

# Aplikasi Decision Tree dalam Menyusun Pertanyaan untuk Menebak Karakter pada Permainan Akinator

Kenneth Dave Bahana - 13521145<sup>1</sup>

Program Studi Teknik Informatika

Sekolah Teknik Elektro dan Informatika

Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia

<sup>1</sup>13521145@mahasiswa.itb.ac.id

**Abstract**— Salah satu bidang yang sangat dikenal banyak orang dalam perkembangan Teknologi yang pesat di era ini adalah *video games*. Salah satu game yang mendunia pada tahun 2009 adalah Akinator, yang merupakan A.I. yang mampu menebak karakter yang pemain pikirkan. Dengan beberapa pertanyaan yang terlihatnya belum mampu menjawab pemikiran pemain, ternyata ditebak jauh lebih cepat dari yang seharusnya atau manusia lain tidak bisa lakukan. Keunikan Akinator ini menggunakan aplikasi dari cabang ilmu Matematika Diskrit berupa pohon keputusan yang dalam makalah ini akan dibahas lebih lanjut.

**Keywords**— Akinator, pohon keputusan, Matematika Diskrit, Teknologi

## I. INTRODUCTION

Perkembangan teknologi hingga era ini berkembang pesat. Salah satu bidang yang terus berkembang adalah *video game*. Kini semakin banyak *genre* atau kategori permainan yang ada karena semakin berkembangnya teknologi, semakin banyak ide permainan yang dapat diwujudkan. Akinator adalah sebuah game yang sangat terkenal akan kemampuan dari A.I. (*Artificial Intelligence*) dalam game tersebut yang dinamakan Akinator, mampu menebak nama dari seseorang atau sebuah karakter yang pemain pikirkan.



Gambar 1.1 Halaman Utama Permainan Akinator

Sumber: <https://en.akinator.com/>

Akinator adalah sebuah permainan yang dikembangkan oleh perusahaan Perancis *Eloquence* dan dirilis pada tahun 2007. Permainan ini sangat terkenal pada tahun 2009 akan kemampuannya yang mampu menebak setiap pemikiran karakter dari pemain. A.I. ini mampu menebak pilihan karakter pemain hanya dengan menanyakan 20 pertanyaan dan melanjutkannya apabila penebakan dari Akinator tidak tepat.

## II. LANDASAN TEORI

Pada bab ini akan dijelaskan landasan teori yang akan digunakan dalam pemilihan Akinator untuk mencari karakter yang ditentukannya. Konsep utama yang digunakan dalam Akinator berkaitan dengan pohon keputusan dan probabilitas.

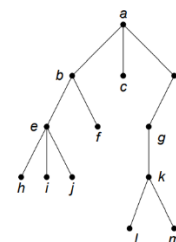
### 2.1 Pohon

Dalam struktur algoritmik, definisi dari pohon adalah sebuah graf tak-berarah terhubung yang tidak mengandung sirkuit. Misalkan sebuah pohon yang didefinisikan sebagai  $G = (V, E)$  dengan jumlah simpul  $n$ , maka sifat – sifat (properti) sebuah pohon tersebut adalah:

1.  $G$  adalah pohon.
2. Setiap pasang simpul di dalam  $G$  terhubung dengan lintasan tunggal
3.  $G$  terhubung dan memiliki  $m = n - 1$  buah sisi
4.  $G$  tidak mengandung sirkuit dan memiliki  $m = n - 1$  buah sisi
5.  $G$  tidak mengandung sirkuit dan penambahan satu sisi pada graf akan membuat hanya satu sirkuit
6.  $G$  terhubung dan semua sisinya adalah jembatan

### 2.2 Pohon Berakar

Pohon berakar (*rooted tree*) merupakan pohon yang memiliki satu simpul sebagai akar dan sisi – sisi yang dibentuk dari akar diberi arah sehingga menjadi graf berarah.



