Recursos d'ajut a l'edició

Ortografia, gramàtica i estil

Martí Quixal Martínez
Francesc Benavent Portabella
Javier Gómez Guinovart

PID_00233514

Temps de lectura i comprensió: **4 hores**

UOC
Universitat Oberta de Catalunya

Índex

Introducció

La redacció de textos és una de les activitats més freqüents quan treballem amb ordinador. Gràcies als programes de processament de textos, l'ordinador esdevé una eina molt valuosa per a la redacció de textos. Entre d'altres conté eines com la col·locació automàtica del guionet a les paraules a final de línia, el còmput de paraules, el control dels canvis efectuats en diferents versions d'un mateix document, o, també, la verificació automàtica de llengua i estil. Dedicarem aquest mòdul a l'estudi de les eines de verificació ortogràfica, gramatical i d'estil automàtiques. La verificació automàtica té com a base un programari que utilitza tècniques de processament del llenguatge natural i pretén ajudar a millorar la qualitat final dels textos que escrivim.

Objectius

Els objectius que assolireu amb els materials que componen aquest mòdul són els següents:

1. Delimitar l'abast i els objectius de la verificació lingüística automàtica de l'ortografia, la gramàtica i l'estil, com a ajut a la redacció de textos.

2. Analitzar les característiques dels errors ortogràfics, gramaticals i d'estil, especialment d'aquells més freqüents quan escrivim amb ordinador.

3. Examinar les tècniques emprades en la verificació automàtica de l'ortografia, gramàtica i estil en el marc general del processament de textos.

4. Conèixer els aspectes essencials del procés de creació d'un verificador automàtic.

5. Conèixer i entendre els recursos bàsics d'un verificador automàtic de codi obert.

1. La revisió de textos i la verificació automàtica

Els verificadors lingüístics automàtics són un tipus de programari que serveix per comprovar l'adequació d'un text a la normativa o als criteris de redacció per als quals han estat dissenyats. Des del punt de vista de l'usuari o usuària finals, la verificació consta de dues fases diferenciades: d'una banda, la identificació de les paraules del text susceptibles de contenir un error segons la normativa o de reflectir una discrepància o incoherència respecte dels criteris lingüístics o d'estil establerts; de l'altra, la selecció d'una proposta de correcció – sempre que això sigui possible – o la reformulació d'un fragment del text.

1.1. La revisió de textos

La revisió de textos, feta per persones, s'entén com l'adequació d'un text "a les normes gramaticals de la llengua en què s'ha produït", i llavors es parla de **revisió gramatical**; i també com l'adequació d'aquest text "a la situació comunicativa de què forma part", i llavors es parla de **revisió d'estil** (Costa, pàg. 11). Segons Costa i altres (pàg. 64), la revisió de textos consta de diverses fases: lectura parcial, identificació d'errors freqüents i aspectes crítics, consultes a obres de referència, intervencions sobre el text i revisió general. Així mateix, és una activitat que pressuposa un domini de la llengua en totes les variants dialectals, rigor, objectivitat, coneixement de les obres de referència lingüística i dels recursos tipogràfics i convencions més esteses. També és convenient tenir un bon nivell de cultura i, no menys, prudència.

Nota

Els termes *revisió* i *correcció* de textos són denominacions comuns per fer referència a una mateixa tasca.

J. Costa i altres (2006). *Curs de correcció de textos orals i escrits: pràctiques autocorrectives* (pàg. 5-6).

Evidentment, la verificació de textos automàtica és una activitat poc comparable a la revisió de textos feta per correctors o correctores professionals, però, en canvi, proporciona un ajut substanciós a les persones que diàriament redactem textos que han de reunir un requisits formals mínims i no tenim l'oportunitat de fer-los passar per un procés de revisió manual fet per un tercer.

Seguint en part la distinció proposada per Costa i altres (2006), distingirem entre verificació ortogràfica i gramatical (revisió gramatical) i verificació d'estil (revisió d'estil).

1.2. Origen, descripció i classificació dels errors

En aquest subapartat examinarem les característiques principals dels errors ortogràfics, gramaticals i d'estil propis de l'escriptura amb ordinador.

Els errors o inconsistències que fem quan escrivim un text amb ordinador poden ser motivats per un desconeixement de la norma o bé per una distracció. En el primer cas, en què no sabem com s'ha d'escriure correctament la paraula, expressió o signe gràfic en qüestió, es parla d'*errors de competència*; en el segon cas, en què sabem com s'ha d'escriure la paraula, expressió o signe gràfic en qüestió però, tot i així, ens errem, es parla d'*errors d'actuació*.

Els errors

La revisió i la verificació de textos no impliquen necessàriament la identificació d'errors, també afecta les inconsistències formals o de contingut. Aquí, per simplificar, parlarem d'*errors* per referir-nos a tots dos, i farem la distinció quan sigui rellevant per a l'explicació.

Els **errors de competència** tenen un vessant individual, ja que no tothom té el mateix nivell de coneixement de la normativa ni les convencions lingüístiques o estilístiques. Tanmateix, hi ha factors lingüístics que n'afavoreixen l'aparició, com ara el grau de disparitat entre ortografia i fonètica d'una paraula (**dons* per *doncs*), les discrepàncies entre normativa i parla (**coneixo* per *conec*), la interferència amb la normativa o el lèxic d'altres llengües (**tràfic* per *trànsit*, interferència de sentit amb el castellà *tráfico*, **el síndrome* per *la síndrome*), o la baixa freqüència d'ús d'una paraula.

Nota

Al llarg de tot el document farem servir l'asterisc (*) per indicar que una frase o expressió lingüística és incorrecta.

Els **errors d'actuació** poden reflectir aspectes fonològics dels errors de la parla (**desmolaritzar* per *desmoralitzar*, amb intercanvi del tret fonològic de localització entre les consonants líquides, /ɾ/ i /l/), poden provenir d'una distracció visual (**problement* per *probablement*, amb el segment *blement* iniciat després la lletra *o* i no pas després de la lletra *a*), poden ser fruit d'un mal ús del teclat (**escriptutra* per *escriptura*, per haver premut alhora dues tecles pròximes en el teclat), d'una modificació en el text (si acabem fent una modificació en un part del text que requereix una modificació en una altra part del text però ens n'oblidem, per exemple, passem de *El nen llegia un conte* a **El nena llegia un conte* en lloc de *La nena llegia un conte*).

Independentment de les causes cognitives o mecàniques d'un error, els errors es poden classificar segons diversos criteris. A continuació, exposem alguns dels criteris que s'empren des d'una perspectiva computacional:

1) Errors que resulten o no en paraules: una mala redacció o mecanografiat poden fer-nos fer un error que resulti en una seqüència de lletres que no siguin una paraula (**cassa* per *casa*), o en una seqüència de lletres que sí que en sigui una paraula (#*casa* per *caça*).

Nota

Farem servir el símbol coixinet (#) per indicar frases, expressions o paraules que tot i ser correctes en termes estrictament formals no ho són en termes semàntics o pragmàtics.

2) Tipus de transformació gràfica que ha patit la paraula o expressió afectada: se sol parlar de quatre transformacions bàsiques (també conegudes com a *transformacions de Damerau*). La inserció d'una o més lletres (**meditterrani* per *mediterrani, #té format quadrada* per *té forma quadrada*); la supressió d'una o més lletres (**exaurir* per *exhaurir, #mol interessant* per *molt interessant*); la substitució d'una lletra per una altra (**reb* per *rep, #pa de plat* per *pa de blat*); i la transposició de dues lletres adjacents (**perposició* per *preposició*). Aquestes transformacions també es poden aplicar a nivell de paraula sencera, no només de lletra o caràcter: supressió d'una paraula (**Li demano tingui comprensió* per *Li demano que*

Paraula

La definició de *paraula* en verificació automàtica sovint es redueix a una seqüència de lletres entre dos espais o un espai i un signe de puntuació, que és la definició més freqüent en processament del llenguatge natural.

tingui comprensió), inserció d'una paraula (**Pensa en que has de venir* per *Pensa que has de venir*) o substitució d'una paraula per una altra (**No vindrà doncs està malalt* per *No vindrà perquè està malalt*).

3) Errors que afecten la frontera entre paraules: tenim l'error per encavalcament, que és la supressió de l'espai entre dues paraules (**elnoi* per *el noi*), i l'error per partició, una separació indeguda mitjançant un espai (**carr etera* per *carretera*).

4) La naturalesa lingüística de l'error: errors en la grafia (**reduit* per *reduït*, **barcelona* per *Barcelona, *trentadós* per *trenta-dos*), en la morfologia (**volguer* per *voler*, **lis diré* per *els diré*), en la sintaxi (**abans de que vingui* per *abans que vingui, *el síndrome* per *la síndrome*), en la semàntica (#*tràfic* per *trànsit*) o en la pragmàtica (#*xoriço* per *lladre* en un text formal, o #*nogensmenys* per *en canvi* dit en un diàleg de bar d'una telenovel·la).

5) Abast de l'error: els errors (ortogràfics i gramaticals) poden diferenciar-se, entre els que poden ser detectats sense tenir-ne en compte el context (**obbvi* per *obvi*, **coneixo* per *conec*) i els que només poden ser detectats tenint en compte el context (**lis* per *els* és incorrecte, però *lis* és correcte a *flor de lis*, **és* per *es* és incorrecte, però no a *aquesta és grossa*).

La classificació dels errors segons aquests o altres criteris tipològics serà convenient en funció de la utilitat perseguida. Per exemple, la distinció entre el tipus de transformació no serà útil per explicar a l'usuari o usuària d'una aplicació final la causa de l'error, però en canvi serà útil per fer un estudi que ens permeti predir, a partir de tenir en compte els tipus de transformacions més freqüents, millors propostes de correcció. En canvi, la classificació per naturalesa lingüística de l'error ens pot ajudar a proporcionar un missatge d'error que acompanyi l'alarma disparada per l'eina i que faci comprendre el criteri lingüístic o d'estil pel qual es considera que hi ha un candidat a error.

Una distinció que, com veurem més endavant, és determinant per triar la tècnica emprada per a la detecció de l'error a l'hora de dissenyar un algorisme per a la verificació automàtica de textos és la distinció entre errors que podem dir que ho són sense tenir en compte el context, i errors que només podem dir si ho són o no tenint en compte el context.

1.3. Verificació ortogràfica i gramatical, i verificació d'estil

Entenem per *verificació ortogràfica i gramatical* la comprovació de tots aquells errors relacionats amb l'incompliment de la normativa lingüística de la llengua del text. Per *verificació d'estil* entenem la comprovació de tots aquells aspectes (errors o inconsistències) que incompleixen criteris i convencions relacionats amb el registre, la varietat dialectal o, per exemple, un estil corporatiu. En tots dos casos, si escau, s'inclourà la generació d'una proposta de correcció o millora.

A continuació presentem una taula amb una llista d'exemples d'allò que considerem errors (**ERR**) i inconsistències (**INC**) ortogràfics, gramaticals i d'estil – no són necessàriament errors tractats pels verificadors automàtics.

Vegeu també

Els verificadors automàtics es tracten al subapartat 3.1.

Taula 1. Classificació dels problemes de redacció segons tipus i àmbit

Tipus	Àmbit	Exemple
ERR	ORTO	**1)** Deixem la **m'suica** barroca per sentir [...]. **2)** T'agradarà **l'oba** que hem preparat [...]. **3)** Ens fa molta **il.lusió** que passeu [...]. **4)** El que vull assegurar **es** la possibilitat de recuperar dades [...].
ERR	GRAM	**5) Aniré** directe després de deixar la nena. **6)** Va anar a recollir els **seves** germans a l'estació. **7) L'hi** van dir que efectivament hi era. **8)** Tu **contaves** que nosaltres vinguéssim, oi? **9)** Hem trobat molt de **tràfic** a la N-340.
ERR	EST	**10)** Benvolguda Dra. Vives [...] **Apa! Fins aviat!** **11)** Hem vist: **garses orenetes pinsans i xoriguers**.
INC	ORTO	**12)** Si **duu** tres taronges, ja en **du** masses.
INC	GRAM	**13)** Trob que portam sa **teva** sabata en es maleter.
INC	EST	**14)** Les "**meves**" són 'meves' i les 'teves' són 'meves' també.

