

# Informe final: Lematització automàtica de textos valencians antics mitjançant tècniques d'estats finits

Alícia Garrido, Amaia Iturraspe, Sandra Montserrat,  
Hermínia Pastor i Mikel L. Forcada

*Departament de Llenguatges i Sistemes Informàtics,  
Universitat d'Alacant,  
E-03071 Alacant, Spain.*

## Resum

Es presenta la primera versió d'una eina informàtica que llig un text en català antic i retorna un text en què cada mot del text original identificat va acompanyat de una o més anàlisis morfològiques possibles (cada una de les quals indica el lema o la forma canònica del mot i la informació sobre la flexió). Aquesta eina —que analitza milers de mots per segon— es genera automàticament en pocs minuts —usant un programa compilador— a partir d'un diccionari morfològic (un arxiu de text que conté els lemes, els paradigmes de flexió, i els paradigmes de variació gràfica més comuns); açò fa que el sistema es pugui estendre i refinar indefinidament.

## 1 Introducció

L'estudi del corpus lèxic dels textos clàssics de la llengua catalana és fonamental per a un coneixement profund de la llengua, i, en particular, per a comprendre els mecanismes bàsics de la generació lèxica i de la modificació del significat, a més de la gènesi dels dialectes i de les variants textuais i de registre.

Una de les operacions fonamentals per a la determinació del lèxic d'un conjunt d'obres és la lematització. En el present treball, s'ha estudiat l'aplicabilitat de tècniques basades en l'ús de models computacionals d'estats

finits, com ara els transductors lèxics (Roche & Schabes 1997), a l'automatització (total o parcial) del procés de lematització de textos literaris antics en llengua catalana. En tota la discussió s'assumeix que els textos estan emmagatzemats en arxius informatitzats de format adequat (ASCII).

La lematització d'un text és un procés similar a l'anàlisi morfològica del mateix: es tracta d'obtenir, per a cada un dels mots del text, els quals es troben, en general, morfològicament flexionats, la forma canònica o lema, la categoria gramatical, i els trets morfològics (gènere, nombre, persona, temps, mode, etc.). Per a fer aquesta tasca es necessita un analitzador morfològic, és a dir, un programa que usa un diccionari i els paradigmes de flexió de l'idioma i entrega aquesta informació per a cada mot del text.

Aquesta operació és senzilla si el text està escrit en una variant estàndard del text i sobre un vocabulari que estiga cobert pel diccionari. En canvi, quan es tracta de textos literaris antics, trobem, entre altres, els problemes següents:

- el diccionari pot no contenir mots arcaics o d'ús no estàndard presents en el text antic, o pot estar incomplet;
- els paradigmes flexius d'un text antic poden variar respecte dels de l'estàndard (per exemple, en el pretèrit perfet simple);
- la grafia del text antic varia notablement respecte de l'estàndard.

Per tot això, el lematitzador ha de contenir, a més d'un analitzador morfològic per a les formes estàndard de la llengua, algun mecanisme per a poder tractar les variacions gràfiques en què es poden presentar els mots. A més:

- Ha d'incloure el lèxic bàsic del valencià antic (començant, per exemple, amb les tres mil paraules que recullen Costa Clos & Tarrés Fernández (1998)).
- Ha de considerar tots els paradigmes de flexió antiga a l'hora de realitzar l'anàlisi morfològica.
- Ha de ser capaç de tractar robustament les variants gràfiques més comunes respecte de l'estàndard.

De qualsevol manera, s'ha de tenir en compte que el procés de determinació i fixació dels textos antics ja ha comportat una normalització considerable de les grafies. Açò simplifica considerablement el tractament de les variants.

## 2 Metodologia

El lematitzador està construït com un analitzador morfològic basat en transductors d'estats finits (Roche & Schabes 1997), el qual es genera automàticament —usant eines informàtiques ja desenvolupades pels investigadors del grup (Garrido *et al.* 1999)— a partir del diccionari i d'una representació estàtica estàndard dels paradigmes de flexió morfològica i de variació gràfica més comuns (el diccionari i els paradigmes formen l'anomenat *diccionari morfològic*).

La secció 3 explica amb més detall tècnic l'aproximació utilitzada en aquest treball; informalment, podem avançar que un transductor d'estats finits és una màquina idealitzada que es pot trobar en un estat entre un nombre finit d'estats possibles; començant en l'anomenat estat inicial, llig els mots lletra per lletra. L'estat on es trobe en cada moment depèn només de la última lletra llegida i de l'estat on es trobava abans de llegir-la. Cada volta que llig una lletra i canvia d'estat, el transductor pot escriure un o més símbols (per ex., lletres). Si l'últim estat visitat és d'acceptació, es considera que la transducció (traducció) és vàlida. Quan els transductors lèxics s'usen com a lematitzadors, lligen una forma superficial i escriuen el lema i els trets gramaticals corresponents.

