Revista Română de Interacţiune Om-Calculator 6 (4) 2013, 357-376

Criminal Detection – O aplicaţie pentru identificarea infractorilor

Ana-Teodora Petrea, Adrian Iftene

Universitatea Alexandru Ioan Cuza din Iaşi, Facultatea de Informatică
General Berthelot 16, 700483, Iaşi
E-mail: {ana.petrea, adiftene}@info.uaic.ro

Rezumat. Prezenta lucrare îşi propune să vină în ajutorul celor ce doresc să identifice rapid un posibil infractor, pe baza informaţiilor, pe care le avem disponibile la un moment dat. Sistemul construit foloseşte ca intrare informaţie nestructurată în format text, ce poate proveni din surse precum ziare, procese verbale de la locul faptei, declaraţii ale martorilor, etc.. Aceste texte sunt prelucrate folosind tehnici specifice lingvisticii computaţionale, pentru a extrage informaţii relevante, pentru identificarea infractorului. Procesările textuale prezentate în cadrul acestei lucrări, permit (1) extragerea entităţilor de tip nume (astfel se poate obţine un grup de suspecţi sau locaţia unde s-a întâmplat infracţiunea), (2) identificarea relaţiilor între cuvinte, pentru extragerea de informaţii utile (aşa se pot obţine caracteristici fizice ale infractorilor sau tipul infracţiunii sau arma folosită) şi pentru identificarea rolurilor (astfel se pot identifica victimele, răufăcătorii sau martorii). Aplicaţia permite în plus crearea unui portret robot pentru infractor, asocierea de poze sau portrete robot la persoanele pe care le gestionăm în aplicaţie şi afişarea pe o hartă a locului evenimentului. Considerăm că, aplicaţia ar putea fi utilizată într-un mediu real, în cadrul diferitelor instituţii, în care procesarea acestor tipuri de texte este realizată manual în prezent, în primul rând de către angajaţi.

Cuvinte cheie: Lingvistică computaţională, Supraveghere pe Internet, Lingvistică judiciară.

1. Introducere

Inteligenţa artificială este ramura informaticii ce are ca scop "crearea de maşini inteligente", după cum a definit-o unul dintre fondatorii ei, John McCarthy. Punctul de pornire în crearea acestui domeniu a fost ipoteza conform căreia inteligenţa umană – *the sapience of Homo Sapiens* – are un mecanism care poate fi descris în detaliu pentru ca, ulterior, sa fie simulat de o maşină (McCarthy et al., 1995). Printre scopurile inteligenţei artificiale se numără crearea de maşini capabile de judecată, cunoaştere, învăţare, percepţie şi care să aibă performanţe comparabile cu cele ale unui individ uman.

Capacitatea de înțelegere a mesajelor scrise este una dintre caracteristicile distinctive ale inteligenței umane (Cristea et al., 2007). În consecință, subdomenii ale Inteligenței artificiale se dedică întocmai procesării limbajului natural. Lingvistica computațională este unul dintre ele şi are ca principal obiectiv modelarea computațională, statistică sau bazată pe reguli, a limbajului natural. Ingineria lingvistică este domeniul complementar lingvisticii computaționale, care fructifică rezultatele acestuia prin crearea de aplicații care să servească sferelor industriale, de comerț, sociale, etc..

Lingvistica computațională a luat naştere în jurul anului 1950, în SUA, odată cu primele tentative de traducere automată (Hutchins, 1999). Deşi, inițial, acest domeniu a stârnit interes şi a atras numeroase investiții, după câțiva ani popularitatea sa a scăzut dramatic, rezultatele fiind nesatisfăcătoare. Cercetările au continuat la o scară mult mai mică. Până la mijlocul anilor 1970, rezultatele au început să apară: sistemul Systran (http://www.systransoft.com/) a început să fie folosit de instituții importante precum Comisia Comunităților Europene şi Forțele Aeriene ale SUA. Între anii 1980-1990, studiul traducerii automate a luat amploare peste tot în lume şi, totodată, au apărut noi direcții în cercetare: metodele statistice şi traducerea pe bază de exemple.

2. Elemente utilizate în lucrare

2.1 Extragerea de informații

Acest domeniu unifică obiective precum extragerea automată a unor informații dintr-un text nestructurat sau semi-structurat, adnotarea automată şi extragerea de conținuturi din documente multimedia (audio, video, foto).

Din cauza nivelului de dificultate ridicat al acestei sarcini, abordările - curente se concentrează asupra extragerii unor anumite tipuri de informații din texte, care aparțin unor domenii specifice (Peng şi McCallum, 2006), (Shimizu, 2006) sau se concentrează pe extragerea de informații din cadrul rețelelor sociale (Tulbure-Dombi et al., 2010). De asemenea, este acordată o atenție deosebită extragerii de informații de pe web (Chang et al., 2006).

Extragerea informațiilor necesită rezolvarea unei alte probleme – simplificarea textului, pentru a obține o structurare a informației prezente în text. Printre subsarcinile acestui domeniu se numără:

1. Extragerea entităților de tip nume, care poate include:
    - *Recunoaşterea entităților de tip nume* – identificarea numelor de persoane, organizații, locații, date, expresii numerice, etc. Tipurile alese spre a fi identificate vor depinde de necesitățile sistemului de căutare documentară (Information Retrieval (IE) în engleză);
    - *Rezoluția anaforei*: detecția de legături coreferențiale şi anaforice, pentru entitățile descoperite;
    - Identificarea de relații între entități de tipuri diferite sau de acelaşi tip.
2. Extragerea de informații semi-structurate (tabele, comentarii) din documente;
3. *Analiza limbii şi a vocabularului*, pentru crearea de reguli care să determine identificarea informațiilor dorite;
4. *Colectarea unui vocabular* de termeni relevanți domeniului.

