KONTROLLÁLT TERMÉSZETES NYELVŰ LEKÉRDEZÉS WEBES ADATBÁZISOKHOZ

Mészáros Tamás, meszaros@mit.bme.hu*
Dobrowiecki Tadeusz, dobrowiecki@mit.bme.hu*
Kiss Margit, kiss.margit@webit.hu**
**BME Méréstechnika és Információs Rendszerek Tanszék*
*** ELTE BTK Nyelvtudományi Doktori Iskola*

Bevezetés

Az elektronikusan tárolt információk hatékony és egyszerű elérése egy komplex informatikai feladat, amely a web rendszereiben tárolt és elérhető információk dinamikus bővülésével a felhasználók egyre szélesebb köre számára mindennapi tevékenységgé vált. Napjainkban legszélesebb körben alapvetően statisztikai és katalogizáló módszereket alkalmaznak e feladat megoldására, melyek azonban a tárolt információ mélyebb megértése nélkül nem képesek a feladat kielégítően jó megoldására. Ezért egyre nagyobb hangsúly kerül mind a tárolás mélyebb szemantikai szintjeinek kidolgozására (W3C Szemantikus Web), mind az elérési módszerek közelítésére az emberi szint felé (új lekérdező nyelvek, természetes nyelvfeldolgozás).

A BME Méréstechnika és Információs Rendszerek tanszékén az elmúlt években több hazai és nemzetközi projektben is vizsgáltuk az információ-elérés sajátosságait, megoldási javaslatokat tettünk ezen rendszerek minőségének javítására. Az elmúlt évben a természetes nyelvek gépi elemzésének olyan alkalmazásait vizsgáltuk, melyek lehetővé tehetik az átlagos felhasználó számára egyszerűbben használható, természetes nyelvű lekérdező felületek megvalósítását. Az általunk javasolt megoldás korlátozza a felhasználó által használható természetes nyelvet (egy speciális szövegbeviteli módszer alkalmazásával), és így lehetővé teszi a felhasználói lekérdezések egyértelmű elemzését és értelmezését.

Az előadás keretében egy olyan módszert mutatunk be, amely lehetőséget teremt webes adatbázisok és dokumentumtárak természetes nyelvű lekérdezésére. Kitérünk a korlátozott nyelv tulajdonságaira, elemzésének (és generálásának) módszerére, a speciális szövegbeviteli módszer működésére és megvalósítására, valamint az elemzett természetes nyelvű kérdés adatbázis-lekérdezéssé történő fordítására. A bemutatott eljárás mellett ismertetjük a megvalósított kísérleti rendszert, amely egy nyelvészeti adatbázis magyar nyelvű lekérdezésére szolgál.

A kísérleti rendszer alkalmazási környezete

A kísérleti rendszer egy magyar főnévi vonzattárhoz valósít meg természetes nyelvű lekérdező felületet. A főnévi vonzattár egy olyan adatbázis, amely körülbelül 30 000 magyar főnév vonzatosságáról tartalmaz adatokat [1]. Az egyes főneveknél mintegy harmincféle vonzat adatát, valamint több vonzat esetén ezek kapcsolódásait, összefüggéseit is bemutatja a rendszerezés.

Főnevek szerint kereshetünk a vonzattárban akár egy adott szót, akár egy vagy több betűkombinációt tartalmazó szavakat. Vonzatok szerint szintén többféle módon lehet keresni attól függően, hogy több vonzat megléte esetén milyen kombinációs feltételeket határozunk meg. Kereshetünk csak fakultatív vonzatokat (f), illetve azon belül élőket (fé) és nem élőket (fn). A kötelezőknél hasonló módon járhatunk el (k, ké, kn). Ha egy főnév több vonzattal is rendelkezik, abban az esetben a vonzatok

közötti viszonyok is jelölve vannak a vonzattárban. A négyféle jelölt kapcsolódási lehetőség a következő: két vagy több vonzat lehet szinonim, vagylagos, lehet köztük és-kapcsolat, illetve kizáró vagy kapcsolat. A több szempontú keresés azt a célt szolgálja, hogy ilyen nagy adatmennyiségből is könnyen emberi elemzésre alkalmas csoportokat kaphassunk.

