

MAKALAH KECERDASAN BUATAN
KECERDASAN BUATAN DAN MESIN TURING

Dosen : Dani Rohpandi, M. Kom



Disusun Oleh :

Amelia Chisa	(22110451)
Tian Sanjaya Junaedi	(22110399)
Aldika	(22110450)
Mely Amalia Hastuti	(22110440)

PROGRAM STUDI TEKNIK INFORMATIKA

STMIK MARDIRA INDONESIA

BANDUNG

2025

KATA PENGANTAR

Penulis berterima kasih kepada Tuhan Yang Maha Esa atas rahmat dan karunia-Nya sehingga makalah yang berjudul "Kecerdasan Buatan" ini disusun dan diselesaikan dengan baik. Tujuan dari tugas mata kuliah adalah untuk meningkatkan pemahaman siswa tentang konsep, perkembangan, dan penerapan kecerdasan buatan (AI) dalam berbagai bidang kehidupan. Makalah ini merupakan bagian dari tugas tersebut.

Kecerdasan buatan telah menjadi salah satu teknologi yang paling inovatif dan berpengaruh di era digital saat ini. AI telah memasuki banyak hal, seperti industri, pendidikan, kesehatan, dan kehidupan sehari-hari. Oleh karena itu, sangat penting untuk memahami konsep dasar AI, jenisnya, dan implikasinya, khususnya bagi generasi muda yang akan memainkan peran penting dalam pengembangan teknologi masa depan.

Penulis mengucapkan terima kasih yang sebesar-besarnya kepada dosen pembimbing, rekan-rekan kelompok, dan semua orang yang telah membantu dengan membuat makalah ini menjadi mungkin. Selain itu, penulis menyadari bahwa makalah ini masih jauh dari sempurna. Oleh karena itu, penulis sangat mengharapkan kritik dan saran yang membangun untuk membantu mereka memperbaikinya di masa mendatang. Semoga pembaca mendapatkan manfaat dan wawasan tambahan dari makalah ini. Selain itu, semoga makalah ini sedikit membantu kemajuan pengetahuan di bidang kecerdasan buatan.

Penulis,

7 April 2024

DAFTAR ISI

KATA PENGANTAR	II
DAFTAR ISI	III
BAB I	1
PENDAHULUAN	1
A. Latar Belakang	1
B. Rumusan Masalah	1
C. Tujuan Penulisan	2
BAB II	3
LANDASAN TEORI	3
A. Expert System	3
B. Natural Language Interface to Database	4
C. Software Agent (Khusus Web)	5
D. Fuzzy System	7
E. Web Data Mining	9
F. Mesin Turing	12
G. Sejarah AI	16
BAB III	20
PENUTUP	20
A. Kesimpulan	20
B. Saran	20
REFERENSI	22

BAB I

PENDAHULUAN

A. Latar Belakang

Berbagai inovasi yang mengubah kehidupan dan pekerjaan manusia telah muncul sebagai hasil dari pertumbuhan yang begitu pesat dalam teknologi informasi dan komunikasi. Kecerdasan buatan (AI) adalah salah satu inovasi teknologi terpenting dalam sepuluh tahun terakhir. Teknologi ini diciptakan untuk meniru cara manusia berpikir dan bertindak, yang memungkinkan mesin untuk belajar, membuat keputusan, dan menyelesaikan masalah secara mandiri.

Meskipun konsep kecerdasan buatan pertama kali muncul pada pertengahan abad ke-20, penggunaan pertamanya baru menunjukkan kemajuan besar dalam era komputer kontemporer. Saat ini, AI telah digunakan di banyak bidang, termasuk perbankan, transportasi, pendidikan, keamanan, dan bahkan hiburan. Asisten virtual, mobil otonom, chatbot, dan sistem rekomendasi hanyalah beberapa contoh nyata bagaimana AI digunakan dalam kehidupan sehari-hari.

Oleh karena itu, penting bagi mahasiswa dan masyarakat umum untuk memahami dasar-dasar kecerdasan buatan, baik secara teori maupun praktik, karena kehadiran AI tidak hanya membawa kemudahan tetapi juga tantangan baru dari segi moral, hukum, dan sosial. Diharapkan kita dapat menggunakan teknologi ini dengan bijak dan bertanggung jawab jika kita memahaminya dengan benar.

Diharapkan makalah ini akan memberikan gambaran komprehensif tentang kecerdasan buatan dan mendorong pembaca untuk terus belajar tentang kemajuan teknologi ini. Ini membahas tentang apa itu kecerdasan buatan, sejarahnya, jenisnya, cara kerjanya, dan penerapannya dalam kehidupan nyata.

B. Rumusan Masalah

Berdasarkan latar belakang tersebut, rumusan masalah dalam makalah ini adalah sebagai berikut :

- 1) Apa arti dan contoh konsep kecerdasan buatan (AI) yang berbeda, seperti fuzzy system, expert system, web data mining, dan sebagainya?

- 2) Bagaimana Mesin Turing berfungsi dan apa hubungannya dengan AI?
- 3) Bagaimana timeline yang diberikan dalam gambar menunjukkan perkembangan AI?

C. Tujuan Penulisan

Tujuan penulisan makalah ini adalah :

- 1) Memberikan penjelasan menyeluruh tentang berbagai konsep AI, seperti definisi, mekanisme kerja, dan contoh penggunaan Expert Systems, Fuzzy Systems, Web Data Mining, dan lainnya.
- 2) Memahami cara Mesin Turing bekerja, serta bagian-bagiannya, dan bagaimana ia memproses data.
- 3) Menganalisis perkembangan AI dari masa ke masa berdasarkan timeline yang diberikan dalam gambar, termasuk tonggak-tonggak penting dan tren yang muncul.