### 2.1 Desenvolupament del projecte

L'estudi ha estat, per la durada especificada en l'ordre de la convocatòria, molt curt, i s'ha executat de manera que en cada etapa es garantira l'obtenció de resultats significatius. Aquestes són les etapes bàsiques del treball:

1. Compilació de diccionaris i vocabularis antics.
2. Construcció del diccionari morfològic d'un lèxic català estàndard i generació automàtica, mitjançant un compilador de diccionaris morfològics ja existent (Garrido *et al.* 1999), d'un analitzador morfològic de català estàndard (vegeu-ne més detalls en la secció 3).
3. Construcció dels paradigmes de la flexió antiga del català i incorporació a l'analitzador.
4. Incorporació del vocabulari específic dels textos antics a partir de vocabularis com el de Costa Clos & Tarrés Fernández (1998).
5. Estudi de les variacions gràfiques existents en els textos antics i formulació i formulació en termes de paradigmes de variació gràfica.

6. Incorporació dels paradigmes de variació gràfica a les paraules rellevants del diccionari morfològic.
7. Prova del sistema (amb obres representatives) per a avaluar-lo. El treball es troba actualment en aquesta fase, en la que s'han trobat alguns problemes que no s'han pogut resoldre abans d'aquesta presentació però que estan sent resolts en l'actualitat: hi ha errades en alguns paradigmes de flexió, falten alguns paradigmes complets, falten alguns mots dels vocabularis esmentats, es produeix algun lema espuri perquè s'accepten algunes variacions gràfiques impossibles en un context determinat o perquè hi ha entrades repetides (una per al català modern i altra per al català antic), etc. En el moment de tancar aquest informe el diccionari conté uns 15.000 lemes.

## 3 Detalls tècnics

### 3.1 Anàlisi morfològica

L'anàlisi morfològica llig la *forma superficial* (flexionada i amb possibles variacions gràfiques) de cada mot del text i n'escriu la *forma lèxica*, que consisteix en la forma canònica o *lema* del mot i un conjunt d'etiquetes que n'indiquen la categoria lèxica i les característiques morfològiques. L'anàlisi es basa en dues fonts d'informació: un diccionari dels lemes vàlids de la llengua i un conjunt de paradigmes de flexió.

Una de les aproximacions més eficients a l'anàlisi morfològica usa *transductors d'estats finits* (TEF) (Mohri 1997; Oncina *et al.* 1993). Els TEF són una classe d'autòmats d'estats finits que es descriuran més endavant. Els TEF poden ser usats com a analitzadors morfològics d'una sola passada i es poden implementar molt eficientment.

Hem desenvolupat una eina per a la construcció i el manteniment d'analitzadors morfològics basats en TEF. És un programa compilador que llig un *diccionari morfològic* i escriu un programa en C que implementa un analitzador morfològic compacte basat en TEF que realitza la tasca indicada. Açò permet que les persones expertes en lingüística es puguin concentrar en la descripció del lèxic i de la morfologia de la llengua en qüestió i les allibera d'haver de pensar en detalls d'implementació informàtica.

### 3.2 Transductors d'estats finits

Els analitzadors morfològics d'aquest treball estan basats en *transductors de lletres* (Roche & Schabes 1997), una subclasse dels transductors d'estats

finits; de fet, qualsevol transductor d'estats finits es pot convertir sempre en un transductor de lletres. Un transductor de lletres es pot definir formalment com  $T = (Q, L, \delta, q_I, F)$ , on  $Q$  és un conjunt finit d'estats,  $L$  un conjunt d'etiquetes de transició,  $q_I \in Q$  és l'estat inicial,  $F \subseteq Q$  el conjunt d'estats finals i  $\delta : Q \times L \rightarrow 2^Q$  és la funció de transició (on  $2^Q$  representa el conjunt de tots els subconjunts de  $Q$ ).

