ACADEMIA ROMÂNĂ



Institutul de Cercetări pentru Inteligență Artificială

**TEZĂ DE DOCTORAT**

**Contributions to semantic processing of texts; Identification of entities and relations between textual units; Case study on Romanian language**

**( Rezumat RO )**

|  |  |
| --- | --- |
| **Conducător științific**:  Prof. Dr. Acad. DAN TUFIȘ | **Doctorand:**  Ing. VASILE-FLORIAN PĂIȘ |

BUCUREȘTI, 2019

**Contents**

[1. Introducere 1](#_Toc17898370)

[2. Procesarea primară a documentelor 2](#_Toc17898371)

[3. Detecția entităților cu nume 4](#_Toc17898372)

[4. Extragere de relații din text 6](#_Toc17898373)

[5. Contribuții personale 7](#_Toc17898374)

[6. Referințe 10](#_Toc17898375)

## 1. Introducere

Pentru a putea înțelege mai bine problematica asociată extragerii de entități cu nume din texte, o serie de definiții sunt necesare. Astfel, dicționarul Merriam-Webster [1] definește un obiect ca fiind ”ceva material care poate fi perceput cu simțurile”. De asemenea, același dicționar definește o entitate drept ”ceva care există de sine stătător; ceva care este separat de alte lucruri; existența unui lucru în contrast cu atributele sale”. În acest context, entitățile cu nume sunt utilizate pentru a reprezenta obiecte din lumea reală (cum ar fi persoane, locații, organizații, produse, etc.) care pot fi identificate utilizând nume proprii.

Pe baza acestor definiții devine clar faptul că o parte centrală în regăsirea entităților cu nume constă în identificarea numelor proprii în text, întrucât acestea sunt utilizate pentru persoane, organizații și locații. Cu toate acestea, o entitate poate fi introdusă în text și prin utilizarea unei expresii. De exemplu, având in vedere faptul că Dl. Klaus Iohannis este actualul președinte al României (la momentul scrierii acestei teze), este evident pentru orice cititor uman că următoarele secvențe de cuvinte ”Klaus Iohannis” și ”președintele României” sunt echivalente și fac referință la aceeași entitate. Cu toate acestea pentru un algoritm de inteligență artificială nu este la fel de ușor să identifice cele două expresii ca fiind referitoare la aceeași entitate cu nume. Un cititor uman nu utilizează doar informație locală, prezentă în textul curent, ci utilizează și cunoștințe dobândite anterior despre subiectul respectiv.

Având in vedere definițiile de mai sus, în special cea referitoare la faptul că o entitate se referă la ”existența unui lucru în contrast cu atributele sale”, este sugerat faptul că un algoritm nu trebuie să caute doar numele proprii și expresiile asociate, dar și posibile atribute care pot conduce la o entitate cu nume.

Identificarea entităților într-un text poate fi considerată doar un prim pas. În continuare, este necesară identificarea tipului entității. Acest proces este cunoscut sub denumirea de ”dezambiguizare”. Potrivit definiției date de dicționarul Merriam-Webster, dezambiguizarea este procesul care urmărește ”stabilirea unei interpretări unice din punct de vedere semantic sau gramatical” pentru a anumită entitate cu nume.

Scopul acestei teze este dezvoltarea unui sistem pentru recunoașterea entităților cu nume (persoane, organizații, locații, timp) și extragerea relațiilor dintre acestea, special construit pentru texte în limba română. În acest context, diferite tehnici de procesare primară a textului sunt prezentate în capitolul 2, urmate de metode pentru identificarea entităților cu nume în capitolul 3 și extragerea relațiilor în capitolul 4. Având în vedere faptul că una dintre resursele cheie utilizate este corpusul CoRoLa, în capitolul 5 sunt prezentate contribuțiile personale la acest proiect. Acestea sunt reprezentate de construirea unei forme de reprezentare distribuțională a cuvintelor bazată pe CoRoLa (capitolul 5.2) care permite codificarea textului la intrarea sistemului de recunoaștere a entităților. De asemenea, în capitolul 5.1 este prezentată platforma realizată pentru partea vorbită a corpusului CoRoLa care a fost utilizată pentru a interacționa cu acesta. În capitolul 6 este descris sistemul final care a fost implementat.

Identificarea entităților combinată cu extragerea relațiilor formează baza oricărui sistem de extragere de informații. Scopul principal al unui asemenea sistem este organizarea informației textuale disponibile într-o manieră care să permită căutarea și realizarea de inferențe de către algoritmi. Prezintă, astfel, o manieră de a construi date structurate din documente digitale nestructurate sau parțial structurate.

Ca exemplu pot fi considerate documente legale sau comerciale, cum ar fi contracte de vânzare cumpărare, moșteniri, înregistrări de companii. O posibilă problemă este dată de încercarea de identificare a proprietarilor unei clădiri sau a proprietăților deținute de o persoană. Mai mult, această informație poate fi extinsă cu lista proprietăților la care o persoană este îndreptățită, într-un fel sau altul, prin intermediul companiilor pe care le deține. Acest gen de interogări avansate nu pot fi realizate direct pe textul neprocesat. În schimb aceste interogări pot fi rezolvate cu ușurință pe date structurate corespunzător, cum este cazul reprezentărilor sub formă de graf (cu noduri și relații).

Un sistem de extragere de informații va începe prin extragerea entităților cu nume și a relațiilor dintre acestea din fiecare document, construind câte un graf asociat fiecărui document în parte. Ulterior, toate documentele sunt unificate într-o bază de date care poate fi interogată prin expresii avansate similare celor din exemplul anterior. Desigur, pentru unificarea mai multor grafuri asociate documentelor, un pas adițional este necesar, constând în dezambiguizarea entităților între documente.

