**PREDIKSI PELABELAN TITIK *BUY*, *SELL*, DAN *HOLD* PADA DATA SAHAM NVDA DENGAN MODEL *CONVOLUTIONAL NEURAL NETWORK* (CNN)**

Hanny Margaretha Aritonang1, Habiburrohman2

Program Studi Matematika, FMIPA Institut Teknologi Bandung

110121008@mahasiswa.itb.ac.id

210121089@mahasiswa.itb.ac.id

***Abstrak***

Pada beberapa tahun terakhir, investasi saham menjadi bagian penting dalam sistem perdagangan global. Salah satu tantangan terbesar dalam bidang ini adalah memprediksi harga saham karena dipengaruhi oleh berbagai faktor yang kompleks dan fluktuatif. Penelitian ini bertujuan untuk memprediksi pelabelan titik *buy*, *sell*, dan *hold* pada saham NVIDIA Corporation (NVDA) dengan model *Convolutional Neural Network* (CNN). Digunakan harga penutupan saham NVDA sejak 1 November 1999 hingga 7 Juni 2024 yang terdiri dari 6190 baris data. Sebelumnya, dilakukan pelabelan data dengan teknik *sliding window*. Kemudian, data 19 indikator teknis saham dengan 19 variasi interval yang telah dihitung dari harga penutupan akan dikonversi menjadi gambar satu *channel* dua dimensi berukuran 19x19. Setelah itu, gambar beserta label dibagi menjadi tiga segmen, yakni data latih, data validasi, dan data uji. Lalu, dibangun model CNN yang terdiri dari tujuh lapisan, meliputi satu *Convolutional Layer*, satu *Pooling Layer*, satu *Flatten Layer*, dua *Dropout Layer*, dan dua *Dense Layer*. Dengan *optimizer* berupa *Adaptive Moment Estimation* (Adam), model dilatih selama 50 *epoch* dan kinerjanya dievaluasi melalui matriks konfusi serta metrik *accuracy*, *precision, recall,* dan *f1-score*. Model CNN yang telah dibuat mencapai akurasi untuk data latih, data validasi, dan data uji berturut-turut sebesar 55,67%, 57,42%, dan 54.36%. Untuk kelas *buy*, *sell*, dan *hold* berturut-turut, nilai *precision* berkisar antara 20-23%, 18-19%, dan 97-99%. Sementara itu, nilai *recall* berkisar antara 93-97%, 87-97%, dan 50-53%. Adapun nilai *f1-score* berkisar antara 33-37%, 30-32%, dan 66-69%. Dengan demikian, model memiliki presisi yang tinggi untuk memprediksi kelas *hold* dan sensitivitas yang tinggi untuk memprediksi kelas *buy* dan *sell*. Namun, sebagai *trade-off*, model memiliki sensitivitas yang rendah untuk memprediksi kelas *hold* dan presisi yang rendah untuk memprediksi kelas *buy* dan *sell*. Oleh karena itu, masih diperlukan pengembangan lebih lanjut untuk meningkatkan performa model sehingga dapat digunakan sebagai rekomendasi aksi investasi saham NVDA dengan lebih baik.

# DAFTAR ISI

[DAFTAR ISI 2](#_Toc169647231)

[BAB I 3](#_Toc169647232)

[1.1 Latar Belakang 3](#_Toc169647233)

[1.2 Rumusan Masalah 4](#_Toc169647234)

[1.3 Tujuan Penelitian 4](#_Toc169647235)

[BAB II 6](#_Toc169647236)

[2.1 Pengumpulan dan Persiapan Data 6](#_Toc169647237)

[2.2 Perhitungan Indikator Teknis 6](#_Toc169647238)

[2.3 Pelabelan Data 10](#_Toc169647239)

[2.4 Pembentukan Gambar 12](#_Toc169647240)

[2.5 Pembuatan Model CNN 13](#_Toc169647241)

[BAB III 17](#_Toc169647242)

[3.1 Hasil 17](#_Toc169647243)

[3.2 Diskusi 19](#_Toc169647244)

[BAB IV 22](#_Toc169647245)

[4.1 Kesimpulan 22](#_Toc169647246)

[4.2 Saran 22](#_Toc169647247)

[LEMBAR KONTRIBUSI 24](#_Toc169647248)

[LAMPIRAN 25](#_Toc169647249)

[REFERENSI 26](#_Toc169647250)

# BAB I

**PENDAHULUAN**

## Latar Belakang

Dalam beberapa tahun terakhir, peramalam pasar saham berbasis kecerdasan komputasional telah menjadi bagian penting dari sistem perdagangan saham global. Inovasi ini telah berkembang pesat dengan diperkenalkannya berbagai pembangkangan anggaran seperti ETF, alternatif, dan sistem penggunaan (seperti valuta asing) yang saat ini dapat diakses oleh investor dan pedagang individu. Akibatnya, pertukaran kerangka kerja yang didasarkan pada model pengambilan keputusan yang independen dan cerdas semakin mendapat perhatian di berbagai pasar anggaran di seluruh dunia.