El conjunt d'etiquetes de transició és  $L = (\Sigma \cup \{\epsilon\}) \times (\Gamma \cup \{\epsilon\})$  on  $\Sigma$  és l'alfabet dels símbols d'entrada,  $\Gamma$  l'alfabet dels símbols d'eixida i  $\epsilon$  representa el símbol buit. D'acord amb aquesta definició, les etiquetes de transició poden ser de quatre classes:  $(\sigma : \gamma)$ , que indica que es llig el símbol  $\sigma \in \Sigma$  i s'escriu el símbol  $\gamma \in \Gamma$ ;  $(\sigma : \epsilon)$ , que indica que es llig un símbol però no s'escriu res;  $(\epsilon : \gamma)$ , que indica que no es llig res però s'escriu un símbol, i  $(\epsilon : \epsilon)$ , que indica que la transició d'estat succeeix sense llegir ni escriure. Les transicions de l'últim tipus no són necessàries ni convenients en el transductor final, però poden ser molt útils durant la construcció. S'acostuma a representar la cadena buida  $\epsilon$  amb un zero ("0"). Es diu que un transductor de lletres és *determinista* quan  $\delta : Q \times L \rightarrow Q$ , és a dir, quan totes les transicions que ixen d'un estat tenen etiquetes de transició diferents. Noteu que un transductor de lletres que siga determinista respecte a l'alfabet de les transicions  $L = (\Sigma \cup \{\epsilon\}) \times (\Gamma \cup \{\epsilon\})$  pot perfectament ser indeterminista respecte de l'alfabet d'entrada  $\Sigma$ .<sup>1</sup>

Es diu que una cadena de  $w' \in \Gamma^*$  és una transducció d'una cadena d'entrada  $w \in \Sigma^*$  quan hi ha almenys un camí des de l'estat inicial  $q_I$  fins a un estat final en  $F$  les etiquetes de transició del qual formen la parella  $w : w'$  quan es concatenen. En general, pot haver-hi més d'un camí; açò s'ha d'evitar i, de fet, s'elimina parcialment amb la determinització (vegeu més avall). D'altra banda, pot haver-hi més d'una transducció vàlida per a una cadena d'entrada  $w$ . En l'anàlisi morfològica, aquesta situació correspondria a l'*ambigüitat lèxica*). En l'anàlisi morfològica, els símbols de  $\Sigma$  són els que es troben en els textos i els de  $\Gamma$  són els necessaris per a formar els lemes i els símbols especials que representen la informació morfològica com ara <nom>, <fem>, <2p>, etc.

### 3.3 Minimització dels transductors

La definició general dels transductors de lletres és completament paral·lela a la dels autòmats finits no deterministes (AFND) i la dels transductors de lletres deterministes, a la dels automats finits deterministes (AFD) (Hop-

---

<sup>1</sup>És a dir, el transductor es pot trobar en més d'un estat per al mateix prefix del mot d'entrada i pot estar considerant diverses transduccions alternatives, algunes de les quals pot abandonar quan haja llegit un prefix més llarg del mot d'entrada.

croft & Ullman 1979; Salomaa 1973). Per això, els transductors de lletres es poden determinitzar i minimitzar (respecte de l'alfabet  $L$ ) usant els algorismes existents per a AFND i AFD (Hopcroft & Ullman 1979; Salomaa 1973; van de Snepscheut 1993). Les transicions etiquetades ( $\epsilon : \epsilon$ ) es poden eliminar durant la determinització usant una tècnica paral·lela a l'anomenada *clausura*  $\epsilon$ .

En contrast amb altres compiladors de TEF com el de Karttunen (1993), el compilador que es descriu en aquest article construeix transductors de lletres que no tenen cicles (les transicions formen un *graf dirigit acíclic*<sup>2</sup>) i que, a més, tenen un estat final únic. L'absència dels cicles es deu al fet que només es permeten concatenacions i alternances en el diccionari morfològic, però no repeticions. Per a minimitzar el transductor resultant, s'usa un algorisme descrit per van de Snepscheut (1993), que consta de dos passos idèntics que es poden resumir com segueix: en cada pas, les fletxes de les transicions del transductor de lletres s'inverteixen, de manera que l'estat inicial és el final i al revés, i el transductor resultant es determinitza respecte de  $L$  (és a dir, es formen nous estats amb conjunts d'estats vells de manera que la nova  $\delta$  és  $\delta : Q \times L \rightarrow Q$ ). El transductor resultant d'aquest procés doble d'inversió-determinització és mínim (té el mínim nombre d'estats possible). Aquest algorisme és particularment eficient en el cas dels transductors de lletres acíclics. A més, els dos passos tenen interpretacions senzilles: el primer pas uneix terminacions comunes (troba regularitats en la transducció dels sufixos) i el segon uneix començaments comuns (troba regularitats en la transducció dels prefixos).

