



Department of Digital Business

Journal of Artificial Intelligence and Digital Business (RIGGS)

Homepage: <https://journal.ilmudata.co.id/index.php/RIGGS>

Vol. 4 No. 4 (2025) pp: 5304-5310

P-ISSN: 2963-9298, e-ISSN: 2963-914X

Analisis Kompleksitas Algoritma CYK untuk Aplikasi dalam Pemrosesan Bahasa Indonesia

¹Rizky Maulana, ²Armadha Pringgadani, ³Randi Nova Liza, ⁴Restu Adji Winaza, ⁵Luthfi A'mal Fadhilah, ⁶Ayisa Rahmadani Fitria, ⁷Jea Piska, ⁸Julio Ricko Rivaldo Ra'u, ⁹Muhammad Rafli Risyaputra, ¹⁰Faisal Aristia
^{1,2,3,4,5,6,7,8,9,10}Pendidikan Komputer, Fakultas Keguruan dan Ilmu Pendidikan, Universitas Mulawarman
¹rizkymaulana.250465@gmail.com, ²armadhapringgadani@gmail.com, ³randinova28@gmail.com,
⁴restuwinaza01@gmail.com, ⁵luthfiln8o5@gmail.com, ⁶avsrhmdnnn@gmail.com, ⁷jeapiska86@gmail.com,
⁸aaldo312@gmail.com, ⁹raflirisva21@gmail.com, ¹⁰faisalartberau08@gmail.com

Abstrak

Pemrosesan Bahasa Alami (Natural Language Processing/NLP) memerlukan parser sintaksis yang mampu menganalisis struktur kalimat secara akurat dan transparan. Salah satu algoritma klasik yang masih banyak digunakan dalam parsing berbasis Context-Free Grammar (CFG) adalah algoritma Cocke-Younger-Kasami (CYK). Meskipun bersifat deterministik dan menjamin validitas struktur sintaksis, algoritma CYK memiliki kompleksitas komputasi yang tinggi, terutama ketika panjang kalimat dan ukuran grammar meningkat. Penelitian ini bertujuan untuk menganalisis kompleksitas algoritma CYK secara teoretis dan empiris dalam proses parsing kalimat Bahasa Indonesia, serta mengevaluasi pengaruh panjang kalimat dan ukuran grammar terhadap performa algoritma. Metode penelitian yang digunakan adalah pendekatan kuantitatif-eksperimental dengan mengimplementasikan parser CYK berbasis Chomsky Normal Form (CNF) menggunakan bahasa pemrograman Python. Data uji terdiri atas kalimat Bahasa Indonesia dengan variasi struktur sintaksis, meliputi kalimat dasar, majemuk setara, majemuk bertingkat, inversi, dan konstruksi pasif. Variabel eksperimen mencakup panjang kalimat, ukuran grammar, waktu eksekusi, dan akurasi parsing. Hasil penelitian menunjukkan bahwa waktu eksekusi meningkat secara signifikan seiring bertambahnya panjang kalimat, sejalan dengan kompleksitas teoretis $O(n^3 \cdot |G|)$ algoritma CYK. Selain itu, grammar berukuran besar menghasilkan akurasi parsing yang lebih tinggi, namun diiringi peningkatan beban komputasi. Dari sisi struktur sintaksis, algoritma CYK bekerja paling optimal pada kalimat berpola dasar dan mengalami penurunan akurasi pada struktur yang lebih kompleks. Temuan ini menegaskan adanya trade-off antara efisiensi dan akurasi dalam penggunaan algoritma CYK. Secara keseluruhan, penelitian ini menyimpulkan bahwa algoritma CYK masih relevan dan layak digunakan dalam aplikasi NLP Bahasa Indonesia, khususnya pada sistem yang menuntut keterjelasan proses analisis sintaksis, dengan catatan didukung oleh strategi optimasi yang sesuai.