Model perkiraan dan klasifikasi berbasis pembelajaran mendalam telah mulai berkembang untuk mencapai eksekusi terbaik dalam berbagai aplikasi, termasuk penentuan harga saham. Model pembelajaran mendalam, seperti Convolutional Neural Organize (CNN), sistem saraf berulang (RNN), dan memori jangka panjang (LSTM), tampaknya mengalahkan strategi wawasan komputasi klasik seperti Support Vector Machine (SVM) dalam banyak kasus. Meskipun pada awalnya lebih dikenal dalam penanganan gambar dan aplikasi visi komputer, penerapan CNN kini mulai terlihat terjamin dalam pertimbangan anggaran juga.

Memperkirakan biaya saham adalah salah satu tantangan terbesar dalam bidang dana dan spekulasi. Harga saham sangat dipengaruhi oleh berbagai faktor, termasuk kondisi keuangan global, kebijakan pemerintah, berita perusahaan, dan sentiment pasar. Kompleksitas dan ketidakstabilan ini membuat ekspektasi harga saham menjadi tugas yang sulit dan menantang. Biasanya, strategi investigasi khusus dan penting digunakan untuk memperkirakan perkembangan harga saham. Namun, dengan munculnya *big data* dan peningkatan kemampuan komputasi, strategi wawasan komputasi, khususnya pembelajaran mendalam, telah menjadi instrumen yang lebih efektif dan efisien. Salah satu prosedur yang mencolok dalam pembelajaran mendalam adalah Convolutional Neural Organize (CNN).

Salah satu pendekatan inovasi dalam penelitian ini adalah mengubah informasi pengaturan waktu terkait uang satu dimensi menjadi gambar dua dimensi untuk memanfaatkan CNN dalam mengantisipasi harga saham. Mampu mengantisipasi harga saham secara akurat mempunyai dampak besar bagi investor. Dengan ekspektasi yang lebih tepat, pakar keuangan dapat membuat pilihan yang lebih tepat, memaksimalkan manfaat, dan meminimalkan risiko. Selain itu, sistem perdagangan terkomputerisasi berdasarkan perkiraan harga saham yang tepat dapat bekerja lebih efektif dan efisien, meningkatkan likuiditas pasar dan mengurangi ketidakstabilan.

Dalam laporan ini, kami akan menyelidiki penggunaan model CNN untuk memperkirakan harga saham NVIDIA Organization (NVDA) pada fokus *buy, sell*, dan *hold*. Kami akan menggunakan informasi harga saham NVDA yang dapat diverifikasi mulai 1 November 1999 hingga 7 Juni 2024, yang mencakup total 6090 informasi. Informasi ini memberikan gambaran luas dan mendalam mengenai perilaku biaya saham NVDA selama lebih dari dua dekade. Pendekatan ini akan mencakup beberapa langkah penting, menghitung pengumpulan informasi dan pra-pemrosesan, mempersiapkan acara CNN, serta menilai eksekusi model berdasarkan ketepatan perkiraan dan manfaat dari teknik pertukaran yang akan datang.

Tujuan dari laporan ini adalah untuk melihat kemampuan CNN dalam meramalkan perkembangan harga saham NVDA pada fokus *buy, sell,* dan *hold*, serta mengenali tempat menarik dan keterbatasan acara ini dalam latar pameran saham yang kompleks dan energik. Hasil penelitian ini diharapkan dapat memberikan pengalaman yang bermanfaat dan menjadi instrumen yang layak bagi para investor dalam mengambil keputusan perdagangan yang lebih baik.

## Rumusan Masalah

1. Apa langkah-langkah untuk mengubah data seri waktu harga saham satu dimensi menjadi gambar dua dimensi dengan menggunakan model CNN?
2. Seberapa akurat model CNN dalam memprediksi harga saham NVDA pada titik *buy, sell,* dan *hold*?
3. Faktor-faktor apa saja yang memengaruhi kinerja model CNN dalam prediksi harga saham?
4. Apakah model CNN dapat digunakan untuk mengembangkan strategi perdagangan yang menguntungkan dan dapat diterapkan di pasar saham nyata?
5. Apa keunggulan dan keterbatasan penggunaan model CNN dalam memprediksi harga saham NVDA?

## Tujuan Penelitian

1. Mengidentifikasi dan mengimplementasikan langkah-langkah transformasi data seri waktu menjadi gambar dua dimensi dengan menggunakan model CNN
2. Mengevaluasi keakuratan model CNN dalam memprediksi harga saham NVDA
3. Menganalisis factor-faktor yang memengaruhi kinerja model CNN dalam prediksi harga saham
4. Mengembangkan dan mengevaluasi strategi perdagangan berdasarkan prediksi model CNN untuk menentukan keuntungan dan aplikasinya di pasar saham nyata
5. Mengidentifikasi keunggulan dan keterbatasan model CNN dalam memprediksi harga saham NVDA

# BAB II

**METODOLOGI**

## Pengumpulan dan Persiapan Data

Data yang digunakan dalam penelitian ini yakni data historis harian saham NVIDIA Corporation (NVDA) dari tanggal 1 November 1999 hingga 7 Juni 2024. Data ini memiliki 6190 baris yang menunjukkan hari dan 6 fitur dengan rincian sebagai berikut.

