

Spanish Triple-to-Text Benchmark on Low-Resource Large Language Models

Virginia Ramón-Ferrer^{a,*}, Carlos Badenes-Olmedo^a and Oscar Corcho^a

^a *Ontology Engineering Group, Universidad Politécnica de Madrid, Madrid, Spain*

E-mails: virginia.ramon@upm.es, carlos.badenes@upm.es, oscar.corcho@upm.es

Abstract. The verbalisation of structured data is a beneficial process for several applications. In the context of knowledge graphs (KGs), transforming RDF triples into natural language facilitates tasks such as KG documentation or alternative exploration methods for different user needs. While significant progress has been made on the English verbalisation of KGs, Spanish remains an under-represented language for this task due to the lack of suitable resources. This hinders developing and evaluating models capable of generating high-quality Spanish verbalisations. To tackle this problem, we create a Spanish adaptation of the WebNLG dataset, a benchmark consisting of over 45,000 verbalisations paired with DBpedia triple sets. To our knowledge, this is the first formal attempt to provide such a dataset in Spanish, which not only serves for data verbalisation but can also potentially support the automated generation of RDF triples from text. We leverage this dataset to conduct a comprehensive evaluation of resource-efficient models for the Spanish triple-to-text task employing two different learning approaches: context learning (zero-shot, one-shot, and few-shot settings) and supervised learning through partial fine-tuning. Our results highlight the challenges of generating fluent and accurate Spanish text and demonstrate that partial fine-tuning of the evaluated models significantly improves performance.

Keywords: data-to-text, triple-to-text, data verbalisation, triples verbalisation, Spanish WebNLG

1. Introduction

Knowledge graphs contain interconnected entities and relationships, usually presented as RDF [69] triples with a *subject-predicate-object* structure. While knowledge graphs are useful for machines, humans naturally understand and interpret plain text more intuitively than structured data, as it aligns with our ability to process language, context and nuances related to it. In the context of Natural Language Processing (NLP), the data-to-text task focuses on converting structured data into natural language text. Its goal is to make complex data more understandable and accessible by generating human-readable summaries or descriptions from raw structured data.

A lot of effort has been put into automating both the generation and verbalisation of triples with a growing interest in the use of language models for said tasks [31]. Most of these efforts only focus on the English language [47], leaving a wide gap between said language and others. In fact, there is a notable lack of resources for other languages. Chinonso Cynthia Osuji et al [47] report that, of 90 works they studied in their review of data-to-text literature, 85% of them were approaches focused exclusively on English and that only 2 of the works, which propose multilingual approaches, feature the Spanish language, none of which tackle the triples-to-text task. This highlights the need to develop resources for other languages, such as Spanish. Over 450 million people around the world speak Spanish

*Corresponding author. E-mail: virginia.ramon@upm.es.

natively [7], which represents around 6% of the world population, without considering the population that speaks Spanish non-natively. Even though Spanish is one of the most spoken languages in the world, the resources found related to the automatic generation and verbalisation of triples in this language are almost non-existent.

In this work, we present our contribution to the task of knowledge graph, specifically triples, verbalisation in Spanish, focusing on the triples-to-text challenge. Spanish is a language characterised by its rich inflectional morphology and great flexibility in word order, which, in comparison to English, includes a more extensive verbal and nominal inflection and multiple acceptable positions for subjects and objects around the verb [2, 41]. As a result, a single structured input may correspond to a broader set of equally natural verbalisation options in Spanish. For example, for the triple *[subject: Maria, relationship: need, object: help]*, English commonly allows expressions such as “Maria needs help.”, “Maria requires help.” or “Maria is in need of help.” In Spanish, the same content can be rendered as “María necesita ayuda.”, “María requiere ayuda.”, “María precisa ayuda.”, “María tiene necesidad de ayuda.” or “María está necesitada de ayuda.”, among others. To tackle this task, we create a semi-supervised Spanish adaptation of the WebNLG dataset [16], which contains the Spanish translation of the triples-verbalisation pairs included in the English WebNLG. For this, we followed an automatic machine translation process which was then verified and partially manually revised through a detection process of potentially problematic cases. We also present a study of the performance of a selection of resource-efficient Large Language Models (LLMs) on the task of translating Spanish triples to text. With this study, we aim to answer three research questions:

- **RQ1:** *How effectively can resource-efficient LLMs verbalise triples into Spanish across different complexity levels?*
- **RQ2:** *How does task contextualisation through examples impact model performance in Spanish triple verbalisation?*
- **RQ3:** *What are the comparative advantages and limitations of context learning versus partial fine-tuning for Spanish triple verbalisation?*

This article presents our adaptation of WebNLG to Spanish and the evaluation of the performance of low-resource LLMs for the verbalisation of triples in Spanish. It is structured as follows: We introduce the background related to structured data-to-text verbalisation in Section 2, followed by the introduction to the proposed Spanish WebNLG and the methodology followed for its development in Section 3. In Section 4 we explain the process followed to fine-tune and evaluate a selection of LLMs on the task of Spanish triples-to-text generation using our proposed dataset, in Section 4.1 and the results obtained in the previous process, in Section 4.2. Finally, in Section 5, we present the conclusions extracted from the previous contributions and future work.

All the code and resources used to develop this work and the results obtained are available in GitHub¹ and Zenodo².

2. Background

Structured data verbalisation, also known as data-to-text generation, is the name given to the task of generating natural language passages from structured data. Structured data can be stored in different forms, with the most common being graphs, tables, and meaningful representations (MR) such as key-value structures or Abstract Meaning Representation (AMR) [3], which is a semantic representation framework that captures the meaning of a sentence as a rooted, labelled graph of concepts and their relationships (*figure 1*).

¹https://github.com/virginia-r99/Spanish_WebNLG_triples-to-text

²<https://doi.org/10.5281/zenodo.15064345>

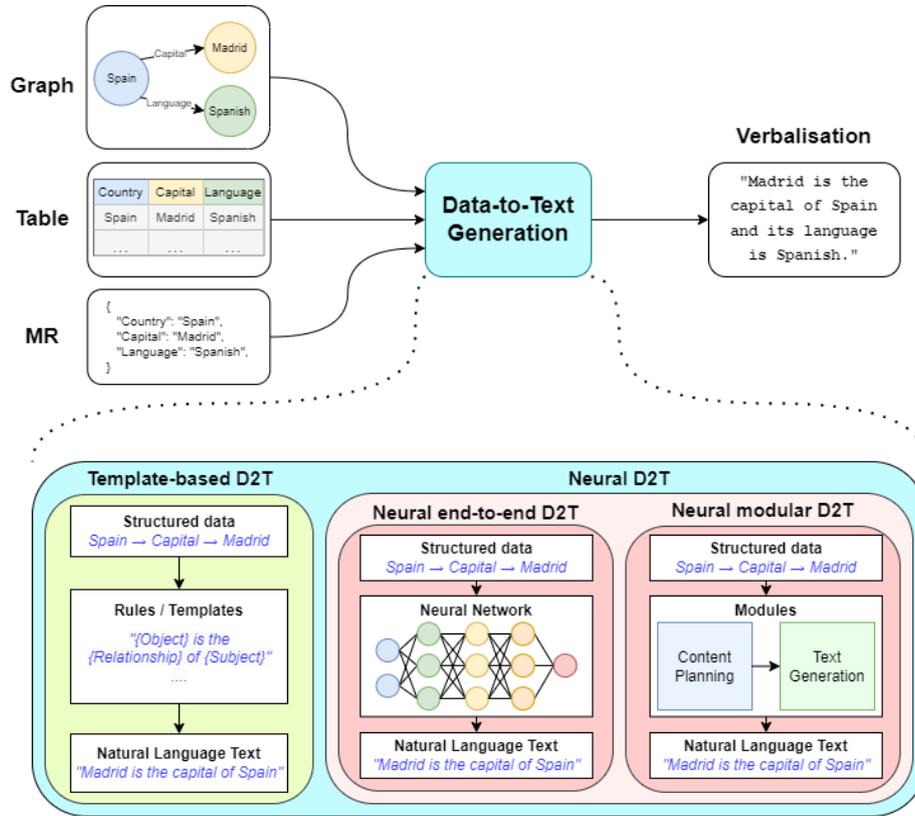


Fig. 1. Data-to-text generation (D2T) overview

2.1. Datasets and Multilingual Coverage for RDF Triple Verbalisation

A lot of work has been done to produce data resources to develop and evaluate data verbalisation systems. In their review of the work presented in the literature related to data-to-text, Chinonso Cynthia Osuji et al. [47] identify 63 distinct data-to-text datasets, mainly for table and AMR data formats. In terms of our specific task, that is RDF-like data verbalisation, specifically triples, they identify four datasets: WebNLG [16], DART [44], T-REx [13] and WITA [15]. WebNLG dataset [16] stems from the WebNLG challenge [9], a shared task to generate text descriptions from structured data found in DBpedia [39]. It consists of pairs of DBpedia triples sets and texts. This dataset was originally presented as an English resource, although with time it has been adapted to other languages and, at present³, it also features Russian, Maltese, Irish, Breton and Welsh [10] verbalisation translations and partial triples translation (only the entities) for Russian, making it the only multilingual dataset of the ones named. Beyond the dataset itself, prior work has briefly addressed Spanish verbalisation in the WebNLG setting. Mille et al. [40] extend the existing rule-based FORGe generator to cover a subset of DBpedia properties in Spanish within a template- and grammar-based framework, but they do not provide a Spanish WebNLG dataset or a parallel benchmark for training and evaluating neural triple-to-text models. This, together with its variety in categories and manual supervision, is the reason we chose WebNLG to adapt to Spanish. In our case, differently from the previous adaptations, we fully translate both the triples and all the available English verbalisations for each entry in all the data splits-train, validation and test-, which results in a fully parallel dataset between English and Spanish. This enables the possibility of not only exploring Spanish triple verbalisation, but also cross-lingual approaches.

³As of February, 2025

1 Similar to WebNLG, DART [44] consists of structured data in the form of RDF triples and their correspond- 1
2 ing textual descriptions. DART is built from multiple sources, including WikiTableQuestions [49], WikiSQL [75], 2
3 WebNLG [16], and E2E [45] datasets, covering diverse domains such as Wikipedia, databases, and e-commerce. 3
4 On the same line, T-REx [13] is built from Wikipedia text and Wikidata triples. It contains millions of automatically 4
5 aligned sentences with structured knowledge, making it useful for pre-training and fine-tuning language models to 5
6 better understand factual knowledge and entity relationships. Lastly, WITA [15] is constructed from Wikipedia ta- 6
7 bles and their associated text, using a distant supervision approach to match table records with textual descriptions. 7
8 Unlike fully supervised datasets, WITA contains partial alignments, reflecting real-world challenges in structured- 8
9 to-text generation. It helps train models to generalise from imperfect data, making it useful for applications requiring 9
10 robust natural language generation from structured inputs. 10
11

12 Regarding the language aspect of the task, as stated in the Introduction, Chinonso Cynthia Osuji et al. [47] re- 12
13 port that, only 15% of the works they reviewed weren't English-exclusive and only 1 of the works [43] focused 13
14 completely on a non-English language, Brazilian Portuguese specifically, while the other 12 focus on multilingual 14
15 approaches. From these 12 articles, only 2 [14, 70] incorporate Spanish to their multilingualism, having that both 15
16 works are based on AMR-formatted data. These numbers, in addition to the fact that the representation of non- 16
17 English languages on well established datasets is quite low, reinforce our belief that more work has to be done in 17
18 the multilingual aspect of the task for a wider spectrum of data formats. 18
19

2.2. Modelling Approaches for RDF Triple-to-Text Generation 21

22
23 As illustrated in figure 1, the approaches proposed in the data-to-text research can be separated into two main 23
24 blocks: rules or template-based approaches and neural approaches. The first block of approaches, template- or rule- 24
25 based data-to-text, relies on predefined templates and handcrafted rules to convert structured data into fixed or 25
26 semi-fixed textual outputs [19, 59]. These systems operate by mapping data inputs to specific linguistic constructs, 26
27 ensuring consistent and contextually appropriate outputs [17]. This approach offers simplicity and control over the 27
28 generated text, making it particularly useful in domains where precision and reliability are essential [18], such as 28
29 weather [4, 55] and triple-like health reporting data [22]. However, the rigidity of templates can limit linguistic 29
30 variability and may require extensive manual effort to create and maintain, especially when scaling across diverse 30
31 domains or languages. Some approaches propose the dynamisation of these templates by adding control expressions 31
32 and or attribute mechanisms to better control the possible inconsistencies during the slot-filling process [38]. De- 32
33 spite these challenges, template-based methods remain a viable solution for applications where the data structure is 33
34 well-understood, and the desired output follows a predictable pattern [66]. 34
35

36 More recently [47], neural approaches, which leverage neural architectures to automate the verbalisation process, 36
37 have taken a more prominent role, moving away from traditional rule-based and modular approaches that require 37
38 handcrafted features and domain-specific templates. Lin et al. [31] propose a taxonomy of neural approaches based 38
39 on two axes: neural end-to-end and modular data-to-text. End-to-end neural models have become the dominant ap- 39
40 proach for structured data verbalisation [32, 52, 64, 67, 72], and more specifically triple verbalisation [8, 33], due to 40
41 their ability to learn complex mappings between structured data and natural language text without requiring explicit 41
42 intermediate steps. These models are generally based on sequence-to-sequence architectures, where an encoder 42
43 processes input triples into a latent representation, and a decoder generates textual descriptions [1, 12]. Various 43
44 other architectures have been introduced to different approaches, such as the use of advanced encoding techniques 44
45 like Graph Neural Networks [34]. Transformer-based architectures, including BERT [11], BART [29], T5 [54], and 45
46 GPT-2 [53], leverage large-scale pretraining and fine-tuning to boost contextual understanding and generalisation 46
47 [35]. Despite their advantages, neural end-to-end models often struggle with weak controllability and factual in- 47
48 consistency, sometimes generating hallucinations in their output not present in the input data [37] or presenting 48
49 semantic errors [26]. To mitigate these issues, modular neural approaches reintroduce intermediate processing steps 49
50 to enhance interpretability and control [42]. A common framework for triple-to-text tasks is the two-stage approach, 50
51 which separates content planning from text generation. The content planning phase determines what information 51

should be included and its structure, with models such as Neural Content Planning (NCP) [74] or a two-step content planning based on an encoder-then-order approach [61]. The text generation phase then transforms the structured content plan into fluent and coherent text while maintaining faithfulness to the input data. Other approaches introduce templates into their neural modular approaches to define initial verbalisation of single triples that will later be used for sentence fusion and scoring based on neural models [25].

2.3. Evaluation of RDF Triple Verbalisation

Evaluation of triple-to-text and, more broadly, data-to-text systems typically relies on human evaluation or automatic reference-based metrics adapted from other Natural Language tasks such as machine translation or summarisation, most prominently BLEU [48], alongside alternatives such as METEOR [28], ROUGE [30], chrF [50] and more recent embedding-based metrics like BERTScore[73], among others [47, 52, 74]. These measures estimate the quality of a system output by comparing it to one or more reference texts, usually in terms of n-gram or token overlap. However, some studies have highlighted important limitations of some reference-based metrics, specifically BLEU, as a proxy for human judgments [5, 51, 58]. They report that BLEU-based system rankings can diverge from human evaluations, that BLEU is often under-specified in practice with inconsistent tokenisation and parameter choices, and that its correlation with real-world utility and user satisfaction is far from guaranteed, particularly for Natural Language Generation tasks. These observations motivate the use of complementary metrics, which we also adopt in our experiments.

In general, the domain of neural data-to-text generation is advancing toward finding an ideal equilibrium among fluency, factual accuracy, and controllability. Although end-to-end methods produce coherent text with little human involvement, modular methods offer enhanced control and dependability, which makes them better suited for applications that demand accuracy. However, a major obstacle persists due to the scarcity of multilingual resources and methodologies, which restricts the accessibility and efficiency of data-to-text systems in various linguistic and cultural settings. Tackling these deficiencies is necessary to create more inclusive and universally applicable solutions.

3. Spanish WebNLG

WebNLG [16] is a structured data-to-text generation benchmark that was initially introduced as an English data-to-text dataset for a challenge [9]. In the last few years it has also been adapted to other languages, specifically Russian, Maltese, Irish, Breton and Welsh [10]. With three major releases, we chose to work with the most recent version, specifically WebNLG v3.0. The dataset is split into training, validation, and test sets, ensuring a standardised evaluation process. It is provided in XML format, where the root element, `<benchmark>`, contains multiple `<entries>`. Each `<entry>` has attributes such as category, ID, shape, shape type, and triple set size. The structure of an entry consists of three key components: `<originaltriple>`, which holds raw RDF triples extracted from DBpedia and wrapped in `<otriple>`; `<modifiedtriple>`, containing the processed original triples which were revised by annotators and wrapped in `<mtriple>`; and `<lex>`, which includes human-generated natural language text (lexicalisations) with quality annotations. The dataset accommodates varying complexity, with each entry containing between one and seven triples. The training set comprises around 13.2k entries with approximately 35.4k triple-lexicalisation pairs, while the validation and test sets contain 1.6k and 1.7k entries, corresponding to around 4.4k and 5.1k triple-lexicalisation pairs, respectively.

WebNLG English v3.0 serves as the foundation for our Spanish adaptation due to its key attributes:

- A well-structured benchmark featuring a diverse set of RDF triples from DBpedia, covering a wide range of topics. The topics are varied yet presented at a sufficiently high level, making them accessible to non-native English speakers with a relatively good command of the language and general knowledge.
- High-quality triples and corresponding verbalisations, carefully verified by annotators and reviewers to maintain accuracy and linguistic clarity.

- A range of complexity levels, with datasets containing between 1 and 7 triples per set, allowing for a thorough evaluation of model performance across different levels of difficulty.
- Demonstrated multilingual adaptability, as evidenced by successful extensions into other languages.
- An active research community and well-established evaluation frameworks that support continuous improvement and benchmarking of generated text.

In Section 3.1, we introduce the methodology followed to adapt the dataset to Spanish.

3.1. Methodology overview

Currently, WebNLG [10] supports a range of languages, but the multilingualism is generally exclusive to the verbalisations, meaning that the original knowledge graph (KG) or triples must strictly be in English, limiting its applicability for multilingual scenarios. To overcome this constraint, we introduce the translation of both triples and verbalisations within WebNLG. This enhancement enables seamless processing in English, Spanish, or a combination of both, expanding the potential for multilingual natural language generation and improving adaptability across different linguistic contexts.

During the adaptation of the dataset, we maintained the original structure of the WebNLG English dataset while translating both the triples and their corresponding verbalisations into Spanish. This ensures consistency in format and allows for direct comparisons between English and translated versions. An overview of this adaptation methodology is shown in Figure 2, which outlines the procedure used to align the translated triples and verbalisations. Specifically, it illustrates the process divided into two steps: the automatic translation phase, explained in Section 3.1.1, and the manual revision and triple composition, explained in Section 3.1.2. In Section 3.2 we introduce the final structure of the dataset and some of its characteristics and limitations.