Els transductors en qüestió són capaços d'entregar totes les anàlisis morfològiques possibles d'un *homògraf*.<sup>3</sup>

### 3.4 Format del diccionari morfològic

El diccionari morfològic és un fitxer de text on es poden inserir lliurement espais en blanc, tabuladors i finals de línia per a millorar-ne la legibilitat. Qualsevol text entre “#” i el final de la línia no es pren en compte i es pot usar per a comentar el diccionari. El diccionari té les tres seccions següents:

1. La *secció de declaració de símbols*, on es declaren explícitament els símbols d'eixida que representen les categories lèxiques o les característiques morfològiques (com ara <adj>, <nom>, <fem> o <sg>).

---

<sup>2</sup>Informalment, un *graf dirigit* té dues parts: un conjunt de nodes, nusos o *vèrtexs* (representats com a punts o cercles) i un conjunt de fletxes cada una de les quals va d'un vèrtex a altre. El graf és *acíclic* si, seguint les fletxes, no es pot passar dues voltes pel mateix *vèrtex*.

<sup>3</sup>Un homògraf és una forma superficial que té dues o més anàlisis morfològiques.

2. La *secció de paradigmes* on es declaren els paradigmes de flexió i els paradigmes de variació gràfica: quan un conjunt de lemes del diccionari comparteixen un patró de flexió o de variació gràfica, es pot donar a aquest patró un nom entre claudàtors (com ara [verbs\_en\_tenir] per als verbs com *retenir*, *abstenir*, *contenir*, etc.) i se'l pot declarar abans, de manera que es pugui usar en el diccionari. Els paradigmes es poden imbricar indefinidament, és a dir, els noms de paradigmes definits prèviament es poden usar per a definir paradigmes nous (el compilador converteix els paradigmes en substransductors que després s'incorporen al transductor complet). Un paradigma és bàsicament un nom per a un conjunt de *transduccions* alternatives: la *transducció* més senzilla és un parell de cadenes (entrada i eixida) com ara “(estim:estimar<verb>)”, anomenat *parella*. Un nom de paradigma, com ara “[pi1]”, és sempre una transducció vàlida i la concatenació d'una o més transduccions, com ara “(estim:estimar<verb>)[V1C]” també és una transducció vàlida.<sup>4</sup>

El compilador tracta les parelles com segueix: si s'usen zeros (“0”) per a alinear explícitament les cadenes d'entrada i d'eixida en una parella de manera que tenen la mateixa longitud, com en “(estim000:estimar<verb>)”, s'entén que els zeros representen la cadena buida i l'alineament es preserva en el transductor; si les dues cadenes tenen longitud diferent i no contenen zeros, com en “(estim:estimar<verb>)”, llavors el compilador les alinea usant una heurística molt simple: la cadena més curta de la parella es completa amb zeros finals.<sup>5</sup> Hem trobat experimentalment que aquesta heurística funciona molt bé, cosa que permet una construcció més ràpida dels diccionaris morfològics. Si, per alguna raó, alguna de les cadenes en la parella ha de ser la cadena buida, com en “(0:<3p><sg>)”, es pot usar un únic zero. Si s'usen zeros però les longituds de la cadena d'entrada i d'eixida no coincideixen, com en “(estim00:estimar<verb>)”, el compilador assumeix que qui ha dissenyat el diccionari volia fer un alineament explícit però no l'ha realitzat correctament; conseqüentment, el compilador dona un avís, no pren en compte els zeros i usa l'heurística indicada més amunt. Cal insistir que els paradigmes no estan restringits a la descripció de terminacions (sufixos), ja que es poden col·locar en qualsevol part d'una transducció.

---

<sup>4</sup>La cadena d'entrada (resp. d'eixida) de la concatenació de dues transduccions simples és la concatenació de les cadenes d'entrada de les cadenes d'entrada (resp. d'eixida). Per exemple, “(est:est)(im:imar<verb>)” és equivalent a “(estim:estimar<verb>)”

<sup>5</sup>Es fa una elecció similar en el compilador de Karttunen (1993).

3. La *secció de diccionari* és simplement un gran paradigma que conté totes les unitats lèxiques del diccionari. Qualsevol transducció vàlida pot ser una entrada del diccionari; per exemple, la persona que ha dissenyat el diccionari pot haver elegit formar un paradigma únic “[tenir]” amb totes les formes del verb irregular català *tenir*; l’entrada del diccionari de *tenir* seria aleshores simplement el nom del paradigma, que es podria usar a més per a definir altres entrades que tenen el mateix patró d’inflexió com ara *retenir* com “(re:re) [tenir]” o *contenir* com “(con:con) [tenir]”.