No hi posem un asterisc per indicar que es tracta de frases amb errors o inconsistències perquè totes elles en tenen. Hi apareixen en negreta.

És important destacar que la frontera entre el que és un error i el que és una inconsistència no sempre és prou clara i que sovint un error pot enquibir-se en més d'una categoria.

Així mateix, hem de tenir en compte que els criteris de classificació en errors o inconsistències que un podria establir seguint les recomanacions normatives o dels llibres d'estil no tenen per què coincidir amb la distinció entre errors i advertiments necessària en el moment de la implementació del verificador. Com veurem més endavant, la fiabilitat del programari o simplement criteris funcionals o d'usabilitat seran el que farà decidir entre missatges d'error o advertiments.

1.4. Definició de la tasca de verificació automàtica

La tasca de verificació automàtica se sol subdividir en tres fases. Per explicar-les ho farem a partir d'un exemple. Suposem que volem verificar automàticament la frase de l'exemple número 4 de la taula 1:

**El que vull assegurar* es *la possibilitat de recuperar dades*.

en lloc de:

El que vull assegurar és *la possibilitat de recuperar dades*.

El primer que cal és localitzar l'error. Per **localització de l'error** entenem l'acció d'identificar la posició en el text en què aquest comença i acaba. La localització servirà posteriorment per poder destacar visualment a l'usuari o usuària final l'error que es vol fer notar. Així en el nostre exemple l'error constaria d'una sola paraula (*es*) que va del caràcter número 23 al 24 de la frase, aquí subratllats:

**El que vull assegurar es la possibilitat de recuperar dades*.

Cal tenir en compte que la localització de l'error no sempre coincidirà amb el context necessari per detectar-lo. Així, per detectar aquest error, ens caldrà comprovar si, donat el seu context lingüístic, la forma *es* té més o menys probabilitats de tenir una lectura verbal (tercer persona singular del verb *ser*, per tant, *és*) o de tenir una lectura pronominal (pronom feble de tercera persona, per tant, *es*).

La segona fase del procés és la **classificació de l'error**, que consisteix a assignar un codi a la paraula o seqüència de paraules identificades com a errònies. Es tracta de categoritzar el problema detectat en una o més classes que permetin triar posteriorment el missatge per a l'usuari o usuària finals. En el nostre exemple, podríem identificar l'error com a:

- *error ortogràfic*,
- *error d'accent diacrític*,
- *error per influència del castellà* (que no posa accent a la forma *es*).

Així n'hi podríem anar afegint tants com fossin possibles (tècnicament) i alhora rellevants per a l'aplicació final del programari.

El darrer pas és la **generació d'una proposta de correcció**. És el procés mitjançant el qual el programari pot oferir una alternativa de millora. En el nostre exemple, es tractaria de proposar la substitució de la forma *es* per *és*.

A l'hora de la veritat aquestes tres fases tampoc no són tasques tan independents com sembla. Així, la localització i la classificació de l'error sovint van lligades, ja que per poder detectar un error s'han de tenir en compte tant els

elements lingüístics que el conformen (una o més lletres, una o més paraules, etc.) com els elements lingüístics que l'envolten, el context. Per exemple, per a la detecció d'errors com ara el de l'exemple 2 de la taula 1 (*t'agradarà *l'oba*), només cal identificar que la seqüència de lletres *oba* no pertany al conjunt de paraules de la llengua catalana. Fet això, podem dir que es tracta d'un error ortogràfic (classificació) a la quarta paraula de la frase (localització).

Així mateix, per detectar un error com ara el de l'exemple 6 de la taula 1 (**Va anar a recollir els* ***seves*** *germans*) caldrà segmentar el text en paraules, assignar a cada una d'elles la lectura més plausible en el context, i detectar que hi ha un error de concordança en el sintagma nominal format per la paraula *seves* i les paraules anterior i posterior. Això implica l'ús d'eines de processament del llenguatge natural.

Aquest exemple (el 6 de la taula 1) és interessant perquè ens permet veure també com la generació de propostes de correcció pot requerir sovint la reutilització d'informació lingüística derivada de la detecció i classificació de l'error. En aquest exemple, aprofitaríem els lemes de les paraules que conformen l'error, i les informacions morfològiques de categoria gramatical, gènere i nombre per poder generar propostes alternatives. Gràcies a una anàlisi que proporcioni la informació que presentem a la taula 2, podrem saber on comença i acaba l'error i quina mena d'error és. El pas següent seria, doncs, generar alternatives a la seqüència **els seves germans*, que, gràcies a la informació de lema, categoria gramatical, gènere i nombre podríem concretar en:

- *els seus germans*,
- *les seves germanes*.

Taula 2. Nivells d'anàlisi lingüística automàtica per a l'exemple **a recollir els seves germans* (només dels mots rellevants per a l'explicació)

Nivell descriptiu						
Identificador	...	3	4	5	6	7
Forma	...	a	recollir	els	seves	germans
Lema	...	a	recollir	el	seu	germà
Categoria gramatical	...	Prep	Verb	Art	AdjPoss	Nom
Gènere				Masc	Fem	Masc
Nombre				Plu	Plu	Plu

2. La verificació automàtica de textos

La verificació automàtica de textos és un procés informàtic en què el contingut lingüístic d'un document és revisat automàticament amb l'objectiu de detectar-hi errors i, si és possible, suggerir-ne la correcció.

Per tant, en la verificació automàtica podem diferenciar dos processos independents: d'un costat, la detecció de l'error (que equivaldria a les fases de localització i classificació explicades anteriorment) i, de l'altre, la generació d'una proposta de correcció.

En principi, aquesta descripció és vàlida independentment del nivell lingüístic al qual pertanyi l'error, però l'estat actual del tractament computacional del llenguatge fa que, quan és aplicada a **textos no restringits**, s'hagi de limitar la verificació automàtica als nivells tipogràfic, ortogràfic, gramatical (excloent sovint aspectes semàntics o pragmàtics) i estilístic.

Cal tenir en compte que, tot i tenir el mateix objectiu que la correcció de textos, la verificació automàtica es basa en una anàlisi superficial i, per tant, utilitza unes tècniques molt diferents de les estratègies (o destreses) que poden emprar els i les professionals que revisen un text.

2.1. Tècniques de verificació automàtica

Des del punt de vista informàtic, l'anàlisi lingüística del text consisteix en l'exploració d'una seqüència d'elements textuals, com són lletres, síl·labes, paraules o oracions. La unitat operativa concreta es pot triar segons el nivell d'anàlisi lingüística, però el més habitual és la paraula. Les tècniques de verificació poden classificar-se en dos grups: tècniques no contextuals i tècniques contextuals.

1) Tècniques no contextuals. Es consideren tècniques no contextuals aquelles que per determinar la correcció o no d'un element lingüístic analitzen únicament les propietats d'aquest element, és a dir, les que es basen exclusivament en informació de caràcter intern. En aquest cas, la forma de l'element és determinant per decidir si és erroni o no. Per exemple, la forma de la paraula *gaxt* és suficient per determinar que es tracta d'un error, o més exactament, d'un element lingüístic que no forma part del conjunt de paraules de la llengua catalana.

2) Tècniques contextuals. Es consideren tècniques contextuals aquelles que determinen la correcció o no d'un element lingüístic a partir del context en què apareix, és a dir, a partir d'informació, que tot i ser local, és externa. En aquest cas, la forma de l'element és tan important com les propietats dels elements que l'envolten. Per exemple, el context de la paraula *nen* a l'oració *Els nen jugaven* és necessari (i suficient) per determinar que es tracta d'un error.

3) Errors i advertiments. Independentment del tipus de tècnica utilitzada en la detecció, contextual o no contextual, cal distingir entre allò que es mostrarà com a error i allò que es mostrarà com a advertiment. En el primer cas el programari informa que aquella paraula és incorrecta i suggereix una possible correcció. En el segon cas es visualitzarà un advertiment que informa que aquella paraula podria ser incorrecta o presentar una inconsistència; també es poden explicar els contextos en què podria ser usada l'expressió i els contextos en què no.

El criteri per determinar si una estructura lingüística ha de mostrar-se com a error o com a advertiment és essencialment la fiabilitat tecnològica. Si la tècnica utilitzada és capaç de determinar amb prou fiabilitat, sense intervenció humana, que es tracta d'una estructura incorrecta, es mostrarà com a error. Si la tècnica utilitzada no pot fer una predicció fiable de la intenció original de l'autor, o no pot desambiguar l'accepció usada sense preguntar a qui redacta el text, s'indica com a advertiment i es deixa a criteri de l'autor o autora del text la possible correcció. Noteu que aquesta distinció no té necessàriament en compte la distinció que fèiem al subapartat anterior entre errors i inconsistències.

Això no és massa diferent del que fan els correctors i correctores humans quan, en no poder determinar el significat que es volia transmetre en l'original o en no comprendre el sentit d'una determinada paraula o expressió, el marquen com a dubte per consultar-ho amb l'autor o autora del text.

Per exemple, tot i que la paraula *tràfic* és correcta si es refereix a l'intercanvi de mercaderies, és freqüent utilitzar-la erròniament per referir-se al flux de vehicles i, en aquest cas, caldria substituir-la per *trànsit*. La dificultat de determinar el sentit en què s'usa fa recomanable indicar-ho com un advertiment que contingui l'explicació corresponent.

Per tant, en aquelles situacions en què les tècniques detectin un ús potencialment incorrecte, però no tinguin prou capacitat per reduir o eliminar els falsos positius, és aconsellable abstenir-se de tractar-ho com a error i en canvi fer-ho com a advertiment, de manera que només s'indiqui a l'usuari o usuària que seria convenient revisar aquella estructura.

En les seccions següents veurem exemples concrets de les tècniques que habitualment s'utilitzen per verificar errors ortogràfics, gramaticals i d'estil. Tot i que en cada un dels tipus de verificació posarem l'accent en alguna tècnica

concreta, cal tenir en compte que no hi ha cap correspondència directa entre un tipus d'error i un tipus de tècnica de detecció. La verificació d'un determinat error o de determinats tipus d'errors pot plantejar-se des d'angles diferents i amb objectius diferents.

Cal tenir en compte, però, que sovint s'associa correcció ortogràfica amb tècniques no contextuals, i correcció gramatical i d'estil amb tècniques contextuals. Això ve del fet que sovint s'associa la verificació ortogràfica a la verificació d'errors que resulten en no-paraules, fet que té una certa lògica si es té en compte que és el tipus d'error més fàcil de tractar (i que efectivament es tractava) en els verificadors lingüístics "de primera generació".

2.2. La verificació ortogràfica automàtica

Idealment, la verificació ortogràfica automàtica hauria de permetre de determinar si l'ortografia d'una determinada paraula és o no correcta. Per fer-ho només caldria comparar la forma que apareix al document amb la forma correcta de la paraula que previsiblement l'autor o autora tenia intenció d'escriure, amb el repte que això comporta.

Donada la impossibilitat de conèixer la intenció original, la verificació automàtica ortogràfica ha de plantejar-se amb uns objectius més modestos. Per exemple, determinar si cada una de les formes (paraules) d'un document es corresponen amb alguna de les formes pertanyents a la llengua. Per això, les tècniques de verificació ortogràfica per al tractament d'errors que resulten en no-paraules acostumen a basar-se en la comparació del candidat a error amb una llista de paraules correctes emmagatzemada en forma de diccionari. Per a aquest problema són adequades les tècniques no contextuals, amb les quals únicament es té en compte la forma de la paraula i no pas el context en què apareix.

2.2.1. Emmagatzemament del diccionari

La llista de paraules correctes d'una llengua és molt extensa habitualment se situa en l'ordre dels centenars de milers de formes, i fàcilment pot arribar a uns pocs milions si s'hi afegeix lèxic especialitzat i noms propis. Per això, la idea aparentment simple d'emmagatzemar totes les paraules correctes i identificar com a errors les paraules que no apareguin a la llista presenta algunes dificultats tècniques. Això, que és cert per a llengües fusionals com ara el català, el castellà o l'alemany, no ho és per a llengües aglutinants com el basc, el turc o el finès per a les quals és pràcticament impossible de generar una llista finita de paraules.