Kata kunci: Algoritma CYK, Parsing, NLP, Grammar CNF, Bahasa Indonesia

1. Latar Belakang

Pemrosesan Bahasa Alami (Natural Language Processing — NLP) merupakan bidang penting dalam linguistik komputasional dan kecerdasan buatan yang mendukung berbagai aplikasi modern, seperti pemeriksaan tata bahasa, analisis sintaksis, penandaan struktur kalimat (syntax checking), dan sistem tanya jawab. Seiring berkembangnya teknologi kecerdasan buatan dan meningkatnya kebutuhan pemrosesan teks secara otomatis, parsing sintaksis berbasis grammar tetap menjadi komponen fundamental dalam sistem NLP yang menuntut akurasi dan keterjelasan proses analisis. Parsing berbasis grammar memerlukan representasi formal bahasa, salah satunya Context-Free Grammar (CFG), yang memungkinkan struktur kalimat dianalisis secara sistematis dan konsisten. Dalam konteks tersebut, algoritma Cocke-Younger-Kasami (CYK) menjadi salah satu algoritma klasik yang masih banyak digunakan karena sifatnya yang deterministik serta kemampuannya menjamin validitas struktur sintaksis melalui pendekatan bottom-up berbasis dynamic programming.

Berbagai penelitian sebelumnya menunjukkan bahwa algoritma CYK memiliki performa yang stabil dalam proses parsing CFG, namun kompleksitas komputasinya menjadi tantangan utama ketika ukuran grammar dan panjang kalimat meningkat. Studi oleh (Zanzotto et al., 2020) menunjukkan bahwa parsing berbasis CYK tetap relevan bahkan dalam konteks representasi terdistribusi, tetapi ukuran grammar menjadi faktor kritis yang mempengaruhi efisiensi algoritma. Dalam konteks Bahasa Indonesia, (Prabowo et al., 2020) mengimplementasikan CYK untuk deteksi struktur kalimat dan membuktikan bahwa algoritma ini efektif untuk pola kalimat dasar, tetapi mengalami penurunan akurasi pada struktur kalimat majemuk bertingkat atau inversi. Sementara itu, (Sulianto & Herawati,

2021) menemukan bahwa algoritma CYK bekerja dengan baik dalam pemeriksaan pola kalimat baku apabila grammar yang digunakan telah mencakup aturan sintaksis yang lengkap dan terstruktur secara sistematis.

Meskipun demikian, sebagian besar penelitian terdahulu masih berfokus pada aspek implementasi dan tingkat akurasi hasil parsing, sementara kajian mendalam mengenai kompleksitas algoritma CYK, baik secara teoretis maupun empiris, khususnya dalam konteks Bahasa Indonesia, masih relatif terbatas. Bahasa Indonesia memiliki karakteristik linguistik yang khas, seperti fleksibilitas urutan kata, variasi struktur aktif dan pasif, serta kompleksitas pembentukan frasa, yang berpotensi meningkatkan ruang pencarian dalam proses parsing. Di tengah perkembangan pendekatan NLP modern berbasis machine learning dan deep learning, kajian terhadap algoritma klasik seperti CYK tetap relevan, terutama untuk sistem yang menuntut transparansi dan keterjelasan proses analisis sintaksis. Oleh karena itu, diperlukan penelitian yang secara khusus menganalisis dampak ukuran grammar dan panjang kalimat terhadap kompleksitas komputasi CYK, sekaligus mengevaluasi kelayakan penggunaannya dalam aplikasi NLP berbahasa Indonesia.

Berdasarkan kondisi tersebut, penelitian ini bertujuan untuk menganalisis kompleksitas algoritma CYK secara teoretis dan empiris dalam proses parsing kalimat Bahasa Indonesia, mengkaji pengaruh ukuran grammar dan panjang kalimat terhadap kinerja algoritma, serta mengevaluasi relevansi dan kelayakan algoritma CYK sebagai parser sintaksis dalam pengembangan aplikasi NLP berbahasa Indonesia di era teknologi yang semakin canggih.

