Journal of Applied Informatics and Computing (JAIC)
Vol.7, No.2, December 2023, pp. 186~191
e-ISSN: 2548-6861

Hyperparameter Tuning on Graph Neural Network for the Classification of SARS-CoV-2 Inhibitors

Salamet Nur Himawa [1]*, Robieth Sohiburoyyan [2], Iryanto [3]****
* Teknik Informatika, Politeknik Negeri Indramayu
** Rekayasa Perangkat Lunak, Politeknik Negeri Indramayu
snhimawan@polindra.ac.id [1], robieth.s@polindra.ac.id [2], iryanto@polindra.ac.id [3]

Article Info

Article history:

Received 2023-10-30
Revised 2023-11-10
Accepted 2023-11-13

Keyword:

Graph Neural Network,
Inhibitor,
Covid-19,
SARS-CoV-2.

ABSTRACT

COVID-19 is caused by the SARS-CoV-2 virus, which results in a range of symptoms, from mild to severe, and can lead to fatalities. As of October 2023, WHO has recorded 771 cases of COVID-19 globally. Various efforts have been made to control the spread of the virus, including vaccination, isolation measures, and intensive medical care. The emergence of new SARS-CoV-2 variants has led to the ongoing evolution of virus transmission. Continued research is essential to understand this virus and develop strategies to address the pandemic. Inhibitors of SARS-CoV-2 play a crucial role in the vaccine development process. Inhibitors can impede the virus's development, helping reduce disease severity and control the pandemic. The classification of inhibitors is expected to serve as a foundation for selecting compounds that can be developed into vaccines. This research develops a Graph Neural Network model for inhibitor classification and uses the random search method for hyperparameter tuning. Graph Neural Networks are chosen due to their excellent performance in modelling graph data. This study demonstrates the success of hyperparameter tuning in improving the performance of the Graph Neural Network for accurate classification of SARS-CoV-2 inhibitors.

I. PENDAHULUAN

World Health Organization menyampaikan bahwa sampai pada bulan oktober terdapat kasus COVID-19 sebanyak 771 juta secara global [1]. Corona Virus Disease 2019 atau yang biasa disingkat COVID-19 adalah penyakit menular yang disebabkan oleh SARS-CoV-2. Jenis virus SARS-CoV-2 masih terus berkembang dan memiliki jenis-jenis baru. Hingga saat ini belum ada pengobatan atau vaksin khusus untuk setiap virus SARS-CoV-2, dan perawatan yang tersedia sebagian besar hanya dapat membantu meredakan gejala dan meningkatkan hasil penyakit. Namun, mengembangkan obat antivirus yang efektif yang dapat secara khusus menargetkan virus SARS-CoV-2, yang menyebabkan COVID-19, sangat penting untuk mengendalikan pandemi dan mengurangi keparahan penyakit.

Penelitian tentang inhibitor COVID-19, yang merupakan obat atau senyawa yang dapat memblokir replikasi atau masuknya virus ke dalam sel inang, sangat penting. Inhibitor secara potensial dapat mengurangi beban virus pada individu yang terinfeksi, memperpendek durasi penyakit, dan mencegah penularan virus ke orang lain. Selain itu, inhibitor dapat digunakan dalam kombinasi dengan perawatan atau vaksin lain untuk meningkatkan efektivitas mereka dan mengurangi risiko resistensi obat.

Beberapa penelitian yang telah berkembang dalam identifikasi ihbitor, seperti yang telah dilakukan oleh Touret dan Gawrilijuk. Dalam penelitiannya Touret menunjukan hasil eksperimen beberapa obat seperti Azithromycine, Opipramol dan Quinidine menunjukan potensi untuk menjadi inhibtors SARS-CoV-2 [2]. Sementara Gawrilijuk telah memembangun model machine learning yang dapat mengidentifikasi senyawa yang merupakan inhibitor dar SARS-CoV-2 [3]. Penelitian lain dalam proses identifikasi inhibitor SARS-CoV-2 menujukan Model Geometric Deep

Learning memiliki potensi dalam identifikasi inhibitor SARS-CoV-2 [4].

