Generating Irish Text with a Flexible Plug-and-Play Architecture

Simon Mille
ADAPT, Dublin City University
simon.mille@adaptcentre.ie

Elaine Uí Dhonnchadha
Trinity College, Dublin
uidhonne@tcd.ie

Lauren Cassidy
ADAPT, Dublin City University
lauren.cassidy@adaptcentre.ie

Brian Davis
ADAPT, Dublin City University
brian.davis@adaptcentre.ie

Stamatia Dasiopoulou
Independent Researcher
stamatia.dasiopoulou@gmail.com

Anya Belz
ADAPT, Dublin City University
anya.belz@adaptcentre.ie

Abstract

In this paper, we describe M-FleNS, a multilingual flexible plug-and-play architecture designed to accommodate neural and symbolic modules, and initially instantiated with rule-based modules. We focus on using M-FleNS for the specific purpose of building new resources for Irish, a language currently underrepresented in the NLP landscape. We present the general M-FleNS framework and how we use it to build an Irish Natural Language Generation system for verbalising part of the DBpedia ontology and building a multilayered dataset with rich linguistic annotations. Via automatic and human assessments of the output texts we show that with very limited resources we can create a system that reaches high levels of fluency and semantic accuracy, while having very low energy and memory requirements.

1 Introduction

Natural Language Generation (NLG) for tasks including dialogue-turn generation and fact verbalisation is increasingly widely used in commercial systems. Despite recent spectacular advances achieved by LLMs, in application contexts where accuracy and reliability are crucial, many commercial systems continue to use the same old template filler systems that have been around at least since the 1980s.[1] The other two main categories of NLG systems are neural language-model based (NLMB) systems, currently extremely popular in research systems, and rule and grammar based (RGB) systems, currently very unpopular. In contrast to template-based (TB) systems, NLMB systems have very high Coverage, while also sharing TB systems' high Fluency and Robustness. However, the disadvantage of an NLMB system is that it cannot be guaranteed that the output will be free of grammatical errors or even that it will be semantically accurate. The latter is of particular concern as NLMB systems cannot be trusted not to omit essential content, make things up, or even insult users. Moreover, such systems also tend not to be built for low-resource languages (LRLs) languages because of the large amounts of data needed to build them. Finally, NLMB systems often suffer from low Variation, and very low Energy Efficiency, with the best current models having shockingly high carbon footprints. RGB systems on the other hand have become increasingly unpopular since the NLP field switched first to statistical systems, then to neural systems. While RGB systems tend to have low Coverage, suprasentential Fluency, and Robustness as well as having to be built manually, they can be guaranteed to have high Accuracy and Grammaticality, as well as being efficient in terms of data and energy requirements, and suitable for LRLs.

According to the European Language Equality report for Irish (Lynn, 2022), Irish is a low-resource language. In a survey of available resources for European languages, on a scale of 1-4, Irish was classified as 4 having "weak or no support", and ranked 31st out of the 33 European languages surveyed. The report identifies a range of language technology gaps, mainly due to the lack of underlying data resources, dedicated funding and skill-sets, and finds that to date there has been little or no system development for Automatic Subtitling, Information Retrieval, Information Extraction, Natu-

[1] E.g. Arria NLG: https://www.arria.com/.

Proceedings of the 2nd Workshop in Pattern-based Approaches to NLP in the Age of Deep Learning , pages 25–42
December 6, 2023.

Reiter&Dale Tasks	M-FleNS Tasks	M-FleNS Input	M-FleNS Output	Output type
Content determination	—	—	—	—
Discourse planning	Linguistic structuring	Structured data	PredArg	DAG
Sentence aggregation	Text planning*	PredArg	PredArg-Agg	DAG
Lexicalisation	Lexicalisation Comm. structuring Deep sent. structuring Surf. sent. structuring Synt. aggregation*	PredArg(-Agg) PredArg-Lex PredArg-Th DSynt SSynt	PredArg-Lex PredArg-Th DSynt SSynt SSynt-Agg	DAG DAG DT DT DT
REG	REG*	SSynt(-Agg)	SSynt-Pro	DT
Linguistic realisation	Word ord. and agree. resolution Surface form retrieval	SSynt(-Agg/-Pro) DMorph	DMorph SMorph	Chain Chain

Table 1: The M-FleNS architecture (see Appendix D for illustration): the tasks, their respective input, output (used as module name), structure type (DAG = Directed Acyclic Graph; DT = Dependency Tree) and correspondence with Reiter and Dale (1997)'s tasks. * Denotes optional modules, i.e., grammatical texts can be produced without them.

ral Language Generation, Semantic Role Labelling, and other areas. The report recommends a long term strategy of support for dedicated LT education and training, investment in data collection and annotation, and the development of LT tools.

The *Digital Plan for the Irish Language* (Department of Tourism, Culture, Arts, Gaeltacht, Sport Media, 2022) notes that urgent action is needed if Irish is to benefit from the digital revolution and to survive the threat of digital extinction. It notes two complementary approaches, knowledge-based and data-driven machine-learning methods, and states that both are needed and each brings specific advantages. A linguistic knowledge base provides a digital, explicit account of the structure of contemporary Irish which is an important goal in itself, while machine-learning approaches can offer a quick and less labour-intensive route to developing certain technologies. Both approaches are needed and, especially in the context or LRLs, can be combined in specific systems.

In this paper, we present a flexible plug-and-play architecture that addresses both knowledge-based and machine-learning-based gaps in Irish Natural Language Processing, by releasing a generation system and a rich dataset. While the current (single) Multilingual Flexible Neuro-Symbolic (M-FleNS) system is multilingual –generating text also e.g. in English, French, Spanish, and Catalan-, we focus here on its instantiation with rule-based modules for the generation of Irish texts from DBpedia triple sets. Below, we start by describing and motivating our architecture (Section 2). Next we describe the WebNLG dataset, the FORGe generator and the Irish morphology tools (Section 3) we use. We present the extension to WebNLG data-to-text for Irish, and evaluate it via metrics and human assessment; we also present a new Irish dataset with rich linguistic annotations produced with our instantiated architecture (Section 4). We finish with a discussion of related work (Section 5). The generation pipeline,[2] dataset[3] and an interactive demo for the generation of short Wikipedia pages in Irish or English[4] are all publicly available.

2 A plug-and-play architecture for system and resource building

2.1 Modular structure

While end-to-end approaches are popular in current NLG systems (Dušek et al., 2018; Castro Ferreira et al., 2020), they are more data-hungry and computationally far more expensive (therefore more energy intensive) than corresponding modular architectures (Dušek et al., 2020). Furthermore, recent evidence shows that splitting the generation process into sub-steps can lead to better output texts (Castro Ferreira et al., 2019; Moryossef et al., 2019; Puduppully and Lapata, 2021; Kasner and Dusek, 2022). We seek to leverage this advantage by giving our M-FleNS framework a sequential architecture where each module corresponds to specific (sub)tasks of the natural language generation process roughly corresponding to the pipeline architecture originally established by Reiter and Dale (1997). Table 1 lists the M-FleNS modules in terms

[2] https://github.com/mille-s/DCU_TCD-FORGe_WebNLG23

[3] https://github.com/mille-s/Mod-D2T/

[4] https://github.com/mille-s/WikipediaPage_Generator

of the tasks they perform, alongside the tasks/-modules identified in Reiter and Dale's pipeline to which they roughly correspond.[5]

2.2 Rich linguistic representations

Each of the 10 different modules shown in Table 1 provides as output one or more well defined, rich, and linguistically motivated representations. The intermediate representations in M-FleNS are all graphs that can be grouped into three main types: (i) predicate-argument *directed acyclic graphs (DAGs)* for semantic information; (ii) *unordered dependency trees (DTs)* for syntactic information; and (iii) *chains* for morphological information. These intermediate representations loosely follow the different levels of Meaning-Text Theory (Mel'čuk, 1973). In the instantiated version of the pipeline presented in this paper, the input structured data is the WebNLG data (Aquilina et al., 2023), made of DBpedia triple sets, and we use the FORGe grammar-based generator to produce the intermediate representations (Mille et al., 2019) and the Irish NLP toolkit (Dhonnchadha et al., 2003) to produce the final representation: details about the dataset and tools are provided in Section 3.