Bàsicament hi ha dues estratègies a l'hora de resoldre aquest problema: la utilització de **lemaris** i la utilització de **formaris**, que correspon a dues maneres diferents de codificar un **lexicó**. Les preferències cap a una o altra són deter-

minades en cada moment per les característiques tècniques dels ordinadors utilitzats i per la importància assignada a l'eficiència a l'hora d'estalviar memòria o de maximitzar la velocitat d'accés al lexicó.

1) Lemaris: diccionaris implícits

En el moment de l'aparició dels ordinadors personals, la restricció tècnica més important dels sistemes de detecció d'errors ortogràfics amb tècniques no contextuals era la memòria: cap ordinador personal no podia emmagatzemar totes les formes flexionades ens els pocs megabytes de memòria disponibles que tenia. En aquests casos la solució idònia combina la utilització d'un diccionari d'arrels (lemes) i d'un conjunt de regles morfològiques que representen els diferents paradigmes flexius.

A l'hora de determinar si una paraula pertany a la llengua, els sistemes basats en lemaris realitzen una anàlisi morfològica de cada paraula, comproven que l'arrel corresponent pugui ser flexionada segons els paradigmes de flexió predefinits, i en cas afirmatiu la paraula es dóna com a bona. En certa manera, aquesta solució basada en lemaris conté implícitament la totalitat de les formes d'una llengua, i les genera dinàmicament quan necessita validar-ne una.

Aquest sistema presenta com a avantatges la reducció en memòria del diccionari i una major facilitat a l'hora d'obtenir la llista de lemes correctes. En canvi, presenta com a inconvenients el risc de la sobregeneració, de manera que consideri com a correctes paraules que no ho són, i, sobretot, el cost computacional que suposa fer una anàlisi morfològica per a cada paraula, cosa que en limita l'aplicació en situacions en què el temps de resposta és crític. En canvi, aquesta tècnica és l'única viable per a llengües aglutinants com ara el basc, el turc o el japonès (Ido 2003).

2) Formaris: diccionaris explícits

A mesura que la potència dels ordinadors personals ha augmentat ha aparegut l'opció real d'utilitzar diccionaris de formes gràcies sobretot a la reducció del cost de la memòria RAM i l'aparició de discs amb milers i milions de bytes de memòria.

A partir d'una llista de formes flexionades possibles es crea el que es coneix com a **formari**. Aquest diccionari de formes inclou "totes" les formes (vegeu els dos primers paràgrafs de l'apartat 2.2.2) d'una llengua i permet determinar fàcilment si una paraula escrita pertany o no a aquest conjunt. En cas de no aparèixer a la llista es considera que es tracta d'un candidat a error i s'indica com a tal.

Fins i tot en els casos en què la memòria necessària no sigui un problema, cal garantir la rapidesa d'accés, és a dir, el temps necessari per comprovar si una paraula és al diccionari. Si la llista s'hagués de recórrer seqüencialment ni els

ordinadors personals més ràpids podrien revisar ortogràficament un text en pocs segons. Afortunadament hi ha tècniques d'indexació, com els **TRIE** (derivat de "reTRIEval"), que permeten reduir considerablement aquests temps. Per exemple, a partir d'un diccionari ordenat alfabèticament una cerca binària pot trobar qualsevol paraula 10.000 vegades més de pressa que una cerca lineal; i mitjançant índexs de *hash* (llistes associatives) es poden aconseguir temps d'accés constants independents del volum del diccionari, cosa que permet multiplicar per 100 la velocitat de la cerca binària.

Així doncs, tot i que els formaris requereixen més memòria i una feina prèvia considerable per a la preparació de recursos, el fet que puguin revisar centenars de milers de paraules per segon fa que siguin una solució més ràpida. Això els converteix en l'opció recomanable en entorns interactius d'edició de textos per a llengües fusionals com ara l'anglès o qualsevol de les llengües romàniques.

2.2.2. Limitacions de la verificació ortogràfica automàtica

Més enllà de les dificultats tècniques a l'hora d'emmagatzemar i consultar el diccionari, les principals limitacions d'aquest enfocament són les inherents a la simplificació de considerar:

1) que podem generar una llista de totes les paraules de la llengua i

2) que una paraula és ortogràficament incorrecta si no apareix en una llista tancada predefinida.

No és estrany que el document inclogui alguna paraula correcta que no hagi estat inclosa en el formari (o ni tan sols en un diccionari), produint-se un fals positiu durant la verificació, és a dir, la detecció d'un error inexistent. Els falsos positius sovintegen amb els noms propis i amb les paraules d'àmbits especialitzats, però també amb els neologismes (paraules noves) que apareixen diàriament. Per tal de reduir-ne la freqüència, els programes de verificació ortogràfica permeten l'ampliació personalitzada del diccionaris; de manera que es puguin anar incorporant paraules al diccionari perquè en el futur no siguin considerades errònies.

De vegades, la identificació d'un error ortogràfic no es produeix perquè el resultat n'origina una altra, diferent de la pretesa, que es troba al diccionari (#*cinc* per *tinc*). En aquest cas el programa no detecta res i es produeix un fals negatiu. Si la seqüència vulnera les regles sintàctiques de la llengua (#*cinc gana* per *tinc gana*), l'error podrà ser identificat durant la verificació gramatical; en canvi, si no les vulnera (#*tinc grana* per *tinc gana*), el més probable és que passi desapercebut.

Vegeu també

La verificació gramatical es tracta al subapartat 2.3.

2.2.3. Correcció d'errors ortogràfics

Una vegada detectat l'error ortogràfic cal proporcionar una o més propostes de correcció. En aquest cas es tracta d'oferir una llista de paraules entre les quals es trobi la que probablement l'autor o autora volia escriure. Per obtenir aquestes propostes de correcció es parteix d'una hipòtesi sobre l'origen de l'error que determina la mena de transformació que du a la paraula incorrecta: una lletra de més, dues de trabucades, etc. En els errors ortogràfics que resulten en no-paraules s'assumeix el model clàssic que considera que la paraula original ha estat alterada mitjançant algun dels quatre processos de transformació descrits al subapartat 1.2.

Per obtenir la llista de paraules alternatives, només cal partir de la paraula desconeguda i, per a cada un dels caràcters, realitzar els processos inversos d'edició. La quantitat de combinacions possibles acostuma a ser molt elevada, però afortunadament la majoria d'elles poden descartar-se perquè generen formes que no apareixen al diccionari. Així, de *cassa* es generarien seqüències de lletres com ara *casa, casba, casca, casda, casea, casfa*... i totes les que no apareixen al diccionari s'eliminarien (casda, casea, casfa...).

El resultat d'aquest procés és una llista de propostes de correcció que es mostra a l'usuari o usuària finals perquè en triïn la correcta. Per exemple, les propostes de correcció per a la forma incorrecta **quarte* inclourien les paraules *quartet* (omissió de *t*), *quart* (inserció de *e*), *quarts* (substitució de *s* per *e*) i *quatre* (transposició de *t* i *r*). En alguns casos aquesta tècnica pot ampliar-se amb la incorporació de suggeriments per als errors que contenen més d'un d'aquests processos d'edició (**oetra* per *oberta*, amb omissió de *b* i transposició de *t* i *r*) i per als que comencen per una lletra incorrecta (**pberta*).

A més, la majoria de sistemes no només obtenen una llista de correccions sinó que les ordenen segons la probabilitat de ser la correcció més adequada. Per fer això s'assumeix que no tots els processos d'edició ni tots els caràcters implicats són igual de freqüents. Així, sabem que els errors originats per la confusió entre *b* i *v*, entre *s, c* i *ç*, o d'omissió de la lletra *h*, són més freqüents que la substitució d'una *a* per una *k*. Per formalitzar aquest coneixement els models incorporen models probabilístics de manera que els diferents processos d'edició tinguin diferents pesos. Aquests pesos es poden assignar a partir de la similitud fonètica dels caràcters (per modelar errors estrictament ortogràfics) o de proximitat entre les tecles (per modelar errors de mecanografiat). D'aquesta manera, a cada un dels candidats se li assigna un valor numèric corresponent a la suma dels pesos de les transformacions que han intervingut en la seva generació. Si els candidats s'ordenen segons aquest valor s'obté una llista encapçalada per les correccions més probables, és a dir, per aquelles formes obtingudes mitjançant els errors més habituals.

D'altres tècniques de correcció més sofisticades no es limiten a aquests quatre processos, sinó que tracten de descobrir les paraules correctes que més s'assemblen fonèticament o ortogràficament a l'error identificat. Per exemple, calculant el nombre de seqüències de dues o de tres lletres, ***n*-grames**, que tenen en comú dues paraules. Així, **pasetxar*, format pels trigrames [#*pa, pas, ase, set, etx, txa, xar, ar*#], en què el símbol # representa un caràcter d'inici i final de paraula, tindria tres trigrames en comú amb *passejar* ([#*pa, pas, ass, sse, sej, eja, jar, ar*#]) i només dos amb *pastera* ([#*pa, pas, ast, ste, ter, era, ra*#]). Igualment, es pot calcular la similitud fonètica entre dues paraules utilitzant un diccionari fonètic (una llista de paraules transcrites fonèticament) i un programa capaç de convertir la paraula incorrecta en la seva transcripció fonètica aproximada, i computant les seqüències de símbols fonètics que comparteixen.

Finalment, hi ha un tipus d'errors ortogràfics, normalment causats per les interferències d'altres llengües, que no es poden modelar a partir de simples processos d'edició, ja que la distància d'edició entre ells (v. distància de Levenshtein) és massa elevada i seria computacionalment massa costosa, o fins i tot impossible de calcular. Així, davant de la paraula **vivenda* un sistema de verificació ortogràfica el podrà detectar com a error amb facilitat. Però a l'hora d'obtenir la seva proposta de correcció (*habitatge*) no el podrà obtenir automàticament a partir d'una o dues edicions. En aquests casos el més habitual és mantenir una llista *ad hoc* d'errors freqüents i les correccions corresponents que pugui ser consultada directament pel mòdul de detecció i generació de propostes de correcció.

2.3. La verificació gramatical automàtica

En la majoria d'eines de verificació automàtica es parla de *verificació gramatical automàtica* per fer referència a la detecció i generació de propostes de correcció dels errors que requereixen tècniques de correcció contextuals. Aquesta distinció és equívoca, ja que no tots els errors que precisen un tractament contextual són de naturalesa gramatical. Així, per exemple, per detectar si hi ha hagut un error en l'ús de la coma o de les cometes que obren i tanquen una expressió entre cometes, caldrà tenir en compte que abans o després del signe de puntuació en qüestió hi hagi o no un espai i una paraula. Per tant, per poder detectar l'error a:

Vegeu també

Les tècniques de correcció contextuals es tracten al subapartat 2.1.

**Vam veure la "seva " casa*.

en què sobra un espai entre la *a* de *seva* i les cometes de tancament, cal tenir en compte que, un cop dividida la frase en elements textuals (també anomenats *tokens*), a l'esquerra de les cometes de tancament hi ha la seqüència:

COMETES OBERTURA + PARAULA + ESPAI

i a la dreta

ESPAI + PARAULA

Això seria per tant un exemple d'error tipogràfic (o d'estil) que caldria tractar amb regles contextuals.

Aquí definim com a *verificació gramatical automàtica* el tractament de tots aquells aspectes que tinguin a veure amb l'ús correcte de la normativa i no tinguin a veure amb l'ortografia.

Així, doncs, considerarem errors gramaticals els exemples inclosos en la fila relacionada corresponent de la taula 1, repetida aquí per comoditat.

Taula 3. Exemples d'errors tractables a la verificació gramatical automàtica

Tipus	Àmbit	Exemple
ERR	GRAM	**1)** **Aniré** directe després de deixar la nena. **2)** Va anar a recollir els **seves** germans a l'estació. **3)** Va preguntar si efectivament aquella mare amb coratge homèric era a la sala. **L'hi** van dir que efectivament hi era. **4)** Tot i que t'oferies per fer la presentació, **contaves** que nosaltres vinguéssim [...]. **5)** Hem trobat molt de **tràfic** a la N-340.