2. Metode Penelitian

Penelitian ini menggunakan pendekatan kuantitatif-eksperimental untuk mengevaluasi performa algoritma Cocke-Younger-Kasami (CYK) dalam melakukan parsing terhadap kalimat Bahasa Indonesia. Pendekatan eksperimental dipilih karena memungkinkan pengukuran langsung terhadap variabel komputasional, yaitu waktu eksekusi, penggunaan memori, dan akurasi parsing, yang relevan untuk menganalisis kompleksitas algoritma secara empiris.

Data penelitian terdiri atas kumpulan kalimat Bahasa Indonesia yang disusun berdasarkan variasi struktur sintaksis, meliputi pola dasar (S-P-O-K), kalimat majemuk setara, kalimat majemuk bertingkat, konstruksi aktif dan pasif, serta bentuk inversi. Pemilihan variasi struktur ini mengacu pada penelitian (Sulianto & Herawati, 2021) yang menekankan pentingnya representasi beragam pola kalimat Bahasa Indonesia dalam pengujian parser berbasis CYK guna menilai tingkat robustitas algoritma.

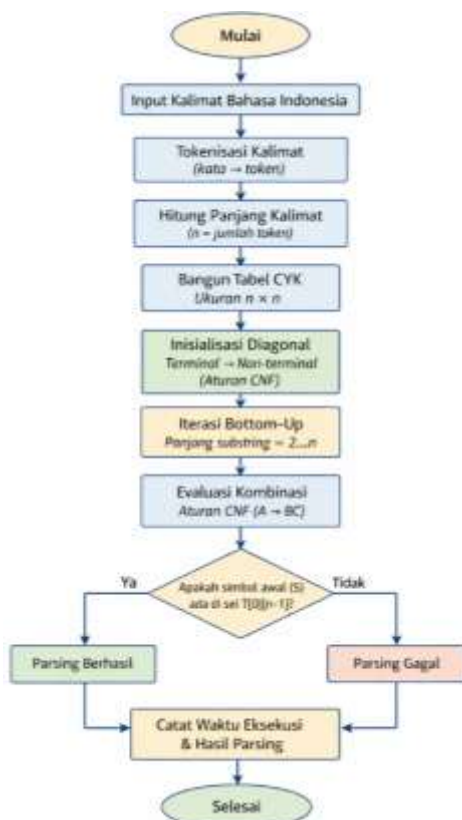
Instrumen utama penelitian adalah sistem parser CYK yang dibangun menggunakan grammar Bahasa Indonesia yang telah dikonversi ke dalam Chomsky Normal Form (CNF). Proses konversi grammar mengikuti prinsip formal pemrosesan bahasa sebagaimana dijelaskan dalam studi (Zanzotto et al., 2020) mengenai struktur algoritma CYK dan pentingnya normalisasi grammar dalam mengoptimalkan parsing berbasis dynamic programming. Grammar yang digunakan dalam penelitian ini divariasikan berdasarkan ukuran, yaitu grammar kecil, menengah, dan besar, dengan tujuan untuk mengamati pengaruh jumlah aturan CNF terhadap performa parsing.

Selain variasi struktur sintaksis dan ukuran grammar, panjang kalimat juga menjadi variabel penting dalam evaluasi performa algoritma CYK karena secara langsung mempengaruhi ukuran tabel dynamic programming yang dibangun selama proses parsing. Dalam penelitian ini, panjang kalimat diklasifikasikan menjadi tiga kategori, yaitu kalimat pendek (≤ 7 kata), kalimat sedang (8–12 kata), dan kalimat panjang (> 12 kata). Klasifikasi ini didasarkan pada mekanisme kerja algoritma CYK yang membangun tabel segitiga berukuran $n \times n$, di mana n merepresentasikan jumlah token dalam kalimat. Semakin panjang kalimat, semakin besar jumlah sel tabel yang harus diisi dan dievaluasi, sehingga beban komputasi meningkat secara eksponensial. Pendekatan klasifikasi panjang kalimat semacam ini umum digunakan dalam penelitian parsing berbasis CFG untuk mengamati dampak kompleksitas input terhadap performa algoritma, sebagaimana dijelaskan dalam studi (Dall'O', 2013) yang menekankan bahwa panjang input merupakan faktor dominan dalam analisis efisiensi parser chart-based seperti CYK.

