Proposta rebuda el de juliol 2022 i acceptada per a publicació el d'octubre 2022.

El corpus paral·lel del *Diari Oficial de la Generalitat de Catalunya*

The parallel corpus of the Official Journal of the Catalan Government

Antoni Oliver
Universitat Oberta de Catalunya (UOC)

Resum

En aquest article presentem el procés de compilació de la nova versió del corpus paral·lel català–castellà creat a partir dels textos del *Diari Oficial de la Generalitat de Catalunya* (DOGC). Es descriuen els processos de descàrrega, conversió a text, segmentació i alineació automàtica. Tots els programes que s'han desenvolupat per dur a terme aquests processos es distribueixen amb una llicència lliure i el corpus compilat es pot descarregar lliurement. A més, es descriu el procés d'entrenament i avaluació de dos motors de traducció automàtica neuronal català–castellà i castellà–català que s'ha dut a terme fent servir aquest corpus paral·lel.

Paraules clau

corpus paral·lel; traducció automàtica neuronal

Abstract

In this paper, the process of compilation of the new version of the Catalan–Spanish parallel corpus of the Official Journal of the Catalan Government (DOGC) is presented. The processes of downloading, conversion to text, segmentation and automatic alignment are described. All the programs that have been developed to perform these processes are distributed under a free license and the compiled corpus can be freely downloaded. Furthermore, the process of training and evaluation of two neural machine translation systems, Catalan–Spanish and Spanish–Catalan, using this corpus is presented.

Keywords

parallel corpus; neural machine translation

1. Introducció

El *Diari Oficial de la Generalitat de Catalunya*[1] (DOGC) és el mitjà de publicació oficial de les lleis, normes, acords, resolucions, edictes, notificacions i anuncis de l'Administració i del Govern de Catalunya. El *Diari* té els seus inicis en el *Butlletí de la Generalitat de Catalunya*, el número 1 del qual aparegué el 3 de maig de 1931. El DOGC apareixia originàriament en paper i des del 2007 es va substituir íntegrament per una versió electrònica d'accés lliure. De la web del DOGC es poden descarregar textos en format electrònic des del número 2456 (18/08/1997), tot i que en aquests primers anys alguns textos no estan encara disponibles en HTML i només es poden obtenir en format PDF. Els textos provinents dels documents del DOGC tenen una llicència lliure i es poden distribuir i processar sense cap limitació legal. El tipus de llicència és la CC0 de Creative Commons[2].

El DOGC té dues edicions separades en català i en castellà. Molts dels textos del DOGC apareixen publicats tant en català com en castellà. Les normes amb rang de llei s'hi publiquen també en occità, així com les disposicions i els actes que afecten exclusivament l'Aran. L'aranès, llengua pròpia de la Vall d'Aran, és una variant de l'occità gascó. L'aranès és llengua oficial, junt amb el català i el castellà, a tot el territori de Catalunya.

En aquest article presentem el procés de creació de la nova versió del corpus DOGC, que comprèn els textos del 18/08/1997 al 31/12/2021. Aquesta nova versió actualitza la versió existent del corpus DOGC (Oliver, 2017), que comprenia els textos fins al 31/12/2015, i que, per tant, afegeix cinc anys més de la publicació. El resultat és un corpus català–castellà de temàtica administrativa i legislativa de gran mida. La versió anterior tenia un total de 5.026.847 segments paral·lels únics i la versió que presentem en aquest article assoleix els 8.472.786 segments paral·lels únics.