O altă aplicabilitate a extragerii de entități se regăsește în domeniul rezumării automate a textelor și identificării de cuvinte cheie. Fiecare entitate cu nume prezentă în text poate deveni un cuvânt cheie asociat documentului. În plus, parcurgând graful generat și generând mici fraze asociate relațiilor, devine posibilă construirea unui rezumat al documentului bazat pe entitățile cu nume. Desigur, algoritmii de sumarizare actuali sunt mai complecși, dar entitățile cu nume continuă să joace un rol cheie [22],[23].

## 2. Procesarea primară a documentelor

În acest capitol sunt prezentate diferite operații primare efectuate asupra textelor în vederea îmbogățirii acestora înaintea unor analize mai avansate, cum ar fi recunoașterea de entități. Aceste operații includ: segmentare, stemizare, lematizare, adnotare cu părți de vorbire.

Uneltele pentru procesarea limbajului natural necesită diferite nivele de **segmentare** a documentelor. Acesta este procesul prin care un text de dimensiuni mari este împărțit în părți de dimensiuni mai mici, utilizabile pentru scopul aplicației.

În funcție de dimensiunea segmentelor, cele mai utilizabile segmentări sunt: la nivel de paragraf, la nivel de frază, la nivel de ”token”. Documente mai complexe, cum ar fi romanele, pot fi segmentate și la nivel de capitol.

Segmentarea la nivel de ”token” reprezintă cel mai de jos nivel util pentru aplicațiile de recunoaștere de entități. Un token este în general asociat unui cuvânt. Cu toate acestea sunt situații în care tokenul este o grupare de mai multe cuvinte. Din punctul de vedere al aplicației un token este un element indivizibil al documentului, care va fi utilizat ca atare, fără alte operațiuni de segmentare ulterioare.

O problemă cu un impact direct asupra algoritmilor de detecție de entități este reprezentată de prezența abrevierilor. De exemplu, același partid politic românesc poate fi abreviat ”PNL” sau ”P.N.L.”. Evident, pentru a detecta cu succes entitatea reprezentată în acest exemplu, operațiunea de segmentare ar trebui sa producă un singur element, indiferent de forma prezentă în text.

**Stemizarea** reprezintă procesul prin care se obține rădăcina asociată unui cuvânt. Motivul principal pentru utilizarea acestei tehnici este dorința de grupare a cuvintelor având aceeași rădăcină, indiferent de forma prezentă în text. În general rădăcinile nu sunt cuvinte propriu-zise ci mai degrabă părți de cuvinte.

Un proces similar este **lematizarea**. Acesta presupune utilizarea „lemei” unui cuvânt în același scop de a grupa cuvintele indiferent de forma în care acestea apar în document [11]. ”Lemma” unui cuvânt mai este cunoscută și sub denumirea de formă de dicționar a acestuia. În vederea detectării corecte a lemei unui cuvânt este necesară înțelegerea semnificației cuvântului în contextul în care apare.

Adnotarea cu **părți de vorbire** [11] presupune asocierea de etichete morfo-lexicale fiecărui token. Astfel de adnotări sunt extrem de utile în antrenarea unui sistem pentru extragerea entităților cu nume.

Construirea unui sistem în vederea adnotării cu părți de vorbire este asemănătoare cu construirea unui sistem pentru extragerea de entități. Astfel, nu este suficientă prezența unei liste cu părțile de vorbire posibile asociate unui cuvânt deoarece detecția propriu-zisă este dependentă de contextul în care apare cuvântul respectiv.

Din punct de vedere istoric, una dintre primele tentative pentru adnotarea cu părți de vorbire este reprezentată de scrierea manuală a unui set de reguli de adnotare. Un exemplu de astfel de sistem este prezentat în [37].

Abordări mai recente folosesc proprietăți statistice extrase din documente adnotate manual. Aceste metode utilizează ulterior modele Markov (HMM) pentru decizia asupra etichetei asociate. Un astfel de sistem este descris in [88].

Adnotatoarele actuale utilizează în general rețele neuronale artificiale pentru a construi automat un model de limbă care va fi apoi utilizat pentru adnotare. Un astfel de sistem este prezentat în [89].

Un sistem pentru etichetarea cu părți de vorbire pentru limba română, utilizând o abordare hibridă, format dintr-un model statistic combinat cu un set de reguli, este descris în [36]. Sistemul utilizează și un dicționar de părți de vorbire cu diferitele etichete care pot fi asociate unui cuvânt. Acest aspect ajută la reducerea complexității găsirii etichetei corecte.

Serviciul TTL [56] oferă un sistem de adnotare cu părți de vorbire utilizând etichete MSD. Acestea pot fi ulterior convertite într-un set simplificat de etichete numit CTAG. Mai multe informații despre aceste seturi de etichete și despre serviciul TTL sunt disponibile în [85],[86].

Limba română scrisă utilizează 31 de caractere distincte, litere mici, și 31 de caractere corespunzătoare majuscule. Acestea conțin 26 de caractere latine (cunoscute drept caractere ASCII) și 5 caractere adiționale cu diacritice.

Există situații în care într-un text nu sunt utilizate caracterele românești corecte. Uneori sunt utilizate caractere care seamănă din punct de vedere vizual cu caracterele românești. Posibile cauze ale acestor erori includ: necunoașterea caracterelor românești corecte, utilizarea unor fonturi necorespunzătoare, erori provenite din recunoașterea automată a caracterelor scanate.

Deși pentru un cititor uman caracterele asemănătoare nu pun probleme de înțelegere a textului, pentru un calculator caracterele au reprezentări interne diferite, ceea ce conduce la cuvinte diferite. De aceea, o **corecție a caracterelor** este necesară pentru a face cuvintele să devină inteligibile din punctul de vedere al algoritmilor.

