

Hibrid következtetőrendszerek

Hybrid inferencing architectures

KILLIÁN Imre

Deutsche Telekom Systems Solutions Hungary, Pécs
7621 Pécs, Nagy Lajos király útja 11.
tel: +36 (52) 591-834
<https://www.deutschetelekomitsolutions.hu/>
Imre-Zoltan.Kilian@t-systems.com

Abstract

After decades-long Sleeping Beauty's dream, neural network based artificial intelligence solutions are in explosion. They have broken out from within the walls of research labs, and in many cases, we can meet them as basic pieces of commercial software solutions. Yet, deep learning solutions are in many cases difficult or at least heavy-weighted, when related to conventional, rule-driven solutions. In the present article we are describing two case studies, where the two approaches are successfully combined.

Összefoglaló

Több évtizedes Csipkerózsika-álom után a neurális hálós mesterséges-intelligencia megoldások robbanni kezdtek. A kutató laboratóriumok berkeiből kitörték, és sok esetben piaci szoftver megoldások részeként is találkozhatunk velük. Ám a mélytanulások megoldások egy sor esetben nehézkesek vagy legalábbis túl nehézsúlyúak a hagyományos, szabály alapú megoldáshoz képest. A cikkben két esettanulmány kapcsán a két megközelítés kombinációjára mutatunk be példákat.

Kulcsszavak

Neurális hálók, logikai programozás, csevegőrobotok, képfeldolgozás

1. BEVEZETÉS

Több évtizedes vajúdás után az elmúlt években a neurális hálós mesterséges-intelligencia megoldások robbanni kezdtek, és a kutató laboratóriumok falai közül kikerülve számos esetben piaci alkalmazások fontos építőköveivé nőttek ki magukat. Mélytanuláson alapuló megoldásokat elsősorban nagy adatállományokból különböző következtetéseket levonó rendszerekben, pl. időjárás, tőzsdemozgás előrejelzésében alkalmaznak, de a nagy közönség szeme előtt inkább az önvezető autók fejlődése zajlik, valamint már ma is igen-igen sok számítógépes alkalmazás telefonos ügyfélszolgája maga a számítógép. Ezeket, a Turing-próbát kísérő rendszereket nevezik *csevegőrobotoknak* (chatbot).

A mélytanulások rendszerek azonban a hagyományos, szabályalapú mesterséges-intelligencia rendszereknél gyakran komolyabb számítási erőforrást igényelnek, miközben az efféle rendszerek együttes alkalmazásának a kérdése (vagyis ilyen rendszerek sorba-párhuzamosan kapcsolása) sem megnyugtatóan megoldott.

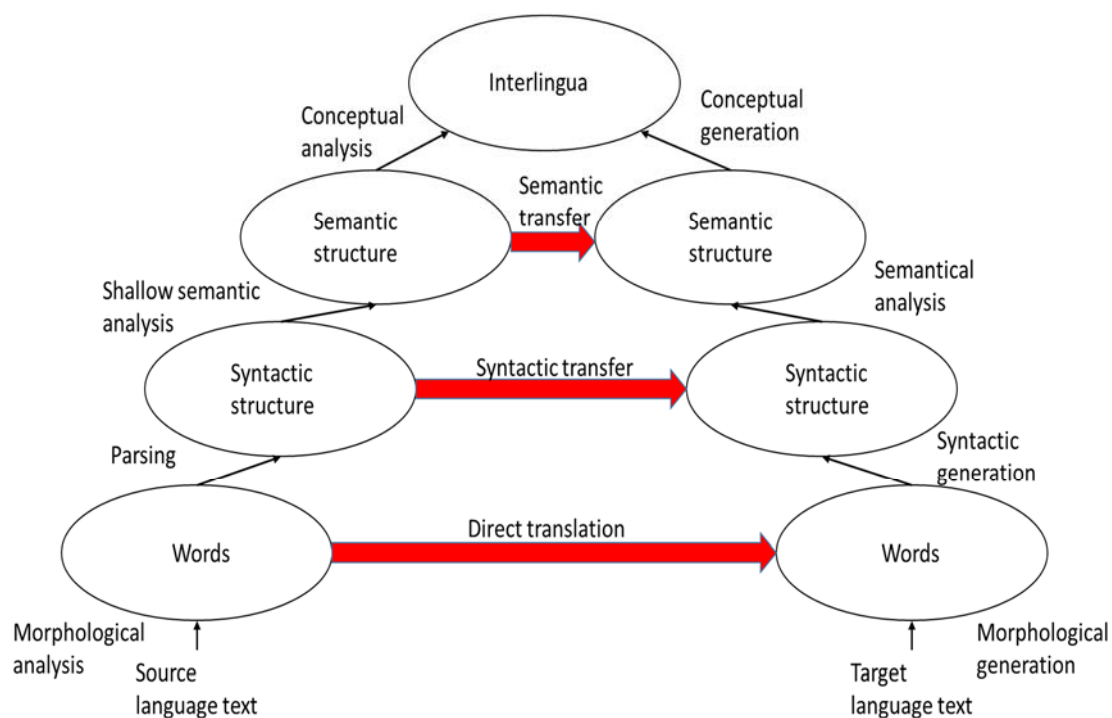
Hibrid rendszerek alkalmazásának az ötlete meglepően kézenfekvő: használjunk mindent arra a célra, amire igazán alkalmas – vagyis a mélytanulások rendszereket inkább bonyolult mintázatfelismerési és -átalakítási feladatokra – pl. beszéd- vagy képfelismerésre – majd a felismert részleteken végezzünk további szimbolikus logikai következtetéseket – hasonlóan a múlt évezred végén sztárolt szakértő rendszer technológiákhoz.

