

Penerapan Algoritma Not So Naive Pada Pencarian Jurnal Berbasis Mobile

Muhammad Alfarizi, Soumi Rohmah Saragih

Prodi Teknik Informatika, STMIK Budi Darma, Medan, Indonesia
Jln. Sisingamangaraja No. 338, Medan, Indonesia

Abstrak

Artikel ilmiah adalah suatu karya ilmiah yang diterbitkan untuk dimuat dalam jurnal ilmiah dengan tata cara yang diperlukan yang mengikuti pedoman atau konvensi ilmiah yang telah disetujui atau ditentukan. Pada dasarnya artikel ilmiah dapat dipertanggungjawabkan keilmiahannya tergantung dari metode yang dipakai dalam pembuatannya. Walaupun demikian kita tetap harus dapat memilih artikel penelitian yang tepat apalagi untuk dijadikan referensiskripsi. Untuk itu dibutuhkan sebuah aplikasi pengganti jurnal yang mudah dibawa serta dapat dibawa serta dapat digunakan kapan dan dimanapun secara efektif. Aplikasi mobile pencarian jurnal bekerja dengan cara mencari kata yang dibutuhkan oleh pengguna smartphone yang berbasis android. Untuk menyelesaikan masalah pencarian jurnal ini, penulis menggunakan salah satu algoritma string matching yaitu algoritma Not So Naive. Dengan algoritma ini diharapkan mampu menyelesaikan masalah pencarian jurnal.

Kata Kunci: Not So Naive, Jurnal, Mobile

1. PENDAHULUAN

Mahasiswa tingkat akhir harus selalu mengisi dirinya dengan pembekalan referensi-referensi jurnal untuk penelitian skripsi. Sehingga bisa menjadi generasi yang tanggap dan mampu menghadapi tantangan dan tanggung jawab dalam penulisan skripsi. Dengan harapan dapat mendekatkan mahasiswa ke masa depan yang diinginkan, dan juga mengingatkan jurnal sebagai referensi dalam memenuhi tugas akhir sebagai mahasiswa. Jurnal penelitian merupakan sebuah laporan peneliti tentang hasil penelitian yang telah dilakukan secara ilmiah atau merupakan kesimpulan dari penelitian skripsi yang telah dilakukan. Pada dasarnya, sebagian besar jurnal penelitian dapat dipertanggungjawabkan keilmiahannya tergantung dari metode yang dipakai dalam pembuatan dan penyusunan laporan jurnal penelitian. Walau demikian mahasiswa tetap harus dapat memilih jurnal-jurnal penelitian yang tepat apalagi untuk di jadikan referensi skripsi.

Seiring dengan perkembangan waktu, tingkat mobilitas mahasiswa semakin cepat. Mahasiswa ketika sudah memasuki semester akhir umumnya memiliki kesibukan baik di dalam perkuliahan maupun diluar perkuliahan. Mahasiswa tingkat akhir dituntut untuk konsisten terhadap penugasan yang diberikan oleh kampus untuk menunjang proses perkuliahan mereka. Tidak sedikit jumlah mahasiswa di lingkungan kampus yang mengeluhkan tentang bagaimana sulitnya mencari judul jurnal yang relevan dengan topik skripsi, terkadang mereka menghabiskan banyak waktu untuk mengandalkan internet hanya untuk mencari referensi yang akan mereka gunakan sebagai sumber sekunder untuk referensi skripsi, namun terkadang hasil yang diharapkan seringkali tidak sesuai dengan kenyataan yang ada.

Semakin banyaknya kebutuhan akan penggunaan layanan secara mobile menyebabkan perusahaan handphone berlomba-lomba dalam menghasilkan handphone yang dapat memenuhi kebutuhan pemakai. Handphone pintar merupakan handphone yang bekerja menggunakan seluruh perangkat lunak sistem operasi yang menyediakan hubungan standar dan mendasar bagi pengembang aplikasi. Sistem operasi android merupakan sistem operasi yang paling pesat perkembangannya saat ini, sistem operasi android yang bersifat open source salah satu faktor pesatnya perkembangan sistem operasi android pada saat ini, dan kebebasan mengembangkan aplikasi, khususnya aplikasi yang bermanfaat buat kegiatan belajar.[1]

Berdasarkan masalah diatas maka penulis menggunakan Algoritma Not So Naive sebagai solusi untuk pencarian jurnal yang ada.

