

DIPLOMADOLGOZAT



Magyar Agrár- és Élettudományi Egyetem
Kaposvári Campus
mesterképzési szak

MENNYIRE TUD EGY SZAKÉRTŐI RENDSZER
SEGÍTENI VÁLLALKOZÁS INDÍTÁSÁBAN?

Belső konzulens: Dr. Varga József László
Egyetemi professzor

Belső konzulens intézete/tanszéke: MATE,
Befektetési, Pénzügyi és Számviteli Tanszék,
Vidékfejlesztés és Fenntartható Gazdaság
intézet

Külső konzulens: Dr. Madaras Szilárd
Egyetemi adjunktus, Sapientia Erdélyi
Magyar Tudományegyetem

Készítette: Albu Zsombor Domokos

Kaposvár

2024

Tartalomjegyzék

1. Bevezető	5
2. Intelligens és szakértői rendszerek története	6
2. 1. Első időszak – 1950 - es évektől	6
2. 2. Második időszak - 1980 -1995-ig	9
2. 3. Harmadik időszak - 1995-től.....	12
3. Intelligens és szakértői rendszerek	16
3. 1. Mesterséges intelligencia	16
3. 2. Gépi tanulás (Machine Learning).....	17
3. 3. Intelligens és szakértői rendszerek előnyei és hátrányai.....	17
4. Bevezetés a szakértői rendszerek világába.....	19
4. 1. Szakértői rendszerek szerkezete.....	19
4. 1. 1. Tudásbázis.....	19
4. 1. 2. Következtető motor	20
4. 1. 2. 1. Forward Chaining	21
4. 1. 2. 2. Backward Chaining.....	21
4. 1. 3. Felhasználói interfész.....	23
4. 1. 4. Magyarázó modul.....	23
4. 1. 5. Tudásszerzési és tanulási modul	23
4. 2. Szakértői rendszerek típusai.....	23
4. 2. 1. Szabályalapú szakértői rendszerek.....	23
4. 2. 2. Framework alapú szakértői rendszerek	25
4. 2. 2. 1. Framework vagy Keret meghatározása.....	25
4. 2. 2. 2. Következtető motor működése a Framework (keretalapú) szakértői rendszerben	26
4. 2. 3. Fuzzy logikán alapuló szakértői rendszerek	26

4. 2. 4.	Neurális hálózaton alapuló szakértői rendszerek	28
4. 2. 4. 1.	Neurális hálózaton alapuló szakértői rendszer felépítése.....	28
4. 2. 4. 2.	Szabályok kinyerése a neurális szakértői rendszerből	29
4. 2. 4. 3.	Neuro-fuzzy rendszerek	29
4. 3.	Az intelligens és szakértői rendszerek felhasználása:	30
5.	Saját alkalmazás bemutatása	32
5. 1.	Felhasznált technológiák.....	32
5. 1. 1.	HTML 5	32
5. 1. 2.	CSS3.....	33
5. 1. 3.	GitHub.....	33
5. 1. 4.	Python	33
5. 1. 5.	PyTorch.....	34
5. 1. 6.	Natural language processing (Természetes nyelvi feldolgozás)	34
5. 2.	Alkalmazás működése.....	35
5. 2. 1.	Felhasználói felület	35
5. 2. 2.	Tudásbázis.....	36
5. 2. 3.	Következtető motor.....	38
6.	Következtetések.....	41
	Irodalomjegyzék	42

1. Bevezető

Mindennapi életünk szerves részévé váltak a szakértői rendszerek, mellyekkel találkozhatunk a streaming platformokon megjelenő, személyre szabott ajánlásoktól kezdve a modern járművekben található vezetőtámogatásig. Napjainkban számos területen van segítségünkre az informatika, melyben nagy szerepet játszik a még fejlődésben levő mesterséges intelligencia.

A mesterséges intelligenciában a szakértői rendszer egy olyan számítógépes rendszer, amely egy emberi szakértő döntéshozatali képességét utánozza. De mik is pontosan az szakértői rendszerek? Hogyan működnek, és miért olyan fontosak? Mely területeken segít a közgazdaságban?

Az államvizsga dolgozatomból többek között megtudhatjuk a választ a fent említett kérdésekre, valamint betekintést nyerhetünk a működésükbe és jelentőségükbe. Ugyanakkor részletesen megismerhetjük ezen rendszerek történetét, típusait, felépítésüket, felhasználásukat, valamint előnyeiket és hátrányaikat.

Nem utolsó sorban létrehoztam egy előreirányított neurális hálózaton alapuló szakértői rendszert, amely vállalkozásindítás, marketing és üzleti tervezés kapcsán nyújt tanácsadást, azon személyeknek, akik szeretnének vállalkozást indítani, vagy szakmailag fejlődni ezen a területen. A webes felületen keresztül a felhasználó egy Chat GPT-hez hasonló, szöveges interakcióba léphet az elkészített alkalmazással.

Az elkészített alkalmazásom felépítését, működését, felhasznált technológiákat szintén bemutattam a dolgozatomban, viszont aki előben is ki szeretné próbálni, az letöltheti a programot a következő GitHub linkről: https://github.com/A-Zsombi/Allamvizsga_Python

2. Intelligens és szakértői rendszerek története

A mesterséges intelligencia szerény kezdeteitől, a koncepcióként való megjelenésétől az átalakító technológiaként való, jelenlegi állapotáig jelentős fejlődésen ment keresztül, amely folyamatosan feszegeti a határokat. Az elkövetkezendő fejezetben bemutatom az intelligens rendszerek fejlődési útját, kiemelve a legfontosabb mérföldköveket és áttöréseket, amelyek hozzájárultak a fejlődéséhez.

Történetüket alapvetően a következő három nagy időszakra csoportosíthatjuk:

2. 1. Első időszak – 1950 - es évektől

A mesterséges intelligencia magvait az 1950-es és 1960-as években vetették el, amikor Alan Turing univerzális számítógépről alkotott koncepciója és John McCarthy "mesterséges intelligencia" kifejezésének megalkotása megalapozta a további kutatásokat. Turing felvetette, hogy az emberek a problémák megoldásához és a döntések meghozatalához a rendelkezésre álló információkat és észérveket használják, miért ne tudnák a gépek is ugyanezt megtenni. Ez volt a logikai kerete az 1950-ben megjelent, Computing Machinery and Intelligence (Számítógépek és intelligencia) című tanulmányának, amelyben arról értekezett, hogyan lehet intelligens gépeket építeni.

(Haocheng Tan, 2017)

A korai erőfeszítések a szabályalapú rendszerek létrehozására és az emberi döntéshozatali folyamatokat utánzó algoritmusok kifejlesztésére összpontosítottak.

1956-ban készítették az első kísérleti programot Allen Newell, Herbert Simon és Cliff Shaw, melynek neve a Logic Theorist lett. A programot az emberi problémamegoldó képességek utánzására képzelték el, melynek finanszírozója a Research and Development (RAND) Corporation volt. Bemutatása 1956-ban történt Dartmouth Summer Research Project on Artificial Intelligence (DSRPAI) konferencián, melyet John McCarthy és Marvin Minsky szervezett. (Solomonoff, R.J., 1985)

Ezen a történelmi jelentőségűnek mondható konferencián McCarthy, egy nagyszabású közös munkát elképzelve, különböző területek vezető kutatóit hívta össze, hogy megvitassák és létrehozzák az akkor még ismeretlennek mondható mesterséges intelligencia foglamát. Tárgyaltak többek között a számítógépekről, a természetes nyelvi feldolgozásról, a neurális hálózatokról, a számításelméletről és az absztrakcióról is. A konferencia állítólag nyolc hétig tartott, melynek 47 résztvevője volt összesen.

Sajnos a konferencia nem McCarthy elképzelései szerint alakult, mivel nem konkrét megállapodás nem született a témákkal kapcsolatosan. A történetek ellenére a konferencián résztvettek egyetértett abban, hogy a mesterséges intelligencia igenis létrehozható. Ugyanakkor az esemény jelentőségét nem lehet alábecsülni, mivel nagymértékben fellendítette a mesterséges intelligencia kutatásának és fejlesztésének elkövetkező húsz-harminc évét. A dartmouthi konferenciát követően, 1956-tól körülbelül 1974-ig tartó időszakot nevezzük a mesterséges intelligencia első nyarának.

Ez időben a számítógépek már egyre gyorsabbak kezdtek lenni, ugyanakkor több információt tudtak tárolni, valamint olcsóbbá és hozzáférhetőbbé is kezdtek válni. A korai próbálkozások, mint például Joseph Weizenbaum ELIZA-ja vagy Allen Newell és Herbert Simon általános problémamegoldója, melyet 1957-ben készítettek, ígéretesnek tűntek a beszélt nyelv értelmezése vagy a problémamegoldás terén. (Norvig; Peter, 1992)

Többek között az előzőekben említett sikerek, valamint a vezető kutatók érdekérvényesítése meggyőzte az amerikai kormányzati szerveket, mint például a Defense Advanced Research Projects Agency-t (DARPA), hogy számos intézményben támogassák és finanszírozzák a mesterséges intelligencia kutatását. A kormányt különösen érdekelte egy olyan gép, amely képes a beszélt nyelv átírására és fordítására, valamint a nagy áteresztőképességű adatfeldolgozásra. Az optimizmus nagy volt, de az elvárások még nagyobbak voltak. 1970-ben Marvin Minsky a Life magazinnak azt nyilatkozta, hogy: "három-nyolc éven belül olyan gépünk lesz, amely egy átlagos ember általános intelligenciájával rendelkezik". Habár az alapvető próbálkozások, kísérletek megvoltak, de az absztrakt gondolkodás eléréséig és a természetes nyelvfeldolgozás tényleges eléréséhez még sok évnek kellett elteltie.

1965-ben a Nemzeti Repülési és Űrhajózási Hivatal követelményei alapján, Edward Feigenbaum és Joshua Lederberg sikeresen fejlesztette ki a DENDRAL nevű rendszert, a kaliforniai Stanford Egyetemen.

A program feladata a molekulák szerkezetének feltérképezése volt, hogy segítse a vegyészeket az ismeretlen szerves molekulák azonosításában. A molekula spektrális vizsgálatához, szabályalapú elemzést használva, a DENDRAL meghatározta a lehetséges szerkezetek halmazát, majd ezeket összehasonlította az adatokkal, hogy megállapítsa melyik a helyes. A DENDRAL program hozzájárult a kémikusok döntéshozatalának a megkönnyebítésében és automatizálásában, ennek köszönhetően a DENDRAL-t tekintik az első szakértői rendszernek a történelemben. (Berk, A. A. LISP, 1985)

Egy másik, a DENDRAL-ból származó szakértői rendszer a MYCIN nevű alkalmazás volt, amelyet 1970-ben fejlesztették ki, ugyancsak a kaliforniai Stanford Egyetemen. Az alkalmazás segíteni kívánt a súlyos okozó baktériumok azonosításában, pl. agyhártyagyulladás és antibiotikumok ajánlásában, a beteg testsúlyához igazított adagolással. Ugyanakkor véralvadási betegségek diagnosztizálására is tervezték használni. A program neve az antibiotikumokról származik, mivel sok antibiotikumnak van "-mycin" utótagja. Következtetési motra meglehetősen egyszerű volt, mely megközelítőleg egy 600 szabályból álló tudásbázisból működött. A program egyszerű igen/nem vagy szöveges kérdések hosszú sorozatával kérdezte le a programot futtató orvost. A végén a lehetségesen jelen levő baktériumok listáját adta meg, magasról alacsonyra rangsorolva az egyes diagnózisok valószínűsége alapján. (Jackson; Peter, 1998)

A Stanfordi egyetemen végzett kutatás szerint a MYCIN 65%-os elfogadhatósági értékelést kapott a kezelési tervre vonatkozóan egy nyolc független szakemberből álló csoportból, ami összevethető volt az akkori oktatói kar, öt tagjának 42,5-62,5%-os értékelésével. (Yu, Victor L, 1979) Ezt a tanulmányt gyakran idézik, hogy megmutassa a terápiás döntésekkel kapcsolatos nézeteltérések lehetőségét, még a szakértők között is.