Tabel 2.1.1 Deskripsi Data

|  |  |  |
| --- | --- | --- |
| No. | Fitur | Deskripsi |
| 1 | timestamp | Tanggal |
| 2 | open | Harga pembukaan saham, yakni saat bursa buka pada hari terkait. Merupakan gambaran awal sentimen pasar. |
| 3 | high | Harga tertinggi pada hari terkait. Penting untuk mengidentifikasi potensi pembalikan harga |
| 4 | low | Harga terendah pada hari terkait. |
| 5 | close | Harga penutupan saham, yakni saat bursa tutup pada hari terkait. Dianggap paling penting karena mencerminkan konsesus terakhir pada hari tersebut |
| 6 | volume | Jumlah lembar saham yang diperdagangkan pada hari terkait. Mengindikasikan besar minat terhadap saham, dapat menandakan keberlanjutan atau pembalikan tren. |

Kombinasi dari keenam fitur ini memungkinkan model untuk menangkap berbagai aspek dari dinamika pasar saham, mulai dari perubahan harga harian hingga volume pasar saham. Dengan mengolah fitur-fitur ini, model CNN dapat mengenali pola yang lebih kompleks dan memberikan perkiraan yang lebih tepat sehubungan dengan titik-titik *buy, sell,* dan *hold* saham. Sebelum digunakan, dilakukan normalisasi untuk menjamin bahwa perbedaan skala data tidak memengaruhi kinerja model. Selanjutnya, informasi yang disiapkan ini digunakan sebagai inputan untuk arsitektur CNN yang dirancang untuk menangkap pola spasial dalam data saham. Dengan pendekatan ini, model dapat membuat prediksi yang informatif dan efektif.

## Perhitungan Indikator Teknis

Dalam penelitian ini, sebanyak 19 indikator teknis dengan 19 interval waktu berbeda, yakni 3 hingga 21 hari, akan dihitung untuk memprediksi titik *buy*, *sell*, dan *hold* saham NVDA. Perlu diperhatikan bahwa akan terdapat baris data yang tidak memiliki nilai indikator teknis karena belum memenuhi interval waktu yang diperlukan untuk perhitungan. Indikator yang digunakan terdiri dari 10 indikator *Moving Average* (SMA, EMA, TMA, LWMA, WilderMA, KAMA, GMA, HMA, WMA, TEMA), 7 indikator momentum (RSI, WILLR, TRIX, ROC, PPO, CCI, CMF), 1 indikator volatilitas (ATR), dan 1 indikator statistik (STD). Berikut ini penjelasan setiap indikator.

1. Simple Moving Average (SMA)

SMA menunjukkan rata-rata bergerak dari harga-harga untuk periode tertentu. SMA dengan nilai interval yang berbeda digunakan untuk menentukan arah tren. Beberapa SMA dapat digabungkan untuk digunakan bersama-sama. Jika harga saham lebih tinggi dari SMA diasumsikan bahwa saham berada dalam tren naik artinya harga saham akan terus meningkat (sinyal beli), sedangkan jika harga saham lebih renddah dari SMA, diasumsikan bahwa saham berada dalam tren turun, menunjukkan harga saham akan turun (sinyal jual). Perhitungan SMA ditunjukkan oleh persamaan berikut ini :

1. Exponential Moving Average (EMA)

EMA adalah jenis indikator yang menunjukkan rata-rata bergerak dari harga dengan memberikan lebih banyak penekanan pada har-hari terbaru. EMA hampir mirip dengan SMA tetapi memberikan bobot lebih pada harga terbaru. Persamaan perhitungan EMA dari harga saham :

1. Triangular Moving Average (TMA)

TMA adalah rata-rata dua kali dari serangkaian harga yang memberikan penghalusan lebih besar. Kegunaan TMA adalah menyediakan sinyal yang lebih halus dengan lebih sedikit noise dibandingkan dengan SMA dan EMA.

1. Linear Weighted Moving Average (LWMA)

LWMA memberikan bobot yang lebih besar pada harga terbaru dengan bobot yang berkurang secara linear untuk harga sebelumnya. LWMA digunakan untuk mengutamakan harga terbaru tetapi dengan metode pembobotan yang berbeda dari EMA.

1. Wilder Moving Average (WilderMA)

WilderMA digunakan dalam perhitungan indikator RSI dan ATR untukm menyediakan penghalusan yang lebih konstan dan stabil.

1. Kaufman’s Adaptive Moving Average (KAMA)

KAMA mempertahankan kehalusan berdasarkan volatilitas atau efisiensu harga. Artinya, KAMA lebih halus selama volatilitas rendah dan lebih responsif selama volatilitas tinggi.

1. Geometric Moving Average (GMA)

GMA adalah jenis rata-rata bergerak menggunakan rata-rata geometris dari harga selama periode waktu tertentu. GMA lebih fokus pada pertumbuhan atau penurunan eksponensial dalam data harga.

1. Average True Range (ATR)

ATR mengukur volatilitas pasar dengan menghitung rata-rata rentang harga sejati untuk periode tertentu. ATR sering digunakan untuk menetapkan stop loss dan target harga.