Az adatbázisban egy webes rendszer segítségével kereshetünk (a publikus rendszer elérhető a http://www.webit.hu/fonevlista/ webcímen). A webes felületen űrlapok segítségével fogalmazhatók meg a főnévi és a vonzatfeltételek. A tapasztalatok szerint a lekérdezések elkészítése gyakran okoz problémákat a felhasználóknak, ezért célul tűztük ki egy egyszerűbb, természetes nyelvű lekérdező felület elkészítését, amely képes ehhez hasonló kérdések megválaszolására: „*melyik főnév rendelkezik fakultatív, előre vonatkozó* szemben *vonzattal*” (lásd 1. ábra).


Főnévi vonzattár

Felhasználó: Mészáros Tamás

Információk

Keresés főnevek és vonzatok szerint

Navigáció: Főnévi vonzattár - kérdés a főnévlistához - válasz a kérdésre

Melyik főnév rendelkezik fakultatív előre vonatkozó szemben vonzattal?

A válasz:

Keresési minta: szemben ilike '%fé%'

Akeresés tárolása | melyik főnév rendelkezik fakultatív előre vonatkozó szemben vonzattal

89 főnevet találtam. Csak a kijelölt (illetve a legfontosabb) oszlopokat mutatom.
Az eredmények letöltése (az összes oszloppal): angol excel táblázat, magyar excel táblázat, XML.

Főnév	gen	ban	ről	hez	ből	nan	va	kor	vel	től	ért	nek	re	vmilyen	vmennyi	vmin	után	között	ellen	szemben	m
agresszió																			fé	fé	
ágyúharc									fé		fn							fé	fé	fé	
aknaharc									fé		fn							fé	fé	fé	
állóharc	é								fé		fn							fé	fé	fé	
ármány																			fé	fé	
atrocitás																				fé	
bandaharc									fé		fn							fé	fé	fé	
harikádharc									fé		fn							fé	fé	fé	

1. ábra: Egy minta lekérdezés futási eredménye a főnévi adatbázisban

Ezen célkitűzés megvalósításához elsőként a természetes nyelvű lekérdezés problémáit kellett megoldanunk.

Lekérdezés természetes nyelven

Egy természetes nyelvű lekérdező rendszer képes az emberi nyelven megfogalmazott kérdések megértésére és megválaszolására valamilyen háttértudás alapján. Az ilyen rendszerek meghatározó komponense egy nyelvi elemző, amely képes a megfogalmazott kérdés szintaktikai és szemantikai elemzésére. A sikeres nyelvi elemzés után a rendszernek képesnek kell lennie a feltett kérdéshez egyértelmű jelentést csatolni, majd elkészíteni annak fordítását egy kiválasztott gépi lekérdező nyelvre (például SQL-re). Egy ilyen rendszer elkészítésének nehézsége a nyelvi elemzés elkészítésének – ma még megoldatlan – problémáiban rejlik. A természetes nyelvek sokszínűsége, időben és személyről-személyre változó természete, valamint a mondatok egyértelmű jelentésének meghatározásához szükséges széles háttértudás teszik igazán nehézzé e feladat sikeres megoldását.

A természetes nyelvek számítógépes elemzésének komoly problémái alternatív megoldások kialakítására is ösztönzik e rendszerek kutatóit és fejlesztőit. Egy ilyen alternatív megoldás az ún. **kontrollált nyelv (controlled language)** nyelv alkalmazása [3], amely a sikeres elemzés érdekében szűkíti a felhasználó által használható természetes nyelvet. Az általunk alkalmazott módszer is ezt a megoldást követi: a lekérdezés során kontrollált magyar nyelvet alkalmazunk. Feladatunk egy

általános kontrollált nyelv használatához képest tovább egyszerűsíthető, hiszen az általunk kitűzött cél csak kérdések kontrollált megfogalmazását teszi szükségessé.