Rinaldi Munir/IF2120 Matematika Diskrit

Gambar 2.1 Contoh Pohon Berakar

Sumber: <https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2021-2022/Pohon-2021-Bag2.pdf>

### 2.2.1 Anak (child atau children) dan Orang tua (parent)

Anak dalam sebuah pohon merupakan simpul yang berperan sebagai *successor* dari simpul sebelumnya yang disebut orang tua (*parent*). Orang tua sendiri merupakan *predecessor* dari simpul anak. Pada contoh gambar 1, h, i, dan j merupakan simpul anak dari e sehingga e merupakan orang tua dari ketiga simpul tersebut dan e sendiri merupakan salah satu simpul anak dari b yang merupakan orang tua dari simpul e.

### 2.2.2 Lintasan (*path*)

Lintasan merupakan keterangan seluruh simpul yang terhubung melalui sisi berarah dari suatu simpul awal ke simpul target. Contoh berdasarkan gambar 1, lintasan d ke m adalah d, g, k, m dengan panjang lintasan 3, yaitu d ke g, g ke k, dan k ke m.

### 2.2.3 Saudara kandung (*sibling*)

Saudara kandung adalah suatu simpul dengan suatu simpul lainnya memiliki simpul orang tua yang sama. Contoh, b merupakan saudara kandung dari c dan d, sedangkan simpul a tidak memiliki saudara kandung karena merupakan akar utama dari pohon.

### 2.2.4 Upapohon (*subtree*)

Upapohon adalah pohon yang terdapat di dalam pohon. Pada gambar 1, b sebagai akar pohon utama merupakan sebuah upapohon dengan simpul anak e dan f, serta simpul e memiliki simpul anak h, i, j. simpul e, h, i, j sendiri juga merupakan sebuah upapohon.

### 2.2.5 Derajat (*degree*)

Derajat sebuah simpul adalah jumlah anak dari simpul tersebut. Contoh pada gambar, simpul a berderajat 3, b berderajat 2, c berderajat 0, dan seterusnya.

### 2.2.6 Daun (*leaf*)

Daun merupakan sebuah simpul yang tidak memiliki anak (ujung pohon) atau berderajat nol. Contoh pada gambar 1, simpul h, i, j, f, c, l, dan m merupakan daun.

### 2.2.7 Simpul Dalam (*internal node*)

Simpul dalam merupakan simpul yang mempunyai anak. Pada gambar 1, simpul a, b, e, f, d, g, dan k merupakan simpul dalam.

### 2.2.8 Aras (*level*) atau Tingkat

Tingkat merupakan jumlah serta letak predecessor dari akar pada pohon. Pada gambar 1 sebagai contoh, pohon tersebut memiliki empat predecessor dengan simpul a sebagai aras 0, simpul b sebagai aras 1, simpul i sebagai aras 4.

### 2.2.9 Tinggi (*height*) atau Kedalaman (*depth*)

Kedalaman merupakan aras terbesar suatu pohon. Dalam contoh gambar 1, kedalaman pohon tersebut adalah 4.

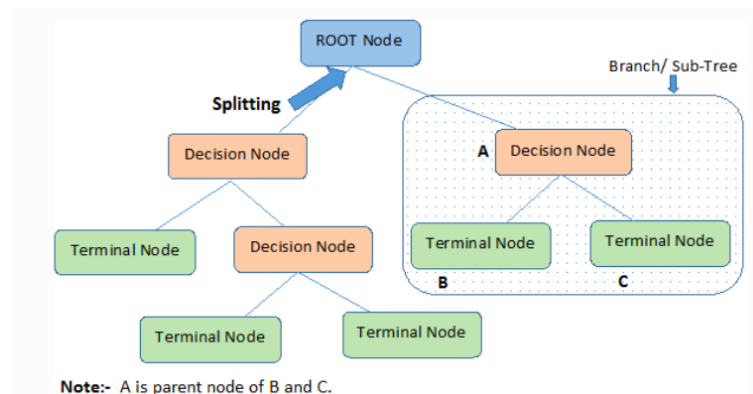
## 2.3 Pohon Biner (*Binary Tree*)

Pohon biner adalah sebuah pohon yang setiap simpul mempunyai paling banyak 2 buah anak. Berdasarkan posisinya, anak dari suatu simpul dibagi menjadi anak kiri (*left child*) dan anak kanan (*right child*). Karena perbedaan urutan anak, pohon biner adalah pohon terurut.