La figura 1 il·lustra el format del diccionari morfològic.

### 3.5 El compilador

El compilador ha estat desenvolupat en Linux usant *bison* (una versió evolucionada de *yacc*, Johnson 1975) i *flex* (una versió evolucionada de *lex*, Lesk 1975) i consisteix en dos mòduls. El primer mòdul llig el fitxer del diccionari morfològic i combina —usant transicions ( $\epsilon : \epsilon$ ) com a “pegament” on convinga— els transductors parcials que es corresponen amb els paradigmes declarats i amb les entrades del diccionari en un únic transductor que conté un estat inicial i un estat final. Els missatges d’error estan dissenyats per a ajudar el dissenyador del diccionari a corregir possibles errors de format (com ara paradigmes o símbols que s’usen abans d’haver estat definits, parèntesis no equilibrats, etc. El segon mòdul minimitza el transductor resultant (com s’ha descrit en la secció 3.2) i combina el codi resultant amb un esquelet estàndard per a produir un programa en C que fa l’anàlisi morfològica. La velocitat típica de l’anàlitzador és d’uns 20.000 mots per segon en un ordinador basat en un processador Pentium o equivalent a 400 MHz.

## 4 Aplicació al català antic

L’aplicació de les tècniques aplicades més amunt per a construir un lematitzador de textos catalans antics sobre un analitzador per al català modern ja existent<sup>6</sup> exigeix que s’aborden els problemes següents:

- L’addició de lèxic propi del català antic que actualment no s’usa.
- L’especificació de paradigmes de flexió antiga (especialment la flexió verbal) diferents dels actuals.

---

<sup>6</sup>Desenvolupat en el marc projecte d’investigació sobre traducció castellà-català finançat per la *Caja de Ahorros del Mediterráneo*.

```

# Declaració de símbols gramaticals
%symbol <PresInd>; # Present d'indicatiu
%symbol <1p>;      # Primera persona
%symbol <sg>;      # Singular
%symbol <2p>;      # Segona persona
%symbol <pl>;      # Plural
# ... etc.

# Definició dels paradigmes

[ó] > (ó:<sg>)      # noms en
      | (ons:<pl>) ; # -ó

[p1] > (e:<1p><sg>) # present d'indicatiu
      | (es:<2p><sg>)
      | (a:<3p><sg>)
      | (em:<1p><pl>)
      | (eu:<2p><pl>)
      | (en:<3p><pl>);

[ii1] > (a:<1p><sg>) # terminacions de l'imperfet
       | (es:<2p><sg>)
       | (a:<3p><sg>)
       | (en:<3p><pl>) ;

[V1C] > (0:<PresInd>)[p1] # primera conj.
       | (av:<ImpInd>)[ii1]
       | (àveu:<ImpInd><2p><pl>)
       | (àvem:<ImpInd><1p><pl>)
#       ... etc.
;

# ... etc

# Diccionari
%dic

(estim:estimar<verb>)[V1C];
(bat:batre<verb>)[V2C];
(acci:acció<noun><fem>)['o] # acció

```

**Figura 1:** Exemple de diccionari morfològic

- El tractament de les variacions gràfiques pròpies d'un estadi de la llengua no normativitzat.
- Els criteris per a assignar un lema al conjunt de variants d'un mateix mot.

Cada una de les seccions següents es dedica a cada un dels aspectes anteriors.

## 4.1 Elaboració d'un corpus lèxic del català

El corpus lèxic del català antic que es presenta procedeix bàsicament de dues fonts:

- El diccionari de Costa Clos & Tarrés Fernández (1998), que conté unes 3000 entrades, i que està basat en diccionaris com el de Coromines *et al.* (1995) o el d'Alcover & Moll (1984) i en "els glossaris d'algunes edicions de textos medievals" sense declarar explícitament.
- El corpus informatitzat elaborat per Torruella i col·laboradors <sup>7</sup>.

## 4.2 Els paradigmes de flexió antiga

Els paradigmes de flexió verbal antiga s'han construït a partir de les definicions del diccionari de Costa Clos & Tarrés Fernández (1998) i del d'Alcover & Moll (1984).

