Cuprins

Capitolul 1

Capitolul 2

Capitolul 3

Capitolul 4

Capitolul 5

Capitolul 6

Capitolul 1
Introducere

Dezambiguizarea automată a sensurilor cuvintelor a fost un subiect de interes încă din anii 1950 (perioada în care a început să se studieze mai intens domeniul lingvisticii computaționale). Dezambiguizarea sensurilor nu este un scop în sine, este un proces intermediar, necesar la un anumit nivel pentru a folosi la procesarea limbajului natural. Este, în mod evident, util pentru aplicații care necesită interpretarea limbajului, (comunicarea prin intermediul mesajelor, interacțiunea om – mașină), dar este folosit şi în domenii al căror scop principal nu este înțelegerea limbajului natural:

- **traduceri asistate de calculator**: dezambiguizarea sensurilor cuvintelor este esențială pentru traducerea riguroasă a unor cuvinte polisemantice (ex.: franțuzescul *grille* , care, în funcție de context, poate fi tradus cu scală, orar, poartă, linie ferată etc.);

- **regăsirea documentară şi parcurgerea hipertextelor**: când căutam anumite cuvinte cheie, este preferabil să eliminăm aparițiile în care sensurile acestora nu sunt cele dorite. De exemplu, când se caută în domeniul juridic cuvântul *curte*, nu este de dorit să obținem şi documentele în care cuvântul *curte* are alt sens decât cel juridic;

- **analiza tematică şi a conținuturilor** - o metodă obişnuită în analiza tematică şi a conținuturilor este să se analizeze distribuția categoriilor predefinite de cuvinte (acele cuvinte care indică un anume concept, o idee, o temă) în cadrul unui text. Importanța dezambiguizării sensurilor în acest domeniu se referă la includerea acelor instanțe cu sens corespunzător ale cuvintelor.

- **analiza gramaticală**: dezambiguizarea sensurilor este utilă ca parte a adnotării limbajului. De exemplu în următoarea frază: „*Am forțat broasca şi aceasta s-a rupt*”, este necesar să dezambiguizăm sensul cuvântului *broască* şi să îl adnotăm în mod corespunzător. Dezambiguizarea sensurilor este necesară şi pentru anumite analize sintactice, sau în parsări.

- **procesarea limbajului**: dezambiguizarea sensurilor este cerută pentru reproducerea corectă din punct de vedere fonetic al cuvintelor, sau pentru segmentarea cuvintelor în cadrul sintetizării limbajului.

- **procesarea textului**: dezambiguizarea este necesară pentru corectitudinea scrierii cuvintelor (un exemplu ar fi introducerea diacriticelor, schimbări gramaticale ale formelor cuvintelor). Alt caz ar fi accesul lexical pentru limbajele semitice (acele limbaje în care nu sunt scrise vocalele).

Problema dezambiguizării sensurilor cuvintelor a fost descrisă ca fiind *AI-completă*. O problemă este *AI-completă* dacă poate fi rezolvată doar prin rezolvarea prealabilă a tuturor problemelor dificile din cadrul inteligenței artificiale (AI), cum ar fi reprezentarea sensurilor cuvintelor şi cunoştințelor. Dificultatea dezambiguizării sensurilor a fost una din punctele centrale ale tezei lui Bar-Hillel [1960] în domeniul traducerii automate, teză în care acesta susținea ca nu există posibilitatea determinării automate a sensului cuvântului *pen* în propoziția: „*The box is in the pen*”. Argumentul lui Bar-Hillel a constituit baza pentru raportul

ALPAC, care e considerat unul din motivele abandonului majorității proiectelor de studiu ale traducerii automate în anii ’60.

Pe de altă parte, cam în aceeaşi perioadă se făcea un progres enorm în domeniul reprezentării cunoştinţelor. Acum au apărut reţelele semantice, care vor fi aplicate în studiul dezambiguizării sensurilor. În următoarele două decenii se continuă munca în domeniul dezambiguizării, în contextul cercetării limbajului natural în cadrul AI, dar şi în domeniul analizei conţinuturilor, analizei stilistice şi literare, precum şi a regăsirii documentare. În ultimii zece ani s-a observat o intensificare a eforturilor dezambiguizării automate a sensurilor, datorită accesului sporit la text procesat de maşină, precum şi datorită îmbunătăţirii metodelor statistice de identificare şi aplicare a modelelor asupra datelor.

Problema dezambiguizării sensurilor a căpătat în ultimii ani o importanţă crescută în domeniul procesării limbajului natural.

1.1 Prezentarea metodelor folosite pentru dezambiguizarea sensurilor

În termeni generali, dezambiguizarea sensurilor cuvintelor înseamnă asocierea anumitor cuvinte dintr-un text sau un discurs cu o definiţie sau un sens care se diferenţiază într-un anume mod de alte sensuri atribuite acelui cuvânt. Acest proces va implica următoarele etape :

- **determinarea tuturor sensurilor** diferite ale unui cuvânt ce prezintă o anumite relevantă pentru textului considerat.

- **modalităţi de atribuire** de sensuri pentru fiecare apariţie a cuvântului din text.

Majoritatea studiilor efectuate recent în acest domeniu pornesc de la premisa că, pentru pasul 1, avem acces la o listă de sensuri, la un grup de caracteristici, categorii şi cuvinte asociate (de ex. sinonime), la o listă de traduceri în anumite limbi străine etc.

Definiţia exactă a ceea ce înseamnă *sens* este încă o problemă care a dat naştere la numeroase polemici. Diversitatea modurilor de definire a ridicat problema compatibilităţii şi comparabilităţii studiilor efectuate în domeniul dezambiguizării sensurilor cuvintelor, şi, datorită dificultăţii găsirii unei definiţii riguroase, nu se întrevede o rezolvare în următorii ani.

Pe de altă parte, încă de la începutul studiului dezambiguizării sensurilor cuvintelor, au existat discuţii pe tema faptului că problemele dezambiguizării morfo - sintactice şi cele ale dezambiguizării sensurilor ar trebui privite din acelaşi unghi de vedere. Aceasta înseamnă că, pentru homonime, care sunt părţi diferite de vorbire (de ex.: *haina*), dezambiguizarea morfo - sintactică reuşeşte să realizeze şi dezambiguizarea sensului. De aceea, dezambiguizarea sensurilor cuvintelor a acordat o importanţă sporită determinării sensurilor homonimelor ce aparţin aceloraşi categorii sintactice.

Pasul 2, cel al atribuirii sensurilor cuvintelor este îndeplinit prin referinţă la:

- **Contextul cuvântului** al cărui sens trebuie determinat. Acesta include informaţiile conţinute în cadrul textului sau discursului în care apare cuvântul, precum şi informaţii asupra textului (aceste ultime informaţii nu ţin neapărat de lingvistică).

- **Surse de cunoaştere externe**, care includ resurse lexicale, enciclopedice, dar şi surse de cunoştinţe construite în scopul furnizării de date utile pentru asocierea cuvânt - sens.

Procesul de dezambiguizare include potrivirea contextului instanţei cuvântului al cărui sens trebuie dezambiguizat cu informaţiile din sursele externe (în acest caz vorbim de **dezambiguizarea sensurilor cuvintelor orientată cunoştinţe**), sau informaţii despre contextele instanţelor cuvintelor care au fost deja dezambiguizate (**dezambiguizarea sensurilor cuvintelor orientată date**). Metodele de asociere sunt utilizate pentru a determina cea mai potrivită asociere între contextul curent (cel din textul considerat) şi oricare din sursele externe de informaţie.

1.1.1 Dezambiguizarea sensurilor cuvintelor aplicată în traducerea automată

Primele încercări de dezambiguizare automată a sensurilor cuvintelor au fost făcute ca parte integrantă a încercării traducerii automate a textelor. Weaver [1949 – publicat 1955] subliniază necesitatea dezambiguizării sensurilor cuvintelor în traducerea automată, şi dă o euristică pentru abordarea acesteia. Conform ideii lui Weaver dezambiguizarea automată independentă de context nu este posibilă. Avem nevoie de informaţiile adiţionale oferite de celelalte cuvinte din jurul cuvântului al cărui sens îl dorim dezambiguizat. Întrebarea este cât de mare este această informaţie suplimentară la care suntem nevoiţi să recurgem.

Pe aceeaşi linie se încadrează şi studiul lui Kaplan [1950], care încearcă să răspundă la întrebarea formulată de Weaver prezentând cuvinte ale căror sensuri este ambiguu, atât în contextul lor original, cit şi într-un context alternativ, în care sunt puse la dispoziţie unul sau două traduceri ale cuvintelor pentru fiecare sens. Kaplan a observat că nu se îmbunătăţeşte cu mult performanţa faţă de cazul când ne este pusă la dispoziţie o frază întreagă.

Reifer [1955] introduce noţiunea de *coincidenţă semantică* dintre un cuvânt şi contextul acestuia. Această noţiune a devenit în scurt timp un factor important în domeniul dezambiguizării sensurilor cuvintelor. Reifer ţine seama şi de complexitatea conţinutului textului în teoria sa, bazându-se mult pe relaţiile sintactice. Observarea şi folosirea structurii gramaticale a unui cuvânt ajută mult la identificarea sensului pe care acel cuvânt îl are în cadrul frazei studiate.

Un pas ulterior a fost încercarea de aplicare a principiilor logice şi matematice dorindu-se aducerea cuvintelor unei limbi la o reprezentare semantic - conceptuală. Richens[1958] şi Masterman[1962] au formulat noţiunea de reţea semantică, şi pe baza acesteia s-a construit prima bază de cunoştinţe automată: **Roget's Thesaurus**.

1.1.2 Metode bazate pe tehnici de inteligenţă artificială

Metodele bazate pe tehnici de inteligenţă artificială, au apărut la începutul anilor '60 şi au încercat abordarea problemei înţelegerii limbajului. Ca o consecinţă, dezambiguizarea sensurilor cuvintelor în inteligenţa artificială a fost realizată în contextul sistemelor mari, folosite pentru înţelegerea deplină a limbajului. La acea vreme aceste sisteme aveau la baza principii derivate din studiul modului uman de înţelegere şi procesare a limbajului. Aceste principii includeau însă şi o bază de cunoştinţe asupra sintaxei şi semanticii cuvintelor, bază de cunoştinţe folosită apoi în scopul dezambiguizării sensurilor cuvintelor.

1.1.3 Metode simbolice

După cum s-a menționat anterior, rețelele semantice au apărut la sfârșitul anilor ’50, începutul anilor ’60, fiind aplicate imediat în reprezentarea sensurilor cuvintelor. Richens[1958] şi Quillian[1961, 1968] au construit la mijlocul anilor ’60 o rețea care făcea legătura între cuvinte (*tokens*) şi concepte (*types*), legăturile fiind etichetate cu relații semantice sau indicând simple asocieri între cuvinte. Rețeaua a fost construită pornind de la definițiile luate din dicționar, dar a fost îmbunătățită prin adăugarea de informații noi dobândite prin intermediul cunoştințelor (aceste informații erau adăugate manual). Când două cuvinte sunt date ca intrare rețelei aceasta simulează activarea nodurilor pe o cale de legături originând din cuvintele de intrare (*marker passing*). Dezambiguizarea este realizată doar datorită faptului că un nod concept asociat unui cuvânt dat ca intrare este posibil să fie situat pe drumul cel mai scurt dintre două cuvinte de intrare.

Hayes [1976, 1977, 1978] va folosi noțiunea de **cadru** (*frame*). El a comprimat informațiile despre cuvinte, precum şi rolurile şi relațiile acestora, şi va combina cadrele şi rețelele semantice pentru a reprezenta sensurile substantivelor şi legăturile acestora cu sensul verbelor. Relațiile din cadrul acestei rețele sunt de tipul IS-A şi PART-OF.

1.1.4 Metode bazate pe conexiuni

Cercetarea în domeniul psiho-lingvisticii în anii ’60 şi ’70 a stabilit că dezvoltarea semantică (*semantic priming*) – un proces prin care introducerea unui anume concept va influența şi facilita procesarea conceptelor introduse ulterior - joacă un rol extrem de important în dezambiguizarea care are loc la nivel mental. Aceasta idee a fost materializată în modele de răspândire a activităților (*spreading activities*) – Collins şi Loftus [1975], Anderson [1976, 1983] - unde conceptele unei rețele semantice sunt activate în timpul folosirii acestora, iar activitățile ating nodurile accesibile lor. Bineînțeles că sensul îşi pierde din intensitate pe măsură ce se ating şi alte cuvinte, dar la anumite centre se poate ajunge din mai multe direcții, drept urmare, pot să îşi reîntregească sensul. Modelul a fost îmbunătățit de McClelland şi Rumelhart [1981], care au adăugat şi noțiunea de inhibiție între noduri, în cazul în care activarea suprimă anumiți vecini.

1.1.5 Metode bazate pe reprezentarea cunoştințelor

Cercetarea din domeniul inteligenței artificiale din anii 1970 şi 1980 a avut o importanță mai mult teoretică decât practică pentru înțelegerea limbajului natural. Un impediment principal în încercarea dezambiguizarea sensurilor cuvintelor a constituit-o dificultatea şi costul ridicat al lucrului cu cantitățile mari de cunoştințe necesare în dezambiguizarea sensurilor cuvintelor.

Schimbarea a intervenit în jurul anilor 1980 când au devenit disponibile resursele lexicale cuprinzătoare cum ar fi dicționare, tezaure şi corpora.

S-a început încercarea extragerii automate a cunoştințelor din aceste surse, şi, mai recent, s-au construit manual baze de cunoştințe.

La aceasta dată se mai observă şi o încercare de trecere de la metodele bazate pe teorii lingvistice la metode empirice, precum şi creşterea interesului pentru procese intermediare din procesarea limbajului natural (cum ar fi dezambiguizarea sensurilor cuvintelor), față de sisteme care să rezolve integral aceasta problemă.

1.1.6 Metode bazate pe corpus

Au existat două impedimente principale în extragerea de informații din corpora: situațiile de repartiție rară a datelor şi dificultățile legate de etichetarea manuală a cantităților mari de date.

Etichetarea automată a sensurilor cuvintelor

Etichetarea manuală a sensurilor cuvintelor este extrem de costisitoare, la ora actuală existând foarte puține texte ale căror cuvinte au sensurile etichetate. S-au încercat mai multe astfel de etichetări : Linguistic Data Consortium pune la dispoziție un corpus de aproximativ 200.000 de propoziții în care toate aparițiile a 191 de cuvinte au sensurile etichetate. Un alt exemplu ar fi Cognitive Science Laboratory care au etichetat aproximativ 1000 de cuvinte cu sensurile din WordNet. Cu toate acestea dimensiunile acestor corpusuri sunt mult prea mici în comparație cu cele necesare pentru metode statistice.

Mai multe eforturi au fost făcute pentru etichetarea automată a sensurilor cuvintelor prin metoda *boot-strapping*. Hearst [1991] a propus un algoritm bazat pe mai multe etape. Într-o primă etapă se încearcă dezambiguizarea manuală a fiecărei apariții a unei mulțimi de substantive. Într-o etapă ulterioară se folosesc informațiile căpătate din prima etapă pentru a se dezambiguiza şi celelalte apariții. De asemeni, în momentul în care dezambiguizarea unui cuvânt se face cu certitudine, se adaugă informația la baza de cunoştințe, realizându-se astfel îmbunătățirea acesteia.

Schutze [1993] a propus o metodă care evita etichetarea fiecărei apariții în corpusul de bază. Metoda lui, bazată pe regăsire documentară, găseşte automat *clustere* ale cuvintelor în text, reprezentând sub forma unui vector fiecare cuvânt al cărui sens vrem să îl dezambiguizăm. Atribuim fiecărui *cluster* un sens, fiecare sens putând fi reprezentat de mai multe *clustere*. Această metodă reduce intervenția manuală asupra textului, dar tot necesită examinarea sutelor de apariții ale fiecărui cuvânt al cărui sens este ambiguu.

Tratarea situațiilor de repartiție rară a datelor

Una dintre problemele de bază pentru procesarea corpusului este problema repartiției datelor. Această problemă are implicații profunde în dezambiguizarea sensurilor cuvintelor: în primul rând cantități mari de text sunt necesare pentru a asigura faptul că toate sensurile unui cuvânt polisemantic sunt reprezentate (având în vedere dispersia frecvenței sensurilor). De exemplu în Brown Corpus (conține 1 milion de cuvinte), cuvântul *ash* (stejar) apare numai de 8 ori, dintre care o singură dată cu sensul de stejar. Alt sens al cuvântului (cenuşă), deşi destul de comun, nu apare niciodată. Este clar că există posibilitatea de a nu găsi multe din sensurile comune ale unui cuvânt polisemantic într-un corpus, oricât de mare ar fi acesta.

Regularizarea este utilizată pentru tratarea aparițiilor evenimentelor, şi în particular pentru a asigura că pentru evenimentele ignorate nu se presupune că ar avea probabilitatea 0.

Cele mai cunoscute astfel de metode sunt ale lui Turing - Good (1953), care presupune o distribuție binomială a evenimentelor şi Jelinek şi Mercer (1985) care combină parametrii de estimare pentru diverse părți ale corpusului.

Modelele orientate pe clase încearcă obținerea celor mai bune estimări prin combinarea rezultatelor obținute asupra claselor de cuvinte ce se consideră aparținând aceleiaşi categorii. Aceste metode rezolvă parțial situațiile de repartiție rară a datelor şi

elimină necesitatea etichetării anterioare a datelor. Pe de altă parte, se pierd anumite informații datorită presupunerii că toate cuvintele unei clase au comportament similar.

1.1.7 Metode bazate pe similaritate

Aceste metode (Dagan, Marcus, Markovitch [1993]; Dagan, Pereira, Lee [1994]; Grishman, Sterling [1993]) pornesc de la ideea grupării observațiilor asupra cuvintelor similare fără a le include în aceeaşi clasă. Fiecare cuvânt are o mulțime potențială de cuvinte similare.

Analog cu metodele orientate pe clase, aceste metode folosesc o metrică de evaluare a modelelor aparițiilor. Karov şi Edelman [1998] propun o extensie a metodelor bazate pe similaritate, cu ajutorul unui proces iterativ care dă rezultate de o acuratețe ridicată (până la 92%). Aceste rezultate sunt cu atât mai impresionante cu cât ele au nevoie de o cantitate relativ mică de date de intrare spre deosebire de celelalte metode la care este necesar un corpus destul de consistent.

1.2 Dicționare pentru procesarea automată

Dicționarele pentru procesarea automată (Machine Readable Dictionaries) au devenit una din sursele principale de cunoaştere folosită pentru procesarea limbajului natural. Acestea pun la dispoziție cantități mari de informații asupra sensurilor cuvintelor, şi, drept urmare, au fost utilizate intens în cercetarea dezambiguizării sensurilor cuvintelor.

Lesk [1986] a pus bazele unei surse de cunoştințe, care asocia fiecărui cuvânt din dicționar o signatură alcătuită din lista cuvintelor care apar în definiția acelei unități lexicale. Dezambiguizarea se realiza selectând sensul cuvântului ca acel sens a cărui signatura conținea cel mai mare număr de sensuri comune între acel cuvânt şi cuvintele vecine din context. Această metodă reuşea dezambiguizarea corectă în procent de 50-70%, în funcție de finețea sensurilor cuvintelor conținute în dicționar. Metoda lui Lesk este extrem de sensibilă la celelalte cuvinte din context: absenta unui cuvânt poate modifica în mod fatal rezultatul dezambiguizării.

Wilks [1990] a încercat să îmbunătățească cunoştințele asociate cu fiecare sens, calculând frecvența aparițiilor cuvintelor în cadrul definițiilor acestora determinând astfel un mod de a calcula gradul de interdependență dintre acestea. Aceasta metrică este folosită prin intermediul unui vector care face legătura dintre fiecare cuvânt şi contextul în care apare acesta. Acuratețea cu care se realizează dezambiguizarea sensurilor cuvintelor este relativ mare: pentru cuvântul *bank* s-a reuşit în procent de 45% identificarea sensului, şi 75% identificarea omonimelor .

Metoda lui Lesk a fost extinsă (Ide şi Veronis [1990]), prin crearea unei rețele neuronale din definițiile cuvintelor (preluate din Collins English Dictionary), în care, fiecare cuvânt este conectat la sensurile sale, care, la rândul lor sunt conectate la cuvintele conținute în definiții, care la rândul lor sunt conectate la sensurile acestora etc.

1.2.1 Tezaure lingvistice

Tezaurele lingvistice pun la dispoziție informații asupra relațiilor dintre cuvinte, în principal despre sinonimia dintre acestea.

Fiecare apariție a aceluiaşi cuvânt în cadrul mai multor categorii ale tezaurului reprezintă sensuri diferite pe care le poate avea cuvântul. O mulțime de cuvinte din aceeaşi categorie reprezintă o mulțime de cuvinte înrudite din punct de vedere semantic.

În mod analog cu dicționarele pentru procesarea automată , un thesaurus este o sursă creată pentru utilizarea de către oameni, deci nu poate fi o sursă perfectă de informație în ceea ce privește relațiile cuvintelor. Este cunoscut faptul că nivelele superioare ale ierarhiei conceptelor sunt, câteodată, inconsistente iar posibilitatea de stabilire a categoriilor semantice este redusă datorită cuprinsului redus de informație. Pe de altă parte un tezaur pune la dispoziție o rețea aproape completă de asocieri între cuvinte, şi o mulțime de categorii semantice ce pot fi utile în procesarea automată a textului.