### 2.3.1 Pohon Keputusan (*Decision Tree*)

Pohon keputusan adalah algoritma non-parametrik yang diutilisasikan untuk pohon klasifikasi dan regresi. Pohon keputusan digunakan untuk mengklasifikasikan sebuah permasalahan menjadi lebih kecil berdasarkan keterangan – keterangan yang diberikan untuk merujuk ke data tertentu. Keterangan tersebut berupa iya atau tidak yang merepresentasikan pemilihan simpul anak (pemilihan cabang) dari suatu permasalahan utama, hingga permasalahan tersebut diklasifikasikan hingga permasalahan yang paling terperinci. Dalam *decision tree*, menggunakan *if-else condition* dan terdapat beberapa terminology yang perlu diperhatikan.

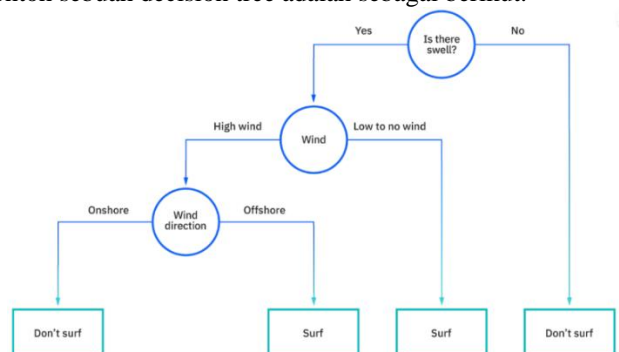
1. Simpul akar (*Root Node*) berperan membagi data menjadi dua bagian atau lebih.
2. Simpul keputusan (*Decision Node*) merupakan sebuah pilihan yang dipilih setelah akar membagi data, pada simpul keputusan dibagi menjadi lebih banyak bagian lagi yang berdasarkan pemilihan “iya” atau tidak”.
3. Simpul terminal (*Terminal Node*) merupakan daun atau akhir dari *decision tree* yang tidak bisa dibagi menjadi semakin banyak bagian lagi, dan dapat dibuat suatu kesimpulan atau perincian permasalahan.



Gambar 2.2 Pemetaan Pohon Kesimpulan

Sumber: <https://towardsai.net/p/programming/decision-trees-explained-with-a-practical-example-fe47872d3b53>

Pada ujung dari pohon yang berupa daun, merupakan hasil perincian permasalahan yang ditargetkan berupa kesimpulan. Contoh sebuah decision tree adalah sebagai berikut.



### Gambar 2.3 Contoh Sederhana Decision Tree

Sumber: <https://www.ibm.com/id-en/topics/decision-trees>

#### 2.3.2 Pohon Keputusan Akinator

Penggunaan pohon keputusan pada permainan Akinator memiliki elemen tambahan dikarenakan faktor *big data* yang diterima A.I. tersebut, yaitu banyaknya data orang – orang di sekeliling dunia, sehingga pohon keputusan yang digunakan Akinator mengandung elemen *randomizer* juga, dimana simpul – simpul pada pohon mengandung ketiga jenis berikut.

- Decision Node* yang merupakan simpul yang memilih suatu keputusan dari dua pilihan atau lebih yang dalam kasus ini digunakan untuk meminimalisir kedalaman pohon pencarian target karakter.
- Chance Node* yang bersifat diluar kendali Akinator, yaitu kemungkinan – kemungkinan *decision tree* yang secara persentase dalam meminimalisir kedalaman pohon berjumlah sama, sehingga *chance node* memilih salah satu dari beberapa pilihan tersebut secara acak.
- Terminal Node* yang merupakan hasil dari proses pemilihan keputusan dan berada di ujung dari pohon keputusan.