Adapun kelebihan dari penelitian ini yaitu proses pencarian jurnal menggunakan perangkat mobile lebih praktis dan cepat dimana saja mahasiswa membutuhkannya dan kekurangan dari penelitian ini aplikasi harus di update sesuai banyaknya data jurnal yang baru sehingga banyak membutuhkan media penyimpanan di handphone.

2. TEORITIS

2.1 Algoritma Not So Naive

Algoritma *Not So Naive* pertama kali dipublikasikan oleh Christophe Hancart tahun 1992. Algoritma *Not So Naive* merupakan variasi turunan dari algoritma *Naive* atau yang sering disebut dengan algoritma *Brute Force*. Cara kerja algoritma ini adalah dengan memiliki fase pencarian mengecek teks dan pola dari kiri ke kanan. Lalu, algoritma *Not So Naive* akan mengidentifikasi terlebih dahulu dua kasus yang dimana di setiap akhir fase pencocokan pergeseran dapat dilakukan sebanyak 2 posisi ke kanan, tidak seperti algoritma *Naive* yang dimana pergeseran tetaplah sebanyak 1 posisi ke kanan[2].

Misal di asumsikan bahwa $P[0] \neq P[1]$. Jika $P[0] = T[s]$ dan $P[1] = T[s+1]$, maka di akhir fase pencocokan pergeseran s bisa dilakukan sebanyak 2 posisi, karena $P[0] \neq P[1] = T[s+1]$. Dan jika $P[0] = P[1]$. Jika $P[0] = P[s]$ tapi $P[1] \neq T[s+1]$, maka sekali lagi pergeseran s dapat dilakukan sebanyak 2 posisi (Cantone & Faro, 2004) dimana P adalah *Pattern*, T adalah Teks dan s adalah nilai posisi.

3. ANALISA DAN PEMBAHASAN

Sebagai mana yang telah diuraikan di latar belakang masalah, tingkat mobilitas mahasiswa semakin cepat. Mahasiswa sudah memasuki semester akhir umumnya memiliki kesibukan baik di dalam perkuliahan maupun di luar perkuliahan. Mahasiswa tingkat akhir dituntut untuk konsisten terhadap penugasan yang diberikan oleh kampus untuk menunjang proses perkuliahan mereka. Tidak sedikit jumlah mahasiswa di lingkungan kampus yang mengeluhkan tentang kesulitannya mencari judul jurnal yang relevan dengan topik skripsi, mereka menghabiskan banyak waktu di perpustakaan hanya untuk mencari referensi yang akan mereka gunakan sebagai sumber untuk referensi skripsi, sehingga dapat menghasilkan yang diharapkan, tapi tidak sesuai dengan kenyataan yang ada, sehingga dibutuhkan suatu aplikasi agar mahasiswa mudah mencari referensi jurnal-jurnal untuk skripsi dengan memanfaatkan kemudahan dari ponsel atau ponsel yang memiliki sistem operasi android.

Android merupakan platform terobosan terbaru pada perangkat seluler yang sangat digemari, hal ini terkait dengan sistem pada android yang mudah untuk digunakan dengan tampilan yang menarik dan dapat melakukan pembaruan aplikasi secara otomatis, selain itu aplikasi yang dibutuhkan dalam Android gratis yang memungkinkan aplikasi yang diunggah tidak berbayar atau gratis. Perancangan aplikasi mobile jurnal akan menggunakan perangkat lunak IDE Eclipse sebagai editor kode program dan SDK Android (Software Development Kit) yang diperlukan untuk memulai mengembangkan aplikasi pada platform android dengan menggunakan bahasa Java serta Android Development Tools (ADT) sebagai plugin yang dirancang untuk IDE Eclipse yang memberikan kenyamanan dalam mengembangkan aplikasi android dengan menggunakan IDE Eclipse.[3]

3.1 Tahap Pencarian Algoritma Not So Naïve

Pada tabel dibawah ini akan digambarkan rincian algoritma dimana terdapat contoh kasus pattern $P = \text{"METODESAW"}$ dan teks $T = \text{"PEMILIHAN KETUA PRODI DENGAN METODE SAW"}$. Dimana karakter urutan 0 dan karakter urutan 1 pada pattern tidak mengalami kesamaan. Maka nilai variabel k akan diinisialisasi dengan nilai 1 dan nilai variabel ell akan diinisialisasi dengan nilai 2 dimana kedua variabel tersebut akan digunakan untuk pergeseran pada proses pencocokkan.