Les tècniques habitualment més emprades en la verificació d'errors gramaticals són les tècniques contextuals, tot i que ja hem dit que no exclusivament per a aquesta mena d'errors. Aquestes tècniques també s'anomenen *tècniques d'anàlisi sintàctica* (o *parsing*) i poden agrupar-se segons diversos criteris:

a) La profunditat o superficialitat de l'estructura amb què es fa l'anàlisi lingüística que sustenta el procés de verificació automàtica, i es parla de *tècniques d'anàlisi sintàctica profunda* o *superficial*.

b) L'abstracció del coneixement lingüístic mitjançant tècniques d'aprenentatge automàtic (de base estadística) o mitjançant la creació de regles per part d'experts, generalment lingüistes computacionals, segons un estudi de corpus o unes especificacions, i es parla de *tècniques de base estadística* o *de creació de gramàtiques amb regles manuals*.

Les tècniques que més es fan servir per la simplicitat computacional que presenten són les que pressuposen una anàlisi superficial, entre les quals es troben els **autòmats finits**. Aquestes tècniques permeten la detecció d'errors mitjançant reconeixement de patrons. Les tècniques basades en anàlisi sintàctica profunda són massa costoses computacionalment i no solen fer-se servir per a tasques com la verificació automàtica (tot i que hi ha alguns sistemes que les fan servir de forma parcial). El principal problema que presenten és que són

difícils d'ajustar tant perquè no pateixin un blocatge en el processament per incapacitat de tractar determinades estructures com perquè no acabin trobant correctes estructures que en realitat no ho són.

2.3.1. El reconeixement de patrons

La tècnica més utilitzada per al tractament automàtic de la verificació gramatical té un enfocament casuístic i es basa en el reconeixement de patrons. Això vol dir que el verificador recorrerà el text cercant les seqüències de paraules que segueixen un ordre o trets lingüístics preestablerts, que caracteritzin els errors.

En relació als errors de la taula 4, de primer veurem per què cal tenir en compte el context per detectar els errors dels tres primers exemples, i després veurem perquè pot ser opcional en els altres dos.

A les frases 1, 2 i 3 de la taula 3, la detecció dels errors implica considerar que:

a) A l'exemple 1 falta un complement direccional que a més es tracta d'un pronom feble que és part d'una forma verbal (*anar-hi*) lexicalitzada.

NOT *Pron HI* OR *CIRCUM_LOC* + *Verb ANAR* + NOT *Pron HI* OR *CIRCUM_LOC*

b) A l'exemple 2 cal tenir en compte que *seves* no concorda amb gènere ni amb *els* ni amb *germans* i que, si totes tres paraules han de formar un sintagma nominal, han de concordar en gènere i nombre.

Det Masc Pl + Adj Fem Pl + Nom Masc Pl

c) A l'exemple 3, per hipotetitzar un error sobre *l'hi*, cal tenir en compte que està acompanyant un forma verbal (perifràstica) de *dir*, i que aquest verb ja té un complement directe (l'oració completiva *que efectivament hi era*).

Pron EL + Pron HI + Verb DIR + QUE + Oració Subord

Nota

És cert que la frase *L'hi van dir que efectivament hi era* podria ser una frase correcta per referir-se a una frase com ara *A l'escola van dir a la Bruna que efectivament hi era*, en què *hi* estaria substituint el complement de lloc *a l'escola* però donem aquesta lectura per poc freqüent donada la major freqüència de confusió entre *li* i *l'hi*.

Aquests patrons d'error poden utilitzar informació del nivell gràfic o fer ús d'alguna informació lingüística accessible per al verificador (lema, categoria gramatical, gènere o nombre, persona, temps, mode, etc.).

Els patrons d'error poden ampliar-se amb el corresponent suggeriment de correcció. Els patrons ampliats de la taula 4, en què el suggeriment de correcció s'indica a la dreta de la fletxa, serveixen per identificar i corregir quatre tipus d'errors gramaticals prou freqüents en català.

Taula 4. Patrons de correcció ampliats amb proposta de correcció

Patrons de correcció ampliats amb proposta de correcció
a fi de que → a fi que
PENSAR en VINF → PENSAR a VINF
tant ADJ → tan ADJ
si ... PLUSQSUBJ ... PLUSQSUBJ → si ... PLUQCSUBJ ... CONDCOMP

Hi trobem regles per detectar la inserció indeguda de la preposició de la locució *a fi que*, la introducció del complement infinitiu de *pensar* amb la preposició *en* (**Pensa en venir* per *Pensa a venir*), l'ús de *tant* davant d'adjectiu en lloc de *tan* (**No és tant fort* per *No és tan fort*), i l'ús del subjuntiu plusquamperfet en lloc del condicional compost com a segona part d'una oració condicional amb temps compostos (**Si ho haguessis dit ho hagués fet* per *Si ho haguessis dit ho hauria fet*).

Utilitzant diferents tipus d'abstraccions sobre les dades (com ara el lema *PENSAR* per a qualsevol forma flexiva del verb *pensar*, o PLUSQSUBJ per a qualsevol verb en plusquamperfet de subjuntiu) i símbols (com els punts suspensius per a qualsevol seqüència de paraules), els patrons assoleixen un grau de generalització que cobreix un bon nombre de seqüències incorrectes.

D'altra banda, el tractament dels errors de les frases 4 i 5 de la taula 3 podria fer-se mitjançant tècniques no contextuals assumint que:

- Tractar-los tenint en compte el context requeria segurament l'ús de tècniques d'anàlisi semàntica o pragmàtica, les quals són encara poc robustes per a aplicacions reals.

- És fàcil confondre *comptar* amb *contar* i *trànsit* amb *tràfic*, per proximitat fonètica i per interferència amb el castellà.

- Com que es tractaria d'errors amb un grau de precisió baix, perquè no tenim en compte el context per detectar-los, podríem tractar-los com a advertiments i no pas com a candidats a error.

Cal tenir en compte que sempre podríem complicar les regles de detecció per a aquests dos errors si per exemple requeríssim que el verb que segueix *contar* estigui en subjuntiu perquè saltés la regla:

Verb CONTAR + QUE + Verb Subj → CONTAR per COMPTAR

Aquesta regla es podria sustentar en un estudi de corpus, per exemple, que mostrés que l'estructura *Verb + QUE + Verb Subj* és més freqüent amb *comptar* que no pas amb *contar*, com per exemple a *compto que vinguis*. Evidentment,

de seguida ens podrien sortir contraexemples, però aquest és el punt en què s'ha de valorar i decidir si prioritzem cobertura (detectar un major nombre d'instàncies de l'error) o precisió (detectar-ne menys però fer-ho millor).

En qualsevol cas, perquè el verificador sigui capaç de detectar un error, aquest ha de correspondre a un dels patrons prèviament especificats; és a dir, perquè la verificació sigui efectiva i factible, cal anticipar la mena d'errors que poden aparèixer, i no tots els errors gramaticals són fàcils de preveure.

2.3.2. Verificació gramatical de base estadística

Lectura recomendada

Per aprofundir en les tècniques de detecció de base estadística, podeu consultar Leacock et al. 2014.

Les tècniques de verificació d'errors gramaticals de base estadística també es basen en l'ús de patrons lingüístics, però en lloc de ser definits manualment, es defineixen a partir de coneixement estadístic derivat de corpus que, generalment, ha d'haver estat prèviament anotat amb informació lingüística. Amb aquest coneixement es creen models de llenguatge que després serveixen de referència per a la detecció d'errors o per a la generació de propostes de correcció. La verificació de textos amb tècniques de base estadística es pot fer modelant la llengua escrita correctament o modelant la llengua escrita incorrectament.

Detecció d'errors a partir de models de llengua correctament escrita

Un model de llengua ben escrita pot servir tant per a la detecció d'errors com per a la generació de propostes de correcció. Si el volem fer servir per a la detecció d'errors ens pot ser útil tant per determinar si una seqüència de paraules forma part de la llengua ben escrita, i per tant la donem com a correcta (o ben formada), com per determinar que no en forma part i per tant és, hipotèticament, incorrecta (o mal formada; tot i que també podria ser simplement poc freqüent).

Per aplicar un model de llengua ben escrita a la tasca de generació de propostes de correcció, el que fem és determinar la validesa o la freqüència de les seqüències que resultarien donada una determinada paraula que forma part de la llista de propostes de correcció. A continuació, exemplifiquem tots dos processos.

Una tècnica freqüent basada en models de llengua correctament escrita és l'extracció de seqüències d'*n*-grames de paraules que la caracteritzin. Per exemple, a la taula següent presentem els bigrames més freqüents extrets del Wikicorpus català (Reese et al. 2010) per a les paraules *niu* i *nou* seguides d'un nom.

Taula 5. Bigrames freqüents on les paraules niu i nou concorren amb noms comuns al Wikicorpus (Reese et al. 2010)

Bigrama	Freqüència	Bigrama	Freqüència
niu hoste	1	nou rei	398

Bigrama	Freqüència	Bigrama	Freqüència
niu descobert	1	nou govern	292
niu cobert	1	nou sistema	196
niu cau	1	nou disc	195
niu any	1	nou estil	154

Segons aquesta taula, si en un text ens trobem la frase

A la venda el niu disc de la banda britànica.

podem afirmar amb gran certesa que la seqüència de paraules "niu disc" és incorrecte. En aquest cas, a més, si tinguéssim un generador d'alternatives que partís de la font, podríem generar la seqüència "nou disc" com a proposta de correcció i comprovar que efectivament és molt més freqüent al mateix corpus. Fixeu-vos que en aquest cas el model (la taula) ni tan sols necessita informació gramatical ni morfològica, és per això que en diem un model de formes, o sovint un model del llenguatge.

Detecció d'errors a partir de models de llengua escrita incorrectament

La detecció d'errors a partir de llengua escrita incorrectament pressuposa l'ús d'un corpus de textos escrits per parlants (estudiants, usuaris) de la llengua que es vol corregir. En aquest corpus s'anoten els errors manualment o, de vegades, semiautomàticament i s'hi apliquen algorismes per modelar-los d'una manera semblant a com es fa amb els models de llengua ben escrita. En aquest cas, el sistema utilitza un conjunt de trets que caracteritzen l'ocurrència de l'error per classificar-lo en ús incorrecte (o correcte) amb un cert grau de probabilitat.

Imaginem que volem tractar un error com ara "la tradició ens té acostumats *a que nens i nenes vesteixin roba d'uns colors determinats", on la preposició "a" d'"acostumats a que" hauria de caure segons la normativa. Per a aquest cas generaríem un vector amb els trets que caracteritzaria l'ús, en aquest cas l'absència, de la preposició, com veiem a la taula següent.

Taula 6. Fenomen lingüístic: caiguda de la preposició "a".

Tret	Valor
Paraula en qüestió (P_0)	NULA
Paraula a l'esquerra (P_{-1})	acostumats
Paraula a la dreta (P_{+1})	que
P_{-1} P_0	acostumats a
P_0 P_{-1}	a que

Tret	Valor
Trigrama	acostumats a que
Pentagrama	té acostumats a que nens
Lema esquerra (L_{-1})	acostumar
Lema dreta (L_{+1})	que
Categoria gramatical esquerra (PoS_{-1})	Verb
Categoria gramatical dreta (PoS_{+1})	Conjunció
L_{-1} P_0	acostumar a
P_0 L_{+1}	a que
PoS_{-1} P_0	Verb a
P_0 PoS_{+1}	a Conjunció

Aquesta taula té trets que de forma aïllada podrien reflectir un ús correcte de la llengua, però justament la seva combinació és el que els fa incorrectes i és el que aprèn a diferenciar un algorisme de classificació. Com veieu, la mena d'informacions que faria servir aquest vector no són tan diferents de les que fa servir una regla feta per lingüistes, però la diferència és justament que aquest vector forma part d'un model derivat d'exemples reals de llengua escrita incorrectament.

2.4. La verificació estilística automàtica

Un altre aspecte en què la verificació automàtica pot suposar un complement a l'edició és mitjançant l'ajut a la revisió d'estil. La revisió d'estil permet controlar tant el nivell de llegibilitat d'un text, com el compliment de determinades convencions de redacció definides per una entitat pública o privada.