BAB II

LANDASAN TEORI

A. Expert System

Sistem pakar, juga dikenal sebagai sistem ahli, adalah program komputer yang dimaksudkan untuk meniru kemampuan pengambilan keputusan seorang ahli dalam menyelesaikan masalah yang kompleks. Untuk memberikan solusi atau saran kepada pengguna, sistem ini menggunakan aturan penalaran dan mengumpulkan informasi dari basis pengetahuannya. Sistem pakar, yang pertama kali dibuat pada tahun 1970, adalah salah satu contoh awal penggunaan kecerdasan buatan (AI) untuk memecahkan masalah dalam bidang tertentu, seperti kedokteran, teknik, dan sains.

a) Karakteristik Expert System

Sistem pakar memiliki beberapa karakteristik yang membedakannya dari sistem berbasis aturan biasa. Karakteristik ini memastikan bahwa sistem pakar mampu menangani masalah kompleks dengan tingkat akurasi dan efisiensi tinggi, memberikan solusi yang dapat dimengerti oleh pengguna, serta memastikan hasil yang konsisten dan andal.

1. Memiliki kemampuan untuk menyelesaikan masalah kompleks dengan akurasi dan efisiensi yang tinggi.
2. Memberikan hasil yang dapat dipahami oleh pengguna dalam bahasa manusia.
3. Memberikan hasil yang dapat diandalkan dan konsisten.
4. Memberikan hasil cepat untuk pertanyaan kompleks.

b) Komponen Utama

Sistem pakar memiliki 3 komponen utama :

1. Basis Pengetahuan (Knowledge Base): Menyimpan informasi dan aturan yang diperlukan untuk pengambilan keputusan.
2. Mesin Inferensi (Inference Engine): Menganalisis data dan menarik kesimpulan berdasarkan aturan yang ada dalam basis pengetahuan.

3. Antarmuka Pengguna (User Interface): Media komunikasi antara pengguna dan sistem, memungkinkan pengguna untuk memasukkan data dan menerima output.

B. Natural Language Interface to Database

Sistem Natural Language Interface to Database (NLIDB) adalah bagian dari proses pengolahan bahasa alami (NLP) yang bertujuan untuk menjembatani kesenjangan antara pengguna non-teknis dan basis data. Memungkinkan pengguna berinteraksi dengan basis data menggunakan bahasa alami, seperti bahasa Inggris, tanpa perlu memahami struktur atau sintaksis pertanyaan formal, seperti SQL.

a) Komponen Pembentuk NLDIB

Beberapa komponen utama membentuk sistem NLIDB:

1. Memecah pertanyaan dalam bahasa alami menjadi token khusus yang dapat diproses lebih lanjut.
2. Memeriksa tata bahasa pengguna untuk memastikan bahwa inputnya valid.
3. Mengubah representasi internal pertanyaan menjadi query SQL yang dapat dijalankan.
4. Mengumpulkan data dari basis data dengan menggunakan query yang dibuat.

b) Arsitektur NLDIB

Arsitektur NLIDB biasanya terdiri dari beberapa tahap berikut:

1. Pengguna dapat memasukkan pertanyaan ke dalam sistem dalam bahasa alami.
2. Untuk memahami pertanyaan, sistem melakukan tokenisasi, analisis sintaksis, dan semantik.
3. Sistem menghasilkan query SQL yang sesuai berdasarkan pemahaman input.

4. Pertanyaan dikirim ke basis data untuk mendapatkan informasi yang diperlukan.
5. Hasil eksekusi pertanyaan disajikan kembali kepada pengguna dalam format yang mudah dipahami.

c) Tantangan Dalam Perkembangan NLDIB

1. Bahasa alami sering ambigu, sehingga sulit bagi sistem untuk memahami maksud sebenarnya dari pertanyaan tanpa konteks tambahan.
2. Banyak sistem NLIDB dibuat untuk digunakan dalam domain tertentu; oleh karena itu, mereka kurang fleksibel jika digunakan di luar domain tersebut.
3. Sistem mungkin kesulitan memahami konteks atau detail tertentu dalam pertanyaan yang kompleks, yang dapat menyebabkan hasil yang tidak akurat.

Beberapa komponen utama membentuk sistem NLIDB:

1. Memecah pertanyaan dalam bahasa alami menjadi token khusus yang dapat diproses lebih lanjut.
2. Memeriksa tata bahasa pengguna untuk memastikan bahwa inputnya valid.
3. Mengubah representasi internal pertanyaan menjadi query SQL yang dapat dijalankan.
4. Mengumpulkan data dari basis data dengan menggunakan query yang dibuat

C. Software Agent (Khusus Web)

Software Agent (agen perangkat lunak) adalah program komputer yang dapat beroperasi secara mandiri, mengambil keputusan, dan menjalankan tugas dalam lingkungan tertentu, terutama dalam platform web. Agen ini dirancang untuk beroperasi secara berkelanjutan tanpa perlu intervensi manusia secara langsung. Dalam konteks web, software agent memanfaatkan infrastruktur internet untuk berbagai keperluan,

seperti pengumpulan data otomatis, analisis tren, pengelolaan informasi, dan optimasi layanan digital.

Keunggulan software agent dalam web adalah kemampuannya untuk menyesuaikan diri terhadap perubahan lingkungan, memproses data dalam jumlah besar dengan cepat, serta memberikan hasil yang lebih efisien dibandingkan interaksi manual. Contoh umum dari software agent dalam web adalah *web crawler*, chatbot berbasis kecerdasan buatan, dan agen pemantau harga e-commerce.

a) Karakteristik Software Agent

Software agent memiliki beberapa karakteristik utama yang membedakannya dari program komputer biasa. Karakteristik ini memungkinkan agen beroperasi dengan lebih fleksibel dan efisien dalam ekosistem web.