Végül élesben az alkalmazást soha nem használták, mivel sokan etikai és jogi kérdéseket vetettek fel a számítógépek orvosi alkalmazásával kapcsolatosan, az orvosok felelősségét illetően, amennyiben a rendszer téves diagnózist adna. A legnagyobb probléma azonban, és az oka annak, hogy a MYCIN-t nem használták a gyakorlatban, a

rendszerintegrációs technológiák akkori állapota volt. (Buchanan, B.G.; Shortliffe, 1984) A MYCIN egy önálló rendszer volt, amely megkövetelte, hogy a felhasználó a program által feltett kérdésekre adott válaszok begépelésével adja meg a betegséggel kapcsolatos összes releváns információt. A program egy nagy, időben megosztott rendszeren futott, amely a korai interneten, az ARPANet keresztül volt elérhető, még a személyi számítógépek kifejlesztése előtt.

A mesterséges intelligencia kutatásának finanszírozása ellenére a számítógépek akkoriban még nem voltak képesek elegendő adat és információ feldolgozására és tárolására. Meglepő módon a mesterséges intelligenciának éppen azokon a területeken voltak nagyobb nehézségei, amelyek egyszerűbbnek tűntek, mint például a gépi fordítás vagy a természetes nyelv reprodukálása. A tényleges kommunikációhoz például rengeteg szó jelentését kell ismerni többféle kontextusban, melyeket ugyanúgy többféle kombinációkban is kell értelmezni. Hans Moravec, McCarthy akkori doktorandusza azt állította, hogy "a számítógépek még milliószor túl gyengék ahhoz, hogy intelligenciát mutassanak". Egy angol nyelvi elemző program például mindössze 20 szóból álló szókincset tudott kezelni.

Az idő teltével és a széleskörűen használható programok hiányában, kezdett fogyni a finanszírozás, mivel az idő teltével a befektetők rájöttek, hogy a kutatások és fejlesztések nem hoztak áttörő eredményeket, melynek eredményeként visszavonták támogatásukat. Így kezdődött a mesterséges intelligencia, úgymond első télje, amely egészen az 1980-as évek elejéig tartott.

2. 2. Második időszak - 1980 -1995-ig

Az 1980-as években újra megnőtt az érdeklődés a szakértői rendszerek alkalmazása iránt, melyhez hozzájárult a finanszírozási források megnövekedése és az algoritmikus eszköztár bővülése is. Megszaporodtak a projektek és kísérletek is a témában, így születtek meg a "második generációs" szakértői rendszerek, melynek következtében bevezetésre került a valószínűségi modell, amely az okokra és a lehetséges hatásokra következtetett. A fejlődésnek ezt az intenzív időszakát az is elősegítette, hogy a szakértői rendszereket az iparban és a kereskedelemben is kezdték alkalmazni.

John Hopfield és David Rumelhart kezdte népszerűsíteni a "Deep Learning", azaz a mélytanulási technikákat, amelyeknek köszönhetően a számítógépeknek lehetőségük lett úgymond tanulni az előző tapasztalatok felhasználásával. Másrészt Edward Feigenbaum, amerikai informatikus olyan típusú szakértői rendszereket hozott létre, amelyek egy valós szakértő döntéshozatali folyamatát képesek voltak utánozni. A programokat megtanították, milyen válaszokat adjon különböző helyzetekre, majd ezt követően sikeresen használhatták az elkészített programot a nem szakértők is, azaz a hétköznapi emberek. (David C. Brock, 2018)

Az első olyan megvalósítás, amely már jelentős gazdasági sikert ért el, az R1 (vagy Xcon, eXpert CONfigurer) nevű program volt. A programot John Mc Dermott fejlesztette ki, az amerikai Carnegie-Mellon nevű egyetemen 1978-ban, melyet 1982-ben a Digital Equipment Corporation vezette be, hogy segítse az ügyfeleket a DEC (Digital Equipment Corporation) VAX típusú számítógépes rendszereinek megrendelésében.

Az XCON előtt, amikor a DEC-től VAX-ot rendeltek, minden kábelt, csatlakozót és szoftverdarabot külön-külön kellett megrendelni, mivel akkoriban a számítógépeket és a perifériákat nem dobozokban, komplett állapotban árulták, mint manapság. Az értékesítők nem mindig voltak túlságosan jártasak a műszaki kérdésekben, így előfordult, hogy a vásárlók nem a megfelelő kábelekkel kaptak hardvert, vagy egy nyomtatót a megfelelő illesztőprogramok nélkül, processzort a megfelelő nyelvi chip nélkül, és így tovább. Ez további késedelmet, veszteségeket és kellemetlenségeket jelentett, többek között kiváltván rengeteg ügyfél elégedetlenségét.

Itt jött a megoldás az XCON program által, mely a vevő igényei alapján automatikusan kiválasztotta, a kívánt számítógépes rendszer megfelelő komponenseit. Az ügyfelek megrendelése alapján az R1 képes volt biztosítani, hogy a megrendelés tartalma teljes legyen, ugyanakkor a komponensek közötti térbeli kapcsolatokat is meg tudta határozni. A rendszer több mint 100 komponensből állt, amelyek között különböző interakciós lehetőségek voltak. (McDermott, John, 1980)

Legelőször a programot a DEC New Hampshire-i üzemében kezdték használni, amely megközelítőleg 2500 szabályból álló tudásbázis felhasználásával működött. 1986-ra 80000 megbízást dolgozott fel, melyből 95-98%-os pontosságot ért el. A becslések szerint a DEC

évente 25 millió dollárt takarított meg azáltal, hogy a technikusok hibái esetén kevesebbet kellett ingyen alkatrészeket adni az ügyfeleknek, felgyorsítva ezáltal az összeszerelési folyamatot, és növelve az ügyfelek elégedettségét.

Fellelkesedvén többek között a fentiekben említett sikeres program eredményei alapján is, több új alkalmazás született az elkövetkezendő időszakban. Japán volt az első ország, amely nagy összegeket fektetett a mesterséges intelligenciára tervezett számítógépekbe, melynek következtében Amerika, az Egyesült Királyság és Európa többi része is követte a példáját. A japán kormány meghirdette az ötödik generációs számítógépes projektet (Fifth Generation Computer Systems, elkövetkezendőkben FGCP) melynek részeként széleskörben támogatták az olyan kutatásokat és kísérleteket melyek a szakértői rendszerekre és más, mesterséges intelligenciával kapcsolatos programokra összpontosítottak. Az 1982-1990 közötti időszakban a japánok több mint 400 millió dollárt fektettek be a mesterséges intelligencia, logikai programozás és számítógépes feldolgozás fejlesztésébe.

Azonban még a második generációs szakértői rendszerekkel is voltak problémák. Mindenekelőtt a szakértők tudását tükröző szabályok megírásának nehézségei, valamint azok kezelése és karbantartása. Ráadásul a szakértői rendszerek körüli felhajtás sokkal gyorsabban nőtt, mint az akkori technológiai fejlettség. Sajnos az FGCP projekt ambiciózus céljainak az elérése is csak részben sikerült. Ugyanakkor azt lehet állítani, hogy az FGCP közvetett hatásai a tudósok és mérnökök egy tehetséges fiatal generációját inspirálták, akik hozzájárultak a későbbi sikerekhez.

A lelkesedés az idő múlásával újra csalódáshoz vezetett. Az Apple és az IBM erősebb, általános célra használt számítógépeket vezetett be, mint amilyeneket a mesterséges intelligenciára terveztek, ezzel is hozzájárulván a mesterséges intelligencia iparágának tönkretételéhez. Emellett 1987-ben a DARPA, az Egyesült Államok Védelmi Minisztériumának kormányzati ügynöksége és a mesterséges intelligencia kutatásának egyik fő finanszírozója, amely 1985-ben csaknem 100 millió dollárt költött a MI kutatására, úgy döntött, hogy leállítja a befektetéseket és inkább a rövid távon jobb kilátásokkal rendelkező technológiákra összpontosít.

A szakértői rendszerek iránti lelkesedés így ismét alábbhagyott és beruházások is a mesterséges intelligencia fejlesztésébe nagy visszaesést szenvedtek. Ennek következtében

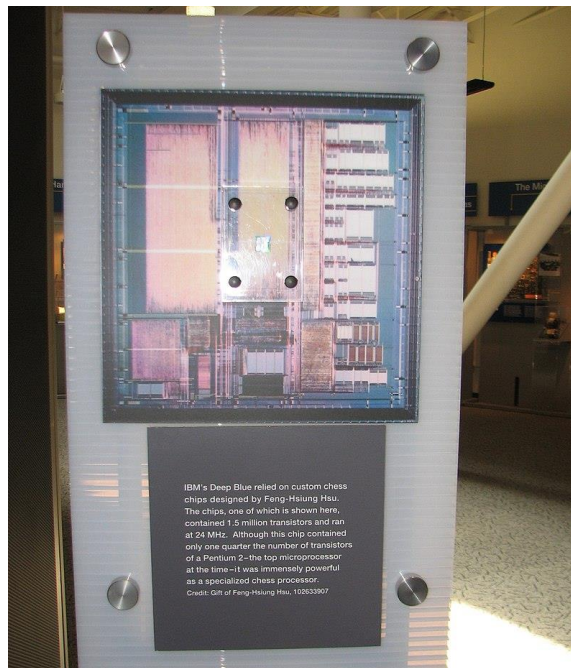
kezdődött a mesterséges intelligencia, úgymond második tékje, amely egészen az 1990-es évek közepéig tartott.

2. 3. Harmadik időszak - 1995-től

Ironikus módon a kormányzati finanszírozás hiányában a mesterséges intelligencia újra nagy fejlődésnek indult. A mérnököknek és fejlesztőknek a mesterséges intelligencia számos mérföldkőnek számító célját sikerült megvalósítaniuk az 1990-2000-es évek közti időszakban. 1997-ben Gary Kasparov sakkvilágbajnokot legyőzte az IBM szuperszámítógépen futó Deep Blue nevű sakkozó számítógépes program. (Chandrasekaran, Rajiv, 1997) Ez a mérkőzés volt a történelemben az első alkalom, hogy egy sakkvilágbajnok vereséget szenvedett egy számítógéptől. Az első nagy mérkőzést 1996-ban, folytatták le, amikor még a Deep Blue első verziója vereséget szenvedett Kasparovtól. Az 1997-es mérkőzésből világhírű sikertörténet lett, amely hatalmas előrelépésként szolgált az újabb döntéshozó programok fejlesztésének irányába.

