

Received July 15, 2019, accepted August 16, 2019, date of publication August 26, 2019, date of current version September 12, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2937505

A System for Automatic English Text Expansion

SILVIA GARCÍA-MÉNDEZ¹, MILAGROS FERNÁNDEZ-GAVILANES²,
ENRIQUE COSTA-MONTENEGRO¹, JONATHAN JUNCAL-MARTÍNEZ¹,
FRANCISCO JAVIER GONZÁLEZ-CASTAÑO¹, AND EHUD REITER³

¹GTI Research Group, Telematics Engineering Department, School of Telecommunications Engineering, University of Vigo, 36310 Vigo, Spain

²Defense University Center, 36920 Pontevedra, Spain

³CLAN Research Group, Department of Computing Science, University of Aberdeen, Aberdeen AB24 3UE, U.K.

Corresponding author: Silvia García-Méndez (sgarcia@gti.uvigo.es)

This work was supported in part by the Mineco, Spain, under Grant TEC2016-76465-C2-2-R, in part by the Xunta de Galicia, Spain, under Grant GRC-2018/53 and Grant ED341D R2016/012, and in part by the University of Vigo Travel Grant to visit the CLAN Research Group, University of Aberdeen, U.K.

ABSTRACT We present an automatic text expansion system to generate English sentences, which performs automatic Natural Language Generation (NLG) by combining linguistic rules with statistical approaches. Here, “automatic” means that the system can generate coherent and correct sentences from a minimum set of words. From its inception, the design is modular and adaptable to other languages. This adaptability is one of its greatest advantages. For English, we have created the highly precise aLexiE lexicon with wide coverage, which represents a contribution on its own. We have evaluated the resulting NLG library in an Augmentative and Alternative Communication (AAC) proof of concept, both directly (by regenerating corpus sentences) and manually (from annotations) using a popular corpus in the NLG field. We performed a second analysis by comparing the quality of text expansion in English to Spanish, using an ad-hoc Spanish-English parallel corpus. The system might also be applied to other domains such as report and news generation.

INDEX TERMS Augmentative and alternative communication, natural language generation, sentence planning, surface realiser, text expansion.

I. INTRODUCTION

Natural language generation (NLG) systems [1] take quantitative, visual and linguistic data (words, sentences, texts) as input. In particular, we are interested in text-to-text expansion, which generates complete sentences or even texts from some meaningful words. A text expansion system must add elements like conjunctions and prepositions to transform the input words into linguistically correct outputs. For example, given the input words ‘she’, ‘look’, ‘picture’, ‘yesterday’, ‘not’, the output might be ‘She did not look at the picture yesterday’. This is useful in Augmentative and Alternative Communication (AAC) [2], for instance. In this case, a user would select pictograms corresponding to input words on a tablet to obtain the equivalent linguistically correct written and oral output.

Our approach, which is modular and extensible to other languages, is based on our previous work on automatic NLG systems for Spanish [3], [4]. We have improved modularity that allows to easily replace domain-dependent components.

The associate editor coordinating the review of this manuscript and approving it for publication was Santhosh Kumar Gopalan.

The rest of this paper is organised as follows. In Section II we review the state-of-the-art, paying special attention to English NLG, automatic generation and text expansion. In Section III we describe our contribution, a system for English text expansion based on the aLexiE English lexicon (Section III-A), a grammar (Section III-B), and a sentence planner and a surface realiser (Section III-C). In Section IV we present our evaluation results, based on a widely used corpus in the NLG field (Section IV-B.1) and an ad-hoc Spanish-English parallel corpus (Section IV-B.2). Finally, Section V concludes the paper.

II. RELATED WORK

NLG tasks generally follow well-defined sub-tasks [1], [5]: content determination, text structuring, sentence aggregation, lexicalisation (expressing information with the right words), referring expression generation (domain objects identification), and linguistic realisation (text correctness).

Content determination filters irrelevant information. It is obviously context- and application-dependent. For this sub-task, it has been proposed to apply data-driven techniques [6].

Text structuring plans the discourse and sets the sentence presentation order. Recent solutions are based on manual rules [7], [8] and Systemic Functional Grammar (SFG) [9], Rhetorical Structure Theory (RST) [10], Meaning-Text Theory (MTT) [11], Machine Learning [12], [13], and the Centred Theory [14], to cite some.

Semantic- and syntactic-level sentence aggregation join data into single fluent and readable sentences [15].

Lexicalisation transforms the result of sentence aggregation into *natural language* (NL) but it must choose between different expression alternatives in NL. Generally, results improve with more alternatives [16].

Referring Expression Generation (REG) selects the phrases or words that can unambiguously describe domain entities. It selects the best properties to distinguish elements, and it discards irrelevant information. REG algorithms include the Full Brevity Algorithm [17], the Greedy Heuristic [18], [19], and the Incremental Algorithm [20].

Linguistic realisation comprises generation of morphological forms as well as insertion of auxiliary verbs, prepositions and punctuation marks. It may fall into the *generation gap*, because input data are often incomplete (i.e. they lack elements to be syntactically comprehensible) [21]. Templates avoid this gap and ensure consistency. However, using templates to automatically transform data into text (which are typically used in applications such as weather, traffic, sports, and health reporting) only yields better results than other approaches in small domains with little variation. Hand-coded grammar-based systems outperform templates when detailed input is available as in the case of KPML¹ [9]. There are other alternatives like statistical methods that produce probabilistic grammars from large corpora increasing coverage at less effort [22]; such as the Head-driven Phrase Structure Grammar (HPSG) [23]; the Lexical-Functional Grammar (LFG) [24]; the Tree-Adjoining Grammar (TAG) [25], and some Deep Learning approaches [26]. We employ a hybrid system that combines the advantages of stochastic and grammar-based systems with low NLG complexity since it uses only keywords to generate complete sentences.

Intelligent NLG architectures may be modular, (roughly following the previous stages of macro-planning and text structuring, micro-planning or sentence aggregation, lexicalisation, referring expression generation, and linguistic realisation with syntactic and morphological rules); planning-oriented (which are less modular); or data-driven, based on statistical learning. Rule- (or template-) based approaches [27], [28], however, are the most extended nowadays in real applications.

According to types of output texts, these may be informative texts, summaries, dialogues, recommendations, and persuasions or creative writings. Most informative text systems generate routinary information from quantitative

data [29]–[31]. Summary generation [32] has applications in areas such as medicine, sports, and finance. Persuasive texts are intended to shape user behaviour [33]. Dialogue systems, of interest for call centres or gaming interfaces, focus on human-machine communication [34], [35]. Creative text generation is extremely difficult. It has been demonstrated that predefined templates are too rigid for it [36]. Affective NLG systems have been able to generate texts beyond factual information such as poetry [37], but the operator has no control over the process. The quality of the results in NLG is measured in terms of adequacy, fluency, readability, and variation [38]. There is a trade-off between efficiency and output quality.

Next we review the most relevant existing systems. SimpleNLG [39] performs surface realisation by following a knowledge-based approach. Originally for English, it is now available for German [40], French [41], Brazilian Portuguese [42], Italian [43], Spanish [44], [45], Dutch [46], Mandarin [47], and Galician [48]. This library has had strong impact in NLG due to its simple usage. Its main disadvantage from the perspective of our research is its manual operation. The NaturalOWL [49] data-to-text manual tool imposes a complex input format. It generates texts from the OWL ontology. SumTime [50], [51], also for data-to-text generation, is highly sensitive to language variations and, thus, it is only adequate for the language it was designed for.

The system in [52] comprises a trainable sentence planner and a probabilistic surface realiser. Its modular design is similar to ours, but it is only available for English and it is not easily adaptable to other languages since certain language-dependent resources are required to train the surface realiser (a TAG grammar, the source of the supertags to annotate the semi-specified TAG derivation tree, a treebank to obtain the tree stochastic model driving the tree chooser and a corpus of sentences to train the language model required for the linear precedence chooser). Our system also needs two language-dependent elements, the lexicon and the grammar, but they can be easily replaced and we provide enough information to create/adapt these elements to any domain.

In the particular field of text expansion in English applied for AAC, we must mention the work by [53], in line with automatic generation. Their *sentence companionship* technique [54], [55]² takes a compressed message and expands it into a well-formed sentence. In practice it is useful as a writing assistant or a conversational aide in situations where grammatically correct output is desired. Its bottleneck is its generation method, based on a unification-type grammar that needs to explore many possibilities to deliver its output. Thus, it is highly time consuming. Besides, most of its operation is based on markers linked to a lexicon, as in the case of plural forms and possessives (the system might interpret the noun ‘apple’ followed by a plural marker as ‘apples’; the pronoun ‘I’ followed by a possessive marker would be interpreted as ‘my’). Input words must be present in the lexicon,

¹Available at <http://www.fb10.uni-bremen.de/anglistik/langpro/kpml/README.html>, July 2019.

²We thank Kathleen F. McCoy for informing us about project COMPANION.

since they are taken from a word board and the user is not free to enter them. To avoid this constraint we have designed an automatic procedure for lexicon acquisition. It does not need markers, so that the entire generation process is much faster.

Summing up, most NLG systems are purpose-built and, as such, they are highly sensitive to problem characterization. On the contrary, thanks to the modularity of our system we can isolate domain-dependent modules (grammar and lexica) from domain-independent ones (NLG surface realiser). It can be tailored to different applications and domains using the corresponding syntactic structures and vocabularies. It can be easily extended to other languages as well.