L'any 2016 l'Institut d'Estudis Catalans (IEC) va publicar una nova ortografia.[3]

[1] https://dogc.gencat.cat/

[2] https://creativecommons.org/publicdomain/zero/1.0/deed.ca

[3] https://www.iec.cat/llengua/documents/ortografia_catalana_versio_digital.pdf

DOI: 10.21814/lm.14.2.380

Els canvis principals es poden resumir en 6 punts:[4] (1) es modifica l'ús del guionet en alguns casos; (2) es redueix a quinze paraules la llista de mots amb accent diacrític: bé, déu, és, mà, més, món, pèl, què, sé, sí, sòl, són, té, ús i vós; (3) es recull la tradició de l'àmbit lingüístic valencià de representar amb accent agut els mots que es pronuncien amb e tancada en aquest parlar (que és oberta o neutra en altres parlars): café, comprén, francés, admés, etc.; (4) se suprimeix la dièresi en alguns derivats amb el sufix -al: laical, trapezoidal, etc.; (5) algunes paraules passen a doblar la r: arrítmia, cefalorraquidi, corresponsable, erradicar, otorrinolaringòleg, etc.; (6) algunes paraules compostes passen a escriure's amb e inicial del segon formant quan va seguida de s + consonant: arterioesclerosi, cirroestrat, electroestàtic, termoestable, etc. El canvi d'ortografia afecta els textos del DOGC des de la seva entrada en vigor, però amb dates canviants, ja que hi havia un període de transició on totes dues normatives estaven en vigor. Com veurem en l'article, s'ha desenvolupat un algorisme que adapta automàticament els textos a la nova normativa. Així doncs, els textos catalans del corpus resultant estan adaptats a la nova normativa ortogràfica.

Un corpus d'aquestes característiques pot tenir diverses utilitats, com per exemple estudis de llenguatges d'especialitat, treballs terminològics o l'entrenament de sistemes de traducció automàtica. En aquest treball presentem el procés d'entrenament de dos sistemes de traducció automàtica neuronal, un de català–castellà i un altre de castellà–català, i l'avaluació d'aquests sistemes, així com la comparació amb un sistema de transferència sintàctica superficial i amb un altre de neuronal.

Aquest corpus, per àmbit temàtic, es pot comparar amb els corpus paral·lels de la Unió Europea, com el JRC-Acquis[5] o el DGT.[6] Aquests corpus, però, estan disponibles per a molts parells de llengües, incloses el castellà i el portuguès. No tenim constància de la disponibilitat de corpus similars per a altres llengües de la península Ibèrica.

[4] https://www.uoc.edu/portal/ca/servei-linguistic/criteris/ortografia/resum-novetats/index.html

[5] https://joint-research-centre.ec.europa.eu/language-technology-resources/jrc-acquis_en

[6] https://joint-research-centre.ec.europa.eu/language-technology-resources/dgt-translation-memory_en

2. Compilació del corpus

Tots els programes i arxius necessaris per a descarregar, convertir a text, segmentar i alinear el corpus DOGC es distribueixen sota una llicència lliure (GNU-GPL v.3) i es poden descarregar del lloc GitHub del projecte.[7]

2.1. Descàrrega dels arxius HTML

Els diferents arxius HTML del DOGC es poden descarregar fent servir el programa `downloadDOGC.py`, que demana quatre paràmetres d'entrada: el número d'article inicial i el número d'article final a descarregar, el directori de sortida per als arxius HTML en català i el directori de sortida per als arxius HTML en castellà. Per saber com fer servir el programa es pot fer servir l'opció `-h`. Per poder fer servir el programa de descàrrega és necessari disposar del *chromedriver*[8] adequat per a la versió de Chrome instal·lat en el sistema.

Per exemple, per descarregar de l'article 90000 al 91000 i deixar-los als directoris `html-ca` i `html-es` podem escriure.

```
python3 dowloadDOGC.py
   --nummin 90000 --nummax 910000
   --outdirCAT ./html-ca --outdirSPA ./html-es
```

El programa introdueix unes pauses aleatòries entre descàrregues d'arxius d'entre 3 i 6 segons per a evitar que el servidor denegui el servei per descàrrega massiva. Per aquest motiu el procés de descàrrega pot durar molta estona. La velocitat mitjana que hem obtingut ha estat de 625 parells d'articles (article en català i el mateix article en castellà) per hora.