1. Autonomi Software agent dapat beroperasi secara mandiri tanpa harus menerima instruksi langsung dari pengguna setiap saat.
2. Reaktivitas Agen memiliki kemampuan untuk merespons perubahan lingkungan secara real-time.
3. Proaktivitas Selain merespons input, software agent juga dapat mengambil inisiatif untuk mencapai tujuan tertentu.
4. Kemampuan Sosial Beberapa agen dapat berkolaborasi dengan agen lain atau sistem yang lebih besar untuk menyelesaikan tugas yang kompleks.

b) Arsitektur Software Agent

Untuk dapat berfungsi dengan optimal dalam lingkungan web, software agent memerlukan arsitektur yang sesuai dengan tugas yang akan dijalankan. Berikut adalah beberapa arsitektur utama yang sering digunakan dalam pengembangan agen web:

1. Reaktif: Agen ini hanya merespons stimulus lingkungan tanpa memiliki model internal dan cocok untuk tugas sederhana yang memerlukan respons cepat.
2. Deliberatif (BDI): Menggunakan model simbolik (*Belief-Desire-Intention*) untuk perencanaan dan pengambilan keputusan serta memiliki model internal yang memungkinkan pengambilan keputusan lebih kompleks.

3. Hibrid: Menggabungkan pendekatan reaktif dan deliberatif untuk keseimbangan antara kecepatan dan kompleksitas, cocok untuk aplikasi yang membutuhkan fleksibilitas tinggi.

c) Implementasi Software Agent

Software agent telah digunakan dalam berbagai aplikasi web untuk meningkatkan efisiensi dan kinerja sistem digital. Berikut adalah beberapa contoh implementasinya:

1. Web Crawler/Spider: Agen ini bertugas untuk mengindeks konten web agar mesin pencari seperti Google dapat memberikan hasil pencarian yang relevan.
2. Chatbot: Memberikan respons otomatis terhadap pertanyaan pengguna di situs e-commerce atau layanan pelanggan.
3. Agen Monitoring: Memantau harga produk atau ketersediaan barang di berbagai platform e-commerce untuk memberikan notifikasi kepada pengguna.

D. Fuzzy System

Sistem kendali fuzzy adalah sistem kendali yang berbasis pada logika fuzzy – sistem matematika yang menganalisis nilai masukan analog dalam bentuk variabel logika yang memiliki nilai kontinu antara 0 dan 1, berbeda dengan logika klasik atau digital yang beroperasi pada nilai diskrit baik 1 atau 0 (benar atau salah).

Logika fuzzy digunakan secara luas dalam kontrol mesin. Istilah "fuzzy" mengacu pada fakta bahwa logika yang terlibat dapat menangani konsep yang tidak dapat dinyatakan sebagai "benar" atau "salah" tetapi sebagai "sebagian benar". Meskipun pendekatan alternatif seperti algoritma genetika dan jaringan saraf dapat bekerja sama baiknya dengan logika fuzzy dalam banyak kasus, logika fuzzy memiliki keuntungan bahwa solusi untuk masalah tersebut dapat dituangkan dalam istilah yang dapat dipahami oleh operator manusia, sehingga pengalaman mereka dapat digunakan dalam desain pengontrol. Hal ini memudahkan mekanisasi tugas-tugas yang telah berhasil dilakukan oleh manusia

Logika fuzzy pertama kali diusulkan oleh Lotfi A. Zadeh pada tahun 1965 dan dikembangkan lebih lanjut pada 1973 dengan konsep "variabel linguistik". Aplikasi industri pertama logika fuzzy muncul pada tahun 1976 dalam pengendalian kiln semen di Denmark.

Pada 1980-an, Jepang menjadi pionir dalam penerapan logika fuzzy. Seiji Yasunobu dan Soji Miyamoto dari Hitachi berhasil mengaplikasikannya dalam sistem kendali Subway Sendai (1987). Takeshi Yamakawa mendemonstrasikan kontrol fuzzy dalam eksperimen bandul terbalik, yang kemudian dikembangkan lebih lanjut dalam berbagai aplikasi industri.

Pada 1988, Jepang mendirikan Laboratory for International Fuzzy Engineering (LIFE), yang melibatkan 48 perusahaan, termasuk Volkswagen. Sejak itu, berbagai produk konsumen Jepang mulai menggunakan logika fuzzy, seperti penyedot debu Matsushita, mesin cuci Hitachi, kamera Canon, serta AC Mitsubishi, yang meningkatkan efisiensi energi dan performa.

Di Amerika dan Eropa, logika fuzzy diterapkan dalam industri otomotif, peralatan rumah tangga, dan eksplorasi ruang angkasa. NASA meneliti kontrol fuzzy untuk dok otomatis, sementara perusahaan seperti Boeing, General Motors, dan Whirlpool menggunakannya dalam efisiensi energi dan peningkatan transmisi otomotif. Pada 1995, Maytag memperkenalkan mesin pencuci piring berbasis fuzzy untuk optimasi siklus pencucian.

Sistem kontrol fuzzy cocok digunakan ketika kompleksitas proses tinggi termasuk ketidakpastian dan perilaku nonlinier, dan tidak ada model matematika yang tepat yang tersedia. Aplikasi sistem kontrol fuzzy yang berhasil telah dilaporkan di seluruh dunia terutama di Jepang dengan solusi perintis sejak tahun 80-an.

