

Mesterséges Intelligencia

Módszertanok, technológiák, eszközök

A modul célja a mesterséges intelligencia területén alkalmazott régi és modern fejlesztési módszertanok, életciklusok rövid bemutatása.

Cél

A lecke célja, hogy a hallgató megismerje és megértse a mesterséges intelligencia területén alkalmazott fejlesztési módszertanok alapvető fogalmait és folyamatait. Az alapfogalmak a tananyag későbbi részének elsajátítása során segítik azok gyakorlati alkalmazásával járó feladatok részletesebb megértését.

Követelmények

Ön akkor sajátította el megfelelően a tananyagot, ha képes

- felsorolni a mesterséges intelligencia területén alkalmazott főbb módszertani irányokat,
- felsorolni a Waterman-féle tudástechnológia lépéseit,
- megnevezni a modellezési folyamat alapú módszertanok főbb jellemzőit,
- saját szavaival bemutatni az adatvezérelt módszertanok jellemzőit, valamint a gépi tanulás életciklusát,

Időszükséglet

A tananyag elsajátításához körül-belül 60 percre, további kiegészítő tevékenységekre hozzávetőlegesen 60 perc lesz szüksége.

Kulcsfogalmak:

- tudástechnológia,
- átadási folyamat megközelítés (transfer process),
- modellezési folyamat megközelítés (modeling process),
- adatvezérelt megközelítés (data-driven),
- digitális ko-kreáció (digital co-creation).

1. Fejlesztési módszertanok és szemléletek

1.1. Korai mesterséges intelligencia fejlesztések és a tudásalapú rendszerek fejlesztésének átadási folyamatként való megközelítése

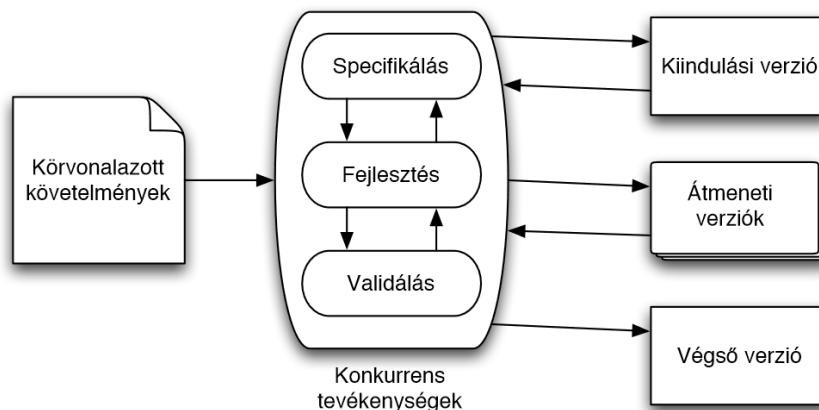
Tevékenység: Jegyezze meg a tudás alapú rendszerek fejlesztésének, valamint az átadási folyamat szemléletmódú fejlesztés főbb jellemzőit!

A korai MI kutatások főként a formalizmusokra, következtetési mechanizmusokra és a tudásalapú rendszerek működtetésére szolgáló eszközök fejlesztéséről szólt, kisebb méretű tudásalapú rendszerek megvalósítások jellemezték, melyek során a különböző megközelítések megvalósíthatóságát is vizsgálták. Az ígéretes eredmények ellenére a legtöbb esetben megbukott a (kereskedelmi) nagyobb méretű tudásalapú rendszerek fejlesztésének többsége.

A késői 1960-as években hasonló folyamat volt megfigyelhető a tradicionális szoftverrendszerek fejlesztésénél is, az úgynevezett szoftver krízis, amikor a kis akadémiai prototípusokat nem tudták skálázni nagyméretű, karbantartható kereskedelmi rendszerek szintjére. E problémák megoldása vezérelte a szoftverfejlesztés/szoftvertechnológia (software engineering) területének megalapozását. A mesterséges intelligencia területén ezt a módszertani háttér tudásfejlesztésnek, vagy tudástechnológiának (knowledge engineering) nevezik. E terület célja a tudásalapú rendszerek fejlesztésének mérnöki területté formálása, ami magába foglalja a fejlesztéshez szükséges eszközök, nyelvek és módszertanok fejlesztését, valamint az építési és karbantartási folyamatok vizsgálatát is.

A korai és a modern mesterséges intelligenciát alkalmazó rendszereknél egyaránt megfigyelhető a klasszikus evolúciós és spirál modell jellegű szoftverfejlesztési életciklus (a modern gyakorlatban az agilis eszközök adta lehetőségeket kiaknázva).