A Deep Blue egy sakkozni tudó szakértői rendszer volt, amelyet egy egyedi, kifejezetten erre a célra épített IBM szuperszámítógépen futtattak. Fejlesztését 1985-ben ChipTest név alatt kezdte Feng-hsiung Hsu, a Pittsburgh-i Carnegie Mellon Egyetem doktorandusz hallgatója. A gép 1987-ben megnyerte az észak-amerikai számítógépes sakkbajnokságot, majd Hsu és csapata 1988-ban egy utóddal, a Deep Thought-tal folytatta. 1989-es doktori címének megszerzése után Hsu és Murray Campbell csatlakozott az IBM Researchhez, hogy folytassák projektjüket, amelynek célja egy olyan gép megépítése volt, amely képes legyőzni egy sakkvilágbajnokot is.

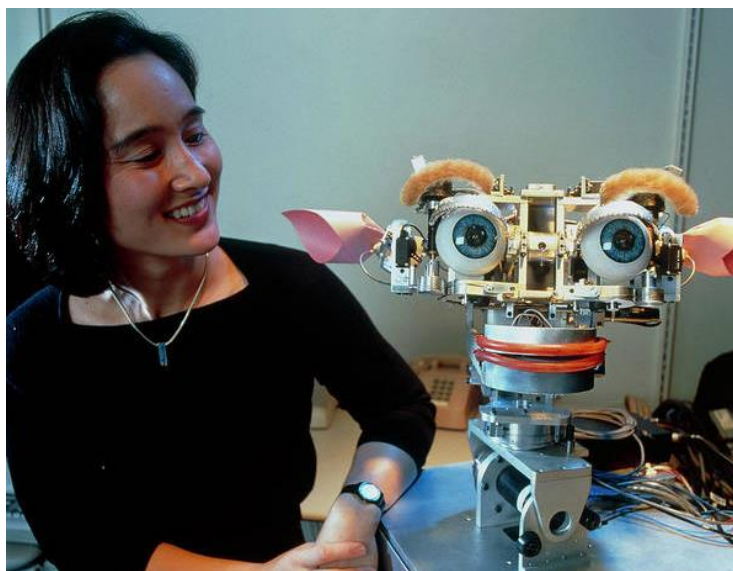
Amint az előzőekben is említettem, Garri Kaszparov világbajnokkal először 1996-ban játszott egy hatjátszmás mérkőzésen, amikor még négy játszmát elvesztett a program. A programot 1997-ben továbbfejlesztették, és egy hatjátszmás újrajátszáson két győzelemmel és három döntetlennel legyőzte Kasparov-ot. Innentől a Deep Blue győzelmét mérföldkőnek tekintették a mesterséges intelligencia történetében, melyről több könyv és film is készült.



Ábra 2.1. A Deep Blue egyik processzora, amely másfél millió 24 MHz tranzisztort tartalmaz

Ugyancsak 1997-ben jelent meg a Windows rendszerben, a Dragon Systems által kifejlesztett, NaturallySpeaking nevű beszédfelismerő szoftver. A NaturallySpeaking nagy előrelépés volt a nyelvi tolmácsolást végző programok fejlesztésének az irányába.

Az egyre újabb számítógépes programok megjelenésével úgy tűnt, hogy nem lesz olyan probléma, amellyel a szakértői rendszerek nem tudnának majd megbirkózni. Az 1990-es évek végén a Massachusettsi Technológiai Intézetben (MIT) Cynthia Breazeal készített olyan robotfejű gépet, amely képes volt felismerni és szimulálni az emberi érzelmeket. A Kismet volt az első olyan robot, amelyet kifejezetten arra terveztek, hogy az embereket természetes szemtől-szembe interakcióba vonja be. Különböző arckifejezésekkel, hanglejtésekkel és mozgással szimulálta az érzelmeket, amely arckifejezéseket a fülek, szemöldökök, szemhéjak, ajkak, az állkapocs és a fej mozdításával hozta létre. Ez nagy úttörőnek számított a szociális robotika avagy társas robotika új területén. (Peter M., Faith D'A., 2000)



Ábra 2.2. Dr. Cynthia Breazeal a Kismet nevű robotjával

Napjainkban a "big data" korszakát éljük, amelyben okostelefonjaink, számítógépeink hatalmas mennyiségű információt képesek gyűjteni rólunk, amelynek köszönhetően személyre szabott ajánlatokkal, javaslatokkal találkozhatunk internetezésünk során. A mesterséges intelligenciát már számos iparágban alkalmazzák, például bankszektorban, szórakoztatóiparban, vagy a marketingben nagyon eredményesnek mutatkozik.

2011-ben az Apple kiadta Siri-t, aki az első és egy széles körben elterjedt virtuális aszisztens. Siri hangalapú lekérdezések és természetes nyelvi feldolgozást használva válaszol a kérdéseinkre, ajánlásokat tesz vagy műveleteket hajt végre számunkra. (Bosker, Biance, 2013) A szoftver alkalmazkodik a felhasználók egyéni használatához, kereséseihez és nyelvi preferenciáihoz, így a visszaküldött eredmények is személyre szabottak lesznek.

2016-ban a Hanson Robotics, Hong Kong-i székhelyű vállalat megalkotta a Sophia nevű humanoid robotot, amely az első "robot emebrként" vált ismertté. (Fitzsimmons, Caitlin, 2017) Ő volt az első olyan robot, amely valóságghű emberi megjelenéssel, az érzelmek érzékelésének és reprodukálásának, valamint a kommunikációnak a képességével hoztak létre.



Ábra 2.3. A 2016-ban létrehozott Sophia nevű robot

Az OpenAI 2020-ban kezdte meg a GPT-3 nyelvi modell bétatesztelését, amely a Deep Learninget használja kódok, versek vagy más hasonló nyelvi és irodalmi szövegek létrehozására. Az OpenAI egy mesterséges intelligencia kutatóintézet, amelyet az OpenAI LP profitorientált vállalat és anyavállalata, a nonprofit OpenAI Inc. hoztak létre. (Hao Karen, 2020)

A Chat GPT nevű chatbot-ot 2022 novemberében adta ki a vállalat, amely azóta is töretlen sikernek örvend, és széleskörben használják a felhasználók.

3. Intelligens és szakértői rendszerek

Az intelligens rendszer a számítógépes rendszerek szélesebb kategóriájára utal, amelyek képesek olyan feladatok elvégzésére, amelyekhez általában emberi intelligencia szükséges. A fogalom magába foglalja a szakértői rendszereket, de ide tartoznak a mesterséges intelligencia más formái is, például a gépi tanulási algoritmusok, számítógépes látás stb.

Ezek olyan gépek vagy szoftverek, amelyek fejlett analitikus vagy kognitív képességekkel rendelkeznek. Az emberekhez hasonlóan képesek lehetnek érzékelni a környezetüket, érvelni, tanulni a tapasztalatokból, döntéseket hozni és problémákat megoldani. A fent említett tulajdonságaik lehetővé teszik, hogy az intelligens rendszerek olyan feladatokat is elvégezzenek, amelyekhez hagyományosan emberi intelligenciára lenne szükség.

Az intelligens rendszerek alkalmazásai számos területen megtalálhatók, beleértve a mesterséges intelligenciát (AI), a gépi tanulást (ML), a robotikát, a virtuális valóságot és még sok más területet. Ilyenek például a Google keresőalgoritmus, az autonóm járművek, a prediktív elemző rendszerek, az ajánlórendszerek, sőt az olyan személyi asszisztensek is mint Siri és Alexa.

3. 1. Mesterséges intelligencia

A mesterséges intelligencia az a tágabb fogalom, amely szerint a gépek képesek olyan feladatokat végrehajtani, amelyeket mi "intelligensnek" tartunk. Célja olyan rendszerek létrehozása, amelyek képesek intelligens és önálló működésre. A mesterséges intelligencia két fő típusba sorolható: Szűk értelemben vett mesterséges intelligencia, amelyet egy szűk feladat elvégzésére terveztek (pl. arcfelismerés vagy internetes keresés), és általános mesterséges intelligencia, amely bármilyen olyan szellemi feladatot képes elvégezni, amelyet egy ember is képes elvégezni. (Russell, Stuart J. & Norvig, Peter, 2021)

Az egészségügyben például a mesterséges intelligencia komplex algoritmusokat és szoftvereket alkalmaz az emberi kognitív folyamatok szimulálására az összetett és gyakran

hatalmas mennyiségű orvosi információk elemzéséhez. Az emberi feldolgozási képességet meghaladó feldolgozási kapacitással az egészségügyi AI-rendszerek gyorsan képesek az egészségügyi szakemberek számára betekintést nyújtani, hatalmas adatbázisok alapján mintákat felismerni, valamint diagnózist és kezelést ajánlani.

3. 2. Gépi tanulás (Machine Learning)

A gépi tanulás a mesterséges intelligencia egy olyan alkalmazása, amely a rendszereknek azt a képességet biztosítja, hogy automatikusan tanuljanak és fejlődjenek a tapasztalatokból, anélkül, hogy kifejezetten programoznák őket. Az az elképzelés köré épül, hogy a gépeknek képesnek kell lenniük tanulni és alkalmazkodni a tapasztalatokon keresztül. A gépi tanulási algoritmusok számítási módszereket használnak arra, hogy közvetlenül az adatokból "tanuljanak" információt anélkül, hogy egy előre meghatározott egyenletre mint modellre támaszkodnának. (Patrick Grieve, 2023)

Egy tipikus intelligens rendszer úgy működik, hogy adatokat vesz fel, azokat egy összetett algoritmuson keresztül feldolgozza, és a megtanult minták alapján előrejelzéseket vagy döntéseket hoz. Egy prediktív elemző rendszer például múltbeli és jelenlegi adatokat használ fel a jövőbeli események előrejelzésére. Egy autonóm jármű érzékelőket és fedélzeti analitikát használ környezetének érzékelésére, döntések meghozatalára és az utakon való navigálásra.

3. 3. Intelligens és szakértői rendszerek előnyei és hátrányai

Az intelligens rendszerek hatalmas mennyiségű adatot képesek elemezni és a másodperc töredékei alatt olyan döntéseket hozni, amelyekhez az embereknek jóval hosszabb időre lenne szükségük. Ez a sebesség és pontosság számos iparágban jelentősen növeli a hatékonyságot és a termelékenységet.

Azáltal, hogy az intelligens rendszerek átveszik az ismétlődő és hétköznapi feladatokat, lehetővé teszik az emberi dolgozók számára, hogy munkájuk összetettebb és

kreatívabb aspektusaira összpontosíthatassanak. Ugyanakkor adatvezérelt meglátásokat használnak a döntéshozatalhoz, ezáltal kiküszöbölve az emberi hibákat.

Számos előnyeikhez tartozik, hogy bizonyos típusok képesek megérteni az egyéni preferenciákat és az egyéni igényekhez igazítani a tapasztalatokat, magas szintű személyre szabást biztosítva. A trendek és minták elemzésével az intelligens rendszerek képesek megjósolni a jövőbeli eredményeket, segítve az olyan iparágakat, mint az egészségügy, a pénzügy és a marketing, hogy megalapozott döntéseket és stratégiákat hozzanak. Az intelligens rendszerek térhódítása jelentős változást jelent a problémamegoldás és a döntéshozatal megközelítésében. A tanulási, alkalmazkodási és döntéshozatali képességüknek köszönhetően ezek a rendszerek még inkább beépülnek az életünkbe. A folyamatos fejlődés ezen a területen egy olyan új korszakot hirdet, amelyben az ember és a gép együttműködése komplex problémákat oldhat meg, hatékonyságot teremthet, ezáltal az innováció és a növekedés lehetőségeinek világát nyitja meg.