We acknowledge that the methodology adopted here does not aim to cover the full range of possible Spanish verbalisation variants. Owing to Spanish’ inflectional system and syntactic flexibility, a single structured input may correspond to many equally natural wording variants [2, 41], and generating all such variants would not be feasible. Moreover, Spanish is an official language in over 20 countries across Europe, the Americas, and Africa [7], which entails substantial national and regional variation in lexical and stylistic preferences. Rather than pursuing exhaustive coverage, our goal is to provide a fully aligned Spanish version of WebNLG with broad, practical coverage for benchmarking. We therefore translate all verbalisations available in the original English WebNLG entries to approximate a wide range of Spanish expressions. This initial release primarily reflects a standard Peninsular variety of Spanish, aligned with the authors’ linguistic expertise to support reliable manual revision in this first version. We recognise that Latin American lexical and stylistic variants are under-represented and acknowledge this as a limitation of the present dataset and a clear target for future, community-driven and dialect-aware expansion.

3.1.1. Automatic translation

To adapt WebNLG into Spanish, we first processed the structured data and textual elements and generated automatic machine translations using DeepL⁴. DeepL is a neural machine translation service known for its high-quality machine translation, especially for European languages like Spanish [24].

For triples, which are initially presented in the format *subject | relationship | object*, we separated each component to handle them individually. This decomposition allowed us to translate subjects, relationships, and objects independently, avoiding potential errors from translating entire structured statements at once. We machine-translate all entities and relationships, which may introduce errors due to the lack of context during translation. These potential errors are later addressed through manual revision of all the triple instances. After extraction, given that the same entities and relationships can be present across different triples, we compiled separate sets for entities (subjects and objects) and relationships. Working on these unique sets makes it possible to ensure consistency across the whole

⁴<https://www.deepl.com>

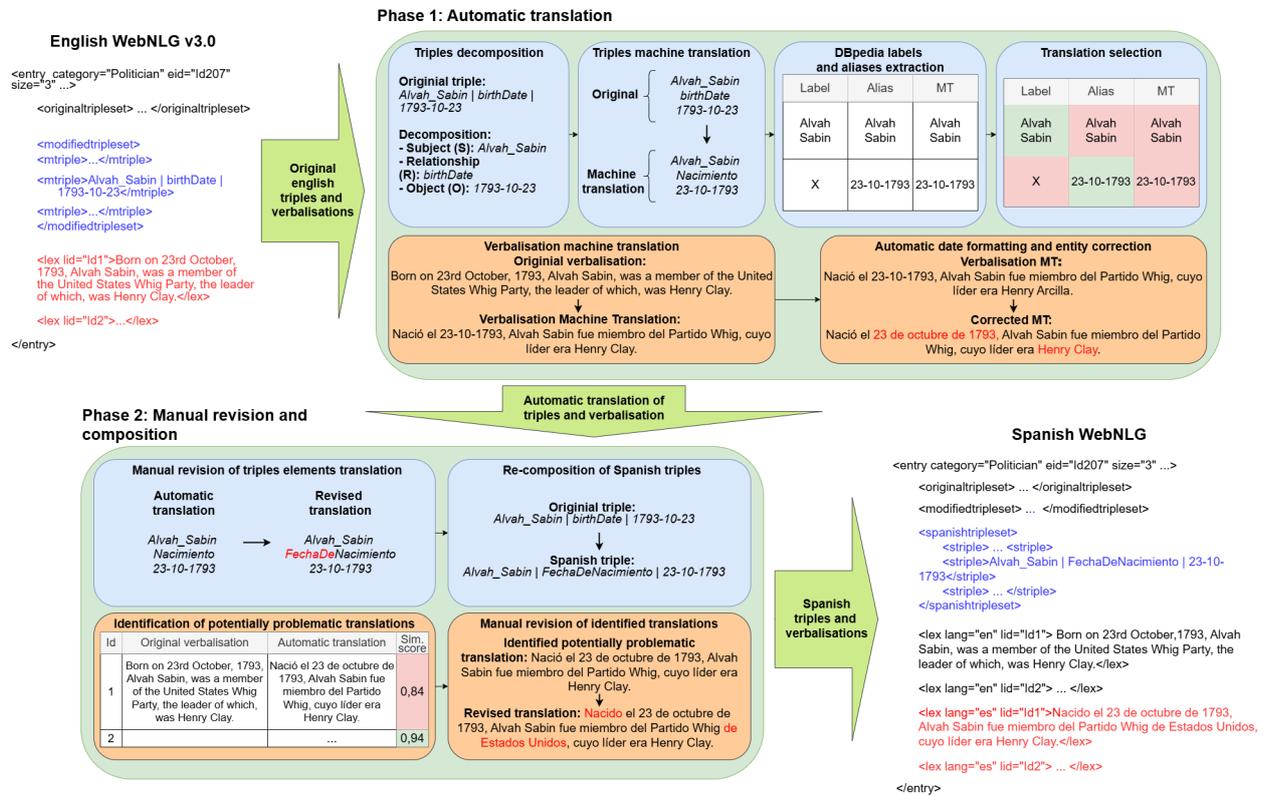


Fig. 2. Methodology followed for the adaptation of WebNLG to Spanish, in blue the process followed for triples adaptation and in red the process followed for verbalisations adaptation.

dataset.

For entities, we combined knowledge-graph (KG) information with machine translation. We queried Wikidata [65] to obtain Spanish labels and/or aliases when available, and only fell back to machine translation when such information was missing. To ensure the best possible translation quality, we applied the following priority order: (i) if a Spanish Wikidata label existed, we selected it as the translation; (ii) otherwise, if an alias was available, we used it; and (iii) if neither was found, we relied on machine translation. In total, this yielded 3,615 unique entities, of which 2,116 were translated using Wikidata labels/aliases and 1,499 via machine translation.

For relationships (properties), we relied exclusively on machine translation. This decision is motivated by the structure of WebNLG v3.0, where the triple sets used for training correspond to manually *modified* triples rather than the raw properties. As a result, there is no simple one-to-one mapping from these modified properties back to canonical properties, which makes property labels from the KG less reliable as a source of truth. Instead, we treated the 412 unique relationships as dataset-specific predicates and translated all of them via machine translation, followed by manual revision (see Section 3.1.2).

For verbalisations (or lexicalisations), which are natural language expressions of the triples, we extracted them as plain text and directly applied machine translation. Since these are full sentences rather than structured elements, they were translated without decomposition, ensuring fluency in the resulting Spanish text. After machine translation, we automatically detected dates in formats such as dd-mm-yyyy, mm/dd/yyyy, etc., and reformatted them into fully textual expressions in Spanish (day, month, and year written out). This avoids misunderstandings related to date formats, such as confusion between day and month order, and matches the way both humans and language

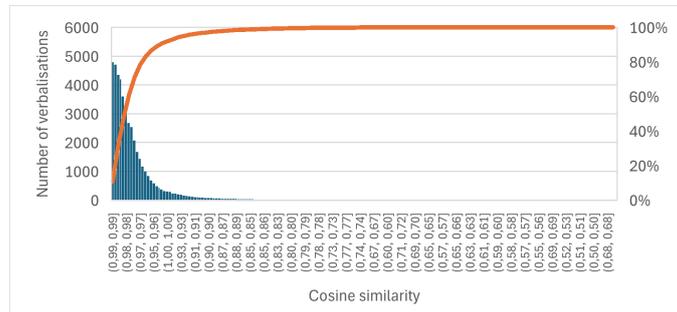


Fig. 3. Histogram of verbalisations similarity results.

models typically realise dates in naturally written text.

Once the translations of entities and relationships had been manually revised (see Section 3.1.2), we performed an automatic consistency check over all verbalisations. In this pass, we verified that (i) the entities mentioned in the Spanish verbalisations matched the final, revised Spanish entities associated with the corresponding triples, and (ii) all dates still conformed to the normalised textual format described above. Whenever mismatches were found (for instance, if a verbalisation still contained a previous version of an entity translation), the verbalisation was automatically updated so that all entity mentions and dates were consistent with the final triple translations.

3.1.2. Detection of problematic cases and manual oversight

The second step of the adaptation was to manually revise the automatic translations. This step was carried out in two stages: first on entities and relationships, and then on a selected subset of verbalisations.

Regarding the triples, with a total of 3,615 unique entities and 412 unique relationships, since the numbers were manageable, we conducted a manual revision of their translations. We reviewed the selected translation for each entity and relationship and corrected any inaccurate or incomplete translations identified. Overall, 301 entities (approximately 8.3% of the total) and 12 relationships (around 2.9%) required modification. Among these 301 corrected entities, 245 corresponded to Wikidata-based translations and 56 to machine-translated ones. Typical errors involved incomplete or partially translated names, such as geographical expressions like *New Jersey*, *New York* being translated only as *Nueva Jersey*, thus omitting relevant contextual information. These cases were manually corrected to ensure accuracy, completeness, and consistency across all triples.

Regarding the verbalisations, the dataset contains 45,031 lexicalisations, which made a complete manual revision of all of them infeasible for this first version of Spanish WebNLG. Instead, we relied on automatic detection of potentially problematic cases, followed by targeted manual revision. Our intention for future work is to enrich the dataset and to apply crowdsourcing to refine these translations at scale.

To identify potentially problematic verbalisations, we computed the cosine similarity between the embeddings of the English and Spanish verbalisations (one-to-one) using three different embedding models. We chose cosine similarity with multilingual embedding models over other metrics because it effectively captures semantic similarities across different languages, enabling accurate cross-linguistic comparison. We chose to use more than one model since the dataset contains a wide variety of topics, so we tried to ensure the representativeness of the models as closely as possible. The models were selected based on the SentenceTransformers [56] documentation on multilingual models⁵ [57]. Specifically, we used the SentenceTransformers models *paraphrase-multilingual-MiniLM-*

⁵https://sbert.net/docs/sentence_transformer/pretrained_models.html#multilingual-models

1 *L12-v2*⁶, *paraphrase-multilingual-mpnet-base-v2*⁷ and *distiluse-base-multilingual-cased-v2*⁸ [56]. For each verbalisation, we took the maximum similarity score across all models. We deliberately use cosine similarity not as a full evaluation metric, but as a risk-based heuristic to prioritise which MT verbalisations should be checked by humans. Concretely, instead of randomly sampling from the 45,031 verbalisations, we rank English–Spanish pairs by their cross-lingual similarity and focus manual effort on those that look atypical in the embedding space. This is conceptually similar to outlier or noise detection in other NLP settings, where multilingual sentence embeddings and distance thresholds are used to flag misaligned or low-quality sentence pairs in large parallel corpora before downstream use [27].

2
3
4
5
6
7
8
9
10 We focused our manual revision on verbalisations whose maximum similarity score was 0.9 or lower. To choose this value, we inspected the empirical distribution of similarity scores and observed a strongly right-skewed shape with a clear “elbow” or “knee” around 0.9, where the density of points drops sharply (Figure 3). Selecting a cut-off at such an elbow is a standard heuristic in data analysis and outlier detection [63], where it marks the transition between a dense region of typical cases and a sparse tail of atypical ones [60]. At the same time, we needed a threshold that was operationally feasible: the selected band had to be sufficiently narrow that a complete manual revision of all retrieved cases was realistic, yet wide enough to cover a meaningful fraction of atypical verbalisations. In our data, a cutoff at 0.9 corresponds to approximately the lowest 3% of the similarity distribution, which we interpret as a high-risk band for potential translation problems. This allows us to concentrate manual effort on the most atypical English–Spanish pairs while keeping the annotation effort realistic. With this criterion, we obtained a selection of 1,239 verbalisations (around 3% of the total) for manual inspection, providing a practical compromise between expected error coverage and annotation cost.

11
12
13
14
15
16
17
18
19
20
21
22
23 It is important to note that this procedure does not guarantee that all problematic translations fall below the threshold, nor that all items below 0.9 are incorrect. Rather, it provides a principled way to concentrate limited human effort on those verbalisations that are statistically more likely to be problematic, instead of relying on uninformed random sampling. Although we cannot manually revise all 45,031 verbalisations, every verbalisation is nevertheless subject to an automatic consistency check: entity mentions are aligned with the final, manually revised entity translations, and dates are enforced to follow the normalised textual format described above. In this way, we at least aim to guarantee global consistency of entities and dates across the dataset, even when the full wording of a verbalisation has not been manually inspected. A full, crowdsourced revision of all verbalisations remains part of our future work.

24
25
26
27
28
29
30
31
32 The manual revision was carried out by a native Spanish speaker with advanced, formally certified proficiency in English. To ensure consistent decisions, the annotator followed a simple set of internal guidelines: a verbalisation was marked as erroneous and corrected if (i) it did not fully preserve the information and context expressed in the original English verbalisation, (ii) it contained clear grammatical errors in Spanish (such as agreement, conjugation, or syntactic well-formedness issues), or (iii) it exhibited inconsistent or clearly inappropriate lexical choices with respect to the intended meaning (such as mistranslations or infelicitous word choice according to the triple sets). Since all manual revision was performed by a single annotator, inter-annotator agreement could not be measured. Instead, we relied on the internal guidelines just described to ensure consistency. Under these criteria, 406 verbalisations were identified as requiring correction, corresponding to 31.5% of the manually inspected subset and less than 1% of all generated verbalisations.

3.2. Spanish WebNLG: statistics, characteristics and limitations

33
34
35
36
37
38
39
40
41
42
43
44
45
46 When creating the Spanish adaptation of WebNLG, we decided to maintain the same structure and content as the English version with the goal of having a parallel translation that enables us to not only compare results along both

47
48
49
50
51
⁶<https://huggingface.co/sentence-transformers/paraphrase-multilingual-MiniLM-L12-v2>

⁷<https://huggingface.co/sentence-transformers/paraphrase-multilingual-mpnet-base-v2>

⁸<https://huggingface.co/sentence-transformers/distiluse-base-multilingual-cased-v2>

languages but potentially create multilingual adaptations. In Table 1, we can observe the statistics about entities and verbalisations in the Spanish adaptation. Specifically, we have a total of 16.657 entries, which are composed of 45.031 triple set-verbalisation pairs. These entries are divided into three splits, following the original grouping of entries, train, validation and test, in a proportion of 80%, 10% and 10%, respectively, which potentially enables the training and evaluation of neural models.

Table 1
Entries and verbalisations statistics of WebNLG datasets

	Train		Validation		Test	
	Entries	Verbalisations	Entries	Verbalisations	Entries	Verbalisations
English	13.211	35.426	1.667	4.455	1.779	5.150
Russian	5.573	14.630	790	2065	1.101	2.780
Breton	13.211	35.426	1.399	1.399	1.778	Not reported
Welsh	13.211	35.426	1.665	1.665	1.778	1.778
Irish	13.211	35.426	1.665	1.665	1.778	1.778
Maltese	13.211	35.426	1.665	1.665	1.778	1.778
Spanish	13.211	35.426	1.667	4.455	1.779	5.150

In terms of category and triple size, in Table 2 we illustrate the distribution of entries per splits for Spanish WebNLG, that mirrors the original English version. The dataset contains a diverse range of 19 categories, including places (*Airport, City, Monument*, etc), people (*Artist, Astronaut, Politician*, etc) and entities (*Company, Film, MusicalWork*, etc), and abstract concepts among others. The number of triples per entry varies significantly, with most entries containing between 1 and 5 triples, though some categories, such as *Astronaut* and *University*, also include larger entries with up to 7 triples. In the train split, categories like *Food, MeanOfTransportation*, and *Politician* have a large number of entries, whereas others such as *Monument* and *Company* have fewer examples. The validation split follows a similar distribution pattern, but with a significantly lower number of examples per category, ensuring a representative yet smaller validation dataset. The test set, however, includes categories absent in training, such as *Film* and *MusicalWork*, which helps testing the model’s generalisation ability. Overall, the dataset balances a variety of domains while varying the complexity of entries by the number of triples, which gives us a wider scope of evaluation.

Table 2
Category and triple size statistics about English and Spanish WebNLG

Split	Num of triples	Train							Validation							Test						
		1	2	3	4	5	6	7	1	2	3	4	5	6	7	1	2	3	4	5	6	7
Triple Category	Airport	301	193	187	206	198	0	0	37	24	24	25	25	0	0	13	22	19	15	14	10	2
	Artist	276	226	246	240	234	0	0	34	28	31	30	30	0	0	22	26	24	20	13	2	2
	Astronaut	71	46	64	82	86	90	90	8	5	8	11	11	11	12	8	18	15	14	11	8	8
	Athlete	285	188	216	147	67	0	0	35	24	27	19	9	0	0	8	12	14	9	7	0	0
	Building	236	171	203	206	156	0	0	30	44	0	25	20	0	0	0	7	8	7	8	8	8
	CelestialBody	169	131	129	118	87	0	0	21	17	16	14	11	0	0	8	9	11	9	7	4	1
	City	243	185	235	218	229	0	0	31	23	30	27	28	0	0	19	16	17	15	7	6	3
	ComicsCharacter	98	77	64	35	11	0	0	13	9	8	5	2	0	0	0	7	8	7	8	0	0
	Company	83	76	76	57	34	16	9	12	11	11	8	5	2	1	15	15	13	11	6	4	2
	Film	0	0	0	0	0	0	0	0	0	0	0	0	0	0	76	41	51	44	31	13	8
	Food	271	277	313	315	230	0	0	34	35	39	40	30	0	0	0	7	8	7	8	8	8
	MeanOfTransportation	298	211	228	242	153	0	0	38	27	28	31	19	0	0	12	15	15	13	2	1	0
	Monument	37	32	41	48	45	35	25	5	4	5	6	5	4	2	0	7	8	7	8	8	8
	MusicalWork	0	0	0	0	0	0	0	0	0	0	0	0	0	0	88	47	58	48	25	16	8
	Politician	299	248	258	243	146	0	0	38	31	32	31	19	0	0	0	6	8	7	8	0	0
	Scientist	0	0	0	0	0	0	0	0	0	0	0	0	0	0	77	52	45	46	27	8	4
	SportsTeam	251	170	169	149	43	0	0	32	22	22	18	5	0	0	0	6	8	7	7	8	8
	University	58	39	58	73	62	62	54	7	5	8	9	7	8	7	10	20	15	14	13	9	9
	WrittenWork	219	202	248	170	98	0	0	28	26	31	21	12	0	0	13	16	5	5	3	1	0
Total entries per triple size		3195	2472	2735	2549	1879	203	178	403	335	320	320	238	25	22	369	349	350	305	213	114	79

Comparing the development process and nature of the data in the other available supported languages in WebNLG with our Spanish dataset, outside of English, the previously existing translations for the other languages were de-

veloped using machine translation and a posterior crowdsourcing process for verbalisations exclusively. In contrast, our Spanish WebNLG dataset includes translations for both triples and verbalisations, which adds more variability and consistency to the dataset.

To obtain the development and test data for each of the low-resource languages (Breton, Irish, Maltese, and Welsh), professional translators manually translated the English text from the WebNLG 2020 [6] development and test sets, given both the English text and the input RDF graph. Only the first reference of each test example in the original English dataset was considered for translation, except in the case of Breton, which contains two translated references for some test items. For Russian, the WebNLG+ 2020 [6] data was used. Although the Russian dataset includes partial triple translation—where only the DBpedia translation of entities was extracted, leaving relationships untranslated—it only contains data for nine of the 19 categories of entries: Airport, Astronaut, Building, CelestialBody, ComicsCharacter, Food, Monument, SportsTeam, and University. In all cases, the training split data is also available, with the verbalisations being generated exclusively through machine translation and thus considered ‘noisy’ data.

In contrast, for Spanish, we translated all triples and verbalisations available in the English version of WebNLG. The translation process began with automatic generation, using Wikidata for triples, and was followed by a posterior manual revision of all triples. Additionally, we detected and manually revised potentially problematic cases for verbalisations across all data splits, further ensuring quality and variability in the dataset. This comprehensive approach ensures the full availability of all translations in the Spanish dataset, unlike the partial triple translation for Russian and the limited translation for the other languages.