2.3 Addressing technology gaps

With our modular approach we aim not only at developing a first system for Irish NLG, but also at producing new data that will allow for addressing more of the technology gaps identified in the European Language Equality report. For instance, with the generator we can produce a large amount of semantic and syntactic structures; syntactic structures paired with texts can be used to train syntactic parsers, while semantic structures paired with text can be used to train semantic role labelers. Using in parallel syntactic and semantic structures, tools can be trained that convert one into the other to build smaller modules to be combined with other tools (e.g. an existing syntactic parser). All ten intermediate representations can be also be used for explainability, language teaching, etc. In Section 4.6 we provide details on how we used our architecture to produce linguistically annotated data.

3 Data and tools

In the following subsections, we describe the dataset (WebNLG) and tools (FORGe and Irish NLP) we use in our experiments.

[5]This table is adapted from (Mille et al., 2023).

3.1 The WebNLG dataset

The WebNLG dataset (Aquilina et al., 2023) is a data-to-text benchmark consisting of {input, output} pairs, where the input is a set of n triples ($1 \leq n \leq 7$), the output a set of m texts that verbalise the triple set. In Figure 1, $n = 3$ and $m = 1$.

DBpedia triples are the building blocks of the inputs, and consist of three related elements called a *Property*, a *Subject* and an *Object* in Semantic Web terminology. A Subject (denoted by *DB-Subj* in this paper) is usually an entity that has a Property and a value for this Property, which is the Object (*DB-Obj*). E.g. in Figure 1, the entity *Agra_Airport* is associated with 3 properties: *location*, *operatingOrganisation* and *icaoLocationIdentifier*. The semantics of each property is defined by DBpedia editors,[6] but in most cases, *the Property of the DB-Subj is DB-Obj* makes it clear (e.g., *The location of Agra Airport is India*, *The operating organisation of Agra Airport the Indian Air Force*, and *the ICAO location identifier of Agra Airport is VIAG.*).

WebNLG 2017 (Gardent et al., 2017) consisted of (only) an English task. For WebNLG 2020, the English dataset was extended with more properties, and it also included Russian texts (Castro Ferreira et al., 2020); in both cases, the texts were collected via manual effort (crowdsourcing) . The third edition of the task in 2023 focused on four low-resource languages: Irish, Welsh, Breton and Maltese, for which the texts for the training data are the machine-translated 2020 English texts, while the texts in the test and development data were translated by professional translators. All inputs are the same as the 2020 inputs.

3.2 The FORGe multilingual generator

FORGe (Mille et al., 2019) is a multilingual rule-based generator that takes as input minimal predicate-argument (PredArg) structures. It realises the last four consecutive steps of the traditional NLG pipeline (Reiter and Dale, 1997) (sentence aggregation, lexicalisation,[7] referring expression generation and linguistic realisation, see Table 1). Each of the four steps is implemented as one or more graph transducer(s) that successively map the input PredArg onto different dependency-based intermediate linguistic representations.

[6]See http://mappings.dbpedia.org/index.php/How_to_edit_the_DBpedia_Ontology.

[7]We refer to a more surface-oriented lexicalisation here, with, e.g., function words, as opposed to the "deep" lexicalisation of the main concepts described in Section 4.1.

```
<entry category="Airport" eid="719" shape="(X (X) (X) (X))" shape_type="sibling" size="3">
  <modifiedtripleset>
    <mtriple>Agra_Airport | location | India</mtriple>
    <mtriple>Agra_Airport | operatingOrganisation | Indian_Air_Force</mtriple>
    <mtriple>Agra_Airport | icaoLocationIdentifier | &quot;VIAG&quot;</mtriple>
  </modifiedtripleset>
  <lex comment="" lang="ga" lid="Id3">Tá Aerfort Agra, atá VIAG mar an cód aitheantais
       áite ICAO aige, á reáchtáil ag Aerfhórsa na hIndia.</lex>
</entry>
```

Figure 1: A WebNLG data point (EN: 'Agra airport, whose ICAO identifier is VIAG, is operated by the IAF.')

A mix of language-independent and language-specific rules build these intermediate representations using additional knowledge contained in language-specific dictionaries. From the perspective of multilingualism, there are 3 types (T1-T3) of rules in FORGe: fully language-independent rules (T1, ∼82% of all rules); rules that apply to a subset of languages (T2, ∼6.5 languages on average, ∼3% of rules); and language-specific rules, which apply to one single language (T3, ∼15% of rules). In the description of the extensions of FORGe for Irish below, we refer to these three types.

FORGe uses three different dictionaries to store:

- Mappings between concepts and lexical units, e.g. *located {GA={lex=lonnaithe_JJ_01}}*.
- Lexical unit descriptions, e.g. *lonnaithe_JJ_01 {lemma = lonnaithe; pos = JJ; preposition_arg2 = i }*, where *i* 'in' is required on the second argument of *lonnaithe*: *lonnaithe i X* 'located in X'.
- Generic language-specific knowledge, such as the type of word order or morphological agreement triggered by surface-oriented dependencies (e.g. in English a direct object is by default after its governing verb in the sentence, and a determiner receives case, number and gender from its governing noun).

The input PredArg structures are very similar to the *Facts* in ILEX's Content potential structures (O'Donnell et al., 2001), or the *Message triples* in NaturalOWL (Androutsopoulos et al., 2013), with the difference that all predicates in the PredArg structures are generally intended to represent atomic meanings (e.g. *main* + *runway* as opposed to *mainRunway*), allowing for more flexible processing. The first part of the generation pipeline, which produces aggregated predicate-argument graphs, is also comparable to ILEX (O'Donnell et al., 2001), while the surface realisation is largely inspired by MARQUIS (Wanner et al., 2010). FORGe shares not only its general architecture with these two systems, but also the use of lexical resources with subcategorisation information and of a multilingual core of rules.

FORGe was adapted to the WebNLG'20 dataset for the generation of English texts and has a multilingual core of rules, but is not able to generate text in a new language off-the-shelf. However, adapting it to a new language is relatively easy, so it is a good candidate for building the first Irish generator. In Section 4, we report on the extensions we carried out to FORGe so as to be able to generate WebNLG Irish texts. We use the whole FORGe pipeline except for the surface form generation, for which we use the existing Irish NLP tools (see Section 3.3).

3.3 Irish NLP tools

The Irish NLP tools suite[8] includes finite-state transducers for Irish morphology generation (Dhonnchadha et al., 2003). These tools handle tokenisation and morphological analysis/generation of the inflected forms of Irish headwords coded in the finite-state lexicons. The tools were initially developed using xfst (Xerox finite state tools) (Beesley and Karttunen, 2003) and later converted to use foma tools (Hulden, 2009).[9] Finite-state transducers model a two-level morphology where a lexical description is mapped to a surface form, e.g. déan+Verb+VT+FutInd maps to the future tense form *déanfaidh* of the transitive verb *déan* 'make'. The transducers can be used to generate inflected forms of words for NLG and CALL applications, and the same transducers work in the opposite direction for morphological analysis as part of NLP applications including PoS tagging and parsing.

4 M-FleNS for Irish Natural Language Generation

In this section, we describe our pipeline for the generation of Irish texts from DBpedia triples,

[8] https://www.scss.tcd.ie/~uidhonne/irish.utf8.htm

[9] https://fomafst.github.io/

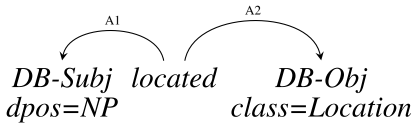

Figure 2: Sample PredArg template corresponding to the *location* property.

including Subject and Object label retrieval and predicate-argument template crafting (4.1), extensions to FORGe and lexical resource building for Irish (4.2), the connection between FORGe and Irish NLP tools (4.3) and post-processing of outputs (4.4). We then provide an evaluation of the generator (4.5) and describe a new dataset (4.6). All resources are available; see Footnotes 2, 3, 4.

4.1 PredArg templates and their instantiation

The linguistic structuring step consists of mapping the WebNLG input triple sets onto abstract linguistic (predicate-argument) structures. For this, we follow the approach of the FORGe submission at WebNLG'17 (Mille et al., 2019), i.e. we use PredArg templates in the PropBank style (Kingsbury and Palmer, 2002) that correspond to each individual property and instantiate them by replacing the DB-Subj and DB-Obj placeholders with their respective lexicalisations. The instantiated templates are then grouped based on their DB-Subj and ordered in descending frequency of appearance of the DB-Subj in the input triple set (e.g. triples with a DB-Subj that has 3 mentions come before those with 2 mentions). Figure 2 shows a PredArg template, instantiated in Figure 4 in Appendix A.