#### 2.4 Probabilitas

Peluang atau probabilitas merupakan sebuah nilai yang digunakan untuk menghitung besar peluang suatu hal akan terjadi. Penilaian ini memiliki cakupan antara nol hingga satu. Perumusan probabilitas sederhana sendiri sebagai berikut.

$$P(A) = \frac{n(A)}{n(S)}$$

Dalam perumusan tersebut, S dilambangkan sebagai jumlah kemungkinan yang ada dan A merupakan kemungkinan dari salah satu pilihan. Ketika sebuah koin di lemparkan, terdapat dua kemungkinan permukaan koin yang tampak ketika koin telah mendarat, yaitu kepala atau ekor. Kedua kemungkinan tersebut melambangkan S dan A melambangkan salah satunya, baik kepala atau ekor. Dalam dunia probabilitas, angka tersebut bisa meningkat seperti pelemparan dadu, pemilihan kartu, dan sebagainya. Dalam permainan Akinator, dengan representasi A sebagai data dengan suatu keterangan tipe data dan S sebagai seluruh jumlah data, probabilitas ini digunakan untuk melihat nilai pembagian data dari setiap pertanyaan yang mendekati ke nilai 0.5 atau setengah, untuk membelah data sehingga lebih efisien dalam mengeliminasi data yang tidak berada dalam pemilihan karakter pemain.

### III. APLIKASI

#### 3.1. Gambaran Lintasan Pohon Keputusan

Pembentukan pohon keputusan dalam menyusun pertanyaan – pertanyaan Akinator dimulai dari pertanyaan yang bersifat general atau dalam kasus ini kurang lebih membagi dua (*split*) mengandung kategori dari pertanyaan – pertanyaan lain sehingga angka probabilitas dari setiap kategori pertanyaan telah berubah.

*database* yang ada, yaitu berbagai orang yang ada di internet, sehingga *root node* yang ada dalam pembentukan pohon keputusan merupakan sebuah *chance node*. Kerangka utama yang dilakukan Akinator dalam menebak pemilihan karakter oleh pemain adalah sebagai berikut.

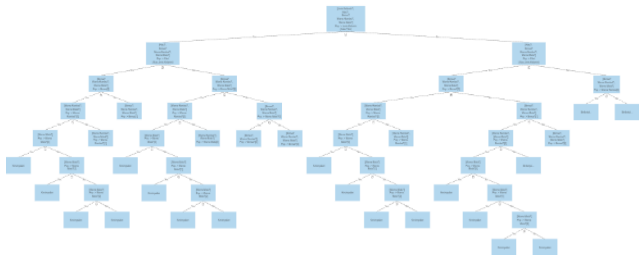
a. Akinator memulai menanyakan pertanyaan pertama, dengan pertanyaan pertama bersifat acak antara kumpulan dari pertanyaan general. Pertanyaan seperti jenis kelamin, karakter manusia atau fiksi, dan berbagai pertanyaan yang bersifat general dan hanya terdapat dua kemungkinan ditanyakan supaya mendapatkan *split* terbesar dalam *database* yang digunakan. Pemain cukup memilih jawaban “iya” atau “tidak” dalam pertanyaan pertama yang diberikan ini.

b. *Decision Node* dimulai setelah pemain memberikan pilihan jawaban yang pertama, dimana Akinator akan melakukan *split* atau pembagian pada data untuk mengeliminasi kontradiksi dari pilihan pemain. Pilihan pertanyaan selanjutnya berdasarkan pemilihan yang telah dilakukan Akinator, sehingga terdapat pertanyaan yang telah tereliminasi. Namun, pemilihan pertanyaan tersebut menggunakan kembali *randomizer* untuk pertanyaan-pertanyaan yang menghasilkan kedalaman tree sependek sehingga pada node tersebut merupakan campuran *Decision Node* dan *Chance Node*. Pemilihan acak yang dilakukan oleh *Chance Node* ini didasarkan oleh tipe data untuk karakter seperti nama, jenis kelamin, dan jenis data lainnya.