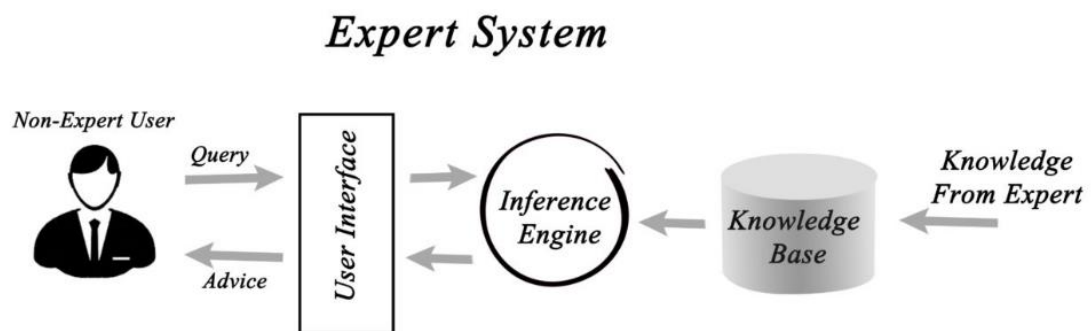
Ugyanakkor az intelligens rendszerek általános hátrányaikhoz sorolhatjuk, hogy a legtöbb nem képes alkalmazkodni egy változó környezethez önmaguktól, sokszor szükséges emberi felülvizsgálat a rendszerek által adott tanácsok ellenőrzéséhez és tudásuk általában csak egy szűk ismeretkörre terjed ki. Említésre méltó az a tény is, hogy sok erőfeszítést és költséget igényel egy jó szakértői rendszer létrehozása. Az olyan fejlődésben levő tulajdonságoknál, mint például a természetes nyelvi feldolgozás folyamata, még mindig van hová fejlődniük. Nem utolsósorban nem valószínű, hogy a szakértői rendszerek új és kreatív megoldásokkal vagy ötletekkel állnak elő.

4. Bevezetés a szakértői rendszerek világába

A szakértői rendszer olyan számítógépes rendszer, amely egy emberi szakértő döntéshozatali képességét utánozza. A rendszer egy tudásbázist és egy szabálykészletet használ a következtetések levonásához és a döntések meghozatalához.

4.1. Szakértői rendszerek szerkezete

Amint az alábbi 4.1 ábra szemlélteti, a szakértői rendszereknek három fő összetevőjük van: Tudásbázis, Következtető motor és Felhasználói interfész. Ugyanakkor említésre méltó még a Magyarázó modul valamint a Tudásszerzési és tanulási modul. (Szilárd Madaras, Norbert Ágoston, 2023)



Ábra 4.1. Szakértői rendszerek összetevői

4.1.1. Tudásbázis

A rendszer legfontosabb eleme a tudásbázis, amely szabályokból és framek-ből épülhet fel, melynek szerepe a szakértői tudás tárolása. A tudásbázishoz való hozzáférés lekérdezéssel történik valamint szabályokkal való feltöltését a tudásszervező készíti elő. Fontos, hogy a tudásbázisnak képesnek kell lennie az új ismeretek megszerzésére, a tudás kifejezésére valamint tárolására. Ennek eléréséhez egy tudásfelépítő programnak hívható, rendszerkomponenst használ a tudásszervező. A létrhozási folyamat befejeztével a tesztelés következik, majd az azt követő próbafuttatás.

4. 1. 2. Következtető motor

A következtető motor, másnéven következtető gép a rendszer válaszadó része, úgymond a szakértői rendszer elméje. Feladata a releváns tudás kinyerése a tudásbázisból, értelmezése és a felhasználó problémájára vonatkozó megoldás megtalálása. A felhasználói interfészen keresztül fogadja a felhasználó által beírt lekérdezéseket, kérdéseket, amely következtében felhasználja a tényeket és szabályokat a tudásbázisból, annak érdekében, hogy a lehető legmegfelelőbb választ adja vissza a felhasználónak. Azaz következtető motor a szabályokat a tudásbázisból szerzi be, és az ismert tényekre alkalmazza őket, hogy új tényekre következtessen. Számos motornak magyarázó és hibakeresési képességeket is van, ugyanakkor fontos megjegyezni, hogy működését nem befolyásolja a szabályok száma vagy típusa.

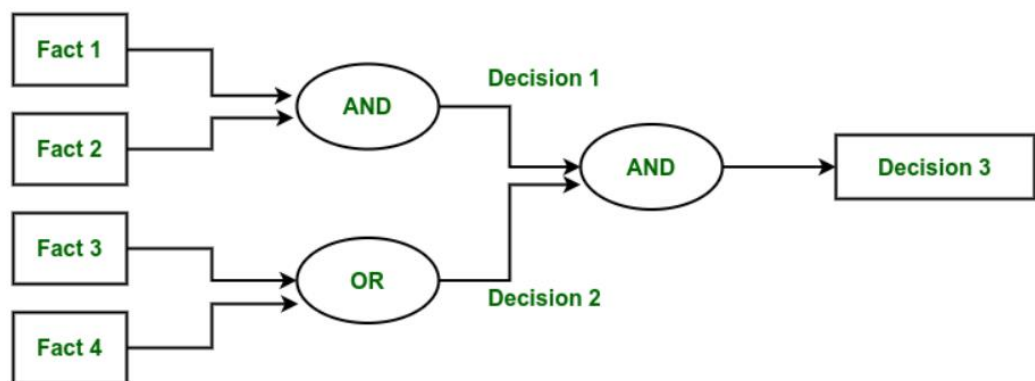
A következtető motor kétféle stratégiát használhat a tudásbázisból történő ismeretszerzés kinyerésére és a kívánt válasz megtalálására. Ezek a Forward Chaining és a Backward Chaining nevű algoritmusok, melyeket részletesebben bemutatok a következő alfejezetekben.

4. 1. 2. 1. Forward Chaining

A forward chaining másneven előre láncoló algoritmus az ismert tényekből indul ki, mely a lentől felfelé stratégiát követi, alulról felfelé haladva. Elindítja az összes olyan szabályt, amelynek a premisszái teljesülnek, és a következtetésüket hozzáadja az ismert tényekhez. Ez a folyamat addig ismétlődik, amíg a probléma meg nem oldódik. Ennél a típusú láncolásnál a következtetési motor a meglévő tények, levezetések és feltételek kiértékelésével kezd, mielőtt új információkat vonna le. A végpontot vagy célt a tudásbázisban meglévő tudás manipulálásával érjük el. (Ajlan S. Al-Ajlan, 2015)

Az algoritmus többféle következtetés levonására használható, mely széleskörű keresést alkalmaz, mivel először az összes tényt át kell néznie. A módszert adatvezéreltnek is nevezik, mivel célunkat a rendelkezésre álló adatok felhasználásával érjük el. Széles körben használják a szakértői rendszerekben, mint például a CLIPS, valamint az üzleti szabályrendszerekben és a gyártási típusú szabályrendszerekben is.

Pl. Mi fog történni legközelebb? Ezt a stratégiát leginkább olyan feladatok kezelésére használják, mint a következtetés vagy egy eredmény létrehozása. Pl. előrejelzések vagy részvénytípi mozgásállapotok.



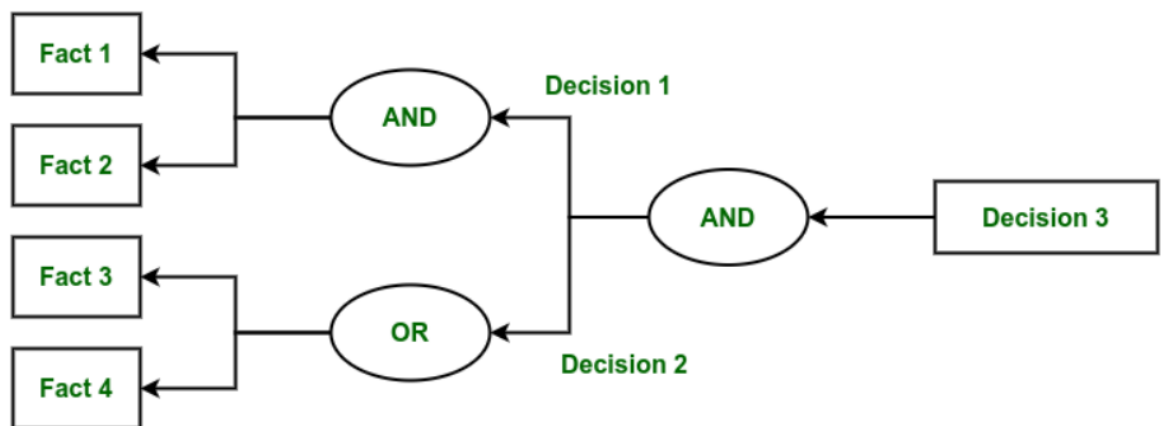
Ábra 4.2. Forward Chaining működési elve

4. 1. 2. 2. Backward Chaining

A backward chaining vagy másneven visszafelé történő láncolás típusú algoritmusnál a következtetési motor ismeri a végső döntést vagy célt. A rendszer a célból indul ki és visszafelé haladva határozza meg, hogy milyen tényeket kell állítani ahhoz, hogy a célt el lehessen érni. Közvetlenül a következtetésből (hipotézisből) indul ki, és azt a tények sorozatán keresztül visszafelé haladva érvényesíti. A visszafelé történő láncolás használható hibakeresés, diagnosztika vagy felírási alkalmazásoknál.

Az algoritmus egy felülről lefelé haladó stratégiát használ, melyben a modus ponens következtetési szabály szolgál a visszaláncolás alapjául. A visszaláncolás során a célt részcélokra bontjuk, hogy bebizonyítsuk a tények igazságát. Ezt nevezzük célvezérelt megközelítésnek, mivel a célok listája határozza meg, hogy mely szabályok kerülnek kiválasztásra és felhasználásra.

Ezt a stratégiát leginkább a kiváltó ok vagy okozat kiderítésére használják, figyelembe véve a már megtörtént eseményeket. Példa Egy betegség vagy gyomorfájás stb. kiváltó okának keresése. Ugyanakkor találkozhatunk vele a játékelméletben, az automatikus tételbizonyító eszközökben, következtetőmotorokban, bizonyítási asszisztensekben, valamint különböző mesterséges intelligencia alkalmazásokban is.



Ábra 4.3. Backward Chaining működési elve

4. 1. 3. Felhasználói interfész

A felhasználói interfész az a pont, ahol az emberi felhasználók kapcsolatba lépnek a számítógéppel. Biztosítja a felhasználó és a szakértői rendszer közötti információs kapcsolatot. Segítségével könnyebben és egyszerűbben tudja használni az átlagember is az elkészített alkalmazást.

4. 1. 4. Magyarázó modul

Ez a modul segít a szakértői rendszernek, hogy magyarázatot adjon a felhasználónak arról, hogyan jutott a szakértői rendszer egy adott következtetésre.

4. 1. 5. Tudásszerzési és tanulási modul

Ennek a komponensnek az a feladata, hogy lehetővé tegye a szakértői rendszer számára, hogy egyre több és több tudást szerezzen különböző forrásokból, és azt a tudásbázisban tárolja.

4. 2. Szakértői rendszerek típusai

Évtizedes erőfeszítések után az emberek fokozatosan kifejlesztettek egy sor különböző elveken alapuló szakértői rendszert. A szabályalapú szakértői rendszertől a neurális hálózaton alapuló szakértői rendszerig mindenik típusnak megvannak a maga előnyei.