Entenem aquí l'estil com un conjunt de característiques objectives del document, tant quantitatives com qualitatives, associades a un determinat registre lingüístic o varietat estilística. Per fer un control d'estil s'assigna al text una classe o categoria predefinida i es mesura les discrepàncies del text en qüestió en relació a altres documents prototípics de la mateixa classe. Es comparen les característiques del document amb els trets lingüístics que el sistema té definits com a preceptius per a la categoria triada. Amb aquesta informació, es genera un informe per a l'usuari/ària.

En general la categoria es pot seleccionar a partir d'un nombre de models estilístics predefinits. En alguns casos, el verificador permet crear nous models estilístics, assignant els valors desitjats a les característiques lingüístiques. Els

trets utilitzats poden incloure mètriques com ara el nombre màxim de paraules per oració, la presència o absència d'expressions col·loquials, o el nombre màxim de sintagmes preposicionals consecutius.

2.4.1. Mètriques de llegibilitat

Entre les primeres tècniques utilitzades per a la verificació d'estil automàtica es troben les mètriques de llegibilitat, valors numèrics que permeten obtenir el grau de dificultat de lectura d'un text. Aquest grau de dificultat es mesura mitjançant un valor numèric obtingut a partir de trets lingüístics quantificables, com la longitud de les oracions, la longitud de les paraules, la quantitat de preposicions en una frase o la raresa (poca freqüència d'ús) de les paraules.

Aquestes mètriques són el resultat d'estudis basats en l'anàlisi de regularitats estadístiques que mostren els textos respecte els trets identificables en el text. Un dels objectius d'aquests estudis és elaborar fórmules de llegibilitat que serveixen per determinar empíricament paràmetres estimatius del grau de dificultat d'un text.

Històricament, la fórmula de llegibilitat més emprada és la proposada per Rudolph Flesch per a l'anglès. Aquesta fórmula determina l'índex de llegibilitat (IL) d'un text a partir de la mitjana de síl·labes per paraula (SP) i de la mitjana de paraules per oració (PO):

$$IL_{Flesch} = 206{,}835 - (84{,}6 \times SP) - (1{,}015 \times PO)$$

Aquest índex genera un valor entre 0 i 100 correlacionat amb la complexitat de lectura. Com més complex el text, més baix el valor; com més alt, més fàcil (taula 5).

Taula 5. Relació entre els índex de llegibilitat i els graus de dificultat

Llegibilitat	Dificultat	Nivell equivalent
90-100	molt fàcil	estudiant d'11 anys
80-90	fàcil	
70-80	més aviat fàcil	
60-70	estàndard	estudiant de 13 a 15 anys
50-60	més aviat difícil	
30-50	difícil	
0-30	molt difícil	graduat universitari

Existeix una fórmula derivada d'aquesta que es coneix com a Flesch-Kincaid Grade Level, que produeix un valor comparable amb la numeració utilitzada en el sistema educatiu dels Estats Units d'Amèrica, des de *1st grade* fins a *12th grade*.

L'índex Gunning-Fog utilitza tant la mitjana de paraules per oració (PO) com la proporció de paraules complexes (PC). Aquest índex defineix com a complexes les paraules de tres o més síl·labes, sense tenir en compte noms propis o compostos:

$$IL_{Gunning\text{-}Fog} = 0{,}4 \times (PO + 100 \times PC)$$

L'índex Coleman-Liau utilitza la mitjana de paraules per oració (PO) i la llargada mitjana en caràcters per paraula (CP):

$$IL_{Coleman\text{-}Liau} = 5{,}89 \times (CP - (29{,}5/PO))$$

O, l'*automated readibility index* (ARI) que també utilitza la mitjana de paraules per oració (PO) i la llargada mitjana en caràcters per paraula (CP):

$$IL_{ARI} = (4{,}71 \times CP) + (0{,}5 \times PO) - 21{,}43$$

Els resultats de tots aquests índexos acostumen a ser similars, i tot i ser prou fiables i objectius, només cal entendre'ls com uns indicadors orientatius per conèixer la complexitat formal del text analitzat. Sovint aquests índexos són considerats poc útils i fiables, ja que si bé són estrictament numèrics, també és cert que no tenen en compte fenòmens lingüístics més complexos. Si més no, no pas directament. Actualment, s'està duent a terme recerca que té en compte característiques lingüístiques més complexes, especialment mesures que tenen en compte altres nivells de descripció lingüística: morfologia, estructures sintàctiques complexes, camps semàntics i marcadors textuals. S'espera que aquesta mena de mesures proporcionin una caracterització quantitativa-qualitativa més acurada (Fitzgerald et al. 2015).

Usos de les mesures de llegibilitat

Cada vegada més s'investiga sobre l'adequació i la validitat de fer servir determinades mesures de llegibilitat (Begeny i Green 2013). En aquest sentit, convé recordar que les proves que permeten associar una dificultat determinada a un text o determinar si aquest text és adequat per a un determinat públic es basen en estudis estadístics. Per tant, es fan prenent una població determinada com a referència, i s'ha de tenir en compte que aquesta mesura sigui aplicada a una població comparable. Per exemple, la llegibilitat d'un text mesurada per a nadius s'haurà de mesurar d'una manera diferent de com es mesura la llegibilitat d'un text per a no nadius, o, de manera semblant, s'haurà de distingir entre lectures per a poblacions adultes i per a poblacions infantils o adolescents.

2.4.2. Lèxics controlats

Una altra tècnica important en la verificació d'estil automàtica és la utilització de lèxics i estructures controlades. La idea és utilitzar les mateixes tècniques utilitzades en la verificació ortogràfica i gramatical, però per validar que el vocabulari i les estructures sintàctiques emprades corresponguin a un registre estilístic.

Un exemple d'aplicació es verificar l'ús d'una determinada varietat dialectal. En aquest cas, el diccionari ortogràfic hauria d'incloure per a cada paraula informació relativa a les variants en què és considerada pròpia (per exemple, *granera* seria vàlida en català balear i valencià, i *escombra* en català oriental i nord-occidental). En el cas que el text inclogui alguna paraula, que, tot i ser present al diccionari, no sigui vàlida de la varietat dialectal seleccionada, l'ordinador la marcarà com a incorrecta i, en alguns casos, hi suggerirà una alternativa. El diccionari pot incloure, a més de varietats dialectals, els registres en què pot aparèixer cada una d'elles. Això permet a l'usuari o usuària indicar que el text que està escrivint és un document formal, col·loquial, familiar o tècnic, i ferne la verificació segons aquests criteris.

En certa manera la idea és etiquetar els recursos lèxics de manera que el verificador ortogràfic utilitzi únicament un subconjunt del diccionari, el corresponent al registre o varietat a què pertany el document. El mateix concepte pot aplicar-se al verificador gramatical: en aquest cas, els recursos són els patrons sintàctics o les regles gramaticals, que, organitzades per criteris d'estil, s'apliquen únicament als textos d'un determinat registre.

3. Disseny, implementació i avaluació d'un verificador automàtic

Per acabar d'integrar els conceptes vistos fins ara i obtenir una visió general de què és un verificador automàtic, presentarem succintament el procés de disseny i implementació d'*El Corrector*, un corrector ortotipogràfic, ortogràfic i gramatical normatiu. A continuació presentarem també una altra eina, LanguageTool, la qual és de codi obert i fàcilment adaptable fins i tot sense gaire coneixements de programació.

El Corrector

El Corrector fou desenvolupat pel Grup de Lingüística Computacional de la Universitat Pompeu Fabra, amb la col·laboració de Barcelona Media Centre d'Innovació; en gran part és el resultat del finançament rebut a partir d'un concurs públic convocat per la Generalitat de Catalunya (SE/CTTI/51/05). En l'actualitat aquest software no és mantingut, però la seva concepció és igualment útil per a l'explicació. Els detalls tècnics de la implementació es poden trobar a les actes de Language Resources and Evaluation Conference, congrés celebrat el 2008 a Marràqueix: Quixal et al. 2008.

A les tres primeres seccions presentem i expliquem les tres fases essencials de què consta el procés de creació d'un verificador automàtic: en primer lloc, la definició d'especificacions; en segon lloc, el disseny tècnic i el desenvolupament del motor de processament; i finalment un procediment d'avaluació. A la quarta, explicarem les nocions essencials necessàries per instal·lar i modificar el comportament d'un motor de correcció de textos que incorpora regles manuals per a la detecció d'errors amb tècniques no contextuals i contextuals.

3.1. Especificacions dels criteris de verificació

Les especificacions d'un programa informàtic són essencialment una descripció del que s'espera que faci. Per tant, les especificacions d'implementació d'un verificador automàtic seran essencialment de caràcter lingüístic, però també tindran en compte la mena d'interacció amb l'usuari o usuària finals, i l'entorn de treball en què es farà servir el verificador. Aquestes darreres tenen a veure amb aspectes com ara el sistema operatiu en el qual haurà de funcionar, l'eina d'ofimàtica en la qual s'integrarà, o saber si ha de funcionar o no en una instal·lació en xarxa. D'aquests aspectes més generals i aplicables al desenvolupament de qualsevol eina de verificació, no en parlarem aquí.

A l'hora de definir les especificacions d'*El Corrector* es va partir de la descripció que en va fer la Generalitat de Catalunya en el document que feia públic el concurs de licitació. Com a criteris generals el concurs establia una sèrie de requeriments tècnics i de llicència, com ara que fos de codi lliure i obert, modular (separant funcionalitats i interfícies gràfiques), multiplataforma (Linux, Mac i Windows), etc., i una sèrie de requeriments sobre la funcionalitat lingüística: que fos d'ús general, que tingués en compte les formes adequades a la llengua l'estàndard de les varietats dialectals del català (finalment es van considerar les variants balear, central, nord-occidental i valenciana), que fos normatiu, que corregís errors ortotipogràfics, ortogràfics i gramaticals, que permetés un mínim d'interacció amb l'usuari o usuària final (activació i desactivació de la correcció gramatical, bloqueig de l'activació de determinats avisos, correcció interactiva, etc.).

Pel que fa a l'especificació dels errors ortogràfics i gramaticals que s'exigia que es tractessin, el document del concurs de licitació proporcionava una llista d'errors el tractament dels quals havia d'estar garantit i una llista d'errors el tractament dels quals era d'interès. En la primera llista, s'hi incloïen errors com ara la correcció ortogràfica del lèxic comú, topònims, antropònims, etc., apostrofació d'articles, preposicions i pronoms febles (**el home*, **de ahir*, **m'el pren*), contracció de l'article masculí, guionets en els numerals, morfologia de noms, adjectius i verbs (**altruiste*, **reflexava*, **recullir*, **coneguent*, etc.), ortografia dels pronoms febles en general, perífrasi d'obligació (**tenir que*, **haver-hi que* + infinitiu), omissió de la conjunció *que* (**Us agrairé estudieu l'informe*). La segona llista incloïa errors com ara l'ús de diacrítics (*és*/*es, dones*/*dónes*), l'ús de l'article neutre (**amb lo útil que és*), canvi i caiguda de preposicions (**Les hem acostumades a que mengin soles*), ús pronominal inadequat (**S'ha caigut del cavall*), etc.

El que ens interessa destacar aquí és que la realització d'aquestes llistes es va fer a partir de l'experiència que tenien les persones que van redactar les bases del concurs. Es tracta dels dubtes i errors més freqüents atribuïts a les persones que es dirigeixen al Servei de Consultes Lingüístiques de la Generalitat de Catalunya. Una altra manera de fer aquestes llistes, però, hagués estat seguint un procés de compilació, anotació i anàlisi d'un volum de textos considerable redactats per parlants identificables amb l'usuari o usuària final d'aquesta eina. Com us podeu imaginar, aquest procés és molt costós i lent i, de fet, és un tema pendent de resoldre en àrees de treball i recerca com ara l'estudi dels trets formals de la llengua en les diverses etapes d'adquisició del llenguatge, tant en aprenents nadius de la llengua com d'aprenents no nadius.