Az alábbiakban két olyan alkalmazási eset elemzése olvasható, amely a témájában meglehetősen eltérő ugyan, de a megoldást mindkét esetben valamiféle hibrid következtető felépítmény kínálta. Mindkettő konkrét alkalmazói szoftverépítési tevékenység eredménye, amely azonban nem mindig lett éles használatba véve, vagyis üzemeltetési tapasztalat a leírtak mögött sajnos nincs. Inkább arról van szó, hogy a szerző a korábbi eredményeit, valamint a Deutsche Telekom multinacionális vállalat berkeiben futó mesterséges intelligencia projekteken szerzett tapasztalatait általánosítva nyert következtetéseit foglalja az írás össze.

1.1. Csevegőrobotok – A Turing próba kísérlete

Klasszikus nyelvi szoftverek készítői előtt jól ismert a gépi fordításra alkalmazható Vauquois háromszög rajza, és persze a mögötte levő elmélet is.

Vauquois természetes nyelvek fordítására használt háromszögű illusztrációját részben csevegőrobotokra is használhatjuk. A szerző által készített ITSy-Bitsy csevegőrobot-modell a Vauquois-féle elveket követve a feldolgozás során betartja a szóalakelemzés-nyelvtani elemzés, majd szemantikus elemzés sorrendet. Az így kapott logika kifejezést egy logika-ontológia leképező szerkezettel lekérdezve megkapjuk az eredményadatot, amit aztán egy fordított irányú szöveg-összeszerelő futószalag szerel össze nyelvtanilag helyes válaszmondattá.



1. ábra. A Vauquois háromszög természetes nyelvek fordítására (és feldolgozására)

A mélytanulásokon elven működő modern csevegőrobot-motorok lényegileg a szövegelemző futószalag szintjeit ugorják át egy rövidzárhoz hasonlóan, és a szövegből egy lépésben állítanak elő valamiféle célorientált logikai nyelvű kifejezést – ennek a neve az *intent-*, vagyis *szándék-kifejezés*. Már a név is elárulja: az egész folyamat szándék-vezérelt, azaz a legfontosabb feladat a konkrét témához kötődő szándék kifejezések nyelvének a megtervezése. Vagyis a technológia igazából témakörfüggő, csak kötött témakör mellett működhet jól, pl. valamilyen elektronikus szolgáltatás üzemeltetése során.