1. Relative Strength Index (RSI)

RSI adalah indikator analisis teknis tipe osilator yang menunjukkan kekuatan dan kelemahan historis harga saham. Saat harga saham berubah, nilai RSI berfluktuasi antara 0 dan 100 yang menunjukkan apakah harga saham berada di wilayah *overbought* atau *oversold*. Persamaan perhitungan RSI :

1. William’s R (WillR)

Williams %R adalah indikator teknis berbasis momentum yang juga menentukan kondisi *overbought* dan *oversold* untuk harga saham. Persamaan Perhitungan WILLR.

1. Triple Exponential Average (TRIX)

TRIX adalah indikator momentum yang menunjukkan tingkat perubahan dari EMA tiga kali harga. TRIX memberikan sinyal beli atau jual berdasarkan persilangan garis Trix dan garis nol.

1. Standard Deviation (STD)

STD mengukur seberapa tersebar harga dari rata-rata. STD digunakan untuk mengukur volatilitas pasar.

1. Rate of Change (ROC)

ROC adalag indikator yang mengukur persentase perubahan harga selama periode waktu tertentu. Persamaan perhitungan ROC :

1. Percentage Price Oscillator (PPO)

PPO adalah indikator momentum yang mengukur perbedaan antara dua EMA dalam persenrase dan memberikan sinyal beli atau jual berdasarkan persilangan garis PPO dan garis sinyal. Persaaman perhitungan PPO :

1. Hull Moving Average (HMA)

HMA adalah jenis indikator yang mengurangi lag yang terkait dengan SMA, EMA, dan WMA. Persamaan perhitungan HMA :

1. Weighted Moving Average (WMA)

WMA aaddalah indikator yang sama dengan EMA perbadaannya adalah pentingnya harga penutupan secara linear menurun saat bergerak kembali ke masa lalu. WMA dihitung dengan

1. Triple Exponential Moving Average (TEMA)

TEMA adalah indikator EMA yang menyediakan pengurangan fluktuasi harga minor dan menyaring volatilitas. Perhitungan TEMA sebagai berikut :

1. Commodity Channel Index (CCI)

CCI adalah indikator yang membandingkan harga saat ini dengan rata-rata harga selama periode waktu tertentu. Persamaan CCI adalah sebagai berikut :

1. Chaikin Money Flow (CMF)

CMF adalah indikator yang digunakan untuk mengukur volume aliran uang selama periode waktu tertentu. . Nilai indikator berfluktuasi antara 1 dan -1. Jika nilainya lebih dekat ke 1, diinterpretasikan bahwa tekanan pembelian lebih tinggi. Sebaliknya, jika nilainya lebih dekat ke -1, diinterpretasikan bahwa tekanan penjualan lebih tinggi. Persamaan perhitungan CMF :

## Pelabelan Data

Setiap baris data akan dilabeli dengan salah satu dari *buy*, *sell*, atau *hold* berdasarkan nilai fitur *close*, yakni harga penutupan saham. Prosedur pelabelan dilakukan dengan cara membuat *sliding window* (jendela geser) yang memuat 11 baris data. Jendela ini akan digeser satu demi satu baris. Pada setiap iterasi, baris data yang merupakan titik tengah dari jendela akan dilabeli. Jika harga penutupan bernilai minimum (relatif terhadap sepuluh nilai lain pada jendela terkait), maka dilabeli dengan *buy*. Sebaliknya, jika bernilai maksimum, maka akan dilabeli dengan *sell*. Adapun jika bukan keduanya, maka akan dilabeli dengan *hold*. Berikut ini algoritma pelabelan yang digunakan.

Tabel 2.3.1 Algoritma Pelabelan Data

|  |
| --- |
| procedure Labeling()  windowSize = 11 days  while(counterRow < numberOfDaysInFile)  counterRow ++  If (counterRow > windowSize)  windowBeginIndex = counterRow – windowSize  windowEndIndex = windowBeginIndex + windowSize – 1  windowMiddleIndex = (windowBeginIndex + windowEndIndex)/2  for (i = windowBeginIndex;i <= windowEndIndex;i ++)  number = closePriceList. get(i)  if(number < min)  min = number  minIndex = closePriceList. indexOf(min)  if(number > max)  max = number  maxIndex = closePriceList. indexOf(max)  if(maxIndex == windowMiddleIndex)  result=”SELL”  elif(minIndex == windowMiddleIndex)  result=”BUY”  else  result=”HOLD” |

Sampel hasil pelabelan dapat dilihat pada grafik berikut. Perlu diperhatikan bahwa titik data yang bukan berlabel *buy* maupun *sell* berarti memiliki label *hold*.

Sebuah gambar berisi diagram, garis, Plot, teks

Deskripsi dibuat secara otomatis

Gambar 2.3.1 Hasil Pelabelan Data dari 28 Oktober 2019 hingga 31 Mei 2024

Berikut ini frekuensi masing-masing label.