c. Kedalaman tree terpanjang yang ditargetkan oleh Akinator adalah kedalaman 19-20. Namun, karakter dapat ditebak oleh Akinator sebelum 20 pertanyaan (yaitu kedalaman 19-20) dan menanyakan kembali kepada pemain apakah pilihannya sudah benar atau belum. Apabila pemain memilih jawaban salah setelah ditunjukkan pilihan karakter dari Akinator, kedalaman tree ditambahkan sebanyak 20 dan Akinator melanjutkan menebak.

#### 3.2. Penerapan Probabilitas dalam Pemilihan Pertanyaan

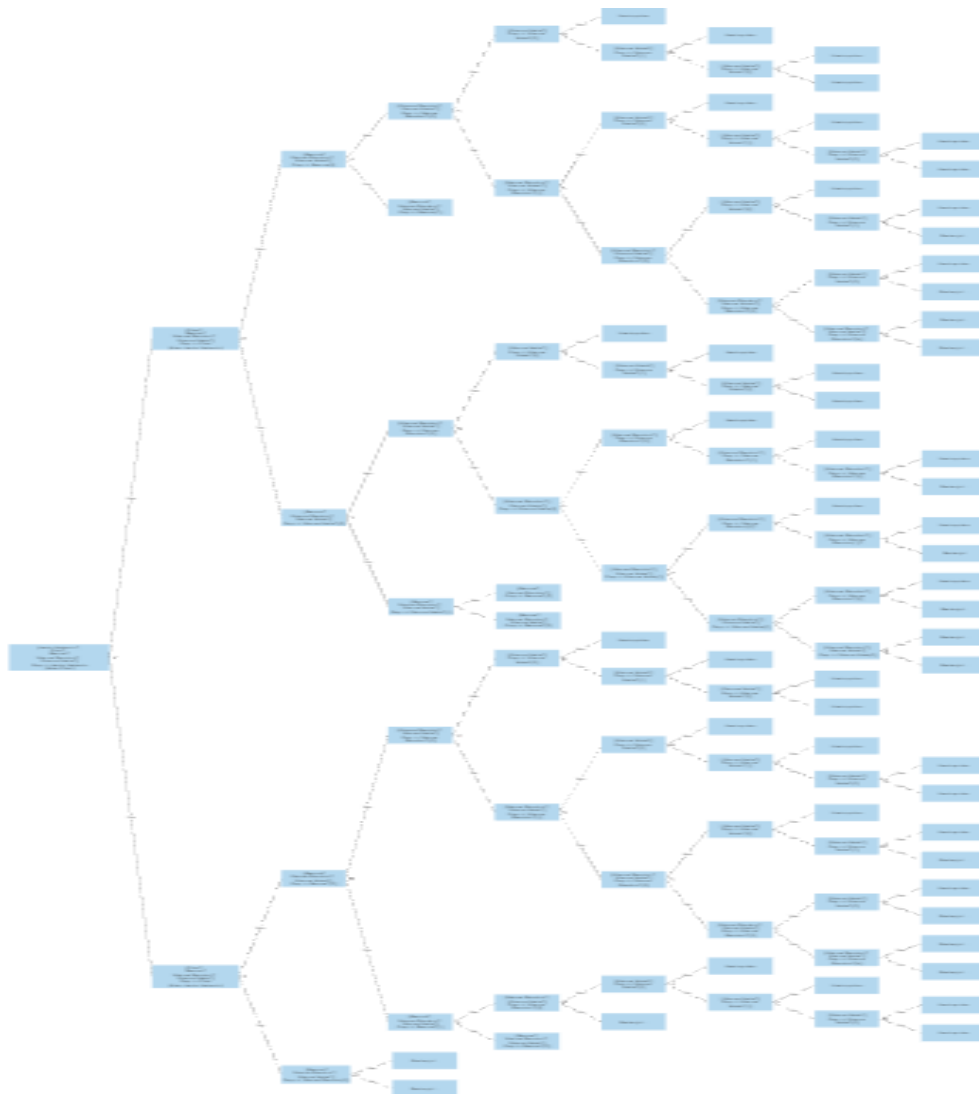
Akinator sendiri memiliki data pertanyaan yang digunakan dalam mempersempit data hingga ke pilihan data yang tepat dalam menunjukkan hasilnya kepada pemain. Pertanyaan – pertanyaan tersebut sendiri berada di sebuah list pertanyaan yang akan dikeluarkan dari list apabila telah ditanyakan pada suatu titik. Setiap jawaban yang diterima, *list* data akan juga hasil evaluasi terbaru, yaitu *list* data dengan eliminasi *split list* hasil pertanyaan. Pada bagian ini, probabilitas pemilihan pertanyaan melakukan perhitungan ulang karena probabilitas dari setiap kategori pertanyaan telah berubah. Hal ini disebabkan karena data yang telah terpotong juga sebelumnya