2.2. Classificació dels arxius per any de publicació

Aquest pas és opcional i s'ha d'executar únicament si estem interessats a tenir els arxius HTML organitzats per anys. Aquesta classificació es pot dur a terme amb els programes `classifyByYear-ca.py` (per als HTML en català) i `classifyByYear-es.py` (per als HTML en castellà). Els programes disposen de l'opció `-h` que mostra l'ajuda. Per classificar els HTML catalans que són al directori `html-ca` podem escriure:

```
python3 classifyByYear-ca.py --dirin html-ca
```

[7] https://github.com/aoliverg/corpusDOGC

[8] https://chromedriver.chromium.org/downloads

El programa crearà un directori `html-pre1995-ca` per a copiar tots els HTML amb data de publicació anterior a 1995 (si n'hi ha cap) i directoris `html-1996-ca`, `html-1197-ca...` per a cada any (a mesura que siguin necessaris). D'aquesta manera, una vegada acabada l'execució del programa disposarem de tots els HTML endreçats en directoris per any de publicació.

2.3. Conversió d'HTML a text

Una vegada tenim els arxius HTML descarregats, i opcionalment classificats per anys, cal convertir aquests arxius a text. Ens interessa molt convertir únicament el text de l'article, sense incloure tot el text addicional que pot presentar la plana web, com pot ser els menús superiors i els enllaços destacats de la part inferior. Per aquest motiu s'ha desenvolupat un algorisme ad-hoc que detecta quan comença i quan acaba l'article i únicament converteix aquesta secció a text. Aquesta conversió es pot dur a terme amb el programa `DOGC2textDIR.py`, que converteix a text tots els arxius HTML d'un determinat directori. El programa disposa de l'opció `-h` que mostra l'ajuda. Per convertir tots els arxius HTML del directori `html-2015-ca` a text i posar els arxius al directori `text-2015-ca`, podem escriure:

```
python3 DOCG2textDIR.py
     --dirin html-2015-ca
     --dirout text-2015-ca
```

2.4. Segmentació

Una vegada tenim els arxius convertits a text ens interessarà segmentar-los per a poder fer una posterior alineació automàtica segment a segment. Per a segmentar els arxius es pot fer servir el programa `text2segmentedtextDIR.py`, que segmenta tots els arxius de text d'un determinat directori i guarda els arxius segmentats en un altre directori. El programa disposa de l'opció `-h` que mostra l'ajuda. El programa fa servir el mòdul `srx_segmenter.py`,[9] que utilitza arxius de definició de regles de segmentació en el format estàndard SRX[10] (*Segmentation Rules eXchange*) (Miłkowski & Lipski, 2009).

Per segmentar tots els arxius de text del directori `text-2015-ca` i guardar els arxius segmentats al directori `seg-2015-ca`, fent servir l'arxiu SRX segment.srx per la llengua *Catalan*, podem escriure.

```
python3 txt2segmentedtextDIR.py
     -i text-2015-ca/ -o seg-2015-ca/
     -s segment.srx -l Catalan
```

2.5. Alineació automàtica

Per dur a terme l'alineació automàtica dels arxius de text segmentat hem fet servir Hunalign[11] (Varga et al., 2007). Per facilitar l'ús d'aquest programa hem desenvolupat una sèrie de programes auxiliars. El primer programa auxiliar és el `createAlignScript.py`, que permet la creació d'un script d'alineació. El programa disposa de l'opció `-h` que mostra l'ajuda del programa. Seguint els exemples, si volem crear l'*script* d'alineació dels arxius segmentats de l'any 2015, podríem escriure:

```
python3 createAlignScript.py
  --dirSL seg-2015-ca/ --dirTL seg-2015-es/
  --dirALI ali-2015-ca-es/
  --dictionary hunapertium-ca-es.dic
  --script align2015.sh
  --r1 ca.txt --r2 es.txt
```