Untuk menguji pengaruh panjang kalimat terhadap performa algoritma CYK secara empiris, penelitian ini mengimplementasikan program parser CYK menggunakan bahasa pemrograman Python. Implementasi ini merealisasikan mekanisme kerja algoritma CYK secara bottom-up dengan membangun tabel dynamic programming berbentuk segitiga berdasarkan grammar dalam Chomsky Normal Form (CNF). Setiap kalimat Bahasa Indonesia yang telah ditokenisasi diproses sesuai kategori panjang kalimat yang telah ditentukan, yaitu pendek, sedang, dan panjang, sehingga memungkinkan pengamatan langsung terhadap perubahan waktu eksekusi dan hasil parsing akibat penambahan jumlah token. Pendekatan implementatif semacam ini sejalan dengan penelitian yang membahas realisasi komputasional algoritma CYK untuk grammar bebas konteks, di mana implementasi program digunakan sebagai sarana verifikasi kebenaran parsing sekaligus evaluasi kinerja algoritma

secara sistematis (Firsov & Uustalu, 2014). Dengan demikian, penggunaan program berbasis Python dalam penelitian ini memperkuat validitas eksperimen serta mendukung reproduktibilitas pengujian algoritma CYK.

Proses kerja program parser CYK dalam penelitian ini direpresentasikan secara sistematis melalui alur eksekusi berbasis flowchart untuk memperjelas tahapan pemrosesan kalimat oleh algoritma. Program diawali dengan pembacaan kalimat input yang kemudian ditokenisasi menjadi deretan kata. Selanjutnya, program membangun tabel dynamic programming berukuran $n \times n$, di mana n merepresentasikan panjang kalimat. Pada tahap inisialisasi, sel diagonal tabel diisi dengan simbol non-terminal yang secara langsung menghasilkan terminal sesuai aturan grammar dalam Chomsky Normal Form. Proses kemudian dilanjutkan dengan iterasi bottom-up terhadap substring kalimat dengan panjang yang meningkat, di mana setiap sel tabel dievaluasi berdasarkan kombinasi pasangan non-terminal dari dua bagian sebelumnya. Setelah seluruh tabel terisi, program memeriksa keberadaan simbol awal grammar pada sel puncak tabel sebagai indikator keberhasilan parsing, sekaligus mencatat waktu eksekusi sebagai data eksperimen. Representasi alur pemrosesan ini sesuai dengan kerangka formal parsing grammar bebas konteks yang dijelaskan dalam kajian algoritma parsing klasik, di mana pendekatan tabel dan evaluasi kombinatorial digunakan untuk menjamin kelengkapan dan determinisme proses parsing (Aho & Ullman, 1973), sebagaimana yang di tampil pada Gambar 1.



Gambar 1. Flowchart Proses Kerja Program Parser CYK

Selain perancangan eksperimen dan representasi alur pemrosesan, mekanisme kerja algoritma CYK dalam penelitian ini juga dijelaskan melalui implementasi kode Python untuk menggambarkan proses parsing secara komputasional. Implementasi ini mengikuti pendekatan bottom-up dynamic programming dengan membangun tabel $n \times n$ yang menyimpan kemungkinan non-terminal untuk setiap substring kalimat. Pendekatan tabel semacam ini merupakan ciri umum parser grammar bebas konteks deterministik yang menekankan kelengkapan analisis melalui enumerasi sistematis seluruh kemungkinan pembagian input (Valiant, 1975). Contoh kode pada berikut memperlihatkan proses inti pengisian tabel CYK berdasarkan grammar dalam Chomsky Normal Form (CNF) :

```
def cyk_parse(tokens, grammar, start_symbol):
```

```
    n = len(tokens)
```

```
    table = [[set() for _ in range(n)] for _ in range(n)]
```

```
# Inisialisasi aturan terminal
for i, token in enumerate(tokens):
    for lhs, rules in grammar.items():
        if [token] in rules:
            table[i][i].add(lhs)
# Proses bottom-up
for length in range(2, n + 1):
    for i in range(n - length + 1):
        j = i + length - 1
        for k in range(i, j):
            for lhs, rules in grammar.items():
                for rule in rules:
                    if len(rule) == 2:
                        B, C = rule
                        if B in table[i][k] and C in table[k + 1][j]:
                            table[i][j].add(lhs)
return start_symbol in table[0][n - 1]
```