Gambar 3.1 Contoh Pemilihan 5 Kategori Pertanyaan Dengan Probabilitas  
Sumber: Dokumentasi Penulis

Dari Gambar tersebut, pemilihan pertanyaan pertama dipilih secara acak antara pertanyaan mengenai jenis kelamin atau karakter tersebut nyata atau fiksi. Hal ini disebabkan dalam suatu contoh kasus, pertanyaan ini membagi dua *data* secara keseluruhan sehingga apabila ditanyakan jenis kelamin

karakter laki – laki atau tidak, data dilakukan pembelahan antara data karakter yang berjenis kelamin laki – laki atau perempuan. Jika pemain menjawab iya, maka data – data yang mengandung karakter berjenis kelamin perempuan akan dikeluarkan dari list *data*. Namun, setelah pertanyaan yang bersifat membelah data secara merata tersebut sudah dikeluarkan, pertanyaan selanjutnya seperti warna rambut, warna mata, dan sebagainya akan dipilih dengan probabilitas, sesuai probabilitas mana yang paling mampu membagi data secara merata, dengan menghitung selisih nilai probabilitas antara hasil pembagian data seminimal mungkin. Contoh sederhana jika keseluruhan dari data, isi setengah data berjenis kelamin laki – laki dan setengah data berjenis kelamin perempuan, maka selisih kedua nilai probabilitas tersebut adalah 0, dimana keduanya bernilai 0.5. Setiap pertanyaan yang telah terpilih akan dihitung ulang nilai probabilitas dari setiap kategori pertanyaan.

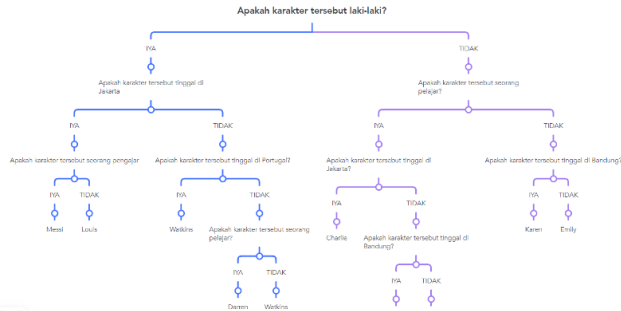


Gambar 3.2 Gambaran Pemilihan Pertanyaan Hingga Aras 8 dari Lima Kategori Pertanyaan.  
Sumber: Dokumentasi Penulis



**3.3 Penerapan Decision Tree dalam Pemilihan oleh Akinator.**

Seperti pemilihan pertanyaan yang telah dijelaskan, *Decision Tree* juga terlibat dalam melakukan *split list* tersebut. Hasil penilaian probabilitas yang dilakukan kemudian dipilih oleh *Decision Tree* sebagai pengambilan keputusan untuk lintasan terpendek.