As far as we know there are no other systems for automatic text-to-text expansion in English with a hybrid architecture.

III. METHODOLOGY AND ARCHITECTURE OF THE PROPOSED SYSTEM

A. ALEXIE LEXICON

This section describes the morphological part, that is, the lexicon providing linguistic knowledge. This is our first contribution.

We pursue a fully automatic NLG system. It must select the grammar structure for the input words and their inflection. Therefore we need an ample vocabulary with linguistic data. The aLexiE lexicon serves this purpose. We created it by interpreting input resources and automatically (without human supervision) merging them with the two-step methodology in [56] followed by a final verification step (similarly to [4]):

- 1) Extraction of all possible entries and translation to a common format (Algorithm 1).
- 2) Automatic comparison and combination of existing lexica to create the new resource (graph unification in [57] and [58]).
- 3) Lexical verification of extracted and translated entries and their categories against the Merriam-Webster Dictionary³ (MWD) (Algorithm 2).

1) LINGUISTIC RESOURCES AND CREATION OF ALEXIE

We built aLexiE from free English linguistic resources. We prioritized correctness and coverage and selected the following:

- The morphological and syntactic English lexicon from the Alexina Project⁴ (EnLex) [59].
- The Specialist Lexicon⁵ (NIH) of medical terms and everyday words.
- The Freeling English dictionary⁶ (EN-FREELING), automatically extracted from wsj (Wall Street Journal) and other corpora.

³Available at <https://www.merriam-webster.com>, July 2019.

⁴Available at <https://gforge.inria.fr/projects/alexina>, July 2019.

⁵Available at <https://lsg3.nlm.nih.gov/LexSysGroup/Projects/lexicon/current/web/index.html>, July 2019.

⁶Available at <http://nlp.lsi.upc.edu/freeling/index.php/node/12>, July 2019.

We performed extraction and mapping independently. Once the information was taken from the selected resources, it was transformed to a common format. Unlike for the Spanish version in [4], input resources were unrelated, and thus we conducted independent extraction and translation stages for each selected resource.

We first extracted entries from EnLex tagged as noun, pronoun, verb, adjective, adverb, determiner, conjunction, and preposition, ignoring interjections, numerals, and proper nouns. Each resulting EnLex word entry was translated to the extensional Alexina format [60]. As an example, Listing 1 illustrates the EnLex entry for the English lemma ‘picture’. It can be observed that this noun (cat = n), has two forms, ‘picture’ and ‘pictures’, respectively for masculine singular and plural.

```
picture N2 100;Lemma;N;;cat=N;
%default #dela+mtext+init
<table name="N2" rads=".*">
  <form suffix=" " tag="s"/>
  <form suffix="s" tag="p"/>
</table>
```

LISTING 1. Example of the English lemma ‘picture’ in EnLex.

We did the same for the other two linguistic resources that we selected: NIH⁷ and the Freeling English Dictionary.⁸

We kept the present, past, present participle, and past participle forms of English verbs. This information allowed adjusting the verbal tense to context-dependent semantic features. In the case of adjectives, we did not save the comparative and superlative forms (we leave comparative and superlative clauses to future work).

The merging process requires to handle the issue of the different formats and tags of word entries in the selected resources. Algorithm 1 converts them to a common format (note: *e* is an entry in a lexicon).

Verification in Algorithm 2 checks the quality of the word entries. It looks for each lemma and its lexical categories in MWD (this dictionary has the advantage that it allows more web queries than other freely available online dictionaries).

Finally, collected entries are merged in a combination step that applies the graph unification in [57] and [58]. This operation validates common information by integrating data of different nature and discarding inconsistent information. Specifically,

- 1) It joins all entries with a common lemma (homography is only considered for different lexical categories).
 - a) For the entries that results from (1), feature structures are unified.
 - b) Next, a new aLexiE entry is created with these structures. The entry comprises all common

⁷We extracted adjectives, adverbs, conjunctions, determiners, prepositions, and pronouns from this resource. There was not inflection information in the entries.

⁸From this resource we extracted adjectives, adverbs, and verbs.

Algorithm 1 Extraction and Conversion Algorithm

```

function extraction_mapping({ENLEX},{NIH},
{EN-FREELING})
  for  $e_{ENLEX} \in \{ENLEX\}$  do
    lemma $_{e_{ENLEX}}$  =  $e_{ENLEX}$ .getLemma()
    category $_{e_{ENLEX}}$  =  $e_{ENLEX}$ .getCategory()
  end for
  for  $e_{NIH} \in \{NIH\}$  do
    lemma $_{e_{NIH}}$  =  $e_{NIH}$ .getLemma()
    category $_{e_{NIH}}$  =  $e_{NIH}$ .getCategory()
  end for
  for  $e_{EN-FREELING} \in \{EN-FREELING\}$  do
    lemma $_{e_{EN-FREELING}}$  =  $e_{EN-FREELING}$ .getLemma()
    category $_{e_{EN-FREELING}}$  =  $e_{EN-FREELING}$ .getCategory()
  end for
  return ({ENLEX},{NIH},{EN-FREELING})
end function

```

Algorithm 2 Verification Algorithm

```

function VERIFICATION({SET})
  for  $e_{SET} \in \{SET\}$  do
    lemma $_{e_{SET}}$  =  $e_{SET}$ .getLemma()
    category $_{e_{SET}}$  =  $e_{SET}$ .getCategory()
    if (lemma $_{e_{SET}}$ .isNOTinMWD() OR
category $_{e_{SET}}$ .isNOTinMWD())
      then
        {SET}.delete( $e_{SET}$ )
      end if
    end for
  return ({SET})
end function

```

Algorithm 3 Lexicon Building Algorithm

```

{ENLEX} = LoadEnLex()
{NIH} = LoadNIH()
{EN-FREELING} = LoadEn-Freeling()
EXTRACTION_MAPPING({ENLEX},{NIH},{EN-FREELING})
{ALEXIE} = {ENLEX}  $\cup$  {NIH}  $\cup$  {EN-FREELING}
VERIFICATION({aLexiE})

```

information plus any particular data in the source entries.

- 2) A new aLexiE entry is created for any lexical entry that cannot be generated by combining entries from other lexica.

We remark that the common extraction and translation format avoids inconsistencies in this merging procedure. Algorithm 3 combines all steps in this section so far.

Therefore, aLexiE was built from inputs extracted from previously existing resources, which were merged into a common format and finally verified. Listing 2 shows the result for the lemma ‘picture’.

```

<?xml version="1.0" encoding="UTF-8"
standalone="no"?>
<lexicon >
  <word>
    <lemma>picture </lemma>
    <category>noun </category>
    <number>singular </number>
    <plural>pictures </plural>
    <semantic_tag>object </semantic_tag>
  </word>
</lexicon >

```

LISTING 2. Example of the English lemma ‘picture’ in aLexiE.

Algorithm 4 Adding Syntactic and Semantic Data

```

function ADD_SYNTACTIC&SEMANTIC_DATA({ALEXIE})
  for  $e_{ALEXIE} \in \{ALEXIE\}$  do
    lemma $_{e_{ALEXIE}}$  =  $e_{ALEXIE}$ .getLemma()
    category $_{e_{ALEXIE}}$  =  $e_{ALEXIE}$ .getCategory()
    if (category $_{e_{ALEXIE}}$ .isNoun() AND lemma $_{e_{ALEXIE}}$ .
.isInMCR())
      then
         $e_{MCR}$  = searchInMCR(lemma $_{e_{ALEXIE}}$ )
        {MCR}.add( $e_{MCR}$ )
      end if
    end for
end function

```

Note that, in Listing 2, this lemma is semantically tagged as an object. We used the Multilingual Central Repository⁹ (MCR) [61] to get the semantic classification of nouns in aLexiE. Algorithm 4 summarises this procedure.

Due to its size, indexing aLexiE allows our system to conduct the whole NLG process much more quickly.

2) AUTOMATIC EXTENSION OF THE LEXICON

NLG can be simplified by avoiding inputs with little meaning, such as prepositions. Consequently, it is necessary to infer a priori which preposition follows a particular verb. Indeed, a major challenge in text expansion is inferring missing prepositions. We trained this process from the text in the English Wikipedia, which was previously POS-tagged with Spacy Tagger.¹⁰ The language model for this training was based on trigrams centered around verbs, using syntactic and semantic information.

As previously mentioned, we used MCR to get the semantic classification of the entries tagged as nouns in aLexiE. In this case we established four semantic categories to start working with: living things, foodstuff, places, and objects. For each verb lemma in the training set we analysed if it was followed by a preposition and a noun or a determiner and a noun. We computed the probability by semantic category. Let us

⁹A lexical database that integrates the Spanish WordNet into the EuroWordNet framework, see <http://adimen.si.ehu.es/web/MCR>, July 2019.

¹⁰Available at <https://spacy.io>, July 2019.

TABLE 1. Automatic lexicon extension. Sentence examples.

Sentence	After processing
I look the business.	look + EMPTY + object
She looks at the picture.	look + at + object
You look at the car.	look + at + object
She looks like her mum.	look + like + living thing

TABLE 2. Automatic lexicon extension probabilities by lemma verb and semantic category.