Fixem-nos que podem indicar un diccionari d'alineació. Hem creat tota una sèrie de diccionaris d'alineació en format Hunalign a partir dels diccionaris de transferència del sistema de traducció automàtica Apertium[12] (Forcada et al., 2011). Aquests diccionaris es poden descarregar de Github.[13] Una vegada executem aquest programa obtenim un script que conté una línia per a cada parell d'arxius català–castellà, com en el següent exemple:

```
timeout 5m ./hunalign hunapertium-ca-es.dic
    -utf -realign -text
    "seg-2015-ca/700000-ca.txt"
    "seg-2015-es/700000-es.txt"
    > "ali-2015-ca-es/ali-700000-ca.txt"
```

Cada comanda comença definint un *timeout* de 5 minuts per evitar que el procés es detingui si en algun procés d'alineació es deté per algun error. Si donem permisos d'execució a aquest *script* i l'executem, s'alinearan automàticament tots els arxius indicats. Una vegada finalitza l'alineació obtenim tota una sèrie d'arxius que contenen el segment en català, el segment en castellà i un valor de confiança, que indica la seguretat de l'alineació. El programa `selectAlignments.py` permet seleccionar tots els segments dels arxius alineats d'un determinat directori que tingui un valor de confiança superior a l'indicat. El valor

[9] https://github.com/narusemotoki/srx_segmenter
[10] https://www.gala-global.org/srx-20-april-7-2008
[11] https://github.com/danielvarga/hunalign
[12] https://www.apertium.org/
[13] https://github.com/aoliverg/hunapertium

de confiança acostuma a estar comprès entre -0,3 i 10, i els valors superiors a 0 acostumen a indicar alineacions vàlides. Per exemple, per seleccionar totes les alineacions amb confiança superior a 0 podem escriure:

```
python3 selectAlignments.py
  --inDir ali-2015-ca-es/
  --outFile alineacio-2015-cat-spa.txt
  --confidence 0
```

Una vegada seleccionades les alineacions s'eliminen els segments repetits amb instruccions estàndard de Unix[14]. Cal tenir en compte que l'eliminació de segments repetits suposa una pèrdua de l'ordre d'aparició dels segments.

2.6. Conversió a la nova ortografia catalana

Per assegurar-nos que tot el corpus està en la nova normativa ortogràfica catalana de l'IEC hem adaptat el programa `MTUOC-novaIEC`,[15] perquè funcioni amb un corpus paral·lel en format de text tabulat on el català està en el primer camp. El programa per dur a terme aquesta conversió és el `modificaIEC-PC1.py`, que disposa de l'opció -h per mostrar l'ajuda. En l'exemple següent adaptem l'alineació de l'any 2015 a la nova normativa ortogràfica catalana.

```
python3 modificaIEC-PC1.py
  --infile alineacio-2015-cat-spa.txt
  --outfile DOGC-2015-cat-spa.txt
```

2.7. Neteja del corpus

Un cop convertit el corpus a la nova ortografia catalana es du a terme un procés de neteja fent servir el programa `MTUOC-clean-parallel-corpus`[16]. La neteja ha consistit en les següents accions: (1) normalització de l'apòstrof; (2) eliminació d'etiquetes HTML/XML; (3) conversió de les entitats HTML, si n'hi ha, en el seu caràcter corresponent; (4) correcció dels errors en la codificació de caràcters, si n'hi ha; (5) eliminació dels segments buits; (6) eliminació dels segments curts, de menys de 10 caràcters; (7) eliminació dels segments que continguin el 60% o més de caràcters numèrics; (8) eliminació dels segments que siguin iguals en les dues llengües; (9) verificació automàtica de les llengües. Un parell de segments català–castellà s'elimina si un dels dos components compleix la condició d'eliminació.