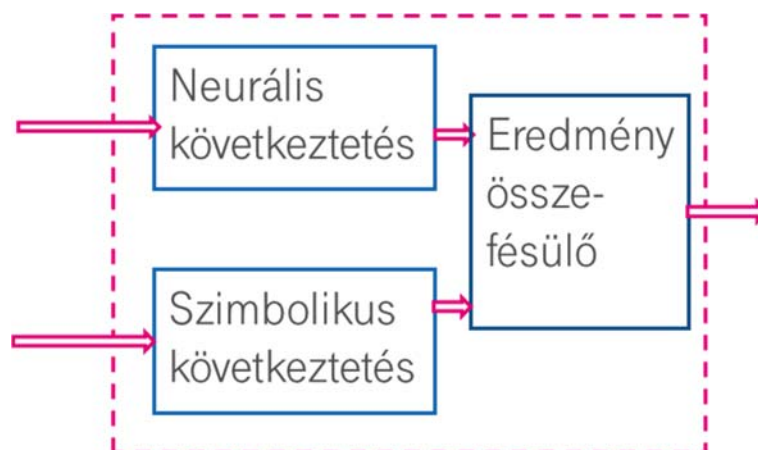
Mindezt a neurális hálós feldolgozási folyamat és a szógyakoriságokra alapuló bemenő mondatmodell (beágyazások/embeddings) jól támogatja annyiban, hogy a be nem tanított mondatokra is képes valamiféle választ előállítani, mert a mondatminták közül hiányzó szavakból valamilyen gyakoriság alapján rokon szó interpolációt (esetleg extrapolációt készít).

Ez a működésmód előrevetíti azt is, ami a neurális hálós megoldás legfontosabb különbsége a szabályalapú megoldáshoz képest. A neurális hálós megoldás – talán extrém esektől eltérően – minden elemzést végigvisz, és ha tudja, meg is válaszolja a kérdéseket – bár kérdések megválaszolásához már a lefordított szándék-kifejezések valamiféle kiértékelése is szükséges. Az elemzés azonban nem feltétlenül 100%-ban megbízható – a neurális hálós algoritmusok eredményként a szándékkifejezés mellett egy megbízhatósági értékkel (confidence) térnek vissza.

A Vauquois lépcsőket lépésenként végigjáró szabályalapú rendszerek ezzel szemben túpontosak – annyira mindenképpen, amennyire pontosan a szabályok be lettek programozva. A be nem programozott nyelvi fordulatokat viszont illegálisnak minősíti – semmi valószínűségi alapú működés, semmi szemantikus hasonlóságra épülő kiértékelés nincs benne, az ilyen esetekre semmilyen megoldás nem ad valamiféle – legfeljebb kisebb valószínűségű eredményt.

2.2. Integrációs lehetőségek. Párhuzamos integráció

A kétféle megközelítés egyfajta hibrid integráció után kiált, amitől azt remélhetjük, hogy mindkét megoldás erősségeit ötvözheti. Erre többféle megközelítés alkalmasnak látszódhat.



2. ábra. Neurális és szimbolikus következtetés párhuzamos integrációja

A *laza csatolásos* (loose coupling) párhuzamos megoldás szerint mindkét rendszernek feltesszük ugyanazt a kérdést. Amennyiben a szabályalapú rendszer képes a kérdést megválaszolni, akkor a választ pontosabbnak tételezhetjük fel. Ha viszont erre nem képes, akkor elindítjuk a neurális hálós rendszert. (Mint az egyetemi vizsgákon: ha a hallgató tudja a kérdésre a pontos választ, akkor bemondja. Ha nem tudná, akkor elkezd mellébeszélni, ami előfordulhat, hogy végül mégiscsak kielégítő lesz a vizsgáztató számára...)

Megjegyezzük, hogy még a tiszta neurális hálós megoldásban is lehetőség van a szándékkifejezésekben üres változók megadására, amelyet a kérdés bizonyos mezői töltenek fel. Ezt leggyakrabban ún. *háttérrendszer integrációra* használják, pl. a szándékkifejezésekből egy SQL utasítást létrehozva, bizonyos adattartalmak kérdezhetőek le. Efféle eszközökkel relációs adatbázisokon túl bármilyen egyéb, programozható felülettel bíró rendszer, nyilvántartások, de akár következtetésre alkalmas, vagy a szemantikus világmodellt (ontológiát) tároló rendszerek is bekapcsolhatók.