Prezenta lucrare prezintă modul în care ne-am propus să extragem informaţii relevante dintr-un text, care descrie o infracțiune şi respectă anumite proprietăți. Pentru acest lucru, s-a apelat la câteva dintre subsarcinile menționate mai sus: identificarea entităților de tip nume, rezoluția coreferinței şi altele precum analiza dependențelor sintactice între cuvinte. În paginile care urmează, vom descrie pe fiecare dintre ele, pentru ca în partea a doua a lucrării să descriem modalitățile folosite pentru atingerea scopului inițial.

2.2 Identificarea entităților de tip nume

Extragerea entităților de tip nume este o cerință a lingvisticii computaționale, care îşi propune să localizeze şi să clasifice cuvintele sau grupurile de cuvinte care apar într-un text şi reprezintă nume proprii (Nadeau şi Sekine, 2007). Acestea pot desemna persoane, organizații, locații, expresii de timp, cantități, valori monetare, procente, etc..

Termenul definit prin sintagma „entitate de tip nume", acum binecunoscut şi larg utilizat în procesarea limbajului natural, a fost introdus cu ocazia celei de-a şasea ediții a Message Understanding Conference (MUC-6) (Grishman şi Sundheim, 1996). Scopul principal al conferinței era, în acea perioadă, extragerea de informații structurate despre activitatea companiilor sau cea a instituțiilor importante, pe baza articolelor din ziare sau din alte publicații. Pentru realizarea acestei sarcini a fost semnalată

necesitatea unui modul responsabil cu identificarea unităților de informație reprezentând nume de persoane, organizații, locații şi a expresiilor numerice reprezentând date calendaristice, cantități, procente, etc. În acest context a fost definită recunoaşterea şi clasificarea entităților de tip nume ("Named Entity Recognition and Classification" – NERC) ca subsarcină a extragerii de informații din texte ("Information Extraction" – IE). Ulterior, ea s-a dovedit a fi necesară multor altor subdomenii ale procesării limbajului natural (Sekine et al., 1998), (Borthwick et al., 1998) şi (Masayuki şi Matsumoto, 2003).

De-a lungul timpului, în abordarea acestui domeniu au fost considerate mai multe strategii (Nadeau şi Sekine, 2007) şi (Iftene et al., 2011). Primele dintre ele au constat în scrierea de algoritmi bazați pe regulile gramaticale ale respectivului limbaj. Se urmărea crearea de algoritmi cât mai elaborați, care să trateze în mod corect toate posibilele situații, fără a crea ambiguități, pentru a obține performanțe cât mai bune. Precizăm că performanța unui sistem de recunoaştere a entităților de tip nume este dată de parametrii "precizie" şi "recall" (Bikel et al., 1997).

3. Arhitectura aplicației. Componente principale

Aplicația dezvoltată are două componente care pot fi considerate, în mare măsură, independente una față de cealaltă:

- *Componenta principală* comunică în mod direct cu utilizatorul, prin intermediul unei interfețe grafice şi totodată, comandă execuția celei de-a doua componente. Aceasta este dezvoltată în C#, pe platforma .NET şi beneficiază de o interfață grafică realizată cu ajutorul Windows Presentation Foundation (WPF). Având la dispoziție două fişiere XML conținând date referitoare la infracțiunile petrecute într-o anumită regiune şi la persoanele care au înfăptuit infracțiunile respective, aplicația furnizează modalități eficiente de vizualizare a datelor şi de căutare a informațiilor dorite într-o bază de date locală.
- *Componenta auxiliară*, care primind un text care descrie o fărădelege, este capabilă să identifice în mod automat informații precum data, locația infracțiunii, numele victimelor, semnalmente ale infractorului, etc.. Datele astfel identificate sunt stocate într-un fişier XML şi pot fi ulterior folosite de aplicația principală. Această aplicație este una de tip consolă şi a fost creată în Java, folosind

mediul de dezvoltare NetBeans IDE 7.1.2.

Sistemul creat lucrează cu informații, respectiv texte în limba engleză. Descrierea detaliată a arhitecturii, a modului de proiectare şi a funcționalităților oferite de fiecare modul al aplicației vor fi prezentate în capitolele următoare.

3.1 Componenta de bază

Aplicația de bază este realizată cu scopul de a facilita accesul simplu şi eficient la informațiile legate de infracțiunile comise într-o anumită regiune, pe o anumită perioadă de timp. Aceasta permite filtrarea datelor, aplicând multiple criterii specificate de utilizator, urmând ca vizualizarea rezultatelor să fie realizată în funcție de necesitățile şi preferințele acestuia.

Scenariul de utilizare vizat este unul conform căruia clientul aplicației ar fi o secție de poliție, care gestionează o bază de date de dimensiuni mari, ce conține informații despre infracțiunile comise în regiunea pe care o supraveghează şi despre infractorii care le-au comis. Atunci când se produce o nouă fărădelege, este consultată baza de date, în vederea găsirii de posibili suspecți. Datorită volumului mare de date, care trebuiesc interogate şi a multitudinii de factori, care trebuie să fie luați în considerare în identificarea infractorului, o astfel de aplicație are nevoie să ofere anumite facilități, care să o facă simplu de utilizat şi să permită realizarea unei filtrări corecte şi complexe a informațiilor.