Senyawa inhibitor dapat direpresentasikan sebagai graf, dimana graf merupakan data non-Euclidean. Broenstein mengembangkan deep learning tradisional untuk menangani data non-Euclidian yang selanjutnya dikenal sebagai GDL [5]. Geometric Deep Learning (GDL) merupakan teknik dengan performa yang baik dalam memodelkan data non-Euclidean. Salah satu model GDL adalah Graph Neural Network (GNN). GNN dirancang untuk memproses data graf dan memiliki kemampuan baik dalam memodelkan struktur dan hubungan kompleks pada graf [6].

Teknik konvolusi graf untuk menggambarkan molekul telah digunakan dalam *drug discovery* dan menunjukan hasil yang dapat mengungguli metode fingerprint-based [7]. Teknik GDL telah digunakan dalam prediksi struktur ikatan obat dan menujukan performa dan waktu komputasi yang baik [8]. Beberapa pengembangan dari GDL yang sering digunakan adalah Graph Attention Network (GAT) [9] dan Unified Massage Passing Model (UniMP) [10].

Mempelajari inhibitor COVID-19 merupakan area penelitian yang penting yang dapat memiliki implikasi yang signifikan bagi kesehatan masyarakat, praktik klinis, dan pengembangan obat. Penelitian ini bertujuan untuk meningkatkan performa Model GNN menggunakan *hyperparameter tuning* sehingga dapat mengklasifikasi inhibitor dengan baik. Dengan mengidentifikasi dan mengevaluasi inhibitor yang efektif terhadap SARS-CoV-2, diharapkan dapat memberikan kontribusi pada upaya global untuk mengendalikan dan pada akhirnya mengeliminasi pandemi COVID-19. Oleh karena itu penilitian ini akan mengembangkan Model GNN dengan *hyperparameter tuning* yang dapat mengidentifikasi senyawa yang merupakan inhibitor dari SARS-CoV-2.

II. METODE PENELITIAN

A. Pengumpulan Data

Data merupakan senyawa kimia dalam bentuk SMILE yang tersimpan pada FDA-Approve Drug Library [2]. Data terdiri dari kode SMILE senyawa dan kelas (potensi menjadi inhibitor). Data terdiri dari 1484 record, 88 data termasuk ke dalam kelas 1 dan 1396 termasuk ke dalam kelas 0. Berikut beberapa bagian dari data yang digunakan.

TABEL I
CONTOH DATA

No	SMILES	Class
1	CC(=C/C(=N/NC1=NN=CC2=CC=CC=C21)/C)C	1
2	C1=CC(=CC=C1N)S(=O)(=O)N	0
3	C1CC(CCC1NCC2=C(C(=CC(=C2)Br)Br)N)O.Cl	1
4	CCCCC(C1=CC=CC=C1)O	0

B. Pre-processing Data

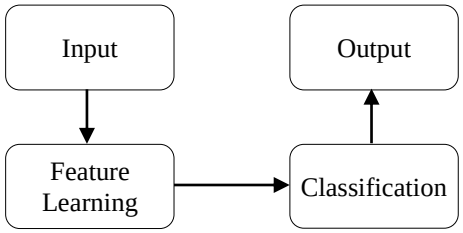
Setelah dilakukan proses pengumpulan data kemudian dilakukan data pre-processing agar mempermudah mesin dalam menemukan pola. Metode yang digunakan dalam proses pre-processing data adalah Fiturisasi data dan Oversampling. Fiturisasi data merupakan tahap ekstraksi fitur dari data graf. Dalam hal ini node merupakan atom dan edge merupakan ikatan antar atom. Graf direpresentasikan dalam bentuk matriks. Setiap node dalam graf diberikan fitur awal berupa representasi numerik yang menggambarkan atribut node.