1.2.2 Dicționare computaționale

La mijlocul anilor 1980, s-a încercat în repetate rânduri construirea ne-automată a bazelor mari de cunoştințe – WordNet (Miller, Fellbaum et al. 1990), CyC (Lenat şi Guha, 1990), ACQUILEX (Briscoe, 1991), COMLEX (Grishman, Macleod şi Meyers, 1994). Există două direcții principale în construirea acestor dicționare semantice : metoda enumerativă în care sunt date explicit sensurile cuvintelor, şi metoda generativă în care informația semantică asociată cu un anumit cuvânt este subînțeleasă şi sunt folosite reguli de generare pentru a deriva sensul exact al cuvântului.

1.2.3 Dicționare generative

Majoritatea cercetării asupra dezambiguizării sensurilor cuvintelor s-a bazat pe sensurile enumerate în dicționare ale unui cuvânt. Pustejovski [1995] va folosi metoda generativă în încercarea de a rezolva problema dezambiguizării, metoda prin care sensurile înrudite ale cuvintelor nu sunt enumerate ci generate pe baza anumitor reguli. Dezambiguizarea sensurilor cuvintelor pentru determinarea unor astfel de reguli începe cu etichetarea semantică ce duce la o reprezentare complexă a cunoştințelor, prin care se pun în evidență, într-un mod sistematic, toate sensurile cuvântului. Un exemplu în acest sens ar fi CORELEX – Buitelaar (1997).

1.3 WordNet

La ora actuală, dintre dicționarele enumerative, WordNet este unul dintre cele mai cunoscute şi mai utilizate instrumente folosite la dezambiguizarea sensurilor cuvintelor în limba engleză. Există o serie de versiuni ale WordNet pentru diferite limbi vest şi est europene.

Primul WordNet a fost realizat, pentru limba engleză, la Universitatea din Princeton, de către o echipă condusă de George Miller.

WordNet combină mai multe instrumente pentru dezambiguizarea sensurilor cuvintelor în unul singur: include definiții pentru sensurile individuale ale cuvintelor, defineşte "*synset-uri*" - mulțimi de cuvinte sinonime între ele care definesc un concept lexical. Aceste mulțimi sunt organizate într-o ierarhie, incluzând o serie de alte legături între cuvinte: hiponimie, hipernimie, antonimie, etc. Un alt motiv pentru care WordNet este atât de utilizat este şi faptul că este disponibil oricui, oricând.

1.3.1 Descrierea conceptului wordnet

Conceptul wordnet este format din: mai multe fişiere lexicografice, cod pentru transformarea acestor fişiere într-o bază de date, şi diferite modalităţi de căutare şi afişare a informaţiilor conţinute în baza de date.

Fişierele lexicografice organizează substantive, verbe, adjective şi adverbe pe grupe de sinonimii, şi descriu relaţiile dintre aceste grupe.

1.3.2 Organizarea bazei de date

Informaţia conţinută în cadrul WordNet, este organizată în grupuri logice numite grupuri de sinonimii (*synset*). Fiecare synset va conţine o listă de sinonime sau de cuvinte ce fac parte din expresii simple (ex.: a lua parte), precum şi o mulţime de pointeri care descriu relaţiile dintre un synset şi alte synset-uri. Un cuvânt (fie că apare ca sinonim, fie că apare ca parte a unei expresii) poate exista în mai multe synset-uri (poate avea şi părţi de vorbire diferite, cu sensuri diferite, etc.). Cuvintele componente unui synset sunt astfel organizate încât pot fi interschimbate.

Pointerii de care aminteam mai sus descriu două tipuri de relaţii: lexicale şi semantice. Relaţiile lexicale sunt definite pe mulţimea formelor cuvintelor, cele semantice pe mulţimea sensurilor cuvintelor. Aceste relaţii includ: hipernimia, antonimia, şi meronimia.

Substantivele şi verbele sunt organizate pe ierarhii determinate de relaţiile de hiper - hiponimie dintre synset-uri. Sunt folosiţi pointeri adiţionali pentru indicarea altor relaţii.

Capitolul 2
Descrierea modelului propus

Pentru un cuvânt, în WordNet, sunt identificate unul sau mai multe sensuri posibile. Informația despre aceste sensuri este structurată folosind grupările logice numite *synset-uri*. Fiecare synset este constituit dintr-o listă de cuvinte sau grupuri de cuvinte sinonime şi din relații între acest synset şi alte synset-uri. Un cuvânt poate apărea în mai multe synset-uri şi ca parte de vorbire diferită. Ideea din spatele listei de cuvinte a unui synset este că, dacă un cuvânt are sensul respectiv el poate fi înlocuit cu oricare din celelalte cuvinte.

Relațiile dintre synset-uri sunt de două tipuri: *lexicale* şi *semantice*. Relațiile lexicale sunt relații între formele cuvintelor, iar cele semantice sunt relații între sensurile cuvintelor. Câteva dintre aceste relații sunt: *hipernimie – hiponimie, antonimia, meronimie – holonimie (relații de tip part of).*

Un synset s1 este în relație de hipernimie cu un synset s2 (este un hipernim al lui s2) dacă s2 este un caz particular (a kind of) al lui s1. În acest caz, s2 este un hiponim (este în relație de hiponimie). De exemplu, pisica este un hiponim pentru felină, care la rândul lui este hiponim pentru mamifer. În general, un synset are un singur hipernim, dar există şi cazuri de synset-uri cu două hipernime.

Putem obține, folosind relațiile de hipernimie şi hiponimie, o reprezentare arborescentă a synset-urilor pentru substantive şi verbe.

Pe scurt, modelul pe care l-am implementat „învață" dintr-un corpus suficient de mare pentru un synset care sunt synset-urile care apar alături de el în text, apoi foloseşte datele obținute pentru a dezambiguiza cuvintele din textul de test. În modulul de învățare apar două probleme principale: numărul mare de synset-uri şi probabilitatea foarte mică ca majoritatea cuvintelor să apară de suficient de multe ori şi acoperind toate sensurile posibile. Experimente anterioare au demonstrat că este posibil, chiar dacă se folosesc corpusuri foarte mari, ca un sens destul de răspândit al unui cuvânt, sau chiar cuvinte importante să nu apară sau să apară de foarte puține ori, ceea ce influențează grav deducțiile bazate pe datele obținute.

Prin **tabelă de co-ocurențe** vom înțelege în continuare o tabelă (matrice), construită pentru synset-uri, în care elementul aflat la poziția [i; j] reprezintă numărul de apariții ale synset-ului numerotat cu *i* în stânga celui numerotat cu *j* într-un corpus de învățare.

Modelul implementat încearcă să rezolve sau, unde nu este posibil, să reducă amploarea acestor probleme folosind structurarea ierarhică a synset-urilor în WordNet. Pentru părțile de vorbire pe care s-a lucrat – verbe şi substantive – în WordNet există 12.745, respectiv 74.488 synset-uri, deci un total de 87.233 synset-uri, pentru care ar fi imposibil şi inutil de creat o tabelă de co-ocurențe, deoarece ar fi nevoie de un corpus în care toate aceste synset-uri să apară cel puțin o dată alături de toate synset-urile care le alcătuiesc vecinătatea semantică. În cele ce urmează, pentru simplificarea acestei probleme, am profitat de faptul că un concept mai general pentru un synset (un hipernim) va include în vecinătatea sa semantică şi pe cea a hiponimelor sale. Astfel, ne putem limita la a construi o tabelă de co-ocurențe numai pentru synset-urile mai generale, în speță synset-urile care apar pe nivelul ierarhic superior în arborele de hipernime descris în WordNet.

Dacă privim structura de hipernime a sensurilor unui cuvânt, se observă că, în general, dacă am putea dezambiguiza synset-ul din vârful ierarhiei, numărul de sensuri din care trebuie să facem selecția se reduce, uneori identificarea acestuia ducând la dezambiguizarea completă a sensurilor cuvântului respectiv.

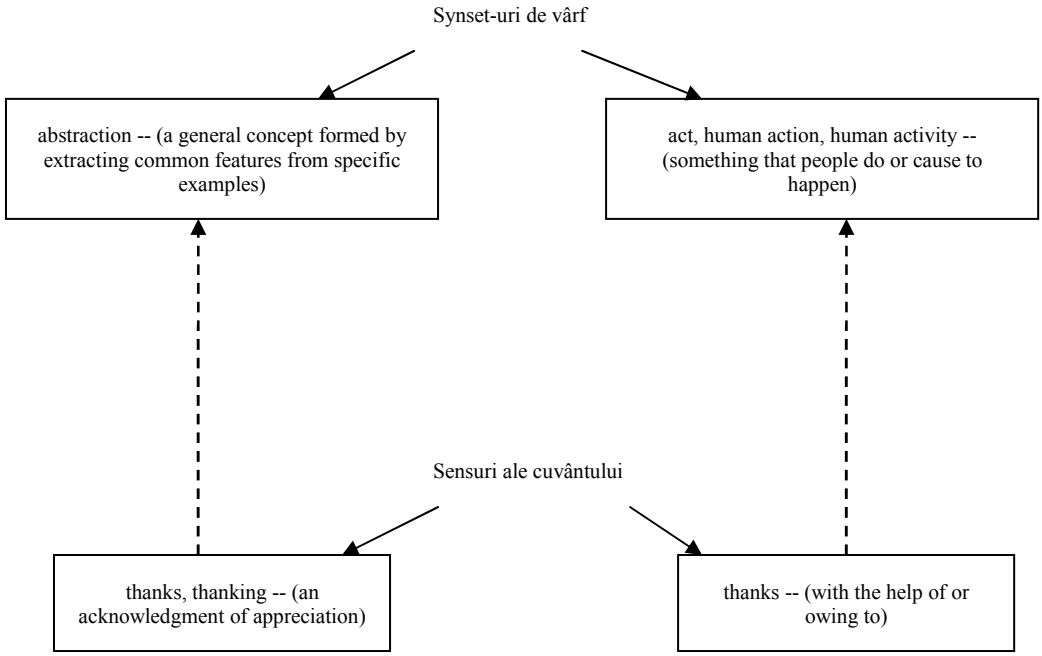

Fig. 2.1 Arborele simplificat de hipernime pentru substantivul *thanks*.

În exemplul de mai sus, alegerea unui synset de vârf pentru substantivul *thanks* duce la dezambiguizarea completă a cuvântului, cele două sensuri ale acestuia având ca hipernime pe nivelul superior al ierarhiei synset-uri distincte.

Modelul propus se bazează pe această observație, pe datele conținute într-un corpus de învățare şi pe organizarea synset-urilor în WordNet. Implementarea sa a fost împărțită în două module:

- **un modul de învățare**, în care se folosesc informații morfologice, sintactice şi semantice pentru a construi o tabelă de co-ocurențe;
- **un modul de dezambiguizare**, în care se selectează un sens pentru fiecare cuvânt pe baza datelor adunate în primul modul.

Capitolul 3
Arhitectura modulului de învățare

3.1 Construcția arborelui de hipernime ale sensurilor unui cuvânt

Primul pas pentru obținerea synset-urilor hipernime supreme pentru synset-urile ce reprezintă sensurile posibile ale unui cuvânt este construcția unui arbore care va avea ca noduri frunze pe acestea din urmă. Vom încerca să evidențiem în arbore nodurile „speciale" – nodurile frunză, fără părinte şi cele cu doi sau mai mulți fii, în vederea construirii unui arbore simplificat.

Datele de intrare se constituie din synset-urile WordNet corespunzătoare sensurilor cuvântului şi din hipernimele acestora. Ieşirea se constituie dintr-un arbore de hipernime – HypernimsTree – compus din HypernimNode-uri. În cele ce urmează prin obiect de tip Synset se va înțelege un obiect care conține datele despre un synset găsite în clasa *wordnet.wn.Synset*, adică date de tipul parte de vorbire, lista de cuvinte sinonime ce pot fi interschimbate dacă apar cu acest sens, glosa WordNet – o definiție - dicționar a sensului, etc.

Un nod din arbore conține, pe lângă Synset, legătura către părinte (dacă are), un identificator unic, precum şi faptul dacă este nod – confluență (un nod cu mai mulți fii) sau nod care are ca părinte un nod – confluență. HypernimsTree – ul este reprezentat „întors", memorându-se o listă cu nodurile de pe nivelul inferior în loc să se memoreze o rădăcină, care ar fi trebuit să fie una fictivă.

Principalele probleme întâlnite în construcția arborelui au fost synset-urile cu mai mult de un hipernim şi câteva situații speciale de genul celei în care un sens de bază are ca hipernim (şi nu neapărat direct) un alt sens de bază al cuvântului.

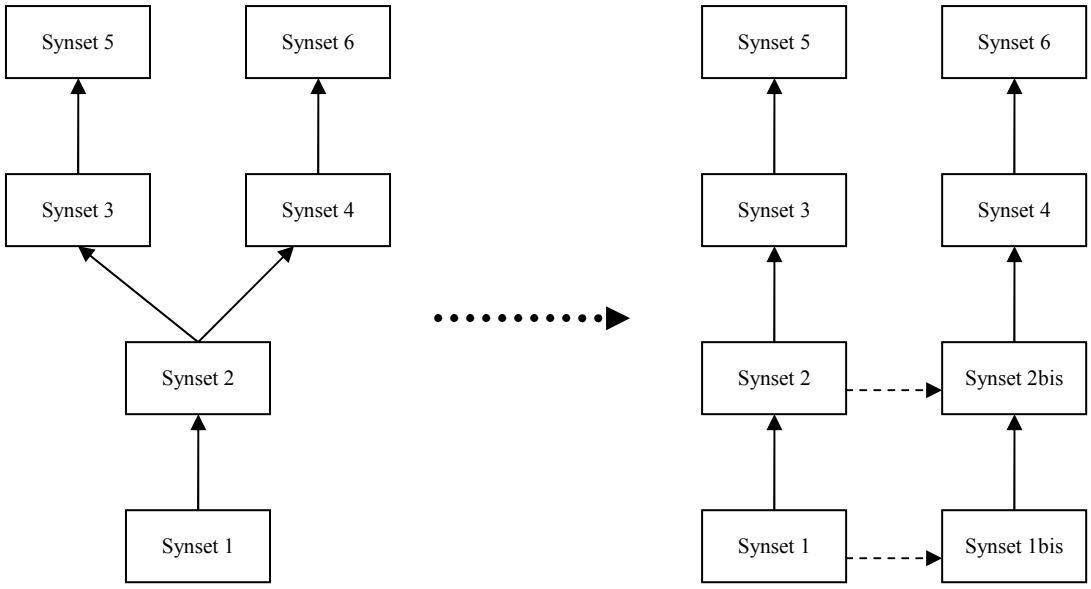

Fig. 3.1 Tratarea cazului special al synset-urilor cu mai multe hipernime.

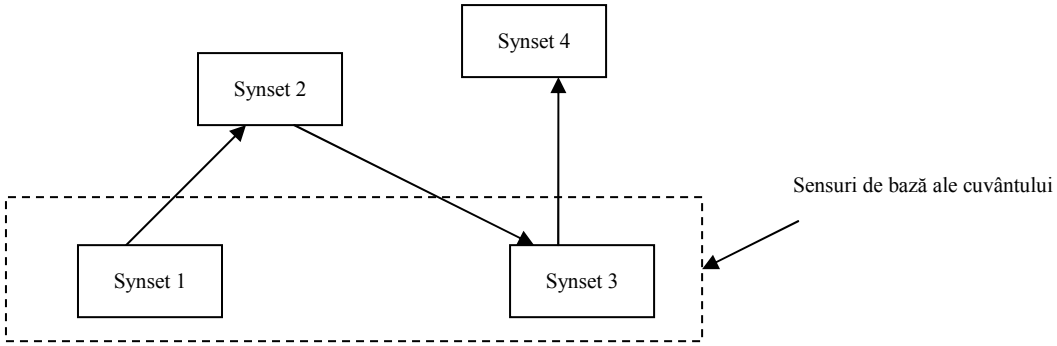

Fig. 3.2 Cazul special al synset-urilor de bază în relație de hipernimie.

Pentru cazul synset-urilor cu mai mult de un hipernim, soluția este dublarea ramurii construite până la synset-ul respectiv (Fig. 3.1). Pentru situația în care două synset-uri de bază ale cuvântului se află în relație de hipernimie (Fig. 3.2), structura de arbore este păstrată, dar ambele synset-uri trebuie să rămână în lista de noduri – frunze, deşi Synset-ul 3 nu mai corespunde definiției acestora.

Construcția arborelui se face de jos în sus, parcurgându-se listele de hipernime pentru sensurile de bază şi unificând ramurile când este necesar.

Algoritmul de construcție a unei ramuri pornind de la un synset de bază este următorul:

```
int ConstructBranch (Synset sense, int NrRamura, int NrMaxRamuri, boolean
bAddFirst)
{
      Coada = sense.GetAllHypernims();
      Maxim = NrMaxRamuri;
      // bAddFirst este true doar la apelarile pentru sensurile de baza,
care sunt deja
      // in arbore
      If (!bAddFirst)
            Sare peste primul element din coada;

      While (!Coada vida && !Unire && !Despartire)
      {
            sens = urmatorul element din Coada;
            if (nivel(sens) == ultimul_nivel)
            {     // bifurcatie - se apeleaza recursiv pentru noile ramuri
                  Creaza o ramura noua identica cu cea construita pana acum
                  Despartire = true;
                  PozitieRamuraNoua = ++Maxim;
                  //Construieste ramura curenta in continuare
                  Maxim = ConstructBranch(sens, NrRamura, Maxim, false);
                  //Construieste ramura paralela
                  Maxim = ConstructBranch(sens, NrRamuraNoua, Maxim,
false);

            }
            else
                 if (!Despartire)
                 {
                  if (sens exista deja in abore)
```

```
                        Unire = true;
                  Else
                        Adauga nod la ramura in constructie
                  // ca sa ma opresc trebuie sa verific daca urmatorul sens
din
                  // Coada nu este pe acelasi nivel – caz special cu o
bifurcare si
                  // unire in acelasi timp
                  sens1 = urmatorul element din Coada;
                  if (nivel(sens) == nivel (sens1))
                  {
                    Creaza o ramura identica cu cea construita pana acum
                    Despartire = true;
                    PozitieRamuraNoua = ++Maxim;
                    //Construieste ramura curenta in continuare
                    Maxim=ConstructBranch(sens1, NrRamura, Maxim, false);
                    //Construieste ramura paralela
                    Maxim=ConstructBranch(sens1, NrRamuraNoua, Maxim,
false);
                  }
                 }
            ultimul_nivel = nivel(sens);
      }
      return Maxim;
}
```

După cum se observă, pentru a păstra structura arborescentă pentru synset-urile cu mai mult de un hipernim se dublează ramurile de la punctul în care există doi părinți în jos. La adăugarea unui nod în arbore se verifică dacă nu există deja un nod similar în altă ramură şi dacă un astfel de nod există acesta este marcat ca nod de confluență. Dacă la construcția unei ramuri se ajunge sau se creează un astfel de nod, construcția poate fi oprită deoarece şirul de hipernime va continua pe ramura deja creată, cu care s-a realizat intersecția.

3.2 Construcția arborelui de hipernime simplificat

Ideea modelului este reducerea numărului de sensuri din care trebuie să alegem pe măsură ce coborâm în arbore. Pentru dezambiguizarea sensurilor unui cuvânt nu avem nevoie de tot şirul de hipernime ale unui sens.

După cum se observă din structura unui arbore de hipernime ale sensurilor unui cuvânt, nodurile de confluență sunt singurele în care trebuie să alegem calea de urmat. Prin urmare, pentru următoarea fază a prelucrării este mult mai util un arbore de hipernime simplificat, construit pe baza arborelui creat anterior, păstrând din acesta numai nodurile frunză, nodurile de confluență şi nodurile de vârf (noduri fără părinte). Utilitatea existenței arborelui intermediar este dată de multitudinea de cazuri speciale (ramuri care trebuie dublate, synset-uri de bază care au ca hipernime alte synset-uri de bază, dublări de ramură concomitente cu intersecția cu o altă ramură) care ar fi fost mult mai greu de detectat şi prelucrat simultan cu simplificarea arborelui.

Spre deosebire de HypernimsTree, în care ne interesa numai o accesare de jos în sus a nodurilor, în SimplifiedHypernimsTree avem nevoie atât de o parcurgere descendentă pornind dintr-un nod rădăcină fictiv, cât şi de o parcurgere ascendentă pornind de la lista de noduri frunză. Pentru a optimiza ambele parcurgeri, am optat pentru memorarea în cadrul structurii arborelui atât a rădăcinii cât şi a listei de synset-uri de bază a cuvântului. Acest model a impus memorarea în cadrul SimplifiedTreeNodes a legăturii către părinte şi a listei de fii. Necesitatea optimizării pentru ambele tipuri de parcurgeri, şi deci necesitatea parcurgerilor,

este dată de faptul că avem nevoie de acces rapid la sensurile de bază, dar şi la toate synset-urile de pe un anumit nivel – în special cele de pe primul nivel, adică toate synset-urile vârfuri în ierarhie – şi la toate synset-urile care au ca hipernim (direct sau indirect) un nod din arbore.