O altă problemă întâlnită la procesarea textelor în limba română este absența parțială sau completă a diacriticelor. În acest caz se pot utiliza tehnici de **restaurare a diacriticelor** [90],[91],[92],[93] pentru a corecta textul.

## 3. Detecția entităților cu nume

Acest capitol al tezei de doctorat prezintă metode pentru detectarea entităților cu nume în text. Sunt incluse și resurse lingvistice disponibile pentru limba română cum ar fi liste de nume, expresii regulare, convenții de scriere. De asemenea, sunt prezentate și formatele de fișiere acceptate și produse de sistemul implementat.

Cea mai simplă metodă pentru detecția entităților în text se bazează pe **liste de nume** care sunt apoi căutate în text. Această abordare poate funcționa în practică dacă se dorește identificarea unui număr redus de entități bine determinate. O abordare generală bazată pe astfel de liste nu poate funcționa deoarece, pe de o parte, numărul de entități este foarte mare (ar fi suficient să ne gândim la numele tuturor persoanelor din lume) iar pe de altă parte tipul entității nu poate fi identificat corect în toate situațiile. Cu toate acestea listele de nume, cunoscute în literatura de specialitate în limba engleză și sub denumirea de resurse ”gazetteer”, ajută în sistemele mai avansate de recunoaștere de entități. Printre resursele de acest tip utile pentru limba română se numără: SIRUTA [2], GNS [6], lista președinților [7], membrii parlamentului, primari, membri ai consiliilor locale. Toate acestea sunt informații publice.

O extensie a listelor simple de nume este reprezentată de listele formate din **expresii regulate**. Acestea permit identificarea de cuvinte aparținând entităților cu nume pe baza unor șabloane de căutare. Pot fi utilizate fie la nivel de token fie prin combinarea mai multor tokeni într-un șir de caractere prin utilizarea unui caracter de separare, cum ar fi caracterul ”spațiu”.

Expresiile regulate pot fi utilizate cu succes în detectarea unor tipuri speciale de entități cum ar fi: adrese de e-mail, date calendaristice, numere de telefon, identificatoare numerice, adrese IP, etc.

În textele scrise, numele proprii respectă anumite reguli cunoscute și sub denumirea de **convenții de scriere**. De exemplu, în limba română un nume începe mereu cu o majusculă iar în cazul organizațiilor pot fi utilizate abrevieri de forma ”PNL”, ”PSD”, ”Min. de Justiție”.

O abordare mai complexă pentru detecția de entități trebuie să ia în calcul și informațiile oferite de contextul în care apare entitatea respectivă. De aceea, operațiile de procesare primară trebuie să fie efectuate pe text înaintea aplicării algoritmului propriu-zis de detecție de entități. Într-o manieră similară unui adnotator cu părți de vorbire, un sistem pentru detecție de entități trebuie să poată învăța dintr-un set de fișiere adnotate manual și să construiască un model de limbă care să poată fi apoi aplicat pe texte noi.

Una dintre metodele bazate pe context este dată de utilizarea unui model de tip Conditional Random Field (CRF). Într-o astfel de implementare, se pot combina diferite caracteristici ale textului, reprezentate sub forma unor funcții, pentru a produce adnotările finale. Importanța fiecărei caracteristici este învățată automat prin utilizarea unor algoritmi de tip Gradient Descent [96].

Un pachet software de top implementând clasificare bazată pe CRF pentru recunoaștere de entități este Stanford Named Entity Recognizer (Stanford NER). Acesta este o aplicație Java care ”oferă o implementare generală pentru modele secvențiale de tip Conditional Random Field”, potrivit site-ului web oficial [79]. Mai multe detalii legate de implementare sunt disponibile în articolul [80].

Această metodă a fost aleasă pentru implementarea sistemului de recunoaștere de entități, parte a acestei teze.

Modele bazate pe rețele neuronale artificiale au fost utilizate cu succes în diferite activități legate de procesarea limbajului natural. Într-un articol publicat în 2016 [48], autorii utilizează rețele de tip LSTM multiple în vederea detectării de entități în documente clinice. Deși acest tip de documente utilizează în general un limbaj specific, acest demers arată faptul că rețelele neuronale de tip LSTM pot produce rezultate bune în detecția de entități. Cu toate acestea, este cunoscut faptul că tehnologiile de tip ”deep learning” se comportă bine doar pe corpusuri de mari dimensiuni. Din punctul de vedere al sistemului de recunoaștere de entități pe limba română (ca și pentru alte limbi) aceasta poate fi o problemă datorită lipsei unei astfel de resurse de dimensiune mare, adnotată cu entități. Acesta poate fi și unul din motivele pentru care unelte ”state-of-the-art”, cum este Stanford NER, continuă să utilizeze modele CRF pentru clasificare.

Activitatea din cadrul conferinței CoNLL-2002 a introdus un format de fișier pentru adnotarea entităților care conține câte un cuvânt pe linie, iar liniile goale reprezintă delimitarea frazelor. În plus fiecare linie conține și o etichetă care arată dacă respectivul cuvânt face sau nu parte dintr-o entitate. Această etichetă arată tipul entității. Cuvintele marcate cu ”O” nu fac parte din nici o entitate. Etichetele de forma B-XXX sunt folosite pentru a marca primul cuvânt dintr-o entitate de tip XXX, iar etichetele de forma I-XXX sunt folosite pentru restul cuvintelor din acea entitate. Fișierele de date conțin etichete pentru 4 tipuri de entități: persoane (PER), organizații (ORG), locații (LOC) și alte denumiri (MISC). Aceasta schemă de adnotare este o variantă a schemei IOB care a fost propusă de Ramshaw și Marcus (1995) [9].