Oversampling merupak metode yang digunakan dalam pemrosesan data yang tidak seimbang. Oversampling bertujuan untuk membuat data menjadi seimbang. Dataset yang digunakan memiliki kelas negatif (0) lebih banyak disbanding kelas positif (1). Penambahan jumlah kelas positif mengikuti proporsi antara kelas positif dan negatif, seperti pada persamaan berikut.

$$multiplier = int\left(\frac{negatif\ class}{positif\ class}\right) - 1 \quad (1)$$

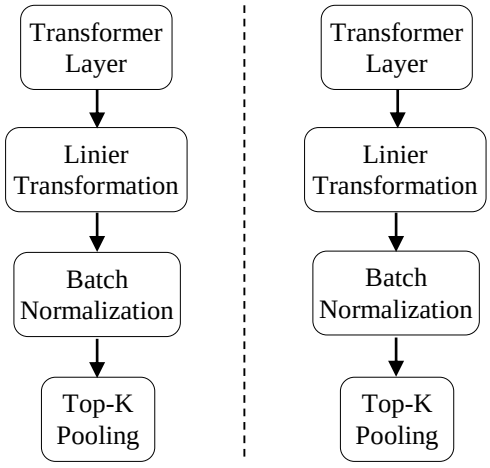
C. Arsitektur Model

Secara umum, hanya ada 2 bagian utama dalam arsitektur yang akan digunakan, yakni feature learning dan classification. Bagian pertama akan mengolah data yang masuk untuk membaca korelasi informasi antar tiap titik data dan menghasilkan suatu kumpulan fitur abstrak yang akan dimanfaatkan predictor untuk memprediksi output. Arsitektur keseluruhan terlihat pada Gambar 1.



Gambar 1. Arsitektur keseluruhan

Feature learning terdiri dari beberapa lapisan, yaitu Transformer Convolution [10], Linear Transformation, Batch Normalization dan Top-K Pooling. Lapisan-lapisan tersebut diulang beberapakali sebanyak n, Dimana n merupakan banyaknya pengulangan atau tumpukan lapisan-lapisan tersebut. Diagram dari proses Feature Learning terlihat pada Gambar 2.



Gambar 2. Feature Learning

Hasil feature learning menjadi masukan pada classification, dimana merupakan proses klasifikasi senyawa yang berpotensi menjadi inhibitor SARS-CoV-2. Pada proses klasifikasi terdiri dari beberapa lapisan fully connected network. Hasil dari predictor merupakan klasifikasi antar senyawa berpotensi menjadi inhibitor (class:1) atau senyawa yang tidak berpotensi menjadi inhibitor (class:0)

D. Loss Function

BCEWithLogitsLoss merupakan Loss function yang digunakan dalam mengukur performa dari model. BCEWithLogitsLoss merupakan kombinasi dari fungsi aktivasi sigmoid dan binary cross-entropy loss. Loss function mengikuti persamaan berikut:

$$l(x, y) = mean\{l_1, \cdots l_N\}^T \quad (2)$$

$$l_n = -w_n\left[y_n \log \delta(x_n) + (1 - y_n) \log\left(1 - \delta(x_n)\right)\right] \quad (3)$$

E. Hyperparameter Tuning

Hyperparameter tuning merupakan proses pada Deep Learning di mana parameter-parameter yang tidak dapat dipelajari oleh model (*hyperparameter*) disesuaikan atau disetel sedemikian rupa untuk meningkatkan kinerja model. *Hyperparameter* dikontrol secara eksternal yang membantu model belajar dari data. Berikut *hyperparameter* yang disesuaikan daintaranya: Batch Size, Learning Rate, Weight Decay, SGD Momentum, Scheduler Gamma, Pos Weight, Model Embedding Size, Model Attention Heads, Model Layers, Model Dropout Rate, Model Top K Ratio, Model Top K Every n, dan Model Dense Neurons. Metode *hyperparameter tuning* yang digunakan adalah Random Search. Pemilihan *hyperparameter* dilakukan denagn memilih nilai *hyperparameter* secara acak dari ruang pencarian dan mengkombinasikannya.