A *szoros csatolásos* (tight coupling) párhuzamos megoldás akkor alkalmazható, ha a neurális hálós rendszerben a szándékkifejezéseket lehetséges olyan módon megtervezni, hogy az megegyezzen a szabályalapú rendszer logikai eredménykifejezésével (vagy legalábbis egyértelműen leképezhető legyen). Ez később kiértékelhető szintén valamilyen integrált háttérrendszerrel, vagy – bizonyos esetekben – választhatunk beégetett választ is.

A fenti ábrán jelölt eredmény összehesülő doboz az egyes párhuzamosan futó megoldások eredményének a kiértékelését és a végeredmény előállítását végzi. Bizonyos esetekben – heurisztikus döntések alapján előfordulhat, hogy a bemenő oldalon is létezik egy hasonló funkciójú blokk, amely az egyes következtető megoldások indításáról dönt (pl. szógyakoriságok elemzése alapján esetleg egyik vagy másik megoldást el sem indítja).

2. MOZGÓKÉP FELDOLGOZÁS – INTELLIGENCIÁT A KAMERÁKBA

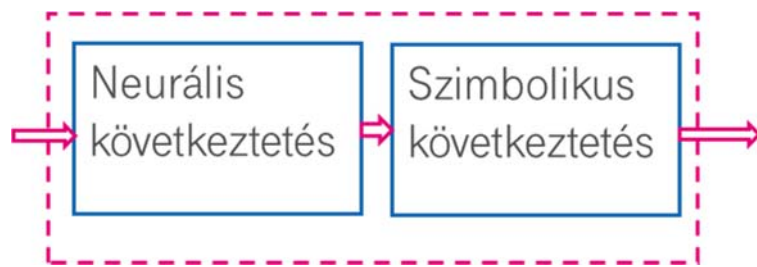
Az alkalmazási eset elsősorban a nyilvános helyeken elhelyezett kamerák intelligenciájának növelését célozza. Nyilvános helyeken – utcákon, tereken, parkolóknak, közlekedési eszközökön, ill. egyes közintézményekben már eddig is elhelyeztek számtalan kamerát, amelyeket emberi erővel értékelték ki. Vagyis, egy diszpécser ült harminc képernyő előtt, és ha bármilyen rendellenes viselkedést tapasztalt, akkor megtette a szükséges intézkedéseket. Az emberi közreműködéssel – mint általában – többféle megbízhatósági gond is lehet:

- az emberi szereplő nem tud 100%-ban hiánytalanul és nem lankadó figyelemmel a képernyő előtt ülni – mosdóba megy, eszik-iszik, netán a családtagjaival vagy más munkatársakkal társalog
- még ha a figyelme teljes is volna, nem elvárható, hogy minden egyes képernyő eseményeit egyforma figyelemmel tudja kísérni

- még a pontosan figyelt képernyőkön is, ezer olyan helyzet lehetséges, amire nem könnyű gyors döntést hoznia – pl. nem várt helyzetek esetében
- ha rendellenes helyzet lépne fel valamelyik kamera helyszínén, akkor a kamerát bemutató képernyőre figyel, vagyis nem tudja a figyelmét több képernyő – több helyzet között megosztani, illetve egyszerre párhuzamosan több rendellenes eseményt nemigen tud kezelni

2.1. Soros integráció

A manapság használatos térfigyelő kamerákat könnyedén ki lehet egészíteni olyan mikroszámítógépekkel (edge computing), amelyek GPU-t is tartalmaznak, így ezek a mélytanulós képfelismerésre különösen alkalmasak. Ezt követően viszont még egy ilyen számítógépecske is könnyen beállítható hagyományos (szabályvezérelt) következtető műveletek végzésére. Ezt a megoldást a kétféle következtetési megközelítés *soros integrációjának* nevezzük.