Un alt tip de codificare este denumit IO (”Inside-Outside”, in limba română ”Interior-Exterior”). Acest format este similar cu cel IOB descris anterior, dar fără părțile de început ”B-” și ”I-”. Conversia din IOB în formatul IO este foarte simplă, eliminându-se pur și simplu începutul fiecărei etichete. Cu toate acestea, conversia inversă este posibilă doar dacă nu există entități adiacente de același tip. În acest caz, prima etichetă a unei entități este prefixată cu ”B-” iar restul sunt prefixate cu ”I-”.

Pentru sistemul implementat în cadrul acestei teze a fost ales un format similar CoNLL. Este un format de fișier bazat pe coloane multiple delimitate prin caractere ”tab”. Printre aceste coloane se regăsește una pentru ”token” și o coloană pentru eticheta asociată entității. Coloane adiționale sunt utilizate pentru diferite elemente utilizate în procesul de detecție, cum ar fi: lema, partea de vorbire, etc.

## 4. Extragere de relații din text

Ulterior identificării entităților cu nume dintr-un text, urmează procesul de extragere a relațiilor dintre acestea. In funcție de context, pot fi identificate diferite modalități de exprimare a relațiilor între entități.

O expresie, conținând de obicei un verb, poate oferi informații referitor la două entități. De exemplu, în propoziția ”Klaus Iohannis a călătorit în Republica Moldova”, pot fi identificate două entități: ”Klaus Iohannis”, de tip persoană, și ”Republica Moldova”, de tip locație. Acestea sunt conectate prin intermediul expresiei ”a călătorit în” care conține verbul ”a călători”. Acesta identifică relația dintre cele două entități.

Relațiile posibile pot fi definite înaintea implementării procesului de extracție sau deduse automat la rulare. În primul caz, sistemul utilizează un mecanism similar unei ontologii, (generală sau specifică unui domeniu). Astfel, se pot exprima tipuri de entități care pot fi conectate prin relații specifice.

Este posibilă definirea de reguli de extracție prin specificarea unor șabloane de text utilizate în exprimarea relațiilor. Definirea șabloanelor poate fi realizată utilizând reguli complexe sau, în anumite situații, expresii regulate. Relația extrasă este exprimată sub forma unui triplet de forma ”entitate1, relație, entitate2”. Această exprimare este similară modelului RDF (Resource Description Framework). Prin combinarea mai multor astfel de relații rezultă un graf RDF. Acesta poate fi vizualizat reprezentând fiecare triplet sub forma nod-muchie-nod (fiecare entitate devine un nod, iar relația devine o legătură între acestea) [12]. RDF este o recomandare W3C (World Wide Web Consortium).

Există două tipuri de relații: cu nume și fără nume. Relațiile fără nume sunt deduse automat prin examinarea entităților care apar împreună în cadrul aceleiași fraze. În acest caz este evidentă existența unei legături între cele două entități exprimată în text, dar aceasta nu poate fi extrasă corespunzător, tipul relației rămânând nedefinit. Această tehnică poate fi aplicată în general cu rezultate mai bune asupra textelor de dimensiune mare, conținând mai multe fraze care fac legătura între entități.

## 5. Contribuții personale

Extragerea de entități și a relațiilor dintre acestea sunt operațiuni foarte importante pentru diferite activități de procesare a documentelor nestructurate. Scopul final este de a organiza informația într-o formă de tip graf, în care entitățile sunt reprezentate prin noduri iar relațiile prin muchii între acestea.

În cadrul acestei teze sunt prezentate diferitele metode care au fost utilizate pentru construcția unui sistem de recunoaștere de entități (persoane, organizații, locații, timp) si relații între acestea pentru limba română.

Pentru proiectul CoRoLa (The Reference Corpus for Contemporary Romanian Language – ”Corpusul de referință pentru limba română contemporană” [57]) au fost antrenate reprezentări vectoriale ale cuvintelor, cunoscute în limba engleză drept ”word embeddings”, conform celor descrise în secțiunea 5.2 și în articolul publicat [98]. Acestea reprezintă unul dintre mecanismele cheie de codificare a cuvintelor în cadrul aplicațiilor moderne de procesare a limbajului natural, incluzând sistemul de detecție de entități care a fost dezvoltat. De asemenea, a fost creat un serviciu web care permite oricărei aplicații să obțină reprezentarea vectorială a unui cuvânt pe baza reprezentărilor din CoRoLa. Acest serviciu este disponibil public ca un serviciu REST la adresa:

*http://89.38.230.23/word\_embeddings/ws/wordvectors\_get.php?w1=WORD*

unde „WORD” este un parametru care va primi cuvântul pentru care se dorește obținerea unei reprezentări vectoriale cu ajutorul serviciului web.

În același context, a fost dezvoltată o platformă pentru interogarea părții audio a corpusului CoRoLa [99]. Aceasta este disponibilă online la adresa http://89.38.230.23/corola\_sound\_search/ și a stat la baza interfeței sistemului de recunoaștere de entități, așa cum este descrisă în capitolul 6.7.

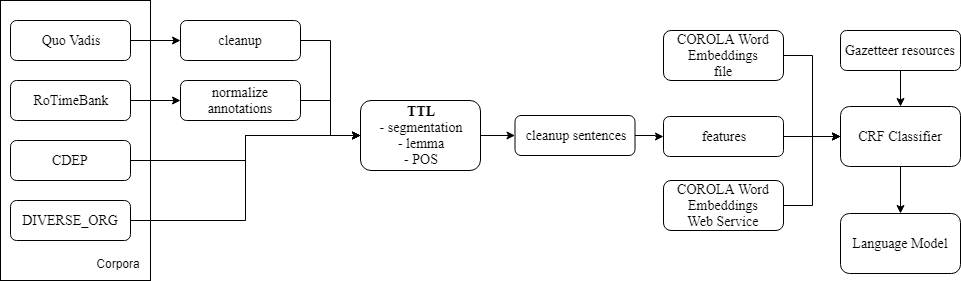
Un prototip de sistem pentru recunoașterea de entități și extragerea relațiilor dintre acestea a fost implementat și descris pe larg în capitolul 6. În acest scop au fost adunate resursele lingvistice disponibile, descrise în capitolul 6.1. Printre acestea se numără romanul adnotat cu entități ”Quo Vadis”, corpusul RoTimeBank și diferite liste de entități așa cum au fost amintite în capitolul 3.1. Totodată, pentru a crește numărul resurselor disponibile, au fost adnotate manual o serie de transcrieri ale dezbaterilor parlamentare din Camera Deputaților.