After processing	Probability
look + EMPTY + object	0.33
look + at + object	0.66
look + like + living thing	1

consider the entries in Table 1. Regarding the verb lemma ‘look’ and the semantic tag *object*, the preposition ‘at’ has the highest probability to go in between according to Table 2.

In this way, the language model along with the semantic classification allow us to infer the most suitable preposition after a verb by applying semantic knowledge rather than by only considering morphological forms.

Listing 3 shows the aLexiE entry for verb ‘look’. Note how the language model has learned to add the preposition ‘at’ when the verb is followed by an object (semantically speaking) or ‘for’ in the case of foodstuff or a place. If ‘look’ is followed by a living thing, the system will add the preposition ‘like’. In the running example in this section, since ‘picture’ is tagged as an object in aLexiE, and given the syntactic and semantic data in the ‘look’ entry; the system will insert preposition ‘at’ between the two words if provided as input to the system.

B. SYNTACTIC STRUCTURE SUPPORTED BY A GRAMMAR

In this section we describe the syntactic stage of our system. It performs syntactic structuring with the Definite-Clause Grammar (DCG) [62].

```
<?xml version="1.0" encoding="UTF-8"
standalone="no"?>
<lexicon>
<word>
<lemma>look </lemma>
<category>verb </category>
<present3s>looks </present3s>
<past>looked </past>
<present_participle>
looking
</present_participle>
<past_participle>
looked
</past_participle>
<object>at </object>
<foodstuff>for </foodstuff>
<living>like </living>
<place>for </place>
</word>
</lexicon>
```

LISTING 3. Example of the English lemma ‘look’ in aLexiE.

Syntactic structuring, also called parsing, creates the tree structure of the desired target sentence. We infer this structure by checking the syntactic trees from the grammar for the input words. Obviously diverse possible syntactic structures may result, depending on the roles of the input words in the sentence. Fig. 1 shows the syntax trees from the grammar for the input words ‘she’, ‘look’, ‘picture’; ‘She looks at the picture’ and ‘She looks the picture’. Context-Sensitive Languages (CSL) are created from this type of grammar. The system picks the most appropriate trees for the input words given the different possibilities within a grammar.

For the English case, we adapted the simple grammar with wide range of basic sentences described in [4], by adding all its linguistics features, such as adjectives preceding nouns. The system can parse sentences regardless of their complexity. Sentence types may be affirmative, negative, interrogative (in positive or negative form) or imperative (in positive

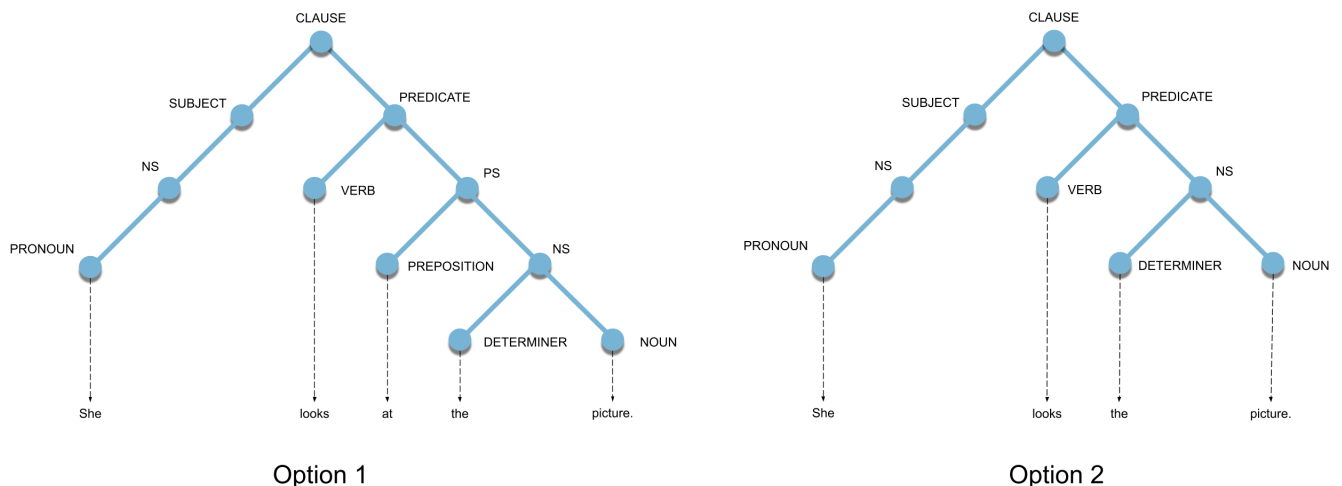


FIGURE 1. Syntax tree resulting from the grammar.

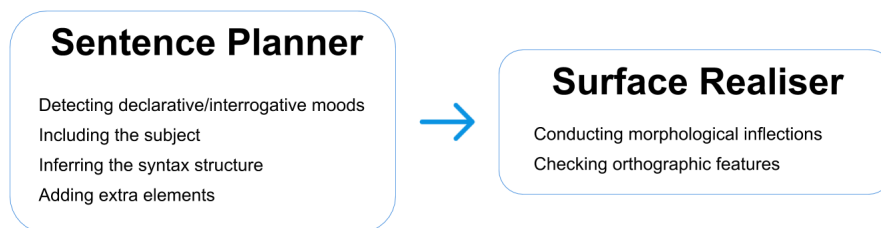


FIGURE 2. Our two-stage NLG architecture.

or negative form), including some of the following features: a nominal syntagm subject; a coordinated nominal syntagm subject (compound subject); a nominal syntagm direct complement; a coordinated nominal syntagm direct complement; an indirect complement, and other place or time complements.

In our notation, upper case corresponds to tree structures and lower case to word components. Fig. 1 illustrates some linguistic rules taken from the grammar.

1) RULES FOR NOMINAL/COORDINATED NOMINAL SYNTAGMS

Nominal syntagms (NS) contain nouns, pronouns or proper nouns. Determiners may precede nouns. A coordinated syntagm is a sentence with two nominal syntagms connected by a conjunction.

2) RULES FOR ADJECTIVAL/ADVERBIAL/PREPOSITIONAL SYNTAGMS

In this case, an adjectival or adverbial syntagm, which consists of an adverb followed by an adjective or vice versa, may precede or follow a noun. Noun-Noun modifiers such as in 'car door' are not considered. In prepositional syntagms (PS), unless empty, a preposition precedes a nominal syntagm. Prepositional syntagms just follow (never precede) a nominal syntagm.

3) PREDICATE RULE

The sentence predicate contains a verb that may be followed by a nominal syntagm (coordinated or not). The verb may be accompanied by an adjectival/adverbial syntagm. Note that a verb can be followed by two nominal syntagms (yet not coordinated ones, that is, without a conjunction in between), as in sentence 'She gave me a cookie yesterday'.

4) SENTENCE RULE

Sentences are composed either of a nominal or coordinated nominal syntagm (subject) and a predicate, or of a single predicate (without subject). The latter is quite common in imperative English clauses. Given the relations among syntagms, the depth level of our system is limited to two iterations to reduce computational load. For example, in case a nominal syntagm includes a prepositional

syntagm, the second nominal syntagm cannot contain another prepositional syntagm (to avoid recursion).

C. PROPOSED NLG LIBRARY: SENTENCE PLANNER AND SURFACE REALISER

The input words for our NLG library should be meaningful, such as adjectives, nouns, and verbs. The library can automatically infer the determiners, conjunctions, and prepositions that complement those input words in the output sentence.

Fig. 2 shows our two-stage architecture, an automatic NLG processing pipeline. The user introduces the words (plus symbol ? for interrogative sentences) in [subject, verb, object] (SVO) order, which is not limiting in practice in many domains. The first Sentence Planner stage performs lexicalisation, which adds words and configures sentences. The second Surface Realiser stage, our main contribution, introduces any extra elements that may be necessary and applies morphology inflections to produce grammatically correct and coherent sentences. Fig. 3 represents an example of generation using the library. Fig. 4 shows the flowchart of our NLG library. The main tasks are the following ones:

1) DETECTION OF THE LINGUISTIC STRUCTURE (AFFIRMATIVE, NEGATIVE OR INTERROGATIVE) OF THE SENTENCE (SENTENCE PLANNER)

The sentence is considered negative if one of the input words is the negation adverb 'not'. It is treated as interrogative if the input words include a question mark. If both elements are present, a negated question is generated by the system. The sentence is considered affirmative in any other case. This is the case in our example in Fig. 3. The library also adds extra elements corresponding to any linguistic realisations in the grammar and the knowledge in aLexiE.