Beberapa aplikasi yang dilaporkan dalam literatur adalah:

1. Pendingin udara
2. Sistem fokus otomatis di kamera
3. Peralatan rumah tangga (kulkas, mesin cuci...)
4. Kontrol dan optimasi proses dan sistem industri
5. Sistem penulisan

6. Efisiensi bahan bakar pada mesin
7. Lingkungan Hidup
8. Sistem pakar
9. Pohon keputusan
10. Robotika
11. Kendaraan otonom

Saat ini, logika fuzzy terus dikembangkan dalam perangkat lunak, sistem pakar, kecerdasan buatan, dan integrasi dengan jaringan saraf untuk membangun sistem adaptif yang dapat belajar sendiri.

E. Web Data Mining

Penggalian data (*data mining*) adalah ekstraksi pola yang menarik dari data dalam jumlah besar. Suatu pola dikatakan menarik apabila pola tersebut tidak sepele, implisit, tidak diketahui sebelumnya, dan berguna. Pola yang disajikan haruslah mudah dipahami, berlaku untuk data yang akan diprediksi dengan derajat kepastian tertentu, berguna, dan baru. Penggalian data memiliki beberapa nama alternatif, meskipun definisi eksaknya berbeda, seperti KDD (*knowledge discovery in database*), analisis pola, arkeologi data, pemanenan informasi, dan intelegensia bisnis. Penggalian data diperlukan saat data yang tersedia terlalu banyak (misalnya data yang diperoleh dari sistem basis data perusahaan, e-commerce, data saham, data sensus dan data bioinformatika), tetapi tidak tahu pola apa yang bisa didapatkan.

1. Proses Pencarian Pola

Penggalian data adalah salah satu bagian dari proses pencarian pola. Berikut ini urutan proses pencarian pola:

- 1) Pembersihan Data: yaitu menghapus data pengganggu (*noise*) dan mengisi data yang hilang.
- 2) Integrasi Data: yaitu menggabungkan berbagai sumber data.
- 3) Pemilihan Data: yaitu memilih data yang relevan.
- 4) Transformasi Data: yaitu mentransformasi data ke dalam format untuk diproses dalam penggalian data.

- 5) Penggalian Data: yaitu menerapkan metode cerdas untuk ekstraksi pola.
- 6) Evaluasi pola: yaitu mengenali pola-pola yang menarik saja.
- 7) Penyajian pola: yaitu memvisualisasi pola ke pengguna.

Perkembangan yang pesat di bidang pengumpulan data dan teknologi penyimpanan di berbagai bidang, menghasilkan basis data yang terlampau besar. Namun, data yang dikumpulkan jarang dilihat lagi karena terlalu panjang, membosankan, dan tidak menarik. Seringkali, keputusan -yang katanya berdasarkan data- dibuat tidak lagi berdasarkan data, melainkan dari intuisi para pembuat keputusan, sehingga lahirlah cabang ilmu penggalian data ini.

Analisis data tanpa menggunakan otomasi dari penggalian data adalah tidak memungkinkan lagi, kalau 1) data terlalu banyak, 2) dimensionalitas data terlalu besar, 3) data terlalu kompleks untuk dianalisis manual (misalnya: data time series, data spatiotemporal, data multimedia, data streams).

2. Teknik Penggalian Data

Pada dasarnya penggalian data dibedakan menjadi dua fungsionalitas, yaitu deskripsi dan prediksi. Berikut ini beberapa fungsionalitas penggalian data yang sering digunakan:

- 1) Karakterisasi dan Diskriminasi: yaitu menggeneralisasi, merangkum, dan mengkontraskan karakteristik data.
- 2) Penggalian pola berulang: yaitu pencarian pola asosiasi (*association rule*) atau pola intra-transaksi, atau pola pembelian yang terjadi dalam satu kali transaksi.
- 3) Klasifikasi: yaitu membangun suatu model yang bisa mengklasifikasikan suatu objek berdasar atribut-atributnya. Kelas target sudah tersedia dalam data sebelumnya, sehingga fokusnya adalah bagaimana mempelajari data yang ada agar klasifikator bisa mengklasifikasikan sendiri.
- 4) Prediksi: yaitu memprediksi nilai yang tidak diketahui atau nilai yang hilang, menggunakan model dari klasifikasi.
- 5) Penggugusan/Cluster analysis: yaitu mengelompokkan sekumpulan objek data berdasarkan kemiripannya. Kelas target tidak tersedia dalam data sebelumnya,

sehingga fokusnya adalah memaksimalkan kemiripan intrakelas dan meminimalkan kemiripan antarkelas.

- 6) Analisis outlier: yaitu proses pengenalan data yang tidak sesuai dengan perilaku umum dari data lainnya. Contoh: mengenali *noise* dan pengecualian dalam data.
- 7) Analisis trend dan evolusi: meliputi analisis regresi, penggalian pola sekuensial, analisis periodisitas, dan analisis berbasis kemiripan.

Berikut ini adalah 10 algoritme penggalian data yang paling populer berdasarkan konferensi ICDM '06, semua algoritme dinominasikan oleh para pemenang ACM KDD Innovation Award dan IEEE ICDM Research Contributions Award:

1. C4.5 (61 suara)
2. k-Means (60 suara)
3. SVM atau *Support Vector Machine* (58 suara)
4. Apriori (52 suara)
5. EM (48 suara)
6. PageRank (46 suara)
7. AdaBoost (45 suara)
8. kNN (45 suara)
9. Naive Bayes (34 suara)

Berikut ini adalah yang hanya masuk nominasi:

1. CART
2. FP-Tree
3. HITS
4. BIRCH
5. GSP
6. PrefixSpan
7. CBA

8. Finding Reduct

9. gSpan

F. Mesin Turing

Mesin Turing adalah model komputasi teoretis yang ditemukan oleh Alan Turing, berfungsi sebagai model ideal untuk melakukan perhitungan matematis. Walaupun model ideal ini diperkenalkan sebelum komputer nyata dibangun, model ini tetap diterima kalangan ilmu komputer sebagai model komputer yang sesuai untuk menentukan apakah suatu fungsi dapat diselesaikan oleh komputer atau tidak (menentukan *computable function*). Mesin Turing terkenal dengan ungkapan "Apapun yang bisa dilakukan oleh Mesin Turing pasti bisa dilakukan oleh komputer."