Működési elvük szerint a szakértői rendszerek négy osztályba sorolhatóak: Szabályalapú szakértői rendszerek, Framework alapú szakértői rendszerek, Fuzzy logikán alapuló szakértői rendszerek és Neurális hálózaton alapuló szakértői rendszerek.

4. 2. 1. Szabályalapú szakértői rendszerek

A szakértői rendszerek típusai közül a legkorábbi, amely az egyik legkézenfekvőbb a fejlesztők számára amennyiben létrehozni kívánt alkalmazás szabályokra épül. Nehezen

algoritmizálható problémák megoldásánál, hatékony eszközt biztosítanak, ezért elterjedten alkalmazzák számos terület problémáinak megoldására. (Crina Grosan & Ajith Abraham, 2011)

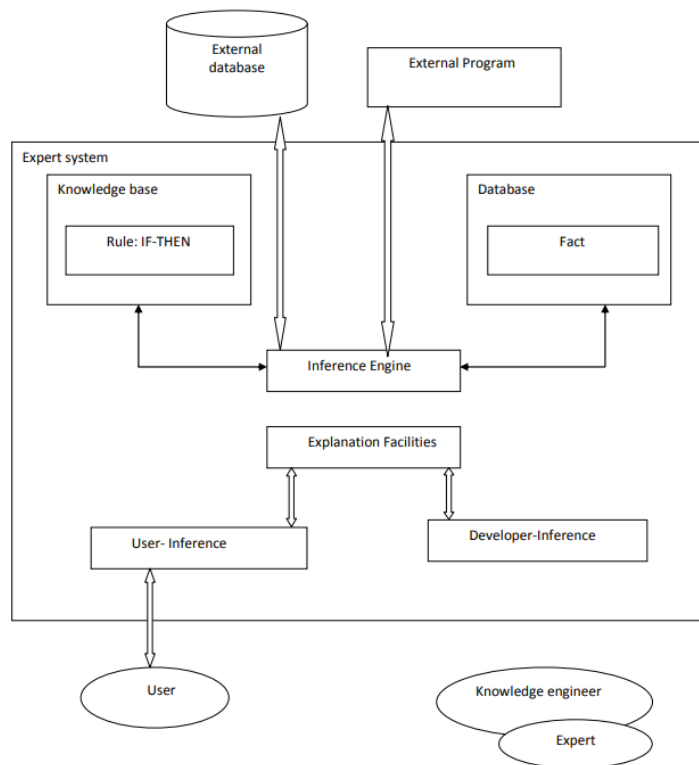
Felépítésükben a legelterjedtebb a szabálybázisú tudás, melyben a tudás, szabályok sorozatában van képviselve. Egy szabály felépítése az IF feltétellel kezdődik, melyet a THEN, azaz a konklúzió követ. A feltételben az AND, OR és a NOT utasításokat is megadhatjuk, vagy ezek különböző kombinációit.

Szabályalapú szakértői rendszerek felépítése:

A szabályalapú szakértői rendszerek öt komponensből állnak (tudásbázis, az adatbázis, a következtetési motor, a magyarázó modul és a felhasználói felület), melyek nagyrészt az előző alfejezetben részletesebben bemutatam. Említésre méltó az adatbázis, amely azokat a tényeket tartalmazza, amelyeket az IF-THEN szabályokkal való összevetéshez használunk.

További kiegészítő komponensek közé tartozhat még a külső interfész, a fejlesztői felület, a szövegszerkesztő és a futásidejű ismeretszerzés. A külső interfész lehetővé teszi, hogy a szakértői rendszer interakcióba lépjen más adatbázisokkal és programokkal. A fejlesztői felület lehetővé teszi a fejlesztő számára, hogy bővítse a tudásbázist új szabályokkal és tényeket vagy módosítsa őket. A futásidejű ismeretszerzési lehetőség lehetővé teszi programfutas közben az ismeretek vagy tények hozzáadását, amelyek nem állnak rendelkezésre a tudásbázisban vagy adatbázisban.

A szabályalapú szakértői rendszerek struktúráját segít jobban megérteni az alábbi 4.4. ábra.



Ábra 4.4. Szabályalapú szakértői rendszer felépítése

4. 2. 2. Framework alapú szakértői rendszerek

Míg a szabályalapú szakértői rendszerben és a fuzzy szakértői rendszerben az IF- THEN szabályokat használják a tudás reprezentálására, addig a Framework alapú, másneven keretalapú szakértői rendszerben kereteket használnak a tudás reprezentálására. Ez lehetővé teszi számunkra, hogy a tudást hierarchikus módon reprezentáljuk

4. 2. 2. 1. Framework vagy Keret meghatározása

A keret egy olyan adatszerkezet, amely egy adott objektumra vagy fogalomra vonatkozó ismereteket tartalmaz. Egy keretalapú szakértői rendszerben a kereteket a tudás rögzítésére és reprezentálására használják. Itt minden keretnek saját neve és attribútumkészlete van. A keret módot biztosít a tudás strukturált és tömör ábrázolására, így egy adott objektummal

vagy fogalommal kapcsolatos összes szükséges tudást egyetlen egységben egyesíthetjük. A keretek előnye a szabálybázissal szemben az, hogy a szabályok végrehajtásához csak a keretekben kell keresni, ellentétben a szabálybázisú szakértői rendszerrel, ahol az összes szabályon végig kell keresni. (John H. Bradley, Richard D. Hauser Jr, 1999)

A keretet a slotok gyűjteménye határozza meg. Minden egyes slot a keret egy adott attribútumát vagy műveletét írja le. A slotok értékek tárolására szolgálnak, ahol a slot tartalmazhat egy alapértelmezett értéket vagy egy másik frame-re mutató mutatót.

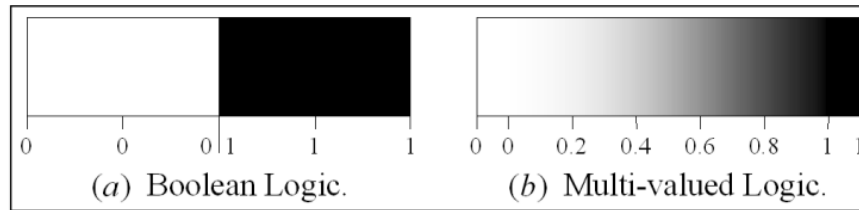
4. 2. 2. 2. Következtető motor működése a Framework (keretalapú) szakértői rendszerben

Egy keretalapú szakértői rendszerben a következtetőmotor a célt vagy egy adott attribútumot keres, míg a szabály másodlagos szerepet játszik. A következtetőmotor megkeresi azokat a szabályokat, amelyek következményei tartalmazzák a keresett célt és a szabálybázis sorrendjében egyesével átvizsgálja őket.

4. 2. 3. Fuzzy logikán alapuló szakértői rendszerek

Amikor olyan szakértői tudást próbálunk kifejezni, amely nem egyértelmű szavakat használ, mint például az "enyhén túlterhelt", "mérsékeltlen nehéz", "nem túl régi", vagy "nagyon magas", akkor használhatjuk a fuzzy halmazelméletet. A fuzzy logika alapja azon az elképzelésen alapul, hogy minden dolgot egy skálán írunk le, ezért többértékű logikának is nevezik. (Szilard Madaras, 2022)

Klasszikus bináris logikánál két, az Igaz (1) és Hamis (0) értékekkel találkozhatunk, míg a fuzzy logikánál az összes értéket valós számokkal fejezik ki a 0 és 1 közötti intervallumból. Az intervallumban található számok azt a lehetőséget jelölik, hogy egy adott állítás mennyire igaz vagy hamis. Az alábbi 4.5. ábra bemutatja a két logika közti intervallum különbséget.



Ábra 4.5. Értékek tartománya Boolean és Fuzzy logikáknál

A következő egyszerű példa jól szemlélteti, hogyan működik a Fuzzy logika a klasszikus bináris logikához viszonyítva, egy auto sebességének megadásánál.

Klasszikus bináris logikánál egy konkrét értéket adunk meg a programnak, amely alapján meghatározza, hogy az autó gyorsan vagy lassan megy.

1. szabály:

IF sebesség > 120

THEN féktávolság hosszú

2. szabály:

IF sebesség < 30

THEN féktávolság rövid

Ellenben a Fuzzy logikánál a konkrét érték helyett már egy jelzőt adunk meg a programnak, a féktávolság hosszának eldöntéséhez.

1. szabály:

IF sebesség gyors

THEN féktávolság hosszú

2. szabály:

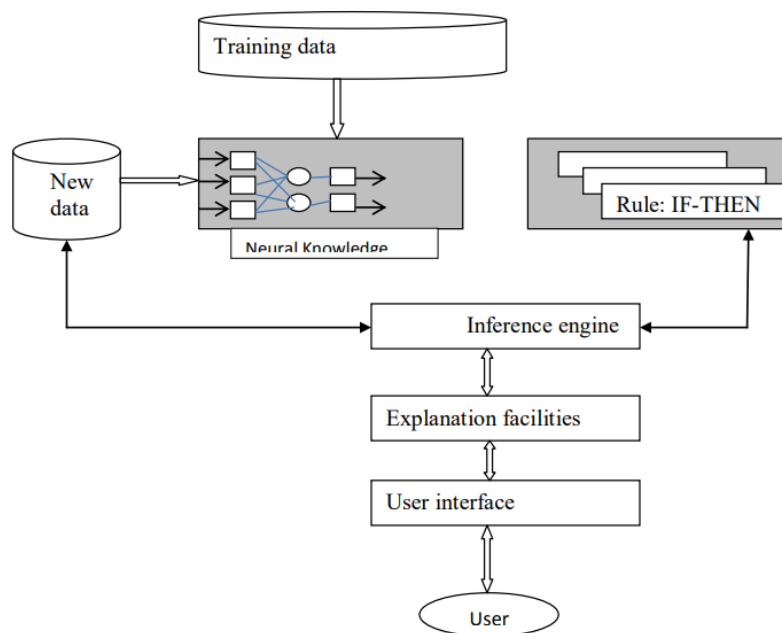
IF sebesség lassú

THEN féktávolság rövid

4. 2. 4. Neurális hálózaton alapuló szakértői rendszerek

A neurális hálózatok és a szakértői rendszerek közös célja egy emberi szakértő döntéshozatali képességének utánzása egy megadott területen. Egy hibrid szakértői rendszert, amely egy neurális hálózatot és egy szabályalapú szakértői rendszert kombinál, neurális szakértői rendszernek nevezzük. Itt a szabályalapú szakértői rendszer előnyeivel kombináljuk a neurális hálózat előnyeit, mint például a tanulás, általánosítás, robusztusság vagy párhuzamos információfeldolgozás. (Riu Mia, 2009)

4. 2. 4. 1. Neurális hálózaton alapuló szakértői rendszer felépítése



Ábra 4.6. Neurális hálózaton alapuló szakértői rendszer felépítése

A neurális szakértői rendszer neurális tudásbázissal rendelkezik a hagyományos tudásbázis helyett, ahol a tudás „neuronok” formájában tárolódnak. A szabály-kivonatoló komponens kombinációja a neurális hálózattal, lehetővé teszi a neurális szakértői rendszernek, hogy képes legyen a következtetéseit igazolni és magyarázattal ellátni. (Yaser A. Jasim, 2013)

Ugyanakkor a neurális szakértői rendszer lehetővé teszi a hiányos adatok kezelését is, mivel képes az általánosításra. Ennélfogva lehetővé teszi a megközelítő következtetést. A szabálykinyerő egység megvizsgálja a neurális tudásbázist, majd előállítja a neurális hálózatban beégetett szabályokból a leginkább megfelelő választ az adott kérdésre.