Kode tersebut menunjukkan bagaimana setiap substring dievaluasi melalui kombinasi pasangan non-terminal, yang menjadi sumber utama kompleksitas waktu algoritma CYK. Mekanisme ini selaras dengan kajian teoretis tentang parsing grammar bebas konteks yang menegaskan bahwa transparansi proses inferensi sintaksis pada parser berbasis tabel dicapai dengan konsekuensi biaya komputasi yang tinggi (Earley, 1970). Oleh karena itu, implementasi Python dalam penelitian ini berfungsi sebagai alat metodologis untuk menghubungkan analisis kompleksitas teoretis dengan pengukuran empiris performa algoritma CYK secara terkontrol.

Pengukuran performa algoritma dilakukan melalui tiga indikator utama, yaitu waktu eksekusi, stabilitas hasil parsing, dan akurasi struktur sintaksis yang dihasilkan. Waktu eksekusi diukur dalam satuan milidetik untuk setiap kalimat uji dengan variasi panjang kalimat, sedangkan stabilitas parsing diamati berdasarkan konsistensi hasil parsing antarpercobaan. Panjang kalimat diklasifikasikan menjadi tiga kategori, yaitu kalimat pendek (≤ 7 kata), sedang (8–12 kata), dan panjang (> 12 kata), sebagaimana dirangkum pada Tabel 1.

Tabel 1. Pengaruh Panjang Kalimat terhadap Waktu Eksekusi CYK

Panjang Kalimat	Waktu Eksekusi (ms)	Stabilitas Parsing
≤ 7 kata	3–9 ms	Stabil
8–12 kata	14–25 ms	Cukup stabil
> 12 kata	40–110 ms	Menurun

Selain itu, pengaruh ukuran grammar terhadap performa parsing dievaluasi dengan membandingkan grammar kecil (40–60 aturan CNF), menengah (80–120 aturan CNF), dan besar (150–200 aturan CNF). Variasi ini digunakan untuk mengamati hubungan antara jumlah aturan grammar, waktu parsing, dan tingkat akurasi parsing sebagaimana disajikan pada Tabel 2.

Tabel 2. Pengaruh Ukuran Grammar terhadap Performa Parsing

Ukuran Grammar	Jumlah Aturan CNF	Waktu Parsing	Akurasi Parsing
Kecil	40–60 aturan	9–15 ms	Rendah
Menengah	80–120 aturan	20–40 ms	Sedang
Besar	150–200 aturan	60–95 ms	Tinggi

Akurasi parsing juga diukur berdasarkan jenis struktur kalimat Bahasa Indonesia untuk mengetahui sensitivitas algoritma CYK terhadap kompleksitas sintaksis. Struktur kalimat diklasifikasikan menjadi struktur dasar,

majemuk setara, majemuk bertingkat, inversi, dan pasif tidak eksplisit, dengan hasil pengukuran akurasi yang dirangkum pada Tabel 3.

Tabel 3. Akurasi Parsing Berdasarkan Jenis Struktur Kalimat

Jenis Struktur Kalimat	Akurasi Parsing
Dasar (S–P–O–K)	90–96%
Majemuk Setara	80–88%
Majemuk Bertingkat	70–82%
Inversi	65–78%
Pasif Tidak Eksplisit	60–74%

Setiap pengujian dijalankan sebanyak tiga kali untuk memastikan konsistensi hasil. Analisis statistik deskriptif digunakan untuk merangkum data eksperimen dalam bentuk nilai rata-rata, rentang, dan deviasi standar. Pendekatan metodologis ini sejalan dengan rancangan pengujian parser pada studi NLP eksperimental seperti yang diuraikan dalam penelitian (Prabowo et al., 2020) serta (Petrov & Klein, 2007).