Lexicalisation of properties. We handcrafted templates for all properties of the training, development and test splits of the WebNLG'23 dataset. There are 411 different properties, and since several properties can be verbalised the same way,[10] the total number of unique templates is lower (381).

In an effort to possibly reduce the human effort in the crafting of the templates in future developments of our (or others') system, we tried to reduce to a maximum the number of different templates to cover all properties. After examining the 411 properties and defining corresponding templates, we assigned each property a specific type according to the kind of information that it is transmitting. We defined 23 type labels such as *PART OF/MEMBER OF*, *ORIGIN LOCATION*, *SET MEMBERSHIP*, *[X] HAS [QTY] ENTITY*, etc. Each type is associated with a sentence template and a basic PredArg that can be used to verbalise the properties associated to it. We plan to use these basic labels to speed up the future extension of the generator.

Lexicalisation of DB-Subj and DB-Obj values. For each triple, the property and its pertinent domain and range classes determine whether the DB-Subj/Obj values will be lexicalised using their English or Irish label (human readable name). To obtain the latter, we take advantage of the *owl:sameAs* relation that links the DB-Subj/Obj entity of the English DBpedia to its equivalent entity in the Irish DBpedia version; if no equivalent entity is contained in the localised DBpedia version, we fall back to Google translate,[11] giving as input the English label without any further context.

4.2 Extensions to FORGe

We extended the available version of FORGe in two aspects: (i) manual crafting of the three types of dictionaries, and (ii) implementation of language-specific rules to cover the idiosyncracies of Irish. With respect to dictionaries, we added 457 mappings between concepts and lexical units and as many lexical unit descriptions, and we manually crafted the generic language-specific dictionary. For rules, we implemented 76 rules that apply exclusively to Irish (T3), which represents 2.78% of rules; Table 2 shows the breakdown of language-agnostic and language-specific rules per module. We also activated 65 existing T2 rules for Irish.

As Table 2 shows, 4 modules require Irish-specific rules: deep sentence structuring, surface sentence structuring, word order and agreement resolution and morphology processing; next we list the phenomena that required T3 and most T2 rules.

Deep sentence structuring

Relative particles (T3): the particle *a* is introduced to link the modified noun and the main verb in relative clauses; in case of prepositional relatives, the particle has a different form depending on the tense of the verb (present *a*, past *ar*).

Passive (T3): in Irish there are two alternative constructions where a passive form would be used in English. If the data refers to an action/event, an autonomous main verb form is used, e.g. for the triple `Acharya_Institute_of_Technology | established | 2000`, *bunaíodh*, the autonomous

[10] Properties such as *municipality*, *district*, or *country* are mapped to the same template as *location*, shown in Figure 2.

[11] We used the publicly available *Translator* module of the *googletrans* (version 3.1.0a0) library.

ID	FORGe module	# rl	% lang. ind. rl	# T3 GA rl	% T3 GA rl
1	Text planning	553	99.82	0	0
2	Lexicalisation	183	97.81	0	0
3	Communicative structuring	258	97.29	0	0
4	Deep sentence structuring	345	78.84	3	0.87
5	Surface sentence structuring	477	68.97	17	3.56
6	Syntactic aggregation	215	93.02	0	0
7	Referring Expression Generation	237	96.2	0	0
8	Word order and agreement resolution	265	50.57	17	6.42
9	Morphology processing	201	45.77	39	19.4
	All modules	2,734	81.82	76	2.78

Table 2: Number of rules, proportion of language-independent rules, and number and % of Irish-specific (T3) rules (rl) per FORGe module.

form of the verb *bunaigh* 'to establish' is used, as in *Bunaíodh Institiúid Teicneolaíochta Acharya sa bhliain 2000*, 'Acharya Institute of Technology was established in the year 2000'. Alternatively, where a state/location is referred to, e.g. for the triple `MotorSport_Vision|city|Longfield`, we have *tá*, the present tense of the auxiliary verb *bí* 'to be', and the past participle *lonnaithe* 'located', as in *Tá MotorSport Vision lonnaithe i gcathair Longfield*, 'MotorSport Vision is located in Longfield'.

Non-verbal copula (T3): Irish has two copular constructions. The verbal copula *bí* is used for changeable properties whereas the non-verbal copula *is* is used for more permanent properties such as area code, e.g. for `Darlington | areaCode | 01325` we have *Is é cód ceantair Darlington ná 01325*, where *is* connects *cód ceantair Darlington* 'Darlington area code' with its value, and the pronoun *é* agrees with the gender and number of the noun *cód* 'code'.

Surface sentence structuring

Determiners (T3): a definite determiner is only introduced on a noun *N* if *N*'s dependent is not a definite noun or a proper noun.

Dependencies (T2, 22 rules in common with Catalan, Greek, Spanish, French, Italian and Portuguese): surface-oriented dependencies are introduced as, e.g., *subject*, *direct object*, *modifier*, etc.

Word order and agreement resolution

Genitive chains (T3): in a chain of genitive elements, only the last element maintains the genitive case, e.g. in the case of 'the length of the runway of the aerodrome', only the last element 'aerodrome' has genitive case as in *Is é fad rúidbhealach an aeradróim 1,095m*.

Word order class (T3): when an element is established as a member of a class, the class name goes right after the copula, as in *Is milseog é Bionico* 'Bionico is a dessert'.

Possessive pronoun agreement (T3): the semantic number and gender of a possessor triggers agreement on the possessed. In the case of the triple `India | leader | T._S._Thakur`, the copular construction generates the text *Tá T.S. Shakur ina cheannaire ar an India*, 'T. S. Thakur is a leader of India', where we have the present tense of the verbal copula *bí*, followed by the subject *'T. S. Thakur'* and the subject complement *'ina cheannaire ar an India'*. The complement has a possessive pronoun *ina* that agrees in gender and number with the subject, i.e. *ina* is masculine singular reflecting the subject 'T. S. Thakur' and it triggers masculine singular agreement on the noun *cheannaire* 'leader'.

Ellipsis (T3): some rules look for pronouns to elide, in particular in relative and non-verbal copular constructions. Irish is a VSO language so a specific rule checks for repeated subjects on the right of the verb and replaces them with pronouns.[12]

Order between siblings (T2, 29 rules in common with Catalan, Greek, Spanish, French, Portuguese and sometimes Italian): for instance, in many languages, the determiner usually goes before all other dependents of the noun.

Morphology processing

Concatenations (T3): *don* is a contraction of *do an* 'for the' as in *Scríobh Nicholas Brodszky an ceol don scannán* meaning 'Nicholas Brodszky wrote the music **for the** film'.

Prefixes (T3): vowel-initial masculine nouns following the determiner *an* receive a *t-* prefix as in *Rugadh an **t-aisteoir** Bill Oddie in Rochdale* meaning 'The actor Bill Oddie was born in Rochdale'.

[12]Strictly speaking, this rule belongs to the REG module but since it has the same conditions of application as ellipsis in other languages, it was left in this module for the time being.

The preposition *le* triggers a prefix *h-* on following nouns starting with a vowel, and some past verbs get the prefix *d'*.

Mutations (T3): word-initial mutations are common in Irish and fulfil many grammatical functions, for example the noun *cathair* 'city' has various mutations depending on the number and gender of the possessive pronoun, e.g. there is lenition in *mo **ch**athair* 'my city', eclipsis in *ár **gc**athair* 'our city' and no mutation in *a **c**athair* 'her city'.

Verbal Adj/N, Prep. declension, V flags (T3): other rules cover the conversion of some adjectives and nouns into their verbal counterparts, the inflection of some prepositions and the insertion of a tag that flags vowel-initial verbs, as required by the morphology generator.

4.3 Interfacing FORGe with Irish NLP tools

In order to match the inputs expected by Irish NLP tools, we process FORGe outputs with regular expressions to replace reserved characters, introduce a '+' separator between morphological tags, and insert single line breaks between consecutive words and double line breaks between consecutive texts.

4.4 Post-processing

The post-processing consists of regular expressions to revert reserved characters to their original form, true-case and clean the texts, and take care of prefixing, hyphenation, contraction, lenition and eclipsis phenomena triggered by the inflected forms of words; see Appendix A for an example.