[14] `cat`, `sort`, `uniq`, `shuf`

[15] https://github.com/aoliverg/MTUOC-novaIEC

[16] https://github.com/aoliverg/MTUOC-clean-parallel-corpus

2.8. Corpus resultant

Hem dut a terme el procés de compilació del corpus DOGC fins als articles corresponents a 31 de desembre de 2021. Hem fet una classificació per anys, i a la taula 1 podem veure el nombre d'articles, segments i segments sense repeticions per a cada any (els anys anteriors al 2.000 estan agrupats, ja que el nombre d'articles disponibles electrònicament és baix per als anys anteriors). El corpus global sense repeticions es pot descarregar lliurement[17] i està publicat sota la mateixa llicència que el DOGC (CC0 de Creative Commons[18]). Aquest corpus també està disponible a la col·lecció Opus Corpus[19] (Tiedemann, 2012). A la taula 2 es poden observar les dades de mida total en segments del corpus resultant.

Any	Articles	Segments	Seg. únics
< 2000	34.766	1.818.109	992.442
2000	23.475	1.352.777	635.971
2001	27.145	1.828.328	810.281
2002	29.281	1.923.290	893.101
2003	28.587	2.093.726	979.864
2004	28.805	1.985.084	957.778
2005	32.294	1.919.271	873.322
2006	38.115	2.049.614	932.173
2007	34.926	2.244.405	1.008.292
2008	33.177	2.276.233	1.027.718
2009	32.328	2.167.934	986.409
2010	31.852	2.046.071	941.081
2011	25.475	1.135.272	562.417
2012	23.239	838.820	468.802
2013	23.018	1.220.403	606.889
2014	22.981	1.275.181	615.327
2015	22.514	2.041.743	855.160
2016	22.810	3.391.425	1.202.426
2017	24.445	3.367.881	1.249.182
2018	23.716	2.723.208	1.083.260
2019	22.583	3.178.228	1.471.438
2020	19.212	2.521.235	1.194.678
2021	21.185	1.767.391	773.634

Taula 1: Nombre d'articles, segments i segments únics per a cada any.

El corpus DOGC pot tenir diverses utilitats. Per una banda, pot ser una bona base per a l'estudi del llenguatge juridicoadministratiu en català i castellà. També pot resultar interessant per a estudis terminològics i per a la confecció de glossaris terminològics català–castellà mit-

[17] http://lpg.uoc.edu/corpusDOGC/DOGC-2021-cat-spa.zip

[18] https://creativecommons.org/publicdomain/zero/1.0/deed.ca

[19] https://opus.nlpl.eu/DOGC.php

	Segments
Total	47.165.629
Únic brut	16.787.380
Únic net	8.472.786

Taula 2: Nombre de segments totals, totals únics i totals únics una vegada fet el procés de neteja.

jançant tècniques d'extracció automàtica de terminologia i de cerca automàtica d'equivalents de traducció. També pot resultar un corpus molt adequat per a entrenar sistemes de traducció automàtica per a textos juridicoadministratius. En la pròxima secció presentem l'entrenament i avaluació de sistemes de traducció automàtica neuronal català–castellà i castellà–català fent servir el corpus DOGC.

3. Entrenament de sistemes de traducció automàtica neuronal

En aquest apartant es descriu el procés d'entrenament de dos sistemes de traducció automàtica neuronal amb el corpus DOGC. S'ha entrenat un sistema català–castellà i un altre castellà-català. Tots els passos de preprocessament i entrenament s'han dut a terme fent servir els programes i *scripts* del projecte MTUOC[20] (Oliver, 2020).

3.1. Preprocessament del corpus

El corpus s'ha preprocessat fent servir l'*script* `MTUOC-corpus-preprocessing`.[21] La segmentació en unitats lèxiques i el càlcul de les unitats subparaules s'han dut a terme amb SentencePiece[22] (Kudo & Richardson, 2018), amb les següents característiques:

- Joining languages: True
- Model type: bpe
- Vocabulary size: 64000
- Vocabulary threshold: 50

El primer paràmetre (*joining languages*) significa que el càlcul de les unitats subparaules es fa a la vegada per a les dues llengües. Aquesta és la pràctica habitual i recomanada per a llengües que, com en el nostre cas, comparteixen alfabet. La resta de paràmetres s'han fixat en valors habituals en aquest tipus d'entrenaments.