**Gambar 4.2** Salah satu alur pemilihan Akinator pada suatu data.

Pada salah satu hasil *decision tree* dari Akinator tersebut, data sederhana yang berisi keterangan data jenis kelamin, tempat tinggal, dan status pekerjaan yang berisi hanya dua sampai tiga jenis data untuk setiap kategori. Ketika Akinator mencapai hingga suatu daun dalam *decision tree* tersebut, Akinator akan menunjukkan hasil kesimpulan atau tebakan karakter pilihan pemain dan menanyakan pemain apakah jawaban tersebut sudah benar atau tidak. Jika belum, maka pohon akan melanjutkan pemilihan pertanyaan – pertanyaan.

No.	Nama	Jenis Kelamin	Status Pekerjaan	Tempat Tinggal
1	Darren	Laki-laki	Pelajar	Bandung
2	Milita	Perempuan	Pelajar	Bandung
3	Louis	Laki-laki	Pelajar	Jakarta
4	Watkins	Laki-laki	Pengajar	Bandung
5	Charlie	Perempuan	Pelajar	Jakarta
6	Amel	Perempuan	Pelajar	Portugal
7	Karen	Perempuan	Pengajar	Bandung
8	Emily	Perempuan	Pengajar	Portugal
9	Chris	Laki-laki	Pengajar	Jakarta
10	William	Laki-laki	Pelajar	Portugal

**Gambar 4.3** Tabel Data pada Pohon Keputusan Gambar 4.2  
Sumber: Dokumentasi Penulis

**3.4 Analisis Pembahasan**

Pengolahan *big data* dengan campuran pohon keputusan dan probabilitas sangat efisien karena jumlah data yang terus meningkat untuk karakter baik orang maupun fiksi di internet terus bertambah, namun kecepatan akinator dalam menebak juga masih relatif cepat memberikan pertanyaan antara satu dengan yang lainnya. Namun, list pertanyaan dari Akinator ini masih harus diperbaharui dan ditambahkan dikarenakan list pertanyaan didasari oleh tipe data baru yang ada. Kategori data karakter terus bertambah beserta keterangan spesifikasi keunikan dari setiap karakter yang harus bersifat berbeda – beda karena terdapat kemungkinan salah satu dari karakter memiliki keunikan yang sama persis dari banyaknya orang atau karakter fiksi yang ada. Pengaplikasian dari Akinator sendiri tidak sepenuhnya menggunakan hanya probabilitas dan

*decision tree*. Hasil dari Akinator sendiri merupakan *Expert System Fuzzy A.I.* yang menerapkan *decision tree* serta probabilitas tersebut. Namun, pada kasus data dan jumlah pertanyaan yang sangat amat banyak, penerapan *decision tree* ini juga tetap berujung pada pembentukkan pohon yang dapat jauh melebihi 20 aras dan sistem yang digunakan Akinator sekarang secara keseluruhan memiliki sistem keputusan yang ditambahkan algoritma dan konsep lain sehingga probabilitas tersebut tidak langsung melebihi 20 aras, bahkan menggunakan pendekatan secara manusiawi, yaitu berupa spekulasi terbaik yang bisa dipilih untuk pemilihan pertanyaan yang tepat serta pertanyaan pemastian jawaban yang menguatkan persentase menebak karakter yang telah dipilih maupun karakter yang merupakan persentase tertinggi kemungkinannya ketika di evaluasi oleh Akinator.