Aplicația lucrează cu informațiile obținute în urma încărcării a două fişiere XML:

- *Criminals.xml* – fiecare nod de tip Criminal (corespunzător unui infractor) are noduri interioare şi atribute în care sunt stocate informații precum: nume, prenume, apelativ, sex, naționalitate, vârstă, înălțime, greutate, caracteristici ale feței (ochi, nas, etc.), calea către o poză a infractorului ş.a.m.d. De asemenea, pentru fiecare infractor avem asociate infracțiunile pe care acesta le-a comis. Tabelul 1 exemplifică structura unui nod XML de tip Criminal.

Tabelul 1: Nod XML de tip Criminal

```
<?xml version="1.0" encoding="utf-8"?>
<Criminals>
  <Criminal Id="1">
```

```
    <PersonalInformation>
      <Nicknames>
        <Nickname>The Zodiac</Nickname>
      </Nicknames>
      <FirstName>John</FirstName>
      <LastName>Walker</LastName>
      <State>Fugitive</State>
        <Sex>Male</Sex>
      <Nationality>Spanish</Nationality>
      <Age>38</Age>
      <Height>189</Height>
      <Weight>180</Weight>
    </PersonalInformation>
    <Appearance>
        <Picture>D:\CrimeWorld\Pozeinfractori\1.jpg</Picture>
        <Skin Color="Black"/>
      <Figure>Solid</Figure>
      <Face Shape="Round" />
      <Eyes Size="Big" Color="Blue" Shape="Hooded" />
      <Lips Shape="Round" Size="" />
      <Nose Size="Small" Shape="Concave"></Nose>
      <Hair Style="Curly" Color="Dark" Length="Short" />
      <DistinctiveSigns>
        <DistinctiveSign Sign="Tattoo" Position="Neck" />
      </DistinctiveSigns>
    </Appearance>
    <Crimes>
      <Crime Id="1" />
    </Crimes>
   </Criminal>
 </Criminals>
```

- *Crimes.xml* – un nod de tip Crime (corespunzător unei infracțiuni) conține următoarele informații: id-urile făptaşilor, numele victimelor, data şi localitatea în care s-a produs fărădelegea, categoriile în care se înscrie aceasta (crimă, jaf, etc.) şi armele utilizate de făptaş (armă de foc, violență fizică, etc.). Tabelul 2 ilustrează structura unui nod XML de tip Crime. Se poate observa legătura dintre acest nod şi cel din Tabelul 1.

Tabelul 2: Nod XML de tip Crime

```
<?xml version="1.0" encoding="utf-8"?>
<Crimes>
  <Crime Id="1">
    <Criminals>
      <Criminal Id="1"></Criminal>
    </Criminals>
    <Victims>
      <Victim>
        <LastName>Fox</LastName>
        <FirstName>Meghan</FirstName>
      </Victim>
    </Victims>
    <Date>
```

```
      <Year>2011</Year>
      <Month>04</Month>
      <Day>06</Day>
    </Date>
    <Location>
      <City>Philipsburg</City>
    </Location>
    <Categories>
      <Category>Murder</Category>
    </Categories>
    <Weapons>
   <Weapon>Gun</Weapon>
    </Weapons>
       </Crime>
</Crimes>
```

În urma încărcării informațiilor din cele două fişiere, vom avea la dispoziție o listă de infracțiuni şi una de infractori (fiecare având anumite caracteristici) şi relațiile dintre ele (vom şti ce infractorul a înfăptuit o anumită fărădelege şi, implicit, ce fărădelegi a comis un infractor).

3.2 Ferestre şi funcționalități

Formularul de start al aplicației are rolul de a asigura accesul către celelalte ferestre, pe care le vom prezenta în continuare. Utilizatorul va alege una din opțiuni, în funcție de funcționalitatea de care are nevoie la un moment dat.

Fereastra de căutare a infractorilor

Această fereastră conține o listă care, la un anumit moment, va conține infractorii ce prezintă caracteristicile selectate de utilizator la momentul respectiv. În momentul deschiderii ferestrei, lista conține toți răufăcătorii din baza de date. Modalități de căutare disponibile:

- *După nume sau alias*: căutarea se face pe baza informațiilor ce se găsesc oriunde în câmpurile nume şi alias, ignorând literele mari sau mici. (vezi Figura 1).
- *Exclusiv după nume*, *folosind Speech Recognition.* Dacă este activată recunoaşterea vocală, (bifând opțiunea corespunzătoare din Figura 1), programul va putea recunoaşte rostirea numelui unui infractor aflat în baza de date şi îl va afişa în lista de rezultate. Desigur, în cazul în care există mai mulți infractori cu acelaşi nume, vor fi afişați toți. Pentru ca această componentă să funcționeze, este

necesar ca un microfon funcțional să fie conectat la calculator şi să fie deschis. Deoarece dorim ca aplicația să fie capabilă să recunoască infractorii după numele lor, trebuie să întocmim o listă care să conțină toate aceste nume. Una din problemele dificile a apărut din faptul că şirurile de caractere formate prin concatenarea numelui şi a prenumelui unui infractor pot fi destul de lungi. De asemenea, deoarece unele nume conțin cuvinte comune, sau foarte asemănătoare, se pot crea confuzii şi pot fi afişate rezultate incorecte. De aceea, s-a ales o altă modalitate de recunoaştere, care va fi descrisă în continuare:

- Inițial, s-a creat un şir de nume şi unul de prenume, în care fiecare cuvânt este introdus o singură dată, chiar dacă se află în componența numelor mai multor infractori;
- În continuare, s-a creat un obiect de tip *GrammarBuilder*, care a fost configurat astfel încât să recunoască rostirea unui cuvânt din lista de prenume, urmată de rostirea unui cuvânt din lista de nume;
- În etapa următoare se verifică dacă şirul astfel format desemnează numele unui infractor şi în caz afirmativ, se adaugă numele infractorul în lista de rezultate.