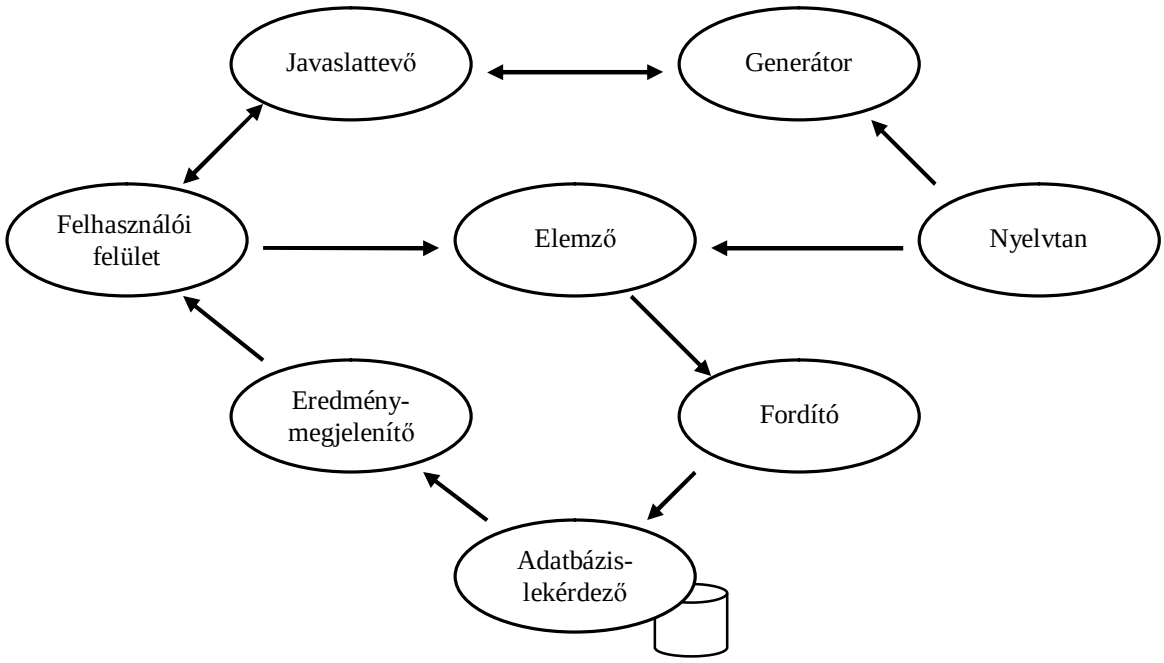
A kontrollált nyelvek alkalmazásának legnagyobb problémája az, hogy megkövetelik a felhasználótól, hogy alkalmazkodjon a korlátozott nyelvtani szabályokhoz (valamint a korlátozott szókincshez). Ezt általában az adott alkalmazást használó emberek előzetes felkészítésével érik el. Az alkalmazás (például egy termék dokumentációjának elkészítésére alkalmas szövegszerkesztő) használatának betanítása során a rendszer kijelzi a hibás mondatokat, és így a felhasználó fokozatosan elsajátítja a kontrollált nyelvet. Az általunk kitűzött általános célú alkalmazási környezetben ez a megoldás nem alkalmazható (hiszen a webes felületeket használó emberek nem esnek át előzetes betanításon), ugyanakkor feladatunk más, egyszerűbb megoldást tesz lehetővé, hiszen nem egy nagy kifejezőerejű, bonyolult kontrollált nyelvet alkalmazunk, hanem pusztán a lekérdezésekre koncentrálunk.

Megoldásként az eredetileg numerikus billentyűzetekhez kifejlesztett, ún. prediktív szövegbevitel alkalmazását választottuk [5]. A szövegbevitel során a beviteli mezőben a felhasználó mondatfogalmazását folyamatosan elemezve kijavítjuk a felhasználó nyelvi hibáit, illetve felkínáljuk számára azokat a választási lehetőségeket, amelyek a kérdés megfogalmazásának adott pillanatában a kontrollált nyelv szabályai szerint a rendelkezésére állnak.

A prediktív szövegbevitel az alkalmazott kontrollált természetes nyelvet nem korlátozza, ugyanakkor szükségessé teszi annak hatékony számítógépes reprezentációját. A rendszernek ugyanis nemcsak a felhasználó által megfogalmazott természetes nyelvű kérdés elemzésének feladatával kell megbirkóznia, hanem a kérdés feltevése során a mondattöredékek hatékony elemzése után lehetséges alternatívákat is generálnia kell a felhasználó számára – mindezt ráadásul a felhasználó gépelésével egy időben, rendkívül rövid idő alatt.

A rendszer felépítésének és működésének áttekintése

Az alábbi ábra szemlélteti a kialakított rendszer főbb komponenseit és azok kapcsolódását.