Besarnya data yang muncul sekarang juga memunculkan permasalahan legalisasi dari Akinator sendiri untuk diperbolehkan mengambil data setiap orang yang ada disertakan dengan kemampuannya untuk menerima data tersebut. Sebelumnya, Akinator menggunakan data orang dari berbagai macam sumber, utamanya pada media sosial seperti *facebook*, *Instagram*, *twitter*, dan media sosial lainnya yang dapat mengambil informasi biodata orang – orang. Namun, karena banyaknya ketentuan dan data – data tersebut tidak bisa dengan mudah secara legal diterima, banyak terjadi keterbatasan Akinator untuk membatalkan data – data yang ditemukan dari beberapa *platform* sehingga keterbatasan tersebut harus melakukan perombakan ulang dalam segi algoritma dikarenakan sumber data yang berbeda juga mempengaruhi hasil pengolahan data yang diterima harus disesuaikan juga penggunaannya dalam algoritma Akinator itu sendiri. Jadi, permasalahan dari penerimaan data di internet yang terus berubah – ubah tersebut perlunya adaptasi dan terusnya perkembangan dari sistem Akinator.

**IV. KESIMPULAN**

Penerapan *Decision Tree* banyak sekali digunakan dalam kehidupan sehari – hari. Menggunakan *Decision Tree* juga bisa diimplementasikan dengan bermacam – macam cabang lain dengan salah satu contohnya Probabilitas dalam pemilihan yang dilakukan Akinator. Semakin besar data yang digunakan, maka probabilitas yang dihitung juga semakin banyak dan kemungkinan kedalaman pohon masih dapat terus meningkat jika menggunakan data seluruh karakter di internet yang mungkin didapatkan. Dengan demikian, Akinator dapat menebak secara keseluruhan pemilihan karakter dari pemain Akinator, walaupun masih dapat terjadi kesalahan menebak disebabkan oleh banyak sekalinya karakter yang dari kategori data yang sangat banyak, banyak kemiripan yang perbedaannya sangat amat terperinci. Kesalahan yang mungkin terjadi ini disebabkan kategori pertanyaan yang dengan perkembangan zaman sekarang tidak hanya semakin banyak, tetapi jenis atau kategori data bertambah sehingga evaluasi setiap keterangan data dan penambahan jenis pertanyaan yang diperlukan untuk memastikan Akinator dapat memilih karakter yang tepat menjadi pertimbangan untuk mengembangkan kekurangan Akinator.

## V. UCAPAN TERIMA KASIH

Penulis ingin mengucapkan terima kasih kepada Tuhan Yang Maha Esa atas berkah dan rahmatnya makalah ini dapat diselesaikan dan dikerjakan dengan sebenar-benarnya. Terima kasih juga kepada dosen pengajar mata kuliah IF2120 Matematika Diskrit K-02, Dr. Fariska Zakhralatifa Ruzkanda, S.T. M.T. dan penulis juga ingin mengucapkan terima kasih kepada keluarga, teman – teman penulis, serta seluruh pihak karena telah membantu dalam penulisan makalah ini. Mohon maaf apa bila masih ada salah kata dan ketidaksempurnaan dalam makalah ini.

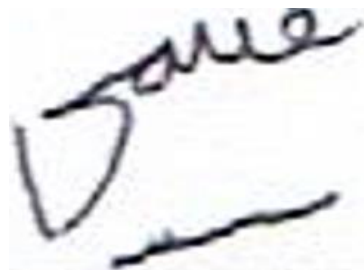
### REFERENCES

1. <https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2021-2022/Pohon-2021-Bag2.pdf> diakses pada 8 Desember 2022
2. <https://www.geeksforgeeks.org/decision-tree/> diakses pada 8 Desember 2022
3. <https://en.akinator.com/content/7/about-us> diakses pada 8 Desember 2022
4. <https://towardsai.net/p/programming/decision-trees-explained-with-a-practical-example-fe47872d3b53> diakses pada 8 Desember 2022
5. <https://vortarus.com/dtace-tutorials/decision-tree-basics/> diakses pada 8 Desember 2022

### PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 3 Desember 2020



Kenneth Dave Bahana 13521145