Az evolúciós fejlesztési módszer egy felfedező típusú fejlesztési folyamat, melynek során a kezdeti specifikáció jól megértett. A követelmények jobb megértése újabb funkciókkal ellátott „eldobható” prototípusok fejlesztése révén történik. A jellegéből adódóan erősen iteratív folyamat, melyet jellemez a bizonytalanság, amit a fejlesztés minél korábbi szakaszában próbál kiküszöbölni. Nem jól látható folyamatok, rosszul strukturált rendszerek, speciális eszköz igény jellemzi. Interaktív rendszerek fejlesztésénél, vagy nagy rendszerek részrendszereinek fejlesztésénél alkalmazzák, jellemzőek még a rövid életciklusú szoftverek.

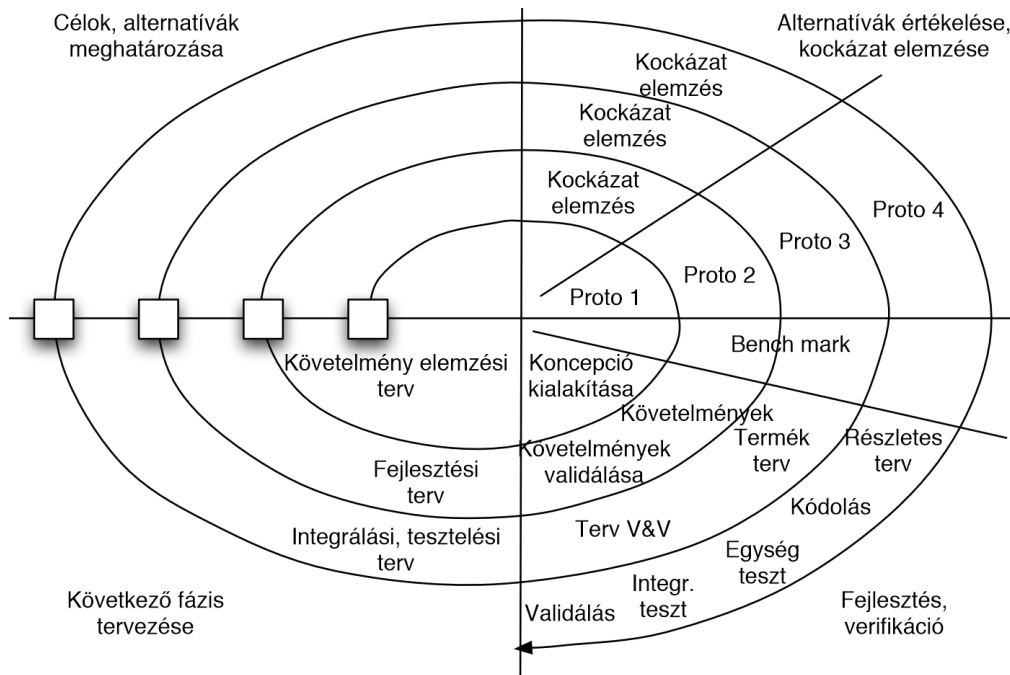


1. ábra. Az evolúciós fejlesztést szemléltető ábra
(Forrás: Heckenast Tamás Szoftver-technológia tananyag)

Boehm nevéhez fűződik az 1988-as spirál fejlesztési modell, melynek neve is tükröz egy erősen iteratív folyamatot. Tevékenység régiók definiálhatók, fontos a fejlesztő - megrendelő

kommunikációja, a feladatok közé tartozik a tervezés, a kockázat elemzés, a prototípusok létrehozása, a konstrukció és a megrendelői értékelés.

Nincsenek rögzített fázisok, tevékenység szekvenciák, lényegében a spirál egy köre felel meg egy-egy fázisnak.