4.5 Evaluation

We report on both automatic and human evaluations of the quality of the texts generated with our pipeline (DCU/TCD in Tables 3 and 4). Both evaluations were carried out as part of the WebNLG'23 shared task by the task organisers; see details in the task overview paper (Aquilina et al., 2023).

	BLEU	BERT_F1
DCU-NLG	20.40	0.81
DCU/TCD	**16.66**	**0.77**
IREL	15.66	0.78
Cuni-Wue	15.87	0.77
Baseline	11.63	0.76

Table 3: WebNLG'23 automatic evaluation results.

For the automatic evaluations, outputs from all systems were compared to the reference human-translated Irish texts (1,779 test texts), and BLEU (Papineni et al., 2002), TER (Snover et al., 2006), chrF++ (Popović, 2017) and BERTScore (Zhang et al., 2019) were computed; see results in Table 3. For the human assessment, the organisers selected randomly the same 100 outputs for each system (and the corresponding 100 reference texts) and asked professional translators to rate the texts on a scale of 1 to 5 according to 4 criteria: **Fluency** and **Absence of Repetition** to capture the intrinsic quality of the texts, and **Absence of Omission** and **Absence of Additions** to capture the semantic faithfulness of the text with respect to the input triple sets; see Table 4 for results.

System	Flu.	Add.	Omi.	Rep.
Human	4.07	0.81	0.82	0.96
DCU-NLG	3.83	0.83	0.85	0.97
DCU/TCD	**3.35**	**0.84**	**0.81**	**0.89**
IREL	3.39	0.65	0.58	0.94
Cuni-Wue	2.98	0.55	0.51	0.92

Table 4: Results of the WebNLG'23 human evaluation; Human = human-translated texts, Flu. = Fluency, Add. = Absence of addition, Omi. = Absence of omission, Rep. = Absence of repetition.

Considering that all other systems including the baseline are combinations of (very) large language models (to generate English texts) and machine translation (to translate to Irish), we were surprised to see that our rule-based pipeline performed well in the automatic evaluations: we obtained a BLEU score only 4 points below the highest scoring system (a combination of GPT3.5 and Google Translate (Lorandi and Belz, 2023)), and higher that all non-GPT-based submissions. As comparison, for English text generation at WebNLG'20 (Castro Ferreira et al., 2020), the FORGe-based submission was 13 BLEU points lower than the highest scoring system and one of the lowest BLEU overall.[13] Our absolute BLEU score is much lower than FORGe's scores on English at WebNLG'20 (over 40); this is at least partly because BLEU was calculated with only one reference (compared to 2,5 on average in English, which produces higher scores), but it could also be due to the fact that we created our lexicalisations without reference to the gold Irish texts, i.e. surface similarity is likely to be low.

[13]There was significantly more gold data available in English compared to Irish.

The results of the human evaluation show that DCU/TCD-FORGe is on a par with the human references and the best system for Absence of Additions, Absence of Omissions and Repetition (no statistical difference in the scores according to the organisers), but is significantly less good in terms of Fluency. Part of the reason for this can be found in our own preliminary quality assessment of the output texts, during which Irish speakers mentioned that the way the information is packaged into sentences (Text planning task) is often unnatural, which directly affects the Fluency of texts. We plan to address this issue by replacing the text planning module by a statistical component.

Our system does not reach the level that can be achieved with very large language models, but unlike the latter, it is inherently energy- and resource-efficient: our complete pipeline has a disk space of about 8MB and runs with less than 1GB of RAM; it generates the whole WebNLG test set (1,779 texts) in about 15 min (0.5 sec/text). The generation pipeline is also reusable; it currently covers datasets such as E2E (Novikova et al., 2017) or Rotowire (Wiseman et al., 2017) in English, and adapting it to new domains is straightforward.

4.6 A new Irish dataset with rich annotations

Along with our architecture and our generation pipeline, we also release an Irish multilayer dataset with rich linguistically motivated intermediate representations. In order to create the dataset, we apply our whole generation pipeline described in Section 4 and save the intermediate representations in the process. The resulting dataset has ten layers, which correspond to the ten layers shown in Table 1.

Representations at all layers are multi-sentence graphs that can be grouped into the three main types from Section 2: directed acyclic graphs for semantic information, unordered dependency trees for syntactic information, and chains for morphological information. Nodes are connected across layers through individual IDs, and coreference is explicitly marked. Intermediate representations are represented as CoNLL-U tables.[14] Because CoNLL-U is a linear format that we use to represent unordered graphs and trees, we delimit sentences by `<SENT>` at the end of a group of nodes. All lines before `<SENT>` belong to the same sentence, but their relative order in the ConNLL-U file is not relevant. However, the order in which the sentences appear does correspond to their order in the text. For levels that are chains, the order of the lines is the order of the elements in the sentence. Detailed descriptions of format and levels can be found in (Mille et al., 2023); tagsets used, dataset statistics and sample structures are provided in Appendix B, C and D.

Due to the modular system architecture, dataset construction is flexible enough to allow the generation of a myriad of dataset variants in terms of verbalisation, sentence grouping/structuring, output simplicity/complexity, etc., simply by (de)activating optional modules (Table 1) or by introducing variation during the linguistic structuring task –thus providing multiple ways of verbalising each input triple. In contrast to neural generation, our approach ensures that output texts are faithful to the input, and will not contain inaccuracies, biases or offensive language. The dataset is publicly available, see Footnote 3.

5 Related work

Rule-based NLG. There is a long tradition of rule-based natural language generation systems such as REALPRO (Lavoie and Rainbow, 1997), ILEX (O'Donnell et al., 2001), IGEN (Varges and Mellish, 2001), SimpleNLG (Gatt and Reiter, 2009), MARQUIS (Wanner et al., 2010; Bouayad-Agha et al., 2012), OpenCCG (White and Rajkumar, 2012), NaturalOwl (Androutsopoulos et al., 2013), GenDR (Lareau et al., 2018) and others. More recently, RDFJSREALB (Lapalme, 2020) and FORGe (Mille et al., 2019) were adapted to WebNLG, but none were able to generate Irish text. Note that the idea of decomposing the generation process into steps has been the standard before the emergence of end-to-end systems, and that previous work on NLG already based their modules on the Meaning-Text theory, going back to REALPRO (Lavoie and Rainbow, 1997) and MARQUIS (Wanner et al., 2010). It is however the first time that a plug-and-play architecture is proposed with these modules, and the first time that an Irish rule-based NLG system is developed.

Irish datasets and language resources. There are few freely available monolingual Irish corpora, and moreover, domain-specific Irish datasets are scarce. Resources are mostly targeted towards machine translation and/or language analysis tasks. With the exception of the WebNLG 2023 data (and now the data presented in this paper), no datasets exist for text generation tasks (Lynn, 2023).

[14] https://universaldependencies.org/format.html

Monolingual corpora. Monolingual data include the New Corpus for Ireland (with fiction, news reports, official documents, etc.) (Kilgarriff et al., 2006), the unshuffled Irish portion of the 2019 OSCAR corpus (Suárez et al., 2019), the Gaois Corpus of Contemporary Irish (Ní Loingsigh et al., 2017), with news media and e-zines, or the Irish Wikipedia Vicipéid,[15] which draws directly from Fréamh an Eolais, an Irish-language encyclopedia of science and technology.[16] Moreover, a corpus of idioms (Ní Loingsigh, 2016) and Universal Dependency treebanks such as Irish UD (Lynn and Foster, 2016), pre-standard Irish UD (Scannell, 2022) and TwittIrish (Cassidy et al., 2022) are available.

Bilingual/Parallel corpora. Significant advances have been made in the collection and availability of bilingual corpora, including: (i) ParaCrawl v7 (Bañón et al., 2020), a collection of parallel corpora crawled from multi-lingual websites; (ii) the Gaois Parallel Corpus[17] of 26M Irish words and 24.5M English words; and in particular, (iii) the Irish-EU English-Irish Parallel Corpus which was a direct outcome of the European Language Resource Coordination project (ELRC[18]). This resource contains 195K+ parallel sentences, collected from various public bodies and government departments released via ELRC-SHARE[19]. In Ireland all national translation data is collected by *eSTÓR*.[20]

Irish tools and Models. The European Language Grid[21] catalogue lists a number of multilingual tools and services that support Irish (e.g. Bitextor, Opus MT, Systran). Irish NLP tools (Uí Dhonnchadha, 2009) offers the only suite of text analysis in Irish. Transformer Language Models (LM) such as multilingual BERT (M-BERT) (Devlin et al., 2019), and the language-agnostic BERT Sentence Embedding (Feng et al., 2022)) support Irish. The monolingual Irish gaBERT LM was trained on over 7.9M sentences, and outperforms baselines for tasks such as dependency parsing and multiword expression identification. (Barry et al., 2022).

