kesalahan yang dibuat oleh Deeplefish, sehingga pertandingan dapat diselesaikan sebelum akurasi Deeplefish mengalami penurunan yang signifikan.

*Hyperparameter tuning* yang dilakukan menunjukkan pentingnya jumlah data yang digunakan. *Hyperparameter tuning* pada awalnya dilakukan menggunakan sampel data sebanyak 200.000 baris data, kemudian dilakukan kembali menggunakan 400.000 baris data. Hasil menunjukkan bahwa peningkatan jumlah data tersebut berhasil menurunkan *loss* data latih dari angka 3,4 menjadi 3,0, dan menambah jumlah kemenangan dari 0 menjadi 2.

## 5. KESIMPULAN

Penelitian ini melatih sebuah model LSTM dengan satu lapisan *embedding*, satu lapisan LSTM, dan satu lapisan *dense*, untuk memprediksi gerakan selanjutnya oleh putih dari sebuah posisi yang diberikan. Pra-pemrosesan data yang dilakukan melibatkan pembuatan kamus gerakan, *encoding* data menjadi numerik, pembuatan *N-Grams*, pemberian *pre-padding*, dan pemisahan label dari data. Konfigurasi *hyperparameter* yang digunakan adalah *batch size* senilai 250, *hidden size* senilai 128, dan *embedding dimension* senilai 128, yang dipilih berdasarkan hasil *hyperparameter tuning* menggunakan metode *Grid Search*. Pengujian dilakukan melawan mesin catur Komodo Level 1, Maia Chess 1, dan Stockfish 16, menunjukkan hasil yang kurang memuaskan dikarenakan penggunaan data yang kurang konsisten.

## 6. SARAN

Pengambilan dan pengolahan data yang dilakukan pada penelitian ini gagal dalam mempertimbangkan inkonsistensi serta perbedaan cara berpikir manusia saat bermain catur, sehingga model LSTM tidak dapat mempelajari data dengan baik, khususnya pada fase pertengahan dan fase akhir pertandingan yang tidak mengandung pola gerakan yang signifikan untuk dipelajari. Pengoptimalan kode program untuk memangkas waktu eksekusi pelatihan

model dapat membuka peluang untuk menggunakan lebih banyak data, ditambah dengan penggunaan data yang lebih konsisten, seperti mengambil data dari pertandingan yang dimainkan oleh Stockfish atau mesin catur lainnya, berpotensi untuk dapat membawa hasil yang lebih baik untuk penelitian-penelitian mendatang.

## DAFTAR PUSTAKA

- ANON. 2024. *Lichess accuracy metric*. [online] Lichess.org. Available at: <<https://lichess.org/page/accuracy>> [Accessed 20 June 2024].
- DAVID, O.E., NETANYAHU, N.S. and WOLF, L., 2016. DeepChess: End-to-end deep neural network for automatic learning in chess. In: *Artificial Neural Networks and Machine Learning – ICANN 2016*. Cham: Springer International Publishing. pp.88–96.
- KLEIN, D., 2022. *Neural Networks for Chess*. *arXiv [cs.LG]*. <https://doi.org/10.48550/ARXIV.2209.01506>.
- MAHARAJ, S., POLSON, N. and TURK, A., 2022. Chess AI: Competing paradigms for machine intelligence. *Entropy (Basel, Switzerland)*, [online] 24(4), p.550. <https://doi.org/10.3390/e24040550>.
- MEHTA, F., RAIPURE, H., SHIRSAT, S., BHATNAGAR, S. and BHOVI, B., 2020. Predicting chess moves with multilayer perceptron and limited lookahead. *Journal of Engineering Research and Application*, 10(4), pp.05-08. doi: 10.9790/9622-10040105088
- METZNER, C., SCHILLING, A., TRAXDORF, M., TZIRIDIS, K., SCHULZE, H. and KRAUSS, P., 2022. *Classification at the accuracy limit -- facing the problem of data ambiguity*. *arXiv [cs.LG]*. <https://doi.org/10.48550/ARXIV.2206.01922>.
- PANCHAL, H., MISHRA, S. and SHRIVASTAVA, V., 2021. Chess moves prediction using deep learning neural networks. In: *2021 International Conference on Advances in Computing and Communications (ICACC)*. IEEE.
- SANTOSA, R.D.W., BIJAKSANA, M.A. and ROMADHONY, A., 2021. Implementasi algoritma long short-term memory (LSTM) untuk mendeteksi penggunaan kalimat abusive pada teks bahasa Indonesia. *eProceedings of Engineering*, 8(1).
- SELLE, N., YUDISTIRA, N. and DEWI, C., 2022. Perbandingan Prediksi Penggunaan Listrik dengan Menggunakan Metode Long Short Term Memory (LSTM) dan Recurrent

Neural Network (RNN). *Jurnal teknologi informasi dan ilmu komputer*, [online] 9(1), p.155.  
<https://doi.org/10.25126/jtiik.2022915585>.

SHANNON, C.E., 1950. XXII. Programming a computer for playing chess. *The London*

*Edinburgh and Dublin Philosophical Magazine and Journal of Science*, [online] 41(314), pp.256–275.  
<https://doi.org/10.1080/14786445008521796>.