Tabel 1. Iterasi Algoritma Not So Naive Pertama

I	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
TEKS	P	E	M	I	L	I	H	A	N	K	E	T	U	A	P	R	O	D	I	M	E	T	O	D	E	S	A	W
POLA	M	E	T	O	D	E	S	A	W																			

Pada tabel 1 perbandingan karakter ke -1 dengan pola "E" dengan karakter ke-1 pada teks yaitu "E" mengalami kecocokkan maka selanjutnya dilakukan pengecekan perbandingan karakter ke-2 dengan pola "T" dengan karakter ke ke-2 pada teks yaitu "M" tidak mengalami kecocokkan, maka akan dilakukan percobaan selanjutnya dengan posisi pola digeser sebanyak 1 posisi sesuai dengan nilai variabel k .

Tabel 2. Iterasi Algoritma Not So Naive Kedua

I	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
TEKS	P	E	M	I	L	I	H	A	N	K	E	T	U	A	P	R	O	D	I	M	E	T	O	D	E	S	A	W
POLA		M	E	T	O	D	E	S	A	W																		

Pada tabel 2 perbandingan karakter ke -1 dengan pola "E" dengan karakter ke-2 pada teks yaitu "M" tidak mengalami kecocokkan, maka akan dilakukan percobaan selanjutnya dengan posisi pola digeser sebanyak 1 posisi sesuai dengan nilai variabel k .

Tabel 3. Iterasi Algoritma Not So Naive Ketiga

I	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
TEKS	P	E	M	I	L	I	H	A	N	K	E	T	U	A	P	R	O	D	I	M	E	T	O	D	E	S	A	W
POLA			M	E	T	O	D	E	S	A	W																	

Pada tabel 3 perbandingan karakter ke -1 dengan pola "E" dengan karakter ke-3 pada teks yaitu "T" tidak mengalami kecocokkan, maka akan dilakukan percobaan selanjutnya dengan posisi pola digeser sebanyak 1 posisi sesuai dengan nilai variabel k .

Tabel 4. Iterasi Algoritma Not So Naive Keempat

I	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
TEKS	P	E	M	I	L	I	H	A	N	K	E	T	U	A	P	R	O	D	I	M	E	T	O	D	E	S	A	W
POLA				M	E	T	O	D	E	S	A	W																

Pada tabel 4 perbandingan karakter ke -1 dengan pola”E” dengan karakter ke-4 pada teks yaitu “L” tidak mengalami kecocokkan, maka akan dilakukan percobaan selanjutnya dengan posisi pola digeser sebanyak 1 posisi sesuai dengan nilai variabel k.

Tabel 5. Iterasi Algoritma Not So Naive Kelima

I	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
TEKS	P	E	M	I	L	I	H	A	N	K	E	T	U	A	P	R	O	D	I	M	E	T	O	D	E	S	A	W
POLA					M	E	T	O	D	E	S	A	W															

Pada tabel 5 perbandingan karakter ke -1 dengan pola”E” dengan karakter ke-5 pada teks yaitu “I” tidak mengalami kecocokkan, maka akan dilakukan percobaan selanjutnya dengan posisi pola digeser sebanyak 1 posisi sesuai dengan nilai variabel k.

Tabel 6. Iterasi Algoritma Not So Naive Keeam

I	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
TEKS	P	E	M	I	L	I	H	A	N	K	E	T	U	A	P	R	O	D	I	M	E	T	O	D	E	S	A	W
POLA						M	E	T	O	D	E	S	A	W														

Pada tabel 6 perbandingan karakter ke -1 dengan pola”E” dengan karakter ke-6 pada teks yaitu “H” tidak mengalami kecocokkan, maka akan dilakukan percobaan selanjutnya dengan posisi pola digeser sebanyak 1 posisi sesuai dengan nilai variabel k.

Tabel 7. Iterasi Algoritma Not So Naive Ketujuh

I	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
TEKS	P	E	M	I	L	I	H	A	N	K	E	T	U	A	P	R	O	D	I	M	E	T	O	D	E	S	A	W
POLA							M	E	T	O	D	E	S	A	W													

Pada tabel 7 perbandingan karakter ke -1 dengan pola”E” dengan karakter ke-7 pada teks yaitu “A” tidak mengalami kecocokkan, maka akan dilakukan percobaan selanjutnya dengan posisi pola digeser sebanyak 1 posisi sesuai dengan nilai variabel k.