Ținând cont de motivele prezentate în capitolele 3.5 și 3.6, pentru implementare a fost ales un clasificator de tip CRF, reprezentat de bine-cunoscutul pachet software Stanford NER. Principala motivație a fost reprezentată de volumul redus de resurse adnotate manual disponibile pentru limba română pe partea de recunoaștere de entități, ceea ce face extrem de dificilă, sau chiar imposibilă, o implementare performantă folosind alte tehnologii, cum ar fi cele bazate pe rețele neuronale.

Pentru a putea utiliza Stanford NER împreună cu reprezentările vectoriale antrenate utilizând corpusul CoRoLa, acesta a trebuit să fie modificat pentru a putea integra apelurile către serviciul web descris mai sus aferente fiecărui cuvânt pentru care nu se cunoștea reprezentarea. În felul acesta a putut fi utilizată generarea automată de reprezentări vectoriale inclusiv pentru cuvinte complet necunoscute, pe baza reprezentărilor asociate secvențelor de caractere (”character embeddings”), așa cum a fost prezentat în capitolele 5.2 și 6.3.

Ca etapă finală a lanțului de procesări, a fost dezvoltat un modul bazat pe reguli pentru a îmbunătăți recunoașterea de entități ulterior procesării cu Stanford NER. Acesta este prezentat în capitolul 6.4, împreună cu o comparație a performanțelor sistemului cu și fără acest modul. Concluzia a fost că se realizează o creștere a performanțelor dacă se utilizează acest ultim modul în cadrul sistemului.

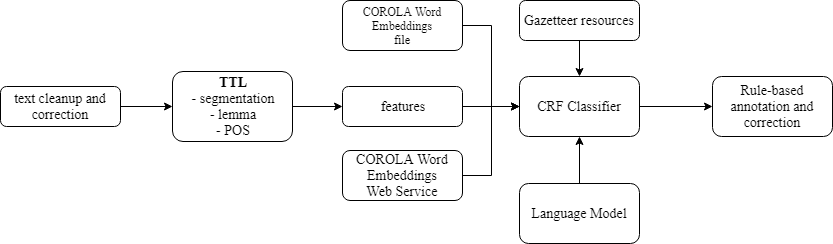
Procesul de antrenare a sistemului este prezentat în Imaginea 1.



***Imaginea 1.*** *Procesul de antrenare*

Un alt modul bazat pe reguli a fost dezvoltat în vederea extragerii relațiilor între entitățile identificate de modulele anterioare. Acesta rulează la finalul procesării, după ce toate celelalte adnotări au fost efectuate. Ca rezultat se obține un fișier GraphML care poate fi deschis cu orice aplicație de vizualizare a grafurilor. Un exemplu este oferit în capitolul 6.6 utilizând aplicația de vizualizare Gephi [83].

Pentru a permite interacțiunea utilizatorilor cu sistemul de recunoaștere de entități, precum și vizualizarea relațiilor identificate, a fost construită o interfață web, disponibilă online la adresa http://89.38.230.23/ner/ .Aceasta este descrisă în detaliu în capitolul 6.7 și utilizează fluxul de procesare prezentat în capitolul 6.5 și în Imaginea 2.



***Imaginea 2.*** *Fluxul de execuție al sistemului*

În cadrul acestei teze a fost construit un sistem pentru recunoașterea de entități în limba română și a relațiilor dintre acestea, cuprinzând: resurse lingvistice, integrare cu sisteme primare de procesare a limbajului natural, flux de antrenare, flux de execuție și o interfață web care permite interacțiunea oricărui utilizator uman cu sistemul.

## 6. Referințe

[1] Merriam-Webster Dictionary, <http://www.merriam-webster.com/>

[2] SIRUTA, <http://data.gov.ro/dataset/siruta>

[3] J. Goyvaerts, S. Levithan, “Regular Expressions Cookbook”, O’Reilly, 2009

[4] S. Memeti, S. Pllana, “PaREM: A Novel Approach for Parallel Regular Expression Matching”, IEEE 17th International Conference on Computational Science and Engineering (CSE), IEEE Press, 2014, 690-697 p.

[5] R. Sin'ya, K.i Matsuzaki, M. Sassa, “Simultaneous Finite Automata: An Efficient Data-Parallel Model for Regular Expression Matching”, ICPP '13 Proceedings of the 2013 42nd International Conference on Parallel Processing, Pages 220-229, IEEE Computer Society

[6] NGA GEOnet Names Server (GNS), <http://geonames.nga.mil/gns/html/>

[7] “List of current heads of state and government”,

<https://en.wikipedia.org/wiki/List_of_current_heads_of_state_and_government>

[8] Text Encoding Initiative, <http://www.tei-c.org/index.xml>

[9] E.F. Tjong, K. Sang, “Introduction to the CoNLL-2002 Shared Task: Language-Independent Named Entity Recognition”, Proceedings of the sixth conference on Natural Language Learning, Taipei, Taiwan, 2002, pages 142-147

[10] R. Munro, C.D. Manning, “Accurate Unsupervised Joint Named-Entity Extraction from Unaligned Parallel Text”, Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics, pages 21–29, Jeju, Republic of Korea, 8-14 July 2012