Figura 1. Formular căutare infractor

- De asemenea, la rostirea frazei "Display All" este afişată lista completă a infractorilor.
- *Advanced Search* (opțiunea din Figura 1 este prezentată detaliat în Figura 2), la apăsarea acestui buton se deschide un alt formular, ce permite utilizatorului să selecteze orice caracteristică a unei infracțiuni sau a unui infractor şi o valoare pentru această caracteristică. Astfel, el îşi poate construi o listă de criterii de căutare, pe care le poate aplica concomitent. Conform conținutului listei de criterii stabilite în Figura 2, sunt căutați infractorii de sex masculin, care au comis infracțiuni din categoria "crimă". În cazul în care au fost selectate mai multe valori pentru aceeaşi caracteristică, vor fi aleşi infractorii care prezintă una dintre acestea. De asemenea, dacă un infractor nu are nicio valoare pentru caracteristica respectivă (de exemplu, dacă nu i se cunoaşte culoarea ochilor), acesta nu este exclus ca potențial suspect.

Figura 2. Formular criterii de căutare

Precizăm faptul că, dacă formularul în care se introduc criteriile de căutare este închis, el poate fi deschis din nou şi va afişa aceleaşi criterii de căutare selectate la ultima folosire a acestuia.

Fereastra de localizare a infracţiunilor

Fereastra destinată localizării infracţiunilor afişează, în momentul deschiderii, o hartă a regiunii a cărei activitate infracţională o monitorizăm. Pe hartă sunt afişate semne distinctive în locaţiile corespunzătoare infracţiunilor care se găsesc în baza de date, fiecare dintre acestea apărând în dreptul oraşului în care a avut loc. Semnele corespunzătoare infracţiunilor rezolvate apar colorate în albastru, iar cele corespunzătoare infracţiunilor ai căror făptaşi nu au fost încă identificaţi sunt marcate cu roşu.

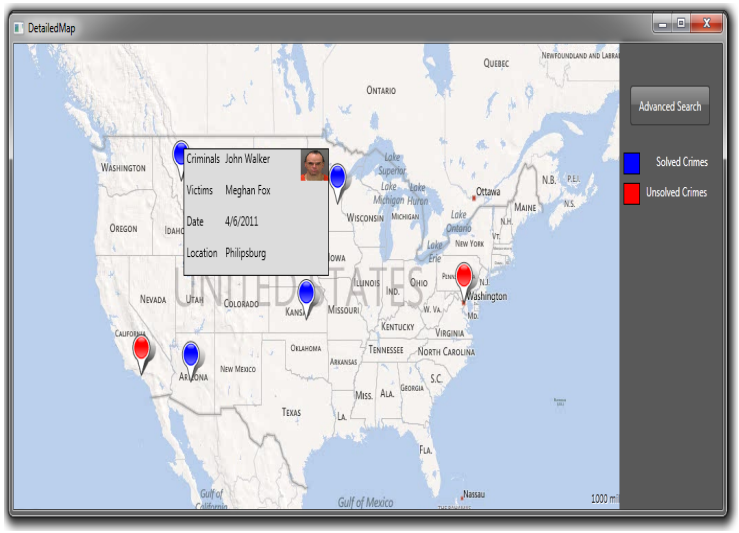

Figura 3. Formular căutare infracţiune

Atunci când utilizatorul selectează una dintre semnele afişate pe hartă, se deschide o fereastră ce conţine informaţii despre infracţiunea respectivă (numele infractorilor, numele victimelor, data şi localitatea în care a avut loc infracţiunea), precum şi poza infractorului (vezi Figura 3).

Pentru ca utilizatorul să poată ajunge uşor la profilul răufăcătorului care a comis o anumită faptă, acesta trebuie doar să apese pe poza acestuia şi se va deschide pagina ce conţine profilul detaliat al acestuia.

Similar căutării de infractori, căutarea locațiilor în care au avut loc infracțiunile poate fi realizată după o serie de criterii de căutare avansate. Acestea pot fi referitoare la fărădelege în sine (dată, locație, tipul infracțiunii, etc.), cât şi la infractorii care au comis acea infracțiune (sex, caracteristici fizice, vârstă, etc.). Formularul care permite selectarea şi aplicarea acestor criterii este cel prezentat mai sus în Figura 2. Vizualizarea rezultatelor se va face, de această dată, pe o hartă, prin afişarea locațiilor corespunzătoare infracțiunilor obținute în procesul de căutare şi filtrare.

Pentru afişarea hărții, respectiv a semnelor distinctive corespunzătoare locațiilor infracțiunilor, s-a folosit biblioteca *Microsoft.Maps.MapControl.WPF* (http://msdn. microsoft.com/en-us/library/microsoft.maps.mapcontrol.wpf.aspx). Pentru fiecare infracțiune, s-a efectuat o cerere la serviciul Bing Maps cerând coordonatele (longitudinea şi latitudinea) locației infracțiunii. Apoi, s-au scalat coordonatele obținute la dimensiunea harții şi s-au afişat semnele distinctive în punctele astfel obținute.