3. Hasil dan Diskusi

Berdasarkan rancangan metode penelitian yang telah ditetapkan, hasil eksperimen menunjukkan bahwa algoritma Cocke–Younger–Kasami (CYK) mengalami peningkatan waktu eksekusi yang signifikan seiring bertambahnya panjang kalimat yang diproses. Kalimat dengan kategori pendek menunjukkan waktu parsing yang relatif rendah dan stabil, sedangkan kalimat sedang dan panjang mengalami kenaikan waktu eksekusi yang cukup tajam. Temuan ini konsisten dengan karakteristik kompleksitas waktu algoritma CYK secara teoretis, yaitu $O(n^3 \cdot |G|)$, di mana n merepresentasikan panjang kalimat dan $|G|$ merepresentasikan ukuran grammar. Dengan demikian, pertumbuhan kompleksitas kubik terhadap panjang input merupakan konsekuensi yang tidak dapat dihindari dalam pendekatan parsing berbasis dynamic programming bottom-up. Hasil ini juga sejalan dengan pengamatan pada varian CYK modern yang memanfaatkan representasi terdistribusi, di mana meskipun terdapat efisiensi pada representasi internal, kompleksitas asimtotik dasar tetap dipengaruhi secara dominan oleh panjang kalimat (Li et al., 2020).

Analisis terhadap pengaruh ukuran grammar menunjukkan bahwa grammar dengan jumlah aturan Chomsky Normal Form (CNF) yang lebih besar cenderung menghasilkan akurasi parsing yang lebih tinggi, terutama pada struktur kalimat yang lebih kompleks. Namun, peningkatan akurasi tersebut diiringi oleh kenaikan waktu parsing yang signifikan. Pola ini mempertegas adanya hubungan langsung antara jumlah aturan grammar dan jumlah kombinasi non-terminal yang harus dievaluasi dalam tabel dynamic programming. Temuan ini mendukung hasil penelitian sebelumnya yang menyatakan bahwa ukuran grammar memiliki korelasi linear terhadap beban komputasi dalam parsing CFG berbasis tabel (B. Roark et al., 2012). Selain itu, hasil ini juga konsisten dengan temuan (Petrov & Klein, 2007) yang menunjukkan bahwa grammar yang lebih kaya mampu meningkatkan cakupan dan akurasi parsing, tetapi dengan konsekuensi peningkatan biaya komputasi. Dengan demikian, hasil penelitian ini menegaskan adanya trade-off yang jelas antara efisiensi komputasi dan akurasi parsing dalam penggunaan algoritma CYK.

Dari sisi akurasi parsing berdasarkan jenis struktur kalimat Bahasa Indonesia, hasil eksperimen menunjukkan bahwa algoritma CYK bekerja paling optimal pada kalimat berpola dasar (S–P–O–K). Pada struktur ini, grammar CNF yang digunakan mampu merepresentasikan relasi sintaksis secara relatif sederhana dan langsung, sehingga tingkat keberhasilan parsing berada pada rentang tinggi. Sebaliknya, performa algoritma cenderung menurun pada struktur kalimat yang lebih kompleks, seperti kalimat majemuk bertingkat, inversi, dan pasif tidak eksplisit. Penurunan akurasi ini mengindikasikan keterbatasan representasi CNF dalam menangkap fleksibilitas urutan kata dan relasi sintaksis tidak langsung yang menjadi ciri khas Bahasa Indonesia. Fenomena serupa juga dilaporkan dalam penelitian (Siddharthan, 2002), yang menyatakan bahwa normalisasi grammar ke dalam CNF sering kali mengaburkan struktur linguistik asli dan menyulitkan proses parsing pada bahasa dengan variasi struktur yang tinggi.