Referring again to Table 1, which reports the entry and verbalisation sizes for each split across languages, we can observe that, in terms of size, English and Spanish have notably more available verbalisations for the validation and test splits. This is because the developers report that only the first English verbalisation for each entry was considered for translation into low-resource languages⁹. In our case, we believe that a wider availability of verbalisations can provide a more comprehensive representation of the various forms a text can take to express the triples.

Nevertheless, while Spanish WebNLG is designed to be a broadly useful benchmark, some aspects of its current release delimit its scope. First, the diversity of Spanish realisations is naturally constrained by the paraphrastic space present in the English WebNLG v3.0 dataset: we obtain our range of expressions by translating all existing English triple sets and verbalisations, which we assume are inherently coherent with each other due to their gold standard nature. This means that, although the corpus offers multiple references per triple set and a wider variety than single-reference resources, it should not be interpreted as an exhaustive sample of all possible Spanish verbalisations. Second, although all triples are manually checked and all verbalisations pass through automatic consistency filters, only a similarity-based subset of verbalisations is manually corrected by an annotator, which may leave some residual noise or annotator preference. Finally, this first release primarily reflects a standard Peninsular variety of Spanish, with more limited coverage of other regional varieties. We view a broader, crowdsourced revision and a dialect-aware expansion of the corpus as natural next steps to further strengthen the resource.

4. Evaluation of Triple-to-Text Generation in Spanish using the Spanish WebNLG

The goal of this study is to conduct a preliminary evaluation of resource-efficient Large Language Models (LLMs) for the task of Spanish triple verbalisation using the Spanish WebNLG dataset presented in Section 3. Specifically, we aim to explore how context learning and fine-tuning can enhance the performance of resource-efficient LLMs in generating natural language text from Spanish triples. We aim to answer three key research questions: **(RQ1)** *How effectively can resource-efficient LLMs verbalise Spanish triples across different complexity levels?* (having

⁹For Breton, since the test split data for low-resource languages is not publicly available, and the authors indicate that two verbalisations may have been considered for some test instances, we do not have a specific number.

that complexity levels is defined by the increase of the number of triples in a triple set), **(RQ2)** *How does task contextualisation through examples impact model performance in Spanish triple verbalisation?* and **(RQ3)** *What are the comparative advantages and limitations of prompt learning versus partial fine-tuning for Spanish triple verbalisation?*.

This Section is structured as follows: In Section 4.1, we present the *Learning Approaches and Models* employed in our study. This includes an exploration of Context learning (Section 4.1.1) and its role in model performance, the methodology behind Supervised learning through Fine-tuning (Section 4.1.2), the criteria for Models' selection (Section 4.1.3) and a description of the Evaluation metrics used to assess our models (Section 4.1.4). Section 4.2 provides a comprehensive overview of the *Evaluation Results and Analysis*. We first examine the Context Learning performance (4.2.1) and compare it with the results obtained from Fine-tuning performance (Section 4.2.2). Additionally, we conduct a Cross-lingual Analysis (Section 4.2.3) to evaluate the adaptability of models across different languages. Finally, we perform an Error Analysis (Section 4.2.4) to identify common failure cases and areas for improvement. To conclude, in Section 4.3, we present a *Discussion* of our findings, reflecting on the key takeaways and outlining potential directions for future research.

With this structure, we aim to provide a clear and comprehensive examination of the learning approaches, their effectiveness, and the insights gained from our evaluation.

4.1. Learning Approaches and Models

To answer **RQ1**, *How effectively can resource-efficient LLMs verbalise Spanish triples across different complexity levels?*, we assess the performance of different LLMs on the Spanish triple-to-text task, employing two approaches: **Context-Based Learning** by prompting and **Data-based Learning** by partial fine-tuning, explained in Sections 4.1.1 and 4.1.2, respectively. With the evaluation of these methods, we aim to answer the remaining two research questions: **(RQ2)**, *How does task contextualisation through examples impact model performance in Spanish triple verbalisation?*, and **(RQ3)**, *What are the comparative advantages and limitations of prompt learning versus partial fine-tuning for Spanish triple verbalisation?*. For a fair comparison, both methods are analyzed using the same environment equipped with an RTX 3060 Laptop GPU (6GB VRAM) and 16GB RAM, with the same models (model selection criteria detailed in Section 4.1.3) and evaluated using the performance metrics detailed in Section 4.1.4. The evaluation presented in Section 4.2 has been conducted on the test split of the Spanish WebNLG. Each test has been conducted twice to assess the variability in the results, where variability refers to the performance differences in the generated outputs across runs. This helps capture the range of possible outcomes (or "hallucinations") that each evaluated model may produce.

Even though our goal is to study the performance of the models on the Spanish task, for a better understanding of the results, we also run all the experiments presented before on the English WebNLG set, that is, the context-based learning and data-based learning of the same models. The English results allow us to gain a clearer understanding of the results in Spanish, as they serve as a comparison of the performance of the models in order to have a better understanding of whether the difficulties they may have in carrying out the task are exclusively rooted in the task itself or are also rooted in the language.

4.1.1. Context learning through Prompts

The context learning approach, sometimes referred to as prompt-based learning, enables models to generate responses based on provided prompts without modifying their internal parameters. This approach leverages contextual cues from the input text to guide the model's behavior, allowing it to adapt dynamically to different tasks. It operates under three main settings: zero-shot (0S), one-shot (1S), and few-shot (FS).

- Zero-shot learning (0S): The model generates responses without any prior examples, relying solely on its pre-trained knowledge.
- One-shot learning (1S): The model is given a single example to help it understand the task before generating responses.

- 1 – Few-shot learning (FS): The model is provided with a small number of examples to better grasp patterns and
2 context, improving response accuracy.

3
4 In our case, we will evaluate the context learning approach using zero-shot (0S), one-shot (1S), and few-shot (FS)
5 (with two examples) settings. We use the same base prompt in all the tests and only add examples, in the case of
6 1S and FS tests. Our base prompt introduces the structured data format and instructs the model to generate fluent,
7 grammatically correct Spanish text from given triples:

8 *"En español, los datos estructurados se representan comúnmente como tripletas o triples, con el formato [sujeto,*
9 *predicado, objeto]. A partir de estas tripletas, genera un texto de un solo párrafo formado por oraciones completas,*
10 *gramaticalmente correctas y naturales. Genera el texto únicamente a partir de las siguientes tripletas:"*

11
12 Depending on the modality of the test, that is, if examples are provided, the triples are added to this base
13 prompt with the format [sujeto: subject, predicado: predicate, objeto: object] as a list,
14 followed by their verbalisation.

15
16 As we stated previously, we also run all the experiments on the English WebNLG set using the same configuration
17 of the experiments as the Spanish evaluation. In this case, we translated the original base prompt to English:

18
19 *"In English, structured data is commonly represented as triples, with the format [subject, predicate, object].*
20 *Based on these triples, generate a single-paragraph text composed of complete, grammatically correct, and natural*
21 *sentences. Generate the text solely from the following triples:"*

22
23 Again, depending of the modality of the test, the triples are added to this base prompt with the format
24 [subject: subject, predicate: predicate, object: object] as a list, followed by their ver-
25 balisation.

26
27 The full prompts used to evaluate the models are available in appendix A.

28 29 4.1.2. Data-based learning through Fine-tuning

30
31 In contrast to the previous approach, the supervised learning method we used involves a Partial Fine-Tuning ap-
32 proach, which adapts the model's weights using Low-Rank Adaptation (LoRA) [23]. This method is particularly
33 efficient for fine-tuning large models, as it introduces small, trainable low-rank matrices into the model's architec-
34 ture rather than updating all of the model's parameters. By doing so, LoRA significantly reduces the computational
35 cost and memory requirements, making it feasible to fine-tune large-scale models on limited hardware resources
36 such as ours.

37
38 The fine-tuning process in our case lasts approximately 4-5 hours per model on the same hardware, which is con-
39 siderably faster and more resource-efficient compared to full fine-tuning methods. LoRA achieves this efficiency
40 by freezing the pre-trained model weights and injecting trainable low-rank decomposition matrices into specific
41 layers (in our case, the attention layers of the LLMs). This approach preserves the generalisation capabilities of the
42 pre-trained model and enables a faster adaptation to downstream tasks, such as triple verbalisation, in comparison
43 to the fine-tuning of the whole model.

44
45 During the fine-tuning phase, we employ the same base prompt used in the zero-shot setting of the context learn-
46 ing approach to ensure a fair comparison and evaluate the differences in performance under identical conditions.
47 This consistency allows us to isolate the impact of LoRA-based fine-tuning and demonstrate its effectiveness in
48 improving model performance without extensive computational overhead.

49
50 In this approach we also evaluate the models with the English WebNLG. That means that we fine-tune each model
51 twice, once with each of the following prompts:

- 1 – Spanish base prompt: "En español, los datos estructurados se representan comúnmente como tripletas o triples, 1
2 con el formato [sujeto, predicado, objeto]. A partir de estas tripletas, genera un texto de un solo párrafo 2
3 formado por oraciones completas, gramaticalmente correctas y naturales. Genera el texto únicamente a partir 3
4 de las siguientes tripletas:" 4
- 5 – English base prompt: "In English, structured data is commonly represented as triples, with the format [subject, 5
6 predicate, object]. Based on these triples, generate a single-paragraph text composed of complete, grammati- 6
7 cally correct, and natural sentences. Generate the text solely from the following triples:" 7

8
9 The models are fine-tuned using the train split of WebNLG (both in English and Spanish) as examples and the 9
10 validation split as reference evaluation. The code and the results are available in GitHub¹⁰ and Zenodo¹¹. 10
11

12 4.1.3. Model selection 12

13 To select the large language models to evaluate, we defined four main criteria: 13

- 14 1. *The models must be (non-exclusively) trained on Spanish:* Models must be able to handle data in Spanish to 14
15 ensure high-quality text generation in our task, as language-specific training enhances fluency, coherence, and 15
16 grammatical accuracy. 16
- 17 2. *The models can have up to 2 billion parameters:* Over time, large language models have tended to grow 17
18 larger and more resource-intensive. This means that they often become inaccessible to regular users who 18
19 lack the necessary hardware to run them efficiently, as high-end GPUs and large amounts of memory are 19
20 typically required. At the same time, when dealing with large amounts of data, in our task sometimes involving 20
21 thousands or millions of triple sets, the resources and/or time needed to process them grow exponentially. To 21
22 address this, we focus on resource-efficient models that can run efficiently (in our case, this means that it takes 22
23 a few seconds to compute each answer) on a machine with an RTX 3060 Laptop GPU (6GB VRAM) and 23
24 16GB RAM. This generally limits us to models with up to around 2 billion parameters, ensuring practical 24
25 usability without compromising too much on performance. 25
- 26 3. *The models must be trained for instruction-following tasks:* Instruct LLMs (Instruction-Tuned LLMs) are lan- 26
27 guage models fine-tuned to follow explicit natural language instructions. Unlike generic LLMs, which predict 27
28 text based on training data patterns, these models are optimised to understand and execute user commands. 28
29 They process structured prompts containing instructions, ranging from open-ended queries to specific tasks 29
30 like summarisation, translation, or coding. In our case, to ensure direct and well-structured outputs, the mod- 30
31 els must be trained for instruction-following tasks, eliminating the need for additional post-processing. Our 31
32 goal is to generate responses that are already aligned with the given prompt, minimizing manual adjustments 32
33 or corrections that could affect the evaluation results of model performance. 33
- 34 4. *The models must be available in HuggingFace [68].* 34

35 Given the previous limitations, the test models selected come from two of the large open-source families of LLM, 35
36 Llama 3 [21] and Qwen [62, 71], and Salamandra [20], a family of LLMs created by the Barcelona Supercomput- 36
37 ing Center¹² in Spain which, although it was trained with data from a wide range of European and programming 37
38 languages, was mainly trained with English and Spanish data. Specifically, based on our criteria, we selected the 38
39 following models¹³: 39
40

- 41 – *Qwen2.5-0.5B-Instruct*¹⁴ and *Qwen2.5-1.5B-Instruct*¹⁵ 41
- 42 – *Llama-3.2-1B-Instruct*¹⁶ 42
- 43 – *Salamandra-2b-instruct*¹⁷ 43

44
45 ¹⁰https://github.com/virginia-r99/Spanish_WebNLG_triples-to-text 45

46 ¹¹<https://doi.org/10.5281/zenodo.15064345> 46

47 ¹²<https://www.bsc.es/> 47

48 ¹³Models selected in February, 2025 48

49 ¹⁴<https://huggingface.co/Qwen/Qwen2.5-0.5B-Instruct> 49

50 ¹⁵<https://huggingface.co/Qwen/Qwen2.5-1.5B-Instruct> 50

51 ¹⁶<https://huggingface.co/meta-llama/Llama-3.2-1B-Instruct> 51

¹⁷<https://huggingface.co/BSC-LT/salamandra-2b-instruct>

4.1.4. Evaluation metrics

We evaluate the quality of the generated text using both lexical semantic similarity and efficiency metrics (Table 3), further explained along this Section.

Table 3
Evaluation Metrics Categorised by Type

Category	Metric	Range	Best Value
Lexical Similarity (Information preservation)	BLEU	0 - 1	Higher is better
	METEOR	0 - 1	Higher is better
Lexical Similarity (Linguistic fluency)	CHRF++	0 - 1	Higher is better
Semantic Similarity	Cosine Similarity	0 - 1	Higher is better
	BERTScore (P, R, F1)	0 - 1	Higher is better
Efficiency	Time (s)	0 - ∞	Lower is better

– Lexical similarity metrics

Lexical similarity metrics measure the surface-level overlap between generated and reference texts, focusing on the exact words or n-grams that appear in both texts. The metrics selected for this purpose can be divided between information preservation metrics and language fluency metrics:

* *Information preservation metrics*: These metrics assess how well the generated text retains the content of the reference text by evaluating n-gram overlap and word alignment. These metrics include:

* *BLEU (Bilingual Evaluation Understudy)* [48]: BLEU is a metric that measures the n-gram overlap between generated and reference texts. It's calculated as a weighted geometric mean of n-gram precision, with a brevity penalty to penalise short translations. In our evaluation, we use $n = 4$ sequences. BLEU scores range from 0 to 1, with higher scores indicating closer matches. This metric is sensitive to exact word matches and does not account for synonyms or paraphrasing, making it less effective in evaluating texts with flexible word choices. However, as discussed in section 2.3, BLEU is sensitive to exact word overlap and has known limitations in natural language generation, including weak correlations with human judgements in some settings [36, 58]. We therefore report BLEU alongside additional lexical and semantic similarity metrics.

* *METEOR (Metric for Evaluation of Translation with Explicit Ordering)* [28]: METEOR improves upon BLEU by incorporating precision, recall, stemming, synonym matching, and word order penalties. Unlike BLEU, METEOR aligns words between the generated and reference texts and computes a harmonic mean of precision and recall. METEOR scores range from 0 to 1, with higher values indicating better alignment between the predicted and reference sentences. This metric is useful for evaluating languages with rich morphology, such as Spanish, where word forms can vary significantly.

* *Language fluency metrics*: These metrics assess the readability and naturalness of generated text by analyzing character-level and subword-level coherence. These metrics include:

* *CHRF++ (Character n-gram F-score, extended version)* [50]: CHRF++ calculates an F-score based on the overlap of character n-grams between the reference and generated texts. In our evaluation, we set $n = 6$. The CHRF++ score ranges from 0 to 1, with higher values indicating better similarity. Since this metric operates at the character level, it is more robust to minor spelling variations and inflectional changes compared to BLEU.

– Semantic similarity metrics

In contrast to previous metrics, semantic similarity metrics focus on meaning rather than exact matches. These metrics compare the underlying meaning or context of the sentences, making them more robust to synonyms,

paraphrasing, or other linguistic variations that don't affect the underlying content. These metrics include, among others:

- * *Cosine Similarity*: This metric computes the cosine similarity between sentence embeddings, which are high-dimensional vector representations of the sentences. Sentence embeddings capture semantic meaning beyond exact word overlap. The value ranges from 0 to 1, where 1 indicates identical meaning. For our experiments we selected the sentence model *paraphrase-multilingual-MiniLM-L12-v2*¹⁸ [56], given that it is currently the most downloaded sentence transformers model with Spanish support¹⁹. We deliberately opt for a multilingual encoder rather than a Spanish-only sentence model to keep the embedding space consistent across our Spanish and English evaluations: both languages are embedded into a shared semantic space, enabling direct cross-lingual comparison. Since our goal is not to benchmark embedding models, we leave a systematic study of other encoders such as monolingual Spanish encoders for future work.

- * *BERTScore* [73]: BERTScore computes contextual token-level similarities using embeddings from a pre-trained BERT model. BERTScore considers precision, recall, and the F1-score for evaluating the match between the predicted and reference sentences at the semantic level taking values between 0 and 1, where the higher the value the better. For BERTScore, we use the HuggingFace [68] "evaluate" wrapping of the metric implementation [73], which in our case corresponds to the *bert-base-multilingual-cased* model [11] with the default layer configuration.

– *Efficiency metric*:

Efficiency metrics measure the computational cost of generating text. In our case, the efficiency metric chosen is the following:

- * *Time*: This metric measures the duration required to generate a text output in seconds, with lower values indicating more efficient text generation.

4.2. Evaluation Results and Analysis

This Section is structured to assess the performance of the models in both context learning and fine-tuning scenarios, with a primary focus on the Spanish evaluation, based on the results presented in Table 4 (*for the full results of the Spanish evaluation, see Appendix B*), which present the average results of the Spanish evaluation across all tests. First, we analyse the model's ability to perform tasks using context learning in Section 4.2.1. Next, in Section 4.2.2, we present the results of fine-tuning on the Spanish dataset, comparing its performance to context learning and examining its efficiency and effectiveness. We then conduct a cross-lingual analysis to explore how the models' performance differs between English and Spanish, in Section 4.2.3 (*for the full results of the English evaluation, see Appendix C*). Finally, in Section 4.2.4, we perform an error analysis to try and understand the types of errors made by the model in Spanish.