III. HASIL DAN PEMBAHASAN

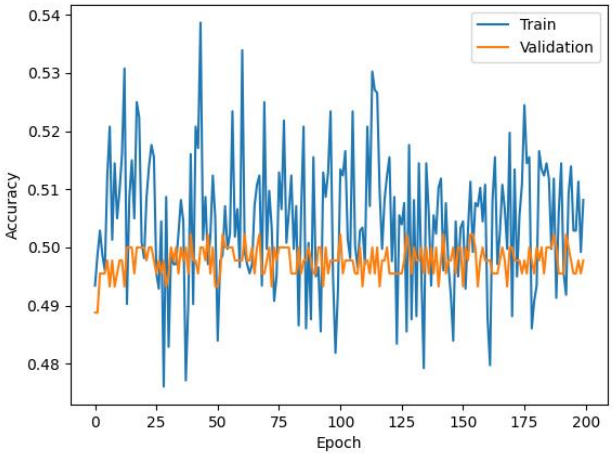
A. Hasil

Dataset dibagi menjadi dua bagian, yaitu *train* dan *validation* dengan proporsi 70:30. baseline model dilatih dan dievaluasi menggunakan data tersebut. Baseline model merupakan model GNN yang dibuat tanpa proses *hyperparameter tuning*. *Hyperparameter* yang digunakan pada Baseline Model ditunjukan pada Tabel II.

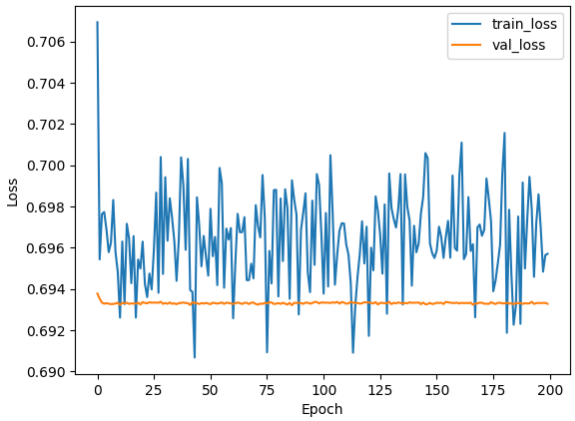
TABEL II
HYPERPARAMETER BASELINE MODEL

Hyperparameter	Nilai
Batch Size	32
Learning Rate	0.005
Weight Decay	0.001
SGD Momentum	0.8
Scheduler Gamma	0.995
Pos Weight	1
Model Embedding Size	16
Model Attention Heads	2
Model Layers	3
Model Dropout Rate	0.5
Model Top K Ratio	0.9
Model Top K Every n	1
Model Dense Neurons	256

Baseline model dilatih sebanyak 200 epoch dengan *hyperparameter* pada Tabel II. Hasil performa dari baseline model diukur oleh akurasi dan loss. Performa baseline model terlihat pada Gambar 3 dan Gambar 4. Nilai akurasi tertinggi yang dapat dicapai oleh baseline model adalah 0.538 pada tahap train dan 0.502 pada tahap validation, sementara nilai loss terbaik yang dicapai adalah 0.691 pada tahap train dan 0.693 pada tahap validation.



Gambar 3. Akurasi train dan validation baseline model



Gambar 4. Loss train dan validation baseline model

Gambar 3 menunjukan akurasi pada setiap epoch dari model. Tren akurasi model terlihat belum menunjukan kenaikan, hal tersebut menunjukan tingkat klasifikasi model belum baik. Gambar 4 menunjukan loss pada setiap epoch dari model. Nilai loss setiap epoch terlihat tidak meningkat. Hal ini menunjukan pada setiap iterasi model belum dapat belajar dengan baik.

TABEL III
RUANG PENCARIAN HYPERPARAMETER

Hyperparameter	Ruang Pencarian
Batch Size	32, 64, 128
Learning Rate	0.1, 0.01, 0.05, 0.005
Weight Decay	0.001, 0.0001, 0.00001
SGD Momentum	0.8, 0.9, 0.5
Scheduler Gamma	0.5, 0.8, 0.9, 0.995, 1
Pos Weight	0.9, 1, 1.1
Model Embedding Size	8, 16, 32, 64, 128
Model Attention Heads	1, 2, 3, 4
Model Layers	1, 2, 3, 4, 5
Model Dropout Rate	0.2, 0.5, 0.9
Model Top K Ratio	0.2, 0.5, 0.8, 0.9
Model Top K Every n layer	1, 2
Model Dense Neurons	16, 32, 64, 128, 256