2. ábra: a rendszer főbb komponensei és azok kapcsolódása

A *Felhasználói felület* feladata egy olyan szövegbeviteli módszer biztosítása, amely folyamatosan figyeli a felhasználó által begépelt szöveget, elküldi azt a *Javaslattevő* felé, majd annak szövegjavaslatait megmutatja a felhasználónak. A *Javaslattevő* a *Generátorral* együttműködve folyamatosan elemzi a beírt szöveget, és előállítja annak lehetséges kiegészítéseit. Az *Elemző* feladata a felhasználó által megfogalmazott teljes kérdés elemzési fájának elkészítése. A *Nyelvtan* komponens támogatja mind az *Elemző*, mind a *Generátor* feladatát egy hatékony nyelvtan reprezentációval. A *Fordító* az elemzési fa alapján elkészíti a természetes nyelvű kérdés SQL megfelelőjét. Az *Adatbázis-lekérdező* végrehajtja az SQL kifejezést az adatbázison, s végül az *Eredmény-megjelenítő* megjeleníti az adatbázis-lekérdezés után visszakapott adatokat a *Felhasználói felületen* keresztül.

A rendszer működése tehát két fázisra bontható: a kérdésfeltevés támogatására és a kérdés megválaszolására. A kérdésfeltevés során a rendszer folyamatosan elemzi a felhasználó által beírt szöveget és kiegészítéseket generál hozzá (*Javaslattevő*, *Generátor*, *Nyelvtan*). A kérdés megválaszolása során elkészül egy elemzési fa, majd ezt SQL kifejezéssé fordítja a rendszer, amit végrehajt az adatbázison (*Elemző*, *Nyelvtan*, *Fordító*, *Adatbázis-lekérdező*, *Eredmény-megjelenítő*). Mindkét fázisban szükség van ugyanarra a beépített nyelvtanra, ugyanakkor eltérő céllal, azt eltérő módon alkalmazva.

A következőkben megvizsgáljuk a rendszer nyelvtanát, annak belső reprezentációját, illetve alkalmazását a generálás és elemzés során.

A kísérleti rendszer nyelvtana

A nyelvtan kidolgozása során a fő hangsúly az egyszerűségen volt: egy olyan környezetfüggetlen nyelvtant [2] alakítottunk ki, amely lehetőleg egyszerűen elemezhető, de támogatja az alkalmazásban szereplő adatbázissal kapcsolatos kérdések megfogalmazását és elemzését.

A környezetfüggetlenség mellett más követelményeket is támasztottunk a nyelvtannal szemben, amelyek egyrészt a generálás szempontjából fontosak, másrészt tovább egyszerűsítették az első kísérleti rendszer megvalósítását. A generálás szempontjából legfontosabb követelmény, hogy egy nem záró szimbólum helyettesítéseit mindig az egyszerűbbtől a bonyolultabb felé kell megadni (a legegyszerűbb helyettesítés egy záró szimbólum, majd ezek listája következik, azután a nem záró szimbólumok, végül ezek kombinációi szerepelnek). Ennek célja az, hogy a felhasználó számára javasolt kérdés kiegészítések mindig az egyszerűbbtől a bonyolultabb felé haladva jelenjenek meg.
Az első rendszer bonyolultságnak csökkentése érdekében nem alkalmaztunk rekurzív nyelvtani szabályokat (egy szimbólum nem hivatkozik – áttételesen sem – saját magára, azaz nem lehet összetett kérdéseket megfogalmazni).

Nyelvtan reprezentáció a generálás és az elemzés számára

A nyelvtant alapvetően két területen alkalmazza a rendszer: magyar nyelvű mondatok részeinek generálására, illetve teljes mondatok elemzésére. A kétféle alkalmazásban vannak közös eljárások, ezért egységes nyelvtanreprezentációt dolgoztunk ki.

A hatékony nyelvtani elemzők a mondatok elemzését egy speciális fával, az ún. **tömörített erdő (packed forest)** segítségével reprezentálják [2]. Ennek lényege, hogy a fa minden csomópontja lehet hagyományos elemzési csomópont, de lehet a csomópontok egy halmaza is. Ezáltal exponenciális számú elemzés ábrázolható polinomiális időben és tárhelyen. A nyelvtan szerint lehetséges mondatok generálása, illetve a részmondatok bővítése (a következő szimbólum generálása) szintén exponenciális feladat, melynek hatékony reprezentálására felhasználható a tömörített erdő.