4.2.1. Context learning Performance

The results of our experiments on Spanish triple generation using a context learning approach (Table 4 and Figure 4) reveal several insights. First, we observe a significant improvement in performance from zero-shot to one-shot learning, with only a slight improvement from one-shot to few-shot learning. This suggests that providing even a single example significantly enhances the model's understanding of the task, likely due to better contextualisation. Notably, this improvement is not only qualitative but also time-efficient: one-shot and few-shot settings generally require less time for verbalisation compared to zero-shot. This efficiency gain could be attributed to the model generating more precise outputs with less non-relevant information, as zero-shot often produced additional content outside the scope of the triples, which also results in nearly double the generation time in some cases. For

¹⁸<https://huggingface.co/sentence-transformers/paraphrase-multilingual-MiniLM-L12-v2>

¹⁹As of February, 2025

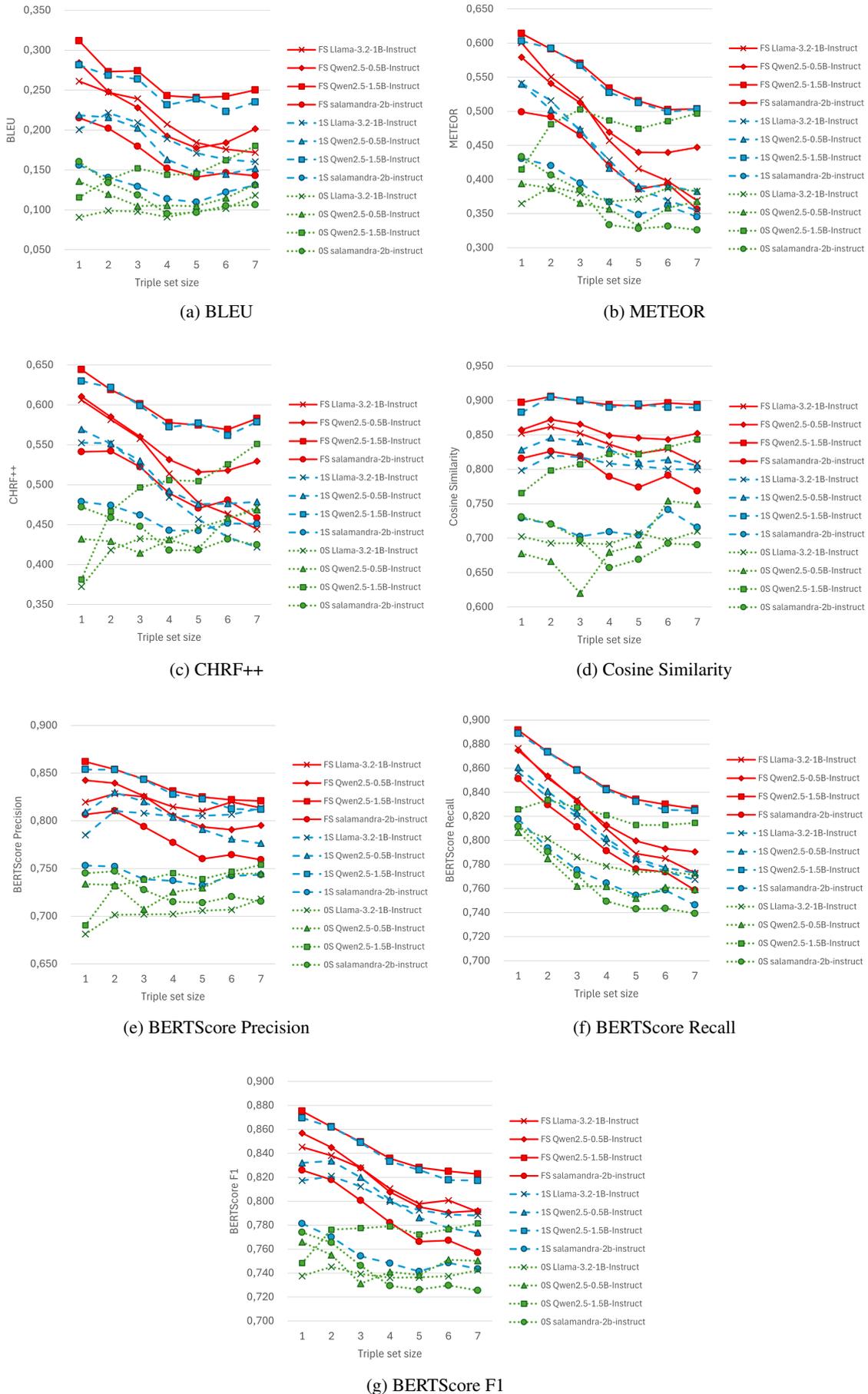


Fig. 4. Average Spanish context learning results by triple set size

Table 4
Spanish evaluation results

Eval. modality	Model	BLEU	METEOR	CHRF++	Cosine Similarity	BERTScore Precision	BERTScore Recall	BERTScore F1	Average time per triple set (s)
Zero	Qwen2.5-0.5B-Instruct	0,097 ± 0,7%	0,376 ± 0,1%	0,422 ± 0,6%	0,697 ± 0,3%	0,699 ± 0,3%	0,790 ± 0,1%	0,739 ± 0,2%	1,911 ± 0,5%
	Llama-3.2-1B-Instruct	0,116 ± 1,2%	0,370 ± 0,5%	0,430 ± 0,4%	0,674 ± 0,2%	0,728 ± 0,1%	0,774 ± 0,1%	0,748 ± 0,1%	1,955 ± 0,6%
	Qwen2.5-1.5B-Instruct	0,141 ± 0,8%	0,473 ± 0,2%	0,474 ± 0,5%	0,804 ± 0,5%	0,729 ± 0,2%	0,824 ± 0,0%	0,771 ± 0,1%	3,129 ± 1,4%
	salamandra-2b-instruct	0,122 ± 0,9%	0,377 ± 0,1%	0,444 ± 0,2%	0,698 ± 0,2%	0,730 ± 0,1%	0,773 ± 0,1%	0,749 ± 0,1%	1,488 ± 0,5%
One	Qwen2.5-0.5B-Instruct	0,197 ± 1,0%	0,465 ± 0,8%	0,510 ± 0,3%	0,809 ± 0,1%	0,803 ± 0,0%	0,817 ± 0,1%	0,808 ± 0,0%	0,787 ± 0,8%
	Llama-3.2-1B-Instruct	0,189 ± 1,4%	0,464 ± 0,5%	0,524 ± 0,6%	0,830 ± 0,6%	0,809 ± 0,1%	0,821 ± 0,2%	0,813 ± 0,1%	1,174 ± 2,1%
	Qwen2.5-1.5B-Instruct	0,256 ± 0,5%	0,559 ± 0,2%	0,599 ± 0,1%	0,894 ± 0,0%	0,839 ± 0,1%	0,858 ± 0,0%	0,847 ± 0,0%	1,454 ± 0,7%
	salamandra-2b-instruct	0,132 ± 0,8%	0,393 ± 0,2%	0,461 ± 0,0%	0,716 ± 0,6%	0,744 ± 0,0%	0,781 ± 0,0%	0,760 ± 0,0%	1,406 ± 0,8%
Few	Qwen2.5-0.5B-Instruct	0,226 ± 0,1%	0,504 ± 0,0%	0,544 ± 0,0%	0,844 ± 0,2%	0,820 ± 0,0%	0,831 ± 0,0%	0,823 ± 0,0%	0,794 ± 0,9%
	Llama-3.2-1B-Instruct	0,228 ± 0,4%	0,508 ± 0,1%	0,561 ± 0,2%	0,858 ± 0,2%	0,821 ± 0,1%	0,834 ± 0,0%	0,826 ± 0,1%	1,210 ± 0,6%
	Qwen2.5-1.5B-Instruct	0,269 ± 0,1%	0,563 ± 0,2%	0,604 ± 0,0%	0,898 ± 0,0%	0,843 ± 0,0%	0,860 ± 0,0%	0,850 ± 0,0%	1,426 ± 0,4%
	salamandra-2b-instruct	0,178 ± 0,6%	0,451 ± 0,3%	0,513 ± 0,1%	0,805 ± 0,2%	0,789 ± 0,0%	0,811 ± 0,1%	0,798 ± 0,1%	1,184 ± 1,7%
Fine-tuned	Qwen2.5-0.5B-Instruct	0,310 ± 0,4%	0,601 ± 0,4%	0,631 ± 0,3%	0,898 ± 0,1%	0,856 ± 0,1%	0,860 ± 0,1%	0,857 ± 0,1%	1,126 ± 0,8%
	Llama-3.2-1B-Instruct	0,141 ± 0,0%	0,475 ± 0,0%	0,435 ± 0,1%	0,848 ± 0,0%	0,665 ± 0,0%	0,782 ± 0,0%	0,713 ± 0,0%	3,565 ± 0,1%
	Qwen2.5-1.5B-Instruct	0,348 ± 0,6%	0,640 ± 0,0%	0,660 ± 0,2%	0,916 ± 0,0%	0,870 ± 0,0%	0,873 ± 0,0%	0,870 ± 0,0%	1,310 ± 0,1%
	salamandra-2b-instruct	0,279 ± 0,9%	0,562 ± 0,4%	0,606 ± 0,1%	0,896 ± 0,0%	0,853 ± 0,1%	0,851 ± 0,1%	0,851 ± 0,1%	0,909 ± 0,0%

example, when given the triple *[sujeto: MotorSport_Vision, predicado: Ciudad, objeto: Fawkham]*²⁰ in OS setting, Qwen2.5-1.5B-Instruct generated "Aunque no tengo información adicional sobre la ciudad Fawkham, puedo generar un párrafo basado en las tripletas proporcionadas: La empresa MotorSport_Vision es conocida por su apasionante interés en documentar eventos deportivos de alta categoría, incluyendo carreras de coches y motocicletas. Sus reportajes son famosos por capturar momentos inolvidables durante competiciones globales, desde la Gran Premi d'Alanya hasta la Copa del Mundo de Motocross. Una de sus últimas iniciativas destacadas fue la producción de una serie exclusiva que se centró en la historia y evolución de la carrera de rallyes en Fawkham, uno de los circuitos más emblemáticos de Europa. Este proyecto ha sido especialmente fascinante para el equipo de MotorSport_Vision, ya que ofrece una perspectiva única sobre cómo han cambiado las reglas y los patrones de la industria de la velocidad en la región."²¹, in contrast to FS setting, which generated "La empresa MotorSport Vision tiene su sede en Fawkham."²².

In terms of evaluation metrics, BLEU scores are notably low, with a maximum of 0.27 (on a scale of 1). As discussed in Sections 2.3 and 4.1.4, this behaviour is consistent with known limitations of BLEU, which relies on strict lexical n-gram overlap and tends to under-reward valid paraphrases and alternative surface realisations. In contrast, other lexical metrics such as METEOR and CHRF++ yield more favourable results, ranging between 0.47 and 0.6, as they are more tolerant to morphological variation and partial lexical matches. These scores suggest that, despite lexical differences from the references, the generated outputs capture more of the intended content than would be indicated by BLEU alone. Additionally, similarity-based metrics provide further evidence of the model's effectiveness. Cosine similarity increases from 0.8 in zero-shot to nearly 0.9 in one-shot and few-shot settings, indicating that the generated content is contextually closer to the reference text when examples are provided. Similarly, BERTScore metrics show consistent improvements: BERTScore recall remains stable (0.82 to 0.86), while precision rises from 0.73 in zero-shot to around 0.84 in one-shot and few-shot, and F1 score improves from 0.77 to 0.85. These results suggest that, while zero-shot outputs may include unnecessary information, one-shot and few-shot settings produce

²⁰Translation: *[subject: MotorSport_Vision, predicate: City, object: Fawkham]*

²¹Transalction: *"While I have no additional information on the city of Fawkham, I can generate a paragraph based on the triplets provided: MotorSport_Vision is known for its passionate interest in documenting top-level sporting events, including car and motorcycle racing. Their reports are renowned for capturing unforgettable moments during global competitions, from the Alanya Grand Premi to the Motocross World Cup. One of their recent notable initiatives was the production of an exclusive series that focused on the history and evolution of rally racing at Fawkham, one of Europe's most iconic circuits. This project has been particularly fascinating for the MotorSport_Vision team as it offers a unique insight into how the rules and patterns of the speed industry have changed in the region."*

²²Translation: *"MotorSport Vision is based in Fawkham."*

1 more focused and contextually aligned responses.

2
3 At the same time, if we observe the graphics from figure 4, where the results can be seen according to triple
4 size, we can see that, in general, the performance tends to drop with the increase of the number of triples. It can
5 also be seen that, in general, 1S and FS (marked in blue and red in the figures) perform better than OS (marked in
6 green). Albeit, if we observe these results more deeply, we can see that for the OS tests, BLEU, CHRF++, cosine
7 similarity and BERTScore Precision, this trend is contrary; the more triples, the better the performance. This could
8 probably be explained with the fact presented before about OS generation, where models often produced additional
9 content outside the scope of the triples. Logically, the more triples a set has, the longer the verbalisation has to be.
10 If the model tends to create larger and richer answers when no examples are given, it could be logical to think that
11 the longer the ground truth, the higher the probability that the evaluation finds more similarity among the reference
12 and generated texts. Moreover, despite our efforts to explain what triples are and clarify our goal, it's important to
13 recognise that we are introducing structured data to models that were generally not trained to handle such informa-
14 tion. These models may lack domain-specific knowledge, making the prompt potentially insufficiently descriptive
15 without examples. Consequently, this lack of context could affect the quality and accuracy of the verbalisation, par-
16 ticularly in cases where structured data is involved.

17
18 Among the models evaluated, Qwen2.5-1.5B-Instruct consistently outperforms the others across all tests. Al-
19 though less time-efficient, its competitive performance in one-shot and few-shot settings highlights its robustness
20 for Spanish triple generation. Surprisingly, Salamandra-2B-Instruct, which we expected to be among the better-
21 performing models due to its Spanish-centric training (Spanish being the second most prominent language in its
22 training data) and development by a Spanish institution, underperforms relative to the other models. This suggests
23 that factors beyond language-specific training, such as task-specific fine-tuning, may play a more critical role in its
24 performance.

25
26 Taken together, these findings highlight the importance of contextualisation in helping models correctly interpret
27 structured inputs such as triples, as well as the limitations of relying on a single evaluation metric. As discussed
28 earlier, lexical-overlap measures alone may fail to reflect improvements in meaning preservation when multiple
29 valid verbalised forms are possible, a limitation widely noted in natural language generation research [36, 58]. By
30 contrast, considering both lexical and similarity-based metrics offers a clearer view of model behaviour, showing
31 that although lexical overlap may remain limited, semantic and contextual alignment improves noticeably in one-
32 shot and few-shot settings.

34 4.2.2. Fine-tuning Performance

35 As explained in Section 4.1.2, we apply a LoRA fine-tuning to the models with our dataset, having the configura-
36 tion of the LoRA training being the same for each model. LoRA is applied to attention layers because they are the
37 most critical for learning contextual relationships while significantly reducing the number of trainable parameters,
38 making fine-tuning more efficient. In Table 5, we can see the parameter configuration of each model. Generally, we
39 only fine-tune between 0.2% and 0.4% of the parameters. It is notable that Salamandra-2B-Instruct has a smaller
40 proportion of trainable parameters despite having more total parameters. Here, we have to take into account that the
41 number of trainable parameters depends on how many attention layers the model has and their size, rather than the
42 total parameter count. A possible explanation is that Salamandra-2B-Instruct might have fewer or smaller attention
43 layers relative to its total size, meaning LoRA modifies a smaller portion of the model. Other architectures, like
44 Qwen or Llama, might distribute their parameters differently, with a larger fraction dedicated to attention layers,
45 leading to more trainable parameters under the same LoRA configuration.

46
47 In Table 4, we can see that when transitioning from few-shot (FS) learning to fine-tuning, we observe notable
48 improvements across all metrics for Qwen2.5-0.5B-Instruct, Qwen2.5-1.5B-Instruct, and Salamandra-2B-Instruct,
49 with the exception of time efficiency. Specifically, Qwen2.5-0.5B-Instruct experiences a slight degradation in time
50 efficiency, while Qwen2.5-1.5B-Instruct and Salamandra-2B-Instruct show improvements. This suggests that fine-
51 tuning enhances the models' ability to generate more accurate and contextually appropriate text, albeit at a potential

Table 5
Models' parameters fine-tune configuration

Model	Parameters		
	Total	Trainable	Non-trainable
Qwen2.5-0.5B-Instruct	496.232.320	2.199.552	494.032.768
Llama-3.2-1B-Instruct	1.238.632.448	2.818.048	1.235.814.400
Qwen2.5-1.5B-Instruct	1.548.330.496	4.616.192	1.543.714.304
Salamandra-2B-Instruct	2.253.490.176	3.729.408	2.249.760.768

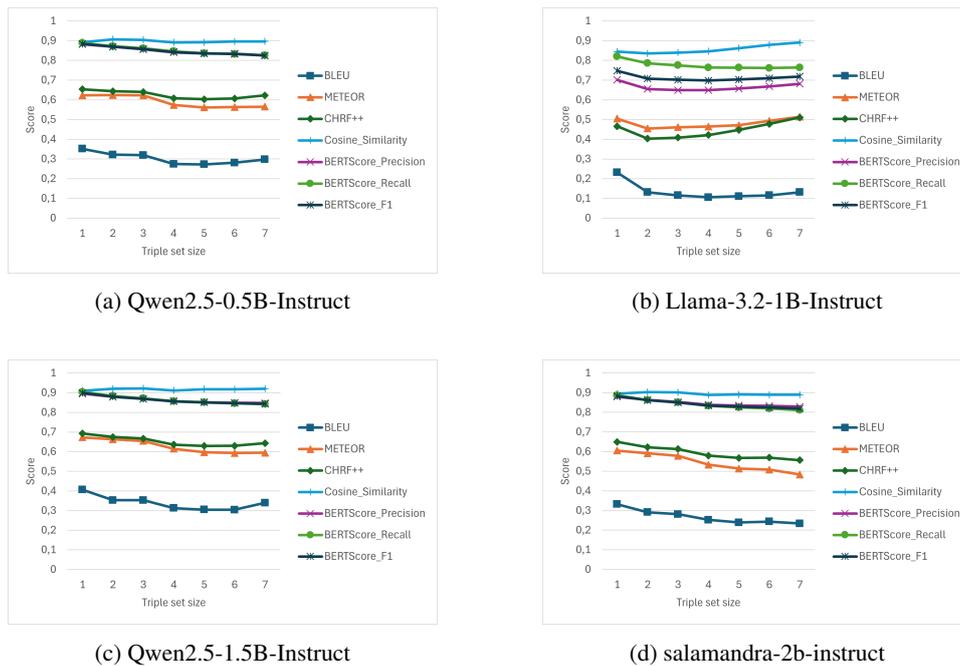


Fig. 5. Average Spanish fine-tuning results by triple set size

cost in computational speed for some models.

As illustrated in Figure 5, fine-tuned models exhibit greater stability in performance as the size of the triple set increases, compared to the variability seen in zero-shot (0S), one-shot (1S), and few-shot (FS) settings. However, Llama-3.2-1B-Instruct stands out as an exception, showing a significant drop in performance relative to its FS results. This indicates that Llama-3.2-1B-Instruct may not have adapted effectively during fine-tuning, potentially due to limitations in its architecture or training dynamics. Additionally, while Qwen2.5-1.5B-Instruct and Salamandra-2B-Instruct improve in computation time, Qwen2.5-0.5B-Instruct and Llama-3.2-1B-Instruct experience degradation, with the latter taking more than double the time to generate text compared to its context learning performance.

Overall, these results demonstrate that even a small-scale fine-tuning process, such as the one employed in this study, can significantly enhance the performance of models for Spanish data-to-text generation. Fine-tuning not only improves the stability and quality of the generated text but also highlights the importance of model-specific adaptability. While some models, like Qwen2.5-1.5B-Instruct, show rather good results in both performance and efficiency after fine-tuning, others, like Llama-3.2-1B-Instruct, may require further optimisation to achieve comparable results. These findings underscore the value of fine-tuning as a practical approach for improving multilingual models, particularly for underrepresented languages.