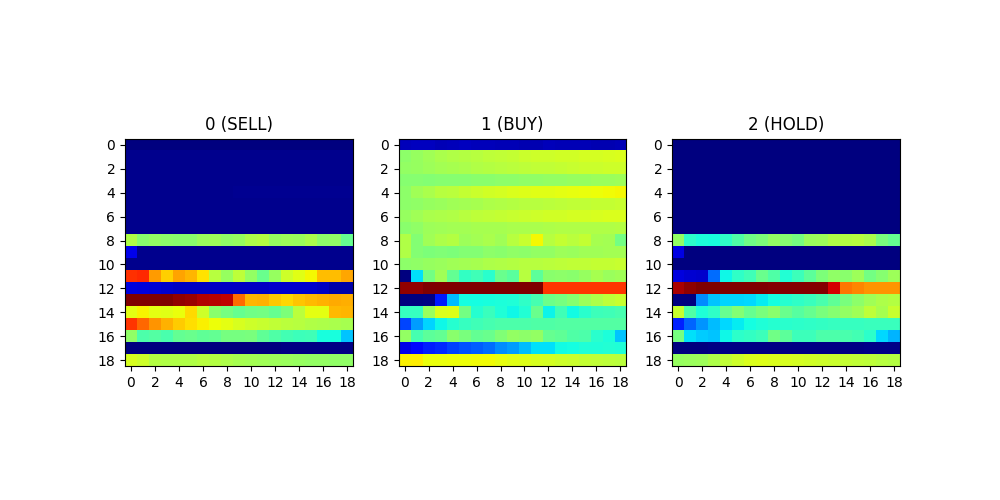
Tabel 2.3.2 Frekuensi Label Data

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Label** | **Frekuensi** | **Persentase** |
| 1 | Buy | 360 | 5,85% |
| 2 | Hold | 5432 | 88,24% |
| 3 | Sell | 364 | 5,91% |
| Total | | 6156 | 100% |

Perlu diperhatikan bahwa lima baris data pertama dan terakhir tidak akan mendapat label karena jendela geser berukuran 11. Setelah tahap ini, baris data yang tidak memiliki label atau fiturnya tidak lengkap akan dihapus dan tidak diikutsertakan dalam proses selanjutnya.

## Pembentukan Gambar

Sebanyak 19 indikator teknis saham yang masing-masing terdiri dari 19 interval berbeda akan menjadi 381 fitur baru pada data. Selanjutnya, untuk setiap baris, akan dibentuk gambar dua dimensi berukuran 19x19 yang berasal dari fitur indikator teknis tersebut. Posisi indikator teknis akan diurutkan berdasarkan jenisnya dengan tujuan agar menghasilkan gambar yang konsisten dan bermakna sehingga model dapat menangkap pola dengan lebih baik. Gambar ini akan memiliki label *buy*, *sell*, atau *hold* sesuai dengan hasil pelabelan sebelumnya. Berikut ini sampel gambar yang dihasilkan beserta labelnya yang divisualisasikan dengan *colormap* jet pada matplotlib agar lebih menarik. Perlu diperhatikan bahwa warna gambar yang sebenarnya hanya berupa skala hitam putih dan sebelum visualisasi, rentang data sudah diskalakan dari 0 hingga 1. Proses penskalaan ini akan dibahas lebih lanjut pada bagian berikutnya.



Gambar 2.4.1 Sampel Gambar dari Fitur Indikator Teknis

## Pembuatan Model CNN

Setelah tahap pembuatan gambar, akan dibangun model CNN untuk mengklasifikasikan data gambar menjadi salah satu dari tiga kelas, yakni *buy*, *sell*, dan *hold*. Sebelumnya, data akan dibagi menjadi tiga, yakni data latih sebesar 56%, data validasi sebesar 14%, dan data uji sebesar 30%. Pembagian ini dilakukan dengan fungsi *train\_test\_split* dari library sklearn dengan menyertakan parameter *stratified* sehingga proporsi label pada ketiga bagian data akan sama. Kemudian, dilakukan penskalaan pada data dengan *MinMaxScaler* untuk memastikan setiap fitur memiliki skala yang sama sehingga tidak ada fitur yang dianggap oleh model lebih penting daripada yang lain. Pada penelitian ini, rentang setiap fitur diskalakan dari 0 hingga 1. Berikut ini adalah proporsi pembagian data.

Tabel 2.5.1 Proporsi Pembagian Data Latih, Validasi, dan Uji

|  |  |  |  |
| --- | --- | --- | --- |
| No | Data | Ukuran | Proporsi |
| 1 | Latih | 3447 | 56% |
| 2 | Validasi | 862 | 14% |
| 3 | Uji | 1847 | 30% |
| Total | | 6156 | 100% |

Perlu diperhatikan bahwa berdasarkan tabel 2.3.2, frekuensi label atau kelas *hold* jauh lebih banyak dibandingkan dengan kelas *buy* dan *sell*. Masalah ini disebut dengan *class* *imbalance*, yakni proporsi kelas yang tidak seimbang. Hal ini dapat mengakibatkan model lebih sering memprediksi kelas *hold* daripada dua kelas lain karena proporsinya lebih besar. Oleh karena itu, akan digunakan pembobotan sampel untuk mengatasi masalah tersebut. Konsep dari pembobotan sampel adalah dengan memberikan bobot yang berbeda untuk masing-masing kelas, bergantung pada proporsi kelas terkait. Bobot ini dinamakan dengan *sample weight*. Lalu, dilakukan pengkodean terhadap ketiga kelas dengan *OneHotEncoder*.