Aquest pas de segmentació en unitats lèxiques i l'ús d'unitats subparaules és necessari en els sistemes de traducció automàtica neuronal, ja que cada una d'aquestes unitats es codificarà com a un vector. Les unitats subparaules, és a dir, unitats més petites que una paraula, es fan servir per evitar el problema de la limitació d'elements en el vocabulari del sistema, ja que les paraules poc freqüents es representen mitjançant fragments de paraules, que són més freqüents i tenen una representació vectorial pròpia. El corpus s'ha dividit en un fragment de 5.000 segments de validació, 5.000 segments per avaluació i la resta per a l'entrenament.

3.2. Entrenament

L'entrenament s'ha dut a terme fent servir `marian-nmt`[23] (Junczys-Dowmunt et al., 2018), amb una configuració de tipus *transformer*. S'han fet servir dues mètriques de validació: *bleu-detok* (el valor de BLEU calculat un cop desfeta la divisió en unitats lèxiques i subparaules) i l'entropia creuada (*cross-entropy*). El criteri de parada (*early-stopping*) s'ha fixat en 5 per les dues mesures, és a dir, es deté l'entrenament quan les dues mesures no milloren en 5 o més validacions. La freqüència de validació s'ha fixat en 5.000. L'entrenament català–castellà ha necessitat 4 èpoques i un total de 46 validacions. En canvi, l'entrenament castellà-català ha necessitat 5 èpoques i un total de 64 validacions. L'entrenament s'ha executat en un ordinador amb una unitat GPU Nvidia Quadro P6000 de 24 GB. L'entrenament català–castellà ha durat 2 dies, 2 hores, 40 minuts i 29 segons; i l'entrenament castellà–català 3 dies, 1 hora, 2 minuts i 3 segons.

3.3. Avaluació

Per a avaluar els motors entrenats s'ha fet servir el programa `MTUOC-eval`[24], que pot proporcionar les següents mètriques automàtiques:

- BLEU (Papineni et al., 2002)
- NIST (Doddington, 2002)
- WER (*Word Error Rate*) (Nießen et al., 2000)
- Distància d'edició percentual (DE%)
- TER (*Translation Error Rate* o taxa d'error per mot) (Klakow & Peters, 2002)

[20] https://github.com/aoliverg/MTUOC-server
[21] https://github.com/aoliverg/MTUOC-corpus-preprocessing
[22] https://github.com/google/sentencepiece
[23] https://marian-nmt.github.io/
[24] https://github.com/aoliverg/MTUOC-eval

Com més altes siguin les mesures BLEU i NIST indiquen millor qualitat; mentre que les mesures WER, distància d'edició percentual i TER, com més baixes siguin, millor. Per a aplicar aquestes mesures s'han fet servir 1.000 segments dels 5.000 segments totals del corpus d'avaluació, assegurant que aquests mateixos segments no apareguin en els corpus d'entrenament ni de validació. Aquests segments d'avaluació s'han traduït amb els motors entrenats, amb Apertium[25] (Forcada et al., 2011) i amb Google Translate[26].

A la taula 3 es poden observar els valors d'avaluació dels diferents sistemes per a les dues direccions. En tots dos casos, els valors són millors per al nostre sistema entrenat amb Marian, amb unes diferències prou importants. Per exemple, en la direcció cat–spa el nostre sistema Marian supera a Apertium en 4,9 punts de BLEU i a Google Translate en 2,3 punts de BLEU. En la direcció spa–cat les diferències són del mateix ordre, superant el nostre sistema Marian a Apertium en 3,8 punts de BLEU i a Google Translate en 6,3 punts de BLEU.