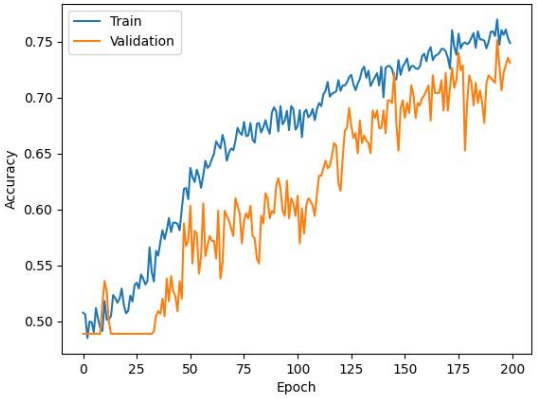
Pencarian Hyperparameter yang digunakan pada model dipilih ulang secara acak mengikuti ruang pencarian pada Tabel III. Ruang pencarian pada setiap *hyperparameter* dipilih berdasarkan ketersedian sumber daya komputasi. Untuk menghindari kombinasi yang terlalu banyak maka dibuat pemilihan pada setiap *hyperparameter* tidak melebihi 5 nilai.

Eksperimen pada setiap pemilihan *hyperparameter* dilakukan sebanyak 50 epoch. Hasil eksperimen dengan nilai *hyperparameter* berbeda dievaluasi berdasakan nilai loss. Nilai loss terbaik didapatkan dengan kombinasi *hyperparameter* seperti pada Tabel IV. Nilai loss yang didapat dengan *hyperparameter* pada Tabel IV adalah 0.64.

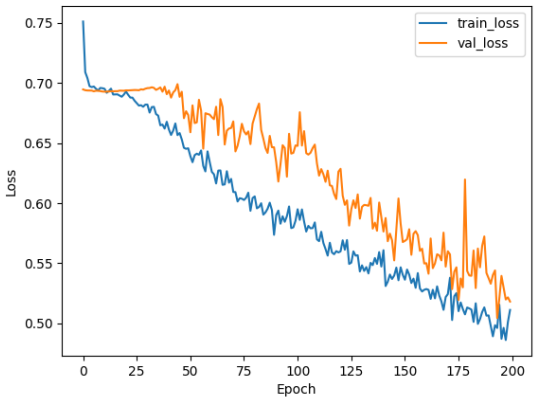
TABEL IV
HYPERPARAMETER DENGAN NILAI TERBAIK

Hyperparameter	Nilai Terbaik
Batch Size	32
Learning Rate	0.005
Weight Decay	0.001
SGD Momentum	0.8
Scheduler Gamma	0.995
Pos Weight	1
Model Embedding Size	16
Model Attention Heads	2
Model Layers	3
Model Dropout Rate	0.5
Model Top K Ratio	0.9
Model Top K Every n layer	1
Model Dense Neurons	256

Setelah mendapatkan *hyperparameter* dengan nilai loss terbaik, model dilatih sebanyak 200 epoch. Hasil akurasi dan loss pada tahap training dan validation terlihat pada Gambar 5 dan Gambar 6. Nilai akurasi tertinggi yang dapat dicapai oleh model adalah 0.782 tahap train dan 0.745 pada tahap validation, sementara nilai loss terbaik yang dicapai adalah 0.478 pada tahap train dan 0.500 pada tahap validation.



Gambar 5. Akurasi train dan validation setelah hyperparameter tuning



Gambar 6. Loss train dan validation setelah hyperparameter tuning

Nilai akurasi dan loss model setelah *hyperparameter tuning* mengalami peningkatan. Nilai akurasi tertinggi model pada *train* dan *validation* adalah 0.538 dan 0.502 meningkat menjadi 0.782 dan 0.745. Nilai loss terbaik model pada *train* dan *validation* adalah 0.691 dan 0.693 trun menjadi 0.478 dan 0.500. Tren grafik akurasi mengalami perubahan dari yang sebelumnya konstan menjadi meningkat dan tren grafik loss dari yang konstan menjadi menurun.