[11] D. Tufis, “Algorithms and Data Design Issues for Basic NLP Tools”, Language Engineering for Lesser-Studied Languages, IOS Press, 2009

[12] “RDF 1.1 Concepts and Abstract Syntax”, W3C Recommendation 25 February 2014, <https://www.w3.org/TR/rdf11-concepts/>

[13] “OWL 2 Web Ontology Language Document Overview (Second Edition)”, W3C Recommendation 11 December 2012, <https://www.w3.org/TR/2012/REC-owl2-overview-20121211/>

[14] Barbara Lust, Introduction to Studies of Anaphora Acquisition, D. Reidel, 1986

[15] S. Lappin, H.J. Leass, “An Algorithm for Pronominal Anaphora Resolution”, Computational Linguistics, Vol 20, Num 4, Pages 535-561

[16] R. Mitkov, “Anaphora resolution”, Routledge, 2013

[17] Black, Paul E., ed. (14 August 2008), "Levenshtein distance", Dictionary of Algorithms and Data Structures [online], U.S. National Institute of Standards and Technology

[18] Winkler, W. E. (1990). "String Comparator Metrics and Enhanced Decision Rules in the Fellegi-Sunter Model of Record Linkage", Proceedings of the Section on Survey Research Methods. American Statistical Association: 354–359

[19] V. F. Pais; D. S. Ciobanu, “Fast Name Searching on GPU”, European Intelligence and Security Informatics Conference (EISIC), 2013, pages 217-217.

[20] R. Mihalcea, A. Csomai, “Wikify!: linking documents to encyclopedic knowledge”, Proceedings of the sixteenth ACM conference on Conference on information and knowledge management, 2007, Pages 233-242.

[21] X. Cheng, D. Roth, “Relational Inference for Wikification”, EMNLP, 2013.

[22] M. J. Pickering, L. Wong, S.M. Rüger, “ANSES: Summarisation of News Video”, Second International Conference, CIVR 2003 Urbana-Champaign, IL, USA, July 24–25, 2003 Proceedings, pp 425-434

[23] E. Aramaki, Y. Miura, M. Tonoike, T. Ohkuma, H. Mashuichi, K. Ohe, “TEXT2TABLE: Medical Text Summarization System based on Named Entity Recognition and Modality Identification“, Proceedings of the Workshop on BioNLP, 2009, pages 185–192

[24] <http://emm.newsexplorer.eu/>

[25] <http://emm.newsexplorer.eu/NewsExplorer/entities/en/2042.html>

[26] <http://emm.newsexplorer.eu/NewsExplorer/readme.html>

[27] RFC5322, “Internet Message Format”, October 2008, <https://tools.ietf.org/html/rfc5322>

[28] R. Navigli and S. Ponzetto. “BabelNet: The Automatic Construction, Evaluation and Application of a Wide-Coverage Multilingual Semantic Network”, Artificial Intelligence, 193, Elsevier, 2012, pp. 217-250.

[29] A. Moro, R. Navigli. “SemEval-2015 Task 13: Multilingual All-Words Sense Disambiguation and Entity Linking”, Proc. of the 9th International Workshop on Semantic Evaluation (SemEval), in the the 2015 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL 2015), Denver, Colorado, June 4-5th, 2015, pp. 288-297.

[30] C.J. van Rijsbergen, S.E. Robertson and M.F. Porter, 1980. New models in probabilistic information retrieval. London: British Library. (British Library Research and Development Report, no. 5587).

[31] Karen Sparck Jones and Peter Willet, 1997, Readings in Information Retrieval, San Francisco: Morgan Kaufmann, ISBN 1-55860-454-4.

[32] http://snowballstem.org

[33] Claudiu Sorin Irimias, “A Romanian stemmer”, Studia Informatica, Sp. Issue 1, 2009

[34] Ana-Maria Barbu, “Romanian Lexical Data Bases: Inflected and Syllabic Forms Dictionaries”, Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08). Marrakech, Morocco, may 2008

[35] A.M. Barbu, Monitorizarea Presei în cadrul Proiectului Neorom, în A. Iftene, N. H. Teodorescu, D. Cristea, D. Tufiş (ed.) „Lucrările conferinţei Resurse lingvistice şi instrumente pentru prelucrarea limbii române, Bucureşti, 6–7 mai 2010”, Editura Universităţii „Alexandru Ioan Cuza”, Iaşi, 2010, p.123-132.

[36] Radu Simionescu. 2011. Hybrid POS Tagger. In Proceedings of “Language Resources and Tools with Industrial Applications” Workshop (Eurolan 2011 summerschool).

[37] E. Brill, “A simple rule-based part of speech tagger”, Proceedings of the third conference on Applied natural language processing (ANLC’92), Pages 152-155, Trento, Italy — March 31 - April 03, 1992

[38] Varshney, L. R.; Chen, B. L.; Paniagua, E.; Hall, D. H.; Chklovskii, D. B. (2011). Sporns, Olaf, ed. "Structural Properties of the Caenorhabditis elegans Neuronal Network". PLoS Computational Biology. 7 (2): e1001066.

[39] Briggman, K. L.; Helmstaedter, M.; Denk, W. (Mar 10, 2011). "Wiring specificity in the direction-selectivity circuit of the retina". Nature. 471 (7337): 183–8.

[40] Bock, Davi D.; Lee, Wei-Chung Allen; Kerlin, Aaron M.; Andermann, Mark L.; Hood, Greg; Wetzel, Arthur W.; Yurgenson, Sergey; Soucy, Edward R.; Kim, Hyon Suk; Reid, R. Clay (2011). "Network anatomy and in vivo physiology of visual cortical neurons". Nature. 471 (7337): 177–182.

[41] Nair, Vinod, and Geoffrey E. Hinton. "Rectified linear units improve restricted boltzmann machines." Proceedings of the 27th International Conference on Machine Learning (ICML-10). 2010.