3.4. Sistemes resultants

Els sistemes de traducció automàtica resultants es poden descarregar dels enllaços que s'indiquen al GitHub del projecte.[27] Els sistemes estan configurats mitjançant un servidor MTUOC-server[28] (Oliver, 2020). Funcionen en un entorn GNU/Linux (o un Windows Subsystem for GNU/Linux). Es distribueix també amb un *marian-server* compilat per a CPU, però pot funcionar també sota GPU si proporcioneu un *marian-server* compilat per a la vostra GPU.

Editant l'arxiu `config-server.yaml` es poden indicar els ports a fer servir i també quin tipus de servidor posar en marxa: MTUOC, Moses, OpenNMT, NMTWizard o ModernMT. MTUOC-server pot emular tots aquests protocols de manera que és compatible amb diversos programes clients (com eines de traducció assistida per ordinador, per exemple).

4. Conclusions i treball futur

En aquest article hem presentat el procés de compilació de la nova versió del corpus del *Diari Oficial de la Generalitat de Catalunya* (DOGC). Tant el corpus, com els algorismes i programes que s'han fet servir per a la descàrrega, conversió a text, segmentació, alineació i adaptació a la nova normativa ortogràfica catalana estan disponibles per a la descàrrega i ús.[29] Aquests programes poden servir d'exemple per a la compilació d'altres corpus a partir de llocs web.

En aquest article també s'ha presentat un cas d'ús del corpus DOGC: l'entrenament de dos sistemes de traducció automàtica neuronal. Els sistemes entrenats s'han avaluat automàticament i comparat amb dos sistemes molt emprats. Totes les mètriques automàtiques analitzades indiquen que la qualitat dels sistemes entrenats, per a la traducció de textos juridicoadministratius, és superior a la de dos sistemes molt emprats per a aquest parell de llengües.

Com a treball futur volem detectar els textos publicats en aranès al DOGC i alinear-los amb el català i el castellà. Tot i que es preveu que el nombre de textos disponibles en aranès sigui petit, es podran fer noves descàrregues i detteccions futures per anar ampliant aquests corpus. També pretenem fer la compilació d'altres corpus similars per al català i el castellà, concretament un corpus a partir del *Butlletí Oficial de les Illes Balears* (BOIB)[30] i del *Diari Oficial de la Generalitat Valenciana* (DOGV)[31]. En el cas del BOIB només seran necessari uns canvis mínims en els programes de descàrrega per indicar les adreces de descàrrega corresponents. En canvi, el DOGV difereix molt en la manera de mostrar els articles al navegador, per la qual cosa serà necessari una modificació més profunda, que encara no hem estudiat.

També tenim la intenció d'estudiar la viabilitat de l'ús del sistema neuronal castellà-català entrenat amb el DOGC per a crear corpus sintètics (Park et al., 2017) entre les llengües oficials de les Nacions Unides (anglès, francès, rus, àrab i xinès, a més de l'espanyol) i el català, traduint automàticament la part espanyola dels corpus paral·lels entre aquestes llengües i l'espanyol. Crearíem també aquests corpus sintètics fent servir Apertium. Amb els corpus resultants entrenaríem sistemes de traducció automàtica i els avaluaríem amb mètriques automàtiques. D'aquesta manera podrem determinar si l'ús del sistema neuronal per a crear els corpus sintètics aporta millores respecte a l'ús d'un sistema basat en regles per a l'entrenament de nous sistemes de traducció automàtica neuronal. Si l'experiència fos satisfactòria s'ampliaria l'estudi a la creació de corpus sintètics entre les llengües oficials de

[25] https://www.apertium.org/

[26] La traducció s'ha fet mitjançant l'API de Google en data 18/07/2022)

[27] https://github.com/aoliverg/corpusDOGC

[28] https://github.com/aoliverg/MTUOC-server

[29] https://github.com/aoliverg/corpusDOGC

[30] http://www.caib.es/eboibfront/

[31] https://dogv.gva.es/va/

	BLEU	NIST	WER	DE(%)	TER
cat–spa					
Aquest treball	87,3	13,396	0,078	5,040	0,076
Apertium	82,4	12,866	0,114	5,923	0,100
Google T.	85,0	13,216	0,095	5,158	0,081
spa–cat					
Aquest treball	88,7	13,51	0.083	4,950	0,068
Apertium	84,9	13,117	0,102	5,435	0,087
Google T.	82,4	13,038	0,115	5,701	0,099