2) SUBJECT INSERTION (SENTENCE PLANNER)

Imperative sentences and other sentences with elided subjects are quite common in English, for example 'Go to your room'. We want the NLG process to be almost transparent to end users and, thus, if the user does not provide a subject, the library takes the personal pronoun 'I' as such. Besides, it generates a second option with elided subject in case the user wants to create an imperative clause. In our example in Fig. 3 and Fig. 4, the library does not include a subject because the

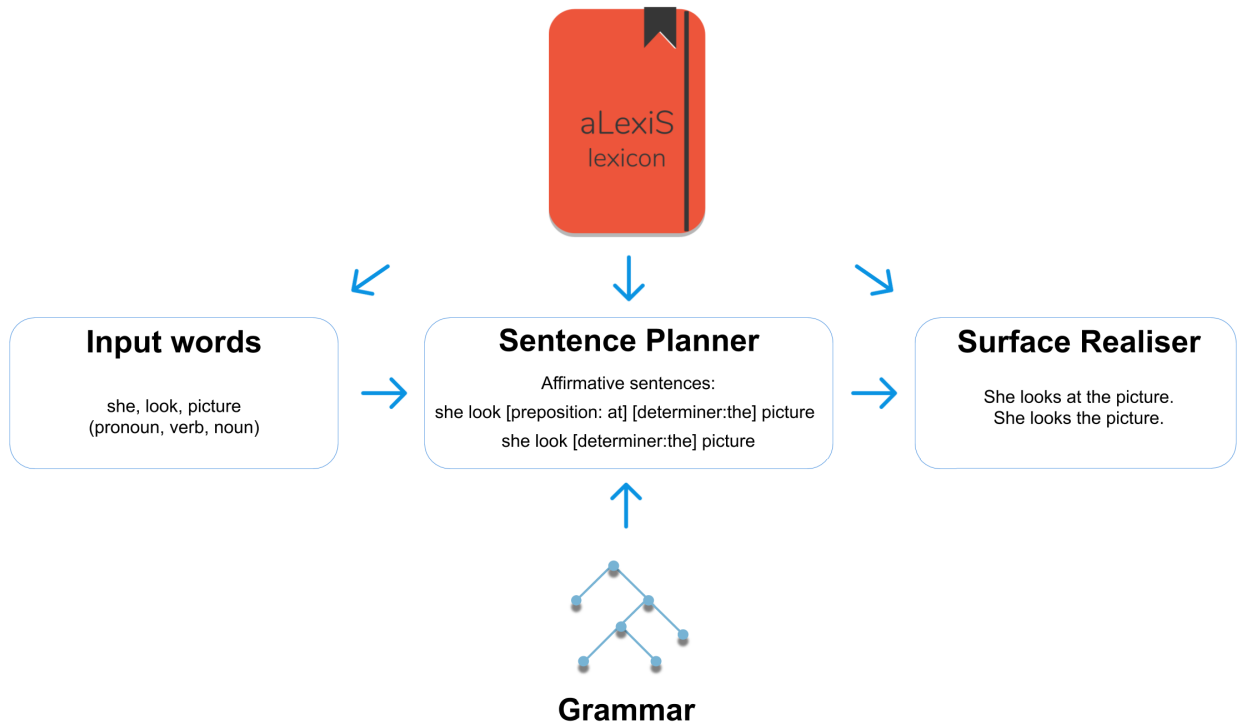


FIGURE 3. Sentence generation with our two-stage NLG library.

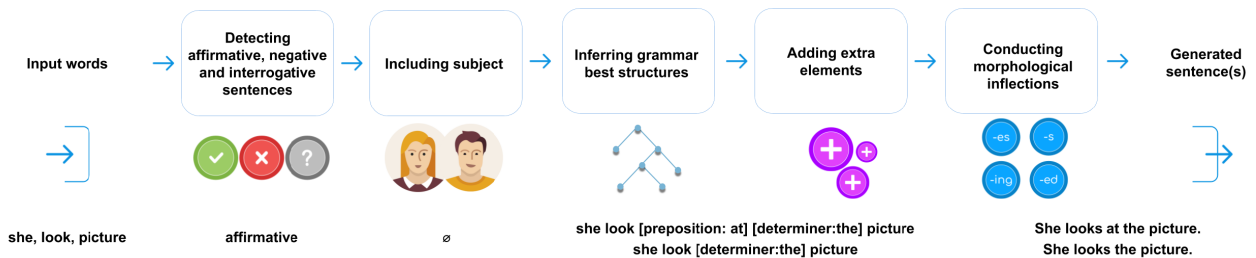


FIGURE 4. Flow chart of the NLG procedure.

user introduces ‘she’ as input. This is detected thanks to the grammar.

3) INFERENCE OF SYNTAX STRUCTURE (SENTENCE PLANNER)

The separation between subject and predicate simplifies the identification of the best syntactic trees for the input words, since they are smaller. Once the Sentence Planner decides the type of sentence, the library sets the boundary between the subject and the predicate taking the main verb as a reference and then searches for the best syntactic structure that matches them. For this purpose we follow a Depth-First Search (DFS) [63] in our grammar departing from the input words. In case that some of the input words are not in the aLexiE lexicon, they will be treated as proper nouns. This means no inflections will be applied to them.

In our example in Fig. 1, the system infers two possible syntax structures (options 1 and 2).

4) INCLUSION OF EXTRA ELEMENTS (SENTENCE PLANNER)

Once the syntactic structure is decided, some extra elements such as determiners, prepositions, and conjunctions may be necessary. These elements are inserted in the sentence if they are associated to feasible grammar realisations. In our example in Figs. 3 and 4, the library adds to the output the extra elements that were inferred in the previous stage from the grammar.

5) MORPHOLOGICAL INFLECTIONS (REALISER)

This encompasses the inflections that are necessary to produce a grammatically correct sentence, in which the subject dictates the morphological conjugation, person, number, and gender inflections of the verb and other components.

The library distinguishes between subject and predicate before generating a sentence. In this regard, it can apply linguistic features to adjust person, gender, and number, to create sentences with coordinate subject. For example, given

TABLE 3. Results of our NLG library.

Input words	Best output sentence/s
something, be, not, right	Something isn't right.
where, my, glasses, be, ?	Where are my glasses?
dinner, be, good, last, night	Dinner was good last night.
appreciate, your, help, concern	I appreciate your help and concern.
live, yellow, house	I live in the yellow house.
how much, stamps, be, these, days, ?	How much are stamps these days?
final, grades, be, available, after, class, today, ?	Are final grades available after class today?

the input words ‘caregiver’, ‘I’, ‘eat’, ‘apples’, the subject of the resulting sentence ‘The caregiver and I eat apples’ is compound.

First, person, gender, and number features must be derived from the input words. The subject (expected to be a nominal or coordinated nominal syntagm) determines them. Continuing with the running example ‘The caregiver and I eat apples’, the subject is a coordinated nominal syntagm. The first nominal syntagm within is composed of the determiner ‘the’ and the singular noun ‘caregiver’. The second is the pronoun ‘I’. Consequently, the person and number of the sentence are first person and plural, and the verb ‘eat’ is inflected accordingly.

By default, the first time the user introduces input words, the library takes masculine gender, singular number and first person. Then, using aLexiE, it adapts these features with grammar rules. For instance, if the subject is a coordinated nominal syntagm, the output sentence is plural. Regarding gender, the output sentence is only feminine if all subject components are so.¹¹ The following rules are applied in strict order to adjust the person feature: (1) if the subject has an element that refers to the first person, the output sentence is adjusted to that person; (2) if the sentence has an element that refers to the second person with no relation with the first person, the output sentence is adjusted to second person; (3) finally, if the sentence has an element that refers to the third person with no relation with the first and second persons, the sentence remains in third person.

aLexiE contains the number and gender inflections of all lemmas and the person features of verbs and pronouns. Once these features are decided, they are applied to all word inputs. However, in case the subject is missing, default features must be set (as previously mentioned).

The verbal tense of the output sentence is present unless a time adverb or a time adverbial locution (e.g. last week) are provided. For example, for the adverb ‘yesterday’, the tense of the output sentence is past. This linguistic information can also be found in aLexiE.

The library handles contraction spellings as well. We implemented those from a freely accessible list.¹²

In case a word is missing in the lexicon, no related features are available or they cannot be inflected, our library treats the word as a proper noun.

¹¹Some gender-specific words, such as waiter and waitress, remain in modern English.

¹²Available at https://en.wikipedia.org/wiki/Wikipedia:List_of_English_contractions, July 2019.

When generating a sentence, it is necessary to create its syntagms and join them while respecting their syntactic and semantic function. For example, to create ‘She looks at the picture’ from the input words ‘she’, ‘look’ and ‘picture’, it is necessary to generate the nominal syntagm the ‘picture’ and integrate it into a prepositional syntagm as ‘at the picture’. It is also necessary to build the subject of the sentence ‘she’ and a predicate with ‘look’ as the main verb. Finally, it is necessary to integrate in the output sentence the subject and the predicate with the prepositional complement (‘she looks at the picture’). All these stages are automatic, even the inclusion of the preposition, from the syntactic and semantic information in aLexiE.

Table 3 shows examples of automatically generated sentences with increasing linguistic complexity. Alongside each example we indicate the input words.

IV. EXPERIMENTAL RESULTS

First, we compared the aLexiE lexicon with other lexica from the state-of-the-art (Section IV-A). Then, we evaluated automatic English text expansion with our system (Section IV-B), both directly (by regenerating corpus sentences) and manually (from annotations) using a widely used corpus in the NLG field (Section IV-B.1). To the best of our knowledge, there is no other system for automatic text expansion. Therefore, we have created the English version of a corpus (Section IV-B.2), to compare automatic text expansion in a multilingual scenario.

A. LEXICON

There are several resources for statistical natural language processing and corpus-based computational linguistics.¹³ The main differences between aLexiE and those freely available online resources are coverage, correctness and completeness of linguistic information (morphology, syntax, and semantics).