4.2.3. Cross-lingual Analysis

Table 6
English evaluation results

Eval. modality	Model	BLEU	METEOR	CHRF++	Cosine Similarity	BERTScore Precision	BERTScore Recall	BERTScore F1	Average time per triple set (s)
Zero	Qwen2.5-0.5B-Instruct	0,160 ± 0,1%	0,528 ± 0,0%	0,528 ± 0,0%	0,841 ± 0,1%	0,896 ± 0,0%	0,924 ± 0,0%	0,909 ± 0,0%	1,466 ± 5,6%
	Qwen2.5-1.5B-Instruct	0,245 ± 0,8%	0,635 ± 0,1%	0,605 ± 0,2%	0,898 ± 0,0%	0,911 ± 0,0%	0,942 ± 0,0%	0,925 ± 0,0%	1,868 ± 0,2%
	salamandra-2b-instruct	0,127 ± 2,1%	0,460 ± 0,3%	0,476 ± 0,2%	0,770 ± 0,2%	0,872 ± 0,0%	0,904 ± 0,0%	0,887 ± 0,0%	1,885 ± 2,9%
One	Qwen2.5-0.5B-Instruct	0,213 ± 0,3%	0,586 ± 0,2%	0,584 ± 0,0%	0,870 ± 0,1%	0,914 ± 0,0%	0,930 ± 0,0%	0,921 ± 0,0%	1,177 ± 4,1%
	Llama-3.2-1B-Instruct	0,260 ± 0,2%	0,584 ± 0,2%	0,584 ± 0,2%	0,873 ± 0,2%	0,929 ± 0,0%	0,924 ± 0,0%	0,925 ± 0,0%	0,611 ± 3,0%
	Qwen2.5-1.5B-Instruct	0,365 ± 0,6%	0,708 ± 0,1%	0,699 ± 0,0%	0,939 ± 0,0%	0,945 ± 0,0%	0,949 ± 0,0%	0,946 ± 0,0%	1,172 ± 1,1%
Few	salamandra-2b-instruct	0,161 ± 0,5%	0,513 ± 0,4%	0,531 ± 0,4%	0,803 ± 0,3%	0,891 ± 0,0%	0,912 ± 0,1%	0,900 ± 0,0%	1,406 ± 2,0%
	Qwen2.5-0.5B-Instruct	0,256 ± 0,5%	0,623 ± 0,1%	0,622 ± 0,2%	0,892 ± 0,2%	0,922 ± 0,0%	0,935 ± 0,0%	0,928 ± 0,0%	1,109 ± 5,7%
	Llama-3.2-1B-Instruct	0,293 ± 0,3%	0,611 ± 0,1%	0,611 ± 0,3%	0,894 ± 0,1%	0,938 ± 0,0%	0,927 ± 0,0%	0,932 ± 0,0%	0,582 ± 2,0%
Fine-tuned	Qwen2.5-1.5B-Instruct	0,358 ± 0,3%	0,700 ± 0,0%	0,696 ± 0,0%	0,940 ± 0,0%	0,945 ± 0,0%	0,948 ± 0,0%	0,946 ± 0,0%	1,175 ± 0,8%
	salamandra-2b-instruct	0,185 ± 0,2%	0,522 ± 0,6%	0,550 ± 0,5%	0,824 ± 0,5%	0,900 ± 0,1%	0,915 ± 0,1%	0,906 ± 0,1%	1,248 ± 2,2%
	Qwen2.5-0.5B-Instruct	0,373 ± 0,9%	0,707 ± 0,0%	0,709 ± 0,1%	0,941 ± 0,0%	0,946 ± 0,0%	0,945 ± 0,0%	0,945 ± 0,0%	0,909 ± 1,4%
Fine-tuned	Llama-3.2-1B-Instruct	0,417 ± 0,1%	0,743 ± 0,1%	0,738 ± 0,0%	0,951 ± 0,0%	0,950 ± 0,0%	0,952 ± 0,0%	0,950 ± 0,0%	0,679 ± 0,8%
	Qwen2.5-1.5B-Instruct	0,379 ± 0,6%	0,722 ± 0,3%	0,721 ± 0,2%	0,950 ± 0,1%	0,947 ± 0,0%	0,949 ± 0,0%	0,947 ± 0,0%	1,100 ± 0,4%
	salamandra-2b-instruct	0,301 ± 0,3%	0,655 ± 0,2%	0,671 ± 0,2%	0,928 ± 0,2%	0,933 ± 0,0%	0,936 ± 0,0%	0,934 ± 0,0%	0,919 ± 0,0%

As stated in the methodology of the evaluation, we also computed the same tests as with the Spanish WebNLG using English WebNLG. Table 6 shows the results we obtained during this evaluation for all the models for zero-shot, one-shot and few-shot setting.

When comparing the results for English and Spanish, it is evident that the models generally perform better with English input than with Spanish. This is expected, given that English constitutes a significant portion of the training data for most multilingual models, making them inherently more proficient in English. However, the performance gap also highlights the challenges of adapting these models to languages like Spanish, which, despite being widely spoken, may not receive the same level of representation in training datasets.

Across all context learning settings — zero-shot (0S), one-shot (1S), and few-shot (FS)— the Qwen2.5-1.5B-Instruct model consistently comes up as the best performer. This suggests that Qwen2.5-1.5B-Instruct may possess some inherent capability to generate text from triples, regardless of the input language. Its robustness in both English and Spanish settings underscores its versatility and effectiveness for structured text generation tasks, even in low-resource or zero-shot scenarios.

Interestingly, the Llama-3.2-1B-Instruct model presents a contrasting case. In the fine-tuning experiments, Llama-3.2-1B-Instruct struggled to adapt effectively to the Spanish triple verbalisation task. However, in the English context learning results, it performs competitively, achieving the best results across all metrics including time efficiency. This discrepancy suggests that the issue may lie not in the task itself but rather in the model’s ability to learn and generalise the task specifically for Spanish. This could be attributed to differences in linguistic characteristics, training data distribution, or the model’s architectural limitations when handling Spanish compared to English.

4.2.4. Error Analysis

To interpret some of the errors observed in our evaluation, it is useful to recall that English and Spanish encode grammatical information differently. English generally relies more on word order and has comparatively limited overt verbal inflection, whereas Spanish marks person and number more systematically on verbs and shows gender and number agreement within the noun phrase [41]. For example, in English the verb *eat* shows relatively limited inflection across subjects (“I eat”, “you eat”, “she eats”, “we eat”), whereas in Spanish the verb exhibits richer inflectional variation (“yo como”, “tú comes”, “ella come”, “nosotros comemos”). Spanish also allows null (silent)

Table 7
Predictions samples and their performance²³.

Model	Test modality	Triples	Ground truth	Prediction	Metrics						
					B	M	C	CS	BP	BR	BF1
Qwen2.5-1.5B-Instruct	Few-Shot	Triple 1: [Subject: MotorSport_Vision, Predicate: Ciudad, Object: Fawkham]	"MotorSport Vision se encuentra en la ciudad de Fawkham." "MotorSport Vision se encuentra en la ciudad de Fawkham, Reino Unido." "MotorSport Vision se ubica en Fawkham."	"La empresa MotorSport Vision tiene su sede en Fawkham."	0,14	0,60	0,63	0,97	0,91	0,94	0,92
Salamandra-2B-Instruct	Zero-shot	Triple 1: [Subject: MotorSport_Vision, Predicate: Ciudad, Object: Fawkham]	"MotorSport Vision se encuentra en la ciudad de Fawkham." "MotorSport Vision se encuentra en la ciudad de Fawkham, Reino Unido." "MotorSport Vision se ubica en Fawkham."	"La ciudad Fawkham atrae cada año a miles de visitantes que disfrutan del hermoso paisaje natural y sus monumentos históricos. Los sitios más populares incluyen el Parque Nacional Mount Abundance, conocido por su gran variedad de flora y fauna; además, cuenta con una amplia gama de actividades recreativas para todas las edades. Para aquellos interesados en la historia local, no pueden perderse el Museo Histórico de Fawkham, donde podrán descubrir detalles interesantes sobre la vida cotidiana durante siglos pasados. Por lo tanto, si estás buscando lugares fascinantes cerca de Londres pero quieres evitar grandes multitudes, definitivamente deberías visitar Fawkham; te garantizo que no te arrepentirás!"	0,01	0,1	0,18	0,48	0,57	0,73	0,64

subjects, since subject features are often recoverable from the verb [46]. In addition, Spanish permits greater flexibility in the placement of subjects and objects than English, which further expands the set of grammatical realisations available for a given triple [2]. Taken together, these properties increase the likelihood that a faithful Spanish output may differ in surface form from a single reference. This is why we interpret lexical overlap metrics alongside more meaning-sensitive measures, in line with prior work showing that automatic metrics capture complementary aspects of NLG quality and are best used in combination [36, 58].

In Table 7, we can see a good example of the importance of variety in evaluation metrics²³. In the first row, we see that Qwen2.5-1.5B-Instruct predicted "La empresa MotorSport Vision tiene su sede en Fawkham." ("*MotorSport Vision is based in Fawkham.*") as the verbalisation, which generally would be correct, as, in this case "based" can be expressed as "se ubica", "está", "se encuentra", "tiene su base en", etc. If we observe the lexical results only, we can see that the BLEU value is 0.14, which is a very low performance, while METEOR and CHRF++ have a score of 0.60 and 0.63 respectively, which is not nearly as low as BLEU, but it is also not that high. On the other hand, we have that the semantic metrics, Cosine Similarity and the BERTScore scores are all above 0.91, which means that, semantically, the answer is very close to the ground truth.

On the other hand, in the second row of Table 7, we can see a case contrary to the previous one. We can see that, for the same triple as the previous instance, Salamandra-2B-Instruct has generated a very long text introducing Fawkham city, which is not what we were aiming for. Similarly to the first case, lexical metrics perform quite poor and the similarity metrics perform better (but nevertheless worse than previously). If we analyse the metrics more deeply, we can see that BLEU, METEOR and CHRF++ all present results under 0.2, which was to be expected given the content of the prediction. The semantic scores give a better result, ranging between 0.48 and 0.73, which can also be expected given that, even though it is not the answer we expected, it also speaks about Fawkham, which is a relevant part of the real content of the triple. These examples make it clear that it is necessary to rely on a range of metrics to better understand the results obtained. As noted earlier, lexical-overlap measures capture only part of text quality in data-to-text generation, particularly when a single reference is used. We therefore treat BLEU as one indicator among others and place additional emphasis on METEOR and CHRF++, which better accommodate

²³BLEU (B), METEOR (M), CHRF++ (C), Cos. Sim. (CS), BERTScore Prec. (BP), BERTScore Rec. (BR) and BERTScore F1 (BF1)

stemming, character-level overlap, and inflectional variation.

The second example of Table 7 also illustrates the problem of hallucinations when the model cannot grasp the task it is asked for. In Section 4.2.1 we explained that, for zero-shot instances, when the model didn't really understand the task we gave them, they tended to generate quite elaborated long texts that were somewhat related to some contents of the triples but did not reflect their meaning. This can also be an example of this case, which reinforces the idea that some sort of learning is needed when we want to execute specific tasks on general-purpose LLM, either context learning, fine tuning, transfer learning or even training them from scratch in some cases.

4.3. Discussion

The results of our study demonstrate the importance of contextualising the task and/or fine-tuning the model. We have seen that the performance greatly improves for both Spanish and English when going from zero-shot to one-shot setting. Going one step further, small-scale fine-tuning, even with limited computational resources, significantly enhances model performance for Spanish data-to-text generation, as evidenced by improvements in both task-specific metrics and stability across varying triple set sizes (Figure 5). These findings underscore the practical value of fine-tuning for adapting multilingual models to underrepresented languages, where pre-trained models often lag behind their English counterparts due to disparities in training data representation.

Regarding the individual performance, model-specific performance reveals a few insights for practitioners: Qwen2.5-1.5B-Instruct emerges as the top performer across most metrics, benefiting substantially from fine-tuning and exhibiting robust cross-lingual capabilities. Similarly, Qwen2.5-0.5B-Instruct, while less efficient, delivers competitive results, positioning it as a viable option for low-resource scenarios where computational constraints prioritise smaller models. On the other hand, Llama-3.2-1B-Instruct struggles with Spanish fine-tuning despite adequate English performance. This pattern suggests that, in our setting, pre-training language representation and model capacity may be more influential than task familiarity alone when adapting to Spanish triple-to-text generation. Finally, Salamandra-2B-Instruct, though computationally efficient, generally underperforms the other models in both Spanish and English settings.

These observations align with our multilingual and error analysis, which reinforces the necessity of language-specific adaptations. While models like Qwen2.5-1.5B-Instruct demonstrate promising cross-lingual transfer (despite a performance gap between English and Spanish), others like Llama-3.2-1B-Instruct may benefit from targeted training strategies or task-specific adaptation. The stark contrast in Llama-3.2-1B-Instruct's performance across languages (excelling in English but faltering in Spanish) suggests that task competence alone is insufficient; successful adaptation hinges on a model's ability to internalise language-specific structural and semantic patterns.

This study also highlights three key implications for Spanish data-to-text that could also be extrapolated to multilingual Natural Language Processing:

- Fine-tuning efficiency: Even minimal fine-tuning can help mitigate linguistic under-representation in pre-trained models, offering a cost-effective pathway to improve performance for languages like Spanish.
- Model selection criteria: Performance, computational cost, and language adaptability must be balanced. For Spanish-centric applications, Qwen2.5-1.5B-Instruct is optimal for low or medium-resource settings, while Qwen2.5-0.5B-Instruct provides a pragmatic compromise for more resource-constrained environments.
- Performance metrics selection: We observed that, for our task, some metrics that were normally used in the English sibling task were not really representative of the results we were obtaining. When selecting metrics, we have to take into account not only the task being evaluated but also the language of the data, given that each language has its own grammatical and vocabulary characteristics that might need specific accommodations.

5. Conclusions and future work

This paper presented a study on the performance of resource-efficient models in the Spanish triples-to-text task. First, we created a Spanish dataset for triple-to-text, Spanish WebNLG. This dataset was developed via a semi-supervised process of automatic translation and formatting, followed by a manual revision of triples and potentially problematic verbalisations. Given the availability of both the triples and verbalisations in Spanish, the dataset also presents the potential to be used bidirectionally for generating triples from plain text as well. Second, we developed a study addressing the three main research questions presented in the Introduction. Regarding *RQ1*, *How effectively can resource-efficient LLMs verbalise Spanish triples across different complexity levels?*, the results presented in the evaluation show that resource-efficient large language models (LLMs) are potentially capable of verbalizing triples sets of different size in Spanish, demonstrating their suitability for this task, as the models present competitive results along most of both the lexical and semantic metrics. For *RQ2*, *How does task contextualisation through examples impact model performance in Spanish triple verbalisation?*, the study showed that contextualizing the task with examples significantly improved the models' performance, as we observed a notable improvement from a zero-shot to a one-shot scenario, both in terms of evaluation metrics and processing time. Lastly, for *RQ3*, *What are the comparative advantages and limitations of prompt learning versus partial fine-tuning for Spanish triple verbalisation?*, the study found that partial fine-tuning, specifically using LoRA, generally led to improved performance in both metrics and time efficiency. Even though fine-tuning requires an initial time and resource investment, we see that fine-tuning also translates into a time efficiency improvement that, although modest, makes the overall performance better with fine-tuning compared to prompt learning. During the multilingual and error analysis, we also observed that Spanish can offer a broader range of valid surface realisations for the same underlying content, supported by richer verbal inflection, gender/number agreement, and more flexible word order. Because of these differences, we cannot always apply the same methods and metrics to both languages. Evaluation must consider linguistic nuances, such as morphology, syntax, and idiomatic expressions, to ensure fair and accurate assessments.

As future work, we plan to explore more concrete modelling strategies for Spanish triple-to-text verbalisation, including alternative parameter-efficient fine-tuning methods, improved prompting and constrained decoding, and the end-to-end training of neural architectures on the Spanish WebNLG dataset under a resource-efficiency perspective. On the data side, we aim to conduct a deeper, crowdsourced revision of the corpus that goes beyond low-similarity cases, in order to systematically assess the quality of MT-derived verbalisations, validate cosine-similarity thresholds, and check coherence between triples and texts. We also see importance in extending the benchmark to better reflect the diversity of Spanish by incorporating other Spanish varieties, while continuing to expand the resource to other languages to broaden its linguistic coverage. In parallel, we plan to adapt the dataset towards ontology verbalisation for ontology documentation and to investigate how linguistic factors such as morphological richness and syntactic flexibility interact with common automatic metrics and model behaviour. Finally, we intend to explore other structured data formats and datasets beyond WebNLG-style triples to further support non-English, resource-efficient NLG in a wider range of scenarios.

Acknowledgements

This work is supported by *Grant IARAG CPP2023-010895 funded by MICIU/AEI/ 10.13039/501100011033 and by ERDF/EU*. We also want to thank the WebNLG creators and especially the CNRS/LORIA team for their guidance during the planning of the Spanish WebNLG development.

References

- [1] O. Agarwal, H. Ge, S. Shakeri and R. Al-Rfou, Knowledge Graph Based Synthetic Corpus Generation for Knowledge-Enhanced Language Model Pre-training, in: *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, K. Toutanova, A. Rumshisky, L. Zettlemoyer, D. Hakkani-Tur, I. Beltagy, S. Bethard, R. Cotterell, T. Chakraborty and Y. Zhou, eds, Association for Computational Linguistics, Online, 2021, pp. 3554–3565. doi:10.18653/v1/2021.naacl-main.278. <https://aclanthology.org/2021.naacl-main.278>.