5. ábra Neurális és szimbolikus következtetés soros integrációja

A hagyományos következtetési fázis alkalmazása többféle okból is célszerű. Egyrészt egy szabályvezérelt következtetési fázis lényegesen olcsóbb, kisebb erőforrással is működhet, és az összekapcsolódási lehetőségei is jobban bejártottak. Másrészt a neurális hálós megoldások sok esetben alkalmatlanok a megfelelő következtetések végzésére – ha valamit valami létező bonyolult előfordulás, mintázat stb. felismerésére fejlesztenek ki, akkor nem kézenfekvő pl. a felismerendő mintázat hiányát felismerni... Harmadrészt, bár léteznek *időfüggő neurális következtetési megoldások*, ezek alkalmazása egyszerűbb esetekben a feladathoz képest túlságosan nehézsúlyú – lényegileg eggyel (az idődimenzióval) megnöveli az alkalmazott neurális technológia dimenziószámát is. Nem túl bonyolult/diszkrét időbeli viselkedések felismerését a legcélszerűbb hibrid következtetési technológiával végezni.

2.2. Használati esetek

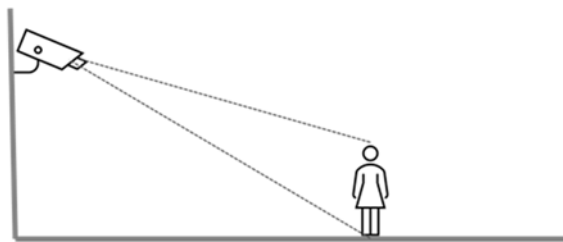
Kamerák intelligenciabővítésének megoldásához érdemes számba venni azokat az előforduló eseteket, amelyek kezelése nyilvános helyeken kívánatos, sőt szükséges lehet, de amelyek egyben a mai eszközökkel meg is valósíthatók.

- Tiltott/védett területekre történő illetéktelen behatolás, legyen szó akár személyekről, akár tárgyakról, csomagokról vagy netán állatokról. Ilyenek lehetnek pl. tűzvédelmi utak, folyosók, ajtók, vágányok és biztonsági sávok, netán autópályán mozgó gyalogosok, kerékpárosok, állatok stb.
- Tiltott, netán veszélyes vagy túlméretes tárgyak, állatok azonosítása (lovak, bikák, fegyverek, túlméretes építési anyagok, csövek, deszkák)
- Felügyelet nélkül hagyott tárgyak, csomagok, gyerekek, állatok azonosítása
- Emberek/állatok azonosítása, akik valamilyen meghatározott állapotban, pl. nem megfelelő ruházatban vagy testhelyzetben vannak, vagy nincsenek szükséges védőeszközzel ellátva (álló-ülő megkülönböztetés, pl. egyházi intézményekben vállat-combot szabadon hagyó ruházat, különböző védőeszközök, maszk-, és sisakviselés, szájkosár, stb. ellenőrzése)
- Rendellenes viselkedés azonosítása (túl gyors futás, dohányzás, evés-ivás, verekedés, lefekvés – pl. rosszullet esetén)

2.3. Descartes és a perspektíva

A fentiekben említett esetek egy nagy részében szükséges a felismert objektumok valamiféle *lokálizációja* – világunkban sokmindent a kétdimenziós vetületével – vagyis egy térképpel – írunk le... Legtöbbször a legegyszerűbb megoldás a felismert objektum ráhelyezése volna a terület efféle alaprajzára vagy

térképére, ahol pl. a tiltott zónák könnyen megjelölhetők. Fontos felismerés: világunk nem három, hanem *csak kétdimenziós*... Az esetek nagy részében valamilyen síkhoz kötöten élünk (pl. mindenkinek földig ér a lába), és így a kamera perspektivikus nézete – valamilyen kalibráció után – néhány trigonometrikus összefüggés, vagy mátrixművelet árán levetíthető erre a kitüntetett síkfelületre is.



6. ábra 3D lokalizálás egyetlen kamerával

A megoldás többféle szempontból is kérdéses, bár bőven vannak korrekciós lehetőségek is:

1. A geometria megköveteli a kamera pontos magasság-távolság, látószög, és egyéb méretadatainak használatát. Ha ezek nem állnának rendelkezésre, akkor a rendszert valamiféle etalon mértékek (mérőrúd, szögek) leképezésével *kalibrálni* kell. A kalibráció mindenképpen emberi beavatkozást igényel, és a környezet változásával (pl. ha a kamerát egy másik pozícióba helyezik), automatikus módon nem oldható meg.
2. A fenti geometriai elrendezésből kikövetkeztethető számítás az egyszerűbb esetben fél tucat szorzásból és kicsit kevesebb trigonometrikus függvényből hozza ki egyetlen pontra vonatkozólag az eredményt, ami viszont csak *kicsi látószögek esetén pontos*. Nagyobb látószögű kamerák (pl. halszem-optika) esetében vagy keresni kell egy pontosabb (és bonyolultabb) leképezési eljárást, vagy vállalni a pontatlanságokat.
3. Talán a legnehezebben elhárítható gond a *részleges kitakarások* (partial occlusion) problémája.