Table 4 shows the information we combined from the selected resources to create aLexiE. According to [64], EnLex contains 508,000 unique lemmas, corresponding to 695,000 unique forms. We only extracted some entries, yielding 212,021 unique lemmas that correspond to 41.74% of the extracted entries. NIH contains 505,145 unique lemmas and 955,564 inflected forms. In this case we only extracted 67,660 unique lemmas that correspond to 13.39% of

¹³E.g. <https://nlp.stanford.edu/links/statnlp.html>, July 2019.

TABLE 4. aLexiE lemma and form extraction statistics.

	Initial		Extracted	
	Lemmas	Forms	Lemmas	%Lemmas
EnLex	508,000	695,000	212,021	41.74%
NIH	505,145	955,564	67,660	13.39%
English-Freeling	37,000	68,000	14,368	38.83%

TABLE 5. Semantic sources for entries tagged as nouns in aLexiE.

	MCR		FrameNet		Total	
	Present	Absent	Present	Absent	Present	Absent
Nouns	69,784	8,182	25	8,157	69,809	8,157

TABLE 6. Merged and verified lemmas in aLexiE by lexical category.

Category	Merged	Verified
Noun	81,380	77,966
Pronoun	86	86
Verb	11,602	10,903
Adjective	47,078	28,971
Adverb	10,353	1,867
Conjunction	99	70
Determiner	98	98
Preposition	140	140
Total	150,836	120,101

TABLE 7. aLexiE lemmas and forms by lexical category.

Category	Lemmas	Forms
Noun	77,966	141,580
Pronoun	86	92
Verb	10,903	39,493
Adjective	28,971	28,971
Adverb	1,867	1,867
Conjunction	70	70
Determiner	98	100
Preposition	140	140
Total	120,101	212,313

the lexicon. Specifically, we only extracted adjective, adverb, conjunction, determiner, preposition, and pronoun entries, since other entries had no associated morphological information. Freeling for English contains 37,000 unique lemmas, corresponding to 68,000 unique forms. We only extracted adjectives, adverbs, and verbs (for the same reason as for NIH), producing 14,368 unique lemmas corresponding to 38.83% of the original set.

Table 5 shows the sources for the semantic classification of the nouns in aLexiE. We only searched in FrameNet¹⁴ the lemmas of the nouns that were missing in MCR. Table 6 shows the amount of lemmas after merging and verification. Table 7 shows the lexical categories of the lemmas and forms in aLexiE. Most were tagged as nouns (77,966), yielding over 141,000 aLexiE inflected forms. Determiners and pronouns were revised manually to include plural forms for the lemmas ‘this’ and ‘that’.

¹⁴Available at <https://framenet.icsi.berkeley.edu/frndrupal>, July 2019.

B. ENGLISH TEXT EXPANSION

As previously said, we evaluated our system by extracting keywords from sentences from a widely used corpus in the NLG field (Section IV-B.1) and an ad-hoc Spanish-English parallel corpus (Section IV-B.2). We evaluated output quality in terms of completeness, correctness and similarity to the original sentence.

We decided to discard some common state-of-the-art metrics such as ROUGE [65] and BLEU [66], because they weakly reflect human assessment of NLG, as discussed in [67].

1) AAC CORPUS

Even though our approach may be used for general NLG scenarios, we chose an AAC corpus¹⁵ for our first evaluation due to the interest of AAC as a representative real application. Some AAC tools such as Talk Together¹⁶ and LetMeTalk¹⁷ have small vocabulary packages with rigid interactions. None of them generates messages taking morphological, syntactic, and semantic information into consideration. The interest of NLG for AAC is illustrated by several previous works [68]–[70]. First we selected sentences without commas or hyphens, to ensure that there was a single sentence/idea in a clause (our system could handle multiple ideas as separate sentences). Since our system performs NLG automatically, we selected sentences in present, past, and future tense because these can be inferred by time adverbial complements. We then filtered the result to obtain the main words (adjectives, adverbs, nouns, pronouns, proper nouns, and verbs). Next we lemmatised all those words but the nouns and pronouns. This was because if we lemmatised the latter, the system would have no clue to generate a sentence with a plural noun or pronoun since the features of these particular words are independent from other components of the sentence (conversely, adjectives depend on the noun they modify). For this purpose we used the Spacy syntactic parser. The resulting dataset had 1,869 English sentences and their main words.

a: ANNOTATION

We introduced the main words of a target sentence into our automatic NLG system and we studied the output sentences. In case of a full match between the target and generated sentences, automatic generation was considered totally successful. This happened for 1,315 sentences, 70.25% of the total. The remaining 554 were manually inspected.

Of these, 15 differed only in few capital letters. This was due to errors in the target and missing words in aLexiE that were treated as proper nouns. Our system correctly replaced words by proper nouns in eight sentences, such as ‘I need a new harry potter book’, which was generated as ‘I need a new Harry Potter book’. Even though the matches were inexact, we consider these sentences success cases rather

¹⁵Available at <http://aactext.org>, July 2019.

¹⁶Available at <https://acecentre.org.uk/resources/talk-together>, July 2019.

¹⁷Available at <http://www.utac.cat/descarregues/cace-utac>, July 2019.

than failures. In five sentences the system failed to detect the lexical category of some input words due to missing data in aLexiE, as in ‘I’m itchy’, which our system generated as ‘I’m Itchy’ (words like itchy are neither present in aLexiE nor in MWD). We did not consider these sentences failures because they were due to missing words in the dictionary. Finally, there were two sentences containing words without an aLexiE entry that indeed existed in MWD: ‘Need a bigger size’ was generated by our system as ‘Need a Bigger size’, and ‘It is 2 o’clock’ was generated as ‘It is 2 O’clock’. These were indeed failures of the system.

There were 22 sentences containing spelling mistakes in the target such as ‘I have an appointment with the doctor today’ (‘appointment’ instead of ‘apointment’).

Consequently, our system was able to generate 1350¹⁸ correct sentences automatically, corresponding to 72.23% of the total.

Finally, it was only necessary to evaluate 517 sentences manually. They were revised by five NLG researchers from *atlanTTic*, University of Vigo, with English skills equivalent to C1 in the Common European Framework of Reference for Languages (CEFR) or 95 or above in the Test of English as a Foreign Language (TOEFL). Table 8 shows the annotation options.

TABLE 8. Annotation tags.

Feature	Values
Error type	morphological (a), syntactic (b), lexicon (c), grammar (d), target (e), lemmatiser (f)
Evaluation	0-5
Best generation	optional
Generation suggestion	optional

Manual evaluation considered six error possibilities: morphological error (a), syntactic error (b), aLexiE error (c), grammar error (d), target error (e), and lemmatiser error (f) (Table 8). The annotators also rated the quality of the generation from 0 (not generated) to 5 (full match between target and output).¹⁹ Moreover, when the system presented different alternative outputs the annotator had to choose one. We noticed that, in some errors, except for SVO order the system would have succeeded in generating the targets. The annotators were requested to provide output suggestions in that situation.

The annotation task took two months. We handed instructions and examples to the annotators in advance to guarantee the consistency of the resulting corpus. The tests exploited various features of English grammar such as sentence type and constructions with different word categories. The annotation script returned an XML file. Listing 4 shows an annotated sentence example.

The final results can be summarised as follows. Firstly, we must distinguish the cases when our NLG system generated

¹⁸1315 + 8 + 5 + 22 = 1350

¹⁹0 and 5 ratings were automatically treated.

```
<?xml version="1.0" encoding="UTF-8"?>
<TAGGING>
<CLAUSE>
  <TARGET>Dropped my change.</TARGET>
  <Generated_Clauses >
    <Clause >
      Drop my change .
      <Error>b</Error >
      <Rating >1</Rating >
    </Clause >
    <Clause >
      I drop my change .
      <Error>a</Error >
      <Rating >2</Rating >
    </Clause >
  </Generated_Clauses >
  <Best_realisation >2</Best_realisation >
  <Suggestion_for_Generation >
    I dropped my change yesterday .
  </Suggestion_for_Generation >
</CLAUSE>
</TAGGING>
```

LISTING 4. Annotation example.

TABLE 9. Distribution of annotations of our dataset.

Annotation result	Sentence number
No consensus about best realisation	0
No consensus about error type but consensus about best realisation	45
Total consensus	472

a single possibility from those with several output sentences. In the first case the error type was set by majority vote between the annotators. If the annotators did not agree, the sentence was tagged with *no consensus about error type*. The final rating of each output sentence was computed as the arithmetic average of annotator ratings. In the second case, first we checked if there was consensus in the *best realisation* field. Otherwise, the sentence was tagged with *no consensus about best realisation*. If the annotators provided suggestions of best realisations and there was a consensus about them, we tagged and rated the best output candidate as in the first case. Table 9 shows the distribution of the annotations.

When the annotators agreed about error and best realisation, their average rating indicates that the information in the target could be understood from the generated sentence. This also happened when the annotators agreed about the best realisation but there was no consensus about error type. The annotators suggested 367 different alternative outputs, of which our library generated 160 correctly and automatically (43.597%). The remaining 207 sentences (56.403%) were not manually inspected. We suppose that many of these generated sentences might be considered appropriate as well.

The main mistakes were due to verbal tense adjustment, since many sentences were in past tense but did not have any time-related complement. Another common failure was adding a different preposition instead of the one in the target (in some cases, however, this change did not modify the meaning of the output sentence).

TABLE 10. Reliability data matrix of the AAC annotated dataset considering the error types in Table 8.