- [2] J. AGUADO-OREA, H. WITHERSTONE, L. BOURGEOIS and A. BASELGA, Learning to construct sentences in Spanish: a replication of the Weird Word Order technique, *Journal of Child Language* 46(6) (2019), 1249–1259–. doi:10.1017/S0305000919000448.
- [3] L. Banarescu, C. Bonial, S. Cai, M. Georgescu, K. Griffitt, U. Hermjakob, K. Knight, P. Koehn, M. Palmer and N. Schneider, Abstract Meaning Representation for Sembanking, in: *Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse*, Association for Computational Linguistics, 2013, pp. 178–186.
- [4] S. Boyd, TREND: a system for generating intelligent descriptions of time series data, in: *Proceedings of the IEEE international conference on intelligent processing systems (ICIPS-1998)*, Citeseer, 1998.
- [5] C. Callison-Burch, M. Osborne and P. Koehn, Re-evaluating the Role of Bleu in Machine Translation Research, in: *11th Conference of the European Chapter of the Association for Computational Linguistics*, D. McCarthy and S. Wintner, eds, Association for Computational Linguistics, Trento, Italy, 2006, pp. 249–256. <https://aclanthology.org/E06-1032/>.
- [6] T. Castro Ferreira, C. Gardent, N. Ilinykh, C. van der Lee, S. Mille, D. Moussallem and A. Shimorina, 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+)*, T. Castro Ferreira, C. Gardent, N. Ilinykh, C. van der Lee, S. Mille, D. Moussallem and A. Shimorina, eds, Association for Computational Linguistics, Dublin, Ireland (Virtual), 2020, pp. 55–76. <https://aclanthology.org/2020.webnlg-1.7/>.
- [7] I. Cervantes, *El español en el mundo: Anuario del Instituto Cervantes 2024*, Instituto Cervantes, 2024, Accessed: 2025-02-05. https://cvc.cervantes.es/lengua/anuario/anuario_24/.
- [8] W. Chen, Y. Su, X. Yan and W.Y. Wang, KGPT: Knowledge-Grounded Pre-Training for Data-to-Text Generation, in: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, B. Webber, T. Cohn, Y. He and Y. Liu, eds, Association for Computational Linguistics, Online, 2020, pp. 8635–8648. doi:10.18653/v1/2020.emnlp-main.697. <https://aclanthology.org/2020.emnlp-main.697/>.
- [9] E. Colin, C. Gardent, Y. M'rabet, S. Narayan and L. Perez-Beltrachini, The WebNLG Challenge: Generating Text from DBPedia Data, in: *Proceedings of the 9th International Natural Language Generation conference*, A. Isard, V. Rieser and D. Gkatzia, eds, Association for Computational Linguistics, Edinburgh, UK, 2016, pp. 163–167. doi:10.18653/v1/W16-6626. <https://aclanthology.org/W16-6626>.
- [10] L. Cripwell, A. Belz, C. Gardent, A. Gatt, C. Borg, M. Borg, J. Judge, M. Lorandi, A. Nikiforovskaya and W. Soto Martinez, The 2023 WebNLG Shared Task on Low Resource Languages. Overview and Evaluation Results (WebNLG 2023), in: *Proceedings of the Workshop on Multimodal, Multilingual Natural Language Generation and Multilingual WebNLG Challenge (MM-NLG 2023)*, A. Gatt, C. Gardent, L. Cripwell, A. Belz, C. Borg, A. Erdem and E. Erdem, eds, Association for Computational Linguistics, Prague, Czech Republic, 2023, pp. 55–66. <https://aclanthology.org/2023.mmmnlg-1.6/>.
- [11] J. Devlin, M. Chang, K. Lee and K. Toutanova, BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, *CoRR abs/1810.04805* (2018). <http://arxiv.org/abs/1810.04805>.
- [12] S. Duong, A. Lumbreras, M. Gartrell and P. Gallinari, Learning from Multiple Sources for Data-to-Text and Text-to-Data, in: *Proceedings of The 26th International Conference on Artificial Intelligence and Statistics*, F. Ruiz, J. Dy and J.-W. van de Meent, eds, Proceedings of Machine Learning Research, Vol. 206, PMLR, 2023, pp. 3733–3753. <https://proceedings.mlr.press/v206/duong23a.html>.
- [13] H. Elsahar, P. Vougiouklis, A. Remaci, C. Gravier, J. Hare, F. Laforest and E. Simperl, T-rex: A large scale alignment of natural language with knowledge base triples, in: *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, 2018.
- [14] A. Fan and C. Gardent, Multilingual AMR-to-Text Generation, in: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, B. Webber, T. Cohn, Y. He and Y. Liu, eds, Association for Computational Linguistics, Online, 2020, pp. 2889–2901. doi:10.18653/v1/2020.emnlp-main.231. <https://aclanthology.org/2020.emnlp-main.231/>.
- [15] Z. Fu, B. Shi, W. Lam, L. Bing and Z. Liu, Partially-Aligned Data-to-Text Generation with Distant Supervision, in: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, B. Webber, T. Cohn, Y. He and Y. Liu, eds, Association for Computational Linguistics, Online, 2020, pp. 9183–9193. doi:10.18653/v1/2020.emnlp-main.738. <https://aclanthology.org/2020.emnlp-main.738/>.
- [16] C. Gardent, A. Shimorina, S. Narayan and L. Perez-Beltrachini, The WebNLG Challenge: Generating Text from RDF Data, in: *Proceedings of the 10th International Conference on Natural Language Generation*, J.M. Alonso, A. Bugarín and E. Reiter, eds, Association for Computational Linguistics, Santiago de Compostela, Spain, 2017, pp. 124–133. doi:10.18653/v1/W17-3518. <https://aclanthology.org/W17-3518/>.
- [17] A. Gatt and E. Krahermer, Survey of the state of the art in natural language generation: core tasks, applications and evaluation, *J. Artif. Int. Res.* 61(1) (2018), 65–170–.
- [18] D. Gkatzia, Content Selection in Data-to-Text Systems: A Survey, *CoRR abs/1610.08375* (2016). <http://arxiv.org/abs/1610.08375>.
- [19] E. Goldberg, N. Driedger and R.I. Kittredge, Using natural-language processing to produce weather forecasts, *IEEE Expert* 9(2) (1994), 45–53. doi:10.1109/64.294135.
- [20] A. Gonzalez-Agirre, M. Pàmies, J. Llop, I. Baucells, S.D. Dalt, D. Tamayo, J.J. Saiz, F. Espuña, J. Prats, J. Aula-Blasco, M. Mina, A. Rubio, A. Shvets, A. Sallés, I. Lacunza, I. Pikabea, J. Palomar, J. Falcão, L. Tormo, L. Vasquez-Reina, M. Marimon, V. Ruíz-Fernández and M. Villegas, Salamandra Technical Report, 2025. <https://arxiv.org/abs/2502.08489>.
- [21] A. Grattafiori, A. Dubey, A. Jauhri, A. Pandey, A. Kadian, A. Al-Dahle, A. Letman, A. Mathur, A. Schelten and A.V. et al., The Llama 3 Herd of Models, 2024. <https://arxiv.org/abs/2407.21783>.
- [22] C. Hallett, R. Power and D. Scott, Summarisation and visualisation of e-Health data repositories, in: *UK E-Science All-Hands Meeting*, 2006. <https://oro.open.ac.uk/5261/>.
- [23] E.J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang and W. Chen, LoRA: Low-Rank Adaptation of Large Language Models, 2021. <https://arxiv.org/abs/2106.09685>.

- [24] M.I. Kamaluddin, M.W.K. Rasyid, F.H. Abqoriyyah and A. Saehu, Accuracy Analysis of DeepL: Breakthroughs in Machine Translation Technology, *Journal of English Education Forum (JEEF)* 4(2) (2024), 122–126–. doi:10.29303/jeeef.v4i2.681. <https://jeeef.unram.ac.id/index.php/jeeef/article/view/681>.
- [25] Z. Kasner and O. Dušek, Data-to-Text Generation with Iterative Text Editing, in: *Proceedings of the 13th International Conference on Natural Language Generation*, B. Davis, Y. Graham, J. Kelleher and Y. Sripada, eds, Association for Computational Linguistics, Dublin, Ireland, 2020, pp. 60–67. doi:10.18653/v1/2020.inlg-1.9. <https://aclanthology.org/2020.inlg-1.9/>.
- [26] Z. Kasner and O. Dusek, Beyond Traditional Benchmarks: Analyzing Behaviors of Open LLMs on Data-to-Text Generation, in: *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, L.-W. Ku, A. Martins and V. Srikumar, eds, Association for Computational Linguistics, Bangkok, Thailand, 2024, pp. 12045–12072. doi:10.18653/v1/2024.acl-long.651. <https://aclanthology.org/2024.acl-long.651/>.
- [27] M. Kurfali and R. Östling, Noisy Parallel Corpus Filtering through Projected Word Embeddings, in: *Proceedings of the Fourth Conference on Machine Translation (Volume 3: Shared Task Papers, Day 2)*, O. Bojar, R. Chatterjee, C. Federmann, M. Fishel, Y. Graham, B. Haddow, M. Huck, A.J. Yepes, P. Koehn, A. Martins, C. Monz, M. Negri, A. Névelo, M. Neves, M. Post, M. Turchi and K. Verspoor, eds, Association for Computational Linguistics, Florence, Italy, 2019, pp. 277–281. doi:10.18653/v1/W19-5438. <https://aclanthology.org/W19-5438/>.
- [28] A. Lavie and A. Agarwal, Meteor: an automatic metric for MT evaluation with high levels of correlation with human judgments, in: *Proceedings of the Second Workshop on Statistical Machine Translation*, StatMT '07, Association for Computational Linguistics, USA, 2007, pp. 228–231–.
- [29] M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov and L. Zettlemoyer, BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension, in: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, D. Jurafsky, J. Chai, N. Schlueter and J. Tetreault, eds, Association for Computational Linguistics, Online, 2020, pp. 7871–7880. doi:10.18653/v1/2020.acl-main.703. <https://aclanthology.org/2020.acl-main.703/>.
- [30] C.-Y. Lin, ROUGE: A Package for Automatic Evaluation of Summaries, in: *Text Summarization Branches Out*, Association for Computational Linguistics, Barcelona, Spain, 2004, pp. 74–81. <https://aclanthology.org/W04-1013/>.
- [31] Y. Lin, T. Ruan, J. Liu and H. Wang, A Survey on Neural Data-to-Text Generation, *IEEE Transactions on Knowledge and Data Engineering* 36(4) (2024), 1431–1449. doi:10.1109/TKDE.2023.3304385.
- [32] T. Liu, F. Luo, Q. Xia, S. Ma, B. Chang and Z. Sui, Hierarchical Encoder with Auxiliary Supervision for Neural Table-to-Text Generation: Learning Better Representation for Tables, *Proceedings of the AAAI Conference on Artificial Intelligence* 33(01) (2019), 6786–6793. doi:10.1609/aaai.v33i01.33016786. <https://ojs.aaai.org/index.php/AAAI/article/view/4653>.
- [33] M. Lorandi and A. Belz, High-quality Data-to-Text Generation for Severely Under-Resourced Languages with Out-of-the-box Large Language Models, 2024. <https://arxiv.org/abs/2402.12267>.
- [34] Y. Lu, J. Bao, Z. Ma, X. Han, Y. Wu, S. Cui and X. He, AUGUST: an Automatic Generation Understudy for Synthesizing Conversational Recommendation Datasets, in: *Findings of the Association for Computational Linguistics: ACL 2023*, A. Rogers, J. Boyd-Graber and N. Okazaki, eds, Association for Computational Linguistics, Toronto, Canada, 2023, pp. 10538–10549. doi:10.18653/v1/2023.findings-acl.670. <https://aclanthology.org/2023.findings-acl.670>.
- [35] K. Ma, H. Cheng, X. Liu, E. Nyberg and J. Gao, Open Domain Question Answering with A Unified Knowledge Interface, in: *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, S. Muresan, P. Nakov and A. Villavicencio, eds, Association for Computational Linguistics, Dublin, Ireland, 2022, pp. 1605–1620. doi:10.18653/v1/2022.acl-long.113. <https://aclanthology.org/2022.acl-long.113>.
- [36] N. Mathur, T. Baldwin and T. Cohn, Tangled up in BLEU: Reevaluating the Evaluation of Automatic Machine Translation Evaluation Metrics, in: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, D. Jurafsky, J. Chai, N. Schlueter and J. Tetreault, eds, Association for Computational Linguistics, Online, 2020, pp. 4984–4997. doi:10.18653/v1/2020.acl-main.448. <https://aclanthology.org/2020.acl-main.448/>.
- [37] J. Maynez, S. Narayan, B. Bohnet and R. McDonald, On Faithfulness and Factuality in Abstractive Summarization, in: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, D. Jurafsky, J. Chai, N. Schlueter and J. Tetreault, eds, Association for Computational Linguistics, Online, 2020, pp. 1906–1919. doi:10.18653/v1/2020.acl-main.173. <https://aclanthology.org/2020.acl-main.173/>.
- [38] S.W. Mcroy, S. Channarukul and S.S. Ali, An augmented template-based approach to text realization, *Nat. Lang. Eng.* 9(4) (2003), 381–420–. doi:10.1017/S1351324903003188.
- [39] P. Mendes, M. Jakob and C. Bizer, DBpedia: A Multilingual Cross-domain Knowledge Base, in: *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, N. Calzolari, K. Choukri, T. Declerck, M.U. Doğan, B. Maegaard, J. Mariani, A. Moreno, J. Odijk and S. Piperidis, eds, European Language Resources Association (ELRA), Istanbul, Turkey, 2012, pp. 1813–1817. http://www.lrec-conf.org/proceedings/lrec2012/pdf/570_Paper.pdf.
- [40] S. Mille, S. Dasiopoulou, B. Fisas and L. Wanner, Teaching FORGE to Verbalize DBpedia Properties in Spanish, in: *Proceedings of the 12th International Conference on Natural Language Generation*, K. van Deemter, C. Lin and H. Takamura, eds, Association for Computational Linguistics, Tokyo, Japan, 2019, pp. 473–483. doi:10.18653/v1/W19-8659. <https://aclanthology.org/W19-8659/>.
- [41] A. Moreno-Sandoval and J.M. Goñi-Menoyo, Spanish Inflectional Morphology in DATR, *Journal of Logic, Language and Information* 11(1) (2002), 79–105. doi:10.1023/A:1013019622647.
- [42] A. Moryossef, Y. Goldberg and I. Dagan, 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)*, J. Burstein, C. Doran and T. Solorio, eds, Association for Computational Linguistics, Minneapolis, Minnesota, 2019, pp. 2267–2277. doi:10.18653/v1/N19-1236. <https://aclanthology.org/N19-1236/>.

- [43] D. Moussallem, T.C. Ferreira, M. Zampieri, M.C. Cavalcanti, G. Xexéo, M. Neves and A.-C.N. Ngomo, RDF2PT: Generating Brazilian Portuguese Texts from RDF Data, 2018. <https://arxiv.org/abs/1802.08150>.
- [44] L. Nan, D. Radev, R. Zhang, A. Rau, A. Sivaprasad, C. Hsieh, X. Tang, A. Vyas, N. Verma, P. Krishna and Y.e.a. Liu, DART: Open-Domain Structured Data Record to Text Generation, in: *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, K. Toutanova, A. Rumshisky, L. Zettlemoyer, D. Hakkani-Tur, I. Beltagy, S. Bethard, R. Cotterell, T. Chakraborty and Y. Zhou, eds, Association for Computational Linguistics, Online, 2021, pp. 432–447. doi:10.18653/v1/2021.naacl-main.37. <https://aclanthology.org/2021.naacl-main.37/>.
- [45] J. Novikova, O. Dušek and V. Rieser, The E2E Dataset: New Challenges For End-to-End Generation, in: *Proceedings of the 18th Annual SIGdial Meeting on Discourse and Dialogue*, K. Jokinen, M. Stede, D. DeVault and A. Louis, eds, Association for Computational Linguistics, Saarbrücken, Germany, 2017, pp. 201–206. doi:10.18653/v1/W17-5525. <https://aclanthology.org/W17-5525/>.
- [46] F. Ordóñez and E. Treviño, Left dislocated subjects and the pro-drop parameter: A case study of Spanish, *Lingua* **107**(1) (1999), 39–68. doi:[https://doi.org/10.1016/S0024-3841\(98\)00020-5](https://doi.org/10.1016/S0024-3841(98)00020-5). <https://www.sciencedirect.com/science/article/pii/S0024384198000205>.
- [47] C.C. Osuji, T.C. Ferreira and B. Davis, A Systematic Review of Data-to-Text NLG, 2024. <https://arxiv.org/abs/2402.08496>.
- [48] K. Papineni, S. Roukos, T. Ward and W.-J. Zhu, BLEU: a method for automatic evaluation of machine translation, in: *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, Association for Computational Linguistics, USA, 2002, pp. 311–318. doi:10.3115/1073083.1073135.
- [49] P. Pasupat and P. Liang, Compositional Semantic Parsing on Semi-Structured Tables, in: *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, C. Zong and M. Strube, eds, Association for Computational Linguistics, Beijing, China, 2015, pp. 1470–1480. doi:10.3115/v1/P15-1142. <https://aclanthology.org/P15-1142/>.
- [50] M. Popović, chrF++: words helping character n-grams, in: *Proceedings of the Second Conference on Machine Translation*, O. Bojar, C. Buck, R. Chatterjee, C. Federmann, Y. Graham, B. Haddow, M. Huck, A.J. Yepes, P. Koehn and J. Kreutzer, eds, Association for Computational Linguistics, Copenhagen, Denmark, 2017, pp. 612–618. doi:10.18653/v1/W17-4770. <https://aclanthology.org/W17-4770/>.
- [51] M. Post, A Call for Clarity in Reporting BLEU Scores, in: *Proceedings of the Third Conference on Machine Translation: Research Papers*, O. Bojar, R. Chatterjee, C. Federmann, M. Fishel, Y. Graham, B. Haddow, M. Huck, A.J. Yepes, P. Koehn, C. Monz, M. Negri, A. Névél, M. Neves, M. Post, L. Specia, M. Turchi and K. Verspoor, eds, Association for Computational Linguistics, Brussels, Belgium, 2018, pp. 186–191. doi:10.18653/v1/W18-6319. <https://aclanthology.org/W18-6319/>.
- [52] R. Puduppully, L. Dong and M. Lapata, Data-to-text generation with content selection and planning, in: *Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence and Thirty-First Innovative Applications of Artificial Intelligence Conference and Ninth AAAI Symposium on Educational Advances in Artificial Intelligence*, AAAI'19/AAAI'19/EAAI'19, AAAI Press, 2019. ISBN 978-1-57735-809-1. doi:10.1609/aaai.v33i01.33016908.
- [53] A. Radford, J. Wu, R. Child et al., Language models are unsupervised multitask learners, *OpenAI blog* **1**(8) (2019), 9.
- [54] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li and P.J. Liu, Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer, *Journal of Machine Learning Research* **21**(140) (2020), 1–67. <http://jmlr.org/papers/v21/20-074.html>.
- [55] A. Ramos-Soto, A.J. Bugarín, S. Barro and J. Taboada, Linguistic Descriptions for Automatic Generation of Textual Short-Term Weather Forecasts on Real Prediction Data, *IEEE Transactions on Fuzzy Systems* **23**(1) (2015), 44–57. doi:10.1109/TFUZZ.2014.2328011.
- [56] N. Reimers and I. Gurevych, Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks, in: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*, Association for Computational Linguistics, 2019. <http://arxiv.org/abs/1908.10084>.
- [57] N. Reimers and I. Gurevych, Making Monolingual Sentence Embeddings Multilingual using Knowledge Distillation, in: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, Association for Computational Linguistics, 2020. <https://arxiv.org/abs/2004.09813>.
- [58] E. Reiter, A Structured Review of the Validity of BLEU, *Computational Linguistics* **44**(3) (2018), 393–401. doi:10.1162/coli_a_00322. <https://aclanthology.org/J18-3002/>.
- [59] E. REITER and R. DALE, Building applied natural language generation systems, *Natural Language Engineering* **3**(1) (1997), 57–87. doi:10.1017/S1351324997001502.
- [60] V. Satopaa, J. Albrecht, D. Irwin and B. Raghavan, Finding a "Kneedle" in a Haystack: Detecting Knee Points in System Behavior, in: *2011 31st International Conference on Distributed Computing Systems Workshops*, 2011, pp. 166–171. doi:10.1109/ICDCSW.2011.20.
- [61] Y. Su, D. Vandyke, S. Wang, Y. Fang and N. Collier, Plan-then-Generate: Controlled Data-to-Text Generation via Planning, in: *Findings of the Association for Computational Linguistics: EMNLP 2021*, M.-F. Moens, X. Huang, L. Specia and S.W.-t. Yih, eds, Association for Computational Linguistics, Punta Cana, Dominican Republic, 2021, pp. 895–909. doi:10.18653/v1/2021.findings-emnlp.76. <https://aclanthology.org/2021.findings-emnlp.76/>.
- [62] Q. Team, Qwen2.5: A Party of Foundation Models, 2024. <https://qwenlm.github.io/blog/qwen2.5/>.
- [63] R.L. Thorndike, Who belongs in the family?, *Psychometrika* **18**(4) (1953), 267–276.
- [64] C. van der Lee, E. Krahmer and S. Wubben, Automated learning of templates for data-to-text generation: comparing rule-based, statistical and neural methods, in: *Proceedings of the 11th International Conference on Natural Language Generation*, E. Krahmer, A. Gatt and M. Goudbeek, eds, Association for Computational Linguistics, Tilburg University, The Netherlands, 2018, pp. 35–45. doi:10.18653/v1/W18-6504. <https://aclanthology.org/W18-6504/>.
- [65] D. Vrandečić and M. Krötzsch, Wikidata: a free collaborative knowledgebase, *Commun. ACM* **57**(10) (2014), 78–85. doi:10.1145/2629489.