Fereastra de realizare a portretului robot

În situația în care un agent de poliție doreşte să identifice posibili suspecți pentru o fărădelege, acesta va selecta criteriile de căutare conform informațiilor care îi sunt puse la dispoziție. Aceste informații sunt, la rândul lor, furnizate de alte persoane precum martori, victime, persoane apropiate victimelor, etc. Un pas important în găsirea unui infractor a cărui identitate nu este cunoscută este întocmirea unui portret robot. Având acest lucru în minte, s-a ataşat formularului de selectare a criteriilor de căutare o componentă suplimentară.

Unele caracteristici fizice umane sunt mai uşor de identificat prin intermediul imaginilor, decât prin denumirea lor. De altfel, la întocmirea portretelor robot, martorii sunt lăsați să aleagă, dintre mai multe poze, pe cea care este cea mai apropiată de înfățişarea infractorului.

Atunci când utilizatorul va alege una din opțiunile: forma feței, forma ochilor, forma nasului sau forma buzelor, se va activa butonul “Choose”. Apăsarea lui are ca efect deschiderea unei ferestre în care utilizatorul va putea vizualiza fotografii corespunzătoare fiecărei forme a feței, nasului, ochilor, etc.. Astfel, el va putea alege pentru fiecare caracteristică varianta care se potriveşte cel mai bine descrierii infractorului căutat. După acest

pas, caracteristica aleasă va fi completată în mod automat în portretul robot al infractorului. În figurile 4, 5 se pot observa pentru forma feței două din variantele disponibile.

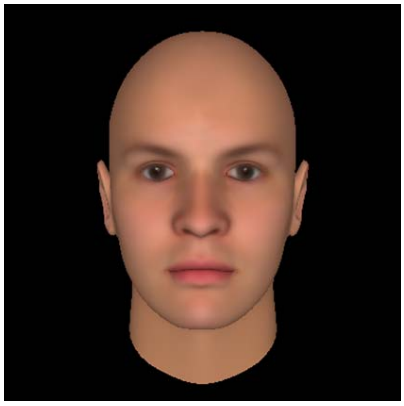

Figura 4. Față ovală

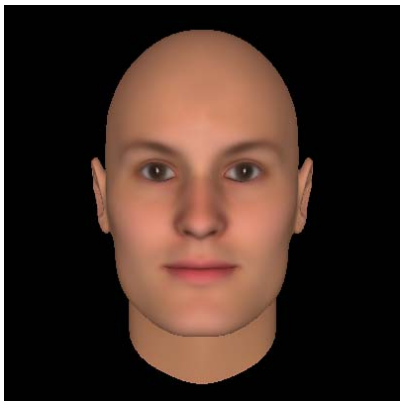

Figura 5. Față dreptunghiulară

Pozele au fost create cu ajutorul programului FaceGen Modeller (http://www.facegen.com/), un program de modelare 3D a feței umane. Printre facilitățile lui principale se numără: posibilitatea de a edita manual fiecare element component al feței, modelarea în funcție de rasă, vârstă, sex. După obținerea înfățișării dorite, fotografia se poate exporta în format 3D sau 2D. Caracteristicile care au fost modelate cu ajutorul acestui modul sunt: ochii, nasul, buzele şi forma feței.

Fereastra pentru vizualizarea profilului unui infractor

Fereastra cu profilul unui infractor conține toate informațiile despre acesta: date personale, caracteristicile feței, toate informațiile disponibile despre infracțiunile comise, precum şi poza sa, dacă o astfel de imagine este disponibilă.

În Figura 6 se poate vedea conținutul profilului unui infractor şi modalitatea de afişare a informațiilor despre acesta.

În cazul în care un infractor nu are nicio fotografie în baza de date, în locul acesteia va apărea o imagine standard, ilustrată în Figura 7. Atunci când cursorul mouse-ului se va muta deasupra ochilor, nasului sau a buzelor, se va deschide o fereastră în care vor fi afişate, pentru infractorul

respectiv, caracteristicile respectivei trăsături a feței (în Figura 7 se pot observa un exemplu cu detaliile despre ochi).

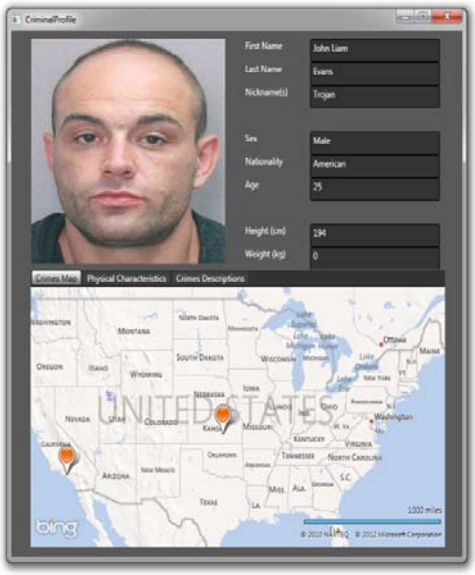

Figura 6. Profil infractor

Vizualizarea profilului unui infractor este posibilă din oricare dintre formularele prezentate anterior.