Selanjutnya, dibangun model CNN dua dimensi dengan arsitektur sebagai berikut.

1. *Convolutional Layer* dengan parameter jumlah filter 64, ukuran kernel 3x3, *strides* 1x1, padding ‘*same*’, fungsi aktivasi ReLU, dan ukuran *input* 19x19x1
2. *Average Pooling Layer* dengan parameter ukuran *pool* 3x3, *strides* 1x1, dan padding ‘*valid*’
3. *Dropout Layer* dengan parameter *rate* 0,1
4. *Flatten Layer*
5. *Dense Layer* dengan paramater *units* 1024 dan fungsi aktivasi ReLU
6. *Dropout Layer* dengan parameter *rate* 0,3
7. *Dense Layer* dengan paramater *units* 3 dan fungsi aktivasi *softmax*

Pertama, *Convolutional Layer* berfungsi untuk mengekstrak fitur penting dari gambar. Kedua, *Average Pooling Layer* berfungsi untuk mengurangi dimensi gambar sekaligus mempertahankan informasi dengan mengambil nilai rataan. Ketiga, *Dropout Layer* yang pertama, berfungsi untuk mencegah *overfitting* dengan cara menonaktifkan 10% neuron secara acak. Keempat, *Flatten Layer* berfungsi untuk mengubah *output* dari *layer* sebelumnya menjadi vektor satu dimensi, yakni menjadi penghubung *Convolutional Layer* atau *Pooling Layer* dengan *Dense Layer*. Kelima, *Dense Layer* yang pertama, berfungsi untuk menangkap pola yang kompleks pada data. Keenam, *Dropout Layer* yang kedua, berfungsi untuk mencegah *overfitting* dengan cara menonaktifkan 30% neuron secara acak. Ketujuh, *Dense Layer* yang kedua, berfungsi untuk mengklasifikasikan data menjadi tiga kelas (*buy*, *sell*, dan *hold*).

Arsitektur model di atas dapat dirangkum pada gambar berikut.

A diagram of a computer

Description automatically generatedGambar 2.5.1 Arsitektur Model CNN

Selanjutnya, model dilatih dengan data latih. Pada pelatihan model, dipilih *optimizer* Adam dengan parameter *learning\_*rate 0.001. Selain itu, dipilih *loss function* berupa *categorical cross-entropy* karena merupakan kasus klasifikasi multikelas dan kelas dikodekan secara *one-hot*. Digunakan *batch\_size* sebesar 64 dengan mengaktifkan mode *shuffle*. *Epoch* diatur sebanyak 50 kali. Ditambahkan *callbacks* berupa *ReduceLROnPlateau* untuk mengurangi parameter *learning\_rate* menjadi seperlimanya jika *loss function* pada data validasi tidak turun sebesar 0.001 selama 5 epoch berturut-turut, dengan batas minimum *learning\_rate* sebesar 0,0001. Model yang sudah selesai dilatih akan diuji dengan data uji.

# BAB III

**HASIL DAN DISKUSI**

## Hasil

Setelah tahap pelatihan model dengan data latih melalui 50 *epoch*, diperoleh akurasi model sebesar 55,67% untuk data latih dan sebesar 57,42% untuk data validasi. Selain itu, *loss* pada model bernilai 0,4548 untuk data latih dan 0,7380 untuk data validasi. Untuk mengevaluasi kinerja model, akan ditinjau matriks konfusi dan beberapa metrik seperti *precision*, *recall*, dan *f1-score*.

Matriks konfusi untuk data latih dapat dituangkan pada tabel berikut.

Tabel 3.1.1 Matriks Konfusi untuk Data Latih

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Data Latih | | Prediksi Kelas | | | Total |
| Sell (0) | Buy (1) | Hold (2) |
| Kelas Aktual | Sell (0) | 198 | 0 | 6 | 204 |
| Buy (1) | 0 | 195 | 7 | 202 |
| Hold (2) | 830 | 700 | 1511 | 3041 |
| Total | | 1028 | 895 | 1524 | 3447 |

Sementara itu, metrik evaluasi untuk data latih dapat dilihat pada tabel berikut.

Tabel 3.1.2 Metrik Evaluasi untuk Data Latih

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Data Latih | | Metrik | | |
| Precision | Recall | F1-Score |
| Kelas | Sell (0) | 0,19 | 0,97 | 0,32 |
| Buy (1) | 0,22 | 0,97 | 0,36 |
| Hold (2) | 0,99 | 0,50 | 0,66 |
| Macro Average | | 0.47 | 0.81 | 0.45 |
| Weighted Average | | 0.90 | 0.55 | 0.62 |

Selanjutnya, matriks konfusi untuk data validasi dapat dituangkan pada tabel berikut.