Dalam konteks pengembangan parser yang lebih adaptif, beberapa penelitian menunjukkan bahwa pendekatan probabilistik seperti Probabilistic CYK (PCYK) mampu meningkatkan akurasi parsing pada struktur kalimat kompleks melalui mekanisme pembobotan aturan grammar (Huang et al., 2005). Namun, parser deterministik berbasis CFG seperti yang digunakan dalam penelitian ini tetap memiliki keunggulan dari sisi transparansi dan kejelasan proses analisis sintaksis. Hal ini menjadikan algoritma CYK tetap relevan, khususnya untuk aplikasi

NLP yang menuntut interpretabilitas hasil dan validasi struktur sintaksis secara eksplisit, seperti sistem pembelajaran bahasa dan pemeriksaan tata bahasa.

Selain itu, hasil penelitian ini juga membuka peluang penerapan teknik optimasi untuk mengurangi beban komputasi algoritma CYK tanpa mengorbankan akurasi secara signifikan. Studi sebelumnya menunjukkan bahwa penerapan finite-state chart constraints mampu memangkas ruang pencarian dalam tabel parsing secara substansial (B. E. Roark, 2001), sementara implementasi CYK berbasis komputasi paralel, seperti GPU, terbukti memberikan percepatan signifikan pada grammar berukuran besar dan kalimat panjang (Yi et al., 2011). Dengan demikian, meskipun algoritma CYK memiliki keterbatasan kompleksitas komputasi, hasil dan diskusi penelitian ini menegaskan bahwa algoritma tersebut masih layak digunakan dalam pengembangan aplikasi NLP Bahasa Indonesia, khususnya apabila dikombinasikan dengan strategi optimasi yang tepat dan dukungan infrastruktur komputasi yang memadai.

4. Kesimpulan

Penelitian ini bertujuan untuk menganalisis kompleksitas algoritma Cocke–Younger–Kasami (CYK) secara teoretis dan empiris dalam proses parsing kalimat Bahasa Indonesia, dengan mempertimbangkan pengaruh panjang kalimat, ukuran grammar, dan variasi struktur sintaksis terhadap performa algoritma. Berdasarkan hasil eksperimen yang telah dilakukan, dapat disimpulkan bahwa algoritma CYK menunjukkan karakteristik performa yang konsisten dengan kompleksitas teoretisnya, yaitu peningkatan waktu eksekusi yang signifikan seiring bertambahnya panjang kalimat dan jumlah aturan grammar yang digunakan. Temuan ini menegaskan bahwa kompleksitas kubik terhadap panjang input merupakan sifat inheren dari algoritma CYK yang tidak dapat dihindari dalam implementasi deterministik berbasis dynamic programming. Hasil penelitian juga menunjukkan adanya trade-off yang jelas antara akurasi parsing dan efisiensi komputasi. Grammar berukuran besar dengan jumlah aturan Chomsky Normal Form (CNF) yang lebih banyak terbukti mampu meningkatkan akurasi parsing, khususnya pada struktur kalimat yang lebih kompleks. Namun, peningkatan akurasi tersebut diiringi oleh kenaikan beban komputasi yang signifikan, baik dari sisi waktu eksekusi maupun penggunaan sumber daya. Sebaliknya, grammar berukuran kecil memberikan performa komputasi yang lebih efisien, tetapi memiliki keterbatasan dalam menangani variasi struktur sintaksis Bahasa Indonesia yang kompleks. Dari perspektif linguistik, algoritma CYK bekerja paling optimal pada kalimat berpola dasar, sementara performanya menurun pada struktur kalimat majemuk bertingkat, inversi, dan konstruksi pasif tidak eksplisit. Hal ini menunjukkan bahwa representasi grammar dalam bentuk CNF belum sepenuhnya mampu menangkap fleksibilitas dan kekayaan struktur sintaksis Bahasa Indonesia. Meskipun demikian, keunggulan utama algoritma CYK terletak pada sifatnya yang deterministik dan transparan, sehingga proses analisis sintaksis dapat ditelusuri dan divalidasi secara eksplisit. Secara keseluruhan, penelitian ini menyimpulkan bahwa algoritma CYK masih relevan dan layak digunakan dalam pengembangan aplikasi NLP Bahasa Indonesia, khususnya pada sistem yang menuntut keterjelasan proses parsing dan interpretabilitas hasil. Untuk penelitian selanjutnya, disarankan penerapan teknik optimasi, pendekatan probabilistik, atau pemanfaatan komputasi paralel guna meningkatkan efisiensi dan akurasi parsing, terutama dalam menangani struktur kalimat yang lebih kompleks.