Sebuah mesin turing terdiri atas barisan sel tersusun berupa pita yang dapat bergerak maju mundur, komponen aktif baca/tulis pita yang memiliki status perhitungan serta dapat mengubah/menulisi sel aktif yang ada di pita tadi, dan suatu kumpulan instruksi bagaimana komponen baca/tulis ini harus melakukan modifikasi terhadap sel aktif pada pita, serta bagaimana menggerakkan pita tersebut. Pada setiap langkah dalam komputasi, mesin ini akan dapat mengubah isi dari sel yang aktif, mengubah status dari komponen baca/tulis, dan mengubah posisi pita ke kiri atau ke kanan.

Jauh sebelum lahirnya program komputer, Alan Turing pada tahun 1936 telah mengeluarkan gagasannya berupa model mesin abstrak sebagai alat mekanik untuk mengerjakan prosedur yang efektif. Model ini disebut Mesin Turing.

Mesin turing dapat diadaptasi untuk mensimulasi logika dari setiap algoritma oleh karena itu cara kerja mesin turing adalah ekivalen dengan cara kerja komputer sekarang ini dan mesin turing juga ekivalen dengan problema komputasi matematika. Mesin turing tidak ditujukan sebagai teknologi komputasi praktis tetapi lebih sebagai eksperimen pemikiran yang mewakili sebuah mesin komputasi. Mesin turing membantu para ilmuwan komputer memahami batas-batas komputasi mekanis.

Sebagai input dari mesin turing adalah kata atau untai atas suatu alfabet T . Mesin turing berhenti dengan keadaan menerima atau menolak untai. Kadang-kadang terjadi pula perulangan atau looping tak terhingga.

Keterangan:

- 1) Tape: Tempat diletakkannya inputan yang berupa kata/untai.
- 2) Head: membaca dan menulisi sel pita mesin turing, bisa bergerak ke kiri atau ke kanan.
- 3) Finite StateControl (FSC): otak dari TM, diimplementasikan dari algoritma pengenalan kalimat.

1. Definisi Mesin Turing

Mesin turing didefinisikan sebagai 7 tuple $M = \{ Q, S, G, S, F, B, \Delta \}$

Q: himpunan hingga state,

S: alfabet input,

G: simbol pada pita (meliputi pula blank)

S: state awal, $S \in Q$

B: simbol kosong (blank) (bukan bagian dari S)

Δ : fungsi transisi

F: state akhir, $F \in Q$

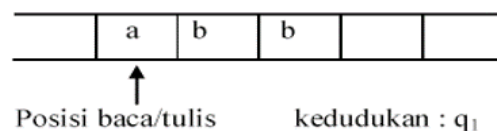
2. Gerakan Mesin Turing

Gerakan mesin turing diwakili oleh fungsi transisi:

$\Delta(q_i, a) = (q_j, b, X)$: Mesin kedudukan q_i membaca simbol masukan a,

gerakan: mesin berubah ke status q_j , menulis b dan posisi baca / tulis bergerak X (berupa R=gerak kekanan atau L=gerak kekiri).

Untuk gerakan fungsi transisi $\Delta(q_1, a) = (q_3, b, R)$ artinya:

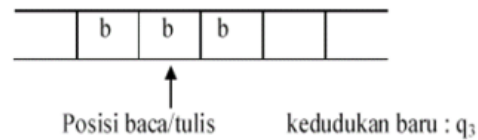


II. Dimiliki mesin turing dengan definisi $M = \{ Q, S, G, S, F, B, \Delta \}$

II. Dimiliki mesin turing dengan definisi $M = \{ Q, S, G, S, F, B, \Delta \}$

Setelah membaca simbol a, kedudukan mesin = q_3 , menulis b dan bergerak kekanan.

Hasil sebagai berikut :



$$Q = \{q_1, q_2\}$$

$$S = \{a, b\}$$

$$G = \{a, b, B\}$$

$$S = \{q_1\}$$

$$F = \{q_2\}$$

$$\Delta: \Delta(q_1, a) = (q_1, a, R)$$

3. Contoh Mesin Turing Sederhana

Sebuah contoh mesin Turing dapat dibangun untuk melakukan komputasi sederhana yang didefinisikan seperti ini:

Tentukan ada berapa angka 1 dalam sebuah string berbentuk 0111...110 (rangkaian angka 1 yang didahului dengan 0 dan diakhiri juga dengan 0), apakah berjumlah genap atau berjumlah ganjil.

- 1) Jika angka 1 di antara dua angka 0 berjumlah genap, tulis sebuah angka 0 pada salah satu sel dari tape mesin Turing.
- 2) Jika angka 1 di antara dua angka 0 berjumlah ganjil, tulis sebuah angka 1 pada salah satu sel dari tape mesin Turing.

Untuk menyelesaikan masalah komputasi ini, kita buat tiga buah State bagi mesin Turing ini, yaitu Start, Even, dan Odd. Di samping itu kita buat sekumpulan aturan Transisi yang digunakan oleh mesin Turing ini untuk melakukan proses komputasinya. Aturan-aturan Transisi tersebut dapat dituliskan demikian:

- 1) Jika mesin Turing berada pada status Start, dan membaca simbol 0 pada Tape, lakukan hal berikut: Pindah status menjadi status Even, Ganti simbol 0 pada Tape dengan Blank (atau Hapus simbol 0 pada Tape), dan Bergerak ke kanan satu sel.
- 2) Jika mesin Turing berada pada status Even, dan membaca simbol 1 pada Tape, lakukan hal berikut: Pindah status menjadi status Odd, Ganti simbol 1 pada Tape dengan Blank, dan Bergerak ke kanan satu sel.