4. 2. 4. 2. Szabályok kinyerése a neurális szakértői rendszerből

A hálózat neuronjait linkek kötik össze, amelyek mindegyikéhez egy-egy numerikus súly tartozik. A súlyok egy betanított neurális hálózatban meghatározzák a kapcsolódó neuronok bemeneteinek erősségét vagy fontosságát.

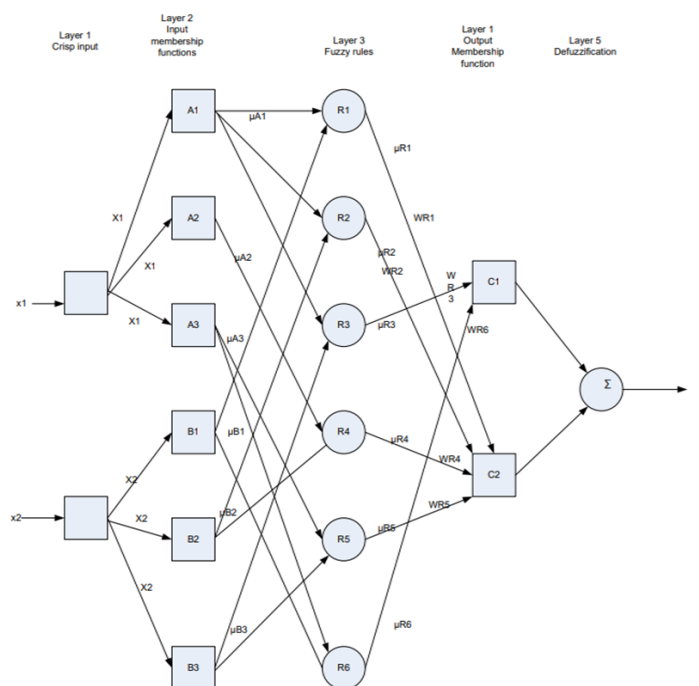
A neurális hálózat képzését egy adott adathalmaz alapján végezhetjük a “back propagation” képzési algoritmus segítségével. A kezdeti képzési fázis befejezése után megvizsgálhatjuk a neurális hálózat tudásbázisát, valamint kinyerehtjük és finomíthatjuk a kezdeti IF-THEN szabályok halmazát. Tehát a neurális szakértői rendszer kétirányú kapcsolatot biztosít a neurális hálózatok és a szabályalapú rendszerek között.

Az egyetlen probléma a neurális szakértői rendszerekkel az, hogy nem tudnak folyamatos bemeneti változót reprezentálni, mivel ez végtelen szabályok létrehozásához és végrehajtásához vezetne. Ezt a problémát küszöbölhetjük ki a következő fejezetben olvasható neuro-fuzzy rendszerekkel.

4. 2. 4. 3. Neuro-fuzzy rendszerek

A neuro-fuzzy fúziója egy olyan hibrid szakértői rendszert eredményez, amely ötvözi a fuzzy rendszerek emberhez hasonló gondolkodási stílusát a neurális hálózatok tanulási és konnektív struktúrájával. (S.G. Tzafestas, Anastasios N. Venetsanopoulos, 1994)

A neuro-fuzzy szakértői rendszer általános felépítését az alábbi 4.7. ábra szemlélteti. Amint látható a bemeneti és kimeneti rétegeken kívül az alábbi példán, három rejtett réteg is található, amelyek függvényeket és fuzzy szabályokat tartalmaznak.



Ábra. 4.7. Minta egy neuro-fuzzy szakértői rendszer felépítésére

4. 3. Az intelligens és szakértői rendszerek felhasználása:

Napjainkban a mesterséges intelligencia szinte minden iparágat áthatott, forradalmasítva a vállalkozások működését. Az egészségügytől a pénzügyeken át, egészen a gyártásig és a közlekedésig, a mesterséges intelligencia ösztönzi az innovációt, racionalizálja a folyamatokat, lehetővé teszi az adatvezérelt döntéshozatalt de ami a legfontosabb összességében, hogy megkönnyíti az emberek munkáját.

Ezen rendszereket széles körben használják a fent már említett területeken. Az alábbiakban összegyűjtöttem néhány ágazatot, melyekben aktívan használják a szakértői rendszereket:

Az első és talán legfontosabb ágazat az egészségügy, ahol a szakértői rendszereket arra használják, hogy segítsék az orvosok munkáját a betegek diagnosztizálásában, valamint a tünetek és a páciens kórtörténete alapján kezelési lehetőségeket javasoljanak. A következő ágazat a különböző termékek gyártása és a termelés. Itt a szakértői rendszereket gyártóüzemekben, minőségellenőrzésben és a folyamatok optimalizálásában is használják. Pl. robotika és automatizálás, autonóm rendszerek, diagnosztika és javítás, készletgazdálkodás, vizuális ellenőrzés. Az ügyfélszolgálatban is jelen vannak a szakértői rendszerek, mivel számos nagyvállalat használ ilyen típusú rendszereket automatizált ügyfélszolgálathoz, amely segíti a felhasználókat a különböző problémák elhárításában. Mezőgazdaságban lényeges a gazdának a termék értékesítési árán kívül, a megtermelt termés minősége és hozama. Ennek érdekében a szakértői rendszerek a gazdák segítségükre lehetnek, hogy hasznos tanácsokat kapjanak a növénytermesztés különböző fázisaiban, a kártevők elleni védekezésben vagy a talaj állapotával kapcsolatban. Forgalomirányítási rendszerekben, autonóm járművekben döntéshozatalra és útvonal-optimalizálásra is használják őket. Pl. forgalomáramlás-elemzés és torlódásfigyelés, szállítási flották koordinálása, tömegközlekedési rendszerek, balesetmegelőzés. Így kijelenthetjük, hogy a közlekedésben is fontos szerepet töltenek be a szakértői rendszerek.

Pénzügyi szektorban is találkozhatunk a szakértői rendszerekkel, mivel a pénzügyi intézményekben kockázatértékelésre, csalásfelismerésre és befektetési tanácsadásra is használják őket. Az oktatásban ezen rendszereket e-tanulási platformokon is használják, személyre szabott tanulási tapasztalatok és adaptív oktatás nyújtására. Pl. Korrepetálás, a tanulók tanulási mintáinak azonosítása, kérdés- és válaszrendszerek létrehozása, kurzustervezés és ütemezés, vagy interaktív online oktatás.

A fent említett ágazatok csak néhány példa, mivel ezen rendszerek tudását a technológia fejlődésével folyamatosan új területeken tudják majd alkalmazni és hasznosítani.

5. Saját alkalmazás bemutatása

Egy vállalkozás indítása előtt, vagy vezetése közben, nagyon fontos, hogy a vezető a lehető legfelkészültebb legyen többek között közgazdasági, vállalatirányítási témákban, különböző üzleti elemzések elkészítésében. Elkészített programom egy előreirányított neurális hálózaton alapuló szakértői rendszer, amely vállalkozásindítás, marketing és üzleti tervezés kapcsán nyújt tanácsadást, azon személyeknek, akik szeretnének vállalkozást indítani vagy szeretnének több információhoz jutni.

Az elkövetkezendőkben részletesen bemutatom az alkalmazásom elkészítésénél felhasznált technológiákat, valamint az alkalmazás működését.

5. 1. Felhasznált technológiák

A szakértői rendszerem elkészítésében különböző technológiák voltak segítségemre. A felhasználói felület megalkotásához HTML5 és CSS stílusleíró nyelvet használtam. A program működéséhez, azaz a Back-end részhez a Python programozási nyelvet, valamint a PyTorch nevű nyílt forráskódú, gépi tanulási könyvtárat használtam, amelyet neurális hálózat alapú “Deep Learning” mély tanulási modellek fejlesztésére és képzésére használnak.

Egy program fejlesztéséhez, követéséhez elengedhetetlen a verziókövetés, melyben a GitHub verziókövető rendszert használtam. Amennyiben az olvasó kíváncsi, akármikor elérheti és letöltheti a szakértői rendszerem a következő GitHub linkre kattintva:

https://github.com/A-Zsombi/Allamvizsga_Python

5. 1. 1. HTML 5

Weboldalak készítéséhez, valamint a világhálón a tartalom megjelenítéséhez és strukturálásához az egyik alapfogalom a HTML5 jelölőnyelv. Az HTML5 elemek használatával szöveget, képeket, űrlapokat jeleníthetünk meg egy weboldalon. A HTML 5.

verziója segít a fejlesztőknek még interaktívabbá és dinamikusabbá tenni a weboldalakat és webes alkalmazásokat. (<http://www.w3.org/TR/REC-html40-971218/conform.html>, 1997)

5. 1. 2. CSS3

Az weboldalak esztétikusabbá tételéhez a CSS stílusleíró nyelvet használtam, amelynek segítségével személyre szabható a HTML5 weboldalak elemei. Felhasználásával szétválasztható egy oldal stílusa a tartalomtól. Ez a szétválasztás javítja a tartalom átláthatóságát, így könnyítve a megjelenítési elemek fejlesztését. (<http://www.w3.org/standards/webdesign/htmlcss#whatcss>, 2010) Ugyanakkor a stílusok külön .css fájlban történő tárolása, csökkenti a szerkezeti tartalom összetettségét és ismétlődését, ezáltal lehetővé téve a .css fájl gyorsítótárba helyezését, amely javítja a weboldal betöltési sebességét.

5. 1. 3. GitHub

Fejlesztés közben a felmerülő hibák előfordulásakor szükséges egy verziókövető rendszer, mely lehetőséget biztosít az előzőekben fejlesztett program biztonsági mentésének eléréséhez. Az oldalt 2008-ban indította Tom Preston-Werner, Chris Wanstrath és PJ Hyett.

A GitHub egy internetes tárhelyszolgáltató, szoftverfejlesztők számára, amely számos funkciót nyújt a programozóknak, mint például verziókövetést, feladatkezelést, hibakövetést valamint dokumentálást minden feltöltött projekthez. Ugyanakkor nyílt forráskódú szoftverfejlesztési projektek tárolására használják.

5. 1. 4. Python

Az egyik legnépszerűbb programozási nyelv, melyet használhatjuk szerveroldali webes alkalmazások létrehozására, szoftverfejlesztésre, nagy mennyiségű adat kezelésére és összetett matematikai feladatok elvégzésére, prototípusok készítésére, valamint rendszer scriptingre is. Számos előnye közé sorolható, hogy különböző platformokon is működik pl.

Windows, Mac, Linux, Raspberry Pi stb., ugyanakkor többféle programozási paradigmát is támogat, beleértve a strukturált, az objektum orientált és a funkcionális programozást.