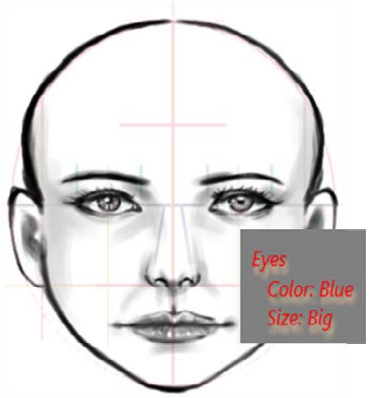

Figura 7. Deschiderea ferestrei cu detalii despre ochi

Fereastra dedicată procesării de texte

Această fereastră permite selectarea unui fişier text în care este descrisă o fărădelege şi sunt specificate semnalmentele infractorului, având ca rezultat afişarea unui tabel cu potenţiali suspecţi pentru noua fărădelege. Utilizatorul îşi poate alege fişierul pe care să-l proceseze şi calea la care să fie generat fişierul de ieşire. Pentru testele pe care le-am realizat până acuma am folosit informaţii în limba engleză preluate din ziare, cu declaraţii ale martorilor sau ale poliţiei.

Restricţiile impuse textului sunt dimensiunea relativ restrânsă şi conţinutul, care trebuie să ofere informaţii relevante despre o infracţiune şi sa fie concentrat pe informaţii concrete.

Datele extrase sunt împachetate într-un fişier XML, pentru folosirea lor ulterioară în cadrul proiectului principal.

Aplicaţia efectuează următoarele tipuri de procesări textuale asupra fişierului încărcat:

- *Identificarea entităţilor de tip nume* - Entităţile care ne interesează momentan sunt cele care desemnează nume de persoane, localităţi şi date, întrucât ne dorim identificarea numelor victimelor, infractorilor acolo unde este cazul, a locaţiei infracţiunii şi a datei la care aceasta a fost comisă.

Pentru realizarea acestei sarcini, s-au folosit resursele şi metodele puse la dispoziţie de binecunoscutul *Gate*, salvând toate entităţile găsite şi frecvenţa apariţiei lor într-un fişier XML.

Această componentă a fost completată cu un modul capabil să identifice numele aceleaşi persoane, chiar dacă acesta este scris în forme diferite. De exemplu, Tairre Lynne West, T. L. West şi Tairee West erau marcate iniţial ca fiind trei entităţi diferite, dar după îmbunătăţire era păstrată o singură entitate, entitatea cu forma cea mai lungă a numelui. În cazul de mai sus se păstrează forma Tairre Lynne West.

De asemenea, s-a folosit biblioteca *StringToTime*, pentru a transpune descrierile textuale ale datelor într-un format standard, care să permită extragerea zilei, lunii şi anului corespunzător. Astfel, dacă în text este specificat faptul că infracţiunea a avut loc “vinerea trecută”, vom putea şti data exactă a acestei zile şi vom putea lucra mai departe cu această informaţie .

În cazul entităţilor de tip *Locaţie*, s-a extras tipul locaţiei desemnate: oraş, ţară, stradă, etc. Acest lucru era necesar pentru a identifica oraşul în

care a avut loc infracțiunea, alte detalii despre țară, stradă, etc. fiind irelevante în contextul creat. S-a folosit aplicația *Gate* de adnotare a textului, care permite adăugarea de tag-uri specifice pentru fiecare tip de entitate descoperită.

- *Rezoluția anaforei* – din lista de lanțuri coreferențiale identificate în text cu ajutorul parserului Stanford NLP, s-au folosit următoarele informații: coreferințele, tipul fiecăreia (nominal, pronominal, propriu), coreferința principală.

Colecția de lanțuri coreferențiale a fost scanată şi organizată în două categorii, care au fost procesate mai departe în moduri diferite: cele care se referă la o entitate de tip nume de persoană (una dintre coreferințe conține numele unei persoane) şi restul lanțurilor, care conțin doar coreferințe nominale şi pronominale.

- *Reguli pentru identificarea relațiilor între cuvinte* – cu ajutorul parserului Stanford NLP s-au extras relații sintactice şi morfologice între cuvintele unei propoziții. Cu ajutorul acestora, pentru o propoziție, s-au construit o colecție de dependențe de diferite tipuri. O dependență este constituită din două cuvinte, între care se stabileşte un anumit tip de relație (sintactică sau morfologică). Primul dintre cuvinte are rolul de guvernant şi cel de-al doilea este numit dependent. Exemple de relații extrase din propoziția: "*Bell, based in Los Angeles, makes and distributes electronic, computer and building products.*": nsubj(make-8, Bell-1), partmod(Bell-1, based-3), root(ROOT-0, makes-8), etc.

Aceste dependințe pe propoziții au fost grupate, pentru a se modulariza procesarea lor:

- *Reguli aplicate pentru extragerea de cunoaştere* – după obținerea entităților de tip nume din text, a lanțurilor coreferențiale şi a dependențelor din fiecare propoziție, s-au creat, pe baza acestora, o mulțime de reguli care să permită identificarea unor anumite tipuri de informații din text.
- *Identificarea rolurilor* – mai întâi, s-au grupat lanțurile coreferențiale în funcție de tipul coreferinței principale. Astfel, lanțurile care referă o persoană sunt analizate pentru identificarea rolurilor fiecărui personaj, în contextul infracțiunii: *răufăcător*, *victimă*, *martor* sau alt tip de persoană. Pentru a identifica un

infractor s-a folosit o metoda bazată pe relațiile gramaticale dintre cuvinte şi coreferințele, ce referă numele persoanei respective.

De asemenea, s-au folosit ca resurse colecții de substantive, verbe, adjective care referă infracțiuni. Exemple de astfel de verbe: *stab, injure, kill, rob, steal, shoot, abuse, assault, attack, beat*, etc.