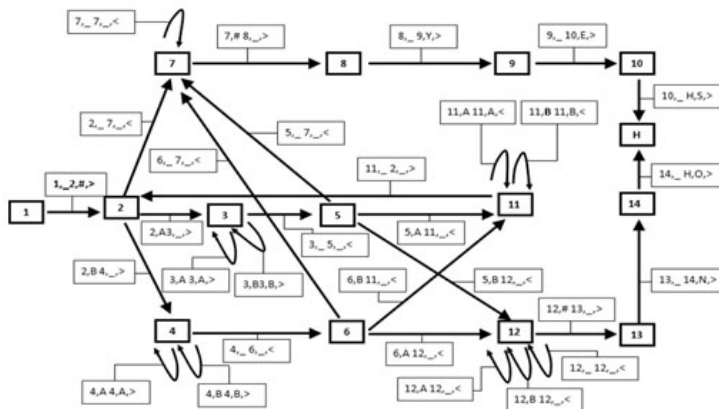
Kelas Bahasa	Mesin Pengenal Bahasa
<i>Unrestricted Grammar (UG)</i>	Mesin Turing (<i>Turing Machine</i>), TM
<i>Context Sensitive Grammar (CSG)</i>	<i>Linear Bounded Automaton</i> , LBA
<i>Context Free Grammar (CFG)</i>	Automata Pushdown (<i>Pushdown Automata</i>), PDA
<i>Regular Grammar, RG</i>	Automata Hingga (<i>Finite Automata</i>), FA

- 3) Jika mesin Turing berada pada status Odd, dan membaca simbol 1 pada Tape, lakukan hal berikut: Pindah status menjadi Even, Ganti simbol 1 pada Tape dengan Blank, dan Bergerak ke kanan satu sel.
- 4) Jika mesin Turing berada pada status Even, dan membaca simbol 0 pada Tape, lakukan hal berikut: Pindah status menjadi Halt, Ganti simbol 0 pada Tape dengan 0, dan tetap pada sel tersebut (tidak perlu berpindah ke kiri maupun ke kanan).
- 5) Jika mesin Turing berada pada status Odd, dan membaca simbol 0 pada Tape, lakukan hal berikut: Pindah status menjadi Halt, Ganti simbol 0 pada Tape dengan 1, dan tetap pada sel tersebut.

Palindrome itu adalah berasal dari bahasa Yunani yaitu Palindromos A Palindrome. Palindromos A Palindrome adalah kata atau kalimat yang sama dieja maju atau mundur(bacaan yang sama dieja pada kedua arah). Sebagai contoh sederhana adalah beberapa kata yang sederhana yaitu rotor, rotator, civic, madam, racecar, level, malam, kapak, dan lain-lain. Untuk contoh lain yaitu kalimat palindrome adalah No lemon no melon, No devil lived on, Swap God for a janitor rot in a jar of dog paws, dll.

Dibawah ini adalah graf dari palindrome detector, merupakan sebuah simulasi mesin turing yang berfungsi untuk mendeteksi kata palindrome yang diinputkan oleh user. Kata atau untai yang dibentuk masih terbatas pada penggunaan huruf "A" dan "B".

Contoh kata yang dibentuk adalah “ABAABBA” untuk kata yang tidak termasuk dalam palindrome, dan “BABAB” untuk kata yang termasuk dalam palindrome.



Gambar Graf Palindrome Detektor lengkap

Pemrograman sederhana jenis mesin Turing ini tidak sesulit yang dibayangkan. Dimana sebenarnya pemrograman ini akan membentuk graph. Transisi state terdiri dari 5-tupel rangkaian pada setiap baris, dengan format seperti ini:

1, _, 2, #, >

2, A, 3, A, >

Karakter '_' dapat digunakan untuk menunjukkan kosong(blank), 'H' untuk menunjukkan sebagai state berhenti/Halt (hanya berlaku pada sisi kanan transisi), dan '<' dan '>' untuk menunjukkan arah masing-masing bergerak kekiri atau kanan.

G. Sejarah AI

1. Awal Konsep AI (1950)

Sejarah kecerdasan buatan dimulai pada tahun 1950 ketika Alan Turing menerbitkan makalah berjudul *Computing Machinery and Intelligence*. Karya ini menjadi fondasi dalam memahami konsep AI serta membuka peluang bagi penelitian lebih lanjut di bidang ini. Secara historis, perjalanan AI hingga penggunaannya dalam kehidupan sehari-hari telah melalui berbagai tahapan yang panjang.

Turing juga memperkenalkan gagasan tentang Mesin Turing Universal, sebuah konsep teoretis mengenai mesin yang mampu menjalankan program yang tersimpan dalam memorinya. Inovasi ini dianggap revolusioner pada masanya

karena memunculkan ide bahwa mesin dapat memperbaiki programnya sendiri dan belajar dari pengalaman.

2. Perkembangan Jaringan Saraf Tiruan (1951-1958)

Kemajuan awal dalam bidang jaringan saraf tiruan terjadi pada tahun 1951 ketika Marvin Minsky dan Dean Edmonds menciptakan SNARC, jaringan saraf tiruan pertama yang meniru fungsi neuron manusia menggunakan sistem elektronik.