Előszeretettel használják Machine Learning területén a fejlesztők, a nagyméretű szabványkönyvtárának köszönhetően. (Piotrowski, Przemyslaw, 2012) Ezen szabványkönyvtára számos feladatra alkalmas eszközöket kínál, és ezt szokták az egyik legnagyobb erősségeként említeni a Python-nak. Az internetre irányuló alkalmazások esetében számos szabványos formátum és protokoll, például a MIME és a HTTP támogatott. Tartalmaz modulokat grafikus felhasználói felületek létrehozásához, relációs adatbázisokhoz való csatlakozáshoz, tetszőleges pontosságú tizedesjegyekkel való aritmetikához, reguláris kifejezések kezeléséhez és egységteszteléshez. (Batista, Facundo, 2008)

5. 1. 5. PyTorch

A PyTorch egy Machine Learning, azaz Gépi Tanulási könyvtár a Python programozási nyelvben, amelyet olyan alkalmazásokhoz használhatunk mint a számítógépes látás vagy a természetes nyelvi feldolgozás. A PyTorch-ra számos Deep Learning szoftver épül, többek között a Tesla Autopilot, az Uber Pyro, az Hugging Face Transformers, vagy a PyTorch Lightning. (Lorica, Ben, 2017)

A Torch.nn nevű modulja a neurális hálózatok fejlesztésére és ezek tudásbázisainak a fejlesztésére szolgál. A modul a neurális hálózatok építőelemeinek átfogó gyűjteményét kínálja, beleértve a különböző rétegeket és aktiválási függvényeket, ezáltal lehetővé téve komplex modellek felépítését. A hálózatok felépítése a torch.nn modul öröklésével és a forward függvényben a műveletek sorrendjének meghatározásával történik.

5. 1. 6. Natural language processing (Természetes nyelvi feldolgozás)

A Python programozási nyelvet gyakran használják többek között természetes nyelvi feldolgozásra. A kifejezés elsősorban arra a felhasználási módra utal, hogy a számítógépeket képessé tegyük az emberi nyelv támogatására és kezelésére. Bár a természetes nyelvi feldolgozási feladatok szorosan összefonódnak, az egyszerűség kedvéért több kategóriákra

osztthatjuk: Szöveg- és beszédfeldolgozás, Morfológiai elemzés, Szintaktikai elemzés, Lexikális szemantika vagy Relációs szemantika. Ezek számos feladatot foglalnak magukban, mint például: beszédfelismerés, szöveg összegzés, gramatikai javítás, szöveg írás, megadott szövegből kép készítés, dialógus folytatása egy emberrel, stemmingelés stb. (Mikolov, Tomáš, 2010)

A felsorolt feladatok közül néhánynak közvetlen valós alkalmazásai vannak, míg mások inkább részfeladatokként szolgálnak, amelyeket nagyobb feladatok megoldásának segítésére használnak.

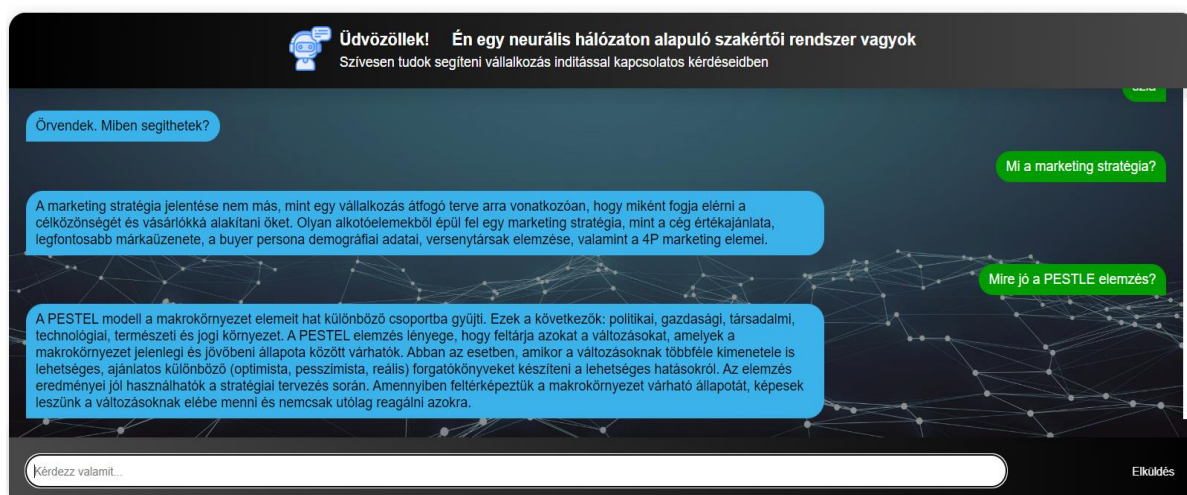
5. 2. Alkalmazás működése

Ebben a fejezetben az olvasónak lehetősége van megismerni az elkészített szakértői rendszert, amely webes felülettel könnyíti meg a felhasználónak a tudászerzést.

Működésének a bemutatását a következő három alfejezetben csoportosítottam.

5. 2. 1. Felhasználói felület

Az alábbi 5.1 ábrán látható a felhasználói felület, mely segítségével a felhasználó, kapcsolatba léphet az alkalmazással.



Ábra 5.1. Alkalmazás felhasználói felülete

Az alkalmazást megnyitva a felhasználó az alsó input mezőbe írhatja kérdéseit. Ezt követően az Enter vagy az Elküldés gomb megnyomásával a program elkezd feldolgozni a beírt információt, majd az általa legmegfelelőbb választ adja a beírt kérdéseinkre.

A felhasználó számos kérdésére kaphat választ vállalkozásindítás, marketing és üzleti tervezés kapcsán, mint például "Milyen költségekkel jár egy SRL indítása?", "Mik a CAEN kódok?" "Miből áll a marketing stratégia felépítése?", "Melyek a PESTEL elemzés lépései?" "Mi a szerepe a piaci pozicionálásnak?" "Mit jelent a SWOT analízis?" "Mi a marketing mix?" stb.

5. 2. 2. Tudásbázis

Fontos volt számomra, hogy az alkalmazás megbízható, könnyen érthető és hasznos információkkal lássa el felhasználóját. Mivel az alkalmazás vállalkozásindítás, marketing és üzleti tervezés kapcsán nyújt információkat, ezért az alkalmazás tudásbázisának elkészítésekor számos értékes publikációt, irodalmat használtam fel. A legfontosabbakat említve, mint például Balaton Károly - Tari Ernő: Stratégiai és üzleti tervezés vagy Philip Kotler - Kevin Lane Keller: Marketing menedzsment stb. Fontos megjegyezni, hogy a tudásbázis nem a teljes üzleti és marketing irodalmat tartalmazza. Nem is ez volt a lényege a projektnek vagy egy Chat GPT létrehozása, hiszen az, óriási munkát vonna maga után. A cél a szakértői rendszerek bemutatásán túl, egy olyan alkalmazás létrehozása volt, amely használható információkkal látja el a felhasználóit az előbb említett ágazatokban.

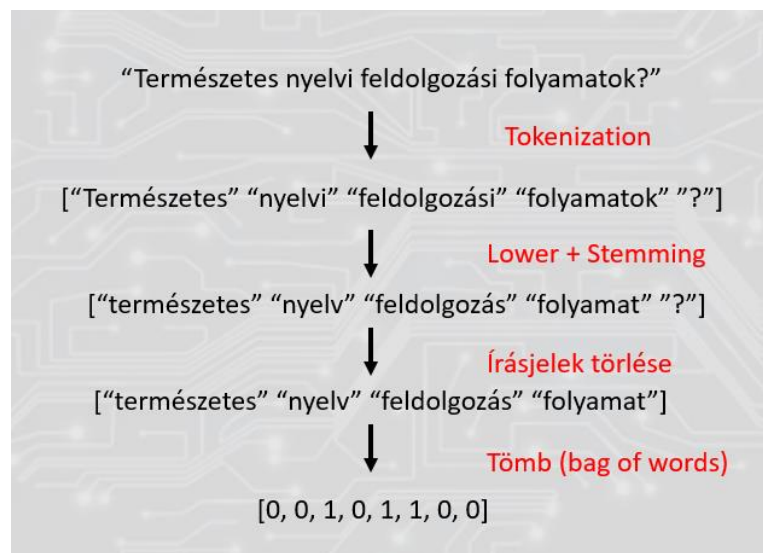
Az alkalmazás tudásbázisának inicializálásához, természetes nyelvi feldolgozást használtam. Ezen fogalom szerepét és felhasználási formáit az előző, Felhasznált technológiák alpontban részletesen bemutatam. A programom a Tokenization – Tokenizáció lépéssel kezd, amely a megadott karakterláncot egységekre, azaz külön szavakra bontja. Ezt követően a Stemming következik, amely a feldarabolt karakterláncban megadott szavaknak úgy mond levágja a végét, meghagyva a szó gyök alakját. Ehhez a "Snowball Stemmer" nevű algoritmust használtam, amely már számos nyelvet képes kezelni, köztük a magyart is. Erre azért van szükség, mivel a felhasználó különböző formákban is beírhat egy adott szavat és azt nem biztos, hogy a program tudná értelmezni.

Jó példa a Stemming bemutatására, az alábbi ábrán látható “változat” szó, amely feldolgozás előtt és utáni formáit szemlélteti.



Ábra 5.2. Stemming folyamata

A Stemming befejeztével a megadott mondatban levő összes karaktert kisbetűsítem, majd a különböző írásjeleket törölöm a mondatból. Ezt követően az összes lehetséges szöveg minta szavait, feldarabolom egy tömbbe. Ez a tömb tartalmazza a szakértői rendszer tudástárában előforduló összes szavat.

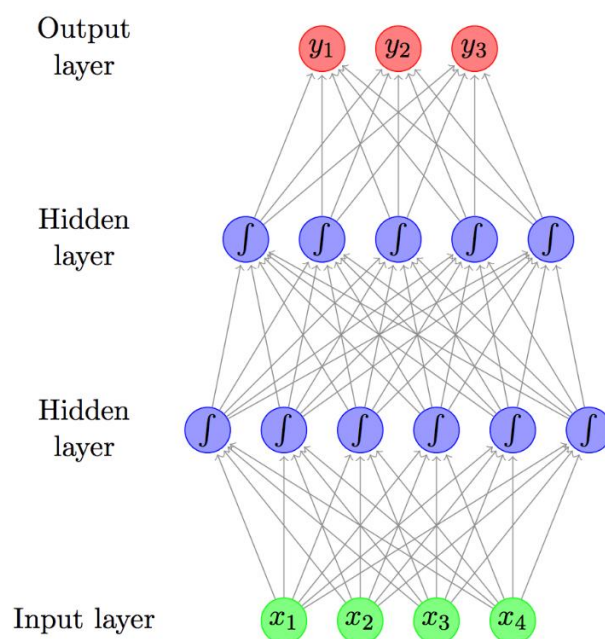


Ábra 5.3. Természetes nyelvi feldolgozás folyamata

Miután a felhasználó begépel az alkalmazásnak egy szöveget, az értelmezést követően amelyik válasznak a tudásbázisában a legnagyobb súlyértéket számolja ki, azt a választ adja vissza a felhasználónak a program. Ezt a folyamatot az alábbiakban részletesebben bemutatom.

5. 2. 3. Következtető motor

Az alkalmazás egy Feedforward Neural Network (Előreirányított neurális hálózat) struktúra alapján működik, melyben az információáramlás egyirányú, azaz a modellben az információ csak egy irányban áramlik. A következő ábra szemlélteti, ahogyan a megadott információ, a bemeneti csomópontoktól a rejtett csomópontokon keresztül a kimeneti csomópontok irányába halad.