## 4.3 Els patrons de variació gràfica

Inicialment, s'havia proposat el següent mètode per a tractar les variants gràfiques dels textos: s'implementaria un mòdul d'anàlisi corrector d'errors que s'executaria simultàniament a l'analitzador morfològic. Aquest model permetria afegir automàticament les variacions gràfiques a l'analitzador i el faria capaç de tractar-les simultàniament a l'anàlisi. La major part de les variacions gràfiques es poden tractar com a *operacions d'edició* senzilles (inserció, esborrat o substitució d'una lletra). Algunes d'aquestes operacions són més comunes que altres; si s'assigna un *cost* a les operacions, és possible determinar la seqüència d'operacions d'edició que genera la forma antiga a partir de la forma estàndard amb el cost mínim (Wagner & Fischer 1974).

---

<sup>7</sup>Arxiu provisional de paraules del català medieval elaborat per Joan Torruella, del Departament de Filologia Espanyola la Universitat Autònoma de Barcelona.

L'ús d'operacions d'edició és completament natural en el model de transductors d'estats finits, ja que aquestes operacions es poden expressar com a transicions d'un transductor d'estats finits. A partir de l'estudi estadístic de les freqüències d'aquestes variacions en els textos antics d'interés es podria definir una funció de distància d'edició que permetria proposar el(s) mot(s) estàndard(s) més propers a l'aparegut en el text antic, i integrar-la en l'anàlitzador morfològic (l'anàlisi morfològica correctora d'errors és una tècnica clàssica de la disciplina anomenada *reconeixement [automàtic] de patrons*, en anglès *pattern recognition*). El problema amb aquesta aproximació —que ha estat abandonada— és la gran quantitat d'interpretacions espúries que es generen per a cada forma superficial observada en el text, cosa que redueix considerablement la utilitat del lematitzador.<sup>8</sup> En canvi, s'ha optat per enriquir (manualment) el diccionari amb patrons que indiquen les variacions possibles en el context concret en què es poden produir. Els resultats són molt més satisfactoris.

Els patrons de variació gràfica utilitzats es basen, d'una banda, en els que donen explícitament Costa Clos & Tarrés Fernández (1998) i, d'altra, en les variacions observades per nosaltres mateixos en el corpus lèxic de Torruella i col·laboradors (vegeu la nota 7). Aquests patrons —els quals serveixen per a descriure fenòmens observats freqüentment en els corpus com ara l'elisió, el canvi, la confusió, la reducció, etc.— es divideixen en quatre grups bàsics:

**Patrons de substitució:** el nom d'aquests patrons té la forma “[subst\_*n*]” on *n* fa referència a la forma gràfica adoptada per al lema. Aquests patrons recullen totes les variants atestades de la forma gràfica *n* com a entrada i tenen la forma *n* com a eixida. Per exemple, el paradigma

```
[subst_c_velar]>(c:c)
                |(ch:c)
                |(qu:c)
                |(g:c);
```

indica que la *c* velar del lema pot aparèixer com a *c* (sense variació), com a *ch*, com a *qu* o com a *g* en els textos antics i per tant, es posa en comptes de la *c* velar en els mots que la contenen i on s'ha observat alguna de les variacions: per exemple, el verb *trocar* (foradar) pot aparèixer en textos antics en formes com ara *trochava* (García Sempere 1999); per això, convé que aparega en el diccionari

---

<sup>8</sup>Això és degut al fet que les operacions d'edició no tenen en compte el context de la inserció, de l'esborrat o de la substitució.

com “(tro:tro) [subst\_c\_velar] (0:ar<verb>) [V1C\_car]” per a reflectir aquesta variació. El fet que el lematitzador accepti formes com *troquava* és irrellevant perquè aquestes formes són molt improbables.<sup>9</sup> Les transposicions (per ex., *nr/rn*) es tracten com a substitucions.

Alguns dels patrons de substitució adoptats corresponen a grafies que representen variacions dialectals (per ex., l’aparició de *u* en comptes de *o*).

**Patrons d’inserció:** el nom d’aquests patrons indica la grafia que s’ha d’inserir quan no apareix en les formes superficials antigues. Dos casos típics són

[inser\_h]>(h:h)  
| (0:h);

que insereix una *h* per a produir el lema de formes antigues com *istòria* o

[inser\_s]>(s:s)  
| (0:s);

que insereix una *s* per a produir el lema de formes antigues com *perea*.

**Patrons d’esborrat:** el nom d’aquests patrons indica la grafia que s’ha d’esborrar de la forma antiga per a produir el lema. Per exemple, el patró

[esborrat\_h]>(h:0);

serveix per a eliminar la *h* intercalada en formes antigues com *rahó*.