2. ábra. A spirál fejlesztési modellt szemléltető ábra
(Forrás: Heckenast Tamás Szoftver-technológia tananyag)

Az első komolyabb módszertanok a tudásalapú rendszerek fejlesztésére irányultak, az 1980-as években jelentek meg. A legelterjedtebb megközelítés az „átadási folyamatként” (transfer process) tekintett a fejlesztésre. Az elnevezés arra utal, hogy ezek a módszertanok abból az alapvetésből indultak ki, hogy egy-egy szakterületre vonatkozó tudás már létezik, csak össze kell gyűjteni a területen dolgozó szakértő (domain expert) tudását, egyéb a területre vonatkozó ismeranyagokat egy a tudásalapú rendszerek fejlesztésén dolgozó tudásmérnök (knowledge engineer) segítségével, mely így formalizálható. Ezt jellemzően a szakirodalom feldolgozása, valamint szakértői interjúk formájában történik, a kinyert tudást pedig a legtöbb esetben valamilyen szabályok formájában tárolták (vagyis így modellezték a problémát). Az ilyen rendszerekkel az a probléma adódott, hogy egyik megoldás sem volt képes megfelelő módon leírni a tudás minden formáját, vagy ha mégis, az annyira bonyolult volt, hogy valós körülmények között jelentősen megnehezítette a karbantartást, vagyis a tárolt tudás frissítését.

Felismerték, hogy a probléma gyökere abban rejlik, hogy nem csak a tényszerűen megfogalmazható ismeretek és (explicit) tudás számít, hanem az úgynevezett tacit (rejtett, hallgatólagos) tudás is, amire a legjobb példa az olyan tudatalatti képesség, amikor egy évtizedek óta egyetlen munkafázist ellátó szakember nem tudatosan felismer különböző helyzeteket és annak megfelelően cselekszik. Jellemzően az ilyen jellegű tudást nem, vagy csak nagyon nehezen tudják megfogalmazni, ezek begyűjtése a legnehezebb. A tudásnak egy harmadik típusa is létezik, a beágyazott tudás, ami jellemzően különböző folyamatokban, valamint tárgyak, munkaeszközök kialakításában jelennek meg, de ezek kevésbé fontosak a szakértő rendszerek szemszögéből.

A transfer process típusú, tudásalapú rendszerek fejlesztésére alkalmazható modellre a legjobb példa az 1986-ban Waterman által publikált tudástechnológia folyamata, melyben a

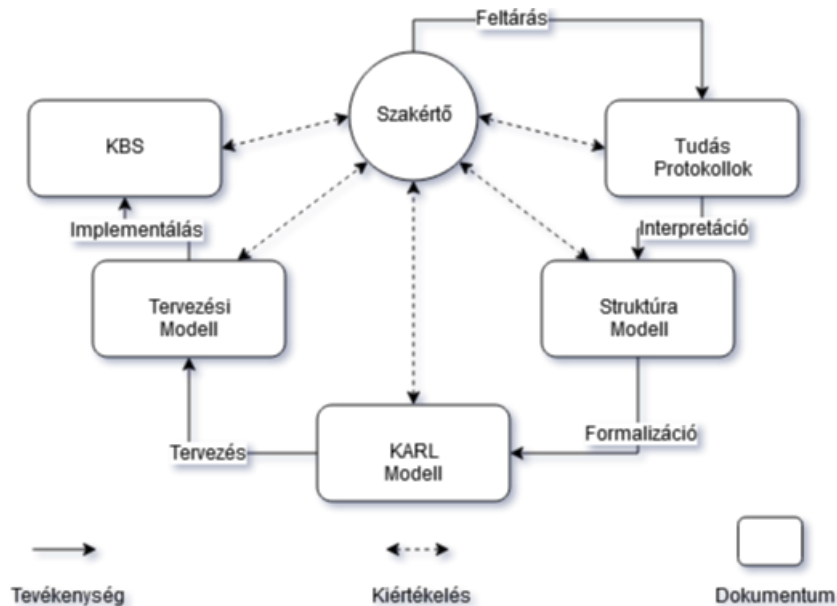
fejlesztést az alábbi hat fázisra osztotta, melyek mindegyikéből lehetőség van az előző fázisokba való visszalépésre:

- Probléma felmérése,
- Adatok és tudás megszerzése,
- Prototípus rendszer kifejlesztése,
- Teljes rendszer kifejlesztése,
- Rendszer kiértékelése, átdolgozása,
- Rendszer integrálása, karbantartása.

1.2. Tudásalapú rendszerek fejlesztésének modellezési folyamatként való megközelítése

Tevékenység: Jegyezze meg a modell vezérelt fejlesztési megközelítés lényegét!