Algoritmul de obținere a arborelui se hipernime simplificat primeşte la intrare un arbore de hipernime normal:

```
SimplifiedTree (HypernimsTree hTree)
{
      // folosesc o lista cu nodurile de confluenta deja introduse in
      // arbore pentru a mari viteza de cautare
      ListaIntersectii = "";
      Creeaza nod fictiv root
      Adauga toate nodurile de baza din hTree
      pentru sensurile de baza fara parinte in hTree
            parent = root

      Pentru fiecare din sensurile de baza
      {
            NodCurent = nodul de baza
            Urc in hTree pana intalnesc un nod in unul din cazurile:
            {
                  if (Nod care se leaga la un nod confluenta)
                  {
                        if (parintele in ListaIntersectii)
                        {
                              NodCurent.Parent = parintele din lista
                              Terminat cu ramura curenta
                        }
                        else
                        {
                              Creeaza NodNou
                              NodCurent.Parent = NodNou
                              NodCurent = NodNou
                              ListaIntersectii += NodNou
                        }
                  }
                  if (Nod confluenta la care se vor mai lega alte noduri)
                  {
                        if (nod in ListaIntersectii)
                        {
                              NodCurent.Parent = parintele din lista
                              Terminat cu ramura curenta
                        }
                        else
                        {
                              Creeaza NodNou;
                              NodCurent.Parent = NodNou;
                              NodCurent = NodNou;
                              ListaIntersectii += NodNou;
                        }
                  }
                  if (Nod fara parinte)
                  {
                        Creeaza NodNou;
                        NodCurent.Parent = NodNou;
                        NodNou.Parent = root;
                  }
            }
      }
}
```

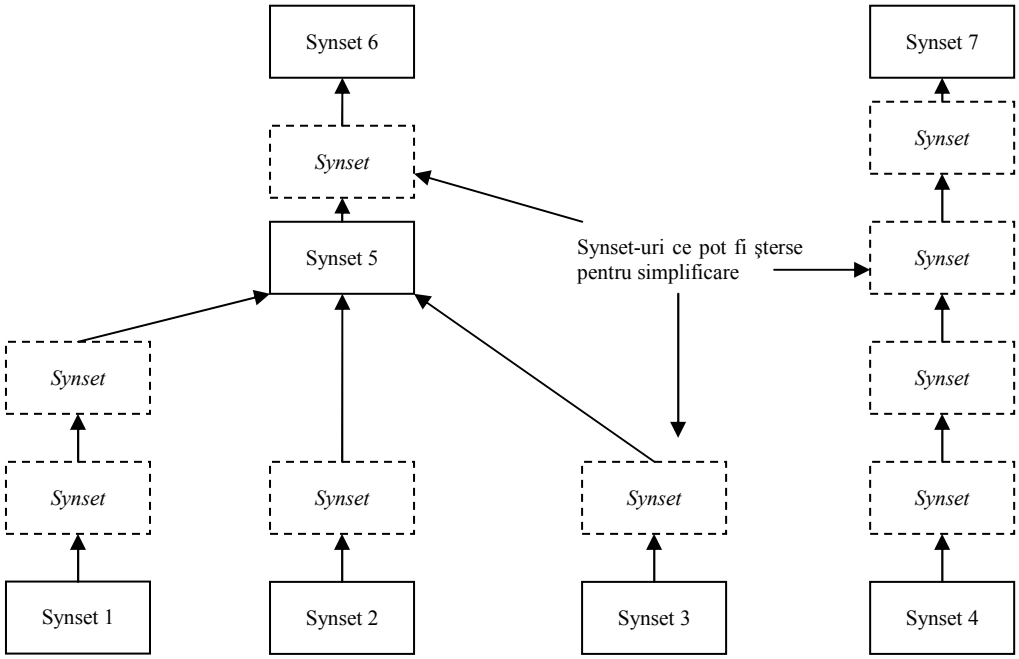

Fig. 3.3 Arbore de hipernime având marcate synset-urile ce pot fi şterse la simplificare.

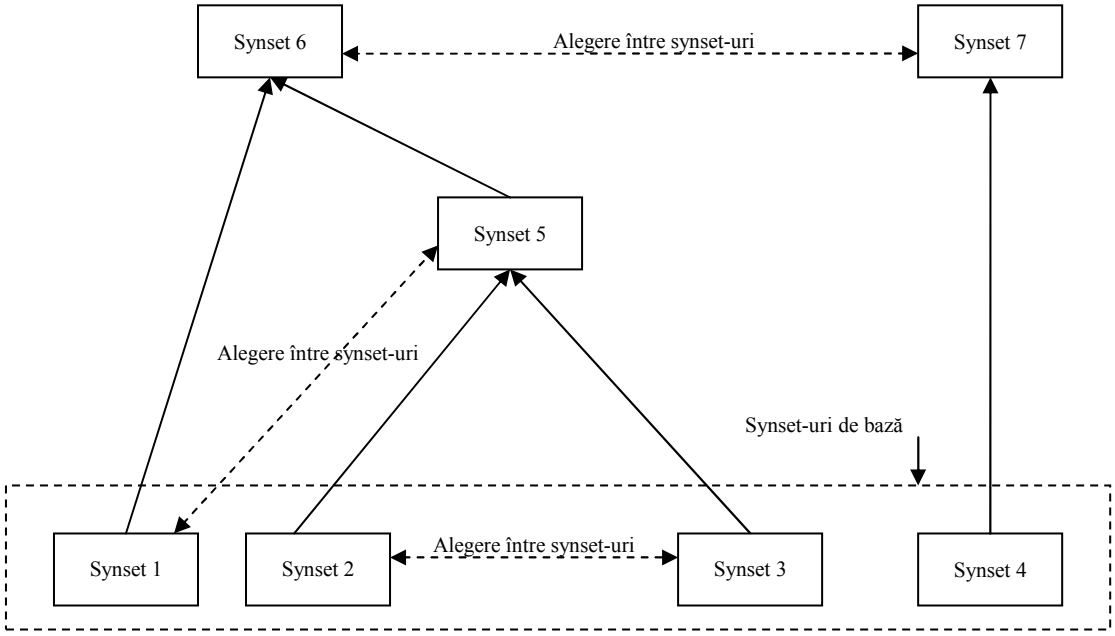

Fig. 3.4 Arbore simplificat de hipernime şi modul de selecţie al sensului pentru un cuvânt.

Pentru modelul implementat, structura de arbore simplificat de hipernime este foarte importantă, ea fiind folosită atât în modulul de învăţare, cât şi în cel de aplicare a dezambiguizării. Este, de asemenea, vital ca părţile de construcţie şi interogare ale arborelui să fie optimizate pentru viteză, deoarece în procesul de învăţare se creează câte o astfel de structură pentru fiecare cuvânt prelucrat din corpus.

După cum se observă din Fig. 3.4, alegerea synset-urilor se face mai întâi la nivelul superior al arborelui, apoi, în funcție de opțiunea făcută, se mai fac alte alegeri sau, cum se întâmplă în cazul synset-ului 4, dezambiguizarea este completă. Se reduce în acest mod, la fiecare coborâre a unui nivel, numărul synset-urilor posibile din care trebuie făcută alegerea finală.

3.3 Structura corpus-ului folosit pentru învățare

Pentru crearea unei tabele de co-ocurențe ale cuvintelor s-a propus o întreagă serie de metode. Acestea țin, în general, de viziunea pe care autorii au ales să o adopte asupra contextului local al cuvintelor şi variază de la considerarea unei „ferestre" de *n* cuvinte în jurul cuvântului prelucrat, până la luarea în considerare a unor factori sintactici sau a unor factori ce țin de o bază de cunoştințe.

În modelul implementat am optat pentru un corpus etichetat la nivel morfologic şi sintactic, reprezentând romanul „1984" de G. Orwell. Folosirea unui text etichetat se justifică prin faptul că, utilizând informațiile suplimentare furnizate de etichete, se reduce numărul datelor redundante, pe de o parte, şi pe alta se obțin unele indicii cu privire la sensul cuvintelor în text, chiar dacă etichetarea nu este făcută la nivel de sensuri ale cuvintelor.

În cele ce urmează este redat un fragment din textul etichetat folosit reprezentând prima frază a romanului.

```
<ROOT>
  <P>
    <S ID="S0">
      <NP ID="NP0" VERBPOS="W1" HEADID="W0">
        <W ID="W0" NUM="SG" LINK="W1" ROLE="SUBJ" POS="PRON" LEMMA="it"
LINKTYPE="subj">It</W>
      </NP>
      <NP ID="NP1" VERBPOS="W1" HEADID="W5">
        <W ID="W2" LINK="W5" ROLE="DN+" POS="DET" LEMMA="a" LINKTYPE="det">a</W>
        <W ID="W3" LINK="W4" ROLE="A+" POS="A" LEMMA="bright"
LINKTYPE="attr">bright</W>
        <W ID="W4" LINK="W5" ROLE="A+" POS="A" LEMMA="cold" LINKTYPE="attr">cold</W>
        <W ID="W5" NUM="SG" LINK="W1" ROLE="ADVL" POS="N" LEMMA="day"
LINKTYPE="tmp">day</W>
        <PP ID="PP0" VERBPOS="W1" HEADID="W6">
          <W ID="W6" LINK="W5" ROLE="ADVL" POS="PREP" LEMMA="in"
LINKTYPE="tmp">in</W>
          <NP ID="NP2" VERBPOS="W1" HEADID="W7">
            <W ID="W7" NUM="SG" LINK="W6" ROLE="-P" POS="N" LEMMA="April"
LINKTYPE="pcomp">April</W>
          </NP>
        </PP>
      </NP>
      <NP ID="NP3" VERBPOS="W12" HEADID="W11">
        <W ID="W10" LINK="W11" ROLE="DN+" POS="DET" LEMMA="the"
LINKTYPE="det">the</W>
        <W ID="W11" NUM="PL" LINK="W12" ROLE="SUBJ" POS="N" LEMMA="clock"
LINKTYPE="subj">clocks</W>
      </NP>
      <W ID="W1" LINK="w0" ROLE="+FMAINV" POS="V" LEMMA="be"
LINKTYPE="main">was</W>
      <W ID="W8" POS="PUNCT">,</W>
      <W ID="W9" LINK="W1" ROLE="CC" POS="CC" LEMMA="and" LINKTYPE="cc">and</W>
```

```
        <W ID="W12" LINK="W13" ROLE="+FAUXV" POS="V" LEMMA="be" LINKTYPE="v-
ch">were</W>
        <W ID="W13" LINK="W1" ROLE="-FMAINV" POS="ING" LEMMA="strike"
LINKTYPE="cc">striking</W>
        <W ID="W14" LINK="W13" ROLE="-OBJ" POS="NUM" LEMMA="thirteen"
LINKTYPE="obj">thirteen</W>
        <W ID="W15" POS="PUNCT">.</W>
      </S>
    </P>
</ROOT>
```

Fig. 3.5 Textul etichetat corespunzător primei propoziții a romanului „1984” de G. Orwell. - *It was a bright cold day in April, and the clocks were striking thirteen.*

Astfel, informația conținută în atributele „POS” (part of speech) din etichetele aplicate cuvintelor, a fost extrem de folositoare prin faptul că s-au prelucrat numai cuvintele care aparțineau părților de vorbire de interes (substantive şi verbe). De asemenea, datorită acestor atribute s-au luat în considerare pentru cuvintele în lucru numai acele sensuri care corespundeau părții de vorbire pe care o aveau în contextul respectiv. Datele astfel obținute au o acuratețe mult mărită şi probabilitățile ce se vor calcula în modulul de dezambiguizare vor fi mai aproape de valorile reale.

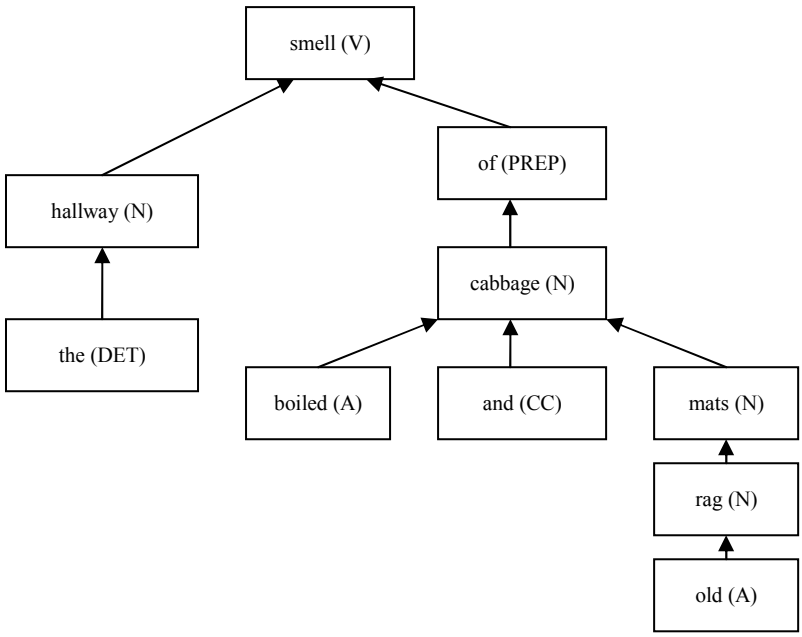

Fig. 3.6 Arborele sintactic al propoziției –*The hallway smelt of boiled cabbage and old rag mats.* -.

Al doilea atribut folosit a fost „LINK”, un atribut ce reprezintă legătura cuvântului către părintele său în arborele sintactic al propoziției. După cum am arătat anterior, diferiți autori au optat pentru considerarea a diferite contexte locale. În acest model am hotărât să profităm de informația despre structura sintactică a propozițiilor şi am considerat contextul local ca fiind constituit din fii şi părintele cuvântului în arborele sintactic. În acest mod se măreşte relevanța „apropierii” dintre cuvinte, evitându-se cazurile în care cuvinte apar alăturate în text dar fără legătură semantică între ele.

De asemenea, deşi în versiunea actuală nu s-a considerat necesară implementarea acestui lucru, ar fi interesant de experimentat varianta în care co-ocurenţele sensurilor în propoziţie sunt ponderate de tipul relaţiei sintactice dintre ele. Astfel, sensurilor unui cuvânt când acesta referă un subiect sau predicat li s-ar putea da o importanţă mai mare decât atunci când acesta referă un complement. Dezavantajul acestei metode ar fi că stabilirea unei scale a importanţei părţilor de propoziţie ar fi foarte dificilă, întotdeauna putându-se găsi contra-argumente pentru o configuraţie aleasă.

Spre deosebire de alte modele care se bazează în procesul de învăţare fie pe etichetarea manuală la nivel de sensuri ale cuvintelor a textului, fie nu folosesc decât textul curat sau cu puţine etichetări, modelul propus alege o cale de mijloc. Astfel, fără a se baza pe o etichetare manuală a sensurilor cuvintelor care predispusă la erori sau discutabilă în unele cazuri, modelul face totuşi uz de o etichetare care, în urma ultimelor reuşite din domeniu, poate fi obţinută automat. Faptul că orice text poate fi etichetat automat este deosebit de important, deoarece se poate construi în acest mod facil şi mai sigur din punct de vedere al erorilor umane un corpus de învăţare suficient de mare.

3.4 Structura şi construcţia arborelui sintactic.

Ca şi pentru celelalte două tipuri de arbori folosite în modelul implementat, şi pentru arborele sintactic este vitală optimizarea pentru viteză, pe structura sa realizându-se calcularea scorului maxim ce conduce la dezambiguizare. Ca urmare, am optat pentru stocarea în cadrul nodurilor sale a cât mai multor informaţii despre cuvântul situat acolo. Pentru a face uşoară mişcarea în arbore, atât de sus în jos cât şi de jos în sus, în fiecare nod există o referinţă către părintele sintactic dar şi o listă cu referinţele către fiii săi. Conţinutul propriu-zis al nodului constă dintr-un obiect de tip W (word) care înglobează toate datele oferite de etichetele XML şi sensul (eventual temporar) care a fost ales pentru cuvântul respectiv. Deoarece sunt folosite foarte des şi ar fi redundant să fie construite de fiecare dată, în nodul arborelui sintactic se memorează şi arborele simplificat de hipernime al cuvântului împreună cu lista synset-urilor ce apar în acesta pe nivelul ierarhic superior. Arborele propriu-zis este format din lista completă de cuvinte care intră în sfera de interes a modelului (substantive şi verbe) şi dintr-o rădăcină, care este un nod fictiv (o propoziţie poate avea unul, mai multe sau nici un cuvânt central).

După cum am menţionat, am preferat, din motive de viteză, să nu se construiască arborele sintactic la prelucrarea propoziţiilor în timpul procesului de învăţare. În cadrul procesului de dezambiguizare însă, construirea sa este esenţială pentru crearea unei imagini corecte asupra cuvintelor din propoziţie şi a relaţiilor dintre ele.

Algoritmul de construcţie al arborelui sintactic pentru o propoziţie primeşte la intrare lista de cuvinte ale acesteia, cu etichetările aferente. Crearea arborelui are loc în două etape:

- crearea arborelui sintactic general, care conţine toate cuvintele propoziţiei, indiferent de partea de vorbire;
- simplificarea acestuia, păstrând numai substantivele şi verbele.

Cele două etape sunt detaliate în continuare:

```
ConstruiesteArboreSintactic(ListaCuvinteIntrare)
{
      Root = NodNouFictiv;
      // ListaCuvinte contine chiar noduri ale arborelui
```

```
      ListaCuvinte.AdaugaToateCuvintele(ListaCuvinteIntrare);
      Pentru fiecare Cuvant din ListaCuvinte
      {
            if (Cuvant.AreParinte)
            {
                  Parinte = GasesteParinte(Cuvant);
                  Parinte.AdaugaFiu(Cuvant);
                  Cuvant.Parinte = Parinte;
            }
            else
            {
                  Root.AdaugaFiu(Cuvant);
                  Cuvant.Parinte = Root;
            }
      }
}
```

Simplificarea arborelui constă din eliminarea acelor noduri care conţin cuvinte ce nu sunt verbe sau substantive. Ştergerea unui nod trebuie să se facă păstrându-se structura ierarhică a construcţiei. Acest lucru presupune:

- ştergerea sa din lista de fii a părintelui;
- toţi fiii săi trebuie adăugaţi la fiii părintelui;
- părintele fiilor săi devine părintele său actual;
- nodul trebuie şters din lista de noduri.

Efectuarea în practică a simplificării se face parcurgând lista de noduri (în care apar toate nodurile nu numai frunzele), ştergându-se cele care reprezintă cuvinte care nu intră în domeniul de interes şi construindu-se arborii simplificaţi de hipernime şi listele de synset-uri de vârf pentru celelalte.

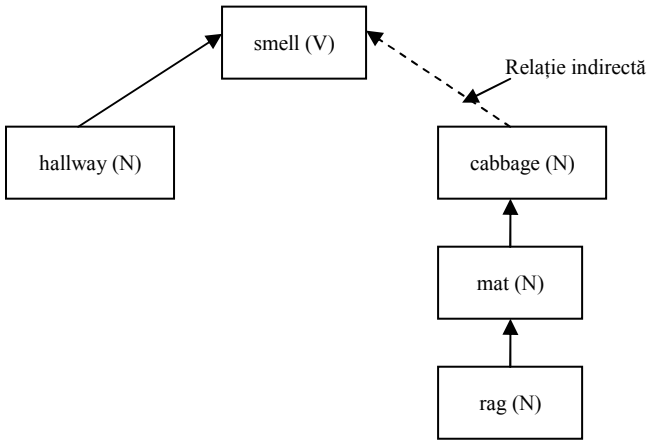

Fig. 3.7 Arborele sintactic simplificat al propoziţiei *The hallway smelt of boiled cabbage and old rag mats.*

Pe arborele rezultat în urma acestor prelucrări se va încerca maximizarea unei funcţii de scor a probabilităţii sensurilor cuvintelor de a se afla împreună în acelaşi context. În fapt, aceasta înseamnă maximizarea scorurilor obţinute de fiecare părinte cu fii săi. Problema de la acest punct înainte se reduce la a găsi, prin experimente, un mod de calcul al probabilităţilor şi

al scorurilor care să apropie cât mai mult atribuirile de sensuri pentru care optează programul de cele pe care le-ar alege un cititor uman. Pentru modelul implementat vom atribui pentru un cuvânt numai un sens din vârful ierarhiei de hipernime.

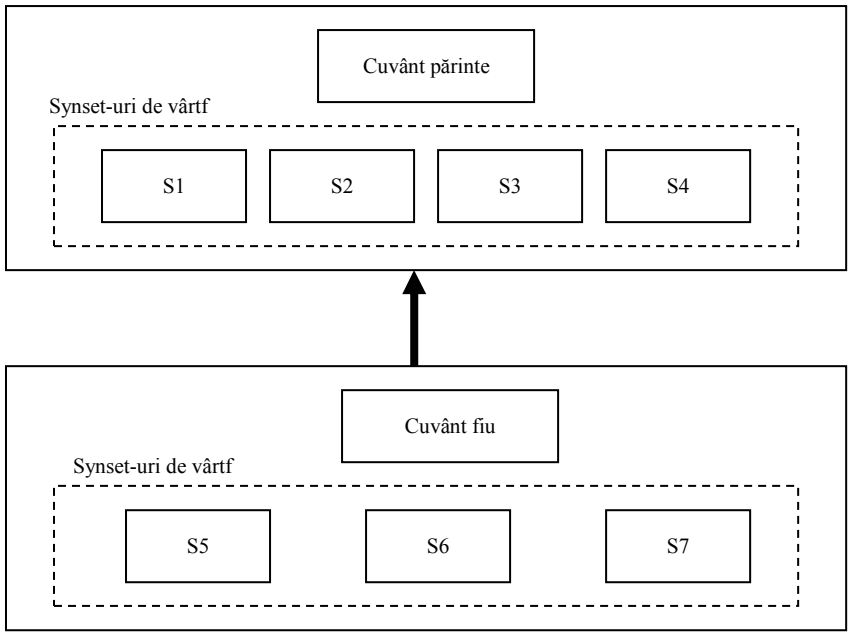

Fig. 3.8 Nodurile arborelui sintactic simplificat conţin informaţii referitoare la structura arborelui de hipernime al cuvântului. Dezambiguizarea se realizează la nivelul synset-urilor de vârf.