2.4. Takart objektumok

Talán épp a kitakarások problémája a legnehezebben megoldható. A kitakarások miatt ugyanis – különösen felső kameraállások mellett - egyes személyek alsó fertálya nem látható, vagyis a felismerés által létrehozott dobozkeretek csak térdtől, deréktól, mellkastól vagy netán csak nyaktól ábrázolják az illetőt, márpedig a síkhoz kötöttség éppen a kitakart részre teljesülhet. Egy ily módon csak részben felismert személyt csupán a dobozkeret alapján viszont nem lehet megkülönböztetni egy távolabb állótól, vagyis a kitakarás mértéke szerint sugárirányban akár komoly elhelyezési bizonytalanságot is okozhat a dolog (a kisebb felismert magasságot távolabbra képezi le az algoritmus).

A gondok többféle korrekcióval is orvosolhatók:

1. Talán a legtisztább korrekció az, amikor az objektumdetektáló – legalábbis személyek esetében – befoglaló dobozként a kitakart résszel együtt számított teljes testmagasságot előre jelzi, vagy esetleg valami magasságarányt is visszaad... természetesen ez némi plusz adatszámítás és betanítási munkát is igényel. A tényleges magasság ezután korrigálható a valószínűsíthető teljes magasságra.



7. ábra A kitakart résszel együttes testmagasság előrejelzése [1]

2. Ennél olcsóbb megoldás annak feltételezése, hogy minden személy legalább egyetlen képkockán teljes magasságában látható, és ez a képkocka a személy megjelenése után hamarosan sorra is kerül (pl. mindenki belép valamilyen ajtón, amikor kitakarás nélkül a kamera látóterében van). Ilyenkor tárolhatjuk az eddig felismert legnagyobb magasság/szélesség arányt, és a felismert arányokat ezzel az értékkel korrigálhatjuk. (A dolog normális viselkedés mellett működhet jól, addig, amíg pl. az ajtón belépők nem jönnek széttárt karral, meghajolva, vagy négykézláb...)

2.5. Valós idejű és időfüggő viselkedés

A vázolt felépítményt megvalósító szoftverrel szemben több kihívás is felmerül.

Talán a legfontosabb, hogy a szoftver *valós idejű* (real-time). Vagyis az említett képfeldolgozási feladatokat minden egyes képre elvégezzük, ezért az alkalmazott képváltási frekvencia mellett azt a periódus időszakaszában mindenképpen be kell fejezni. Könnyítés, hogy *nem szigorúan valós idejű* – vagyis, ha egy-egy esetben a feldolgozás nem tudna időre befejeződni, az csak a termék minőségét rontja, de nem okoz egyéb kárt emberben és vagyonban mérve sem.

Valós idejű és időfüggő viselkedésű szoftvereket nem könnyű tesztelni, hiszen az ilyen szoftverek a beérkező eseményekre vonatkozólag *nem állapotmentesek*. Vagyis nem elég egy-egy eseményfajtát a megelőzőktől függetlenül tesztelni, mert azok felépítenek egy *állapotkörnyezetet*, és a tesztelendő esemény hatása csak az adott környezetben vizsgálható. A lehetséges eseménysorozatok száma viszont – ha megszámlálhatóan is – végtelen. A probléma a következő megoldásokkal hidalható át:

- A végtelen számosságú eseménysorokból – emberi megfontolások alapján – kiválasztunk véges számú, tipikus lefutási mintát, kiiktatva egyes periodikusan ismétlődő vagy, érdektelen rész-eseménysorokat. Ügyes teszt-tervezéssel véges – és nem kezelhetetlenül nagy – számú esetet tesztelhetünk.
- A tesztesetek leírásánál a bemenő paraméterek és az eredmények között tekintjük magukat *az állapotleíró adatszerkezeteket* is. Ha ezek számossága-bonyolultsága nem túl magas, akkor valószínűleg ismét csak véges és kezelhető számú tesztesetet kaphatunk.
- Valós idejű szoftverek rendszerint nagyobb időtávban futnak, és idővel paraméterezett eseményeket – ill. következtetéseket tárolnak. Ilyen szoftverek tesztelésekor célszerű meghatározni az időfüggő kezelésmód *léptékét* – vagyis azt a leghosszabb időtartamot, amin belül még jelentőséggel bíró eseménysorok fordulhatnak elő, és ezek detektálása még érdekes lehet. Vagyis a léptékválasztás megad egy olyan időbeli felső korlátot, amin túl már nem szükséges az eseménysorokat tesztelnünk. Ügyes választással megintcsak komolyan, de legalábbis kezelhető méretre csökkenthetjük a vizsgálandó tesztesetek számát.

Mindenesetre a fenti általános megfontolások ellenére egy fontos és mély elméleti kérdésen mégis érdemes elgondolkodnunk... Minden lehetséges eseménysorozat tesztelése elvileg is lehetetlen, hiszen a végtelen

hosszúak is lehetnek. Sajnos pusztán elméleti alapon ennél több is igaz lehet: minden kezelési módhoz található olyan (esetleg mesterséges, sőt, mesterkélt) eseménysorozat, amivel meg lehet bolondítani.

2.6. Rácsos elrendezések foglaltsági problémája és a valószínűségi következtetések

Nyilvános kamerák egyik fajtáját a mennyezeten, de legalábbis magasan helyezik el, lehet halszem optikával vagy akár nélküle. Az ilyen kamerákat gyakran olyan helyeken alkalmazzák, ahol valamiféle rácsos elrendezésben emberek (pl. székeken), állatok, járművek (pl. parkolóokban) láthatók, és a megoldandó feladat lehet a rácsos elrendezés foglaltságának, netán rendellenes vagy felügyelet nélküli tárgyakkal a felismerése és kijelzése a rács-cellákhoz kötötten.



8. ábra Felismerés magasan elhelyezett kamerával és ráccsal

A rácsos elrendezésre a legalább részlegesen látható cellákat elhatároló rácsszerkezetet érdemes ráhúzni. Ezáltal egy felismert objektum valamelyik cellára illeszthető. Az illesztés azonban teljesen bizonyosan megköveteli egy valószínűségi következtetési rendszer használatát.

Előfordulhat ugyanis, hogy a felismert objektum kilóg a cellájából. Lehet, hogy csak a dobozkeretének a sarkai lógnak ki. Lehet, hogy a perspektíva miatt a testének egy része is kilóg, de az is lehet, hogy egy nagyobb testű ember vagy objektum ténylegesen több cellát foglal el.

A megoldás fontos újdonsága a *valószínűségi következtetési rendszerek* alkalmazása.

A neurális hálós képfelismerés maga is egy bizonyos valószínűségi (confidence) értékkel tér vissza, amelyet mindeddig - a nagyon gyenge értékkel bíró felismeréseket eldobva - legfeljebb *értékhátárolásra* használtak. Rácsalapú következtetések esetében még a további feltételezések tehetők:

- amennyiben egy felismert objektum egy cellába nem teljesen fér be (vagyis kilóg), akkor a valószínűségi értékét a cellába eső közös rész arányában számítjuk
- amennyiben egy felismert objektum több cellába benyúlik, akkor a teljes valószínűségi érték az egyes cellalefedések összege
- valamely reláció valószínűségének kiszámítása a felismert objektumok között azok rácsbeli pozíciója alapján történik. Például a csomag-utas hozzátartozási reláció esetében a legvalószínűbb, hogy az utas a csomagját az ölében vagy a lábánál helyezi el. Kicsit kevésbé valószínű, hogy a szomszédos ülésre teszi. Alig hihető azonban, sem az, hogy a csomagot két üléssel arrébb, sem az, hogy netán valamelyik másik sorba, vagy (pl. vonaton) az ülések közti folyosó túloldalára tenné.
- Egy reláció valószínűségének kiszámítása alapján a negált reláció valószínűsége a biztos eseményre vonatkozó komplementerként értelmezhető és számítható.