Pada tahun 1958, Frank Rosenblatt mengembangkan konsep perceptron, sebuah model jaringan saraf sederhana yang mampu mengenali pola dari data yang diberikan. Teknologi ini menjadi dasar bagi perkembangan deep learning, yang kini banyak digunakan dalam berbagai aplikasi AI modern. Penemuan ini membuktikan bahwa mesin tidak hanya bisa diberi perintah tetapi juga dapat belajar dari pengalaman.

3. Pengenalan Istilah Kecerdasan Buatan (1956)

Istilah Artificial Intelligence (AI) atau kecerdasan buatan pertama kali diperkenalkan dalam sebuah workshop di Dartmouth College pada tahun 1956. Acara ini dipimpin oleh John McCarthy, seorang ilmuwan komputer yang kemudian dikenal sebagai Bapak AI.

Workshop tersebut dihadiri oleh para pemikir terkemuka seperti Marvin Minsky, Nathaniel Rochester, dan Claude Shannon. Selain mencetuskan istilah AI, McCarthy juga mengembangkan bahasa pemrograman LISP, yang menjadi salah satu bahasa utama dalam penelitian AI.

Beberapa ilmuwan lain yang dianggap sebagai pelopor AI antara lain Alan Turing, Marvin Minsky, Allen Newell, dan Herbert A. Simon, yang berkontribusi dalam berbagai aspek pengembangan AI sejak awal.

4. Awal Konsep Pembelajaran Mesin (1959)

Pada tahun 1959, Arthur Samuel memperkenalkan istilah machine learning (pembelajaran mesin) untuk menggambarkan kemampuan komputer dalam belajar dari data tanpa instruksi eksplisit.

Samuel menciptakan sebuah program permainan catur yang dapat meningkatkan kemampuannya dengan menganalisis permainan yang telah dimainkan

sebelumnya. Ini menjadi bukti awal bahwa komputer dapat belajar dan beradaptasi tanpa perlu diatur secara langsung oleh manusia.

5. Terobosan Robotika dan Pemrosesan Bahasa Alami (1960-an)

Pada tahun 1966, Stanford Research Institute mengembangkan robot mobile pertama bernama Shakey. Robot ini menjadi terobosan dalam dunia robotika karena mampu menggabungkan teknologi visi komputer, navigasi, dan pemrosesan bahasa alami.

Pada periode yang sama, Joseph Weizenbaum menciptakan chatbot pertama yang dikenal sebagai Eliza. Program ini mampu meniru percakapan manusia dan menjadi awal dari teknologi asisten virtual seperti Siri, Alexa, dan Google Assistant yang ada saat ini.

6. AI Winter (1970-an – 1980-an)

Dekade 1970-an hingga 1980-an dikenal sebagai "AI Winter", periode ketika ekspektasi tinggi terhadap AI tidak sejalan dengan kemajuan teknologi yang ada. Karena banyak proyek AI yang gagal memenuhi harapan, pemerintah dan industri mulai mengurangi pendanaan untuk penelitian AI. Salah satu tantangan utama saat itu adalah keterbatasan komputasi dan kesulitan dalam menciptakan sistem yang benar-benar dapat berpikir seperti manusia.

7. AI dalam Dunia Bisnis dan Riset (1980-an)

Pada awal 1980-an, AI kembali mendapat perhatian dengan hadirnya komputer Lisp, yang dirancang khusus untuk penelitian kecerdasan buatan.

Pada tahun 1985, Judea Pearl memperkenalkan jaringan Bayesian, yang memungkinkan sistem AI menangani ketidakpastian dalam pengambilan keputusan. Teknologi ini kini digunakan dalam sistem rekomendasi, diagnosis medis, dan pengenalan suara.

8. Kemajuan AI dalam Permainan dan Kecerdasan Visual (1990-an)

Pada tahun 1997, IBM Deep Blue mencetak sejarah dengan mengalahkan Garry Kasparov, juara dunia catur. Kemenangan ini menjadi bukti bahwa AI dapat mengolah data dalam jumlah besar dan membuat keputusan optimal dalam kondisi kompetitif.

Di era yang sama, Yann LeCun mengembangkan Convolutional Neural Networks (CNN), yang menjadi dasar teknologi pengenalan gambar dalam berbagai aplikasi seperti pengenalan wajah, analisis gambar medis, dan mobil otonom.

9. Revolusi Deep Learning (2010-an)

Pada dekade 2010-an, AI mengalami lonjakan besar berkat deep learning. Pada tahun 2012, Geoffrey Hinton, Ilya Sutskever, dan Alex Krizhevsky memenangkan kompetisi ImageNet dengan menggunakan arsitektur CNN, yang membawa AI ke tingkat akurasi yang lebih tinggi dalam mengenali pola visual.

Perusahaan seperti DeepMind juga mengembangkan AI berbasis reinforcement learning, yang memungkinkan sistem belajar sendiri melalui pengalaman. Salah satu pencapaiannya adalah AI yang mampu menguasai permainan video dan mengalahkan pemain profesional manusia.

10. AI Generatif dan Kehidupan Sehari-hari (2020 – Sekarang)

Pada dekade 2020-an, AI generatif menjadi tren utama dengan teknologi seperti GPT-3 dan DALL-E, yang dikembangkan oleh OpenAI. Teknologi ini memungkinkan AI menghasilkan teks dan gambar yang hampir tidak dapat dibedakan dari karya manusia.

AI juga semakin banyak digunakan dalam kehidupan sehari-hari, mulai dari asisten virtual (Siri, Alexa, Google Assistant), diagnostik medis berbasis AI, hingga mobil tanpa pengemudi. Namun, seiring dengan perkembangan pesat AI, muncul tantangan baru seperti keamanan data, privasi, dan dampak AI terhadap lapangan kerja. Oleh karena itu, berbagai negara mulai merancang regulasi untuk memastikan AI digunakan secara etis dan bertanggung jawab.