6 Conclusions

We have presented a high-accuracy, energy and resource-efficient system for generating Irish text which achieves a satisfactory quality of output. Its modular architecture means that shortcomings can potentially be remedied by training statistical modules, such as a text structuring module for improved fluency, or by including enhanced rule-based modules which can be added to the pipeline.

This type of modular rule-based NLG system is particularly suitable for low-resource languages, where large amounts of training data is not available, and can play an important role in generating accurate fact-based online language content, such as Wikipedia pages. Such systems can be developed incrementally and language documentation is an inherent and valuable by-product of the system. In addition, rule-based systems tend to suffer less from the negative and harmful biases which have been identified in the application of some LLMs.

Limitations

Generation pipeline. Coverage and robustness of rule-based NLG: Although our experiments show that we are able to overcome some of the drawbacks of LLMs, the main bottleneck of any rule-based system remains coverage and robustness. In addition, it can be difficult for someone who is not familiar with the rule systems to edit it, and it usually requires knowledge of the language.

Dataset. Our dataset differs from previous work in that we do not use human-written texts; since texts are synthetic and produced by a deterministic generator, their variety and quality is limited by the knowledge encoded in the generator (in particular, they generally lack the naturalness of human-written texts), and they represent only a fraction of what is possible for a language to express.

The current intermediate representations are well-formed at all layers, but we are conscious that some phenomena would require some additional analysis; as e.g. the syntactic representations of copulas and their *é* pronoun (see Section 4.2).

Acknowledgements

Mille's contribution was funded by the European Union under the Marie Skłodowska-Curie grant agreement No 101062572 (M-FleNS).

Ethics Statement

Given that we do not resort to using language models nor to human evaluation with people who are not authors of this paper, the present work has no ethics implication that we are aware of.

[15] https://dumps.wikimedia.org/gawiki/
[16] https://ga.wikipedia.org/wiki
[17] https://www.gaois.ie/crp/ga/
[18] https://lr-coordination.eu/node/2
[19] https://elrc-share.eu/
[20] https://estor.ie/
[21] https://live.european-language-grid.eu/

References

Ion Androutsopoulos, Gerasimos Lampouras, and Dimitrios Galanis. 2013. Generating natural language descriptions from owl ontologies: the naturalowl system. *Journal of Artificial Intelligence Research*, 48:671–715.

Enrico Aquilina, Anya Belz, Claudia Borg, Liam Cripwell, Claire Gardent, Albert Gatt, John Judge, Michela Lorandi, Anna Nikiforoskaya, William Soto-Martinez, and Craig Thomson. 2023. The 2023 WebNLG shared task on low resource languages: Overview and evaluation results. In *Proceedings of the Workshop on Multimodal, Multilingual Natural Language Generation and Multilingual WebNLG Challenge*, page tbd, Prague, Czech Republic.

Marta Bañón, Pinzhen Chen, Barry Haddow, Kenneth Heafield, Hieu Hoang, Miquel Esplà-Gomis, Mikel L. Forcada, Amir Kamran, Faheem Kirefu, Philipp Koehn, Sergio Ortiz Rojas, Leopoldo Pla Sempere, Gema Ramírez-Sánchez, Elsa Sarrías, Marek Strelec, Brian Thompson, William Waites, Dion Wiggins, and Jaume Zaragoza. 2020. ParaCrawl: Web-scale acquisition of parallel corpora. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4555–4567, Online. Association for Computational Linguistics.

James Barry, Joachim Wagner, Lauren Cassidy, Alan Cowap, Teresa Lynn, Abigail Walsh, Mícheál J Ó Meachair, and Jennifer Foster. 2022. gaBERT–an Irish Language Model. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 4774–4788, Marseille,France.

Kenneth R Beesley and Lauri Karttunen. 2003. Finite-state morphology: Xerox tools and techniques. *CSLI, Stanford*.

Nadjet Bouayad-Agha, Gerard Casamayor, Simon Mille, Marco Rospocher, Horacio Saggion, Luciano Serafini, and Leo Wanner. 2012. From ontology to nl: Generation of multilingual user-oriented environmental reports. In *International Conference on Application of Natural Language to Information Systems*, pages 216–221. Springer.

Lauren Cassidy, Teresa Lynn, James Barry, and Jennifer Foster. 2022. TwittIrish: A Universal Dependencies treebank of tweets in modern Irish. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6869–6884, Dublin, Ireland.

Thiago Castro Ferreira, Claire Gardent, Nikolai Ilinykh, Chris van der Lee, Simon Mille, Diego Moussallem, and Anastasia Shimorina. 2020. The 2020 bilingual, bi-directional WebNLG+ shared task: Overview and evaluation results (WebNLG+ 2020). In *Proceedings of the 3rd International Workshop on Natural Language Generation from the Semantic Web (WebNLG+)*, pages 55–76, Dublin, Ireland (Virtual). Association for Computational Linguistics.

Thiago Castro Ferreira, Chris van der Lee, Emiel van Miltenburg, and Emiel Krahmer. 2019. Neural data-to-text generation: A comparison between pipeline and end-to-end architectures. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 552–562, Hong Kong, China. Association for Computational Linguistics.

Department of Tourism, Culture, Arts, Gaeltacht, Sport Media. 2022. Digital plan for the irish language: Speech and language technologies 2023-2027. Technical report, Government of Ireland: Available at `https://assets.gov.ie/250129/1425436f-e1da-4661-8483-92d9ddb4a716.pdf`.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Uí Dhonnchadha, Caoilfhionn Nic Pháidín, and Josef Van Genabith. 2003. Design, implementation and evaluation of an inflectional morphology finite state transducer for Irish. *Machine Translation*, 18:173–193.

Ondřej Dušek, Jekaterina Novikova, and Verena Rieser. 2018. Findings of the E2E NLG challenge. In *Proceedings of the 11th International Conference on Natural Language Generation*, pages 322–328, Tilburg University, The Netherlands. Association for Computational Linguistics.

Ondřej Dušek, Jekaterina Novikova, and Verena Rieser. 2020. Evaluating the state-of-the-art of end-to-end natural language generation: The e2e nlg challenge. *Computer Speech & Language*, 59:123–156.

Fangxiaoyu Feng, Yinfei Yang, Daniel Cer, Naveen Arivazhagan, and Wei Wang. 2022. Language-agnostic BERT sentence embedding. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 878–891, Dublin, Ireland. Association for Computational Linguistics.

Claire Gardent, Anastasia Shimorina, Shashi Narayan, and Laura Perez-Beltrachini. 2017. Creating training corpora for micro-planners. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Vancouver, Canada. Association for Computational Linguistics.

Albert Gatt and Ehud Reiter. 2009. SimpleNLG: A realisation engine for practical applications. In *Proceedings of the 12th European Workshop on Natural*

Language Generation (ENLG 2009), pages 90–93, Athens, Greece. Association for Computational Linguistics.

Mans Hulden. 2009. Foma: a finite-state compiler and library. In *Proceedings of the Demonstrations Session at EACL 2009*, pages 29–32, Athens, Greece. Association for Computational Linguistics.

Zdeněk Kasner and Ondrej Dusek. 2022. Neural pipeline for zero-shot data-to-text generation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3914–3932, Dublin, Ireland. Association for Computational Linguistics.

Adam Kilgarriff, Michael Rundell, and Elaine Uí Dhonnchadha. 2006. Efficient corpus development for lexicography: building the new corpus for ireland. *Language resources and evaluation*, 40:127–152.

Paul Kingsbury and Martha Palmer. 2002. From TreeBank to PropBank. In *Proceedings of the Third International Conference on Language Resources and Evaluation (LREC'02)*, Las Palmas, Canary Islands - Spain. European Language Resources Association (ELRA).

Guy Lapalme. 2020. RDFjsRealB: a symbolic approach for generating text from RDF triples. In *Proceedings of the 3rd International Workshop on Natural Language Generation from the Semantic Web (WebNLG+)*, pages 144–153, Dublin, Ireland (Virtual). Association for Computational Linguistics.