Pornind de la acesta se va efectua o restrângere a mulţimii de sensuri posibile pentru cuvântul respectiv, uneori chiar până la un singur element. În cazul în care rămân în continuare mai multe sensuri între care se poate opta, dezambiguizarea va continua cu synset-urile ce apar pe următorul nivel în arborii de hipernime. De asemenea, metoda implementată în model poate fi folosită în paralel cu alte căi de dezambiguizare pentru a se obţine o precizie mărită. În cadrul modelului implementat m-am limitat la restrângerea mulţimii de sensuri posibile prin dezambiguizarea la nivelul synset-urilor de vârf. Scopul urmărit a fost testarea ipotezei pe care se bazează şi apoi găsirea formulelor prin care se măreşte exactitatea răspunsurilor. Din acest motiv, în cele ce urmează se va considera dezambiguizare corectă o dezambiguizare în care sensul de vârf ales automat este hipernim pentru sensul ales de un cititor uman.

3.5 Structura tabelei de co-ocurenţe

Având în vedere necesitatea unei accesări rapide a câmpurilor tabelei, am optat pentru reprezentarea clasică sub formă de matrice bidimensională, deşi matricea este rară. Deşi în WordNet există câteva zeci de mii de synset-uri, în urma experimentelor efectuate am constatat că numărul synset-urilor care apar pe nivelul superior al ierarhiilor de hipernime este acceptabil de mic, în jur de 500 (477 de synset-uri găsite în corpusul de învăţare folosit). Un alt avantaj al reprezentării sub formă de matrice decurge din faptul că în mod constant în timpul învăţării se pot găsi synset-uri noi care trebuiesc adăugate la tabelă, redimensionarea ei făcându-se mult mai rapid în acest fel.

Prin tabela de co-ocurențe este compusă de fapt dintr-o listă cu definițiile synset-urilor, două matrice de co-ocurențe şi doi vectori cu numărul total de apariții ale fiecărui synset. După cum am arătat, singurele synset-uri luate în considerare sunt synset-urile care apar pe nivelul ierarhic cel mai înalt în arborele de hipernime ce are la bază sensurile unui cuvânt. În cele ce urmează prin „*synset-ul i*" voi referi synset-ul care apare în lista de synset-uri a tabelei la poziția i.

Tabela de ocurențe prin referințe directe (A_{nxn}) are ca elemente A[i, j] numărul de apariții ale synset-ului *i* ca synset de vârf în arborele de hipernime al unui cuvânt ce referă sintactic direct un cuvânt ce are între synset-urile sale de vârf synset-ul *j*. Ambele cuvinte trebuie să aparțină ariei de interes, adică să fie substantive sau verbe.

Tabela de ocurențe prin referințe indirecte (B_{nxn}) este definită asemănător cu prima tabelă, cu deosebirea că primul cuvânt referă sintactic indirect pe al doilea.

Prin referirea indirectă se înțelege că există un şir de cuvinte $x_1, x_2, ..., x_n$, $n \geq 3$, în care:

- primul şi ultimul cuvânt sunt substantive sau verbe;
- toate celelalte cuvinte ($x_2, ..., x_n$) sunt prepoziții, articole sau conjuncții;
- x_i îl referă sintactic pe x_{i+1}, i=1,n-1.

Se observă că $n \geq 3$, deci mulțimile aparițiilor numărate pentru acelaşi synset în prima şi a doua tabelă sunt disjuncte.

Vectorii cu numărul total de apariții ale synset-urilor conțin, separat pentru referințe directe şi indirecte, numărul total de apariții ale fiecărui synset ca synset de vârf în arborele de hipernime al unui cuvânt ce apare în partea stângă a unei relații sintactice.

3.6 Construcția tabelei de co-ocurență

Am efectuat un experiment pentru a verifica dacă cuvintele care referă sintactic un alt cuvânt, atunci când acesta are un anumit sens, au synset-uri de vârf comune. Am considerat verbul *pay* cu sensul de *<give, pay -- (convey, as of a compliment, regards, attention, etc.; bestow) >* şi cuvintele *attention, thanks, regards, devotions, respect, heed*. Am constatat că aceste cuvinte au un synset de vârf care apare la toate - *<abstraction -- (a general concept formed by extracting common features from specific examples)>*. Acesta poate fi considerat un criteriu pentru a dezambiguiza un alt cuvânt care ar apărea referind verbul *pay* cu acest sens. De asemenea, putem avea şi situația inversă, în care alegem un sens pentru *pay* în funcție de cuvintele care îl referă.

Pentru dezambiguizarea sensurilor cuvintelor trebuie căutate indicii cu în contextul local al cuvintelor. După cum am arătat în experimentul anterior, un indiciu puternic este furnizat de sensurile de vârf ale cuvintelor ce referă sintactic cuvântul prelucrat. Cuvintele care îl referă atunci când acesta are un anumit sens au în general unul sau mai multe synset-uri de vârf comune. Dacă se contorizează numărul de apariții ale acestor synset-uri, în mod normal, cele comune vor apare de mai multe ori şi deci vor fi candidate favorite în cadrul dezambiguizării.

Procesul de învățare al modelului se transpune în parcurgerea fişierelor XML care conțin corpusul etichetat, din fiecare din acestea extrăgându-se pe rând câte o propoziție ale cărei cuvinte sunt prelucrate. Având în vedere că ne interesează relațiile sintactice la nivel de

propoziție, aceasta va fi unitatea de bază pe care se vor realiza atât învățarea cât şi dezambiguizarea.

Pentru modulul de învățare am considerat că o construire a arborelui sintactic al propoziției nu ar aduce un spor de viteză ci ar îngreuna chiar procesul de prelucrare a cuvintelor. Am optat deci pentru un algoritm care primeşte la intrare lista de cuvinte etichetate şi, pe baza acesteia, actualizează datele din tabelă:

```
ActualizareTabela(ListaCuvinte)
{
      pentru fiecare cuvant din ListaCuvinte
      {
            if (cuvant.POS == subst SAU cuvant.POS == verb)
            {
            // gasesc primul parinte subst. sau
            parinte = GasesteParinte(cuvant)
            direct = EsteParinteDirect(cuvant, parinte)
            if (parinte != null)
            {
                  if (direct)
                        ActualizeazaTabelaDirect(cuvant, parinte)
                  Else
                        ActualizeazaTabelaIndirect(cuvant, parinte)
            }
            }
      }
}
```

Actualizarea tabelelor cu referințe directe sau indirecte se face în felul următor:

```
ActualizeazaTabelaDirect(cuvant, parinte)
{
      hTreeCuvant = HypernimsTree(cuvant);
      hSTreeCuvant = SimplifiedTree(hTreeCuvant);
      ListaCuvant = hSTreeCuvant.SensuriDePeNivel(1); // synseturi de varf

      hTreeParinte = HypernimsTree(parinte);
      hSTreeParinte = SimplifiedTree(hTreeParinte);
      ListaParinte = hSTreeParinte.SensuriDePeNivel(1); //synseturi de varf

      Pentru fiecare SynsetFiu din ListaCuvant
      {
            PosFiu = GasesteSynsetInLista(SynsetFiu);
            If (PosFiu == -1)
                  Adauga SynsetFiu la Lista
            NumarAparitiiDirecte[PosFiu]++;
            Pentru fiecare SynsetParinte din ListaParinte
            {
                  PosParinte = GasesteSynsetInLista(SynsetFiu);
                  If (PosParinte == -1)
                        Adauga SynsetParinte la Lista
                  TabelaDirect[PosFiu][PosParinte] ++;
            }
      }
}
```

După prelucrarea fişierelor XML reprezentând romanul „1984” de George Orwell lista cu synset-urile de vârf găsite conținea 477 de synset-uri. Synset-ul <*be -- (have the quality of being; (copula, used with an adjective or a predicate noun))*>, de exemplu, apare de 1245 ori. O parte din linia din tabela de co-ocurențe prin referințe directe este redată mai jos (în paranteze sunt trecute pozițiile din listă ale synset-urilor referite):

(0)387 |(1)54 |(2)16 |(3)22 |(4)635 |(5)206 |(6)189 |(7)174 |(8)33 |(9)92 |(10)192 |(11)25 |(12)2 |(13)57 |(14)156 |(15)236 |(16)179 |(17)211 |(18)169 |(19)172 |(20)177 |(21)156 |(22)128 |(23)3 |(24)68 |(25)120 |(26)96 |(27)2 |(28)188 |(29)224 |(30)23 |(31)0 |(32)63 |(33)0 |(34)265 |(35)70 |(36)5 |(37)0 |(38)14 |(39)15 |(40)15 |(41)99 |(42)0 |(43)4 |(44)36 |(45)57 |(46)44 |(47)11 |(48)2 |(49)67 |(50)134 |(51)45 |(52)22 |(53)22 |(54)68 |(55)46 |(56)25 |(57)21 |(58)70 |(59)49 |(60)97 |(61)21 |(62)148 |(63)7 |(64)39 |(65)2 |(66)2 |(67)19 |(68)16 |(69)35 |(70)41 |(71)29 |(72)15 |(73)10 |(74)8 |(75)17 |(76)8 |(77)8 |(78)185 |(79)64 |(80)36 |(81)0 |(82)9 |(83)24 |(84)36 |(85)44 |(86)91 |(87)0 |(88)25 |(89)37 |(90)32 |(91)25 |(92)25 |(93)29 |(94)29 |(95)25 |(96)25 |(97)31 |(98)25 |(99)1 |(100)2

Fig. 3.9 O parte din tabela de co-ocurențe prin referințe directe

După cum se observă, există câteva synset-uri „preferate” care sunt referite de un număr de ori mult mai mare decât restul. De exemplu synset-ul numărul 4 - <*act, move -- perform an action*> - este referit de 635 de ori, adică în jumătate din apariții, ceea ce denotă o legătură strânsă între aparițiile lor, spre deosebire de alte synset-uri care sunt referite o singură dată sau deloc.

Datorită modului de construire, se poate spune că un cuvânt care are un sens hiponim pentru <*be -- (have the quality of being)*>, există o probabilitate destul de mare ca el să refere sintactic un cuvânt care are un sens hiponim pentru <*act, move -- perform an action*>. În fapt, dezambiguizarea va avea loc în sens invers, adică vom alege un sens pentru cuvântul subordonat în funcție de sensul părintelui său.

Construcția şi actualizarea tabelelor de co-ocurențe constituie practic procesul de învățare al modelului. În principiu, cu cât cantitatea de text prelucrat este mai mare, cu atât probabilitățile ce vor fi calculate în modulul de dezambiguizare vor fi mai apropiate de realitate. De asemenea, calitatea etichetării corpusului are un impact puternic asupra procesului de învățare şi, prin urmare, şi asupra acurateței dezambiguizării ulterioare.

Capitolul 4
Arhitectura modulului de dezambiguizare

Datele adunate în modulul de învățare al modelului implementat vor trebui exploatate la maxim în cadrul celui de-al doilea modul cel de dezambiguizare. Acesta primeşte la intrare o propoziție etichetată similar cu cele din textul folosit pentru învățare şi determină pentru fiecare cuvânt, pe baza informațiilor conținute în tabelele de co-ocurențe, sensul de vârf în arborele de hipernime care este părinte pentru sensul cu care respectivul cuvânt apare în context.

Necesitatea ca textul introdus pentru dezambiguizare să fie etichetat implică într-adevăr o prelucrare preliminară a acestuia, dar am optat pentru menținerea acestei cerințe din două motive:

- corpusul folosit pentru învățare a avut aceeaşi structură, deci rezultatele vor fi maxime dacă aceasta va fi păstrată şi vor fi folosite informațiile suplimentare despre structura propoziției conținute în etichete;
- în ultimii ani au fost dezvoltate un număr de etichetatoare morfologice şi sintactice automate cu precizii destul de ridicate care reduc efortul suplimentar care trebuia depus prin etichetarea manuală.

Ideea de bază folosită pentru găsirea sensurilor de vârf ce sunt atribuite cuvintelor este maximizarea unei funcții de scor pe arborele sintactic al propoziției. Din această cauză, cel mai important element al modului este modul de calcul al scorului unei configurații de perechi sens – cuvânt.

După extragerea din text a cuvintelor unei propoziții, se construieşte arborele sintactic (descris pe larg în secțiunea 3.4).

Găsirea configurației de perechi sens – cuvânt care are scor maxim se face alegând-o pe aceea care are în nodul părinte sensul care a obținut cel mai mare scor, calculat recursiv pe tot arborele. Modul de calcul recursiv al scorului este următorul:

```
CalculeazaScorMaxim(EsteNodFaraParinte)
{
      MaxScor = 0;
      SynsetMaxim = null;
      Pentru fiecare Synset din ListaSynseturiDeVarf
      {
            Scor = 0;
            SensAles = Synset;
            if (!EsteNodFaraParinte)
            {
                  Scor = CalculeazaScor(Synset, Parinte.SensAles);
            }
            Pentru fiecare Fiu din ListaFii
            {
                  Scor += Fiu.CalculeazaScorMaxim(false);
            }
            if (Scor > MaxScor)
            {
                  MaxScor = Scor;
                  SynsetMaxim = Synset;
```

```
            }
        }
        SynsetAles = SynsetMaxim;
        return MaxScor;
}
```

Apelarea inițială a metodei de calcul a scorului şi a configurației maxime se face pentru toți fii rădăcinii fictive, parametrul *EsteNodFaraParinte = true*. De remarcat este că, la final, se obțin două rezultate: alegerea, pentru fiecare cuvânt, a synset-ului optim precum şi scorul obținut de această configurație.

Dacă se consideră necesar, acest mod de calcul poate fi rafinat, prin atribuirea de priorități pentru cuvinte, în funcție de nivelul pe care se află în arbore sau de rolul sintactic pe care îl dețin. Locul în care optimizările trebuie obligatoriu făcute, însă, este la modul în care se calculează probabilitatea ca un synset să refere sintactic un altul. Pe lângă acestea se mai pot, desigur testa moduri de calculare ale scorului total în care se dă o importanță mai mare scorurilor obținute de cuvinte cu mai mulți fii, sau care dețin roluri sintactice importante, cum ar fi acelea de subiect sau predicat. În cele ce urmează însă, scopul urmărit a fost găsirea şi compararea unor metode de evaluare a probabilității ca două sensuri să fie alese ca sensuri de dezambiguizare pentru două cuvinte ce sunt în relație sintactică.

4.1 Funcții de scor pentru compatibilitatea a două sensuri

Prin funcție de scor a compatibilității a două sensuri vom înțelege o funcție care pentru oricare două synset-uri, folosind cuvintele pentru care se face dezambiguizarea şi datele din tabelele de co-ocurențe, returnează un număr real (de preferat, dar nu obligatoriu, în intervalul [0, 1]). Acest număr reprezintă posibilitatea ca, în cazul în care se aleg synset-urile respective ca sensuri pentru cuvintele prelucrate, dezambiguizarea să fie una corectă.

De notat este faptul că acuratețea unei astfel de funcții este puternic influențată de cantitatea de informații extrasă din corpusul folosit pentru învățare. De asemenea, pentru anumite cazuri o dezambiguizare folosind numai datele furnizate de modelul pe care îl propunem este imposibilă, ideală fiind combinarea mai multor modele într-un mod în care părțile mai slabe ale unuia din ele să fie tot timpul acoperite de către un altul. Un caz în care dezambiguizarea este dificilă, dacă nu irealizabilă, folosind numai datele furnizate de acest model, este cel al cuvintelor foarte utilizate ale limbii engleze, cum ar fi de exemplu verbul *to be*. Acesta apare atât de sine stătător, cât şi ca particulă pentru formarea anumitor timpuri verbale. Această abundență a sa în text, mai ales cu al doilea rol, duce la o uniformizare a datelor despre co-ocurențe, pe de o parte, şi la un număr foarte mare de apariții ale sensului său de particulă, pe de alta. Acest sens va deveni, deci, principala opțiune în cazul unei dezambiguizări, excluzând din start celelalte opțiuni. Se observă în acest caz necesitatea combinării modului de calcul al probabilității bazat numai pe datele obținute din modulul de învățare cu un altul care să folosească date furnizate de un alt model.

Pentru testarea funcțiilor de scor propuse am ales câteva exemple care ni s-au părut reprezentative, testul final constituindu-se din dezambiguizarea primelor paragrafe ale romanului „1984” de G. Orwell.

4.2 Perechile de cuvinte folosite pentru testarea funcțiilor

Pentru testările inițiale ne-am axat pe studierea alegerilor făcute de funcțiile de scor pentru cuvintele ce apar referind verbul *pay* atunci când acesta apare cu anumite sensuri.

The noun pay has 1 sense (first 1 from tagged texts)
1. wage, pay, earnings, remuneration, salary -- (something that remunerates; "wages were paid by check"; "he wasted his pay on drink"; "they saved a quarter of all their earnings")
The verb pay has 11 senses (first 11 from tagged texts)
1. pay -- (give money in exchange for goods or services; "I paid four dollars for this sandwich"; "Pay the waitress, please")
2. give, pay -- (convey, as of a compliment, regards, attention, etc.; bestow; "Don't pay him any mind"; "give the orders"; "Give him my best regards"; "pay attention")
3. pay up, ante up, pay -- (cancel or discharge a debt; "pay up, please!")
4. pay, pay off, make up, compensate -- (do or give something to somebody in return; "Does she pay you for the work you are doing?")
5. pay -- (render; "pay a visit"; "pay a call")
6. pay -- (bear (a cost or penalty), in recompense for some action; "You'll pay for this!"; "She had to pay the penalty for speaking out rashly"; "You'll pay for this opinion later")
7. yield, pay, bear -- (bring in; as of investments; "interest-bearing accounts"; "How much does this savings certificate pay annually?")
8. pay -- (be worth it; "It pays to go through the trouble")
9. give, pay, devote -- (as in the expressions "give thought to"; "give priority to", etc.)
10. pay -- (discharge or settle; "pay a debt"; "pay an obligation")
11. pay -- (make a compensation for; "a favor that cannot be paid back")

Fig. 4.1 Sensurile cuvântului *pay*

Deşi ca verb *pay* are 11 sensuri, numărul synset-urilor care apar pe nivelul superior al arborelui de hipernime este de 8:

Sense 1
pay -- (give money in exchange for goods or services; "I paid four dollars for this sandwich"; "Pay the waitress, please")
=> give -- (transfer possession of something concrete or abstract to somebody; "I gave her my money"; "can you give me lessons?" "She gave the children lots of love and tender loving care")
=> transfer -- (cause to change ownership; "I transferred my stock holdings to my children")
Sense 2
give, pay -- (convey, as of a compliment, regards, attention, etc.; bestow; "Don't pay him any mind"; "give the orders"; "Give him my best regards"; "pay attention")
=> communicate, intercommunicate -- (transmit thoughts or feelings; "He communicated his anxieties to the psychiatrist")
=> interact -- (act together or towards others or with others; "He should interact more with his colleagues")
=> act, move -- (perform an action; "think before you act"; "We must move quickly")
Sense 3
pay up, ante up, pay -- (cancel or discharge a debt; "pay up, please!")
=> pay -- (discharge or settle; "pay a debt"; "pay an obligation")
=> settle -- (dispose of; make a financial settlement)
=> arrange, fix up -- (make arrangements for; "Can you arrange a meeting with the President?")
=> agree, concur -- (be in accord; be in agreement; "We agreed on the terms of the settlement"; "I can't agree with you!")
Sense 4
pay, pay off, make up, compensate -- (do or give something to somebody in return; "Does she pay you for the work you are doing?")
=> settle -- (dispose of; make a financial settlement)
=> arrange, fix up -- (make arrangements for; "Can you arrange a meeting with the President?")
=> agree, concur -- (be in accord; be in agreement; "We agreed on the terms of the settlement"; "I can't agree with you!")
Sense 5
pay -- (render; "pay a visit"; "pay a call")
=> make -- (perform or carry out; "make a decision"; "make a move"; "make advances"; "make a phone call")
=> perform, execute, do -- (to act or perform an action; "John did the painting, the weeding, and he cleaned out the gutters")