B. Pembahasan

Dataset inhibitor SARS-CoV-2 memiliki perbandingan cukup jauh antara kelas positif dan negatif. Perbedaan tersebut dapat membuat model sulit dalam proses klasifikasi. Proses oversampling data membantu model lebih baik dalam membedakan antara kelas positif dan negatif. Arsitektur GNN dibuat mengikuti Gambar 1 dilatih dengan data yang telah melewati tahap oversampling. Model GNN dilatih dengan hyperparameter pada Tabel II, model tersebut menjadi baseline model. Baseline model dapat membantu menunjukan peningkatan performa model setelah dan sebelum dilakukan proses hyperparameter tuning.

Performa baseline model diukur menggunakan akurasi dan loss. Performa baseline model terlihat pada Gambar 3 dan Gambar 4. Pada Gambar 3 terlihat tingkat akurasi model masih rendah dan loss model masih tinggi. Akurasi terlihat tidak meningkat dan cenderung konstan. Hal tersebut menunjukan model masih buruk dalam prediksi antara kelas positif dan negatif. Pada grafik loss, nilai loss model pada setiap epoch tidak mengalami penurunan. Nilai loss tersebut menunjukan model tidak belajar dengan baik pada setiap iterasi. Secara umum, berdasarkan pengukuran akurasi dan loss, Baseline model belum dapat membedakan atau mengklasifikasikan inhibitor SARS-CoV-2 dengan baik.

Penelitian mengenai Graph Neural Network dan Geometric Deep Learning menunjukan kemampuan dan potensi yang baik model dalam prediksi menggunakan data non-Euclidean [4], [11]. Untuk meningkatkan performa model dalam proses klasifikasi inhibitor SARS-CoV-2 dilakukan proses *hyperparameter tuning*. Proses *hyperparameter tuning* membantu model mencapai performa terbaik, dengan membantu model mencari parameter yang tidak dapat dipelajari sendiri oleh model.

Hyperparameter dipilih berdasarkan ruang pencarian pada Tabel III. Eksperimen dilakukan secara acak pada setiap nilai *hyperparameter* dengan iterasi sebanyak 50 epoch. Evaluasi eksperimen dilakukan menggunakan nilai loss. Nilai loss terbaik didapatkan dengan nilai *hyperparameter* yang tercantum pada Tabel III. Model dilatih kembali dengan menggunakan nilai *hyperparameter* terbaik. Hasil akurasi dan loss model setelah hyperparameter tuning terlihat pada Gambar 5 dan Gambar 6. Nilai akurasi terlihat mengalami kenaikan. Nilai akurasi mencapai 0.782 pada tahap train dan 0.745 pada tahap validation. Nilai akurasi menunjukan model dapat membedakan kelas positif dan negatif dengan baik. Klasifikasi model dapat bernilai benar sebesar 78%. Pada nilai loss terlihat memiliki kecenderungan untuk terus menurun. Hal ini menunjuka pada setiap tahap iterasi model mampu belajar dengan baik. Nilai loss mencapai 0.478 pada tahap train dan 0.500 pada tahap validation.

Hyperparameter tuning memiliki peran penting dalam meningkatkan performa model. Terlihat pada performa model sebelum dan setelah *hyperparameter* tuning memiliki perbedaan yang signifikan, kecenderungan nilai akurasi terus meningkat dan nilai loss menurun setelah tuning. Hal tersebut menunjukan *hyperparameter* tuning berhasil dalam meningkatkan performa dari model. Setelah *hyperparameter* tuning model dapat memprediksi dengan baik mencapai nilai akurasi 78%. Hal tersebut relevan dengan penelitian sebelumnya yang menunjukan Graph Neural Network memiliki performa baik untuk data non-Euclidian. Performa model dapat lebih lebih ditingkatkan, terlihat dari kecenderungan nilai akurasi yang terus meningkat tiap epoch. Hal tersebut menunjukan bahwa dengan menambah jumlah epoch memungkinkan untuk meningkatkan performa model.