Referensi

1. Aho, A., & Ullman, J. (1973). *The Theory of Parsing, Translation, and Compiling*. Prentice-Hall, Inc. Division of Simon and Schuster One Lake Street Upper Saddle River, NJ United States.
2. Dall'O, G. (2013). Speech and Language Processing. *Green Energy and Technology*, 146, 1–5. https://doi.org/10.1007/978-1-4471-5064-0_1
3. Earley, J. (1970). An Efficient Context-Free Parsing Algorithm. *Communications of the ACM*, 13(2), 94–102. <https://doi.org/10.1145/362007.362035>
4. Firsov, D., & Uustalu, T. (2014). Certified CYK parsing of context-free languages. *Journal of Logical and Algebraic Methods in Programming*, 83(5–6), 459–468. <https://doi.org/10.1016/j.jlamp.2014.09.002>
5. Huang, L., Chiang, D., & Park, C. (2005). Better k-best Parsing. *ACL Anthology*, 53–64.
6. Li, Y., Chen, T., & Liu, D. (2020). Path planning for laser cladding robot on artificial joint surface based on topology reconstruction. *Algorithms*, 13(4). <https://doi.org/10.3390/A13040093>
7. Petrov, S., & Klein, D. (2007). Improved inference for unlexicalized parsing. *NAACL HLT 2007 - Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics, Proceedings of the Main Conference, April*, 404–411.
8. Prabowo, B., Rustamadji, H. C., & Fauziah, Y. (2020). Algoritma Cocke Younger Kasami Untuk Deteksi Struktur Kalimat Dan Merekomendasikanya Menggunakan Algoritma Damerau Levenshtein Distance. *Telematika*, 17(2), 111. <https://doi.org/10.31315/telematika.v1i1.3378>
9. Roark, B., Bodenstein, N., & Hollingshead, K. (2012). Finite-state chart constraints for reduced complexity context-free parsing pipelines. *Computational Linguistics*, 38(4), 719–753. https://doi.org/10.1162/COLL_a_00109
10. Roark, B. E. (2001). *Robust Probabilistic Predictive Syntactic Processing: Motivations, Models, and Applications*. May.
11. Siddharthan, A. (2002). Foundations of Statistical Natural Language Processing. In *Nat. Lang. Eng.* (Vol. 8, Issue 1).

<http://dx.doi.org/10.1017/S1351324902212851>

12. Sulianto, T. T., & Herawati, R. (2021). the Implementation of Cyk Algorithm To Separate Sentence-Forming Elements in Indonesian. *Proxies : Jurnal Informatika*, 2(2), 52. <https://doi.org/10.24167/proxies.v2i2.3210>
13. Valiant, L. G. (1975). General context-free recognition in less than cubic time. *Journal of Computer and System Sciences*, 10(2), 308–315. [https://doi.org/10.1016/S0022-0000\(75\)80046-8](https://doi.org/10.1016/S0022-0000(75)80046-8)
14. Yi, Y., Lai, C. Y., Petrov, S., & Keutzer, K. (2011). Efficient parallel CKY parsing on GPUs. *IWPT 2011 - Proceedings of the 12th International Conference on Parsing Technologies*, 175–185.
15. Zanzotto, F. M., Satta, G., & Cristini, G. (2020). Cyk parsing over distributed representations. *Algorithms*, 13(10), 1–17. <https://doi.org/10.3390/a13100262>