Sense 6
pay -- (bear (a cost or penalty), in recompense for some action; "You'll pay for this!"; "She had to pay the penalty for speaking out rashly"; "You'll pay for this opinion later")
=> endure, stomach, bear, stand, tolerate, brook, abide, suffer, put up -- (put up with something or somebody unpleasant; "I cannot bear his constant criticism"; "The new secretary had to endure a lot of unprofessional remarks")
=> permit, allow, let, countenance -- (give permission; "She permitted her son to visit her estranged husband"; "I won't let the police search her basement"; "I cannot allow you to see your exam")
Sense 7
yield, pay, bear -- (bring in; as of investments; "interest-bearing accounts"; "How much does this savings certificate pay annually?")
=> gain, take in, clear, make, earn, realize, pull in, bring in -- (earn on some commercial or business transaction; earn as salary or wages; "How much do you make a month in your new job?" "She earns a lot in her new job"; "this merger brought in lots of money"; "He clears $5,000 each month")
=> get, acquire -- (come into the possession of something concrete or abstract; "She got a lot of paintings from her uncle"; "They acquired a new pet"; "Get your results the next day"; "Get permission to take a few days off from work")
Sense 8
pay -- (be worth it; "It pays to go through the trouble")
=> be -- (have the quality of being; (copula, used with an adjective or a predicate noun); "John is rich"; "This is not a good answer")
Sense 9
give, pay, devote -- (as in the expressions "give thought to"; "give priority to", etc.)
=> think, cogitate, cerebrate -- (use or exercise the mind or one's power of reason in order to make inferences, decisions, or arrive at a solution or judgments; "I've been thinking all day and getting nowhere")
Sense 10
pay -- (discharge or settle; "pay a debt"; "pay an obligation")
=> settle -- (dispose of; make a financial settlement)
=> arrange, fix up -- (make arrangements for; "Can you arrange a meeting with the President?")
=> agree, concur -- (be in accord; be in agreement; "We agreed on the terms of the settlement"; "I can't agree with you!")
Sense 11
pay -- (make a compensation for; "a favor that cannot be paid back")
=> requite, repay -- (make repayment for or return something)
=> give -- (transfer possession of something concrete or abstract to somebody; "I gave her my money"; "can you give me lessons?" "She gave the children lots of love and tender loving care")
=> transfer -- (cause to change ownership; "I transferred my stock holdings to my children")

Fig. 4.2 Arborele de hipernime pentru verbul *pay*

Arborele de hipernime prezentat mai sus poate fi simplificat, obținându-se următorul arbore simplificat de hipernime:

pay -- (give money in exchange for goods or services; "I paid four dollars for this sandwich"; "Pay the waitress, please")
=> give -- (transfer possession of something concrete or abstract to somebody; "I gave her my money"; "can you give me lessons?" "She gave the children lots of love and tender loving care")
=> transfer -- (cause to change ownership; "I transferred my stock holdings to my children")
---
give, pay -- (convey, as of a compliment, regards, attention, etc.; bestow; "Don't pay him any mind"; "give the orders"; "Give him my best regards"; "pay attention")
=> act, move -- (perform an action; "think before you act"; "We must move quickly")
---
pay up, ante up, pay -- (cancel or discharge a debt; "pay up, please!")
=> settle -- (dispose of; make a financial settlement)
=> agree, concur -- (be in accord; be in agreement; "We agreed on the terms of the settlement"; "I can't agree with you!")
---
pay, pay off, make up, compensate -- (do or give something to somebody in return; "Does she pay you for the work you are doing?")

```
    => settle -- (dispose of; make a financial settlement)
        => agree, concur -- (be in accord; be in agreement; "We agreed on the terms of the settlement"; "I can't agree with you!")
---------------------------------------------------------------------------------------------------------------------------------------------
pay -- (render; "pay a visit"; "pay a call")
    => perform, execute, do -- (to act or perform an action; "John did the painting, the weeding, and he cleaned out the gutters")
---------------------------------------------------------------------------------------------------------------------------------------------
pay -- (bear (a cost or penalty), in recompense for some action; "You'll pay for this!"; "She had to pay the penalty for speaking out rashly"; "You'll pay for this opinion later")
    => permit, allow, let, countenance -- (give permission; "She permitted her son to visit her estranged husband"; "I won't let the police search her basement"; "I cannot allow you to see your exam")
---------------------------------------------------------------------------------------------------------------------------------------------
yield, pay, bear -- (bring in; as of investments; "interest-bearing accounts"; "How much does this savings certificate pay annually?")
    => get, acquire -- (come into the possession of something concrete or abstract; "She got a lot of paintings from her uncle"; "They acquired a new pet"; "Get your results the next day"; "Get permission to take a few days off from work")
---------------------------------------------------------------------------------------------------------------------------------------------
pay -- (be worth it; "It pays to go through the trouble")
    => be -- (have the quality of being; (copula, used with an adjective or a predicate noun); "John is rich"; "This is not a good answer")
---------------------------------------------------------------------------------------------------------------------------------------------
give, pay, devote -- (as in the expressions "give thought to"; "give priority to", etc.)
    => think, cogitate, cerebrate -- (use or exercise the mind or one's power of reason in order to make inferences, decisions, or arrive at a solution or judgments; "I've been thinking all day and getting nowhere")
---------------------------------------------------------------------------------------------------------------------------------------------
pay -- (discharge or settle; "pay a debt"; "pay an obligation")
    => settle -- (dispose of; make a financial settlement)
        => agree, concur -- (be in accord; be in agreement; "We agreed on the terms of the settlement"; "I can't agree with you!")
---------------------------------------------------------------------------------------------------------------------------------------------
pay -- (make a compensation for; "a favor that cannot be paid back")
    => give -- (transfer possession of something concrete or abstract to somebody; "I gave her my money"; "can you give me lessons?" "She gave the children lots of love and tender loving care")
        => transfer -- (cause to change ownership; "I transferred my stock holdings to my children")
```

Fig. 4.3 Arborele simplificat de hipernime pentru verbul *pay*

Din acest arbore simplificat se pot obține uşor synset-urile de vârf, cele de confluență sau cele frunză care au un părinte comun. Astfel, pentru verbul *pay* synset-urile care apar pe nivelul superior sunt următoarele:

```
0 - transfer -- (cause to change ownership; "I transferred my stock holdings to my children")
1 - act, move -- (perform an action; "think before you act"; "We must move quickly")
2 - agree, concur -- (be in accord; be in agreement; "We agreed on the terms of the settlement"; "I can't agree with you!")
3 - perform, execute, do -- (to act or perform an action; "John did the painting, the weeding, and he cleaned out the gutters")
4 - permit, allow, let, countenance -- (give permission; "She permitted her son to visit her estranged husband"; "I won't let the police search her basement"; "I cannot allow you to see your exam")
5 - get, acquire -- (come into the possession of something concrete or abstract; "She got a lot of paintings from her uncle"; "They acquired a new pet"; "Get your results the next day"; "Get permission to take a few days off from work")
6 - be -- (have the quality of being; (copula, used with an adjective or a predicate noun); "John is rich"; "This is not a good answer")
7 - think, cogitate, cerebrate -- (use or exercise the mind or one's power of reason in order to make inferences, decisions, or arrive at a solution or judgments; "I've been thinking all day and getting nowhere")
```

Fig. 4.4 Synset-urile de vârf pentru verbul *pay*

Cuvintele pentru care s-a făcut testarea au fost alese astfel încât apropierea semantică de verbul *pay* să fie un indiciu destul de puternic pentru stabilirea sensului corect. Testarea a urmărit capacitatea funcției de scor de a discerne între sensurile acestor cuvinte într-un mod în care rezultatul să fie cât mai apropiat de alegerea umană pentru acel caz.

Primul cuvânt pentru care am optat a fost substantivul *attention*. Sensurile acestuia, arborele simplificat de hipernime şi synset-urile de vârf sunt următoarele:

```
The noun attention has 6 senses (first 6 from tagged texts)

1. attention, attending -- (the process whereby a person concentrates on some features of the environment to the (relative) exclusion of others)
2. care, attention, aid, tending -- (the work of caring for or attending to someone or something; "no medical care was required"; "the old car needed constant attention")
3. attention -- (a general interest that leads people to want to know more; "She was the center of attention")
4. attention -- (a courteous act indicating affection; "she tried to win his heart with her many attentions")
5. attention -- (the faculty or power of mental concentration; "keeping track of all the details requires your complete attention")
6. attention -- (a motionless erect stance with arms at the sides and feet together; assumed by military personnel during drill or review; "the troops stood at attention")
```

Fig. 4.5 Sensurile cuvântului *attention*

```
attention, attending -- (the process whereby a person concentrates on some features of the environment to the (relative) exclusion of others)
    => cognition, knowledge -- (the psychological result of perception and learning and reasoning)
        => psychological feature -- (a feature of the mental life of a living organism)
---------------------------------------------------------------------------------------------------------------------------------------
care, attention, aid, tending -- (the work of caring for or attending to someone or something; "no medical care was required"; "the old car needed constant attention")
    => act, human action, human activity -- (something that people do or cause to happen)
---------------------------------------------------------------------------------------------------------------------------------------
attention -- (a general interest that leads people to want to know more; "She was the center of attention")
    => cognition, knowledge -- (the psychological result of perception and learning and reasoning)
        => psychological feature -- (a feature of the mental life of a living organism)
---------------------------------------------------------------------------------------------------------------------------------------
attention -- (a courteous act indicating affection; "she tried to win his heart with her many attentions")
    => act, human action, human activity -- (something that people do or cause to happen)
---------------------------------------------------------------------------------------------------------------------------------------
attention -- (the faculty or power of mental concentration; "keeping track of all the details requires your complete attention")
    => cognition, knowledge -- (the psychological result of perception and learning and reasoning)
        => psychological feature -- (a feature of the mental life of a living organism)
---------------------------------------------------------------------------------------------------------------------------------------
attention -- (a motionless erect stance with arms at the sides and feet together; assumed by military personnel during drill or review; "the troops stood at attention")
    => abstraction -- (a general concept formed by extracting common features from specific examples)
```

Fig. 4.6 Arborele simplificat de hipernime al substantivului *attention*

```
0 - psychological feature -- (a feature of the mental life of a living organism)
1 - act, human action, human activity -- (something that people do or cause to happen)
2 - abstraction -- (a general concept formed by extracting common features from specific examples)
```

Fig. 4.7 Synset-urile de vârf pentru substantivul *attention*

Pentru *attention* numărul de synset-uri între care trebuie să se opteze s-a redus de la 6 la 3. Putem extrage din tabelele de co-ocurențe construite în modulul de învățare cazurile referitoare la synset-urile de interes pentru aceste cuvinte, adică datele pentru cele 3 synset-uri de vârf ale substantivului *attention* referind unul dintre cele 8 ale verbului *pay*.

Tab. A	0	1	2	3	4	5	6	7
0	124	1175	21	67	70	257	937	159
1	115	970	17	64	78	222	792	151
2	203	1744	25	98	126	371	1408	279

Tabel 3.1 Tabela de co-ocurențe pentru synset-urile de vârf ale cuvintelor *attention* și *pay*

De asemenea, va fi probabil util pentru funcțiile de scor şi numărul total de apariții pentru fiecare din synset-urile cuvântului *attention*.

Tabel N	0	1	2
Număr apariții	2405	2027	3806

Tabel 3.2 Numărul de apariții pentru synset-urile de vârf ale cuvântului *attention*

Pentru *attention*, atunci când referă verbul *pay*, synset-ul de vârf care ar trebui să obțină scor maxim este 0 - *<psychological feature -- (a feature of the mental life of a living organism)>*, deoarece printre fii acestuia se găseşte sensul pentru care ar opta un utilizator uman: *<attention, attending -- (the process whereby a person concentrates on some features of the environment to the (relative) exclusion of others)>*.

Cel de-al doilea cuvânt pe care l-am ales pentru teste referind verbul *pay* este *thanks*. Motivul acestei selecții este faptul că *attention, thanks, regards, devotions, respects* sunt cuvinte ce apar referindu-l pe *pay* atunci când acesta are sensul *<give, pay -- (convey, as of a compliment, regards, attention, etc.; bestow; "Don't pay him any mind"; "give the orders"; "Give him my best regards"; "pay attention")>* şi au synset-uri de vârf comune ce le determină sensul în acest context. Am optat pentru *attention* şi *thanks* deoarece au structura arborilor de hipernime diferită (primul are 3 synset-uri de vârf, iar al doilea 2), cu două synset-uri de vârf comune, dar nu trebuie selectat acelaşi synset pentru ambele cuvinte.

```
The noun thanks has 2 senses (first 2 from tagged texts)

1. thanks, thanking -- (an acknowledgment of appreciation)
2. thanks -- (with the help of or owing to; "thanks to hard work it was a great success")

The verb thank has 1 sense (first 1 from tagged texts)

1. thank, give thanks -- (express gratitude or show appreciation to)
```

Fig. 4.8 Sensurile cuvântului *thanks.*

```
thanks, thanking -- (an acknowledgment of appreciation)
    => abstraction -- (a general concept formed by extracting common features from specific examples)
---------------------------------------------------------------------------------------------------------------------------
thanks -- (with the help of or owing to; "thanks to hard work it was a great success")
    => act, human action, human activity -- (something that people do or cause to happen)
```

Fig. 4.9 Arborele de hipernime simplificat pentru substantivul *thanks.*

0 - abstraction -- (a general concept formed by extracting common features from specific examples)
1 - act, human action, human activity -- (something that people do or cause to happen)

Fig. 4.10 Synset-urile de vârf pentru substantivul *thanks*.

Datele extrase din tabela de co-ocurențe pentru perechea *thanks* => *pay* sunt următoarele:

Tab. A	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**
0	203	1744	25	98	126	371	1408	279
1	115	970	17	64	78	222	792	151

Tabel 3.3 Tabela de co-ocurențe pentru synset-urile de vârf ale cuvintelor *thanks* şi *pay*

Tabel N	**0**	**1**
Număr apariții	3806	2027

Tabel 3.4 Numărul de apariții pentru synset-urile de vârf ale cuvântului *thanks*

Pentru cuvântul *thanks* synset-ul care trebuie să obțină scor maxim este 0 - *<abstraction -- (a general concept formed by extracting common features from specific examples)>*, deoarece sensul *<thanks, thanking -- (an acknowledgment of appreciation)>*, care ar fi ales de un utilizator uman se numără printre hiponimele sale.

Pentru cea de-a treia pereche de cuvinte folosită pentru teste am optat pentru un verb referind sintactic un substantiv – *try* şi *use.* Alegerea lor s-a făcut din două motive: testarea funcțiilor şi pentru relații între verbe referind substantive şi faptul că acestea s-au dovedit a fi un caz mai special la testele de dezambiguizare efectuate pe primele paragrafe ale romanului „1984” de G. Orwell.

The noun use has 7 senses (first 7 from tagged texts)
1. use, usage, utilization, utilisation, employment, exercise -- (the act of using; "the steps were worn from years of use")
2. use -- (a particular service; "he put his knowledge to good use"; "patrons have their uses")
3. function, purpose, role, use -- (what something is used for; "the function of an auger is to bore holes"; "ballet is beautiful but what use is it?")
4. consumption, economic consumption, usance, use, use of goods and services -- ((economics) the utilization of economic goods to satisfy needs or in manufacturing; "the consumption of energy has increased steadily")
5. habit, use, wont -- (a pattern of behavior acquired through frequent repetition; "she had a habit twirling the ends of her hair"; "long use had hardened him to it")
6. use, enjoyment -- (the exercise of a right to benefits; "we were given the use of his boat"; "deprived of the enjoyment of civil rights")
7. manipulation, use -- (exerting shrewd or devious influence especially for one's own advantage; "his manipulation of his friends was scandalous")

The verb use has 6 senses (first 6 from tagged texts)

1. use, utilize, utilise, apply, employ -- (put into service; make work; make use of of employ for a particular purpose: "use your head!" "I can't make use of this tool"; "Apply a magnetic field here"; "This thinking was applied to many projects"; "How do you utilize this tool?"; "I apply this rule to get good results")
2. use -- (take or consume (regularly); "She uses drugs rarely")
3. use -- (seek or achieve an end by using; "She uses her influential friends to get jobs")

4. use, expend -- (use up, consume fully; " The legislature expended its time on school questions.")
5. use -- (conduct oneself toward; treat or handle; "You can't use people to achieve your own evil plans!")
6. practice, apply, use -- (avail oneself to; "apply a principle"; "practice a religion"; "use care when going down the stairs")

Fig. 4.11 Sensurile cuvântului *use.*

use, usage, utilization, utilisation, employment, exercise -- (the act of using; "the steps were worn from years of use")
=> activity -- (any specific activity or pursuit; "they avoided all recreational activity")
=> act, human action, human activity -- (something that people do or cause to happen)
---
use -- (a particular service; "he put his knowledge to good use"; "patrons have their uses")
=> utility, usefulness -- (the quality of being of practical use)
=> abstraction -- (a general concept formed by extracting common features from specific examples)
---
function, purpose, role, use -- (what something is used for; "the function of an auger is to bore holes"; "ballet is beautiful but what use is it?")
=> utility, usefulness -- (the quality of being of practical use)
=> abstraction -- (a general concept formed by extracting common features from specific examples)
---
consumption, economic consumption, usance, use, use of goods and services -- ((economics) the utilization of economic goods to satisfy needs or in manufacturing; "the consumption of energy has increased steadily")
=> phenomenon -- (any state or process known through the senses rather than by intuition or reasoning)
---
habit, use, wont -- (a pattern of behavior acquired through frequent repetition; "she had a habit twirling the ends of her hair"; "long use had hardened him to it")
=> activity -- (any specific activity or pursuit; "they avoided all recreational activity")
=> act, human action, human activity -- (something that people do or cause to happen)
---
use, enjoyment -- (the exercise of a right to benefits; "we were given the use of his boat"; "deprived of the enjoyment of civil rights")
=> psychological feature -- (a feature of the mental life of a living organism)
---
manipulation, use -- (exerting shrewd or devious influence especially for one's own advantage; "his manipulation of his friends was scandalous")
=> act, human action, human activity -- (something that people do or cause to happen)

Fig. 4.12 Arborele simplificat de hipernime pentru substantivul *use.*

0 - act, human action, human activity -- (something that people do or cause to happen)
1 - abstraction -- (a general concept formed by extracting common features from specific examples)
2 - phenomenon -- (any state or process known through the senses rather than by intuition or reasoning)
3 - psychological feature -- (a feature of the mental life of a living organism)

Fig. 4.13 Synset-urile de vârf pentru substantivul *use.*

The noun try has 1 sense (first 1 from tagged texts)

1. attempt, effort, endeavor, endeavour, try -- (earnest and conscientious activity intended to do or accomplish something: "made an effort to cover all the reading material"; "wished him luck in his endeavor"; "she gave it a good try")

The verb try has 9 senses (first 4 from tagged texts)

1. try, seek, attempt, essay, assay -- (make an effort or attempt; "He tried to shake off his fears"; "The infant had essayed a few wobbly steps"; "The police attempted to stop the thief"; "He sought to improve himself"; "She always seeks to do good in the world")
2. test, prove, try, try out, examine, essay -- (put to the test, as for its quality, or give experimental use to; "This approach has been tried with good results"; "Test this recipe")

3. judge, adjudicate, try -- (put on trial or sit as the judge at the trial of; "The football star was tried for the murder of his wife"; "The judge tried both father and son in separate trials")
4. sample, try, try out, taste -- (take a sample of; "Try these new crackers"; "Sample the regional dishes")
5. hear, try -- (examine or hear (evidence or a case) by judicial process; "The jury had heard all the evidence"; "The case will be tried in California")
6. try -- (give pain or trouble to; "I've been sorely tried by these students")
7. try, strain, stress -- (test the limits of; "You are trying my patience!")
8. try, render -- (melt (fat, lard, etc.) in order to separate out impurities; "try the yak butter")
9. try on, try -- (put on a garment in order to see whether it fits and looks nice; "Try on this sweater to see how it looks")

Fig. 4.14 Sensurile cuvântului *try*.

try, seek, attempt, essay, assay -- (make an effort or attempt; "He tried to shake off his fears"; "The infant had essayed a few wobbly steps"; "The police attempted to stop the thief"; "He sought to improve himself"; "She always seeks to do good in the world")
    => act, move -- (perform an action; "think before you act"; "We must move quickly")
---
test, prove, try, try out, examine, essay -- (put to the test, as for its quality, or give experimental use to; "This approach has been tried with good results"; "Test this recipe")
    => judge -- (form an opinion of or pass judgment on)
---
judge, adjudicate, try -- (put on trial or sit as the judge at the trial of; "The football star was tried for the murder of his wife"; "The judge tried both father and son in separate trials")
    => decide, make up one's mind, determine -- (reach, make, or come to a decision about something; "We finally decided after lengthy deliberations")
---
sample, try, try out, taste -- (take a sample of; "Try these new crackers"; "Sample the regional dishes")
    => consume, ingest, take in, take, have -- (serve oneself to, or consume regularly; "Have another bowl of chicken soup!" "I don't take sugar in my coffee")
---
hear, try -- (examine or hear (evidence or a case) by judicial process; "The jury had heard all the evidence"; "The case will be tried in California")
    => analyze, analyse, study, examine -- (consider in detail and subject to an analysis in order to discover essential features or meaning; "analyze a sonnet by Shakespeare"; "analyze the evidence in a criminal trial"; "analyze your real motives")
---
try -- (give pain or trouble to; "I've been sorely tried by these students")
    => upset, discompose, untune, disconcert, discomfit -- (cause to lose one's composure)
        => make, create -- (make or cause to be or to become; "make a mess in one's office"; "create a furor")
---
try, strain, stress -- (test the limits of; "You are trying my patience!")
    => upset, discompose, untune, disconcert, discomfit -- (cause to lose one's composure)
        => make, create -- (make or cause to be or to become; "make a mess in one's office"; "create a furor")
---
try, render -- (melt (fat, lard, etc.) in order to separate out impurities; "try the yak butter")
    => change -- (undergo a change; become different in essence; losing one's or its original nature; "She changed completely as she grew older"; "The weather changed last night")
---
try on, try -- (put on a garment in order to see whether it fits and looks nice; "Try on this sweater to see how it looks")
    => change -- (undergo a change; become different in essence; losing one's or its original nature; "She changed completely as she grew older"; "The weather changed last night")

Fig. 4.15 Arborele simplificat de hipernime ale verbului *try*.