A fejlesztést modellezési folyamatként (modeling process) tekintő módszertanok során nem a szakértő tudásának „lemásolása”, „átvitele” a cél, hanem probléma a szakértővel felérő pontosságú modelljének megalkotása, mely eltérhet a szakértői tudás valós modelljétől. A szemlélet előnye, hogy a rejtett, nehezen megszerezhető és leírható szakértői tudás megfogalmazására nincs szükség a módszer jellegéből adódóan. A megalkotott modell lényegében a valóság egy vetülete, maga a modellezési folyamat ciklikus jellegű, a modellezendő rendszer/folyamat mély ismerete szükséges, költséges és időigényes. E megközelítést követi a Model-based and Incremental Knowledge Engineering (MIKE) módszertan, melyet az alábbi ábra szemléltet.



3. ábra. A MIKE fejlesztési folyamat lépéseit szemléltető ábra

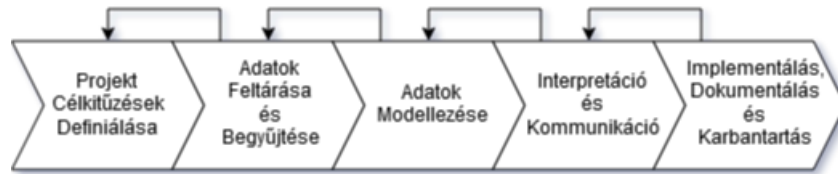
1.3. Tudásalapú rendszerek fejlesztésének adatvezérelt megközelítése

Tevékenység: Jegyezze meg az adatvezérelt módszertani megközelítés lényegét!

A napjainkban korszerűnek számító mesterséges intelligencia megoldások, mint például az ajánlórendszerek, egyéb profilalkotó rendszerek, gépi látás megoldások, valamint nyelvfeldolgozó rendszerek területén egyaránt jellemző a hatalmas adatigény. Ezek fejlesztésére használják az adatvezérelt (data-driven) megközelítésen alapuló módszereket. A modern technológiák jellemzően a gépi tanulás révén, a mintaadatokra alapozva, automatikusan építik fel a modelleket. Az ilyen rendszerek számos módon megvalósíthatóak,

de az utóbbi időben a legjellemzőbbek a mesterséges neurális háló alapú megoldások. Jelentős előnye, hogy egyáltalán nincs szükség a szakértői tudás megfogalmazására, kizárólag megfelelő minőségű és mennyiségű adatra.

A gépi tanulás alapú fejlesztések életciklusát az alábbi, DataRobot Inc. által készített ábra szemlélteti:



4. ábra. A gépi tanulási életciklust szemléltető ábra

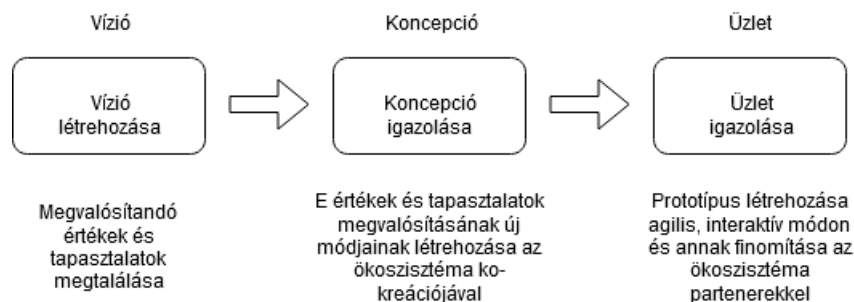
A kutatók dolgoznak azon, hogy miképpen lehetne az efféle megoldások adatigényét csökkenteni (például az adatok dúsításával, mesterséges minták előállításával), a jelölésüket és feldolgozásukat automatizálni.

További említésre méltó (nem adatvezérelt megközelítésű) módszertanok: Problem Solving Methods (Clancey), mint például Cover-and-Differentiate, a Propose-and-Revise, a SALT (PSM: Propose-and-Revise); a Role-Limiting Methods (RLM), mint például a Configurable Role-Limiting Methods; a Knowledge Acquisition and Documentation Structuring (KADS), melyek közül talán a CommonKADS a legelterjedtebb; valamint a PROTÉGÉ-II.

1.4. Digitális ko-kreáció,

Tevékenység: Jegyezze meg a digitális ko-kreáció lényegét!

A digitális ko-kreáció (digital co-creation) egy a Fujitsu által is követett fejlesztési megközelítés, melynek lényege, hogy különböző területek képviselőit fogja össze olyan új megoldások és folyamatok megoldása érdekében, melyek korszerű megoldásokra épülnek, mint például a mesterséges intelligencia, vagy az IoT (Internet-of-Things, dolgok internete). A folyamatot az alábbi ábra szemlélteti:



5. ábra. A digitális ko-kreáció folyamatát szemléltető ábra