Tabel 8. Iterasi Algoritma Not So Naive Kedelapan

I	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
TEKS	P	E	M	I	L	I	H	A	N	K	E	T	U	A	P	R	O	D	I	M	E	T	O	D	E	S	A	W
POLA								M	E	T	O	D	E	S	A	W												

Pada tabel 3.8 perbandingan karakter ke -1 dengan pola”E” dengan karakter ke-8 pada teks yaitu “N” tidak mengalami kecocokkan, maka akan dilakukan percobaan selanjutnya dengan posisi pola digeser sebanyak 1 posisi sesuai dengan nilai variabel k.

Tabel 9. Iterasi Algoritma Not So Naive Kesembilan

I	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
TEKS	P	E	M	I	L	I	H	A	N	K	E	T	U	A	P	R	O	D	I	M	E	T	O	D	E	S	A	W
POLA									M	E	T	O	D	E	S	A	W											

Pada tabel 3.9 perbandingan karakter ke -1 dengan pola”E” dengan karakter ke-9 pada teks yaitu “K” tidak mengalami kecocokkan, maka akan dilakukan percobaan selanjutnya dengan posisi pola digeser sebanyak 1 posisi sesuai dengan nilai variabel k.

Tabel 10. Iterasi Algoritma Not So Naive Kesepuluh

I	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
TEKS	P	E	M	I	L	I	H	A	N	K	E	T	U	A	P	R	O	D	I	M	E	T	O	D	E	S	A	W
POLA										M	E	T	O	D	E	S	A	W										

Pada tabel 3.10 perbandingan karakter ke -1 dengan pola “E” dengan karakter ke-10 pada teks yaitu “E” mengalami kecocokkan maka selanjutnya dilakukan pengecekan perbandingan karakter ke-2 dengan pola”T” dengan karakter ke ke-11 pada teks yaitu”T” mengalami kecocokkan maka selanjutnya dilakukan pengecekan perbandingan karakter ke-3 dengan pola”O” dengan karakter ke ke-12 pada teks yaitu”U” tidak mengalami kecocokkan , maka akan dilakukan percobaan selanjutnya dengan posisi pola digeser sebanyak 1 posisi sesuai dengan nilai variabel k.

Tabel 11. Iterasi Algoritma Not So Naive Kesebelas

I	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
TEKS	P	E	M	I	L	I	H	A	N	K	E	T	U	A	P	R	O	D	I	M	E	T	O	D	E	S	A	W
POLA												M	E	T	O	D	E	S	A	W								

Pada tabel 11 perbandingan karakter ke -1 dengan pola”E” dengan karakter ke-11 pada teks yaitu “T” tidak mengalami kecocokkan, maka akan dilakukan percobaan selanjutnya dengan posisi pola digeser sebanyak 1 posisi sesuai dengan nilai variabel k.

Tabel 12. Iterasi Algoritma Not So Naive Keduabelas

I	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
TEKS	P	E	M	I	L	I	H	A	N	K	E	T	U	A	P	R	O	D	I	M	E	T	O	D	E	S	A	W
POLA												M	E	T	O	D	E	S	A	W								

Pada tabel 12 perbandingan karakter ke -1 dengan pola”E” dengan karakter ke-12 pada teks yaitu “U” tidak mengalami kecocokkan, maka akan dilakukan percobaan selanjutnya dengan posisi pola digeser sebanyak 1 posisi sesuai dengan nilai variabel k.

Tabel 13. Iterasi Algoritma Not So Naive Ketigabelas

I	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
TEKS	P	E	M	I	L	I	H	A	N	K	E	T	U	A	P	R	O	D	I	M	E	T	O	D	E	S	A	W
POLA													M	E	T	O	D	E	S	A	W							

Pada tabel 3.13 perbandingan karakter ke -1 dengan pola”E” dengan karakter ke-13 pada teks yaitu “A” tidak mengalami kecocokkan, maka akan dilakukan percobaan selanjutnya dengan posisi pola digeser sebanyak 1 posisi sesuai dengan nilai variabel k.

Tabel 14. Iterasi Algoritma Not So Naive Ketigabelas

I	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
TEKS	P	E	M	I	L	I	H	A	N	K	E	T	U	A	P	R	O	D	I	M	E	T	O	D	E	S	A	W
POLA														M	E	T	O	D	E	S	A	W						

Pada tabel 3.14 perbandingan karakter ke -1 dengan pola”E” dengan karakter ke-14 pada teks yaitu “P” tidak mengalami kecocokkan, maka akan dilakukan percobaan selanjutnya dengan posisi pola digeser sebanyak 1 posisi sesuai dengan nilai variabel k.