François Lareau, Florie Lambrey, Ieva Dubinskaite, Daniel Galarreta-Piquette, and Maryam Nejat. 2018. GenDR: A generic deep realizer with complex lexicalization. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Benoit Lavoie and Owen Rainbow. 1997. A fast and portable realizer for text generation systems. In *Fifth Conference on Applied Natural Language Processing*, pages 265–268.

Michela Lorandi and Anya Belz. 2023. Data-to-text generation for severely under-resourced languages with GPT-3.5: A bit of help needed from Google Translate. In *Proceedings of the Workshop on Multimodal, Multilingual Natural Language Generation and Multilingual WebNLG Challenge*, page tbd, Prague, Czech Republic.

Teresa Lynn. 2022. Language report Irish. In *European Language Equality: D120 Report on the Irish Language*, pages 1–24. https://european-language-equality.eu/.

Teresa Lynn. 2023. Language report Irish. In *European Language Equality: A Strategic Agenda for Digital Language Equality*, pages 163–166. Springer.

Teresa Lynn and Jennifer Foster. 2016. Universal Dependencies for Irish. In *Proceedings of the 2nd Celtic Language Technology Workshop*, pages 79–92, Paris, France.

Igor A. Mel'čuk. 1973. Towards a linguistic 'Meaning ↔ Text' model. *Trends in Soviet theoretical linguistics*, pages 33–57.

Igor A. Mel'čuk. 1988. *Dependency Syntax: Theory and Practice*. State University of New York Press, Albany, NY.

Simon Mille, Stamatia Dasiopoulou, and Leo Wanner. 2019. A portable grammar-based nlg system for verbalization of structured data. In *Proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing*, pages 1054–1056.

Simon Mille, François Lareau, Anya Belz, and Stamatia Dasiopoulou. 2023. Mod-D2T: A Multi-layer Dataset for Modular Data-to-Text Generation. In *Proceedings of the 16th International Conference on Natural Language Generation*, page tbd, Prague, Czech Republic.

Amit Moryossef, Yoav Goldberg, and Ido Dagan. 2019. Step-by-step: Separating planning from realization in neural data-to-text generation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2267–2277, Minneapolis, Minnesota. Association for Computational Linguistics.

Katie Ní Loingsigh. 2016. Towards a lexicon of irish-language idioms. In *Proceedings of the 2nd Celtic Language Technology Workshop*, pages 69–78, Paris, France.

Jekaterina Novikova, Ondřej Dušek, and Verena Rieser. 2017. The E2E dataset: New challenges for end-to-end generation. In *Proceedings of the 18th Annual SIGdial Meeting on Discourse and Dialogue*, pages 201–206, Saarbrücken, Germany. Association for Computational Linguistics.

Katie Ní Loingsigh, Brian Ó Raghallaigh, and Gearóid Ó Cléircín. 2017. The design and development of Corpas na Gaeilge comhaimseartha (corpus of contemporary Irish). In *Proceedings of the 9th International Corpus Linguistics Conference*.

Mick O'Donnell, Chris Mellish, Jon Oberlander, and Alistair Knott. 2001. Ilex: an architecture for a dynamic hypertext generation system. *Natural Language Engineering*, 7(3):225.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Maja Popović. 2017. chrF++: words helping character n-grams. In *Proceedings of the Second Conference on Machine Translation*, pages 612–618, Copenhagen, Denmark. Association for Computational Linguistics.

Ratish Puduppully and Mirella Lapata. 2021. Data-to-text generation with macro planning. *Transactions of the Association for Computational Linguistics*, 9:510–527.

Ehud Reiter and Robert Dale. 1997. Building applied natural language generation systems. *Natural Language Engineering*, 3(1):57–87.

Kevin Scannell. 2022. Diachronic parsing of pre-standard Irish. In *Proceedings of the 4th Celtic Language Technology Workshop within LREC2022*, pages 7–13, Marseille, France.

Matthew Snover, Bonnie Dorr, Rich Schwartz, Linnea Micciulla, and John Makhoul. 2006. A study of translation edit rate with targeted human annotation. In *Proceedings of the 7th Conference of the Association for Machine Translation in the Americas: Technical Papers*, pages 223–231, Cambridge, Massachusetts, USA. Association for Machine Translation in the Americas.

Pedro Javier Ortiz Suárez, Benoît Sagot, and Laurent Romary. 2019. Asynchronous pipeline for processing huge corpora on medium to low resource infrastructures. In *7th Workshop on the Challenges in the Management of Large Corpora (CMLC-7)*, Marseille, France. Leibniz-Institut für Deutsche Sprache.

Elaine Uí Dhonnchadha. 2009. *Part-of-speech tagging and partial parsing for Irish using finite-state transducers and constraint grammar*. Ph.D. thesis, Dublin City University.

Sebastian Varges and Chris Mellish. 2001. Instance-based natural language generation. In *Second Meeting of the North American Chapter of the Association for Computational Linguistics*.

Leo Wanner, Bernd Bohnet, Nadjet Bouayad-Agha, Francois Lareau, and Daniel Nicklaß. 2010. MARQUIS: Generation of user-tailored multilingual air quality bulletins. *Applied Artificial Intelligence*, 24(10):914–952.

Michael White and Rajakrishnan Rajkumar. 2012. Minimal dependency length in realization ranking. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 244–255, Jeju Island, Korea. Association for Computational Linguistics.

Sam Wiseman, Stuart Shieber, and Alexander Rush. 2017. Challenges in data-to-document generation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2253–2263, Copenhagen, Denmark. Association for Computational Linguistics.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*.

A Sample input and output structures

The figures in the next page illustrate the generation process starting from an input triple set that corresponds to the following English text:

> *Agra Airport, operated by Indian Air Force, is located in India. Its ICAO location identifier is VIAG.*

Figure 3 shows a WebNLG'23 input, and Figure 4 shows the output of the lexicalisation module. The FORGe, morphology and post-processing outputs are shown in a one-word-per-line format in Table 5. The output Irish text is the following:

> *Tá Agra Airport, reáchtáilte ag Indian Air Force, lonnaithe ins An India. Tá VIAG in a aitheantóir suímh ICAO.*

B Irish dataset: Tagsets used

The edge labels for semantic graphs come mainly from PropBank (Kingsbury and Palmer, 2002), plus some generic labels such as `Location` and `Time`; see Table 6. The ones for deep syntactic trees come from Meaning-Text Theory (Mel'čuk, 1988); see Table 7. As for surface syntactic edge labels, they are our own; see Table 8.

C Irish dataset: Statistics

There are 13,211, 1,667 and 1,779 texts in the training, development and test splits respectively. Tables 9-10 provide an overview of the number of nodes and sentences per text for all splits. Our 10 intermediate layers contain over 2 million nodes.

D Irish dataset: Sample structures

The annotations are released in CoNLL-U format, but because of space constraints, we have truncated the data in Tables 11–20 below: (i) we dropped unused columns and renamed the remaining ones for readability; (ii) we removed feature names to retain only their values; (iii) we omit the metadata, which specifies the text ID, the level of representation (see the captions) and the corresponding text string. The showcased structures all correspond to the same text as in Appendix A.

```
<entry category="Airport" eid="719" shape="(X (X) (X) (X))" shape_type="sibling" size="3">
  <modifiedtripleset>
    <mtriple>Agra_Airport | location | India</mtriple>
    <mtriple>Agra_Airport | operatingOrganisation | Indian_Air_Force</mtriple>
    <mtriple>Agra_Airport | icaoLocationIdentifier | &quot;VIAG&quot;</mtriple>
  </modifiedtripleset>
</entry>
```

Figure 3: A sample WebNLG input with 3 triples (same as Figure 1)