IV. KESIMPULAN

Proses *hyperparameter tuning* memiliki peran penting pada model yang telah dibuat. Berdasarkan hasil penelitian menunjukan bahwa *hyperparameter tuning* berhasil meningkatkan performa model. Terjadi kenaikan performa yang signifikan setelah dilakukan *hyperparameter tuning*. Akurasi model meningkat dari 0.53 menjadi 0.78. Akurasi model mencapai 0.78, hal ini menunjukan model dapat mengklasifikasikan inhibitor dengan baik. Tren grafik akurasi masih terus meningkat, memungkinkan model dapat menghasilkan akurasi yang lebih baik dengan penambahan epoch. Penelitian telah mencapai tujuan, yaitu performa model meningkat setelah proses *hyperparameter tuning* dan dapat mengklasifikasikan inhibitor dengan baik.

UCAPAN TERIMA KASIH

Penelitian ini didukung oleh Pusat Penelitian dan Pengabdian Masyarakat, sebagai Penelitian Dosen Pemula 2023, yang dikelola oleh Politeknik Negeri Indramayu.

DAFTAR PUSTAKA

[1] World Health Organization, “Current COVID-19 Situation: Overview Of SARS-CoV-2 Circulating Variants,” 2023.

[2] F. Touret *et al.*, “In vitro screening of a FDA approved chemical library reveals potential inhibitors of SARS-CoV-2 replication,” *Sci Rep*, vol. 10, no. 1, p. 13093, 2020, doi: 10.1038/s41598-020-70143-6.

[3] V. Gawriljuk *et al.*, “Machine Learning Models Identify Inhibitors of SARS-CoV-2,” *J Chem Inf Model*, vol. 61, Aug. 2021, doi: 10.1021/acs.jcim.1c00683.

[4] R. S. I. Salamet Nur Himawan, “Klasifikasi Inhibitor Sars-Cov-2 Menggunakan Geometric Deep Learning,” in *Prosiding Seminar SeNTIK*, 2023, pp. 78–82.

[5] M. M. Bronstein, J. Bruna, Y. LeCun, A. Szlam, and P. Vandergheynst, “Geometric Deep Learning: Going beyond

Euclidean data," *IEEE Signal Process Mag*, vol. 34, no. 4, pp. 18–42, 2017, doi: 10.1109/MSP.2017.2693418.

[6] F. Scarselli, M. Gori, A. C. Tsoi, M. Hagenbuchner, and G. Monfardini, "The Graph Neural Network Model," *IEEE Trans Neural Netw*, vol. 20, no. 1, pp. 61–80, 2009, doi: 10.1109/TNN.2008.2005605.

[7] S. Kearnes, K. McCloskey, M. Berndl, V. Pande, and P. Riley, "Molecular graph convolutions: moving beyond fingerprints," *J Comput Aided Mol Des*, vol. 30, no. 8, pp. 595–608, 2016, doi: 10.1007/s10822-016-9938-8.

[8] H. Stärk, O. Ganea, L. Pattanaik, Dr. R. Barzilay, and T. Jaakkola, "EquiBind: Geometric Deep Learning for Drug Binding Structure Prediction," in *Proceedings of the 39th International Conference on Machine Learning*, K. Chaudhuri, S. Jegelka, L. Song, C. Szepesvari, G. Niu, and S. Sabato, Eds., in Proceedings of Machine Learning Research, vol. 162. PMLR, Nov. 2022, pp. 20503–20521. [Online]. Available: https://proceedings.mlr.press/v162/stark22b.html

[9] A. C. P. L. G. C. A. R. Y. B. Petar Velickovic, "Graph attention networks," *6th International Conference on Learning Representations, ICLR 2018 - Conference Track Proceedings*, 2018.

[10] Y. Shi, H. Zhengjie, S. Feng, H. Zhong, W. Wang, and Y. Sun, *Masked Label Prediction: Unified Message Passing Model for Semi-Supervised Classification*. 2021. doi: 10.24963/ijcai.2021/214.

[11] Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang, and P. S. Yu, "A Comprehensive Survey on Graph Neural Networks," *IEEE Trans Neural Netw Learn Syst*, vol. 32, no. 1, pp. 4–24, 2021, doi: 10.1109/TNNLS.2020.2978386.