A generálás céljainak megfelelően a tömörített erdő egy módosított változatát dolgoztuk ki a kontextus-független nyelvtan belső reprezentálására: a **rendezett tömörített erdőt**[1], amelyet a következő alfejezet ismertet. A későbbiekben azt is bemutatjuk, hogy ez reprezentáció a generálás mellett a nyelvtani elemzésre is használható.

A rendezett tömörített erdő reprezentáció

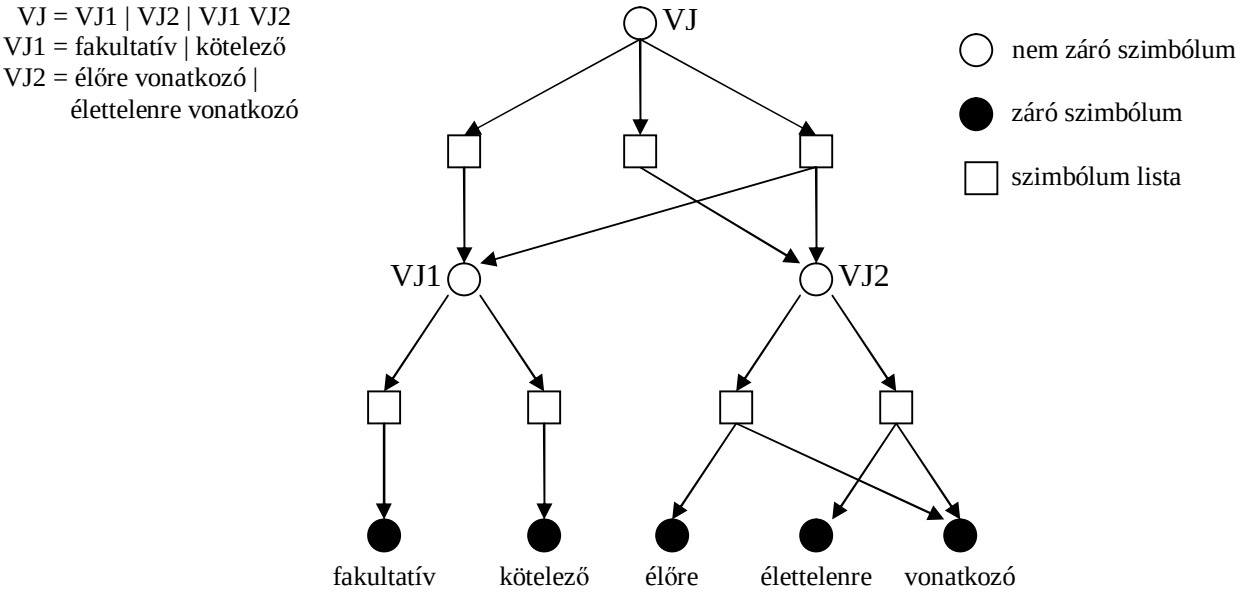
A kidolgozott megoldás célja, hogy a generálás számára hatékony, polinomiális idő és tár komplexitású nyelvtan reprezentációt nyújtson. A tömörítés lényege, hogy az összes generálható mondatfát egyetlen gráfban reprezentáljuk úgy, hogy a nyelv minden szimbóluma pontosan egyszer szerepel benne. A rendezés a gráfban meghatározza az élek kiindulási sorrendjét a csomópontokból annak érdekében, hogy a generálás a felhasználó számára az egyszerűbbtől a bonyolultabbakig sorrendben ajánlja fel a beírt kérdésrészletek kiegészítését.

A gráf kétféle csomópontot tartalmazhat: szimbólum és szimbólumlista csomópontot. A tömörítést a szimbólumcsomópontok tartalmazzák oly módon, hogy nem csak egy lehetséges helyettesítésüket adjuk meg (azaz nem csak egy lehetséges helyettesítési részfájuk van), hanem az összes lehetséges helyettesítést reprezentáljuk egyszerre a csomópontból kiinduló élek mentén. Ehhez a szimbólumok reprezentációját ketté bontjuk: a szimbólum csomópontra és az alternatív helyettesítések szimbólumlista csomópontjaira. Egy szimbólumcsomópontból kiinduló irányított élek a szimbólum alternatív helyettesítéseire mutatnak. A helyettesítések szimbólumlisták (melyeket szimbólumlista csomópontok reprezentálnak). A szimbólumlista-csomópontokból kiinduló irányított élek a listát alkotó szimbólumokra mutatnak. Ily módon a teljes gráf az élek útvonalain felváltva szimbólum és szimbólumlista csomópontokat tartalmaz, a szimbólumcsomópontokból VAGY logikai kapcsolatú, míg a szimbólumlista-csomópontokból ÉS kapcsolatú irányított élek indulnak ki.