[42] LeCun, Yann; Bengio, Yoshua; Hinton, Geoffrey (2015). "Deep learning". Nature. 521 (7553): 436–444.

[43] Sepp Hochreiter; Jürgen Schmidhuber (1997). "Long short-term memory". Neural Computation. 9 (8): 1735–1780. doi:10.1162/neco.1997.9.8.1735.

[44] Felix A. Gers; Jürgen Schmidhuber; Fred Cummins (2000). "Learning to Forget: Continual Prediction with LSTM". Neural Computation. 12 (10): 2451–2471. doi:10.1162/089976600300015015.

[45] https://en.wikipedia.org/wiki/Recurrent\_neural\_network

[46] Helmut Schmid. 1994. Part-of-speech tagging with neural networks. In Proceedings of the 15th conference on Computational linguistics - Volume 1 (COLING '94), Vol. 1. Association for Computational Linguistics, Stroudsburg, PA, USA, 172-176. DOI: <http://dx.doi.org/10.3115/991886.991915>

[47] F. Alam, S. A. Chowdhury and S. R. H. Noori, "Bidirectional LSTMs — CRFs networks for bangla POS tagging," 2016 19th International Conference on Computer and Information Technology (ICCIT), Dhaka, 2016, pp. 377-382, doi: 10.1109/ICCITECHN.2016.7860227

[48] S. R. Kundeti, J. Vijayananda, S. Mujjiga and M. Kalyan, "Clinical named entity recognition: Challenges and opportunities," 2016 IEEE International Conference on Big Data (Big Data), Washington, DC, 2016, pp. 1937-1945, doi: 10.1109/BigData.2016.7840814

[49] Dan Cristea, Gabriela-Eugenia Dima (2001). An integrating framework for anaphora resolution. In Information Science and Technology, Romanian Academy Publishing House, Bucharest, vol. 4, no. 3-4, p 273-291

[50] <http://www.cdep.ro/>

[51] <http://www.cnts.ua.ac.be/conll2003/ner/>

[52] Pontus Stenetorp, Sampo Pyysalo, Goran Topić, Tomoko Ohta, Sophia Ananiadou and Jun'ichi Tsujii (2012). brat: a Web-based Tool for NLP-Assisted Text Annotation. In Proceedings of the Demonstrations Session at EACL 2012.

[53] <https://www.tensorflow.org/>

[54] Diederik P. Kingma, Jimmy Ba, “Adam: A Method for Stochastic Optimization”, CoRR, vol abs/1412.6980, year 2014, <http://arxiv.org/abs/1412.6980>

[55] Tjong Kim Sang, Erik F.; De Meulder, Fien, “Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition”, CoNLL 2003

[56] D. Tufiş, R. Ion, A. Ceauşu, D. Ştefănescu, “RACAI's Linguistic Web Services”, in Proceedings of the 6th Language Resources and Evaluation Conference – LREC’08, Marrakech, Morocco, 2008.

[57] V. Barbu Mititelu, D. Tufiș, E. Irimia, The Reference Corpus of Contemporary Romanian Language (CoRoLa), in Proceedings of the 11th Language Resources and Evaluation Conference – LREC’18, Miyazaki, Japan, European Language Resources Association (ELRA), 2018

[58] T. Mikolov, K. Chen, G. Corrado, J. Dean, “Efficient Estimation of Word Representations in Vector Space”, arXiv:1301.3781, 2013

[59] Pre-trained word embeddings using the Romanian Wikipedia corpus, https://github.com/facebookresearch/fastText/blob/master/docs/pretrained-vectors.md

[60] P. Bojanowski, E. Grave, A. Joulin, T. Mikolov, “Enriching Word Vectors with Subword Information”, arXiv:1607.04606, 2016

[61] L. Finkelstein, E. Gabrilovich, Y. Matias, E. Rivlin, Z. Solan, G. Wolfman, and E. Ruppin, “Placing search in context: the concept revisited”, In WWW, pages 406–414, 2001

[62] S. Hassan, R. Mihalcea, “Cross-lingual semantic relatedness using encyclopedic knowledge”, In Proc. EMNLP, 2009

[63] Pre-trained CoRoLa word vectors, analogies application and data sets: <http://89.38.230.23/word_embeddings/>

[64] Common Romanian names: first names <https://ro.wikipedia.org/wiki/List%C4%83_de_prenume_rom%C3%A2ne%C8%99ti> , last names:<https://ro.wikipedia.org/wiki/Categorie:Nume_de_familie_rom%C3%A2ne%C8%99ti>

[65] D. Cristea, D. Gifu, M. Colhon, P. Diac, A. Bibiri, C. Maranduc, L. Scutelnicu, “Quo Vadis: A Corpus of Entities and Relations”. In: Gala N., Rapp R., Bel-Enguix G. (eds) Language Production, Cognition, and the Lexicon. Text, Speech and Language Technology, vol 48. Springer, Cham, 2015

[66] Component Metadata Infrastructure (CMDI), <https://www.clarin.eu/content/component-metadata>

[67] Firth J.R.: Papers in linguistics, 1934–1951, Oxford University Press, 1957

[68] S. Deerwester, S.T. Dumais, G.W. Furnas, T.K. Landauer, R. Harshman, “Indexing by Latent Semantic Analysis”, Journal of the Association for Information Science and Technology, Vol 41, Issue 6, September 1990, Pages 391-407

[69] H. Schutze, “Dimensions of meaning”, in “Proceedings of Supercomputing ’92”, pp. 787–796. IEEE Press., 1992

[70] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, R. Salakhutdinov, “Dropout: A Simple Way to Prevent Neural Networks from Overfitting”, Journal of Machine Learning Research 15 (2014) 1929-1958