**Patrons d’acceptació:** Els tres tipus de patrons de variació gràfica anteriors s’usen quan la variació afecta la part comuna a totes les formes flexionades d’un lema (per exemple, totes les formes del verb *raonar* tenen en comú l’arrel *rao-* i l’aparició de la *h* intercalada afecta totes les formes igualment). En aquest cas, el paradigma de variació s’insereix en el diccionari. En canvi, si la variació gràfica afecta formes

---

<sup>9</sup>S’ha optat per generalitzar al màxim els paradigmes, i per això s’han construït paradigmes com: “[substitucio\_t]>(t:t)|(ct:t)|(d:t)|(tt:t);” on l’alternança principal era “(t:t)|(d:t)” però s’ha ampliat per a englobar totes les substitucions que afectaven la *t*. Així en una paraula com: *attendre* (lema *attendre*), s’aplica el paradigma perquè s’estima que les altres possibilitats de lectura (*d*, *ct*) no són possibles en aquest context.

flexionades en posicions que no apareixen en el lema, la variació s'ha de tractar de manera diferent: s'introdueixen dins dels paradigmes de flexió els *patrons d'acceptació*, els quals permeten acceptar variants en les formes superficials flexionades sense produir cap eixida abans de començar a escriure la part corresponent del lema: per exemple, *feya* és una variant comú de *feia* però el lema és *fer*: la variació apareix fora de la part que tenen en comú el lema i la forma flexionada (*f-*); la solució podria consistir, en aquest cas, a introduir el patró d'acceptació

[accep\_y\_i]>(y:0)  
|(i:0);

en el paradigma de flexió de *fer*; per exemple,

(fe:f)[accep\_y\_i](a:er<verb><ImpInd><3p><sg>);

Aquest cas és especialment important en verbs irregulars com l'indicat.

#### 4.4 Criteris de lematització

Els criteris de lematització usats en aquest primer prototipus són provisionals; açò no és problemàtic si es considera que el lema que s'ha de produir es pot canviar de manera molt senzilla en el diccionari morfològic per a ajustar-lo a un nou conjunt de criteris. Afortunadament, en la major part dels casos l'elecció del lema que ha de produir l'analitzador no és problemàtica perquè les formes són simples variacions gràfiques o flexives respecte del lema estàndard. No obstant això, hi ha casos problemàtics. A tall d'exemple, mereix la pena esmentar alguns problemes i les decisions preses:

- Quan algun dels corpus lèxics defineix un lema remetent a un altre que té una flexió diferent, es fan dues entrades diferents en el diccionari, s'associa a cada una el paradigma de flexió corresponent i es fa que cada una entregue un lema diferent (p.e., *sofertar* (1a. conj.) i *sofrir* (3a. conj.)).
- En general, quan el corpus atesta dues formes diferents d'un mateix mot, una més moderna i altra més antiga, relacionades fonològicament (p.e. *aur* i *or*, *pauc* i *poc*, *paubre* i *pobre*, etc.) s'ha optat per produir el lema modern. Una excepció important són els noms en *-ença* (com ara *absença*) per als quals s'ha mantingut aquesta terminació en comptes

de substituir-la per la més moderna en *-ència*, ja que en aquest cas la normativització de la llengua estàndard ha optat per fixar formes que es podrien considerar etimològicament més arcaïques.

- En el cas de les formes reforçades dels demostratius, hi ha formes que semblen indicar una pronúncia fricativa alveolar (*cell*, *acell*) i altres que semblen indicar una pronúncia oclusiva velar (*aquell*). En aquest cas particular s'ha optat per lematitzar totes aquestes formes com a velars i tractar les variacions com a simples variacions gràfiques.

## 5 Presentació del treball

El programa lematitzador es pot provar de dos maneres. La primera consisteix a copiar els arxius del disquet adjunt (preparats per al sistema operatiu DOS) a aquest informe a un directori i executar-hi el programa *morfo*. Si es tecleja *morfo* sense arguments, el programa llig text del teclat i escriu el text lematitzat en la pantalla. Si s'escriu *morfo -f* i el nom d'un arxiu de text, es lematitza aquest text. La segona manera consisteix a accedir, via Internet, a l'adreça <http://www.torsimany.ua.es/plaeva> on es pot provar el lematitzador —en l'estat de desenvolupament en què es trobe— interactivament.