A fenti megfontolások alapján az elemi valószínűségi számítási szabályokkal könnyen kiszámítható pl. annak valószínűsége, hogy egy konkrét csomag vagy egyéb felismert tárgy tekinthető-e magára hagyottnak. Pusztán kezelői felület tervezési kérdés ezután, hogy ilyen tárgyak esetében milyen módon értesítjük a feltételezett tulajdonost, vagy az utazóközönséget.

3. EREDMÉNYEK, ÖSSZEGZÉS

Az ITSy-Bitsy csevegőrobot működéséről korábbi konferenciákon [5] már beszámoltunk. A rendszer a Micimackó tudásbázisán, az angol nyelvi elemző (parser) bevonásával lett tesztelve, talán 1-2 tucatnyi jellegzetes (és egyszerűbb) tesztkérdés segítségével. A rendszerhez tesztelés alatt áll egy német elemző is, amely még – a szerző egyéb elfoglaltságai miatt – nem fejeződött be. A rendszer neurális hálós hibrid integrációjára vonatkozó elképzelések viszont mindezülig csupán elméleti jellegűek, ilyen gyakorlati megvalósítás egyelőre nem kezdődött.

Hibrid következtetési megoldások viszont használatosak a Deutsche Telekom egyes képfeldolgozó projektjeiben. Ezek jellegzetessége, hogy a neurális hálós felismerési fázis után klasszikus, szabály alapú következtetési fázis hajtódik végre.

Ilyen megoldás született pl. az utasszámlálás problémájára, amelyben a szabályalapú következtetési lépések SWI-Prolog nyelven [2] íródtak a rendszerhez csatolt Contralog [3] [4] előrekövetkeztető kiterjesztéssel, de a megoldás az integrációs nehézségek miatt végül éles alkalmazásba nem került. Egy másik esetben tömegközlekedési környezetben a magukra hagyott csomagok felismerése volt a cél. Ez a megoldás a neurális hálós felismerést amúgy is vezérlő Python nyelven készült, így integrációs gond nem merült fel. Az elkészült modell sikeresen alkalmazza a rács alapú valószínűségi következtetési stratégiáról elmondottakat. A rendszer elsősorban a mennyezetre szerelt kamerával alkalmazható a legeredményesebben, a tömegközlekedésen túl más, rácsos elrendezésű környezetben is, pl. járművekben, előadótermekben, parkolóházakban, stb.

HIVATKOZÁSOK

- [1] https://docs.openvino.ai/latest/omz_models_model_person_detection_action_recognition_0006.html (elérés: 2022-07-10)
- [2] J. Wilemaker: An overview of the SWI-Prolog programming environment, Proc. 13-th International Workshop on Logic Programming Environments, pp.1-16. ed: F. Mesnard, A. Serebenik, Katholieke Universiteit, Leuven, Belgium, 2003.
- [3] Kilián, I.: Contralog: egy előre haladó, Prolog-konform következtető motor és alkalmazása ReALIS nyelvi elemzésre
In: Bíró Károly, Sebestyén-Pál György (szerk.)
ENELKO 2011 SZÁMOKT2011: XII Nemzetközi Energetika-Elektrotechnika Konferencia; XXI Nemzetközi Számítástechnika és Oktatás Konferencia.
Kolozsvár: Erdélyi Magyar Műszaki Tudományos Társaság (EMT) 2011., pp. 199-205.
- [4] Kilián, I.: Contralog: a Prolog-conform forward chaining environment and its application for dynamic programming and natural language parsing
Acta Univ. Sapientiae, 8. 1. (2016) pp. 41-62
- [5] Kilián, I.: Általános célú természetes nyelvi elemzőprogramok létrehozásának tapasztalatai
In: Bíró Károly, Sebestyén-Pál György (szerk.)
ENELKO 2011 SZÁMOKT2021: XXII Nemzetközi Energetika-Elektrotechnika Konferencia; XXXI Nemzetközi Számítástechnika és Oktatás Konferencia.
Kolozsvár: Erdélyi Magyar Műszaki Tudományos Társaság (EMT) 2011., pp. 199-205.