[71] V. Păiș, D. Tufiș, „More Romanian word embeddings from the ReTeRom project”, in „Proceedings of the 13th International Conference Linguistic Resources and Tools for Processing the Romanian Language”, pp. 91-100, Iasi, 2018

[72] L.J.P. van der Maaten and G.E. Hinton, “Visualizing High-Dimensional Data Using t-SNE”, Journal of Machine Learning Research 9(Nov):2579-2605, 2008

[73] L.J.P. van der Maaten, “Accelerating t-SNE using Tree-Based Algorithms”, Journal of Machine Learning Research 15(Oct):3221-3245, 2014

[74] Anca-Diana Bibiri , Mihaela Colhon , Paul Diac , Dan Cristea, “Statistics Over A Corpus Of Semantic Links:“QuoVadis””, Proceedings of the 10th International Conference “Linguistic Resources and Tools for Processing the Romanian Language”, pp 33-44, Iasi, 2014

[75] Mihaela Colhon, Paul Diac, Cătălina Mărănduc, Cenel Augusto Perez, ““Quo Vadis” Research Areas – Text Analysis”, Proceedings of the 10th International Conference “Linguistic Resources and Tools for Processing the Romanian Language”, pp 45-56, Iasi, 2014

[76] Corina Forăscu, Radu Ion, “Timebank 1.2: O versiune adnotată în Limba Română”, Lucrările atelierului “Resurse lingvistice şi instrumente pentru prelucrarea limbii române”, pp 69-74, Iasi, 2006

[77] CDEP annotated corpus, <http://89.38.230.23/ner/resources/ner_cdep.zip>

[78] “Gazette” resource used for name entity recognition, <http://89.38.230.23/ner/resources/ner_gazette.zip>

[79] Stanford NER website, <https://nlp.stanford.edu/software/CRF-NER.shtml>

[80] Jenny Rose Finkel, Trond Grenager, and Christopher Manning. 2005. Incorporating Non-local Information into Information Extraction Systems by Gibbs Sampling. Proceedings of the 43nd Annual Meeting of the Association for Computational Linguistics (ACL 2005), pp. 363-370. <http://nlp.stanford.edu/~manning/papers/gibbscrf3.pdf>

[81] Angel X. Chang and Christopher D. Manning. 2012. SUTIME: A Library for Recognizing and Normalizing Time Expressions. 8th International Conference on Language Resources and Evaluation (LREC 2012)

[82] GraphML format specifications, <http://graphml.graphdrawing.org/>

[83] Gephi - The Open Graph Viz Platform, <https://gephi.org/>

[84] VisJS library, <http://visjs.org/>

[85] Tomaz Erjavec. MULTEXT-East Version 3: Multilingual Morpho-syntactic Specifications, Lexicons and Corpora. In Proceedings of the Fourth International Conference on Language Resources and Evaluation, LREC ’04, Lisbon, Portugal, 2004.

[86] Radu Ion (2007) “Word Sense Disambiguation Methods Applied to English and Romanian” (in Romanian). PhD Thesis, Romanian Academy, <http://www.racai.ro/media/radu-ion-tezadoc.pdf>

[87] <https://fasttext.cc/>, Library for efficient text classification and representation learning

[88] J. Kupiec. Robust part-of-speech tagging using a hidden Markov model. Computer Speech and Language, 6:225-242, 1992.

[89] SCHMID, Helmut. Part-of-speech tagging with neural networks. In: *Proceedings of the 15th conference on Computational linguistics-Volume 1,* Association for Computational Linguistics, 1994. p. 172-176.

[90] Tufiș, D., Ceaușu, A., DIAC+: A Professional Diacritics Recovering System. In *Proceedings of the 6th Language Resources and Evaluation Conference - LREC 2008*, Marrakech, Morocco, May 2008. ELRA - European Language Resources Association.

[91] R. Mihalcea and V. Nastase, “Letter level learning for language independent diacritics restoration,” in proceedings of the 6th conference on Natural language learning-Volume 20. Association for Computational Linguistics, 2002, pp. 1–7.

[92] C. Ungurean, D. Burileanu, V. Popescu, C. Negrescu, and A. Dervis, “Automatic diacritic restoration for a tts-based e-mail reader application,” UPB Scientific Bulletin, Series C, vol. 70, no. 4, pp. 3–12, 2008

[93] Maria Nutu, Beata Lorincz, Adriana Stan, "Deep Learning for Automatic Diacritics Restoration in Romanian", In Proceedings of the IEEE 15th International Conference on Intelligent Computer Communication and Processing, Cluj-Napoca, Romania, 2019.

[94] Navarro, Gonzalo & Baeza-Yates, Ricardo & Sutinen, Erkki & Tarhio, Jorma. (2001). Indexing Methods for Approximate String Matching. IEEE Data Eng Bull. 24.

[95] Jonathan Raiman and Olivier Raiman. 2018. DeepType: Multilingual Entity Linking by Neural Type System Evolution. CoRR abs/1802.01021 (2018).

[96] Snyman, Jan A.; Wilke, Daniel N. (2018). Practical Mathematical Optimization - Basic Optimization Theory and Gradient-Based Algorithms. Springer Optimization and Its Applications. 133 (2 ed.). Springer. ISBN 978-3-319-77585-2.

[97] Image from <https://commons.wikimedia.org/wiki/File:LSTM.png> , licensed under free license Creative Commons Attribution-Share Alike 4.0 International.

[98] V. Paiș, D. Tufiș, "Computing distributed representations of words using the COROLA corpus", In *Proceedings of the Romanian Academy*, Series A, Volume 19, Number 2/2018, pp. 403–409.

[99] T. Boros, S.D. Dumitrescu, V. Pais, Tools and resources for Romanian text-to-speech and speech-to-text applications. In *Proceedings of the International Conference on Human-Computer Interaction - RoCHI 2018*, pp 46-53.