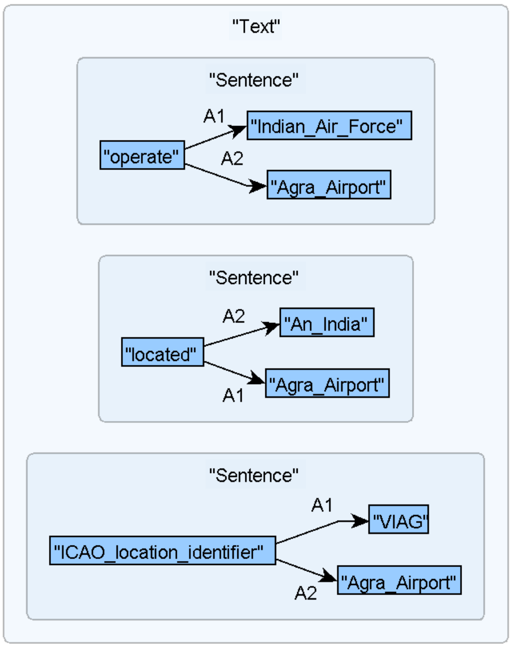

Figure 4: Lexicalisation output: instantiated PredArg templates

FORGe	Morphology	Post-processing
bí+Verb+PresInd	tá	Tá
Agra_Airport+Noun+Masc+Com+Sg	Agra_Airport	Agra Airport
,	,	,
reáchtáilte	reáchtáilte	reáchtáilte
ag	ag	ag
Indian_Air_Force+Noun+Masc+Com+Sg	Indian_Air_Force	Indian Air Force
,	,	,
lonnaithe+Adj+Masc+Com+Sg	lonnaithe	lonnaithe
i	i	ins
An_India+Noun+Masc+Com+Sg	An_India	An India
.	.	.
bí+Verb+PresInd	tá	Tá
VIAG+Noun+Masc+Com+Sg	VIAG	VIAG
i	i	in
a	a	a
aitheantóir+Noun+Masc+Com+Sg	aitheantóir	aitheantóir
suímh	suímh	suímh
ICAO+Noun+Masc+Com+Sg	ICAO	ICAO
.		

Table 5: FORGe, morphology and post-processing outputs (one word per line for convenience)

Label	Description	Example
`A0—A6`	n-th argument of a predicate or quasi-predicate	speak→ English
`Location`	location	born→ Paris
`Time`	time	build→ 1932
`NonCore`	inverted first argument of a predicate	runway→ second
`Set`	list of elements	and→ speak
`Elaboration`	(i) none of governor or dependent are argument of the other (ii) unknown argument slot	above me→ 610m

Table 6: Edge labels of semantic graphs

Label	Description	Example
`I—VI`	n-th complement of a syntactic predicate	`speak`→ English
`ATTR`	modifier	runway→ second
`COORD`	coordination	staff members→ and
`APPEND`	parenthetical modifier	Hypermarcas Brazil→ (s.a.)

Table 7: Edge labels of deep syntactic trees

Label	Description
`adjunct`	backgrounded adverbial
`adv`	general adverbial (not restrictive nor backgrounded)
`agent`	between non-finite verb and its 1st argument
`analyt_pass`	between passive auxiliary and main verb
`appos`	nominal noun modifier (apposition)
`attr`	prepositional noun modifier (attributive)
`aux_phras`	between elements of multi-word proper nouns
`compar`	between adjective and comparative
`compar_conj`	complement of a comparative conjunction
`coord`	between 1st conjunct and conjunction
`coord_conj`	between conjunction and 2nd conjunct
`copul`	complement of a copula
`det`	determiner of a noun
`dobj`	direct object
`iobj`	indirect object
`modal`	between modal verb and main verb
`modif`	adjectival or participial noun modifier
`obl_compl`	complement (argument) of a noun
`obl_obj`	prepositional object (not direct or indirect)
`prepos`	complement of a preposition
`quant`	numeral noun modifier (quantificative)
`quasi_subj`	grammatical (usually empty) subject
`restr`	restrictive adverbial or modifier (adjacent to governor)
`relat`	clausal noun modifier (relative)
`sub_conj`	complement of a subordinating conjunction
`subj`	subject of verb

Table 8: Edge labels of Irish surface syntactic trees

Layer	N	S
PredArg	152,750	48,776
PredArg-Agg	134,008	31,065
PredArg-Lex	134,008	31,065
PredArg-Comm	143,343	31,065
DSynt	175,019	31,065
SSynt	254,128	31,065
SSynt-Agg	255,499	29,215
REG	254,355	29,215
DMorph	283,593	29,228
Text	285,727	29,228

Table 9: Total number of nodes (N) and sentences (S) per layer.

Layer	N	S	N/S
PredArg	9.2	2.9	3.1
PredArg-Agg	8.0	1.9	4.4
PredArg-Lex	8.0	1.9	4.4
PredArg-Comm	8.6	1.9	4.7
DSynt	10.5	1.9	5.7
SSynt	15.3	1.9	8.3
SSynt-Agg	15.3	1.8	8.9
REG	15.3	1.8	8.8
DMorph	17.0	1.8	9.8
Text	17.2	1.8	9.9

Table 10: Average number of nodes (N), sentences (S) and nodes per sentence (N/S) for each text, per layer.

<table>
<tr><th>ID</th><th>Semanteme</th><th>Features</th><th>Head</th><th>Rel</th><th>Misc</th></tr>
<tr><td>1</td><td>located</td><td>_</td><td>0</td><td>root</td><td>src=1</td></tr>
<tr><td>2</td><td>Agra_Airport</td><td>ne</td><td>1</td><td>A1</td><td>coref=0|src=2</td></tr>
<tr><td>3</td><td>An_India</td><td>location|ne</td><td>1</td><td>A2</td><td>coref=1|src=3</td></tr>
<tr><td>4</td><td><SENT></td><td>_</td><td>_</td><td>_</td><td>_</td></tr>
<tr><td>5</td><td>operate</td><td>pres</td><td>0</td><td>root</td><td>src=4</td></tr>
<tr><td>6</td><td>Indian_Air_Force</td><td>ne</td><td>5</td><td>A1</td><td>coref=2|src=6</td></tr>
<tr><td>7</td><td>Agra_Airport</td><td>def|ne</td><td>5</td><td>A2</td><td>coref=0|src=5</td></tr>
<tr><td>8</td><td><SENT></td><td>_</td><td>_</td><td>_</td><td>_</td></tr>
<tr><td>9</td><td>ICAO_location_identifier</td><td>def</td><td>0</td><td>root</td><td>src=7</td></tr>
<tr><td>10</td><td>Agra_Airport</td><td>_</td><td>9</td><td>A2</td><td>coref=0|src=8</td></tr>
<tr><td>11</td><td>VIAG</td><td>ne</td><td>9</td><td>A1</td><td>coref=3|src=9</td></tr>
<tr><td>12</td><td><SENT></td><td>_</td><td>_</td><td>_</td><td>_</td></tr>
</table>

Table 11: Predicate-argument structure (PredArg).

<table>
<tr><th>ID</th><th>Semanteme</th><th>Features</th><th>Head</th><th>Rel</th><th>Misc</th></tr>
<tr><td>1</td><td>located</td><td>rheme</td><td>0</td><td>root</td><td>src=1</td></tr>
<tr><td>2</td><td>An_India</td><td>location|ne</td><td>1</td><td>A2</td><td>coref=1|src=3</td></tr>
<tr><td>3</td><td>operate</td><td>pres</td><td>0</td><td>root</td><td>src=4</td></tr>
<tr><td>4</td><td>Indian_Air_Force</td><td>ne</td><td>3</td><td>A1</td><td>coref=2|src=6</td></tr>
<tr><td>5</td><td>Agra_Airport</td><td>ne</td><td>1,3</td><td>A1,A2</td><td>coref=0|src=2</td></tr>
<tr><td>6</td><td><SENT></td><td>_</td><td>_</td><td>_</td><td>_</td></tr>
<tr><td>7</td><td>ICAO_location_identifier</td><td>def</td><td>0</td><td>root</td><td>src=7</td></tr>
<tr><td>8</td><td>Agra_Airport</td><td>_</td><td>7</td><td>A2</td><td>coref=0|src=8</td></tr>
<tr><td>9</td><td>VIAG</td><td>ne</td><td>7</td><td>A1</td><td>coref=3|src=9</td></tr>
<tr><td>10</td><td><SENT></td><td>_</td><td>_</td><td>_</td><td>_</td></tr>
</table>

Table 12: Aggregated predicate-argument structure (PredArg-Agg; corresponds to Figure 4).

ID	Semanteme	POS	Features	Head	Rel	Misc
1	located	JJ	jj|rheme	0	root	src=1
2	An_India	NP	location|ne	1	A2	src=3
3	operate	VB	pres|vb	0	root	src=4
4	Indian_Air_Force	NP	ne	3	A1	src=6
5	Agra_Airport	NP	ne	1,3	A1,A2	coref=0|src=2
6	<SENT>	_	_	_	_	_
7	ICAO_location_identifier	NN	def|nn	0	root	src=7
8	Agra_Airport	NN	_	7	A2	coref=0|src=8
9	VIAG	NP	ne	7	A1	src=9
10	<SENT>	_	_	_	_	_

Table 13: Lexicalised predicate-argument structure (PredArg-Lex).