0 - act, move -- (perform an action; "think before you act"; "We must move quickly")
1 - judge -- (form an opinion of or pass judgment on)
2 - decide, make up one's mind, determine -- (reach, make, or come to a decision about something; "We finally decided after lengthy deliberations")

3 - consume, ingest, take in, take, have -- (serve oneself to, or consume regularly; "Have another bowl of chicken soup!" "I don't take sugar in my coffee")
4 - analyze, analyse, study, examine -- (consider in detail and subject to an analysis in order to discover essential features or meaning; "analyze a sonnet by Shakespeare"; "analyze the evidence in a criminal trial"; "analyze your real motives")
5 - make, create -- (make or cause to be or to become; "make a mess in one's office"; "create a furor")
6 - change -- (undergo a change; become different in essence; losing one's or its original nature; "She changed completely as she grew older"; "The weather changed last night")

Fig. 4.16 Synset-urile de vârf pentru verbul *try*.

Pentru perechea *try – use*, în tabela de co-ocurențe, se găsesc următoarele date:

Tabel A	**0**	**1**	**2**	**3**
0	113	210	35	155
1	32	51	12	46
2	8	16	6	11
3	11	20	2	7
4	5	9	2	7
5	49	80	14	62
6	44	80	13	59

Tabel 3.5 Tabela de co-ocurențe pentru synset-urile de vârf ale verbului *try* şi substantivului *use.*

Tabel N	**0**	**1**	**2**	**3**	**4**	**5**	**6**
Număr apariții	1812	414	206	161	50	717	573

Tabel 3.6 Numărul de apariții ale synset-ului de vârf pentru verbul *try.*

O funcție de scor ar trebui să returneze în acest caz o valoare maximă pentru synset-ul 0 - *<act, move -- (perform an action; "think before you act"; "We must move quickly")>*, deoarece acesta este hipernimul suprem pentru sensul corect al verbului *try* în acest context, respectiv *<try, seek, attempt, essay, assay -- (make an effort or attempt; "He tried to shake off his fears"; "The infant had essayed a few wobbly steps"; "The police attempted to stop the thief"; "He sought to improve himself"; "She always seeks to do good in the world")>*.

Problemele care vor fi puse în evidență cel mai bine în acest caz de test vor fi cele legate de numărul total de apariții al synset-urilor (care variază de la 50 la 1812), mai exact importanța pe care acest număr o are în alegerea unui sens.

În tabelele de valori ce urmează, conținând valori returnate de funcțiile de scor testate pentru diferite perechi de cuvinte, vom nota cu paranteze numărul synset-ului care ar trebui să obțină scor maxim şi cu * synset-ul care obține acest scor. Trebuie menționat că dezambiguizarea se face pentru cuvântul care referă sintactic, dar, în general, ca produs

auxiliar se obține şi dezambiguizarea cuvântului părinte. Pentru funcțiile de scor testate am dat însă importanță numai rezultatelor obținute pentru cuvântul fiu.

4.3 Funcția de scor 1

Prima funcție de scor testată a fost una simplă, intuitivă, care de altfel va sta la baza tuturor celorlalte. Practic, aceasta calculează posibilitatea ca un synset să îl refere pe altul prin formula clasică a calculului probabilității: raportul dintre numărul de cazuri favorabile şi numărul total de posibilități.

$$f(s1, s2) = \frac{A[s1, s2]}{N[s1]}$$

Pentru perechile de test funcția returnează următoarele valori:

	0	**(1)***	**2**	**3**	**4**	**5**	**6**	**7**
(0)*	0,0515	0,4885	0,0087	0,0278	0,0291	0,1068	0,3896	0,0661
1	0,0567	0,4785	0,0083	0,0315	0,0384	0,1095	0,3907	0,0744
2	0,0533	0,4582	0,0065	0,0257	0,0331	0,0974	0,3699	0,0733

Tabel 3.7 Scorurile obținute pentru perechea *attention - pay* folosind funcția de scor 1.

	0	**(1)***	**2**	**3**	**4**	**5**	**6**	**7**
(0)	0,0533	0,4582	0,0065	0,0257	0,0331	0,0974	0,3699	0,0733
1*	0,0567	0,4785	0,0083	0,0315	0,0384	0,1095	0,3907	0,0744

Tabel 3.8 Scorurile obținute pentru perechea *thanks – pay* folosind funcția de scor nr. 1.

	(0)	1*	2	3
(0)	0,0623	0,1158	0,0193	0,0855
1	0,0772	0,1231	0,0289	0,1111
2	0,0388	0,0776	0,0291	0,0533
3	0,0683	0,1242	0,0124	0,0434
4*	0,1000	0,1800	0,0400	0,1400
5	0,0683	0,1115	0,0195	0,0864
6	0,0767	0,1396	0,0226	0,1029

Tabel 3.9 Scorurile obținute pentru perechea *try – use* folosind funcția de scor 1.

Funcția se comportă bine pentru perechea *attention – pay*, identificând corect synset-urile pentru ambele cuvinte. Pentru perechea *thanks – pay* însă, deşi scorurile returnate sunt apropiate, alegerea synset-ului pentru *thanks* este incorectă. În cel de-al treilea caz diferența de scor este considerabilă, iar synset-urile alese nu mai sunt nici măcar în apropierea celor corecte. Din acest ultim test însă, putem ajunge la o concluzie referitor la datele care au generat această situație şi cum putem îmbunătăți funcția pentru o acuratețe mărită.

După cum se observă din tabelul 3.9, synset-ul cu numărul 4 a obținut scoruri mari în relație cu toate synset-urile cuvântului părinte. Corelând acest fapt cu datele din tabelul 3.6, devine clar că aceste valori mari au fost obținute datorită faptului că synset-ul 4 apare de puține ori în corpusul de învățare şi, din această cauză, îi este uşor să atingă probabilități mari. Se impune, deci, ponderarea scorului obținut în funcție de importanța synset-ului în raport cu celelalte synset-uri de vârf ale cuvântului.

4.4 Funcția de scor 2.

Cea de-a doua funcție de scor testată reprezintă de fapt varianta ponderată după observațiile de mai sus a primei funcții. Ponderarea se realizează prin înmulțirea scorului obținut inițial cu probabilitatea ca synset-ul respectiv să apară ca synset ales pentru cuvântul prelucrat.

$$f(S_1, S_2) = \frac{A[S_1][S_2]}{N[S_1]} * \frac{N[S_1]}{\sum_k N[S_k]} = \frac{A[S_1][S_2]}{\sum_k N[S_k]}$$

Pentru perechile de test funcția returnează următoarele valori:

	0	(1)*	2	3	4	5	6	7
(0)	0,0150	0,1426	0,0025	0,0081	0,0084	0,0311	0,1137	0,0193
1	0,0139	0,1177	0,0020	0,0077	0,0094	0,0269	0,0961	0,0183
2*	0,0246	0,2117	0,0030	0,0118	0,0152	0,0450	0,1709	0,0338

Tabel 3.10 Scorurile obținute pentru perechea *attention - pay* folosind funcția de scor 2.

	0	(1)*	2	3	4	5	6	7
(0)*	0,0348	0,2989	0,0042	0,0168	0,0216	0,0636	0,2413	0,0478
1	0,0197	0,1662	0,0029	0,0109	0,0133	0,0380	0,1357	0,0258

Tabel 3.11 Scorurile obținute pentru perechea *thanks – pay* folosind funcția de scor nr. 2.

	(0)	1*	2	3
(0)*	0,0287	0,0533	0,0088	0,0394
1	0,0081	0,0129	0,0030	0,0116
2	0,0020	0,0040	0,0015	0,0027
3	0,0027	0,0050	0,0005	0,0017
4	0,0012	0,0022	0,0005	0,0017
5	0,0124	0,0203	0,0035	0,0157
6	0,0111	0,0203	0,0033	0,0150

Tabel 3.12 Scorurile obținute pentru perechea *try – use* folosind funcția de scor 2.

Metoda ponderării în funcție de numărul total de apariții ale synset-urilor a dat rezultate parțiale. Pentru a doua şi a treia pereche de test s-a reuşit dezambiguizarea corectă a sensului cuvântului – fiu, dar pentru prima nu s-a mai păstrat sensul corect ales folosind funcția de scor 1. Totuşi, dacă privim noile valori obținute pentru synset-ul 4 al cuvântului *try*, observăm că introducerea acestei ponderări în formula de calcul a scorului este corectă. Astfel, synset-ul 4 are acum unele dintre cele mai mici valori, ceea ce este mai aproape de realitate.

Deşi corectă, introducerea sa trebuie făcută în alt mod, deoarece astfel se reduce uneori prea mult din importanța probabilității calculate prin prima formulă. Ca soluții la această problemă am încercat diferite metode de reducere a diferențelor mari de scor în cazuri cu probabilități inițiale apropiate:

- introducerea ponderării sub formă de logaritm natural, creşterea sa pentru valori mari fiind mai mică;

- introducerea în formulă şi a unor probabilităţi calculate în funcţie de importanţa synset-ului pe o anumită coloană sau linie;

- calcularea scorului folosind formula lui Bayes pentru probabilităţi condiţionate.

Deşi în unele cazuri s-au obţinut îmbunătăţiri ale acurateţei, nu s-a reuşit obţinerea unei formule care să satisfacă majoritatea cazurilor. În continuare ne vom limita la prezentarea testului efectuat pe baza formulei lui Bayes, care, deşi nu obţine rezultate notabile, va influenţa cursul viitor al cercetării problemei.

4.5 Funcţia de scor 3 – formula lui Bayes.

Prin faptul că trebuie calculată probabilitatea optării pentru un synset pentru cuvântul fiu în funcţie de synset-ul ales pentru părinte, problema găsirii unei funcţii de scor care să aproximeze probabilitatea reală se aseamănă destul de mult cu problema calculării unei probabilităţi pentru un eveniment condiţionat de un altul.

Mai precis, în cadrul formulei lui Bayes avem evenimentele X_1, X_2, ..., X_k, care realizează o partiţie a spaţiului se selecţie S, şi un eveniment A din S. Se cunosc

$$P(X_i|A) = \frac{P(X_i) * P(A \mid X_i)}{\sum_{j=1}^{k} P(X_j) * P(A \mid X_j)}, \forall i = \overline{1,k}$$

probabilităţile $P(X_1)$, $P(X_2)$, ..., $P(X_k)$, precum şi probabilităţile condiţionate $P(A|X_1)$, $P(A|X_2)$, ..., $P(A|X_k)$. În a cest caz, determinarea $P(X_i|A)$ se face prin formula:

În cazul de faţă putem considera partiţia spaţiului de selecţie ca fiind formată din synset-urile X_1, X_2, ..., X_k ale cuvântului – fiu pentru care se face dezambiguizarea.

Probabilităţile fiecăruia le putem considera:

$$P(X_i) = \frac{N[X_1]}{\sum_k N[X_k]}, unde \sum_i P(X_i) = 1$$

Evenimentul care condiţionează alegerea este synset-ul pentru care s-a optat la nivelul părintelui sintactic. Probabilitatea synset-ului a_j, ales pentru părinte, condiţionat de fiecare synset al cuvântului - fiu, va fi calculată prin formula:

$$P(A \mid X_i) = \ln\left(\frac{A[X_i][a_j]}{N[X_i]} * e + 1\right)$$

Formula este practic aceeaşi care a fost folosită la prima funcţie de scor, dar uniformizată pentru valori mai mari prin aplicarea unui logaritm natural, înmulţirea raportului cu *e* şi adunarea cu 1 având rolul de a aduce valorile formulei în intervalul [0, 1].

Funcţia de scor care va fi testată în continuare va folosi formula lui Bayes în care probabilităţile vor fi calculate conform definiţiilor de mai sus.

	(0)	1	2*
$P(X_i)$	0,2919	0,2460	0,4620

Tabel 3.13 Probabilitățile calculate pentru synset-urile de vârf ale substantivului *attention*.

	0	(1)*	2	3	4	5	6	7
(0)*	0,1311	0,8450	0,0234	0,0729	0,0761	0,2550	0,7222	0,1652
1	0,1434	0,8332	0,0255	0,0823	0,0994	0,2606	0,7237	0,1843
2	0,1353	0,8089	0,0176	0,0676	0,0861	0,2350	0,6959	0,1817

Tabel 3.14 Probabilitățile condiționate calculate pentru synset-urile de vârf ale substantivului *attention* atunci când acesta referă verbul *pay*.

	0	(1)	2	3	4	5	6	7*
(0)	0,2812	0,2988	0,3329	0,2926	0,2569	0,3012	0,2967	0,2717
1	0,2592	0,2483	0,2696	0,2782	02829	0,2529	0,2506	0,2555
2*	0,4594	0,4527	0,3974	0,4291	0,4601	0,4393	0,4525	0,4727

Tabel 3.15 Scorurile obținute prin formula lui Bayes pentru perechea *attention – pay*.

După cum se observă din tabelul 3.15 alegerea făcută folosind această funcție este departe de a fi corectă. Dacă analizăm scorurile obținute, se constată că probabilitățile, pe coloană, sunt foarte apropiate de valorile probabilităților calculate pentru synset-urile de vârf ale substantivului *attention*. Două sunt concluziile care pot fi trase din aceste rezultate:

- probabilitățile condiționate de synset-ul părintelui trebuie să aibă o importanță mărită față de cele calculate numai în funcție de celelalte synset-uri de vârf ale cuvântului – fiu;
- probabilitățile calculate în funcție de suma numărului total de apariții ale synset-urilor de vârf sunt corecte, dar mai utile pentru cazul nostru ar fi probabilitățile calculate în funcție de suma pe coloană a numărului de referiri ale synset-ului ales pentru părinte.

4.6 Funcția de scor 4.

Importanța care trebuie acordată probabilității alegerii synset-ului în funcție de numărul său total de apariții, față de cea calculată în funcție de celelalte synset-uri de vârf ale cuvântului prelucrat poate diferi de la caz la caz.

În principiu, am concluzionat că o funcție de forma $f(Xi) = P(Xi \mid Aj) * P(Xi)$, unde lui $P(Xi)$ îi este dată o importanță mai mică, este mai aproape de ceea ce căutăm. Un alt motiv

pentru care am optat pentru o funcție de această formă este acela că se poate realiza mult mai uşor o „reglare fină” a rezultatelor.

În urma diverselor teste efectuate, am ajuns la concluzia că importanța acordată lui *P(Xi)* trebuie să fie mai mare pentru cazul în care aceasta este mică (sub 25%), dar pentru valori mai mari diferența de scor rezultată din valoarea lui *P(Xi)* nu mai trebuie să fie decisivă.

Mai clar, s-a impus găsirea unei funcții, *w: [0; 1] -> [0; 1]* ale cărei valori să urce brusc pe intervalul [0; 0,25], dar pentru restul intervalului creşterea să fie mult mai mică, nulă chiar. Prima funcție în jurul căreia am încercat construirea acestei funcții a fost logaritmul. Rezultatele obținute nu au fost însă mulțumitoare, creşterea logaritmului fiind continuă pe tot domeniul de definiție. De asemenea, încercările de particularizare – modificarea intervalului pentru care creşterea este bruscă – s-au dovedi dificile.

O funcție care are un grafic asemănător cu cel pe care îl căutăm, chiar dacă „întors”, este tangenta. Pentru construcția funcției de care avem nevoie ne-a interesat porțiunea din graficul tangentei corespunzătoare intervalului $\left[\frac{\Pi}{4}+\frac{\Pi}{4}*P(X_i)\right)$.

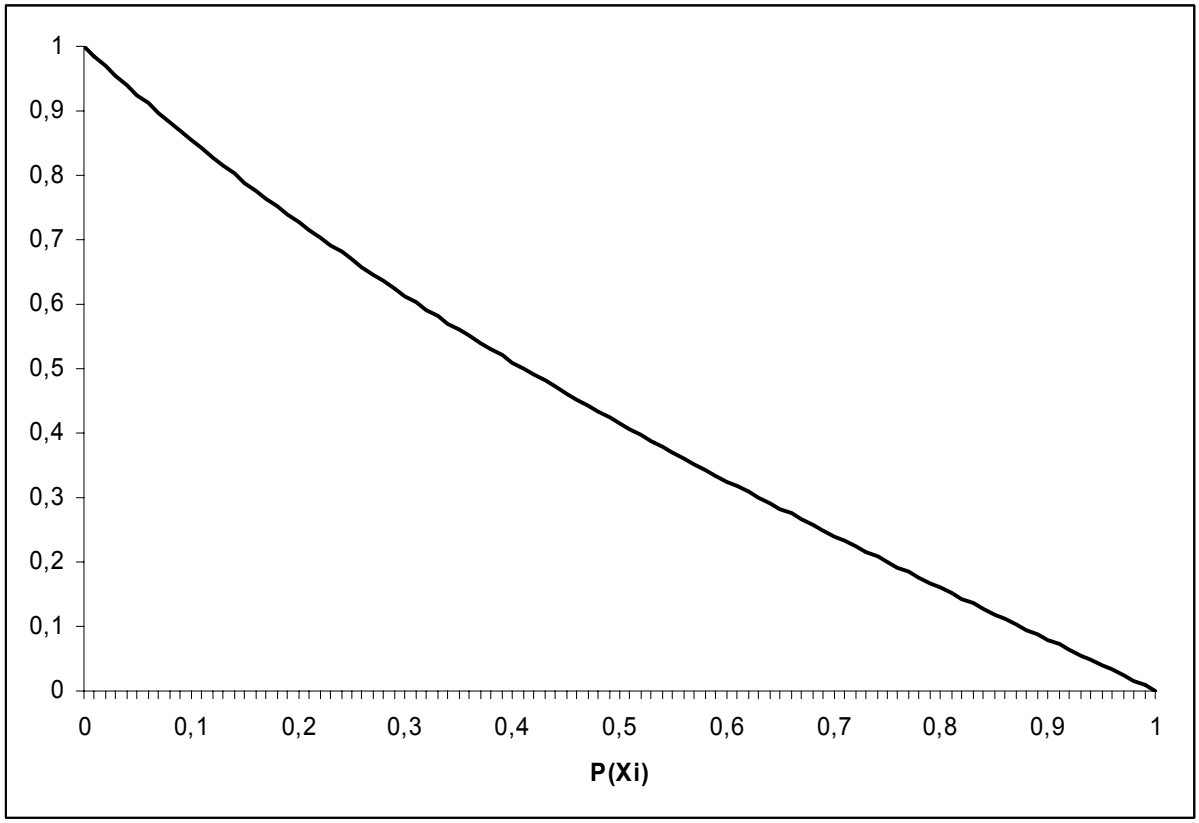

Fig. 4.17 Graficul funcției $\frac{1}{tg\left(\frac{\Pi}{4}+\frac{\Pi}{4}*P(X_i)\right)}, P(X_i)\in[0;1]$.