Tabel 15. Iterasi Algoritma Not So Naive Kelimabelas

I	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
TEKS	P	E	M	I	L	I	H	A	N	K	E	T	U	A	P	R	O	D	I	M	E	T	O	D	E	S	A	W
POLA															M	E	T	O	D	E	S	A	W					

Pada tabel 3.15 perbandingan karakter ke -1 dengan pola”E” dengan karakter ke-14 pada teks yaitu “P” tidak mengalami kecocokkan, maka akan dilakukan percobaan selanjutnya dengan posisi pola digeser sebanyak 1 posisi sesuai dengan nilai variabel k.

Tabel 16. Iterasi Algoritma Not So Naive Keenamabelas

I	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
TEKS	P	E	M	I	L	I	H	A	N	K	E	T	U	A	P	R	O	D	I	M	E	T	O	D	E	S	A	W
POLA																M	E	T	O	D	E	S	A	W				

Pada tabel 3.16 perbandingan karakter ke -1 dengan pola”E” dengan karakter ke-16 pada teks yaitu “O” tidak mengalami kecocokkan, maka akan dilakukan percobaan selanjutnya dengan posisi pola digeser sebanyak 1 posisi sesuai dengan nilai variabel k.

Tabel 17. Iterasi Algoritma Not So Naive Ketujuhbelas

I	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
TEKS	P	E	M	I	L	I	H	A	N	K	E	T	U	A	P	R	O	D	I	M	E	T	O	D	E	S	A	W
POLA																	M	E	T	O	D	E	S	A	W			

Pada tabel 3.17 perbandingan karakter ke -1 dengan pola”E” dengan karakter ke-17 pada teks yaitu “D” tidak mengalami kecocokkan, maka akan dilakukan percobaan selanjutnya dengan posisi pola digeser sebanyak 1 posisi sesuai dengan nilai variabel k.

Tabel 18. Iterasi Algoritma Not So Naive Kedelapanbelas

I	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
TEKS	P	E	M	I	L	I	H	A	N	K	E	T	U	A	P	R	O	D	I	M	E	T	O	D	E	S	A	W
POLA																		M	E	T	O	D	E	S	A	W		

Pada tabel 3.18 perbandingan karakter ke -1 dengan pola”E” dengan karakter ke-18 pada teks yaitu “I” tidak mengalami kecocokkan, maka akan dilakukan percobaan selanjutnya dengan posisi pola digeser sebanyak 1 posisi sesuai dengan nilai variabel k.

Tabel 19. Iterasi Algoritma Not So Naive Kesembilan belas

I	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
TEKS	P	E	M	I	L	I	H	A	N	K	E	T	U	A	P	R	O	D	I	M	E	T	O	D	E	S	A	W
POLA																			M	E	T	O	D	E	S	A	W	

Pada tabel 3.19 perbandingan karakter ke -1 dengan pola”E” dengan karakter ke-18 pada teks yaitu “I” tidak mengalami kecocokkan, maka akan dilakukan percobaan selanjutnya dengan posisi pola digeser sebanyak 1 posisi sesuai dengan nilai variabel k.

Tabel 20. Iterasi Algoritma Not So Naive Keduapuluh

I	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
TEKS	P	E	M	I	L	I	H	A	N	K	E	T	U	A	P	R	O	D	I	M	E	T	O	D	E	S	A	W
POLA																				M	E	T	O	D	E	S	A	W

Pada tabel 3.20 perbandingan karakter ke-1 dengan pola “E” dengan karakter ke 20 pada teks yaitu “E” mengalami kecocokkan, maka pencarian dihentikan sampai disini.

4. KESIMPULAN

1. Algoritma Tidak Begitu Naif dapat diimplementasikan pada pencarian jurnal berbasis Mobile
2. Algoritma Not So Naïve tidak dapat mencari 1 huruf saja. Karena pada saat pra-proses Algoritma ini membutuhkan minimal 2 huruf awal yang sesuai untuk menentukan nilai variabel k dan ell pada proses pencarian.

REFERENCES

- [1] S. T. O. Santoso, “Perkembangan Teknologi Handphone,” *Ilmuti*, 2014.
- [2] G. I. Webb, J. R. Boughton, and Z. Wang, “Not so naive Bayes: Aggregating one-dependence estimators,” *Mach. Learn.*, 2005.
- [3] D. Louis, P. Müller, D. Louis, and P. Müller, “Android,” in *Android*, 2016.