	Sentences					
	1	2	...	125	...	523
Annotator 1	b	b	...	a	...	a
Annotator 2	b	b	...	d	...	a
Annotator 3	b	d	...	d	...	a
Annotator 4	b	b	...	b	...	a
Annotator 5	b	b	...	b	...	a

TABLE 11. Coincidence matrix of our annotated dataset considering the error types in Table 8.

	a	b	c	d	e	f
a	698	12.5	3	46.75	39.5	8.25
b	12.5	378	7.5	91	17.75	7.25
c	3	7.5	37	10.75	3.5	14.25
d	46.75	91	10.75	635.5	83.25	39.75
e	39.5	17.75	3.5	83.25	55.5	18.5
f	8.25	7.25	14.25	39.75	18.5	4

TABLE 12. Overall inter-annotator agreement.

Agreement measure	Value
<i>Alpha</i>	0.582
Accuracy	0.691

b: AGREEMENT METRICS

Manual evaluation was monitored with two recognized agreement metrics that yield robust estimations of the differences between annotators: *Alpha*-reliability [71], [72] and accuracy.

When the annotators perfectly agree, $Alpha = 1$. When their agreement seems by chance $Alpha = 0$. Obviously both extremes should be avoided.

Our evaluation focused on nominal data because we measured error annotation agreement between five observers. As previously said, we computed the agreement in error type and obtained the average rate. The first step was to build a 5-observers-by-523-sentences reliability data matrix containing 5×523 values.

Table 10 shows that our system generated 523 sentences for the 517 target sentences in the corpus. This was because there were several generated candidates for some targets.

Next we tabulated the coincidence matrix in Table 11, in units. Coincidence matrices take into account the values in a reliability data matrix. They differ from contingency matrices in that the latter consider units in two dimensions, not values. Our coincidence matrix accounted for all pairable errors from the five annotators into a 6-by-6 square matrix, omitting references to annotators. This type of matrix is

TABLE 13. *Alpha* between annotator pairs.

	Annotator 1	Annotator 2	Annotator 3	Annotator 4	Annotator 5
Annotator 1	-	0.726	0.485	0.675	0.649
Annotator 2	0.726	-	0.451	0.614	0.646
Annotator 3	0.485	0.451	-	0.508	0.447
Annotator 4	0.675	0.614	0.508	-	0.624
Annotator 5	0.649	0.646	0.447	0.624	-

symmetric with respect to its main diagonal, which holds all perfect matches. Note that the coincidences were counted twice in the coincidence matrix. Disagreements (represented by off-diagonal cells) were also counted twice, yet in different cells.

We then estimated inter-agreement accuracy between pairs of annotators. This simply averages the proportions given by the diagonal of the coincidence matrix. Note that it neither accounts for fortuitous (dis)agreement nor for value ordering. The results for *Alpha* and accuracy in Table 12 are promising. Tables 13 and 14 represent inter-agreement between pairs of annotators [73]–[75].

2) SPANISH-ENGLISH PARALLEL CORPUS

We are not aware of the existence of other systems for automatic text expansion as we have defined it. Therefore, we decided to apply our automatic NLG system to Spanish and English and compare the results.

For this purpose, we manually created the English version of the Spanish corpus used in [4].²⁰

The final parallel corpus is composed of 948 English/Spanish sentences covering various grammar features such as different sentence types and constructions with different word categories.

Table 15 shows a comparison between the English and Spanish versions in terms of automatic generation using the parallel corpus. First, our system generated 613 English sentences automatically. The remaining 123 sentences were inspected manually. We noticed that the most relevant mistakes (in 106 sentences) were due to Wikipedia training, since the system failed to add a certain preposition in the target. Two of the other 17 sentences were actually correct, since they only differed in some capital letters, and four had differences in determiners that did not affect their meaning.

Second, our system was able to automatically generate 72 English sentences out of the 212 that the system in [4] failed to generate in Spanish. The remaining 140 sentences were manually inspected. The most common mistakes were related to verbal tense adjustment (64 sentences) and wrong prepositions (34 sentences).

Next we compared the approach in this paper with the system in [4]. We correctly generated 77.64% and 72.26% of the Spanish and English sentences in the parallel corpus, respectively.

²⁰Available at https://www.gti.uvigo.es/images/manual_evaluation_EN.txt, July 2019.

TABLE 14. Accuracy between annotator pairs.

	Annotator 1	Annotator 2	Annotator 3	Annotator 4	Annotator 5
Annotator 1	-	0.803	0.608	0.759	0.751
Annotator 2	0.803	-	0.585	0.717	0.753
Annotator 3	0.608	0.585	-	0.621	0.587
Annotator 4	0.759	0.717	0.621	-	0.728
Annotator 5	0.751	0.753	0.587	0.728	-

TABLE 15. Comparison between the Spanish (ES) and English (EN) systems using the parallel corpus (automatic evaluation).

		ES version		
		Automatic	Manual	Total
EN version	Automatic	613	72	685
	Manual	123	140	263
	Total	736	212	948

The most important difference between the two languages was the use of prepositions. In Spanish there were few mistakes of this kind, but they were the most common in English. In our opinion this was due to the difficulty to detect phrasal verbs.

V. CONCLUSION

We have developed an automatic hybrid system for English text expansion. Relying on the aLexiE lexicon and our grammar, our system is able to perform fully automatic text expansion from few input words. The integration of new lexical resources for any language is simple. The architecture separates domain-independent from domain-dependent components, so that the latter can be substituted. We remark that the aLexiE lexicon and the grammar we have developed for English expansion are relevant results in themselves. They could be useful to other NLG researchers.

As far as we know this is the first fully automatic hybrid system for English text expansion, combining a knowledge base of vocabulary and grammar realisations with a statistical language model for preposition inference. Our system has a good success rate when generating coherent and grammatically correct sentences from user-selected input words.

The surface realiser relies on aLexiE and our grammar to take its decisions. For this reason, we provide a detailed description of the procedure to create them. As future work we plan to predict the best grammar realisation for input words without SVO order.

We have offered an insightful analysis of semantic similarities and differences between target and output texts, and we have assessed system performance using state-of-the-art metrics. First, we evaluated the system by regenerating texts from an AAC corpus. The system succeeded 72.23% of the time. As future work we will evaluate the system with Aac users taking advantage of the broad community of our Pictodroid suite.²¹ Second, we conducted automatic text expansion in English and Spanish and compared the results. Due to the lack

²¹Available at <http://www.acegal.org/en/pictodroid>, July 2019.

of a multilingual dataset, we had to create a parallel corpus for this purpose. Even though generation was slightly better for Spanish, thanks to its more predictable preposition usage, performances were comparable.

As another future research line we plan to create similar systems for new languages such as French and Portuguese.

REFERENCES

- [1] E. Reiter and R. Dale, *Building Natural Language Generation Systems*. New York, NY, USA: Cambridge Univ. Press, 2000.
- [2] Y. Elshahar, S. Hu, K. Bouazza-Marouf, D. Kerr, and A. Mansor, "Augmentative and alternative communication (AAC) advances: A review of configurations for individuals with a speech disability," *Sensors*, vol. 19, no. 8, p. 1911, 2019.
- [3] S. García-Méndez, M. Fernández-Gavilanes, E. Costa-Montenegro, and J. Juncal-Martínez, F. J. González-Castaño, "Automatic natural language generation applied to alternative and augmentative communication for online video content services using SimpleNLG for spanish," in *Proc. 15th Internet Accessible Things*, 2018, Art. no. 19.
- [4] S. García-Méndez, M. Fernández-Gavilanes, E. Costa-Montenegro, J. Juncal-Martínez, and F. J. González-Castaño, "A library for automatic natural language generation of Spanish texts," *Expert Syst. Appl.*, vol. 120, pp. 372–386, Apr. 2019.
- [5] E. Reiter and R. Dale, "Building applied natural language generation systems," *Natural Lang. Eng.*, vol. 3, no. 1, pp. 57–87, Mar. 1997. doi: 10.1017/S1351324997001502.
- [6] R. Kutlak, C. Mellish, and K. van Deemter, "Content selection challenge—University of Aberdeen entry," in *Proc. 14th Eur. Workshop Natural Lang. Gener.*, Aug. 2013, pp. 208–209. [Online]. Available: <http://www.aclweb.org/anthology/W13-2133>
- [7] F. Mairesse and M. Walker, "PERSONAGE: Personality generation for dialogue," in *Proc. 45th Annu. Meeting Assoc. Comput. Linguistics*, 2007, pp. 496–503.
- [8] O. Dušek and F. Jurčiček, "Training a natural language generator from unaligned data," in *Proc. 53rd Annu. Meeting Assoc. Comput. Linguistics, 7th Int. Joint Conf. Natural Lang. Process.*, vol. 1, 2015, pp. 451–461.
- [9] J. A. Bateman, "Enabling technology for multilingual natural language generation: The KPML development environment," *Natural Lang. Eng.*, vol. 3, no. 1, pp. 15–55, Mar. 1997. doi: 10.1017/S1351324997001514.
- [10] S. Williams and E. Reiter, "Generating basic skills reports for low-skilled readers," *Natural Lang. Eng.*, vol. 14, no. 4, pp. 495–525, 2008.
- [11] L. Wanner, B. Bohnet, N. Bouayad-Agha, F. Lareau, and D. Nicklaß, "MARQUIS: Generation of user-tailored multilingual air quality bulletins," *Appl. Artif. Intell.*, vol. 24, no. 10, pp. 914–952, 2010.
- [12] G. Lampouras and A. Vlachos, "Imitation learning for language generation from unaligned data," in *Proc. 26th Int. Conf. Comput. Linguistics (COLING)*, 2016, pp. 1101–1112.
- [13] H. Mei, M. Bansal, and M. R. Walter, "What to talk about and how? Selective generation using LSTMs with coarse-to-fine alignment," in *Proc. NAACL-HLT*, 2016, pp. 720–730.
- [14] R. Barzilay and M. Lapata, "Modeling local coherence: An entity-based approach," *Comput. Linguistics*, vol. 34, no. 1, pp. 1–34, 2008.
- [15] M. A. Walker, A. Stent, F. Mairesse, and R. Prasad, "Individual and domain adaptation in sentence planning for dialogue," *J. Artif. Intell. Res.*, vol. 30, pp. 413–456, Nov. 2007.
- [16] R. Hervás and P. Gervás, "Case retrieval nets for heuristic lexicalization in natural language generation," in *Proc. Portuguese Conf. Artif. Intell.* Berlin, Germany: Springer, 2005, pp. 55–66.
- [17] R. Dale, "Cooking up referring expressions," in *Proc. 27th Annu. Meeting Assoc. Comput. Linguistics*, 1989, pp. 68–75.