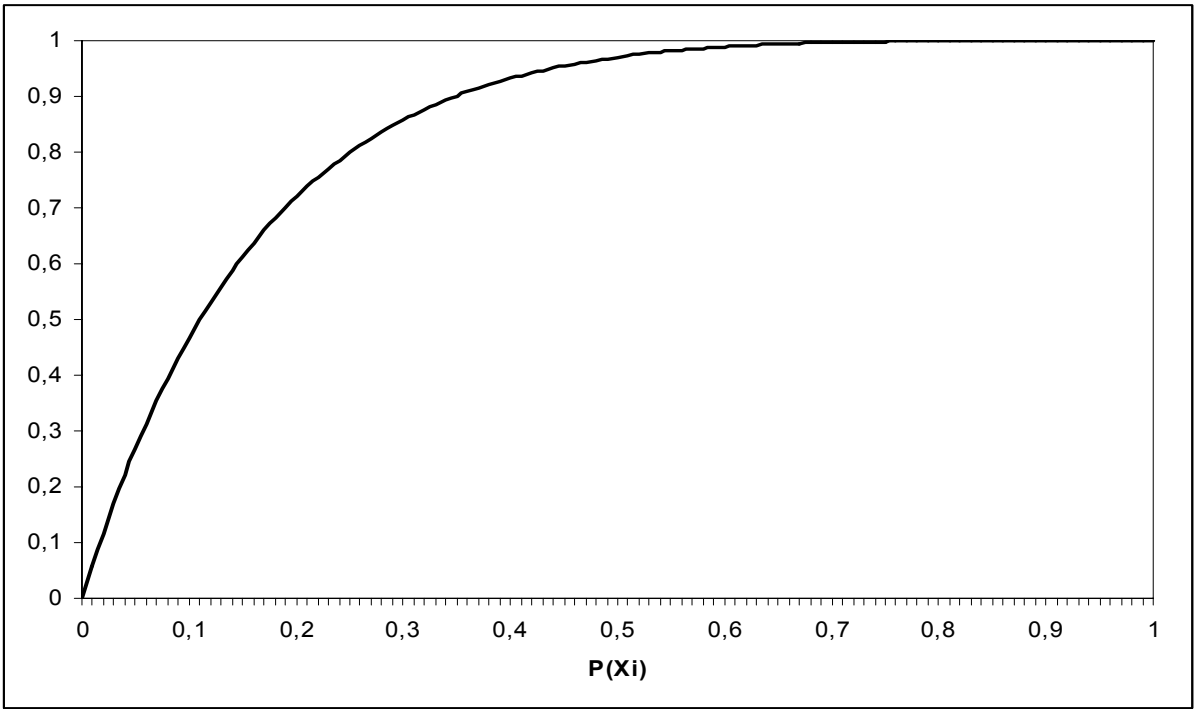

Fig. 4.18 Graficul funcției $w(X_i) = 1 - \frac{1}{tg^4\left(\frac{\Pi}{4} + \frac{\Pi}{4} * P(X_i)\right)}, P(X_i) \in [0;1].$

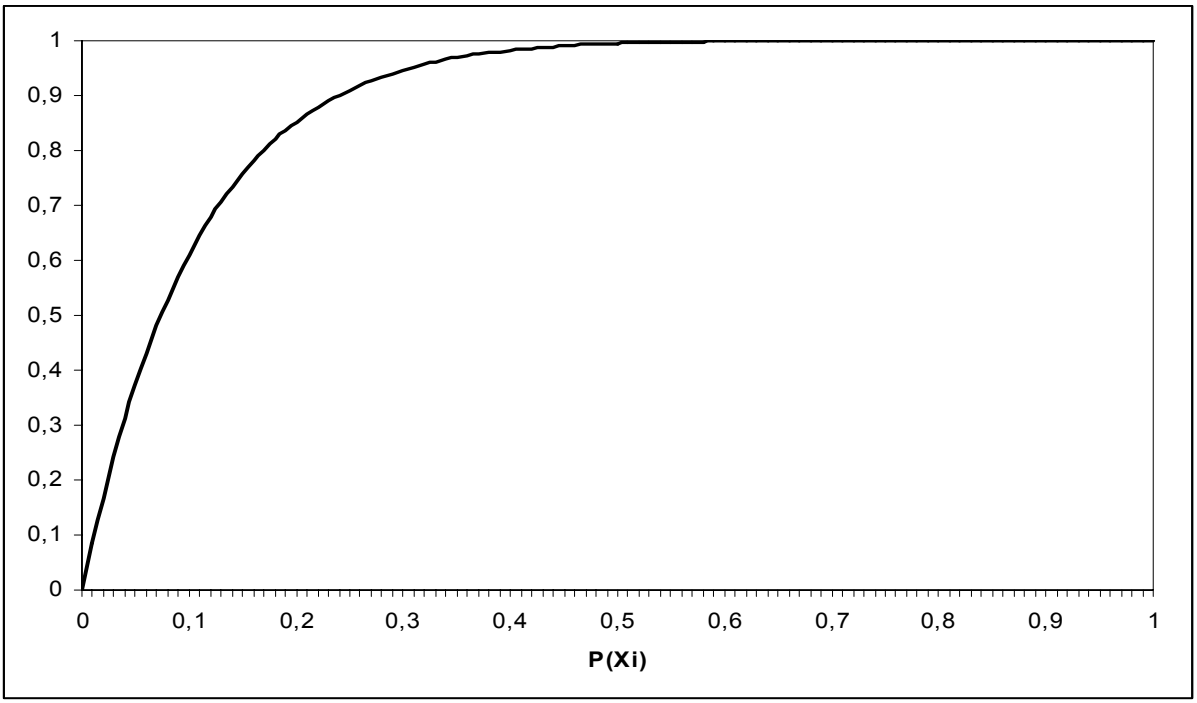

Fig. 4.19 Graficul funcției $w(X_i) = 1 - \frac{1}{tg^6\left(\frac{\Pi}{4} + \frac{\Pi}{4} * P(X_i)\right)}, P(X_i) \in [0;1].$

Avantajul acestei funcții este că putem regla uşor modul în care creşte modificând puterea tangentei. Dacă pentru puterea întâi tangenta descreşte aproape constant, începând cu puterea a doua graficul ei începe să semene cu cel al funcției pe care o căutăm. Trebuie, de asemenea, „inversată" descreşterea prin scăderea valorii obținute din 1.

Din cele două grafice prezentate se observă cum, prin mărirea puterii la care se află funcția tangentă, se micşorează intervalul pentru care creşterea funcției *w* este mai accentuată.

Mai rămâne, deci, de stabilit valoarea optimă pentru această putere astfel încât funcția de scor obținută să dea rezultate corecte în majoritatea cazurilor. Funcția de scor pentru care s-au efectuat testele în continuare este:

$$w(X_i, a_j) = \ln\left(\frac{A[X_i, a_j]}{N[X_i]} * \left(1 - \frac{1}{tg^{\alpha}\left(\frac{\Pi}{4} + \frac{\Pi}{4} * P(X_i)\right)}\right) * e + 1\right), \text{unde } P(X_i) = \frac{A[X_i, a_j]}{\sum_{k=1}^{n} A[X_k, a_j]}.$$

Am considerat pentru α valorile 4 şi 6, acestea fiind obținute din teste ca posibile pentru cifra căutată.

Rezultatele obținute pentru perechile de test sunt prezentate în continuare:

α = 4

	0	**(1)***	**2**	**3**	**4**	**5**	**6**	**7**
(0)	0,1110	0,7624	0,0208	0,0624	0,0619	0,2232	0,6462	0,1384
1	0,1182	0,7131	0,0186	0,0693	0,0845	0,2173	0,6173	0,1517
2*	0,1300	0,7832	0,0164	0,0641	0,0827	0,2244	0,6728	0,1755

Tabel 3.16 Scorurile obținute pentru perechea *attention - pay* folosind funcția de scor 4, cu α = 4.

	0	**(1)***	**2**	**3**	**4**	**5**	**6**	**7**
(0)*	0,1344	0,8051	0,0174	0,0669	0,0854	0,2332	0,6923	0,1806
1	0,1312	0,7791	0,0210	0,0767	0,0921	0,2416	0,6755	0,1678

Tabel 3.17 Scorurile obținute pentru perechea *thanks – pay* folosind funcția de scor nr. 4, cu α = 4.

	(0)	1*	2	3
(0)*	0,1489	0,2630	0,0482	0,2002
1	0,1070	0,1545	0,0458	0,1582
2	0,0182	0,0401	0,0282	0,0258
3	0,0421	0,0768	0,0046	0,0139
4	0,0302	0,0544	0,0149	0,0443
5	0,1216	0,1835	0,0340	0,1418
6	0,1283	0,2248	0,0378	0,1697

Tabel 3.18 Scorurile obținute pentru perechea *try – use* folosind funcția de scor 4, cu α = 4.

α = 6

	0	(1)*	2	3	4	5	6	7
(0)*	0,1231	0,8150	0,0225	0,0689	0,0699	0,2433	0,6942	0,1541
1	0,1326	0,7813	0,0209	0,0771	0,0935	0,2422	0,6780	0,1702
2	0,1343	0,8035	0,0173	0,0668	0,0854	0,2326	0,6910	0,1805

Tabel 3.19 Scorurile obținute pentru perechea *attention - pay* folosind funcția de scor 4, cu α = 6.

	0	(1)*	2	3	4	5	6	7
(0)	0,1353	0,8086	0,0176	0,0675	0,0860	0,2348	0,6956	0,1816
1*	0,1397	0,8170	0,0221	0,0808	0,0974	0,2551	0,7094	0,1792

Tabel 3.20 Scorurile obținute pentru perechea *thanks – pay* folosind funcția de scor nr. 4, cu α = 6.

	(0)	1*	2	3
(0)*	0,1548	0,2715	0,0504	0,2071
1	0,1346	0,1956	0,0568	0,1957
2	0,0260	0,0576	0,0381	0,0368
3	0,0589	0,1065	0,0067	0,0202
4	0,0437	0,0782	0,0216	0,0638
5	0,1438	0,2186	0,0413	0,1759
6	0,1541	0,2667	0,0464	0,2026

Tabel 3.21 Scorurile obținute pentru perechea *try – use* folosind funcția de scor 4, cu $\alpha = 6$.

Din testele de mai sus se observă că stabilirea unei valori fixe pentru α nu este posibilă. Aceasta variază în funcție de cuvânt, mai exact de modul în care sunt repartizate probabilitățile synset-urilor de vârf.

Testele au arătat că, pentru cuvintele cu diferențe mari între valorile probabilităților synset-urilor, $\alpha = 4$ este o valoare care duce la o comportare bună a funcției, pe când, în cazul cuvintelor cu diferențe mai mici, $\alpha = 6$ este alegerea corectă. Explicația acestui fapt decurge din faptul că suma probabilităților este 1 şi modul în care arată graficul funcției pentru diferite valori ale lui α. Astfel, diferențe mari între valorile probabilităților înseamnă una sau mai multe valori mari, iar celelalte mici – suma fiind 1, deci punctul în care funcția îşi încetineşte creşterea se poate situa mai sus (0,3 – 0,4). În cazul diferențelor mici, majoritatea probabilităților sunt apropiate ca valoare, şi deci mai mici (sunt situate în jurul valorii de 1/n), deci trebuie coborât şi punctul în care funcția îşi stopează creşterea accelerată.

Pentru a stabili în ce caz ne aflăm vom folosi deviația standard a eşantionului format din valorile probabilităților synset-urilor de vârf ale cuvântului.

Media eşantionului este:

$$\overline{X} = \frac{\sum_{i=1}^{n} P(X_i)}{n} = \frac{1}{n}$$

Dispersia:

$$v = \frac{\sum_{i=1}^{n} \left(P(X_i) - \overline{X} \right)^2}{n}$$

Deviația standard:

$$SD = \sqrt{v}$$

Pentru perechile de test, valorile obținute sunt următoarele (valorile sunt mărite de 100 de ori):

	P(Xi)							Media	Deviația standard
I					29	24	47	33,33333	12,09683154
II						34	66	50	22,627417
III	47	14	18	1	4	5	11	14,28571	15,61745056

Tabel 3.22 Datele statistice pentru probabilitățile cuvintelor de test.

Pe baza acestor teste efectuate pe un număr mai mare de cuvinte, am stabilit pragul pentru care α se schimbă la 0,17. Astfel, pentru cuvintele care au deviația standard a synset-urilor de vârf mai mică decât 0,17 - α = 6, iar în rest α = 4.

Capitolul 5
Dezambiguizarea unui text

Am ales pentru testarea modelului pe propoziții întregi primele două paragrafe din „1984” de G. Orwell. Alegerea unui text care a fost folosit şi pentru învățare a fost făcută şi pentru a ne asigura că textul de test aparține aceluiaşi domeniu cu textul pe care s-a efectuat învățarea.

It was a bright cold day in April, and the clocks were striking thirteen. Winston Smith, his chin nuzzled into his breast in an effort to escape the vile wind, slipped quickly through the glass doors of Victory Mansions, though not quickly enough to prevent a swirl of gritty dust from entering along with him.

The hallway smelt of boiled cabbage and old rag mats. At one end of it a coloured poster, too large for indoor display, had been tacked to the wall. It depicted simply an enormous face, more than a metre wide: the face of a man of about forty-five, with a heavy black moustache and ruggedly handsome features. Winston made for the stairs. It was no use trying the lift. Even at the best of times it was seldom working, and at present the electric current was cut off during daylight hours. It was part of the economy drive in preparation for Hate Week.

Folosind funcția de scor 4 cu îmbunătățirile arătate mai sus, am obținut următoarea ieşire:

Sensuri posibile cuvant be :
embody, be, personify -- (represent, as of a character on stage; "Derek Jacobi was Hamlet")
Sensuri posibile cuvant strike :
strike -- (indicate a certain time by striking, of clocks)
Sensuri posibile cuvant be :
embody, be, personify -- (represent, as of a character on stage; "Derek Jacobi was Hamlet")
Sensuri posibile cuvant clock :
clock -- (a timepiece that shows the time of day)
Sensuri posibile cuvant day :
day, twenty-four hours, solar day, mean solar day -- (time for Earth to make a complete rotation on its axis; "two days later they left"; "they put on two performances every day"; "there are 30,000 passangers per day")
day -- (some point or period in time; "it should arrive any day now"; "after that day she never trusted him again"; "those were the days"; "these days it is not unusual")
day, daytime, daylight -- (the time after sunrise and before sunset while it is light outside; "the dawn turned night into day"; "it is easier to make the repairs in the daytime")
day -- (a day assigned to a particular purpose or observance; "Mother's Day")
day -- (the recurring hours established by contract or usage for work; "it was a busy day on the stock exchange")
day -- (an era of existence or influence; "in the day of the dinosaurs"; "in the days of the Roman Empire"; "in the days of sailing ships")
day -- (the period of time taken by a particular planet (e.g. Mars) to make a complete rotation on its axis; "how long is a day on Jupiter?")
sidereal day, day -- (the time for one complete rotation of the earth relative to a particular star, about 4 minutes shorter than a mean solar day)
sidereal day, day -- (the time for one complete rotation of the earth relative to a particular star, about 4 minutes shorter than a mean solar day)
Sensuri posibile cuvant April :
April, Apr -- (the month following March and preceding May)

Sensuri posibile cuvant slip :
slip, drop off, drop away, fall away -- (get worse; "My grades are slipping")
Sensuri posibile cuvant smith :
smith -- (someone who works at something specified)

smith, metalworker -- (someone who works metal (especially by hammering it when it is hot and malleable))
smith -- (someone who works at something specified)
Sensuri posibile cuvant chin :
chin, mentum -- (the protruding part of the lower jaw)
Sensuri posibile cuvant breast :
breast -- (the front part of the trunk from the neck to the abdomen; "he beat his breast in anger")
breast, bosom, knocker, boob, tit, titty -- (either of two soft fleshy milk-secreting glandular organs on the chest of a woman)
Sensuri posibile cuvant effort :
attempt, effort, endeavor, endeavour, try -- (earnest and conscientious activity intended to do or accomplish something: "made an effort to cover all the reading material"; "wished him luck in his endeavor"; "she gave it a good try")
effort, elbow grease, exertion, travail, sweat -- (use of physical or mental energy; hard work; "he got an A for effort"; "they managed only with great exertion")
deed, feat, effort, exploit -- (a notable achievement: "the book was her finest effort")
campaign, cause, crusade, drive, movement, effort -- (a series of actions advancing a principle or tending toward a particular end; "he supported populist campaigns"; "they worked in the cause of world peace"; "the team was ready for a drive toward the pennant"; "the movement to end slavery"; "contributed to the war effort")
Sensuri posibile cuvant escape :
elude, escape -- (be incomprehensible to; escape understanding by; "What you are seeing in him eludes me")
Sensuri posibile cuvant wind :
wind, idle words, jazz, nothingness -- (empty or insincere or exaggerated talk; that's a lot of wind"; "don't give me any of that jazz")
tip, lead, steer, confidential information, wind, hint -- (an indication of potential opportunity; "he got a tip on the stock market"; "a good lead for a job")
Sensuri posibile cuvant door :
door -- (a swinging or sliding barrier that will close the entrance to a room or building; "he knocked on the door"; "he slammed the door as he left")
doorway, door, room access, threshold -- (the space in a wall through which you enter or leave a room or building; the space that a door can close; "he stuck his head in the doorway")
door -- (a swinging or sliding barrier that will close off access into a car; "she forgot to lock the doors of her car")
door -- (a house that is entered via a door; "the house next door"; "they live just two doors up the street from us")
door -- (a room that is entered via a door; "his office is three doors down the hall on the left")
Sensuri posibile cuvant glass :
glass -- (a brittle transparent solid with irregular atomic structure)
glass, drinking glass -- (a glass container for holding liquids while drinking)
field glass, glass, spyglass -- (a small refracting telescope)
looking glass, glass -- (a mirror; usually a ladies' dressing mirror)
glass -- (glassware collectively; "She collected old glass")
Sensuri posibile cuvant mansion :
sign of the zodiac, sign, mansion, house, planetary house -- (one of 12 equal areas into which the zodiac is divided)
mansion, mansion house, manse, hall, residence -- (a large and imposing house)
Sensuri posibile cuvant victory :
victory, triumph -- (a successful ending of a struggle or contest; "the general always gets credit for his army's victory"; "the agreement was a triumph for common sense")
Sensuri posibile cuvant prevent :
prevent, keep -- (prevent from doing something or being in a certain state; "We must prevent the cancer from spreading")
Sensuri posibile cuvant swirl :
whirl, swirl, vortex, convolution -- (the shape of something rotating rapidly)
Sensuri posibile cuvant dust :
dust -- (fine powdery material such as dry earth or pollen that can be blown about in the air; "the furniture was covered with dust")
debris, dust, junk, rubble, detritus -- (the remains of something that has been destroyed or broken up)
dust -- (free microscopic particles of solid material; "astronomers say that the empty space between planets actually contains measurable amounts of dust")

Sensuri posibile cuvant enter :
enter, come in, get into, get in, go into, go in, move into -- (to come or go into: "the ship entered an area of shallow marshes..")

Sensuri posibile cuvant smell :
smell -- (inhale the odor of; perceive by the olfactory sense)
Sensuri posibile cuvant hallway :
hallway, hall -- (an interior passage or corridor onto which rooms open; "the elevators were at the end of the hall")
Sensuri posibile cuvant cabbage :
cabbage, chou -- (any of various types of cabbage)
cabbage, cultivated cabbage, Brassica oleracea -- (any of various cultivars of the genus Brassica oleracea grown for their edible leaves or flowers)
Sensuri posibile cuvant mat :
mat -- (a thick flat pad used as a floor covering)
mat, matting -- (a border or background for a picture)
mat -- (a piece of thick padding up on the floor for gymnastic sports)
place mat, mat -- (table linen for an individual place setting)
Sensuri posibile cuvant rag :
rag, shred, tag, tag end, tatter -- (a small piece of cloth)
tabloid, rag, sheet -- (newspaper with half-size pages)

Sensuri posibile cuvant have :
have -- (have a personal or business relationship with someone; "have a postdoc"; "have an assistant"; "have a lover")
Sensuri posibile cuvant poster :
post horse, poster -- (a horse kept at an inn or post house for use by mail carriers or for rent to travelers)
Sensuri posibile cuvant display :
display, presentation -- (a visual representation of something)
display -- (a device that represents information in visual form)
Sensuri posibile cuvant end :
end -- (either extremity of something that has length: "the end of the pier"; "she knotted the end of the thread"; "they had reached the end of the road")
end -- (the surface at either extremity of a three-dimensional object: "one end of the box was marked `This side up'")
end -- ((football) the person who plays at one end of the line of scrimmage; "the end managed to hold onto the pass")
end -- (a boundary marking the extremities of something: "the end of town")
end -- (one of two places from which people are communicating to each other; "the phone rang at the other end" or "both ends wrote at the same time")
end, remainder, remnant, scrap, oddment -- (a piece of cloth that is left over after the rest has been used or sold)
end -- ((football) the person who plays at one end of the line of scrimmage; "the end managed to hold onto the pass")
Sensuri posibile cuvant wall :
wall -- (an architectural partition with a height and length greater than its thickness; used to divide or enclose an area or to support another structure; "the south wall had a small window"; "the walls were covered with pictures")
wall -- (anything that suggests a wall in structure or effect; "a wall of water"; "a wall of smoke"; "a wall of prejudice")
wall, paries -- ((anatomy) a layer (a lining or membrane) that encloses a structure; "stomach walls")
wall -- (a masonry fence (as around an estate or garden); "the wall followed the road"; "he ducked behind the garden wall and waited")
rampart, bulwark, wall -- (an embankment built around a space for defensive purposes; "they stormed the ramparts of the city"; "they blew the trumpet and the walls came tumbling down")
rampart, bulwark, wall -- (an embankment built around a space for defensive purposes; "they stormed the ramparts of the city"; "they blew the trumpet and the walls came tumbling down")

Sensuri posibile cuvant depict :
picture, depict, show -- (show in, or as in, a picture; "This scene depicts country life")

portray, depict, limn -- (make a portrait of: "showing society what it looked like..portraying..its ugliness and its beauties..")
Sensuri posibile cuvant face :
expression, look, aspect, facial expression, face -- (the expression on a person's face; "a sad expression"; "a look of triumph"; "an angry face")
face -- (the general outward appearance of something; "the face of the city is changing")
grimace, face -- (a contorted facial expression; "she made a grimace at the prospect")
font, fount, typeface, face -- (a specific size and style of type within a type family)
boldness, effrontery, nerve, brass, face, cheek -- (impudent aggressiveness; "I couldn't believe her boldness"; "he had the effrontery to question my honesty")
Sensuri posibile cuvant metre :
meter, metre, m -- (the basic unit of length adopted under the System International d'Unites (approximately 1.094 yards))
meter, metre, m -- (the basic unit of length adopted under the System International d'Unites (approximately 1.094 yards))
Sensuri posibile cuvant face :
face, human face -- (the front of the head from the forehead to the chin and ear to ear; "he washed his face"; "I wish I had seen the look on his face when he got the news")
face -- (the striking or working surface of an implement)
face -- ((synecdoche) a part of a person is used to refer to a person; "he looked out at a roomful of faces"; "when he returned to work he met many new faces")
side, face -- (a surface forming part of the outside of an object; "he examined all sides of the crystal"; "dew dripped from the face of the leaf")
face -- (the part of an animal corresponding to the human face)
face -- (the side upon which the use of a thing depends (usually the most prominent surface of an object); "he dealt the cards face down")
face -- (a vertical surface of a building or cliff)
face -- ((synecdoche) a part of a person is used to refer to a person; "he looked out at a roomful of faces"; "when he returned to work he met many new faces")
Sensuri posibile cuvant man :
man, adult male -- (an adult male person (as opposed to a woman); "there were two women and six men on the bus")
serviceman, military man, man, military personnel -- (someone who serves in the armed forces; "two men stood sentry duty")
man -- (the generic use of the word to refer to any human being; "it was every man for himself")
homo, man, human being, human -- (any living or extinct member of the family Hominidae)
man -- (a male subordinate; "the chief stationed two men outside the building"; "he awaited word from his man in Havana")
man -- (an adult male person who has a manly character (virile and courageous competent); "the army will make a man of you")
man -- ((informal) a male person who plays a significant role (husband or lover or boyfriend) in the life of a particular woman; "she takes good care of her man")
valet, valet de chambre, gentleman, gentleman's gentleman, man -- (a manservant who acts as a personal attendant to his employer; "Jeeves was Bertie Wooster's man")
Man, Isle of Man -- (one of the British Isles in the Irish Sea)
man, piece -- (a small object used in playing certain board games; "he taught me to set up the men on the chess board"; "he sacrificed a piece to get a strategic advantage")
man, adult male -- (an adult male person (as opposed to a woman); "there were two women and six men on the bus")
valet, valet de chambre, gentleman, gentleman's gentleman, man -- (a manservant who acts as a personal attendant to his employer; "Jeeves was Bertie Wooster's man")
Sensuri posibile cuvant moustache :
mustache, moustache -- (an unshaved growth of hair on the upper lip; "he looked younger after he shaved off his mustache")
Sensuri posibile cuvant feature :
feature, lineament -- (the characteristics parts of a person's face: eyes and nose and mouth and chin; "an expression of pleasure crossed his features"; "his lineaments were very regular")
feature -- (an article of merchandise that is displayed or advertised more than other articles)
feature, feature article -- (a special or prominent article in a newspaper or magazine; "they ran a feature on retirement planning")