ID	Semanteme	POS	Features	Head	Rel	Misc
1	reáchtáil	VB	pres	0	root	src=4
2	lonnaithe	JJ	rheme	0	root	src=1
3	Agra_Airport	NP	ne	1,2	A2,A1	coref=0|src=2
4	An_India	NP	location|ne	2	A2	src=3
5	Indian_Air_Force	NP	ne	1	A1	src=6
6	<SENT>	_	_	_	_	_
7	aitheantóir	NN	def|rheme	0	root	src=7
8	Agra_Airport	NN	_	7	A2	coref=0|src=8
9	VIAG	NP	ne	7	A1	src=9
10	<SENT>	_	_	_	_	_

Table 14: Predicate-argument structure with thematicity (PredArg-Th).

ID	Lexeme	POS	Features	Head	Rel	Misc
1	bí	VB	fin|decl|act	0	root	src=1
2	Agra_Airport	NP	_	1	I	coref=0|src=2
3	reáchtáil	VB	part|pres	2	ATTR	src=4
4	Indian_Air_Force	NP	_	3	I	src=6
5	lonnaithe	JJ	_	1	II	src=1
6	An_India	NP	location	5	II	src=3
7	<SENT>	_	_	_	_	_
8	bí	VB	masc|act|fin|decl	0	root	src=7
9	VIAG	NP	_	8	I	src=9
10	aitheantóir	NN	masc|gen|sg	8	II	src=7
11	Agra_Airport	NN	sg	10	II	coref=0|src=8
12	<SENT>	_	_	_	_	_

Table 15: Deep syntactic representation (DSynt).

ID	Lexeme	POS	Features	Head	Rel	Misc
1	bí	VB	decl|fin|ind|pres	0	root	src=1
2	lonnaithe	JJ	acc	1	dobj	src=1
3	Agra_Airport	NP	nom|masc|sg|ne	1	subj	coref=0|src=2
4	reáchtáil	VB	part	3	modif	src=4
5	ag	IN	_	4	agent	src=6
6	i	IN	_	2	obl_compl	src=3
7	An_India	NP	sg|dat|location|masc|ne	6	prepos	src=3
8	Indian_Air_Force	NP	nom|masc|sg|ne	5	prepos	src=6
9	<SENT>	_	_	_	_	_
10	bí	VB	pres|decl|fin|masc|ind	0	root	src=7
11	i	IN	gen	10	obl_obj	src=7
12	aitheantóir	NN	dat|masc|sg|gen	11	prepos	src=7
13	ar	IN	_	12	obl_compl	src=8
14	Agra_Airport	NN	dat|masc|sg	13	prepos	coref=0|src=8
15	VIAG	NP	nom|masc|sg|ne	10	subj	src=9
16	suímh ICAO	NN	sg|masc|nom	12	restr	src=7
17	a	DT	-	12	det	src=7
18	<SENT>	_	_	_	_	_

Table 16: Surface syntactic representation (SSynt).

ID	Lexeme	POS	Features	Head	Rel	Misc
1	bí	VB	ind|sg|sg|decl|fin|pres	0	root	src=1
2	lonnaithe	JJ	sg|sg|acc	1	dobj	src=1
3	i	IN	sg|sg	2	obl_compl	src=3
4	Agra_Airport	NP	sg|nom|sg|masc|masc|ne	1	subj	coref=0|src=2
5	reáchtáil	VB	sg|sg|part	4	modif	src=4
6	ag	IN	sg|sg	5	agent	src=6
7	Indian_Air_Force	NP	nom|masc|sg|masc|sg|ne	6	prepos	src=6
8	An_India	NP	masc|sg|dat|location|masc|sg|ne	3	prepos	src=3
9	<SENT>	_	_	_	_	_
10	bí	VB	pres|sg|sg|decl|fin|masc|masc|ind	0	root	src=7
11	i	IN	sg|sg|gen	10	obl_obj	src=7
12	aitheantóir	NN	dat|sg|masc|gen|masc|sg	11	prepos	src=7
13	ar	IN	sg|sg	12	obl_compl	src=8
14	Agra_Airport	NN	masc|dat|masc|sg|sg	13	prepos	coref=0|src=8
15	VIAG	NP	nom|sg|masc|masc|sg|ne	10	subj	src=9
16	suímh ICAO	NN	sg|sg|masc|nom|masc	12	restr	src=7
17	a	DT	-|sg|sg	12	det	src=7
18	<SENT>	_	_	_	_	_

Table 17: Aggregated surface syntactic representation (SSynt-Agg).

ID	Lexeme	POS	Features	Head	Rel	Misc
1	bí	VB	sg|sg|decl|fin|pres|ind	0	root	src=1
2	lonnaithe	JJ	sg|acc|sg	1	dobj	src=1
3	i	IN	sg|sg	2	obl_compl	src=3
4	Agra_Airport	NP	masc|sg|sg|nom|masc|ne	1	subj	coref=0|src=2
5	reáchtáil	VB	part|sg|sg	4	modif	src=4
6	An_India	NP	location|masc|masc|sg|dat|sg|ne	3	prepos	src=3
7	ag	IN	sg|sg	5	agent	src=6
8	Indian_Air_Force	NP	masc|masc|sg|sg|nom|ne	7	prepos	src=6
9	<SENT>	_	_	_	_	_
10	bí	VB	pres|ind|masc|sg|decl|sg|fin|masc	0	root	src=7
11	i	IN	sg|sg|gen	10	obl_obj	src=7
12	aitheantóir	NN	masc|gen|masc|sg|sg|dat	11	prepos	src=7
13	_PRO_	PP	masc|sg|dat|masc|sg	12	obl_compl	coref=0|src=8
14	suímh ICAO	NN	masc|sg|nom|masc|sg	12	restr	src=7
15	VIAG	NP	masc|sg|sg|nom|masc|ne	10	subj	src=9
16	<SENT>	_	_	_	_	_

Table 18: Pronominalised surface syntactic representation (SSynt-Pro).

ID	Word	POS	Features	Misc
1	bí	VB	pres|vi|decl|fin|sg|ind	src=1
2	Agra_Airport	NP	nom|masc|sg|invar	coref=0|src=2
3	reáchtáil	VB	nom|part|masc|sg|vti	src=4
4	ag	IN	sg	src=6
5	Indian_Air_Force	NP	sg|nom|masc|invar	src=6
6	lonnaithe	JJ	sg|acc|masc	src=1
7	i	IN	sg	src=3
8	An_India	NP	dat|masc|sg|invar	src=3
9	.	_	_	src=-
10	bí	VB	ind|pres|vi|sg|decl|fin|masc	src=7
11	VIAG	NP	nom|masc|sg|invar	src=9
12	i	IN	sg	src=7
13	_PRO_	PP	dat|masc|sg	coref=0|src=8
14	aitheantóir	NN	masc|sg|dat	src=7
15	suímh ICAO	NN	nom|masc|sg	src=7
16	.	_	_	src=-

Table 19: Deep morphological representation (DMorph).

ID	Word	POS	Misc
1	bí%Verb%PresInd	VB	src=1
2	Agra_Airport%Noun%Masc%Com%Sg	NP	coref=0|src=2
3	,	_	src=-
4	reáchtáilte	VB	src=4
5	ag	IN	src=6
6	Indian_Air_Force%Noun%Masc%Com%Sg	NP	src=6
7	,	_	src=-
8	lonnaithe%Adj%Masc%Com%Sg	JJ	src=1
9	i	IN	src=3
10	An_India%Noun%Masc%Com%Sg	NP	src=3
11	.	_	src=-
12	bí%Verb%PresInd	VB	src=7
13	VIAG%Noun%Masc%Com%Sg	NP	src=9
14	i	IN	src=7
15	a	PP	coref=0|src=8
16	aitheantóir%Noun%Masc%Com%Sg	NN	src=7
17	suímh ICAO%Noun%Masc%Com%Sg	NN	src=7
18	.	_	src=-

Table 20: Surface morphological representation (SMorph; corresponds to Table 5).