Tabel 3.1.3 Matriks Konfusi untuk Data Validasi

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Data Validasi | | Prediksi Kelas | | | Total |
| Sell (0) | Buy (1) | Hold (2) |
| Kelas Aktual | Sell (0) | 47 | 0 | 4 | 51 |
| Buy (1) | 0 | 48 | 2 | 50 |
| Hold (2) | 201 | 160 | 400 | 761 |
| Total | | 248 | 208 | 406 | 862 |

Sementara itu, metriks evaluasi untuk data validasi dapat dilihat pada tabel berikut.

Tabel 3.1.4 Metrik Evaluasi untuk Data Validasi

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Data Validasi | | Metrik | | |
| Precision | Recall | F1-Score |
| Kelas | Sell (0) | 0,19 | 0,92 | 0,31 |
| Buy (1) | 0,23 | 0,96 | 0,37 |
| Hold (2) | 0,99 | 0,53 | 0,69 |
| Macro Average | | 0.47 | 0.80 | 0.46 |
| Weighted Average | | 0.89 | 0.57 | 0.65 |

Setelah dilakukan pengujian dengan data uji, diperoleh akurasi sebesar 54,36% dengan *loss* bernilai 0,7632. Adapun matriks konfusi untuk data uji dapat dituangkan pada tabel berikut.

Tabel 3.1.3 Matriks Konfusi untuk Data Uji

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Data Uji | | Prediksi Kelas | | | Total |
| Sell (0) | Buy (1) | Hold (2) |
| Kelas Aktual | Sell (0) | 95 | 0 | 14 | 109 |
| Buy (1) | 0 | 100 | 8 | 108 |
| Hold (2) | 424 | 397 | 809 | 1630 |
| Total | | 519 | 497 | 831 | 1847 |

Sementara itu, metriks evaluasi untuk data uji dapat dilihat pada tabel berikut.

Tabel 3.1.4 Metrik Evaluasi untuk Data Uji

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Data Uji | | Metrik | | |
| Precision | Recall | F1-Score |
| Kelas | Sell (0) | 0,18 | 0,87 | 0,30 |
| Buy (1) | 0,20 | 0,93 | 0,33 |
| Hold (2) | 0,97 | 0,50 | 0,66 |
| Macro Average | | 0.45 | 0.76 | 0.43 |
| Weighted Average | | 0.88 | 0.54 | 0.62 |

Terakhir, akan dilakukan prediksi. Berikut ini prediksi titik *buy* dan *sell* saham NVDA sejak 1 Januari 2017 hingga 31 Mei 2024 dari model yang telah dibuat.

A graph with red and green lines

Description automatically generated

Gambar 3.1.1 Hasil Prediksi Model

## Diskusi

Hasil pelatihan model CNN menunjukkan bahwaa model ini memiliki akurasi yang cukup baik untuk memprediksi harga saham NVDA.

1. Keakuratan Model

Model CNN menunjukkan akurasi 55,67% pada data latih dan 57,42% pada data validasi. Akurasi yang lebih tinggi pada data validasi menunjukkan bahwa model tidak mengalami overfitting yang signifikan tetapi masih ada ruang untuk meningkatkan akurasi prediksi.

1. Evaluasi metrik

Dari metrik evaluasi precision, recall, dan f1-score untuk kelas Hold lebih tinggi daripada kelas buy dan sell. Ini bisa disebabkan oleh distribusi data yang tidak seimbang, yaitu jumlah data untuk kelas Hold jauh lebih besar daripada kelas Sell dan Buy. Ini menyebabkan model lebih terfokus pada kelas Hold. Karena kelas Hold lebih mendominasi data, maka model CNN menunjukkan performa terbaik dalam memprediksi kelas Hold. Metrik *macro average* dan *weighted average* menunjukkan bahwa meskipun ada prediksi yang akurat, keseluruhan performa model masih perlu ditingkatkan untuk mencapai prediksi yang lebih andal.

1. Faktor-faktor yang mempengaruhi kinerja model

Kinerja CNN dipengaruhi oleh beberapa faktor diantaranya sebagai berikut.

1. Kualitas data. Data yang mengandung noise, missing value, atau anomaly dapat menurunkan kinerja model sehingga dibutuhkan preprocessing data sebelum melakukan training.
2. Distribusi data. Ketidakseimbangan jumlah data dalam kelas Sell, Buy, dan Hold menyebabkan model bias terhadap kelas yang dominan. Untuk membantu mengatasi masalah ini perlu dilakukan teknik pembobotan sampel.
3. Arsitektur model. Jumlah dan ukuran lapisan konvolusi, ukuran kernel, dan jumlah neuron mempengaruhi kinerja model dalam menangkap fitur dari data. Arsitektur yang terlalu sederhana tidak mampu untuk menangkap kompleksitas data sedangkan arsitektur yang terlalu kompleks bisa menyebabkan overfitting.
4. Hyperparameter tuning. Learning rate yang terlalu tinggi menyebabkan model melewati titik minimum fungsi loss namun dapat membuat proses training data menjadi sangat lambat. Ukuran batch dan jumlah epoch mempengaruhi stabilitas dan konvergensi proses training. Penentuan yang tepat dapat mempercepat training dan menghasilkan model yang baik.
5. Implementasi strategi perdagangan

Mengingat rendahnya angka precision pada titik-titik buy dan sell, implementasi strategi perdagangan berdasarkan prediksi ini cukup berisiko. Oleh karena itu, strategi perdagangan dari model ini bisa digunakan sebagai panduan tambahan dan bukan satu-satunya basis keputusan. Untuk meningkatkan efektivitas, model ini dapat dikombinasikan dengan metode lain.