Sensuri posibile cuvant make :
make -- (carry out or commit; "make a mistake"; "commit a faux-pas")
Sensuri posibile cuvant stair :
step, stair -- (a place to rest the foot while ascending or descending a stairway; "he paused on the bottom step")

Sensuri posibile cuvant be :
embody, be, personify -- (represent, as of a character on stage; "Derek Jacobi was Hamlet")
Sensuri posibile cuvant use :
use -- (a particular service; "he put his knowledge to good use"; "patrons have their uses")
function, purpose, role, use -- (what something is used for; "the function of an auger is to bore holes"; "ballet is beautiful but what use is it?")
Sensuri posibile cuvant try :
try, seek, attempt, essay, assay -- (make an effort or attempt; "He tried to shake off his fears"; "The infant had essayed a few wobbly steps"; "The police attempted to stop the thief"; "He sought to improve himself"; "She always seeks to do good in the world")
Sensuri posibile cuvant lift :
ski tow, ski lift, lift -- (carries skiers up a hill)
elevator, lift -- (a platform or cage that is raised and lowered mechanically in a vertical shaft in order to move people from one floor to another in a building)

Sensuri posibile cuvant work :
work -- (work one's way through a problem or task; "Start from the bottom and work towards the top")
cover, treat, handle, work, plow, deal, address -- (deal with verbally or in some form of artistic expression; "This book deals with incest"; "The course covered all of Western Civilization")
Sensuri posibile cuvant be :
embody, be, personify -- (represent, as of a character on stage; "Derek Jacobi was Hamlet")
Sensuri posibile cuvant be :
embody, be, personify -- (represent, as of a character on stage; "Derek Jacobi was Hamlet")
Sensuri posibile cuvant current :
current, stream -- (a steady flow (usually from natural causes); "the raft floated downstream on the current"; "he felt a stream of air")
Sensuri posibile cuvant hour :
hour, hr, 60 minutes -- (a period of time equal to 1/24th of a day; "the job will take more than an hour")
hour -- (a special and memorable period; "it was their finest hour")
hour, minute -- (distance measured by the time taken to cover it; "we live an hour from the airport"; "its just 10 minutes away")
Sensuri posibile cuvant daylight :
day, daytime, daylight -- (the time after sunrise and before sunset while it is light outside; "the dawn turned night into day"; "it is easier to make the repairs in the daytime")
Sensuri posibile cuvant time :
time, clip -- (an instance or single occasion for some event; "This time he succeeded"; "He called four times"; "he could do ten at a clip")
time -- (a person's experience on a particular occasion; "he had a time holding back the tears" or "they had a good time together")

Sensuri posibile cuvant be :
embody, be, personify -- (represent, as of a character on stage; "Derek Jacobi was Hamlet")
Sensuri posibile cuvant part :
region, part -- (the extended spatial location of something; "the farming regions of France"; "religions in all parts of the world"; "regions of outer space")
part, portion -- (something less than the whole of a human artifact: "the rear part of the house"; "glue the two parts together")
part, piece -- (a portion of a natural object; "they analyzed the river into three parts"; "he needed a piece of granite")
part, voice -- (the melody carried by a particular voice or instrument in polyphonic music; "he tried to sing the tenor part")
part -- (a line where the hair is parted; "his part was right in the middle")

Sensuri posibile cuvant drive :
drive -- (a mechanism by which force or power is transmitted in a machine; "a variable speed drive permitted operation through a range of speeds")
driveway, drive, private road -- (a road leading up to a private house; "they parked in the driveway")
drive -- ((computer science) a device that writes data onto or reads data from a storage medium)
drive, parkway -- (a wide scenic road planted with trees; "the riverside drive offers many exciting scenic views")
Sensuri posibile cuvant economy :
economy, economic system -- (the system of production and distribution and consumption)
Sensuri posibile cuvant preparation :
formulation, preparation -- (a substance prepared according to a formula)
Sensuri posibile cuvant week :
week, hebdomad -- (any period of seven consecutive days; "it rained for a week")
week, calendar week -- (a period of seven consecutive days starting on Sunday)
workweek, week -- (hours or days of work in a calendar week; "they worked a 40-hour week")
Sensuri posibile cuvant hate :
hate, hatred -- (the emotion of hate; a feeling of dislike so strong that it demands action)

În general sensurile selectate pentru cuvinte sunt cele corecte. Există totuşi câteva cazuri în care alegerea este greşită. Pentru substantivul *wind* de exemplu, este ales synset-ul <*abstraction*>, deşi un utilizator uman ar fi optat pentru un sens hiponim pentru <*phenomenon*>. Dacă privim datele din tabela de co-ocurență însă, acesta apare ca fiind puțin probabil. Această problemă nu este singulară şi este normală pentru dimensiunea redusă a corpusului pe care s-a făcut învățarea. Soluția este construirea unui corpus de învățare cât mai reprezentativ pentru limba engleză pentru care să fie construită tabela de co-ocurențe.

Capitolul 6
Concluzii

Modelul propus foloseşte atât informaţii dintr-o bază de cunoştinţe cât şi date obţinute din contextul local al cuvintelor conţinute intr-un corpus de învăţare pentru a dezambiguiza cuvintele dintr-un text. Astfel, modulul de învăţare parcurge corpusul primit ca intrare, extrage câte o propoziţie şi pentru substantivele şi verbele conţinute în aceasta construieşte arborii de hipernime. În următorul pas, aceştia sunt simplificaţi şi sunt extrase synset-urile de vârf. Apoi, pentru două cuvinte aflate în relaţie sintactică, sunt actualizate într-o tabelă de co-ocurenţe apariţiile synset-urilor lor de vârf. După terminarea acestui proces, în tabela de co-ocurenţe se vor găsi, pentru un synset de vârf oarecare, numărul total de apariţii, numărul de apariţii referind sintactic un alt synset, precum şi numărul de apariţii ca fiind referit de alte synset-uri.

Datele adunate în modulul de învăţare sunt folosite în modulul de dezambiguizare pentru calcularea probabilităţii ca două synset-uri să fie alese ca sensuri pentru două cuvinte aflate în relaţie sintactică. Dezambiguizarea are loc tot la nivelul de propoziţie, căutându-se o configuraţie de sensuri pentru cuvintele acesteia, care să obţină un scor maxim, calculat ca sumă a scorurilor între toate perechile părinte – fiu (în relaţie sintactică de subordonare). Cel mai important segment al acestui modul este modul în care se calculează scorul între două synset-uri candidate pentru o astfel de pereche. În scopul de a găsi o funcţie ale cărei alegeri să se apropie cât mai mult de cele umane, am încercat diferite modalităţi de calcul, pe care le-am testat pe câteva exemple de perechi de cuvinte selectate în acest scop. Concluzia la care am ajuns este că pentru alegerea sensului unui cuvânt trebuie luate în considerare atât date legate de synset-ul candidat, cât şi date legate de celelalte sensuri ale cuvântului. Combinarea lor, însă, necesită o reglare fină, şi poate diferi de la un cuvânt la altul. Din experimentele efectuate, am concluzionat că importanţa cea mai mare o are probabilitatea calculată în funcţie de numărul total de apariţii ale synset-ului şi numărul de apariţii referind synset-ul ales pentru părinte, dar, în funcţie de distribuţia probabilităţilor sensurilor cuvântului, acestea trebuie să poată, în unele cazuri, influenţa puternic rezultatul funcţiei.

Modelul implementat a dovedit că vecinătatea sintactică a unui cuvânt, combinată cu date referitoare la hipernimele sensurilor acestuia, constituie un indiciu puternic care poate fi folosit cu succes pentru dezambiguizare. Desigur, acesta nu poate constitui singurul criteriu. Întotdeauna vor exista cazuri speciale care nu au fost acoperite de corpusul pe care s-a făcut învăţarea sau care necesită metode specifice de dezambiguizare.

Desigur, principalul avantaj decurge din folosirea WordNet – ului ca bază de cunoştinţe. Acesta oferă mai mult decât simpla descriere a sensurilor unui cuvânt. Toate sensurile sunt legate între ele prin relaţii semantice şi lexicale care pot fi folosite pentru stabilirea vecinătăţii semantice a cuvântului. WordNet – ul nu este, totuşi, imun la problemele generale ale dicţionarelor, cum ar fi numărul de sensuri pe care le poate avea un cuvânt. Acestea sunt probleme la care chiar lexicografii umani nu se pot întotdeauna pune de acord.

Modelul propus poate fi, bineînţeles, îmbunătăţit prin experimente. Astfel, se pot introduce în funcţiile de scor date referitoare la numărul de apariţii ale unui synset în dreapta relaţiilor sintactice, care se află de altfel calculat în tabela de co-ocurenţe. De asemenea, se pot da, la construcţia tabelei, importanţe diferite diverselor relaţii sintactice sau părţilor de propoziţie referite. Un alt punct în care se poate face o reglare a rezultatelor este modul de

calcul al scorului total pe propoziție, la fel, dându-se ponderi diferite scorurilor dintre synset-uri, în funcție de nivelul pe care se găsesc în arborele sintactic, partea de vorbire, partea de propoziție sau relația sintactică în care se află.

O îmbunătățire sigură se poate obține prin combinarea sa cu alte modele de dezambiguizare. Desigur problemele care apar în acest caz sunt multiple: realizarea interoperabilității modelelor, folosirea unei baze de cunoştințe comună şi, nu în ultimul rând, stabilirea unei priorități între modele în momentul în care rezultatele furnizate de acestea nu converg.

O problemă avută în vedere în viitorul apropiat este cea a co-locațiilor (cuvinte care apar împreună şi au un sens cumulat – de exemplu *give ear, give up, pay attention*). Acestea trebuie identificate înainte de aplicarea modelului, atât la învățare, cât şi la dezambiguizare şi tratate ca un tot unitar.

De asemenea, avem în vedere prelucrarea unui corpus de test pentru modele de dezambiguizare pentru a fi compatibil cu tipul de intrare folosit de noi (modelul are nevoie de etichetare morfologică şi sintactică), pentru a obține rezultate comparabile cu cele obținute de alte modele. Acest mod de testare va oferi şi un sistem mai eficient de testare a diferitelor funcții de scor.

În altă ordine de idei, recurgerea, pentru învățare, la etichetări morfologice şi sintactice, mult mai uşor de obținut decât etichetările semantice, rezolvă, cel puțin parțial, problema „*acquisition bottleneck*" întâlnită la modelele de dezambiguizare *corpus-based*.

Găsirea de formule matematice care să descrie complet limbajul natural este imposibilă. Se pot, totuşi, găsi reprezentări bazate pe statistica unui corpus destul de mare care să fie aplicabile într-un număr cât mai larg de cazuri. Ideea este micşorarea numărului cazurilor speciale, care pot fi tratate apoi separat.

Bibliografie

1. Anderson, John Robert (1976). *Language,Memory, and Thought.* Lawrence Erlbaum and Associates, Hillsdale, New Jersey.
2. Anderson, John Robert (1983). "A Spreading Activation Theory of Memory." *Journal ofVerbal Learning and Verbal Behavior,* 22(3), 261-95.
3. Bar-Hillel, Yehoshua (1960). "Automatic Translation of Languages." In Alt, Franz; Booth, A. Donald and Meagher, R. E. (Eds), *Advances in Computers,* Academic Press, New York.
4. Briscoe, Edward J. (1991). "Lexical issues in natural language processing." In Klein, Ewan H. and Veltman, Frank (Eds.). *Natural Language and Speech. [Proceedings of the Symposium on Natural Language and Speech,* 26-27 November 1991, Brussels, Belgium.] Springer-Verlag, Berlin, 39-68.
5. Buitelaar, Paul (1997). "A lexicon for under-specified semantic tagging." ACL-SIGLEX Workshop "Tagging Text with Lexical Semantics: Why, What, and How?" April 4-5, 1997, Washington, D.C., 25-33.
6. Collins, Allan M. and Loftus, Elisabeth F. (1975). "A spreading activation theory of semantic processing." *Psychological Re-view,* 82(6), 407-428.
7. Dagan, Ido; Marcus, Shaul; and Markovitch, Shaul (1993). "Contextual word similarity and estimation from sparse data." *Proceedings of the 31st Annual Meeting of the Association for Computational Linguistics,* 22-26 June 1993, Columbus, Ohio.
8. Dagan, Ido; Peireira, Fernando and Lee, Lilian (1994). "Similarity-based estimation of word cooccurrence probabilities." *Proceedings of the 32nd Annual Meeting of the Association for Computational Linguistics,* Las Cruces, New Mexico, 272-278.
9. Grishman, Ralph and Sterling, John (1993). "Smoothing of automatically generated selectional constraints." *Human Language Technology.* Morgan Kaufmann, 254-259.
10. Grishman, Ralph; MacLeod, Catherine; and Meyers, Adam (1994). "COMLEX syntax: Building a computational lexicon." *Proceedings of the 15th International Conference on Computational Linguistics,COLING'94,* 5-9 August 1994, Kyoto, Ja-pan, 268-272.
11. Hayes, Philip J. (1976). *A process to implement some word-sense disambiguation.* Working paper 23. Institut pour les Etudes Sémantiques et Cognitives, Université de Genève. Hayes, Philip J. (1977a). On semantic nets, frames and associations. *Proceedings of the 5th International. Joint Conference on Artificial Intelligence,* Cambridge, Massachusetts, 99-107.
12. Hayes, Philip J. (1977b). *Some association-based techniques for lexical disambiguation by machine.* Doctoral dissertation, Département de Mathématiques, Ecole Polytechnique Fédérale de Lausanne.
13. Hayes, Philip J. (1978). *Mapping input into schemas.* Technical report 29, Department of Computer Science, University of Rochester.
14. Hearst, Marti A. (1991). "Noun homograph disambiguation using local context in large corpora." *Proceedings of the 7th Annual Conf. of the University of Waterloo Centre for the New OED and Text Research,* Oxford, United Kingdom, 1-19.
15. Ide, Nancy and Véronis, Jean (1990). "Very large neural networks for word sense disambiguation." *Proceedings of the 9th European Conference on Artificial Intelligence, ECAI'90*, Stockholm, 366-368.
16. Ide, Nancy and Véronis, Jean (1993). "Refining taxonomies extracted from machine-readable dictionaries." In Hockey, Susan and Ide, Nancy (Eds.) Research in Humanities Computing II, Oxford University Press, 145-59.
17. Ide, Nancy and Véronis, Jean (1998). "Word sense disambiguation – The state of the art.", Computational Linguistics, 24(1).
18. Jelinek, Frederick and Mercer, Robert L. (1985). "Probability distribution estimation from sparse data." *IBM Technical Disclosure Bulletin,* 28, 2591-2594.
19. Kaplan, Abraham (1950). "An experimental study of ambiguity and context." Mimeo-graphed, 18pp, November 1950. [Published as: Kaplan, Abraham (1955). "An experi-mental study of ambiguity and context." *Mechanical Translation,* 2(2), 39-46.]

20. Karov, Yael and Edelman, Shimon (1998). “Similarity-based word sense disambiguation”. Computational Linguistics, 24(1).

21. Kilgarriff, Adam (1992). Polysemy. Ph. D. Thesis. University of Sussex, United Kingdom.

22. Kilgarriff, Adam (1993). “Dictionary word sense distinctions: An enquiry into their nature.” Computers and the Humanities, 26, 365-387.

23. Leacock, Claudia; Towell, Geoffrey; and Voorhees, Ellen (1993). “Corpus-based statistical sense resolution.” Proceedings of the ARPA Human Language Technology Worskshop San Francisco, Morgan Kaufman.

24. Leacock, Claudia; Miller, George A.; and Chodorow, Martin (1998). “Using corpus statistics and WordNet relations for sense identification”.

25. Lenat, Douglas B. and Guha, Ramanathan V., (1990). *Building large knowledge-based systems.* Addison-Wesley, Reading, Massachusetts.

26. Lesk, Michael (1986). “Automated Sense Disambiguation Using Machine-readable Dictionaries: How to Tell a Pine Cone from an Ice Cream Cone.” *Proceedings of the 1986 SIGDOC Conference,* Toronto, Canada, June 1986, 24-26.

27. Masterman, Margaret (1961). “Semantic message detection for machine translation, using an interlingua.” *1961 International Conference on Machine Translation of Languages and Applied Language Analysis,* Her Maj-esty’s Stationery Office, London, 1962, 437-475.

28. McClelland, James L. and Rumelhart, David E. (1981). “An interactive activation of context effects in letter perception: part 1. An account of basic findings.” *Psychological review,* 88, 375-407.

29. Miller, George A.; Beckwith, Richard T. Fellbaum, Christiane D.; Gross, Derek; and Miller, Katherine J. (1990). “WordNet: An on-line lexical database.” International Journal of Lexicography, 3(4), 235-244.

30. Miller, George A. and Charles, Walter G. (1991). “Contextual correlates of semantic similarity.” Language and Cognitive Processes, 6(1), 1-28.

31. Pustejovsky, James (1995). *The Generative Lexicon.* The MIT Press, Cambridge, Massachusetts.

32. Pustejovsky, James; Boguraev, Bran; and Johnston, Michael (1995). “A core lexical engine: The contextual determination of word sense.” Technical report, Department of Computer Science, Brandeis University.

33. Quillian, M. Ross (1961). “A design for an understanding machine.” Communication presented at the colloquium *Semantic problems in natural language.* September 1961. King’s College, Cambridge University, Cambridge, United Kingdom.

34. Quillian, M. Ross (1968). “Semantic memory.” In Minsky, M. (Ed.), *Semantic Information Processing,* MIT Press, 227-270.

35. Reifler, Erwin (1955). The mechanical determination of meaning. In Locke, William N. and Booth, A. Donald (Eds.), M*achine translation of languages.* John Wiley & Sons, New York, 136-164.

36. Richens, Richard H. (1958). “Interlingual machine translation.” *Computer Journal,* 1(3), 144-47

37. Schütze, Hinrich (1993). “Word space.” In Hanson, Stephen J.; Cowan, Jack D.; and Giles, C. Lee (Eds.) Advances in Neural Information Processing Systems 5, Morgan Kauffman, San Mateo, California, 5, 895-902.

38. Schütze, Hinrich (1998). “Automatic word sense discrimination”, Computational Linguistics, 24(1).

39. Schütze, Hinrich and Pedersen, Jan (1995). “Information retrieval based on word senses.” Proceedings of SDAIR'95. April 1995, Las Vegas, Nevada.31.

40. Véronis, Jean and Ide, Nancy (1990). “Word Sense Disambiguation with Very Large Neural Networks Extracted from Machine Readable Dictionaries.” 13th International Conference on Computational Linguistics, COLING'90, Helsinki, Finland, vol. 2, 389-394.

41. Weaver, Warren (1949). *Translation.* Mimeo-graphed, 12 pp., July 15, 1949. Reprinted in Locke, William N. and Booth, A. Donald (1955) (Eds.), *Machine translation of languages.*John Wiley & Sons, New York, 15-23.

42. Wilks, Yorick A.; Fass, Dan; Guo, Cheng-Ming; MacDonald, James E.; Plate, Tony; and Slator, Brian A. (1990). “Providing Machine Tractable Dictionary Tools.” In Pustejovsky, James (Ed.), *Semantics and the Lexicon.* MIT Press, Cambridge, Massachusetts.

43. Yarowsky, David (1992). “Word sense disambiguation using statistical models of Roget's categories trained on large corpora.” Proceedings of the 14th International Conference on Computational Linguistics, COLING'92, 23-28 August, Nantes, France, 454-460.

Pentru accesarea datelor conținute în WordNet am folosit biblioteca de clase JAVA creată de Oliver Steele, steele@cs.brandeis.edu, corectată şi îmbunătățită de Mihai Lupu, mihai@infoiasi.ro .