Un dels rols fonamentals de les especificacions és la creació d'una bateria de proves que serveixi per garantir la qualitat i l'adequació funcionals del programari de verificació a cada nova versió, i també per validar l'eina durant el procés de desenvolupament. Així, per exemple, podem tenir conjunts de tests que ens permetin determinar els percentatges de cobertura i precisió del programari donades una sèrie de frases que sabem que exemplifiquen els errors que volem tractar.

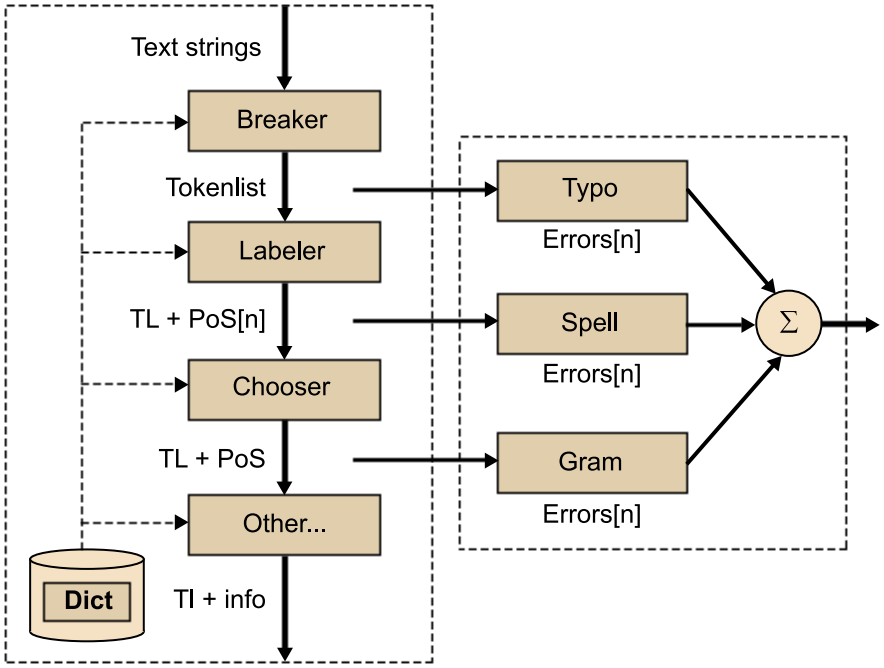
Cal tenir en compte que idealment aquestes llistes no han de tenir només els errors que el programari tracta correctament, sinó tots aquells que pensem que *segons les especificacions* hauria de ser capaç de tractar o, com veurem més endavant, ignorar. Tampoc no haurien de contenir exemples d'errors que hem decidit que no haurien de formar part del conjunt de fenòmens que hauria de tractar el programari. Per evident que sembli aquesta darrera afirmació, convé tenir-la present, perquè de vegades s'espera que un programari presenti unes funcionalitats per a les quals no ha estat dissenyat.

3.2. Disseny i desenvolupament de l'arquitectura

A l'hora de desenvolupar qualsevol sistema complex és imprescindible dividir-lo en mòduls independents que siguin responsables de tasques especialitzades i alhora que tots ells junts realitzin una tasca de forma conjunta que només s'aconsegueix fent-los treballar de manera interrelacionada.

L'arquitectura bàsica del verificador automàtic *El Corrector* té dos tipus de mòduls: els de processament i de detecció. Els primers analitzen el text per segmentar-lo i anotar-lo lingüísticament mitjançant tècniques estàndard de **processament de llenguatge natural** (PLN). Els segons revisen la seqüència d'elements lingüístics analitzats per detectar errors i generar propostes de correcció i missatges d'error.

Figura 1. Arquitectura del motor de correcció i generació de propostes de correcció d'*El Corrector*



A la figura 1 es mostra l'arquitectura utilitzada pel motor de correcció d'*El Corrector*. A l'esquerra es troben els mòduls de processament lingüístic, concretament:

- *Breaker*: aquest mòdul segmenta la seqüència de caràcters en diferents unitats: els paràgrafs, les oracions i els ***tokens*** (o elements textuals bàsics de processament lingüístic).

Tokens

Es tracta essencialment de paraules, però també es tenen en compte signes de puntuació, espais, xifres, etc.

- *Labeler*: aquest mòdul consulta el diccionari i assigna a cada paraula la categoria morfosintàctica associada a la seva forma (o més d'una categoria si hi ha ambigüitat); també inclou informació del registre, si escau, i de la variant a què pertany. Quan una paraula no és al diccionari la marca com a desconeguda.

- *Chooser*: aquest mòdul determina, en cas d'ambigüitat, quina és la lectura, o categoria morfosintàctica més probable per a una paraula donades les paraules amb què concorre. Per exemple, donada la forma *casa* hauria de decidir si es tracta d'un nom femení singular o d'un verb en tercera persona singular del mode indicatiu i temps present.

- *Other*: l'arquitectura permet afegir mòduls opcionals que analitzin estructures de nivells lingüístics superiors, com ara locucions, estructures sintagmàtiques o entitats.

A mesura que es realitza el processament lingüístic els mòduls de la part dreta de la figura 1 van verificant la correcció de la seqüència de text enriquit amb informació lingüística. Concretament els tres mòduls són:

- *Typo*: aquest mòdul utilitza tècniques contextuals basades en regles sobre els aspectes gràfics (formals) dels *tokens* per determinar la utilització correcta dels espais, les majúscules i els signes de puntuació, i detecta errors ortotipogràfics.

- *Spell*: aquest mòdul utilitza tècniques no contextuals per determinar si les paraules apareixen al diccionari, i si formen part de la variant i registre corresponent, majoritàriament detecta errors ortogràfics.

- *Gram*: aquest mòdul utilitza tècniques contextuals basades en regles amb informació morfosintàctica per detectar seqüències corresponents a errors típics, i principalment detecta errors gramaticals.

Cada un dels mòduls verificadors genera una llista amb els errors trobats. Cada un dels errors inclou informació sobre la posició de l'element incorrecte, el tipus i codi d'error (que va sovint associat a un missatge d'error o advertiment), i les possibles correccions, sempre que hagi estat capaç de generar-ne. Aquestes llistes parcials proporcionades per cada un dels mòduls verificadors s'integren en una llista única ordenada que és transferida a la interfície gràfica.

El resultat és filtrat segons les preferències de l'usuari o usuària final (variant, registre i tipus de correcció) i és visualitzat segons un codi de colors per facilitar-ne la comprensió. A partir d'aquest punt, l'usuari o usuària pot revisar cada un dels errors i advertiments, confirmar-ne la validesa i, en cas necessari, triar una de les correccions proposades.

3.3. Avaluació del sistema de verificació

Una vegada desenvolupat el sistema cal avaluar-ne el funcionament. En aquesta mena d'aplicacions el control de qualitat és fonamental, ja que una quantitat proporcionalment alta de falsos positius o de falsos negatius faria l'eina inservible. En el cas d'*El Corrector*, aquesta avaluació es va realitzar mitjançant

el control de dos aspectes diferenciats: en primer lloc, l'acompliment de les especificacions funcionals i, en segon lloc, l'avaluació sobre una mostra de textos reals.

1) Validació automàtica de les especificacions funcionals

Durant el desenvolupament de l'aplicació és important disposar d'un sistema d'avaluació automàtica que permeti garantir la bona marxa del projecte. Durant aquest període els diferents mòduls evolucionen i es milloren, els recursos lingüístics (diccionaris i regles) es modifiquen i s'amplien, i hi ha un risc real que els canvis interfereixin entre ells o amb altres mòduls i el funcionament general canviï. Pot millorar, que és allò que un esperaria, però també empitjorar, i llavors requeriria reconsiderar els canvis fets o fer-ne de nous. Per tant, cal un mecanisme de control.

Per això, tot just definides les especificacions lingüístiques és aconsellable crear un corpus d'errors, una base de dades amb centenars d'exemples de textos incorrectes amb l'error perfectament localitzat, classificat i amb la seva correcció indicada. Aquesta base de dades permet validar en tot moment la qualitat del verificador, i obtenir mesures precises i objectives del seu funcionament.

La idea és que un procés automàtic executi el verificador per a cada un dels casos (exemples) presents al corpus i analitzi si hi ha o no un error, el localitzi, el classifiqui i en generi la llista de propostes de correcció. Cada una d'aquestes informacions es compara amb les existents a la base de dades (o al corpus) i, per tant, es pot saber exactament si l'error ha estat identificat i si la correcció proposada és correcta.

En finalitzar la validació automàtica s'obtenen un conjunt d'estadístiques globals, com els percentatges de precisió i cobertura del verificador o la quantitat de falsos positius i de falsos negatius. A més, es poden generar informes detallats sobre la precisió i la cobertura per a cada tipus i codi d'error.

En darrer terme, aquesta validació automàtica permet garantir, o com a mínim avaluar, la cobertura dels errors definits i, per tant, l'acompliment de les especificacions funcionals.

2) Validació manual sobre textos reals

Més enllà de l'acompliment de les especificacions, també és important i interessant conèixer la qualitat "real" del programari quan sigui utilitzat per usuaris o usuàries finals, és a dir, quan s'apliqui a textos diversos de diferent registre i amb una distribució no uniforme dels errors. En el cas d'*El Corrector*, se'n va avaluar el comportament davant de diferents textos corresponents a publicacions científiques, notícies de premsa, textos de blocs i correus electrònics (vg. Quixal et al., 2008).

A partir dels errors detectats pel verificador es van mesurar la precisió i la cobertura, així com l'adequació de la proposta de correcció. La cobertura es va mesurar com la proporció d'errors detectats entre els errors reals continguts en els textos, i va resultar en un valor del 83%. La precisió es va mesurar com la proporció d'errors que efectivament ho eren entre tots els errors assenyalats pel programari, amb un resultat del 70%.

A més, aquesta avaluació va permetre fer una anàlisi dels punts flacs d'*El Corrector*: això va permetre saber que la majoria de falsos positius (57%) van produir-se per l'absència de paraules al diccionari, un fet previsible atès que es tracta d'un verificador normatiu i, per tant, no admet moltes paraules que sí s'admeten en àmbits específics. També que la majoria de falsos negatius (52%) es tractava d'errors que haurien necessitat una anàlisi lingüística més avançada (sintàctica o semàntica) per poder ser detectats i tractats correctament.

3.4. Un verificador de codi obert adaptable

En aquest subapartat introduïm un verificador de codi obert que és disponible a data de redacció d'aquest material: LanguageTool. LanguageTool és desenvolupat per una comunitat de programadors coordinada per Daniel Nabert, qui va concebre l'aplicació en el marc d'un treball de fi de carrera. A principis de 2016 LanguageTool compta amb una petita comunitat de col·laboradors que manté el software i els recursos que proporcionen la verificació automàtica ortogràfica, gramatical i d'estil per a més de 20 llengües amb diferents graus de completitud i funcionalitat per a cada una d'elles. La comunitat de LanguageTool ha desenvolupat connectors per a diverses aplicacions ofimàtiques de codi obert, entre elles OpenOffice, Firefox i Chrome (parcialment obert), i també per a aplicacions comercials de codi propietari com ara Microsoft Word.

LanguageTool permet qualsevol tipus de modificació sobre el seu codi, tret de les parts del codi que, provenint de terceres parts, presenten una llicència amb restriccions. Essencialment, s'hi poden fer tres tipus de contribucions: integració en eines de processament de textos; millora o ampliació dels algorismes d'anàlisi lingüística; i millora o ampliació dels recursos lingüístics. En aquest capítol introduirem el procés d'ampliació del diccionari, el procés d'adaptació i creació de regles per a la verificació automàtica, i la detecció d'errors amb tècniques estadístiques.

3.4.1. Ampliar els recursos lèxics

LanguageTool fa servir tres tipus de recursos lèxics: un diccionari per a l'anàlisi lingüística (segons la seva terminologia en anglès tagger o *part-of-speech dictionary* o *tagger dictionary*), un diccionari per a la generació de paraules (en la terminologia interna *diccionari sintetitzador*) i un diccionari per a la verificació automàtica de l'ortografia. En realitat, només cal mantenir un sol diccionari i amb les eines que proporciona el maquinari es poden generar la resta.