S-au stabilit tipurile de dependințe care apar în interiorul propozițiilor, atunci când se exprimă faptul că o persoană a comis o fărădelege. Câteva dintre acestea sunt:

- dependențe de tip *nsubj* (subiect-predicat), în care subiectul este un cuvânt din componența numelui personajului respectiv şi verbul exprimă cominterea unei infracțiuni.

Exemplu: Dan Scott severely injured a woman last night, on the street. nsubj(injured, Scott).

- dependențe de tip *nsubjpass* (subiect-verb diateza pasivă), în care subiectul este un cuvânt din componența numelui personajului respectiv şi verbul exprimă faptul că infractorul a fost descoperit, capturat, pedepsit. S-a folosit o colecție de verbe care indică acest lucru.

Exemplu: Dan Scott was arrested Monday morning, at his house. nsubjpass(arrested, Scott).

- dependențe de tip *appos* (substantiv-apoziție), în care substantivul este un cuvânt din componența numelui personajului respectiv şi apoziția conține un substantiv care denumeşte un răufăcător. În acest caz, s-a folosit o colecție de substantive care exprimă acest lucru.

Exemplu: Dan Scott, the criminal that the police has a hard time capturing, claimed another victim. appos(Scott, criminal).

- dependențe de tip *cop* (verb auxiliar-nume predicativ), în care numele predicativ denumeşte un răufăcător.

Exemplu: Dan Scott is the killer that the police was searching for. cop(killer, is).

Dependențele de mai sus sunt căutate în propozițiile care au ca subiect numele unei persoane. S-au găsit o serie de dependențe şi pentru propozițiile în care numele personajului este prezent, dar nu ca subiect al propoziției. Câteva dintre acestea sunt:

- dependențe de tip *agent. Exemplu*: *The young girl was assaulted by Dan Scott.* agent(assaulted, Scott).
- dependențe de tip *dobj. Exemplu*: *The police captured Dan Scott,*

after looking for him for three days. dobj(captured, Scott)

Modalitatea de identificare a victimelor este similară, fiind formulate, în mod evident, alte reguli.

- *Identificarea fragmentelor de text relevante* – s-a făcut pe baza textelor analizate anterior, în care s-au găsit roluri de victime, răufăcători sau martori. Căutarea informațiilor despre caracteristicile fizice ale infractorilor, categoria din care face parte infracțiunea, etc. va fi efectuată numai în propozițiile selectate în urma efectuării acestui pas. Acest lucru a fost realizat pentru a evita identificarea informațiilor din propoziții, care oferă detalii, ce nu au o relevanță atât de mare, pentru atingerea scopului inițial.
- *Extragerea caracteristicilor fizice ale infractorilor* – s-a realizat cu metode specifice următoarelor caracteristici:
    - *Caracteristici ale feței* (forma feței, ochilor, nasului, buzelor, dimensiunea acestora, culoarea pielii, culoarea ochilor, culoarea şi lungimea părului, etc.). Dependențele căutate sunt cele de tip *amod*(substantiv-adjectiv), iar substantivele luate în considerare sunt din următoarele categorii:
        - face, eyes, hair, nose, lips, skin, figure, etc. Exemple: **round** face, **blue** eyes, **full** lips, **solid** figure, **long** hair, etc.
        - shape, color, size, style, length, type, etc. Exemple: **medium** length (hair), **square** shape (face), **blue** color (eyes), **small** size (nose), **afro** style (hair), etc.
    - *Vârstă, înălțime, greutate* – Dependențele avute în vedere pentru realizarea acestei sarcini sunt cele de tipurile:
        - *npadvmod* – Exemple: *The director is 65 years old*. – npadvmod(old, years). *He is 6 feet long*. – npadvmod (long, feet);
        - *num* – Exemple: *The director is 65 years old*. – num (years, 65). *He is 6 feet long*. – num (feet, 6).
- *Categorii de infracțiuni şi arme folosite* - Pentru extragerea acestor tipuri de date, s-au folosit două liste de substantive populate anterior cu posibile categorii de infracțiuni şi tipuri de arme, care au fost căutate în rândul relațiilor ce au în componență substantive.

După procesarea textului, rezultatele sunt împachetate într-un fişier XML, iar caracteristicile extrase din text sunt încărcate în memorie.

Infractorii care prezintă măcar o parte dintre aceste caracteristici sunt selectați şi adăugați într-un tabel, care afişează: numele şi poza răufăcătorului, caracteristicile care se potrivesc şi cele care nu se potrivesc cu cele extrase din text, precum şi un procent de asemănare. Acest procent este calculat prin raportul dintre numărul de caracteristici între care există o potrivire şi numărul total al caracteristicilor descoperite în text. Modalitatea de calcul ar putea fi extinsă, oferind posibilitatea ponderării factorilor, cu scopul rafinării şi îmbunătățirii rezultatelor.

Figura 8 conține o exemplificare a structurii şi conținutului acestui tabel. Pentru accesul uşor către profilul unui infractor care figurează în tabel, utilizatorul trebuie numai să efectueze un click pe poza acestuia. Pe viitor dorim să ponderăm caracteristicile identificate în funcție de relevanța lor, cu scopul de a îmbunătăți predicția finală.