Taula 3: Valors de les mètriques d'avaluació dels sistemes analitzats.

la Unió Europea i el català i entrenar també sistemes per a aquests parells de llengües. Actualment, només existeix un sistema comercial d'àmbit general que inclogui aquestes llengües, Google Translate.

Referències

Doddington, George. 2002. Automatic evaluation of machine translation quality using n-gram co-occurrence statistics. En *2nd International Conference on Human Language Technology Research (HLT)*, 138–145.

Forcada, Mikel L., Mireia Ginestí-Rosell, Jacob Nordfalk, Jim O'Regan, Sergio Ortiz-Rojas, Juan Antonio Pérez-Ortiz, Felipe Sánchez-Martínez, Gema Ramírez-Sánchez & Francis M. Tyers. 2011. Apertium: a free/open-source platform for rule-based machine translation. *Machine translation* 25(2). 127–144. doi 10.1007/s10590-011-9090-0.

Junczys-Dowmunt, Marcin, Roman Grundkiewicz, Tomasz Dwojak, Hieu Hoang, Kenneth Heafield, Tom Neckermann, Frank Seide, Ulrich Germann, Alham Fikri Aji, Nikolay Bogoychev, André F.T. Martins & Alexandra Birch. 2018. Marian: Fast neural machine translation in C++. En *56th Annual Meeting of the Association for Computational Linguistics (ACL: System Demonstrations)*, 116–121. doi 10.18653/v1/P18-4020.

Klakow, Dietrich & Jochen Peters. 2002. Testing the correlation of word error rate and perplexity. *Speech Communication* 38(1-2). 19–28. doi 10.1016/s0167-6393(01)00041-3.

Kudo, Taku & John Richardson. 2018. SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. En *2018 Conference on Empirical Methods in Natural Language Processing (EMNLP: System Demonstrations)*, 66–71. doi 10.18653/v1/D18-2012.

Miłkowski, Marcin & Jarosław Lipski. 2009. Using SRX standard for sentence segmentation. En *Language and Technology Conference (LTC)*, 172–182. doi 10.1007/978-3-642-20095-3_16.

Nießen, Sonja, Franz Josef Och, Gregor Leusch & Hermann Ney. 2000. An evaluation tool for machine translation: Fast evaluation for MT research. En *2nd International Conference on Language Resources and Evaluation (LREC)*, .

Oliver, Antoni. 2017. El corpus paral·lel del Diari Oficial de la Generalitat de Catalunya: compilació, anàlisi i exemples d'ús. *Zeitschrift für Katalanistik* 30. 269–291.

Oliver, Antoni. 2020. MTUOC: easy and free integration of NMT systems in professional translation environments. En *22nd Annual Conference of the European Association for Machine Translation*, 467–468.

Papineni, Kishore, Salim Roukos, Todd Ward & Wj Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. En *40th annual meeting of the Association for Computational Linguistics (ACL)*, 311–318. doi 10.3115/1073083.1073135.

Park, Jaehong, Jongyoon Song & Sungroh Yoon. 2017. Building a neural machine translation system using only synthetic parallel data. *ArXiv* abs/1704.00253.

Tiedemann, Jörg. 2012. Parallel data, tools and interfaces in OPUS. En *8th International Conference on Language Resources and Evaluation (LREC)*, 2214–2218.

Varga, Dániel, Péter Halácsy, András Kornai, Viktor Nagy, László Németh & Viktor Trón. 2007. Parallel corpora for medium density languages. En *Recent Advances in Natural Language Processing IV*, 247–258. John Benjamins. doi 10.1075/cilt.292.32var.