- [66] S. Wiseman, S. Shieber and A. Rush, Challenges in Data-to-Document Generation, in: *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, M. Palmer, R. Hwa and S. Riedel, eds, Association for Computational Linguistics, Copenhagen, Denmark, 2017, pp. 2253–2263. doi:10.18653/v1/D17-1239. <https://aclanthology.org/D17-1239/>.
- [67] S. Wiseman, S. Shieber and A. Rush, Learning Neural Templates for Text Generation, in: *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, E. Riloff, D. Chiang, J. Hockenmaier and J. Tsujii, eds, Association for Computational Linguistics, Brussels, Belgium, 2018, pp. 3174–3187. doi:10.18653/v1/D18-1356. <https://aclanthology.org/D18-1356/>.
- [68] T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, J. Davison, S. Shleifer, P. von Platen, C. Ma, Y. Jernite, J. Plu, C. Xu, T.L. Scao, S. Gugger, M. Drame, Q. Lhoest and A.M. Rush, HuggingFace’s Transformers: State-of-the-art Natural Language Processing, 2020. <https://arxiv.org/abs/1910.03771>.
- [69] World Wide Web Consortium, RDF 1.1 Concepts and Abstract Syntax, 2014, Accessed: 2025-02-17.
- [70] D. Xu, J. Li, M. Zhu, M. Zhang and G. Zhou, XLPT-AMR: Cross-Lingual Pre-Training via Multi-Task Learning for Zero-Shot AMR Parsing and Text Generation, in: *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, C. Zong, F. Xia, W. Li and R. Navigli, eds, Association for Computational Linguistics, Online, 2021, pp. 896–907. doi:10.18653/v1/2021.acl-long.73. <https://aclanthology.org/2021.acl-long.73/>.
- [71] A. Yang, B. Yang, B. Hui, B. Zheng, B. Yu, C. Zhou, C. Li, C. Li, D. Liu and F.H. et al., Qwen2 Technical Report, *arXiv preprint arXiv:2407.10671* (2024).
- [72] S. Yang, Y. Liu, D. Feng and D. Li, Text Generation From Data With Dynamic Planning, *IEEE/ACM Transactions on Audio, Speech, and Language Processing* **30** (2022), 26–34. doi:10.1109/TASLP.2021.3129346.
- [73] T. Zhang*, V. Kishore*, F. Wu*, K.Q. Weinberger and Y. Artzi, BERTScore: Evaluating Text Generation with BERT, in: *International Conference on Learning Representations*, 2020. <https://openreview.net/forum?id=SkeHuCVFDr>.
- [74] C. Zhao, M. Walker and S. Chaturvedi, Bridging the Structural Gap Between Encoding and Decoding for Data-To-Text Generation, in: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, D. Jurafsky, J. Chai, N. Schluter and J. Tetreault, eds, Association for Computational Linguistics, Online, 2020, pp. 2481–2491. doi:10.18653/v1/2020.acl-main.224. <https://aclanthology.org/2020.acl-main.224>.
- [75] V. Zhong, C. Xiong and R. Socher, Seq2SQL: Generating Structured Queries from Natural Language using Reinforcement Learning, 2017. <https://arxiv.org/abs/1709.00103>.

Appendix A. Context learning prompts

– Spanish prompts:

* Zero-shot prompt:

User: En español, los datos estructurados se representan comúnmente como tripletas o triples, con el formato [sujeto, predicado, objeto]. A partir de estas tripletas, genera un texto de un solo párrafo formado por oraciones completas, gramaticalmente correctas y naturales. Genera el texto únicamente a partir de las siguientes tripletas:

* One-shot prompt:

User: En español, los datos estructurados se representan comúnmente como tripletas o triples, con el formato [sujeto, predicado, objeto]. A partir de estas tripletas, genera un texto de un solo párrafo formado por oraciones completas, gramaticalmente correctas y naturales. Genera el texto únicamente a partir de las siguientes tripletas:

Tripletas:

[sujeto: 'Arion_(personaje_de_cómic)', predicado: 'Creador', objeto: 'Jan_Duursema']

[sujeto: 'Jan_Duursema', predicado: 'Premio', objeto: 'Premio_Eisner']

[sujeto: 'Arion_(personaje_de_cómic)', predicado: 'NombreAlternativo', objeto: 'Ahriáhn']

[sujeto: 'Arion_(personaje_de_cómic)', predicado: 'Creador', objeto: 'Paul_Kupperberg']

System: Arion (también conocido como Ahri'ahn) es un personaje de cómic creado por Paul Kupperberg y Jan Duursema, que ganó el premio Eisner.

* Few-shot prompt:

User: En español, los datos estructurados se representan comúnmente como tripletas o triples, con el formato [sujeto, predicado, objeto]. A partir de estas tripletas, genera un texto de un solo párrafo formado por oraciones completas, gramaticalmente correctas y naturales. Genera el texto únicamente a partir de las siguientes tripletas:

Tripletas:

[sujeto: 'Arion_(personaje_de_cómic)', predicado: 'Creador', objeto: 'Jan_Duursema']

[sujeto: 'Jan_Duursema', predicado: 'Premio', objeto: 'Premio_Eisner']

[sujeto: 'Arion_(personaje_de_cómic)', predicado: 'NombreAlternativo', objeto: 'Ahriáhn']

[sujeto: 'Arion_(personaje_de_cómic)', predicado: 'Creador', objeto: 'Paul_Kupperberg']

System: Arion (también conocido como Ahri'ahn) es un personaje de cómic creado por Paul Kupperberg y Jan Duursema, que ganó el premio Eisner.

User:

Tripletas:

[sujeto: 'Monumento_a_la_11°_Infantería_del_Mississippi', predicado: 'Categoría', objeto: 'Propiedad_contribuidora']

[sujeto: 'Monumento_a_la_11°_Infantería_del_Mississippi', predicado: 'Municipio', objeto: 'Gettysburg,_Pennsylvania']

System: El monumento a la 11° Infantería del Mississippi pertenece a la categoría de propiedad contribuyente y se encuentra en el municipio de Gettysburg, en Pensilvania.

– English prompts:

* Zero-shot prompt:

User: In English, structured data is commonly represented as triples, with the format [subject, predicate, object]. Based on these triples, generate a single-paragraph text composed of complete, grammatically correct,

and natural sentences. Generate the text solely from the following triples:

* **One-shot prompt:**

User: In English, structured data is commonly represented as triples, with the format [subject, predicate, object]. Based on these triples, generate a single-paragraph text composed of complete, grammatically correct, and natural sentences. Generate the text solely from the following triples:

Triples:

[subject: 'Arion_(comicsCharacter)', predicate: 'creator', object: 'Jan_Duursema']

[subject: 'Jan_Duursema', predicate: 'award', object: 'Eisner_Award']

[subject: 'Arion_(comicsCharacter)', predicate: 'alternativeName', object: 'Ahri'ahn']

[subject: 'Arion_(comicsCharacter)', predicate: 'creator', object: 'Paul_Kupperberg']

System: Arion (also known as Ahri'ahn) is a comic character created by Paul Kupperberg and Jan Duursema, who won the Eisner award.

* **Few-shot prompt:**

User: In English, structured data is commonly represented as triples, with the format [subject, predicate, object]. Based on these triples, generate a single-paragraph text composed of complete, grammatically correct, and natural sentences. Generate the text solely from the following triples:

Triples:

[subject: 'Arion_(comicsCharacter)', predicate: 'creator', object: 'Jan_Duursema']

[subject: 'Jan_Duursema', predicate: 'award', object: 'Eisner_Award']

[subject: 'Arion_(comicsCharacter)', predicate: 'alternativeName', object: 'Ahri'ahn']

[subject: 'Arion_(comicsCharacter)', predicate: 'creator', object: 'Paul_Kupperberg']

System: Arion (also known as Ahri'ahn) is a comic character created by Paul Kupperberg and Jan Duursema, who won the Eisner award.

User:

Triples:

[subject: '11th_Mississippi_Infantry_Monument', predicate: 'category', object: 'Contributing_property']

[subject: '11th_Mississippi_Infantry_Monument', predicate: 'municipality', object: 'Gettysburg,_Pennsylvania']

System: The 11th Mississippi Infantry Monument is categorised as a contributing property and can be found in Gettysburg, Pennsylvania.

Appendix B. Complete Spanish evaluation results

Table 8
 Prompt-based learning Spanish test 1 full evaluation results

Triple size	Prompt type	Model	Metrics							Average time per triple set in seconds
			BLEU	METEOR	CHRf++	Cosine Similarity	BERTScore Precision	BERTScore Recall	BERTScore F1	
1	zero	Llama-3.2-1B-Instruct	8.71	0.356	36.35	0.694	0.676	0.808	0.734	1.640
		Qwen2.5-0.5B-Instruct	12.96	0.391	42.90	0.682	0.732	0.805	0.764	1.363
		Qwen2.5-1.5B-Instruct	11.90	0.419	38.47	0.772	0.693	0.825	0.750	2.948
		salamandra-2b-instruct	16.48	0.428	47.30	0.729	0.745	0.809	0.773	0.962
	one	Llama-3.2-1B-Instruct	19.69	0.534	54.88	0.794	0.783	0.855	0.815	0.615
		Qwen2.5-0.5B-Instruct	22.15	0.539	57.30	0.836	0.811	0.861	0.833	0.657
		Qwen2.5-1.5B-Instruct	28.29	0.597	62.51	0.884	0.854	0.888	0.869	0.762
		salamandra-2b-instruct	15.37	0.424	47.86	0.731	0.752	0.817	0.780	0.975
	few	Llama-3.2-1B-Instruct	26.08	0.599	60.52	0.854	0.819	0.876	0.845	0.578
		Qwen2.5-0.5B-Instruct	27.96	0.574	60.82	0.856	0.840	0.874	0.855	0.600
		Qwen2.5-1.5B-Instruct	31.24	0.612	64.41	0.897	0.862	0.892	0.875	0.708
		salamandra-2b-instruct	21.35	0.501	54.33	0.821	0.810	0.854	0.829	0.740
2	zero	Llama-3.2-1B-Instruct	9.81	0.391	41.60	0.688	0.699	0.802	0.745	1.625
		Qwen2.5-0.5B-Instruct	12.01	0.386	42.91	0.663	0.730	0.784	0.753	1.738
		Qwen2.5-1.5B-Instruct	13.91	0.484	47.12	0.809	0.735	0.835	0.779	2.540
		salamandra-2b-instruct	13.53	0.411	46.28	0.719	0.748	0.791	0.766	1.176
	one	Llama-3.2-1B-Instruct	21.53	0.505	54.61	0.817	0.809	0.834	0.819	0.665
		Qwen2.5-0.5B-Instruct	22.13	0.509	55.66	0.850	0.832	0.843	0.835	0.796
		Qwen2.5-1.5B-Instruct	27.23	0.594	62.06	0.906	0.856	0.873	0.863	1.026
		salamandra-2b-instruct	13.61	0.413	46.62	0.713	0.749	0.790	0.767	1.234
	few	Llama-3.2-1B-Instruct	24.52	0.544	57.86	0.862	0.829	0.851	0.838	0.678
		Qwen2.5-0.5B-Instruct	24.21	0.535	58.23	0.871	0.839	0.853	0.844	0.826
		Qwen2.5-1.5B-Instruct	26.72	0.588	61.47	0.904	0.851	0.873	0.860	1.052
		salamandra-2b-instruct	20.98	0.499	55.08	0.833	0.812	0.832	0.820	0.907
3	zero	Llama-3.2-1B-Instruct	9.74	0.383	43.03	0.692	0.700	0.785	0.738	1.784
		Qwen2.5-0.5B-Instruct	10.70	0.367	41.73	0.623	0.713	0.765	0.735	2.020
		Qwen2.5-1.5B-Instruct	14.82	0.501	49.51	0.808	0.736	0.827	0.776	2.863
		salamandra-2b-instruct	11.46	0.374	44.00	0.689	0.723	0.769	0.743	1.506
	one	Llama-3.2-1B-Instruct	20.70	0.471	52.29	0.819	0.810	0.819	0.813	0.787
		Qwen2.5-0.5B-Instruct	20.39	0.472	53.13	0.841	0.819	0.823	0.820	1.031
		Qwen2.5-1.5B-Instruct	26.52	0.568	59.99	0.899	0.843	0.858	0.849	1.350
		salamandra-2b-instruct	13.03	0.408	46.94	0.723	0.743	0.780	0.759	1.360
	few	Llama-3.2-1B-Instruct	23.91	0.520	56.03	0.852	0.825	0.835	0.828	0.780
		Qwen2.5-0.5B-Instruct	23.13	0.517	55.87	0.863	0.825	0.832	0.827	1.105
		Qwen2.5-1.5B-Instruct	27.93	0.570	60.45	0.900	0.846	0.860	0.852	1.318
		salamandra-2b-instruct	17.26	0.458	51.82	0.805	0.791	0.808	0.797	1.071
4	zero	Llama-3.2-1B-Instruct	9.38	0.372	43.65	0.701	0.704	0.781	0.738	2.028
		Qwen2.5-0.5B-Instruct	10.36	0.355	42.77	0.681	0.725	0.761	0.741	2.113
		Qwen2.5-1.5B-Instruct	14.59	0.488	50.71	0.822	0.745	0.822	0.779	3.082
		salamandra-2b-instruct	9.53	0.338	41.53	0.659	0.713	0.751	0.729	1.807
	one	Llama-3.2-1B-Instruct	19.06	0.431	48.54	0.812	0.805	0.799	0.800	0.859
		Qwen2.5-0.5B-Instruct	16.42	0.419	49.19	0.834	0.806	0.804	0.803	1.298
		Qwen2.5-1.5B-Instruct	23.02	0.524	57.15	0.890	0.827	0.841	0.832	1.724
		salamandra-2b-instruct	11.74	0.369	44.64	0.712	0.742	0.765	0.751	1.483
	few	Llama-3.2-1B-Instruct	20.91	0.460	51.85	0.842	0.814	0.811	0.810	0.902
		Qwen2.5-0.5B-Instruct	19.36	0.473	53.31	0.848	0.805	0.814	0.808	1.457
		Qwen2.5-1.5B-Instruct	24.46	0.539	58.10	0.896	0.832	0.843	0.836	1.686
		salamandra-2b-instruct	14.98	0.415	48.45	0.786	0.774	0.789	0.779	1.356
5	zero	Llama-3.2-1B-Instruct	9.80	0.371	44.15	0.702	0.705	0.771	0.735	2.247
		Qwen2.5-0.5B-Instruct	10.46	0.331	41.61	0.687	0.728	0.750	0.736	2.316
		Qwen2.5-1.5B-Instruct	14.89	0.474	50.78	0.822	0.741	0.813	0.774	3.699
		salamandra-2b-instruct	9.85	0.329	41.39	0.663	0.716	0.740	0.726	1.781
	one	Llama-3.2-1B-Instruct	16.75	0.377	45.12	0.802	0.804	0.782	0.791	0.899
		Qwen2.5-0.5B-Instruct	14.94	0.397	48.14	0.815	0.791	0.787	0.788	1.695
		Qwen2.5-1.5B-Instruct	23.96	0.514	58.08	0.896	0.824	0.833	0.828	2.015
		salamandra-2b-instruct	10.55	0.341	43.73	0.702	0.729	0.751	0.738	1.759
	few	Llama-3.2-1B-Instruct	18.97	0.421	47.92	0.823	0.811	0.791	0.799	0.937
		Qwen2.5-0.5B-Instruct	17.60	0.438	51.42	0.842	0.794	0.797	0.794	1.773
		Qwen2.5-1.5B-Instruct	24.29	0.516	57.45	0.892	0.826	0.834	0.829	2.020
		salamandra-2b-instruct	14.39	0.385	47.19	0.778	0.761	0.777	0.767	1.660
6	zero	Llama-3.2-1B-Instruct	10.34	0.390	45.82	0.701	0.704	0.775	0.736	2.669
		Qwen2.5-0.5B-Instruct	10.82	0.348	45.09	0.744	0.742	0.758	0.748	2.570
		Qwen2.5-1.5B-Instruct	16.51	0.482	52.82	0.838	0.749	0.812	0.778	4.023
		salamandra-2b-instruct	11.10	0.347	43.98	0.706	0.724	0.750	0.735	2.098
	one	Llama-3.2-1B-Instruct	16.88	0.378	44.36	0.805	0.808	0.777	0.791	1.045
		Qwen2.5-0.5B-Instruct	15.03	0.398	48.19	0.821	0.779	0.778	0.777	1.994
		Qwen2.5-1.5B-Instruct	22.28	0.501	56.46	0.891	0.814	0.826	0.819	2.475
		salamandra-2b-instruct	12.05	0.353	44.27	0.738	0.738	0.755	0.744	2.019
	few	Llama-3.2-1B-Instruct	17.78	0.400	46.37	0.834	0.823	0.785	0.802	1.027
		Qwen2.5-0.5B-Instruct	18.76	0.442	51.84	0.840	0.793	0.795	0.793	2.153
		Qwen2.5-1.5B-Instruct	24.34	0.496	56.60	0.895	0.822	0.828	0.824	2.413
		salamandra-2b-instruct	13.97	0.386	47.30	0.789	0.767	0.771	0.767	1.696
7	zero	Llama-3.2-1B-Instruct	11.33	0.375	46.32	0.687	0.711	0.766	0.736	2.778
		Qwen2.5-0.5B-Instruct	13.36	0.360	46.86	0.727	0.744	0.757	0.749	2.650
		Qwen2.5-1.5B-Instruct	18.25	0.496	55.63	0.846	0.759	0.815	0.785	4.133
		salamandra-2b-instruct	10.85	0.331	42.57	0.702	0.714	0.740	0.726	2.514
	one	Llama-3.2-1B-Instruct	16.40	0.360	42.44	0.801	0.814	0.770	0.790	1.037
		Qwen2.5-0.5B-Instruct	15.31	0.384	48.05	0.812	0.777	0.774	0.774	2.241
		Qwen2.5-1.5B-Instruct	23.85	0.506	57.81	0.890	0.813	0.825	0.818	2.771
		salamandra-2b-instruct	13.06	0.353	46.84	0.737	0.742	0.751	0.745	1.986
	few	Llama-3.2-1B-Instruct	16.46	0.359	43.05	0.804	0.809	0.768	0.787	1.086
		Qwen2.5-0.5B-Instruct	20.96	0.450	53.32	0.859	0.800	0.794	0.796	2.230
		Qwen2.5-1.5B-Instruct	24.29	0.495	57.91	0.893	0.817	0.824	0.820	2.739
		salamandra-2b-instruct	13.51	0.348	44.05	0.752	0.751	0.751	0.749	1.831