- [18] D. Robert, *Generating Referring Expressions: Constructing Descriptions in a Domain of Objects and Processes* (ACL-MIT Series in Natural Language Processing). Cambridge, MA, USA: MIT Press, 1992.
- [19] M. C. Frank, N. D. Goodman, and J. B. Tenenbaum, "Using speakers' referential intentions to model early cross-situational word learning," *Psychol. Sci.*, vol. 20, no. 5, pp. 578–585, 2009.
- [20] R. Dale and E. Reiter, "Computational interpretations of the Gricean maxims in the generation of referring expressions," *Cognit. Sci.*, vol. 19, no. 2, pp. 233–263, Apr. 1995.
- [21] M. W. Meteer, "Bridging the generation gap between text planning and linguistic realization," *Comput. Intell.*, vol. 7, no. 4, pp. 296–304, 1991.
- [22] I. Langkilde-Geary and K. Knight, "Halogen statistical sentence generator," in *Proc. Assoc. Comput. Linguistics*, Philadelphia, PA, USA, 2002, pp. 102–103.
- [23] H. Nakanishi, Y. Miyao, and J. Tsujii, "Probabilistic models for disambiguation of an HPSG-based chart generator," in *Proc. 9th Int. Workshop Parsing Technol.*, 2005, pp. 93–102.
- [24] A. Cahill and J. Van Genabith, "Robust PCFG-based generation using automatically acquired LFG approximations," in *Proc. 21st Int. Conf. Comput. Linguistics, 44th Annu. Meeting Assoc. Comput. Linguistics*, 2006, pp. 1033–1040.
- [25] C. Gardent and S. Narayan, "Multiple adjunction in feature-based tree-adjointing grammar," *Comput. Linguistics*, vol. 41, no. 1, pp. 41–70, 2015.
- [26] T.-H. Wen, M. Gasic, N. Mrksic, P.-H. Su, D. Vandyke, and S. Young, "Semantically conditioned LSTM-based natural language generation for spoken dialogue systems," 2015, *arXiv:1508.01745*. [Online]. Available: <https://arxiv.org/abs/1508.01745>
- [27] A. Cheyer and D. Guzzoni, "Method and apparatus for building an intelligent automated assistant," U.S. Patent 11 518 292, May 3, 2007. [Online]. Available: <https://www.google.com/patents/US20070100790>
- [28] D. Mirkovic and L. Cavedon, "Dialogue management using scripts," EP Patent 2006 0 759 358, Feb. 27, 2008. [Online]. Available: <https://encrypted.google.com/patents/EP1891625A1?cl=und>
- [29] E. Reiter, C. Mellish, and J. Levine, "Automatic generation of technical documentation," *Appl. Artif. Intell.*, vol. 9, no. 3, pp. 259–287, 1995.
- [30] R. Dale, S. Geldof, and J.-P. Prost, "Using natural language generation in automatic route description," *J. Res. Pract. Inf. Technol.*, vol. 37, no. 1, p. 89, 2005.
- [31] W. P. Liu, B. Georgescu, S. K. Zhou, and D. Comaniciu, "Automatic generation of radiology reports from images and automatic rule out of images without findings," U.S. Patent 15 158 375, Nov. 23, 2017.
- [32] I. Mani and M. Maybury, *Automatic Summarization*. Amsterdam, The Netherlands: John Benjamins Publishing Company, 2001.
- [33] E. Reiter, R. Robertson, and L. M. Osman, "Lessons from a failure: Generating tailored smoking cessation letters," *Artif. Intell.*, vol. 144, pp. 41–58, Mar. 2003.
- [34] A. Fiedler, *Natural Language Proof Explanation* (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), vol. 2605 LNAI. 2005, pp. 342–363. doi: 10.1007/978-3-540-32254-2_20.
- [35] T. W. Morris, "Conversational agents for game-like virtual environments," in *Proc. AAAI Spring Symp. Artif. Intell. Interact. Entertainment*, 2002, pp. 82–86.
- [36] F. Peinado, P. Gervás, and B. Díaz-Agudo, "A description logic ontology for fairy tale generation," in *Proc. 4th Int. Conf. Lang. Resour. Eval., Proc. Workshop Lang. Resour. Linguistic Creativity*, 2004, pp. 56–61.
- [37] P. Gervás, "An expert system for the composition of formal spanish poetry," in *Applications and Innovations in Intelligent Systems*. London, U.K.: Springer, 2001, pp. 19–32. doi: 10.1007/978-1-4471-0275-5_2.
- [38] A. Stent, M. Marge, and M. Singhai, "Evaluating evaluation methods for generation in the presence of variation," in *Computational Linguistics and Intelligent Text Processing*. Berlin, Germany: Springer, 2005, pp. 341–351. doi: 10.1007/978-3-540-30586-6_38.
- [39] A. Gatt and E. Reiter, "SimpleNLG: A realisation engine for practical applications," in *Proc. 12th Eur. Workshop Natural Lang. Gener.*, 2009, pp. 90–93. [Online]. Available: <http://dl.acm.org/citation.cfm?id=1610195.1610208>
- [40] M. Bollmann, "Adapting SimpleNLG to german," in *Proc. 13th Eur. Workshop Natural Lang. Gener.*, 2011, pp. 133–138.
- [41] P.-L. Vaudry and G. Lapalme, "Adapting SimpleNLG for bilingual English-French realisation," in *Proc. 14th Eur. Workshop Natural Lang. Gener.*, Aug. 2013, pp. 1–8.
- [42] R. de Oliveira and S. Sripada, "Adapting SimpleNLG for brazilian portuguese realisation," in *Proc. 8th Int. Natural Lang. Gener. Conf. (INLG)*, 2014, pp. 93–94.
- [43] A. Mazzei, C. Battaglini, and C. Bosco, "SimpleNLG-IT: Adapting SimpleNLG to Italian," in *Proc. 9th Int. Natural Lang. Gener. Conf.*, 2016, pp. 1–9.
- [44] S. G. Méndez, "Natural language generation library for Spanish," M.S. thesis, Telecommun. Eng. School, Univ. Vigo, Vigo, Spain, 2017.
- [45] A. Ramos-Soto, J. Janeiro-Gallardo, and A. Bugarín Diz, "Adapting SimpleNLG to Spanish," in *Proc. 10th Int. Conf. Natural Lang. Gener.*, 2017, pp. 144–148.
- [46] R. De Jong and M. Theune, "Going dutch: Creating SimpleNLG-NL," in *Proc. 11th Int. Natural Lang. Gener. Conf.*, E. Krahmer, A. Gatt, and M. Goudbeek, Eds. Stroudsburg, PA, USA: Association for Computational Linguistics, 2018, pp. 73–78.
- [47] G. Chen, K. Van Deemter, and C. Lin, "SimpleNLG-ZH: A linguistic realisation engine for Mandarin," in *Proc. 11th Int. Natural Lang. Gener. Conf.*, 2018, pp. 57–66.
- [48] A. Cascallar-Fuentes, A. Ramos-Soto, and A. B. Diz, "Adapting SimpleNLG to Galician language," in *Proc. 11th Int. Conf. Natural Lang. Gener.*, 2018, pp. 67–72.
- [49] I. Androutsopoulos, G. Lampouras, and D. Galanis, "Generating natural language descriptions from OWL ontologies: The NaturalOWL system," 2014, *arXiv:1405.6164*. [Online]. Available: <https://arxiv.org/abs/1405.6164>
- [50] E. Reiter, S. Sripada, J. Hunter, J. Yu, and I. Davy, "Choosing words in computer-generated weather forecasts," *Artif. Intell.*, vol. 167, pp. 137–169, Sep. 2005.
- [51] A. Belz, "Automatic generation of weather forecast texts using comprehensive probabilistic generation-space models," *Natural Lang. Eng.*, vol. 14, no. 4, pp. 431–455, Oct. 2008. doi: 10.1017/S1351324907004664.
- [52] J. Chen, S. Bangalore, O. Rambow, and M. A. Walker, "Towards automatic generation of natural language generation systems," in *Proc. 19th Int. Conf. Comput. Linguistics*, vol. 1, 2002, pp. 1–7.
- [53] P. Demasco and K. F. McCoy, "Generating text from compressed input: An intelligent interface for people with severe motor impairments," *Commun. ACM*, vol. 35, no. 5, pp. 68–78, May 1992. [Online]. Available: <http://doi.acm.org/10.1145/129875.129881>
- [54] P. Demasco, K. McCoy, Y. Gong, C. Pennington, and C. Rowe, "Towards more intelligent AAC interfaces: The use of natural language processing," in *Proc. 12th Annu. Conf.*, 1989, pp. 141–142.
- [55] K. McCoy, P. Demasco, M. Jones, C. Pennington, and C. Rowe, "Applying natural language processing techniques to augmentative communication systems," in *Proc. 13th Conf. Comput. Linguistics*, vol. 3, 1990, pp. 413–415.
- [56] D. Crouch and T. H. King, "Unifying lexical resources," in *Proc. Interdiscipl. Workshop Identificat. Represent. Verb Features Verb Classes*, 2005, pp. 32–37.
- [57] S. Neculescu, N. B. Rafecas, M. Padró, M. Marimon, and E. Revilla, "Towards the automatic merging of language resources," in *Proc. 1st Int. Workshop Lexical Resour. ESSLLI*, Ljubljana, Balkans, Aug. 2011, pp. 70–77.
- [58] N. Bel, M. Padró, and S. Neculescu, "A method towards the fully automatic merging of lexical resources," in *Proc. Workshop Lang. Resour. Technol. Services Sharing Paradigm*, Nov. 2011, pp. 8–15.
- [59] B. Sagot, "The Lefff, a freely available and large-coverage morphological and syntactic lexicon for French," in *Proc. 7th Int. Conf. LREC*, N. Calzolari, K. Choukri, B. Maegaard, J. Mariani, R. Odiijk, S. Piperidis, M. Rosner, and D. Tapias, Eds. Valletta, Malta: ELRA, 2010.
- [60] L. Danlos and B. Sagot, "Constructions pronominales dans Dicovalence et le lexique-grammaire-intégration dans le Lefff," in *Proc. 27th Lexicon-Grammar Conf.*, L'Aquila, Italy, 2008, pp. 1–8. [Online]. Available: <https://hal.inria.fr/inria-00524741>
- [61] A. González-Agirre, E. Laparra, and G. Rigau, "Multilingual central repository version 3.0: Upgrading a very large lexical knowledge base," in *Proc. 6th Global WordNet Conf.*, Matsue, Japan, 2012, pp. 1–8.
- [62] E. Maggiori, S. Schiaffino, and S. Godoy, "Desarrollo de una gramática para asercionessimples en español y su implementación en Prolog," Universidad Nacional del Centro de la Provincia de Buenos Aires, Tandil, Argentina, 2013, pp. 380–391. [Online]. Available: <https://docplayer.es/35491177-Desarrollo-de-una-gramatica-para-asercciones-simples-en-espanol-y-su-implementacion-en-prolog.html>
- [63] R. Tarjan, "Depth-first search and linear graph algorithms," in *Proc. 12th Annu. Symp. Switching Automata Theory*, Oct. 1971, pp. 114–121.
- [64] B. Sagot, "A multilingual collection of CoNLL-U-compatible morphological lexicons," in *Proc. 11th Int. Conf. Lang. Resour. Eval. (LREC)*, 2018, pp. 1–7.