A gráf rendezettségét a csomópontokból kiinduló élek sorrendjének meghatározása adja. Szimbólumlisták esetében a rendezési sorrend a nyelvtanban megkötött szimbólumsorrenddel egyezik meg. A szimbólumcsomópontokból kiinduló élek sorrendjét az határozza meg, hogy az adott szimbólumhoz tartozó alternatív helyettesítések milyen sorrendben vannak a nyelvtan helyettesítési szabályaiban. (Emiatt kötöttük meg a nyelvtanban a helyettesítések sorrendjét a korábbiakban ismertetett módon.) Mindezeknek megfelelően a teljes gráf rendezett éleket tartalmaz, amit a generálás során ki fogunk használni.

A 3. ábra mutat egy példát az alkalmazási környezetet ismertető fejezetben leírt különféle (fakultatív, kötelező, illetve élőre és élettelenre vonatkozó) vonzatjelzők nyelvtani szabályaira, és azok reprezentációját a tömörített erdővel. A három nem záró szimbólum (VJ, VJ1, VJ2) mindegyikének több lehetséges helyettesítése van (belőlük több, VAGY kapcsolatú él indul ki). A nyelvtani szabály a helyettesítéseket az egyszerűbbektől a bonyolultabbakig sorrendben tartalmazza, a csomópontokból kiinduló élek balról jobbra sorrendben vannak. A gráf reprezentáció a szabályban található helyettesítési sorrendeket, illetve a helyettesítések szimbólumainak sorrendjét az élek ilyen sorrendbe rendezésével őrzi meg.

[1] Szigorúan véve ez az elnevezés pontatlan, mivel a reprezentáció nem fa. Ugyanakkor a tömörítés célja fa reprezentációk összeolvasztása, az elemzés során fát választunk ki a gráfból, ezért választottuk ezt az elnevezést.



3. ábra: a vonzatjelzők (VJ) reprezentációja tömörített erdővel

Generálás a rendezett tömörített erdővel

A generálás célja, hogy a felhasználó által eddig megadott szimbólum sorozathoz megkeresse azokat a szimbólumokat, melyek a nyelvtan szerint követhetik az eddigi sorozatot. (Általánosságban a generálás a szimbólumsorozat tetszőleges ponton történő kiegészítését és módosítását is lehetővé teheti, az első megvalósításban azonban csak a sorozat folytatásának vizsgálatával foglalkoztunk.)
A lehetséges következő szimbólumok előállítása elvileg egy elemzésből és egy generálásból áll. Az elemzés elkészíti a felhasználó által eddig megadott záró szimbólumok elemzési fáját, majd megkeresi azokat a további mondatfákat, amelyek ezt a fát új záró szimbólummal egészíti ki. Az alkalmazott nyelvtan reprezentációban ez a két lépés egy adatstruktúrában, szorosan összekapcsolódva tehető meg a következők szerint.

Első lépésként azonosítani kell a felhasználó által megadott záró szimbólumokat a nyelvtanban (így a gráfban is), majd meg kell határozni azt a szimbólumot, amihez képest a sorozat folytatását keressük (**kiinduló szimbólum**). Nem üres bemeneti sorozat esetén ez az utolsó záró szimbólum. Üres bemeneti szimbólumlista esetén a nyelvtan mondat szimbólumának első záró szimbólumát (szimbólumait) adjuk vissza eredményként.

Második lépésként a kiinduló szimbólum lehetséges **következő szimbólum**ait kell meghatározni. Ez a kiinduló szimbólumot tartalmazó szimbólumlista következő gyermekének meghatározását jelenti, illetve annak hiányában a szimbólumlista által helyettesített szimbólumot követő szimbólumot kell meghatározni. Amennyiben a kiinduló szimbólum több szimbólumot is helyettesíthet, úgy a kiinduló szimbólumot megelőző záró szimbólumo(ka)t figyelembe véve kell választani a szimbólumok közül.

Ez a lépés lényegében a nyelvtani elemzést helyettesíti, kihasználja a nyelvtanra tett megkötéseket. A kiinduló szimbólumot és a szomszédos záró szimbólumokat csak addig elemzi a nyelvtani szabályoknak megfelelően, amíg a helyettesítések közötti választás nem egyértelmű. Ez csak legrosszabb esetben igényli a teljes elemzési fa elkészítését, általában annál jelentősen kisebb feladat.