BAB III

PENUTUP

A. Kesimpulan

Dalam makalah ini, telah dibahas perkembangan kecerdasan buatan (AI) dan konsep Mesin Turing, dua elemen penting dalam sejarah dan perkembangan ilmu komputer. Dari pembahasan yang telah dilakukan, dapat ditarik beberapa kesimpulan sebagai berikut :

- 1) Mulai dari sistem pakar berbasis aturan hingga algoritma pembelajaran mendalam yang kompleks, AI telah menunjukkan kemampuan untuk menyelesaikan berbagai masalah di berbagai bidang. Perkembangan pesat saya didorong oleh berbagai metode dan teknik inovatif.
- 2) Konsep Mesin Turing, yang diperkenalkan oleh Alan Turing, merupakan fondasi teoretis yang kuat bagi komputasi modern dan AI. Mesin Turing memberikan model abstrak tentang bagaimana komputer dapat memproses informasi dan menyelesaikan masalah, yang menjadi dasar bagi pengembangan sistem AI modern.
- 3) Berbagai sistem AI, seperti Expert System, Fuzzy System, dan Web Data Mining, telah menunjukkan aplikasi yang luas dalam berbagai bidang. Sistem-sistem ini telah membantu meningkatkan efisiensi, produktivitas, dan pengambilan keputusan di berbagai sektor, mulai dari industri hingga kesehatan.

B. Saran

Berdasarkan kesimpulan di atas, berikut adalah beberapa saran yang dapat dipertimbangkan :

- 1) Diperlukan penelitian lebih lanjut tentang implementasi AI di berbagai bidang. Meskipun AI telah menunjukkan potensi yang besar, masih banyak tantangan dan peluang yang perlu dieksplorasi. Penelitian lebih lanjut diperlukan untuk mengembangkan aplikasi AI yang lebih inovatif dan efisien, serta untuk mengatasi masalah etika dan sosial yang terkait dengan AI.

- 2) Pemahaman dasar tentang AI sangat penting bagi mahasiswa dan praktisi IT. Seiring dengan perkembangan AI yang pesat, penting bagi mahasiswa dan praktisi IT untuk memiliki pemahaman yang kuat tentang konsep-konsep dasar AI, termasuk Mesin Turing, algoritma pembelajaran mesin, dan teknik-teknik AI lainnya. Pemahaman ini akan memungkinkan mereka untuk mengembangkan dan menerapkan solusi AI yang efektif dalam berbagai konteks.
- 3) Perlunya kolaborasi antara akademisi, industri, dan pemerintah dalam pengembangan dan penerapan AI. Kolaborasi ini dapat membantu mempercepat inovasi AI, memastikan bahwa AI dikembangkan dan diterapkan secara bertanggung jawab, dan mengatasi masalah-masalah yang terkait dengan AI.

Dengan demikian, makalah ini diharapkan dapat memberikan kontribusi yang berarti dalam pemahaman tentang perkembangan AI dan konsep Mesin Turing, serta mendorong pengembangan dan penerapan AI yang bermanfaat bagi masyarakat.

REFERENSI

- Lestari, R. (2012). *Penerapan Sistem Pakar dalam Berbagai Bidang*. Jurnal Teknologi Informasi, 5(1), 23-30.
- Kusumadewi, T., & Turban, E. (2003). *Sistem Pakar: Teori dan Aplikasi*. Jakarta: Rineka Cipta.
- Pazos, R., et al. (2016). Comparative study on the customization of natural language interfaces to databases. *PMC*.
- Nihalani, S., Silakari, S., Motwani, M. (n.d.). Natural language Interface for Database: A Brief review. *Semantic Scholar*.
- Chin, K. O., et al. (2014). *Agent Architecture: An Overview*. Transactions on Science and Technology, 1(1), 18-35.
- Wooldridge, M. (2001). *An Introduction to MultiAgent Systems*. John Wiley & Sons.
- Rolf Herken: *The Universal Turing Machine - A Half-Century Survey*, Springer Verlag, [ISBN 3-211-82637-8](#)
- Paul Strathern: *Turing and the Computer - The big idea*, Anchor Books/Doubleday, [ISBN 0-385-49243-X](#)
- Turing, A., *On Computable Numbers, With an Application to the Entscheidungsproblem*, Proceedings of the London Mathematical Society, Series 2, Volume 42, 1936; reprinted in M. David (ed.), *The Undecidable*, Hewlett, NY: Raven Press, 1965;
- Boolos, G. and Jeffrey, R., *Computability and Logic*, 2nd ed., Cambridge: [Cambridge University Press](#), 1980.
- Rogozhin, Yurii, "A Universal Turing Machine with 22 States and 2 Symbols" [Diarsipkan](#) 2005-03-08 di [Wayback Machine](#)., *Romanian Journal Of Information Science and Technology*, 1(3), 259-265, 1998. (surveys known results about small universal Turing machines)
- Wolfram, Stephen, *A New Kind of Science*, Wolfram Media, [ISBN 1-57955-008-8](#)
- Quinlan, J. R. C4.5: Programs for Machine Learning. Morgan Kaufmann., 1993.
- MacQueen, J. B., Some methods for classification and analysis of multivariate observations, in Proc. 5th Berkeley Symp. Mathematical Statistics and Probability, 1967.
- Vapnik, V. N. 1995. The Nature of Statistical Learning Theory. Springer-Verlag.
- Rakesh Agrawal and Ramakrishnan Srikant. Fast Algorithms for Mining Association Rules. In VLDB '94.
- McLachlan, G. and Peel, D. (2000). Finite Mixture Models. J. Wiley, New York.