- [65] C.-Y. Lin and E. Hovy, "Automatic evaluation of summaries using N-gram co-occurrence statistics," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics Hum. Lang. Technol.*, vol. 1, 2003, pp. 71–78.
- [66] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "BLEU: A method for automatic evaluation of machine translation," in *Proc. 40th Annu. Meeting Assoc. Comput. Linguistics*, 2002, pp. 311–318.
- [67] J. Novikova, O. Dušek, A. C. Curry, and V. Rieser, "Why we need new evaluation metrics for NLG," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2017, pp. 2241–2252.
- [68] A. Copestake, "Applying natural language processing techniques to speech prostheses," in *Proc. AAAI Fall Symp. Develop. Assistive Techn. People Disabilities*, 1996, pp. 1–7.
- [69] P. Moreda and E. Lloret, "The use of natural language processing techniques to support health literacy: An evidence-based review," *SYLWAN*, 2015. [Online]. Available: <http://hdl.handle.net/10045/58085>
- [70] C. Nikolopoulos, D. Kuester, M. Sheehan, S. Ramteke, A. Karmarkar, S. Thota, J. Kearney, C. Boirum, S. Bojedla, and A. Lee, "Robotic agents used to help teach social skills to children with Autism: The third generation," in *Proc. RO-MAN*, 2011, pp. 253–258.
- [71] K. Krippendorff. (2012). *Content Analysis: An Introduction to Its Methodology*. [Online]. Available: https://books.google.es/books?hl=en&lr=&id=nE1aDwAAQBAJ&oi=fnd&pg=PP1&dq=K.+Krippendorff,+Content+Analysis:+An+Introduction+to+Its+Methodology+2012&ots=yYdIWtKfB&sig=EOHq2P7YND4vxp7AJTAdQpsJrRQ&redir_esc=y#v=onepage&q=K.+Krippendorff%2C+Content+Analysis%3A+An+Introduction+to+Its+Methodology+2012&f=false and https://books.google.nl/books?id=q657o3M3C8cC&dq=Krippendorff,+Klaus.+2004.+DCcontent+analysis:+an+introduction+to+its+methodology.&hl=en&sa=X&ei=C7sZVZm7KlJYO42VgOgJ&redir_esc=y
- [72] K. Klaus, "Computing Krippendorff's alpha-reliability," Tech. Rep., 2011.
- [73] H. Dorussen, H. Lenz, and S. Blavoukos, "Assessing the reliability and validity of expert interviews," *Eur. Union Politics*, vol. 6, no. 3, pp. 315–337, 2005.
- [74] M. Poesio and R. Artstein, "The reliability of anaphoric annotation, reconsidered: Taking ambiguity into account," in *Proc. Workshop Frontiers Corpus Annotations II, Pie Sky Assoc. Comput. Linguistics*, 2005, pp. 76–83.
- [75] J. P. Pestian, P. Matykievicz, M. Linn-Gust, B. South, O. Uzuner, J. Wiebe, K. B. Cohen, J. Hurdle, and C. Brew, "Sentiment analysis of suicide notes: A shared task," *Biomed. Inform. Insights*, vol. 5, p. BII-S9042, Jan. 2012.



ENRIQUE COSTA-MONTENEGRO received the Ph.D. degree from the University of Vigo, Spain, in 2007.

He is currently an Associate Professor with the Department of Telematics Engineering, University of Vigo. His research interests include mobile services, natural language processing, wireless networks, multiagent systems, and recommendation technologies.



JONATHAN JUNCAL-MARTÍNEZ received the B.S. degree in technical telecommunications engineering, and the M.S. and Ph.D. degrees in telematics engineering from the University of Vigo, Spain, in 2011, 2013, and 2017, respectively.

Since 2011, he has been a Researcher with the Information Technologies Group, University of Vigo. His research interests include mobile communication solutions for people with disabilities, natural language processing, data analytics, and machine learning.

Dr. Juncal-Martínez received the Second Prize in the second edition of the Final Project Awards, Madrid, in 2012, the Best ICT Project with social benefits in the 17th edition of the Galician Night of Telecommunications, Pontevedra, in 2012, and the First prize in modality A in the 12th edition of the Technological Projects Competition, Ourense, in 2012.



FRANCISCO JAVIER GONZÁLEZ-CASTAÑO received the B.S. degree from the University of Santiago de Compostela, Spain, in 1990, and the Ph.D. degree from the University of Vigo, Spain, in 1998.

He is currently a Professor with the University of Vigo, where he leads the Information Technologies Group. He has authored over 100 papers in several international journals, in the fields of telecommunications and computer science, and has participated in several relevant national and international projects. He holds three U.S. patents.



EHUD REITER received the A.B. degree in mathematics and the Ph.D. degree in computer science from Harvard University, in 1982 and 1990, respectively.

He is currently a Professor of computing science with the University of Aberdeen, and also a Chief Scientist with Arria NLG. He specializes in natural language generation, and has a Google Scholar h-index of 40. He also holds eight patents.

Prof. Reiter is currently the Chair of ACL SIGGEN, the special interest group in (natural language) generation of the Association for Computational Linguistics.



SILVIA GARCÍA-MÉNDEZ received the B.S. degree in telecommunication technologies engineering and the M.S. degree in telecommunication engineering from the University of Vigo, Spain, in 2014 and 2016, respectively, where she is currently pursuing the Ph.D. degree in information and communications technology. Since 2015, she has been a Researcher with the Information Technologies Group, University of Vigo. Her research interest includes the development of automatic solutions for natural language generation for Spanish and English.

Ms. García-Méndez received the Connecting for Good Vodafone Award, in 2017.



MILAGROS FERNÁNDEZ-GAVILANES received the Ph.D. degree in computer science from the University of A Coruña, Spain, in 2012.

She is currently an Assistant Professor with the Defense University Center, University of Vigo, Spain. Her research interests include natural language processing, natural language generation, machine learning, information retrieval, multiagent systems, and recommendation technologies.