Harmadik lépésként a következő szimbólumok mindegyikéhez megkeressük az első záró szimbólumokat. Az összes így előállított záró szimbólum (kiszűrve a többször szereplőket) adja azt a szimbólumlistát, ami a kontrollált bemeneti interfészen a következő szó kiválasztásához a rendszer felkínál. Ez a generálás végeredménye.

Az algoritmus ügyesebbé tehető azáltal, hogy azon záró szimbólumok esetében, melyeknek van következő záró szimbólumuk a helyettesített szimbólum listán, megadjuk azokat is eredményként. Ezáltal egy adott záró szimbólum után kötelezően szereplő záró szimbólum már szerepelni fog felhasználónak felajánlott listában a következő szó kiválasztása során (egy szó helyett egy kifejezést ajánl fel a rendszer). A 3. ábra VJ2 szimbólumának gyermekei mutatnak erre példát: a *vonatkozó* szimbólum kötelezően szerepel az *élőre* és az *élettelenre* szimbólumok után, azaz a rendszer eredményként a következő választásokat kínálja majd fel: *élőre vonatkozó, élettelenre vonatkozó*.

Elemzés a rendezett tömörített erdővel

Az elemzés célja, hogy egy teljes kérdéshez előállítsa annak elemzési fáját. Erre a célra egy lentről felfele elemzőt használ a rendszer [2].

Az elemzés elvi menete a következő. Első lépésként azonosítani kell a felhasználó által megadott záró szimbólumokat a nyelvtant reprezentáló gráfban. Üres bemeneti sorozat, vagy nem azonosítható záró szimbólum esetén az elemzés hibajelzést ad. Második lépésként az azonosított szimbólum lista elemeit helyettesítjük a nyelvtan átíró szabályainak megfelelően nem záró szimbólumokkal. Itt külön járunk el szimbólumok és szimbólumlisták esetében. Ezt a lépést addig ismételjük, míg az elemzési fa el nem készül (azaz eljutunk a fa gyökér csomópontjához), vagy zsákutcába nem jutunk (azaz nem tudunk több helyettesítést elvégezni, és nem értük el a célállapotot).

Az elemzés hatékonyságának növelése érdekében az elemzési fa előállítására is a korábban ismertetett rendezett tömörített erdőt használjuk, azaz az elemzési algoritmust egy gráf keresési feladattá alakítjuk. Az első lépés a záró szimbólumok azonosítása a gráfban. A második lépés egy ezekből kiinduló keresési feladat, amelynek során a helyettesítési kapcsolatokon lentről felfele haladva a tömörítési csomópontok (azaz a szimbólumok) alternatív elágazásai közül azt az elágazást választjuk ki, amelyiken lefele elindulva a konkrét mondatban szereplő záró szimbólumokhoz jutunk. A keresés során az alternatívák közül kiválasztott élek lesznek a mondat elemzési fájában is az élek. Ily módon az elemzés során a tömörített erdőből kiválasztódik az a fa, amelyik az adott mondat elemzési fája (vagy elakad a keresés, s így sikertelen lesz az elemzés). A fa kiválasztása a tömörítést jelentő VAGY kapcsolatok közötti választásra vezethető vissza – miután minden ilyen döntést meghoztunk a keresés során, az elemzési fa előáll.

A feltett kérdés elemzési fájának ismeretében elkészíthető annak gépi fordítása egy másik nyelvtanra, például SQL kifejezéssé.

A kontrollált lekérdező felület webes megvalósítása

A természetes nyelvű lekérdezésről szóló fejezetben már megállapítottuk, hogy rendszerünkben a kontrollált természetes nyelv sikeres használatához prediktív szövegbeviteli technika alkalmazása szükséges. Mivel a választott technológiai környezet a web, így a webrendszerekben alkalmazott megoldásokra támaszkodva alakítottuk ki a lekérdező felületet.