El diccionari arrel

El diccionari arrel és una llista d'entrades lèxiques (paraules). Cada línia és una entrada lèxica i cada entrada lèxica es compon de forma, lema i etiqueta morfosintàctica. En algunes de les llengües disponibles per a LanguageTool la informació pot incloure informació relativa a la freqüència d'ús d'una paraula. Per exemple, a la taula 7 podem veure la informació d'algunes de les formes del verb tenir (*tin*, *tinc*, *tindran* i *tindre*). Fixeu-vos com la forma i l'etiqueta morfosintàctica són diferents per a totes elles i el lema és sempre el mateix.

Taula 7. Entrades del diccionari arrel de LanguageTool per al català

Forma	Lema	Etiqueta morfosintàtica
tin	tenir	VMM02S0V_A
tinc	tenir	VMIP1S00_Q
tindran	tenir	VMIF3P00_L
tindre	tenir	VMN00000_M

Totes les etiquetes morfosintàctiques de la taula 7 comencen per la lletra V (verb), segueixen amb una M (*main,* en anglès, és a dir, és tracta d'un verb que funciona com a verb principal en una oració o grup verbal, no pas com a auxiliar) i a partir d'aquí cada una té valors diferents per a diversos trets lingüístics. A la tercera posició el mode: en aquest cas veiem iMperatiu a *tin*, Indicatiu a *tinc* i *tindran*, i iNfinitiu a *tindre*. Després vénen altres informacions: temps verbal, persona, nombre, gènere i varietat dialectal (fixeu-vos en la V, valencià, de just abans del guionet baix de l'etiqueta de *tin*). Com podeu veure, en alguns casos algunes posicions presenten un zero, cosa que vol dir que per a aquella paraula aquella informació no és pertinent; per exemple, la forma *tindre* no té ni temps verbal, ni persona, ni nombre, ni gènere, ni varietat dialectal. La lletra que segueix el guionet de la penúltima posició és un identificador de grup de freqüència que va de la A (menys freqüent) a la Z (més freqüent).

Enllaços interès

Per a més informació sobre aquest etiquetari i sobre EAGLES/PAROLE podeu consultar les següents pàgines web: http://nlp.lsi.upc.edu/freeling/doc/tagsets/tagset-es.html i http://www.ilc.cnr.it/EAGLES96/home.html.

Convé destacar que l'origen del diccionari de LanguageTool és una altra eina de codi obert, Freeling, iniciada i coordinada per Lluís Padró, investigador de la Universitat Politècnica de Catalunya. Aquesta eina fa servir un etiquetari (llista d'etiquetes morfosintàctiques) basat en un sistema anomenat PAROLE, el qual fou proposat per un grup d'investigadors europeus que col·laboraren durant la dècada dels noranta per crear EAGLES, unes recomanacions per a la creació d'etiquetaris morfosintàctics aplicables a diferents llengües.

Extensió de la cobertura lèxica de LanguageTool

Versió 3.2 de LanguageTool

Totes les informacions relacionades a fitxers i estructura de directoris i funcions del programari estan comprovats amb la versió 3.2 de LanguageTool.

Essencialment hi ha dues maneres d'ampliar la cobertura lèxica de LanguageTool, és a dir, d'augmentar el nombre de paraules que és capaç d'analitzar o de fer servir per generar propostes de correcció. El més senzill és la incorporació manual d'entrades lèxiques en fitxers específicament pensats per a això. Actualment aquests fitxers s'anomenen *added.txt* i *spelling.txt* i es troben en directoris específics per a cada llengua. Assumint que us trobeu al directori arrel de Language tool, la ruta dels directoris d'aquests fitxers en la versió catalana de l'eina seria:

- org/languagetool/resource/ca/added.txt
- org/languagetool/resource/ca/spelling.txt

Per a altres llengües, heu de substituir el directori "ca" pel codi corresponent de la llengua que voleu treballar. Per exemple, "en" per a l'anglès, "es" per al castellà, o "ru" per al rús. Els principals inconvenients d'aquesta manera d'ampliar el diccionari són que, d'una banda, cal entrar totes les paraules dues vegades, i, de l'altra, a mesura que el nombre de paraules incorporat augmenta, pot afectar considerablement el temps de resposta del programari.

L'altra manera de fer extensions permet fer-ne d'un gran nombre de paraules i incorporar-les al diccionari codificat en binari. Això darrer fa que el rendiment del programari no es vegi (tan) afectat per la incorporació de noves paraules. Així mateix, les funcions de generació de diccionaris a partir d'un diccionari arrel nou que proporciona LanguageTool evita haver d'entrar dues vegades cada paraula, i alhora això redueix la possibilitat d'introduir errors deguts a la falta d'atenció o cansament. Els passos per fer això per al català serien:

1) Codificació manual d'un diccionari en format text (txt) del diccionari arrel.

2) Binarització del diccionari d'anàlisi morfològica i del de síntesi, així com del diccionari de correcció ortogràfica.

El diccionari arrel es pot codificar des de zero amb el format descrit un parell de paràgrafs més amunt – i tenint en compte que l'etiquetari per a cada llengua és diferent. LanguageTool disposa de funcions d'exportació i d'importació tant per exportar el diccionari proporcionat inicialment com per binaritzar-ne les noves versions. La informació detallada es pot trobar a la web de la comunitat desenvolupadora.

3.4.2. Regles per a la verificació automàtica

LanguageTool fa servir un format propi per a la creació de regles de verificació automàtica sensitives al context. Les regles de LanguageTool tenen com a mínim la informació següent:

1) Context: es pot definir de manera positiva i de manera negativa. S'anomena *pattern* (patró en anglès) si és positiu, o *antipattern*, si és negatiu. Un exemple de context positiu seria: Determinant Masculí Singular + Nom Femení Singular. Un context definit així trobaria seqüències com ara "*el síndrome" o "*el metgessa". El context pot estar basat en qualsevol de les informacions contingudes al diccionari o generades pel procés d'anàlisi automàtica – incloent-hi coincidències parcials si aquestes es defineixen.

2) Missatge d'error: informa l'usuari o usuària finals de la norma lingüística que hipotèticament s'ha infringit; seguint l'exemple anterior, el missatge diria "Error de concordança." o "Possible error de concordança."

3) Proposta de correcció: sempre que sigui possible, i, idealment, basada en el text escrit es genera una paraula o estructura per millorar el text. En el exemple anterior, es generarien les propostes "la síndrome", per al primer cas, i "el metge" i "la metgessa" per al segon cas.

4) Exemple: n'hi ha un de positiu i un de negatiu, que serviran per comprovar que la regla s'aplica correctament.

LanguageTool fa servir un llenguatge formal propi per expressar aquesta informació. Aquest llenguatge formal està basat en XML (eXtensible Markup Language), un llenguatge estàndard de codificació d'informació utilitzat per moltes aplicacions informàtiques. La sintaxi de les regles s'expressa de la manera següent:

```
<rule>
   <pattern>
      <token>la</token>
      <token>seu</token>
      <token>amiga</token>
   </pattern>
   <message>Possible error de concordança.</message>
   <suggestion>la seua amiga</suggestion>
   <suggestion>el seu amic</suggestion>
<example correction="el seu amic|la seua amiga">Vam trobar-nos <marker>la seu amiga</marker> al
tren.</example>
</rule>
```

El parell d'etiquetes XML principal és <rule>...</rule>, que indica l'inici i el final de cada regla. Dins de cada regla trobem quatre parells d'etiquetes més, les etiquetes inicials dels quals són <pattern>, <message>, <suggestion>, i <example>. Dins l'etiqueta <pattern> s'hi defineix el context lingüístic que ha de servir per fer saltar la regla. El context lingüístic es pot basar en qualsevol de les informacions que el programari fa disponibles: paraula, lema i diversos trets morfosintàctics; i també en un nombre específic de tokens, en aquest cas tres. L'etiqueta <message> conté el missatge d'error que l'usuari o usuària veurà

quan se li detecti un possible error. L'etiqueta <suggestion> conté propostes de correcció. N'hi pot haver una o més. Es mostarien totes com a alternatives. Finalment, l'etiqueta <example> conté frases sobre les quals s'avalua la gramàtica de regles contextuals cada vegada que es vol fer una avaluació global del sistema. De frases d'avaluació n'hi pot haver diverses, i n'hi pot haver tant de correctes (sobre les quals no s'haurien d'aplicar les regles) com d'incorrectes (sobre les quals sí que s'haurien d'aplicar les regles).

L'exemple que hem posat per exemplificar la sintaxi XML de les regles és molt senzill. El context (<pattern>) conté tres tokens però tots tres són paraules literals. Podríem fer servir seqüències d'etiquetes gramaticals detectar seqüències de determinant i nom que no concordin en gènere o nombre. També podríem fer servir una etiqueta (<marker>) que serveix per definir les paraules del text que s'han de subratllar en pantalla per a l'usuari o usuària. A <suggestion> es pot definir la generació de propostes de correcció tenint en compte la informació de les paraules analitzades i coincidents amb el context. Finalment a l'etiqueta <example> es poden incloure informacions com ara les possibles correccions i les seqüències d'error que s'hagin de detectar.

Les regles de LanguageTool inclouen funcions i opcions força més sofisticades de manera que els contextos de detecció poden incloure seqüències de paraules definides per expressions regulars. També permeten operacions complexes de comparació i comprovació de trets lingüístics com ara la unificació, que permet requerir que dues paraules concordin en gènere o nombre sense haver d'especificar en cada regla quin gènere o nombre es vol tractar. També permeten la definició de seqüències de paraules que són excepcions a una regla mitjançant <antipattern>.

Enllaç d'interès

Per a una explicació detallada de la sintaxi de les regles de LanguageTool consulteu la pàgina dels desenvolupadors.

3.4.3. Detecció d'errors amb tècniques de base estadística

LanguageTool incorpora una funcionalitat per a la detecció d'errors amb tècniques de base estadística. En concret s'aplica a la detecció d'errors causats pels anomenats parells (o conjunts) confusibles. Per exemple, en castellà, *ahí* i *hay* són un parell confusible, dues paraules quasi homòfones y, en aquest cas, amb una ortografia molt semblant. Per al castellà (el català no en té), el fitxer de configuració dels parells confusibles de LanguageTool és a:

- org/languagetool/resource/es/confusion.set

El format del fitxer és aquest:

```
a; ha; 1000000 # p=1.000, r=0.889, 907+1000, 3grams, 2016-03-30
```

```
be; ve; 1000000 # p=1.000, r=0.734, 91+673, 3grams, 2016-03-30
```

En cada línia s'observa, el parell confusible (*a/ha*, *be/ve*), seguit d'un factor numèric que indica quantes vegades més probable ha de ser l'altra paraula (qualsevol de les dues) per tal que la paraula en qüestió sigui considerada error. A continuació s'indica per a cada parell la precisió (p) y la cobertura (r, de l'anglès *recall*) que obté aquesta regla en un determinat corpus i model de llenguatge. Opcionalment s'hi poden afegir propostes de correcció o explicacions. Per exemple:

```
a|Preposición; ha|Verbo haber; 1000000 (…)
```

Activitats

1. Prepareu un document amb força errors ortogràfics i apliqueu-hi el verificador ortogràfic del vostre processador de textos, i tracteu d'esbrinar les tècniques de correcció que fa servir per a elaborar la llista de suggeriments. Si voleu fer proves amb algun verificador que inclogui tècniques de detecció d'errors de base estadística podeu provar-ho amb versions modernes de MS Word, fent servir l'anglès o el castellà (errors subratllats en vermell, verd i blau).

2. Què opineu que hauria de fer el verificador gramatical del vostre processador de textos amb les seqüències agramaticals però freqüents en català col·loquial com ara *Jo em sembla que no tens raó*? I amb les seqüències amb desviacions semàntiques com ara *La finestra llegia un conte*?

3. Comproveu si el verificador estilístic del vostre processador de textos inclou el càlcul d'algun índex de llegibilitat i proveu d'aplicar-lo a un document. Tracteu d'esbrinar-ne la fórmula utilitzada.

4. Instal·leu-vos una versió recent de LanguageTool. Mireu d'accedir als diccionaris i a les gramàtiques. Proveu de fer una regla que tracti un tipus d'error que sou conscients que feu sovint. Si us en sortiu o sou atrevits, proveu-ho també amb una llengua estrangera. Repetiu l'exercici 1 ara amb LanguageTool.