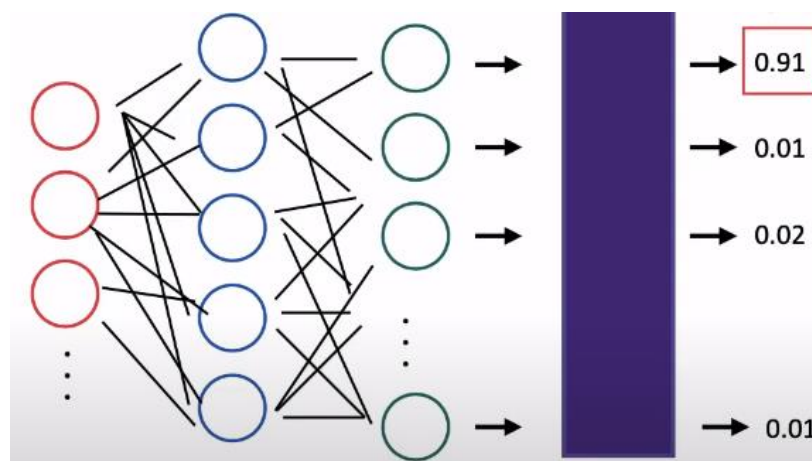
Ábra 5.4. Információáramlás folyamata és iránya

A bemenetek sorozata belép a rétegbe és ebben a modellben megszorzódnak a súlyokkal. A súlyozott bemeneti értékek ezután összegződnek, és így kapunk egy végösszeget. Ezt követően a neurális hálózat összehasonlítja a csomópontjainak kimeneteit a kívánt

értékekkel, így a hálózat a képzés során módosíthatja a súlyait, hogy pontosabb kimeneti értékeket hozzon létre.

Ez a modell a bemeneteket súlyokkal szorozza meg, amint azok belépnek a rétegbe. Ezután a súlyozott bemeneti értékeket összeadjuk, hogy megkapjuk az összeget. Amíg az értékek összege egy bizonyos, nulla értékre beállított küszöbérték fölé emelkedik, addig a kimeneti érték általában 1, míg ha a küszöbérték alá esik, akkor általában -1 lesz.

A neurális hálózat utolsó rétegében a Softmax metódust használtam. Ezt a funkciót használják nyelvi modellekben, gépi fordításban, és szövegosztályozásban is, különböző nyelvi kimenetek vagy osztályok feletti valószínűségi eloszlások létrehozására. A függvény a kimeneti vektorok valószínűségi formába való átalakítására szolgál. Ezeket a kimeneti eredményeket 0 és 1 közötti tartományba teszi, mely eredmények összegének az összértéke 1 lesz. Ezt próbálja szemléltetni a következő, 5.5. ábra és az azt követő 5.6. ábra a választadó metódussal:



Ábra 5.5. Softmax metódus

```

def get_response(msg):
    sentence = tokenize(msg)
    X = bag_of_words(sentence, all_words)
    X = X.reshape(1, X.shape[0])
    X = torch.from_numpy(X).to(device)

    output = model(X)
    _, predicted = torch.max(output, dim=1)

    tag = tags[predicted.item()]

    probs = torch.softmax(output, dim=1)
    prob = probs[0][predicted.item()]
    if prob.item() > 0.75:
        for intent in intents['intents']:
            if tag == intent["tag"]:
                return random.choice(intent['responses'])

```

Ábra 5.6. Választ adó metódus

6. Következtetések

A jövőre nézve egyértelmű, hogy az intelligens rendszerek továbbra is jelentős és egyre nagyobb szerepet fognak játszani életünk számos területén. Az intelligens otthonoktól és az önvezető autóktól kezdve, a fejlett orvosi diagnózisokig és a személyre szabott ajánlásokig ezek a rendszerek már most is átalakították életvitelünket és munkánkat. Fejlesztésük, elfogadásuk és a társadalomra gyakorolt hatásuk körül azonban még mindig számos kérdés vet fel a jövőre nézve.

Egy dolog biztos, az intelligens rendszerek iránti kereslet csak tovább fog nőni, ahogy egyre több vállalkozás és iparág igyekszik kihasználni előnyeiket. Így valószínűleg növekedni fognak a kutatási és fejlesztési erőfeszítések, amelyek célja e rendszerek pontosságának, hatékonyságának és biztonságának javítása. Ez olyan területeken vezethet áttörésekhez, mint a természetes nyelvfeldolgozás, a gépi tanulási algoritmusok, a robotikai technológia és még sok más. Jelenleg az ágazat fejlődése nem mutatja a lassulás jeleit, így várhatóan továbbra is átalakítja az iparágakat, megkönnyítve az ember-gép együttműködést egy intelligensebb és termelékenyebb jövő érdekében.

Végezetül az összegzés maradt hátra, a dolgozatomban címében levő kérdésre, hogy mennyire tud egy szakértői rendszer segíteni és tanácsokat adni vállalkozás indítási kérdésekre. Kijelenthető, hogy tud, célirányosan, viszont a program tudásbázisának feltöltése, folyamatos fejlesztése és naprakészen tartása nagyon sok munkát igényel.

Irodalomjegyzék

1. Haocheng Tan, A brief history and technical review of the expert system research, Purpose Led Publishing, 2017
2. Berk, A A. LISP: the Language of Artificial Intelligence. New York: Van Nostrand Reinhold Company, 1985
3. Yu, Victor L. "Antimicrobial Selection by a Computer", JAMA The Journal of the American Medical Association, 1979
4. Buchanan, B.G.; Shortliffe, E.H., Rule Based Expert Systems: The MYCIN Experiments of the Stanford Heuristic Programming Project, Publisher: Addison-Wesley, 1984
5. David C. Brock, Learning from Artificial Intelligence's Previous Awakenings: The History of Expert Systems, AI Magazine, 2018
6. McDermott, John "R1: An Expert in the Computer Systems Domain", AAAI Conference on Artificial Intelligence, 1980
7. Chandrasekaran, Rajiv „Kasparov Proves No Match for Computer”. Washington Post, 1997
8. Peter Menzel and Faith D'Aluisio. Robosapiens, The MIT Press, 2000.
9. Bosker, Bianca „SIRI RISING: The Inside Story Of Siri's Origins – And Why She Could Overshadow The iPhone”, Huffington Post, 2013
10. Fitzsimmons, Caitlin „Why Sophia the robot is not what it seems”, The Sydney Morning Herald, 2017
11. Szilárd Madaras, Norbert Ágoston “A rule-based expert system for financial investment decisions in Romania”, Researchgate, 2023

12. John H. Bradley, Richard D. Hauser Jr. "A framework for expert system implementation", Semantic Scholar, 1999
13. Szilard Madaras "Szakértői Rendszerek", Kolozsvári Egyetemi Kiadó, 2022
14. Yaser A. Jasim, Safwan Hasoon "Diagnosis windows problems based on hybrid intelligence systems", Journal of Engineering Science and Technology (JESTEC), 2013
15. "What is CSS?". World Wide Web Consortium. Forrás: <http://www.w3.org/standards/webdesign/htmlcss#whatcss>, 2010
16. Piotrowski, Przemyslaw "Build a Rapid Web Development Environment for Python Server Pages and Oracle", Oracle Technology Network, 2012.
17. Batista, Facundo, "PEP 327 – Decimal Data Type". Python Enhancement Proposals, 2008
18. Lorica, Ben "Why AI and machine learning researchers are beginning to embrace PyTorch". O'Reilly, 2017
19. Mikolov, Tomáš; Karafiát, Martin; Burget, Lukáš; Černocký, Jan; Khudanpur, Sanjeev "Recurrent neural network based language model", ResearchGate, 2010
20. "What is HTML?" World Wide Web Consortium, forrás: <http://www.w3.org/TR/REC-html40-971218/conform.html>, 1997
21. S.G. Tzafestas, Anastasios N. Venetsanopoulos, Fuzzy Reasoning in Information, Decision and Control Systems, Kluwer Academic Publishers 1994
22. Riu Mia, What is Neural Network Expert System, Neural Networks for Intrusion Detection, Beijing Institute of Technology, China, 2009
23. Crina Grosan & Ajith Abraham, Intelligent Systems, Springer, 2011

24. Ajlan S. Al-Ajlan, The Comparison between Forward and Backward Chaining, International Journal of Machine Learning and Computing, 2015
25. Russell, Stuart J. & Norvig, Peter, Artificial intelligence: a modern approach. Pearson series in artificial intelligence (Fourth ed.). Hoboken: Pearson, 2021
26. Patrick Grieve, Deep learning vs. machine learning, ZendeskBlog, 2023
27. Hao Karen, „The messy, secretive reality behind OpenAI's bid to save the world”. MIT Technology Review, 2020
28. Solomonoff, R.J., "The Time Scale of Artificial Intelligence; Reflections on Social Effects", Human Systems Management, 1985
29. Norvig; Peter, ELIZA: Paradigms of Artificial Intelligence Programming, San Francisco: Morgan Kaufmann Publishers, 1992
30. Jackson; Peter, Introduction To Expert Systems (3 ed.), Addison Wesley, 1998

NYILATKOZAT

A diplomadolgozat nyilvános hozzáféréséről és eredetiségéről

A hallgató neve: Albu Zsombor Domokos
A Hallgató Neptun kódja: YRLYAS
A dolgozat címe: Mennyire tud egy szakértői rendszer segíteni vállalkozásindításban?
A megjelenés éve: 2024
A konzulens intézetének neve: Magyar agrár és élettudományi egyetem
A konzulens tanszékének a neve: MATE, Befektetési, Pénzügyi és Számviteli Tanszék, Vidékfejlesztés és Fenntartható Gazdaság intézet

Kijelentem, hogy az általam benyújtott diplomadolgozat egyéni, eredeti jellegű, saját szellemi alkotásom. Azon részeket, melyeket más szerzők munkájából vettem át, egyértelműen megjelöltem, és az irodalomjegyzékben szerepeltettem.

Ha a fenti nyilatkozattal valótlan állítotam, tudomásul veszem, hogy a záróvizsga-bizottság a záróvizsgából kizár és a záróvizsgát csak új dolgozat készítése után tehetek.

A leadott dolgozat, mely PDF dokumentum, szerkesztését nem, megtekintését és nyomtatását engedélyezem.

Tudomásul veszem, hogy az általam készített dolgozatra, mint szellemi alkotás felhasználására, hasznosítására a Magyar Agrár- és Élettudományi Egyetem mindenkori szellemi tulajdon-kezelési szabályzatában megfogalmazottak érvényesek.

Tudomásul veszem, hogy dolgozatom elektronikus változata feltöltésre kerül a Magyar Agrár- és Élettudományi Egyetem MATER Hallgatói Dolgozatok repozitóriumba. Tudomásul veszem, hogy a megvédett és

- nem titkosított dolgozat a védést követően
- titkosításra engedélyezett dolgozat a benyújtásától számított 5 év eltelte után

nyilvánosan elérhető és kereshető lesz az Egyetem MATER Hallgatói Dolgozatok repozitóriumban.

Kelt: 2024.11.04.


Hallgató aláírása

NYILATKOZAT

Dr. Varga József László (hallgató Neptun azonosítója: YRLYAS) konzulenseként nyilatkozom arról, hogy a diplomadolgozatot áttekintettem, a hallgatót az irodalmi források korrekt kezelésének követelményeiről, jogi és etikai szabályairól tájékoztattam.

A záródolgozatot/szakdolgozatot/diplomadolgozatot/portfóliót a záróvizsgán történő
védelemre **javaslom**.

A dolgozat állam- vagy szolgálati titkot tartalmaz: igen nem*¹

Kelt: 2024.11.04.

belső konzulens

¹ A megfelelő aláhúzendő.