A megfogalmazott követelmények kielégítésére alkalmas, a felhasználó gépelését figyelő, korrigáló illetve javaslattevő rendszert egy web böngészőn belül kellett megvalósítanunk. Mivel az alkalmazás bizonyos elemei kliens oldalon nem érhetők el, illetve nem célszerű a kliens oldalt összetett számításokkal terhelni, ezért egy olyan dinamikus web alkalmazást valósítottunk meg, amely a böngésző és a webszerveren futó alkalmazás állandó, aszinkron kapcsolatára épül, a feladatokat a két

komponens között a rendelkezésre álló adatoknak és a terhelhetőségnek megfelelően osztva el. Ez a feladat az AJAX technológia [4] segítségével valósítható meg.

Az AJAX lehetővé teszi, hogy a böngészőben futó Javascript programok aszinkron módon XML adatokat cseréljenek a szerveren futó alkalmazásokkal. Esetünkben a böngészőben futó program a felhasználó gépelését figyeli, javítja az esetleges hibákat, illetve felkínálja a beírt részmondatok lehetséges folytatásait. Ehhez a szerver oldalon futó nyelvi elemző és generátor a nyelvtan és az adatbázis ismeretében szolgáltat adatokat XML formában. A teljes rendszer aszinkron módon, a felhasználó elől elrejtve működik a webes környezetben.

Összefoglalás, értékelés, továbblépési irányok

A kísérleti rendszer megvalósításának alapvető célja kettős volt: egyrészt egy kontrollált szövegbeviteli webrendszer megalkotása, másrészt egy ezt támogató hatékony nyelvtan reprezentáció, elemző és generáló algoritmus kialakítása. Az elkészült rendszer ezt teljes egészében megvalósítja, képes kontrollálni a kérdések feltevését, valamint elemezni azokat. Az elemzési fa alapján a rendszer összeállít egy SQL kifejezést, amit az adatbázis lekérdezésére használva előállítja a kérdésre adott választ.

A rendszer az alkalmazott nyelvtanra erős megkötéseket alkalmaz annak érdekében, hogy a prediktív szövegbevitel során jelentkező generálási feladatokat minél hatékonyabban oldja meg. Vizsgálandó, hogy ezek a megkötések hogyan enyhíthetők. Az alkalmazott nyelvtan reprezentáció (a rendezett tömörített erdő) hatékony, azonban alkalmazását a jelenleginél lényegesen összetettebb nyelvtanokra nem vizsgáltuk. A generálási folyamat jelenleg csak egy kérdésrészlet lehetséges folytatásainak előállítására alkalmas. A kérdések tetszőleges kiegészítésének (módosításának) lehetősége valószínűleg csak az algoritmus átgondolása után dolgozható ki.

A rendszerben alkalmazott fordítási algoritmus (az elkészült elemzési fa SQL kifejezésre alakítása) az alkalmazáshoz kötött. Egy általános fordító elkészítésénél perspektivikusabb és könnyebben megvalósítható megoldásnak tűnik az elemzési fa XML lekérdezéssé alakítása. Egy másik lehetséges továbbfejlesztési irány a rendszerben alkalmazott nyelvtan automatikus generálása az alkalmazást leíró ontológia alapján. A nyelvtanra alkalmazott megkötések, valamint a minta nyelvtan szerkezete az ontológia építésére és ilyen célú alkalmazására is jó támpontokat adnak.

Bár a rendszer bizonyos részei alkalmazáshoz kötöttek, lényegi komponensei (a nyelvtan reprezentációja, a nyelvi elemzésen alapuló prediktív szövegbevitel és annak webes megvalósítása, valamint a természetes nyelvű kérdések elemzése) alkalmazásfüggetlenek, a webrendszerek széles körében felhasználhatók webes űrlapok kiváltására, illetve kiegészítésére.

Hivatkozások

[1] Kiss Margit: Főnévi vonzatosság a magyar nyelvben, doktori értekezés, ELTE, 2005.

[2] Russel, Norvig: Mesterséges Intelligencia modern megközelítésben. Panem Könyvkiadó, pp. 920., 2005

[3] Allen, Jeffrey; Barthle, Kathleen: Introductory overview of Controlled Languages. Society for Technical Communication meeting of the Paris, France chapter. 2 April 2004.

[4] Garret, Jesse James: Ajax: A New Approach to Web Applications, Adaptive Path, http://www.adaptivepath.com/publications/essays/archives/000385.php, 2005.

[5] Smith, Sindey L; Goodwin, C. Nancy: Alphabetic Data Entry Via the Touch-Tone Pad: A Comment, The Mitre Corporation, HUMAN FACTORS, 13(2) pp 189-190., 1971.