Exercicis d'autoavaluació

1. Determineu a quina categoria d'errors ortogràfics pertanyen els exemples següents i raoneu la vostra resposta.
a) **Creu el tap* per *treu el cap*.
b) **Dificilment* per *difícilment*.
c) **Apit* per *api*.
d) **Satifactori* per *satisfactori*.

2. Seguint les pautes de definició de regles per a la detecció d'errors mitjançant patrons, proposeu els patrons ampliats d'errors per identificar i corregir errors gramaticals com els següents.
a) **Vinc de el camp* per *vinc del camp*.
b) **Hi han persones* per *hi ha persones*.
c) **Hi ha que venir d'hora* per *cal venir d'hora*.
d) **El fet de que vingui* per *el fet que vingui*.

3. Per què són difícils de preveure els errors gramaticals comesos en una llengua per les persones que la tenen com a llengua estrangera? Com afecta aquesta dificultat la verificació gramatical automàtica?

4. És raonable aplicar la fórmula de Flesch a textos escrits en llengua catalana? Justifiqueu la vostra resposta.

Solucionari

Exercicis d'autoavaluació

1.a) Podria tractar-se d'un error d'actuació relacionat amb els errors de la parla, per l'intercanvi del tret fonològic de localització entre les oclusives /t/ i /k/.

b) Error de competència per interferència amb les regles ortogràfiques del castellà. Formalment, és un error per substitució.

c) Error de competència per discrepància entre la parla i la normativa. Quant a la forma, es tracta d'un error per inserció.

d) Podria ser un error d'actuació resultat d'una distracció visual, per l'omissió de la segona lletra *s* en proximitat de la primera.

2.a) *de el* > *del*.

b) *hi han*... NPL > *hi ha*... NPL.

Aquest patró (en què NPL representa un nom en plural i els punts suspensius qualsevol seqüència de paraules) preveu la verificació d'errors com ***hi han tres senyors***, amb paraules entre el verb presentador i el nom en plural.

c) *hi ha que* VINF > *cal* VINF.

d) *el fet de que* > *el fet que*.

3. Els errors gramaticals comesos en una llengua per les persones que la tenen com a llengua estrangera són difícils de preveure per la impossibilitat de tenir en compte totes les llengües del món i les seves possibles interferències. Per això, els verificadors gramaticals estan dissenyats per a la verificació dels errors gramaticals comesos per persones nadiues de la llengua, encara que també s'han elaborat verificadors gramaticals específicament adaptats a persones no natives. Per exemple, per verificar l'anglès escrit per estudiants alemanys de l'anglès com a segona llengua.

4. La fórmula de Flesch aplicada al català produeix uns resultats inadequats, perquè els paràmetres emprats per la fórmula (la longitud mitjana en síl·labes de les paraules i la mitjana de paraules per oració) s'han calibrat d'acord amb les pautes de la llengua anglesa.

Glossari

array* associatiu** *m* sin. ***hash

autòmat finit *m* Un autòmat finit (o màquina d'estats finits) és un model matemàtic d'un sistema compost per estats, transicions i accions.

cerca binària *f* Algorisme informàtic que permet accelerar la cerca d'un element en una llista ordenada, en aquest cas, alfabèticament. Es basa a comparar l'element buscat amb el situat a la meitat de la llista; a partir del resultat es descarta una de les meitats de la llista. El procés es repeteix fins a localitzar l'element trobat.

cobertura *f* En sistemes de verificació, mesura de la quantitat d'errors que el sistema és capaç de detectar respecte al total que hi ha. En altres paraules, quants errors ha trobat del total que hi havia.

corrector gramatical *m* Eina informàtica, normalment integrada en un editor de textos, que s'utilitza per detectar errades gramaticals en un text escrit.

corrector ortogràfic *m* Eina informàtica, normalment integrada en un editor de textos, que s'utilitza per detectar errades ortogràfiques en un text escrit.

etiquetador *m* En l'àmbit de la lingüística de corpus, és un programa informàtic que permet l'assignació automàtica d'una *etiqueta* (*tag*, en anglès) morfosintàctica amb de la seva categoria gramatical a cada paraula d'un text.
sin. ***tagger***

fals positiu *m* En sistemes de correcció, un element correcte marcat pel corrector com a incorrecte, és a dir, un *fals error*.

fals negatiu *m* En sistemes de correcció, un element incorrecte no marcat pel corrector, és a dir, una detecció fallida.

forma *f* Cada una de les formes flexionades que poden obtenir-se a partir d'un paradigma flexiu. Per exemple: *roig*, *roja*, *rojos* i *roges*, són les formes corresponents a la flexió del lema *roig*.

formari *m* Diccionari de formes, llista o base de dades que recull el conjunt de paraules flexionades d'una llengua. Per exemple, les formes masculines i femenines, les formes singulars i plurals, o tots els elements presents en les conjugacions verbals.

hash *m* En informàtica, estructura de dades que permet indexar de manera molt eficient una informació. Es basa a indexar una dada a partir d'un valor numèric obtingut a partir de la mateixa informació, garantint que no hi pugui haver dues dades diferents que comparteixin el mateix índex.
sin. ***array* associatiu** o **llista associativa**

lema *m* Forma arbitrària seleccionada per referir-se a una conjunt de flexions. Habitualment es pren com a lema la forma singular masculina dels noms i dels adjectius o l'infinitiu dels verbs. Per exemple: *roig*, és el lema de les formes *roig*, *roja*, *rojos* i *roges*; i *jugar* és el lema de les formes *jugo, jugues, jugava, jugant, jugaríem...*

lemari *m* Diccionari de lemes, llista o base de dades que recull el conjunt de lemes d'una llengua.

lexicó *m* Llista de paraules ordenades amb informació lingüística associada (lema, categoria gramatical, gènere, nombre, etc.).

llista associativa *f* sin. ***hash***

mètrica de llegibilitat *f* En el camp de l'estilometria, equació que permet estimar el grau de dificultat de lectura d'un text en funció de les seves característiques superficials.

model estilístic *m* En sistemes de verificació, idealització d'una variant lingüística geogràfica, social o contextual, utilitzada per agrupar un conjunt de regles, elements lèxics i valors de llegibilitat, per prendre'l com a referència a l'hora de verificar l'estil d'un document.

N-grama *m* Seqüència d'ena (n) elements que ocorren de forma consecutiva en un objecte lingüístic. Els elements poden pertànyer a diversos nivells de descripció lingüística. Així, podem parlar de trigrames de lletres (els de la paraula lletres serien: *lle, let, etr, tre, res*), de

trigrames de paraules (per exemple, entre els deu trigrames de paraules més freqüents del WikiCorpus del català creat per Reese et al. 2009 hi ha *de la seva, a la ciutat* i *el nom de*).

precisió *f* En sistemes de verificació, mesura de la quantitat d'errors autèntics de tots els que el sistema ha indicat. En altres paraules, quants dels errors trobats eren realment errors.

reconeixement de patrons *m* Tècnica informàtica utilitzada en el camp del processament del llenguatge natural que serveix per detectar les seqüències de paraules d'un text que responen a unes seqüències i característiques predeterminades.

registre *m* Varietat funcional d'una llengua definida d'acord amb els factors de la situació comunicativa, com ara el nivell de formalitat, el tema del qual es parla, etc.

tagger *m* Vegeu **etiquetador**.

text no restringit *m* Expressió per referir-se a textos de qualsevol tipus (origen, format, registre, variant dialectal, etc.). Es fa servir en oposició a l'expressió *textos de domini* o *controlats*, que vol dir que són textos amb un ús uniforme de la llengua.

token *m* En processament de llenguatge natural, unitat mínima d'anàlisi. Tot i que s'associa a una paraula, també es consideren *tokens* els signes de puntuació i els elements compostos com nombres decimals, dates, telèfons...

TRIE *m* En informàtica, estructura de dades que permet indexar de manera molt eficient dades seqüencials. S'utilitza habitualment per emmagatzemar paraules mitjançant la creació d'un arbre jeràrquic, de manera que les paraules que comparteixen prefixos, s'emmagatzemen en branques properes.

variant *f* Expressió lingüística diferent d'una altra per la forma.

varietat *f* Ús específic que es fa d'una llengua d'acord amb la procedència geogràfica, històrica o social dels parlants o amb la funció comunicativa, i que es caracteritza per una determinada concurrència de variants lingüístiques.

verificació lingüística *f* En el camp de l'escriptura assistida per ordinador, terme general que inclou la verificació ortogràfica, la verificació gramatical i la verificació estilística d'un text. Els programes informàtics que la duen a terme s'anomenen, respectivament, *verificadors ortogràfics*, *verificadors gramaticals* i *verificadors estilístics*. El terme *verificador* és més genèric que el de *corrector* i no implica necessàriament la correcció dels problemes lingüístics detectats.

Bibliografia

Begeny, J. C. & D. J. Greene. (2013). Can Readability Formulas be successfully used to gauge the difficulty of reading materials?*Psychology in the Schools* 51(2), 1520–6807.

Carreras, Xavier, Isaac Chao, Lluís Padró i Muntsa Padró. (2014). "FreeLing: An Open-Source Suite of Language Analyzers"*A Proceedings of the 4th International Conference on Language Resources and Evaluation* (LREC'04), 2004.

Costa Carreras, J.; Nogué Serrano, N. (coord.) (2006). *Curs de correcció de textos orals i escrits: pràctiques autocorrectives* (3a. ed. rev.). Vic: Eumo Editorial.

Damerau, F. J. (1964, març). "A technique for computer detection and correction of spelling errors". A: *Communications of the ACM* (vol. 7, núm. 3, pàg. 171-176).

Fitzgerald, J., J. Elmore, H. Koons, E. H. Hiebert, K. Bowen, E. E. Sanford-Moore & A. J. Stenner. (2015). "Important text characteristics for early-grades text complexity".*Journal of Educational Psychology* 107, 4–29.

Gamon, M.; Leacock, C.; Brockett, C.; Dolan, W. B.; Gao, J.; Belenko, D.; Klementiev, A. (2009, juny). "Using Statistical Techniques and Web Search to Correct ESL Errors". *CALICO Journal* (vol. 26, núm. 3).

Gómez Guinovart, J. (1999). *La escritura asistida por ordenador: problemas de sintaxis y de estilo*. Vigo: Universidade de Vigo (Servicio de Publicacións).

Granger, S. (2003). "Error-tagged Learner Corpora and CALL: A Promising Synergy". *CALICO Journal* (vol. 20, pàg. 465-480).

Ido, Sinji. (2003). Agglutinative Information: A Study of Turkish Incomplete Sentences.*Otto Harrassowitz Verlag.*

Leacock, C.; Chodorow, M. (2003). "Automated grammatical error detection". A: M. D. Shermis, J. Burstein (editors).*Automated essay scoring: A cross-disciplinary perspective*. Hillsdale, NJ: Lawrence Erlbaum Associates, Inc.

Leacock, Claudia, Chodorow, Mike, Gamon, Michael, and Tetrault, Joel. (2014). Automated Grammatical Error Detection for Language Learners. Morgan Claypool Publishers. Series: Synthesis Lectures on Human Language Technologies.*Ed. Graeme Hirst*. Vol. 25, 2a edició.

Mitton, R. (1996). *English Spelling and the Computer. Londres: Longman.*

Quixal, M.; Badia, T.; Benavent, F.; Boullosa, J. R.; Domingo, J.; Grau, B.; Massó, G.; Valentín, O. (2008, maig). "User-Centred Design of Error Correction Tools". A: *Proceedings of the Sixth International Language Resources and Evaluation* (LREC'08). Marràqueix, el Marroc. URL: http://www.lrec-conf.org/proceedings/lrec2008/pdf/509_paper.pdf.

Samuel Reese, Gemma Boleda, Montse Cuadros, Lluís Padró, German Rigau. (2010). Wikicorpus: A Word-Sense Disambiguated Multilingual Wikipedia Corpus. *In Proceedings of 7th Language Resources and Evaluation Conference* (LREC'10), La Valleta, Malta. May, 2010.