Photo	Name	Sex	Skin Color	Figure Size	Face Shape	Hair Color	Hair Length	Matching Rate
	John Walker	✓	✓	✓	✗	✗	✗	50%
	Mark Mckormick	✓	✓	✗	✗			50%
	Charles Arthur Floyd	✓	✗	✓	✗	✗	✗	33%
	John Liam Evans	✓	✓	✓	✗	✗	✗	50%
	Johnny Lamb	✓	✗	✓	✗	✗	✗	33%

Figura 8: Tabel rezultate potriviri

Momentan, aplicația a fost folosită oferindu-i la intrare articole de ziare preluate de pe Internet, dar ea ar putea fi folosită pe viitor având la intrare declarațiile martorilor sau observațiile celor care anchetează o infracțiune.

4. Concluzii

Lingvistica Computațională este o ramură destul de dificil de abordat a Inteligenței Artificiale, acest lucru datorându-se nivelului sporit de complexitate al limbii fiecărui popor. Progresele care s-au făcut în ultimii

ani permit, însă, folosirea cu succes a uneltelor de procesare existente pentru limba engleză.

Această lucrare şi-a propus să fructifice rezultatele obținute până acum în domeniu, pentru a crea un sistem care să poată extrage dintr-un text care descrie o infracțiune, acele informații care sunt relevante pentru identificarea infractorului.

Pentru atingerea acestui obiectiv, s-a creat contextul în care acest sistem să poată funcționa, adică aplicația principală, care a fost completată şi cu alte funcționalități, pe care le considerăm folositoare şi necesare.

Alegerea unor parametri precum datele care să fie stocate în baza de date au depins de informațiile pe care le-am obținut din ziare. În funcție de sursa acestora, pe viitor, acestea pot fi înlocuite fără a fi făcute schimbări majore structurii aplicației. Schimbările necesare cuprind modificarea schemei fişierelor XML aferente şi, eventual, adăugarea de reguli pentru extragerea respectivei categorii de date.

În aceste condiții, considerăm că aplicația ar putea fi utilizată într-un mediu real, în cadrul diferitelor instituții în care procesarea acestor tipuri de texte este realizată de angajați.

Direcțiile de dezvoltare pentru o astfel de aplicație sunt, în mod evident, numeroase. Proiectul poate fi extins cu posibilitatea procesării unor texte de dimensiuni mai mari, care conțin resurse multimedia (poze, videoclipuri).

Mulțumiri

Cercetarea prezentată în această lucrare a fost finanțată de către proiectul MUCKE (Multimedia and User Credibility Knowledge Extraction), de tip ERA-NET CHIST-ERA, numărul 2 CHIST-ERA/01.10.2012.

Referințe

Bikel, D. M., Miller, S., Schwartz, R., Weischedel, R. *Nymble: a High-Performance Learning Name-finder*. Proc. Conference on Applied Natural Language Processing, 1997.

Borthwick, A., Sterling, J., Agichtein, E., Grishman, R. *Exploiting Diverse Knowledge Sources via Maximum Entropy in Named Entity Recognition*. In: Proceedings of the 6th Workshop on Very Large Corpora, 1998.

Chang, C. H., Kayed, M., Girgis, M. R., Shaalan, K. *A Survey of Web Information Extraction Systems*. IEEE Transactions on Knowledge and Data Engineering, vol. 18, no. 10, Pp. 1411-1428, Oct. 2006,

Cristea, D., Ionita, M., Pistol, I. *Inteligența Artificială*. Editura Universității "Al.I.Cuza", Iaşi, 2007.

Grishman, R. and B. Sundheim. *Message understanding conference- 6: A brief history*. In Proceedings of COLING, 1996.

Hutchins, J. *Retrospect and prospect in computer-based translation*. In Machine Translation Summit VII, 13th-17th September 1999, Kent Ridge Labs, Singapore. Proceedings of MT Summit VII „MT in the great translation era" (Tokyo: AAMT), Pp. 30-44, 1999.

Iftene, A., Trandabăț, D., Toader, M., Corîci, M. *Named Entity Recognition for Romanian*. In Proceedings of the 3th Conference on Knowledge Engineering: Principles and Techniques Conference (KEPT2011). In Studia Universitatis, Babeş Bolyai, Vol. 2, Cluj-Napoca, România, Iulie 4-6, Pp.19-24, 2011.

Masayuki, A., Matsumoto, Y. *Japanese Named Entity Extraction with Redundant Morphological Analysis*. In Proc. Human Language Technology conference – North American chapter of the Association for Computational Linguistic, 2003.

McCarthy, J., Minsky, M., Rochester, N., Shannon, C. A. *Proposal for the Dartmouth Summer Research Project on Artificial Intelligence*. 1955.

Nadeau, D., Sekine, S. *A Survey of Named Entity Recognition and Classification*. 2007.

Peng, F., McCallum, A. *Information extraction from research papers using conditional random fields*. In Information Processing & Management, vol. 42, nr. 963, 2006.

Sekine, S., Grishman, R., Shinnou, H. *A Decision Tree Method for Finding and Classifying Names in Japanese Texts*. In: Proceedings of the Sixth Workshop on Very Large Corpora, 1998.

Shimizu, N. *Extracting Frame-based Knowledge Representation from Route Instructions*. In HLT-NAACLWs. on Computationally Hard Problems and Joint Inference in Speech and Language Processing. Late Breaking Paper. 2006.

Tulbure-Dombi, M., Amariei, D., Iftene, A. *Supravegherea pe Internet: Extragerea de informații de pe Facebook, Twitter, Wikipedia şi Yahoo*. Conferința RoCHI 2010, 139-140, Bucureşti, 2010