## 6 Treball futur

No volem acabar aquest informe sense indicar les possibles línies de treball futur. Les més immediates són:

- Completar els paradigmes de flexió i corregir els errors que es vagen detectant.
- Ampliar el diccionari amb nous mots per a millorar la cobertura del lematitzador.
- Millorar els patrons de variació gràfica per tal d'evitar generalitzacions que produïsquen lemes espuris per a alguns mots.
- Revisar els criteris de lematització per a fer-los més conformes a les directrius usades pels experts en textos catalans antics.
- Millorar el format d'eixida del lematitzador per a fer-lo més útil per a les persones expertes en català antic que l'hagen d'usar.
- Afegir més característiques dialectals als paradigmes de variació.

Per a més endavant, pretenem:

- Afegir al lematitzador un desambiguador opcional (basat l'estadística dels contextos proporcionats pels mots anteriors i posteriors en el text) per evitar la producció dels lemes que no siguen possibles en el context donat.<sup>10</sup>
- Construir d'un "traductor" català antic–català modern, que produïska un text en català modern equivalent a l'antic.
- Estudiar la possibilitat d'introduir automàticament els patrons de variació gràfica en les entrades concretes del diccionari.

**Agraïments:** Aquest treball hauria estat molt més difícil si Maribel Tarrés i Mercé Costa no hagueren estat tan amables de passar-nos una versió informatitzada del seu vocabulari (Costa Clos & Tarrés Fernández 1998). Agraïm molt especialment l'ajuda tècnica del Prof. Francisco Moreno, del Departament de Llenguatges i Sistemes Informàtics, especialment en detalls del programa compilador i del servidor d'internet. També agraïm l'ajut d'Hèctor Gozávez amb el vocabulari de Torruella.

## Referències

- ALCOVER, ANTONI M., & FRANCESC DE B. MOLL. 1984. *Diccionari català–valencià–balear*. Palma de Mallorca: Moll. (10 vols.).
- COROMINES, JOAN, JOSEPH GULSOY, MAX CAHNER, CARLES DUARTE, & ÀNGEL SATUÉ. 1995. *Diccionari etimològic i complementari de la llengua catalana*. Barcelona: Curial–La Caixa. (9 vols.).
- COSTA CLOS, M., & M. TARRÉS FERNÁNDEZ. 1998. *Diccionari del català antic*. Barcelona: Edicions 62.
- GARCÍA SEMPÈRE, M. 1999. *La versió catalana medieval dels tractats de falconeria Dancus Rex i Guillelmus Falconarius*. Alacant: Publicacions de la Universitat d'Alacant.
- GARRIDO, A., A. ITURRASPE, S. MONTSERRAT, H. PASTOR, & M.L. FORCADA. 1999. A compiler for morphological analysers and generators based on finite-state transducers. *Procesamiento del Lenguaje Natural* 25:93–98.

---

<sup>10</sup>En l'actualitat es disposa d'aquesta similar en el context del projecte sobre traducció automàtica esmentat en la nota 6; només caldria adaptar-lo al català antic.

- HOPCROFT, J. E., & J. D. ULLMAN. 1979. *Introduction to automata theory, languages, and computation*. Reading, MA: Addison–Wesley.
- JOHNSON, S.C. 1975. Yacc – yet another compiler compiler. Technical Report Technical Report 32, AT&T Bell Laboratories, Murray Hill, N.J.
- KARTTUNEN, LAURI. 1993. Finite-state lexicon compiler. Technical Report Technical Report ISTL-NLTT-1993-04-02, Xerox Palo Alto Research Center, Palo Alto, California.
- LESK, M.E. 1975. Lex — a lexical analyzer generator. Technical Report Technical Report 39, AT&T Bell Laboratories, Murray Hill, N.J.
- MOHRI, MEHRYAR. 1997. Finite-state transducers in language and speech processing. *Computational Linguistics* 23(2):269–311.
- ONCINA, JOSE, PEDRO GARCÍA, & ENRIQUE VIDAL. 1993. Learning sequential transducers for pattern recognition interpretation tasks. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 15:448–458.
- ROCHE, E., & Y. SCHABES. 1997. Introduction. In *Finite-State Language Processing*, ed. by E. Roche & Y. Schabes, 1–65. Cambridge, Mass.: MIT Press.
- SALOMAA, ARTO. 1973. *Formal Languages*. New York, NY: Academic Press.
- VAN DE SNEPSCHEUT, J.L.A. 1993. *What computing is all about*. New York: Springer-Verlag.
- WAGNER, R.A., & M.J. FISCHER. 1974. The string-to-string correction problem. *Journal of the ACM* 21(1):168–174.