1. Kelebihan dan Keterbatasan Model CNN

Kelebihan model CNN adalah mampu menangkap pola kompleks data harga saham NVDA dengan mengubah data seri waktu menjadi gambar dua dimensi yang memberikan perspektif baru yang tidak dapat diakses oleh model tradisional. Di sisi lain, keterbatasan model CNN yaitu rendahnya performa pada prediki titik-titik jual dan beli menunjukkan perlunya pengoptimalan lanjutan untuk model ini. Keterbatasan ini bisa diatasi dengan meningkatkan arsitektur model dan kualitas data.

# BAB IV

**KESIMPULAN DAN SARAN**

## Kesimpulan

Dari penelitian dan eksperimen yang telah dilakukan, diperoleh kesimpulan sebagai berikut.

1. Model CNN dapat digunakan untuk memprediksi titik *buy*, *sell*, dan *hold* saham NVDA meskipun akurasinya tidak terlalu besar.
2. Nilai metrik evaluasi yang hampir sama untuk data latih, data validasi, dan data uji menunjukkan bahwa model CNN yang dibuat tidak mengalami *overfitting*.
3. Model CNN yang dibuat dapat mendeteksi kelas buy dan sell dengan baik, dibuktikan dengan nilai recall yang tinggi untuk kedua kelas tersebut, yakni sekitar 87 hingga 97 persen. Namun, sebagai *trade-off*, presisi kedua kelas tersebut menjadi rendah, yakni sekitar 18 hingga 23 persen.

## Saran

Dari hasil yang telah diperoleh, berikut ini beberapa saran untuk mengembangkan model lebih lanjut.

1. Memasukkan lebih banyak indikator teknis, kemudian melakukan seleksi fitur indikator teknis sebelum proses pembentukan gambar. Hal ini bertujuan agar data yang dikonversi menjadi gambar lebih bermakna dan berpengaruh tinggi terhadap label.
2. Menggunakan teknik lain untuk mengatasi masalah *class imbalance*, misalnya dengan melakukan *resampling* pada data.
3. Melakukan pembagian segmen data latih, data validasi, dan data uji dengan teknik *sliding window train test split*. Dengan melatih model secara bertahap dan berurutan dalam beberapa periode kemudian memprediksi periode berikutnya, diharapkan model dapat menangkap informasi deret waktu yang lebih komprehensif sehingga performa model meningkat.

# LEMBAR KONTRIBUSI

Tugas ini dikerjakan oleh dua orang yaitu Habib dan Hanny. Kedua penulis bekerja sama dengan baik dalam pengerjaan tugas ini. Habib mencari data saham NVDA kemudian membuat kode untuk *preprocessing* data dan model CNN.

Hanny juga menjalankan program yang telah dibuat habib, memperbaiki error yang muncul serta mencari sumber kode lain untuk memprediksi harga saham NVDA namun tidak sepenuhnya digunakan pada program akhir karena kurang efisien dan membutuhkan waktu yang lama untuk menjalankannya (lebih dari 3 jam). Selanjutnya, kedua program dikombinasikan.

Laporan dibuat bersama-sama oleh kedua penulis dengan rincian sebagai berikut.

1. Hanny membuat bagian abstrak, latar belakang, rumusan masalah, tujuan penelitian, pengumpulan dan persiapan data, indikator teknis, pelabelan data, dan diskusi.
2. Habib mengerjakan bagian abstrak, pengumpulan dan persiapan data, pelabelan data, pembentukan gambar, pembuatan model CNN, hasil, kesimpulan, saran, serta merapikan penulisan laporan.

# LAMPIRAN

Berikut ini link Google Drive yang berisi data dan kode hasil pengerjaan kami.

<https://drive.google.com/drive/folders/1xQ0lR2tCHmW0TkBK03u1hji5_Q-JvFGn?usp=sharing>

# REFERENSI

Johnson, Kevin. *Pandas TA - A Technical Analysis Library in Python 3*. https://github.com/twopirllc/pandas-ta

Kusdiantara, Rudy dan Islahuddin, Muhammad. *Lecture Notes on CNNs for Image Classification*. Bahan Kuliah MA4072 Pembelajaran Mendalam. Program Studi Matematika ITB. Bandung.

McGuire, Trevor. Python Library for Technical Analysis. https://github.com/trevormcguire/technical-analysis

Rizvi, M. S. Z. 2024. *CNN image classification: Image Classification using CNN*. Analytics Vidhya. Retrieved June 17, 2024, from <https://www.analyticsvidhya.com/blog/2020/02/learn-image-classification-cnn-convolutional-neural-networks-3-datasets/>

Sezer, Omer Berat. 2018.  *Algorithmic Financial Trading with Deep Convolutional Neural Networks: Time Series to Image Conversion Approach*. Retrieved June 17, 2024 from <https://github.com/omerbsezer/CNN-TA>