Table 9
 Prompt-based learning Spanish test 2 full evaluation results

Triple size	Prompt type	Model	Metrics							Average time per triple set in seconds
			BLEU	METEOR	CHR++	Cosine Similarity	BERTScore Precision	BERTScore Recall	BERTScore F1	
1	zero	Llama-3.2-1B-Instruct	9.44	0.374	38.11	0.711	0.686	0.814	0.742	1.597
		Qwen2.5-0.5B-Instruct	14.21	0.397	43.54	0.674	0.735	0.808	0.767	1.386
		Qwen2.5-1.5B-Instruct	11.25	0.410	37.77	0.759	0.688	0.826	0.747	3.053
		salamandra-2b-instruct	15.58	0.439	47.10	0.733	0.745	0.814	0.775	0.966
	one	Llama-3.2-1B-Instruct	20.39	0.548	55.64	0.802	0.787	0.858	0.819	0.613
		Qwen2.5-0.5B-Instruct	21.57	0.541	56.55	0.821	0.807	0.860	0.831	0.751
		Qwen2.5-1.5B-Instruct	28.02	0.609	63.44	0.882	0.854	0.890	0.870	0.748
		salamandra-2b-instruct	15.82	0.437	47.88	0.727	0.754	0.819	0.783	0.980
	few	Llama-3.2-1B-Instruct	26.16	0.602	60.70	0.850	0.820	0.877	0.846	0.568
		Qwen2.5-0.5B-Instruct	28.90	0.584	61.25	0.859	0.845	0.876	0.859	0.592
		Qwen2.5-1.5B-Instruct	31.10	0.616	64.44	0.897	0.862	0.892	0.875	0.712
		salamandra-2b-instruct	21.71	0.497	53.93	0.811	0.803	0.849	0.823	0.792
2	zero	Llama-3.2-1B-Instruct	9.97	0.388	42.01	0.697	0.703	0.800	0.746	1.579
		Qwen2.5-0.5B-Instruct	11.88	0.388	42.90	0.670	0.736	0.786	0.757	1.614
		Qwen2.5-1.5B-Instruct	13.66	0.478	46.23	0.788	0.728	0.832	0.773	2.614
		salamandra-2b-instruct	13.26	0.402	45.47	0.722	0.746	0.790	0.765	1.248
	one	Llama-3.2-1B-Instruct	22.76	0.526	55.80	0.823	0.812	0.838	0.823	0.682
		Qwen2.5-0.5B-Instruct	21.03	0.495	54.62	0.842	0.828	0.839	0.832	0.835
		Qwen2.5-1.5B-Instruct	26.46	0.591	62.29	0.904	0.851	0.873	0.860	1.065
		salamandra-2b-instruct	14.48	0.427	48.19	0.728	0.755	0.797	0.773	1.186
	few	Llama-3.2-1B-Instruct	24.91	0.557	58.46	0.861	0.827	0.853	0.838	0.698
		Qwen2.5-0.5B-Instruct	25.41	0.547	58.78	0.873	0.840	0.854	0.846	0.854
		Qwen2.5-1.5B-Instruct	27.89	0.595	62.34	0.907	0.856	0.875	0.864	1.034
		salamandra-2b-instruct	19.43	0.484	53.37	0.819	0.809	0.827	0.816	0.946
3	zero	Llama-3.2-1B-Instruct	9.85	0.377	43.47	0.693	0.704	0.787	0.741	1.757
		Qwen2.5-0.5B-Instruct	10.25	0.364	41.11	0.617	0.702	0.759	0.727	2.230
		Qwen2.5-1.5B-Instruct	15.57	0.504	49.79	0.806	0.740	0.828	0.779	2.838
		salamandra-2b-instruct	12.27	0.395	45.55	0.706	0.733	0.773	0.750	1.458
	one	Llama-3.2-1B-Instruct	21.15	0.472	52.37	0.816	0.806	0.820	0.811	0.803
		Qwen2.5-0.5B-Instruct	20.09	0.476	52.84	0.839	0.821	0.822	0.820	1.081
		Qwen2.5-1.5B-Instruct	26.23	0.566	59.80	0.902	0.843	0.858	0.849	1.380
		salamandra-2b-instruct	12.83	0.382	45.44	0.682	0.734	0.770	0.749	1.407
	few	Llama-3.2-1B-Instruct	23.91	0.514	55.55	0.853	0.825	0.833	0.827	0.803
		Qwen2.5-0.5B-Instruct	22.48	0.508	56.16	0.868	0.827	0.833	0.829	1.096
		Qwen2.5-1.5B-Instruct	26.91	0.570	59.84	0.898	0.841	0.857	0.847	1.369
		salamandra-2b-instruct	18.65	0.472	52.63	0.833	0.797	0.815	0.804	1.154
4	zero	Llama-3.2-1B-Instruct	8.82	0.364	42.54	0.682	0.701	0.776	0.734	2.112
		Qwen2.5-0.5B-Instruct	10.75	0.358	43.43	0.678	0.726	0.762	0.741	2.110
		Qwen2.5-1.5B-Instruct	14.23	0.486	50.44	0.823	0.745	0.820	0.779	3.082
		salamandra-2b-instruct	9.53	0.329	42.02	0.655	0.718	0.748	0.730	1.617
	one	Llama-3.2-1B-Instruct	18.78	0.426	48.29	0.805	0.804	0.797	0.799	0.871
		Qwen2.5-0.5B-Instruct	16.21	0.414	49.26	0.824	0.802	0.799	0.799	1.371
		Qwen2.5-1.5B-Instruct	23.25	0.531	57.29	0.890	0.828	0.843	0.834	1.737
		salamandra-2b-instruct	11.01	0.365	43.93	0.706	0.732	0.764	0.745	1.650
	few	Llama-3.2-1B-Instruct	20.54	0.453	50.90	0.831	0.815	0.808	0.810	0.904
		Qwen2.5-0.5B-Instruct	19.12	0.466	52.97	0.851	0.806	0.811	0.807	1.464
		Qwen2.5-1.5B-Instruct	24.15	0.529	57.45	0.892	0.830	0.843	0.835	1.692
		salamandra-2b-instruct	15.41	0.425	49.44	0.793	0.781	0.794	0.785	1.331
5	zero	Llama-3.2-1B-Instruct	9.87	0.372	45.08	0.713	0.707	0.776	0.738	2.272
		Qwen2.5-0.5B-Instruct	10.49	0.332	42.43	0.694	0.732	0.754	0.741	2.273
		Qwen2.5-1.5B-Instruct	14.16	0.474	50.15	0.822	0.736	0.813	0.771	3.847
		salamandra-2b-instruct	9.50	0.326	42.24	0.675	0.712	0.746	0.727	1.976
	one	Llama-3.2-1B-Instruct	17.45	0.395	46.22	0.807	0.807	0.786	0.794	0.924
		Qwen2.5-0.5B-Instruct	14.72	0.383	47.00	0.806	0.791	0.783	0.785	1.618
		Qwen2.5-1.5B-Instruct	23.81	0.511	57.31	0.892	0.821	0.831	0.825	2.011
		salamandra-2b-instruct	11.36	0.356	44.76	0.707	0.735	0.758	0.744	1.750
	few	Llama-3.2-1B-Instruct	17.91	0.411	47.56	0.823	0.810	0.787	0.797	0.948
		Qwen2.5-0.5B-Instruct	17.92	0.442	51.75	0.849	0.793	0.802	0.796	1.839
		Qwen2.5-1.5B-Instruct	23.86	0.514	57.50	0.893	0.824	0.834	0.828	2.050
		salamandra-2b-instruct	13.84	0.387	46.90	0.770	0.759	0.776	0.765	1.623
6	zero	Llama-3.2-1B-Instruct	10.01	0.384	45.60	0.692	0.710	0.773	0.739	2.569
		Qwen2.5-0.5B-Instruct	12.11	0.368	46.54	0.764	0.747	0.764	0.754	2.529
		Qwen2.5-1.5B-Instruct	15.92	0.489	52.28	0.825	0.744	0.813	0.775	4.234
		salamandra-2b-instruct	9.91	0.316	42.35	0.679	0.717	0.737	0.725	2.044
	one	Llama-3.2-1B-Instruct	15.76	0.361	42.46	0.796	0.805	0.771	0.787	1.052
		Qwen2.5-0.5B-Instruct	13.91	0.382	47.09	0.807	0.782	0.776	0.778	2.029
		Qwen2.5-1.5B-Instruct	22.33	0.498	55.87	0.889	0.811	0.825	0.817	2.522
		salamandra-2b-instruct	12.37	0.370	46.07	0.745	0.746	0.763	0.753	1.987
	few	Llama-3.2-1B-Instruct	17.40	0.395	46.32	0.825	0.817	0.785	0.799	1.078
		Qwen2.5-0.5B-Instruct	18.07	0.437	51.70	0.847	0.788	0.792	0.789	2.130
		Qwen2.5-1.5B-Instruct	24.09	0.509	57.19	0.898	0.822	0.831	0.826	2.457
		salamandra-2b-instruct	15.32	0.401	48.83	0.793	0.762	0.776	0.767	1.858
7	zero	Llama-3.2-1B-Instruct	12.31	0.390	48.05	0.733	0.725	0.777	0.748	2.615
		Qwen2.5-0.5B-Instruct	13.09	0.376	46.83	0.772	0.745	0.761	0.752	2.866
		Qwen2.5-1.5B-Instruct	17.81	0.498	54.56	0.841	0.749	0.814	0.778	4.663
		salamandra-2b-instruct	10.42	0.320	42.34	0.678	0.717	0.738	0.725	2.332
	one	Llama-3.2-1B-Instruct	15.66	0.349	41.92	0.798	0.813	0.765	0.787	1.052
		Qwen2.5-0.5B-Instruct	15.04	0.382	47.66	0.800	0.776	0.772	0.772	2.429
		Qwen2.5-1.5B-Instruct	23.18	0.500	57.91	0.890	0.810	0.825	0.817	2.894
		salamandra-2b-instruct	13.15	0.338	43.34	0.694	0.745	0.741	0.742	1.900
	few	Llama-3.2-1B-Instruct	17.92	0.379	45.80	0.813	0.818	0.777	0.796	1.159
		Qwen2.5-0.5B-Instruct	19.34	0.445	52.55	0.845	0.791	0.787	0.788	2.354
		Qwen2.5-1.5B-Instruct	25.71	0.511	58.68	0.895	0.825	0.828	0.825	2.694
		salamandra-2b-instruct	15.05	0.365	47.60	0.785	0.767	0.766	0.766	1.925

Table 10
Fine tuning Spanish test 1 full evaluation results

Triple size	Model	Metrics							Average time per triple set in seconds
		BLEU	METEOR	CHR++	Cosine Similarity	BERTScore Precision	BERTScore Recall	BERTScore F1	
1	Llama-3.2-1B-Instruct	22,13	0,494	45,44	0,843	0,694	0,817	0,741	2,47
	Qwen2.5-0.5B-Instruct	35,46	0,623	65,31	0,891	0,882	0,888	0,884	0,48
	Qwen2.5-1.5B-Instruct	41,06	0,678	69,67	0,911	0,897	0,906	0,900	0,57
	salamandra-2b-instruct	33,46	0,606	65,09	0,893	0,882	0,889	0,885	0,46
2	Llama-3.2-1B-Instruct	13,56	0,460	40,95	0,837	0,659	0,789	0,712	3,29
	Qwen2.5-0.5B-Instruct	31,31	0,610	63,27	0,904	0,867	0,868	0,866	0,78
	Qwen2.5-1.5B-Instruct	34,99	0,662	67,34	0,920	0,879	0,885	0,881	0,93
	salamandra-2b-instruct	29,23	0,591	62,06	0,903	0,864	0,863	0,862	0,67
3	Llama-3.2-1B-Instruct	12,15	0,466	41,25	0,840	0,652	0,776	0,704	3,76
	Qwen2.5-0.5B-Instruct	31,94	0,619	63,96	0,903	0,856	0,861	0,857	1,03
	Qwen2.5-1.5B-Instruct	34,42	0,652	66,25	0,923	0,867	0,871	0,868	1,24
	salamandra-2b-instruct	28,53	0,584	61,68	0,903	0,854	0,852	0,852	0,88
4	Llama-3.2-1B-Instruct	10,89	0,467	42,74	0,847	0,651	0,764	0,700	4,00
	Qwen2.5-0.5B-Instruct	27,22	0,576	60,86	0,891	0,841	0,847	0,843	1,39
	Qwen2.5-1.5B-Instruct	31,10	0,615	63,36	0,912	0,857	0,856	0,855	1,58
	salamandra-2b-instruct	25,81	0,535	57,88	0,887	0,839	0,833	0,835	1,09
5	Llama-3.2-1B-Instruct	11,27	0,476	44,89	0,864	0,660	0,763	0,705	4,14
	Qwen2.5-0.5B-Instruct	27,45	0,562	60,35	0,892	0,834	0,835	0,833	1,71
	Qwen2.5-1.5B-Instruct	30,34	0,588	62,22	0,918	0,850	0,850	0,849	1,94
	salamandra-2b-instruct	23,69	0,515	56,75	0,892	0,834	0,825	0,828	1,30
6	Llama-3.2-1B-Instruct	10,81	0,484	46,79	0,878	0,661	0,757	0,704	4,51
	Qwen2.5-0.5B-Instruct	28,61	0,566	60,93	0,901	0,835	0,832	0,832	2,08
	Qwen2.5-1.5B-Instruct	30,11	0,595	63,03	0,915	0,847	0,844	0,845	2,31
	salamandra-2b-instruct	24,49	0,515	57,12	0,889	0,832	0,820	0,825	1,51
7	Llama-3.2-1B-Instruct	13,36	0,518	51,24	0,886	0,682	0,764	0,719	4,53
	Qwen2.5-0.5B-Instruct	29,76	0,569	62,82	0,899	0,827	0,827	0,826	2,32
	Qwen2.5-1.5B-Instruct	34,15	0,602	64,24	0,919	0,849	0,844	0,846	2,53
	salamandra-2b-instruct	23,89	0,493	56,05	0,893	0,830	0,813	0,820	1,61

Table 11
Fine tuning Spanish test 2 full evaluation results

Triple size	Model	Metrics							Average time per triple set in seconds
		BLEU	METEOR	CHR++	Cosine Similarity	BERTScore Precision	BERTScore Recall	BERTScore F1	
1	Llama-3.2-1B-Instruct	24,27	0,516	47,68	0,846	0,710	0,823	0,754	2,33
	Qwen2.5-0.5B-Instruct	34,91	0,623	65,43	0,894	0,882	0,889	0,884	0,47
	Qwen2.5-1.5B-Instruct	40,23	0,668	68,89	0,909	0,894	0,903	0,897	0,56
	salamandra-2b-instruct	32,98	0,605	64,79	0,896	0,877	0,885	0,880	0,45
2	Llama-3.2-1B-Instruct	12,81	0,450	39,79	0,833	0,651	0,782	0,704	3,26
	Qwen2.5-0.5B-Instruct	33,14	0,639	65,49	0,910	0,872	0,876	0,873	0,77
	Qwen2.5-1.5B-Instruct	35,63	0,663	67,55	0,921	0,879	0,883	0,879	0,95
	salamandra-2b-instruct	28,97	0,592	62,39	0,904	0,862	0,863	0,861	0,68
3	Llama-3.2-1B-Instruct	10,99	0,456	40,43	0,840	0,646	0,774	0,700	3,75
	Qwen2.5-0.5B-Instruct	31,87	0,626	63,98	0,907	0,857	0,862	0,858	1,06
	Qwen2.5-1.5B-Instruct	36,11	0,656	67,12	0,921	0,870	0,873	0,870	1,24
	salamandra-2b-instruct	27,75	0,572	60,89	0,900	0,851	0,849	0,849	0,88
4	Llama-3.2-1B-Instruct	10,25	0,462	41,59	0,846	0,647	0,763	0,697	4,04
	Qwen2.5-0.5B-Instruct	27,78	0,572	60,78	0,891	0,841	0,845	0,842	1,36
	Qwen2.5-1.5B-Instruct	31,42	0,614	63,81	0,911	0,857	0,858	0,857	1,58
	salamandra-2b-instruct	24,65	0,532	58,05	0,889	0,839	0,833	0,835	1,10
5	Llama-3.2-1B-Instruct	10,92	0,467	44,57	0,861	0,654	0,763	0,701	4,34
	Qwen2.5-0.5B-Instruct	27,07	0,560	60,25	0,893	0,837	0,837	0,836	1,64
	Qwen2.5-1.5B-Instruct	30,56	0,607	63,57	0,918	0,854	0,853	0,853	1,94
	salamandra-2b-instruct	24,11	0,511	56,84	0,889	0,835	0,825	0,829	1,29
6	Llama-3.2-1B-Instruct	12,35	0,503	48,83	0,879	0,676	0,767	0,716	4,49
	Qwen2.5-0.5B-Instruct	27,70	0,560	60,48	0,890	0,834	0,833	0,833	2,00
	Qwen2.5-1.5B-Instruct	30,78	0,591	62,96	0,919	0,850	0,848	0,848	2,28
	salamandra-2b-instruct	24,29	0,502	56,76	0,890	0,833	0,819	0,825	1,50
7	Llama-3.2-1B-Instruct	12,96	0,510	50,89	0,895	0,681	0,764	0,718	4,53
	Qwen2.5-0.5B-Instruct	29,75	0,561	61,65	0,894	0,825	0,824	0,824	2,31
	Qwen2.5-1.5B-Instruct	33,87	0,587	64,44	0,923	0,846	0,839	0,842	2,62
	salamandra-2b-instruct	22,90	0,473	55,23	0,886	0,828	0,809	0,817	1,58

Appendix C. Complete English evaluation results

Table 12
 Prompt-based learning English test 1 full evaluation results

Triple size	Prompt type	Model	Metrics						Average time per triple set in seconds	
			BLEU	METEOR	CHRF++	Cosine Similarity	BERTScore Precision	BERTScore Recall		BERTScore F1
1	zero	Llama-3.2-1B-Instruct	8.71	0.356	36.35	0.694	0.676	0.808	0.734	1.640
		Qwen2.5-0.5B-Instruct	12.96	0.391	42.90	0.682	0.732	0.805	0.764	1.363
		Qwen2.5-1.5B-Instruct	11.90	0.419	38.47	0.772	0.693	0.825	0.750	2.948
		salamandra-2b-instruct	16.48	0.428	47.30	0.729	0.745	0.809	0.773	0.962
		Llama-3.2-1B-Instruct	19.69	0.534	54.88	0.794	0.783	0.855	0.815	0.615
		Qwen2.5-0.5B-Instruct	22.15	0.539	57.30	0.836	0.811	0.861	0.833	0.657
	one	Qwen2.5-1.5B-Instruct	28.29	0.597	62.51	0.884	0.854	0.888	0.869	0.762
		salamandra-2b-instruct	15.37	0.424	47.86	0.731	0.752	0.817	0.780	0.975
		Llama-3.2-1B-Instruct	26.08	0.599	60.52	0.854	0.819	0.876	0.845	0.578
		Qwen2.5-0.5B-Instruct	27.96	0.574	60.82	0.856	0.840	0.874	0.855	0.600
		Qwen2.5-1.5B-Instruct	31.24	0.612	64.41	0.897	0.862	0.892	0.875	0.708
		salamandra-2b-instruct	21.35	0.501	54.33	0.821	0.810	0.854	0.829	0.740
2	zero	Llama-3.2-1B-Instruct	9.81	0.391	41.60	0.688	0.699	0.802	0.745	1.625
		Qwen2.5-0.5B-Instruct	12.01	0.386	42.91	0.663	0.730	0.784	0.753	1.738
		Qwen2.5-1.5B-Instruct	13.91	0.484	47.12	0.809	0.735	0.835	0.779	2.540
		salamandra-2b-instruct	13.53	0.411	46.28	0.719	0.748	0.791	0.766	1.176
		Llama-3.2-1B-Instruct	21.53	0.505	54.61	0.817	0.809	0.834	0.819	0.665
		Qwen2.5-0.5B-Instruct	22.13	0.509	55.66	0.850	0.832	0.843	0.835	0.796
	one	Qwen2.5-1.5B-Instruct	27.23	0.594	62.06	0.906	0.856	0.873	0.863	1.026
		salamandra-2b-instruct	13.61	0.413	46.62	0.713	0.749	0.790	0.767	1.234
		Llama-3.2-1B-Instruct	24.52	0.544	57.86	0.862	0.829	0.851	0.838	0.678
		Qwen2.5-0.5B-Instruct	24.21	0.535	58.23	0.871	0.839	0.853	0.844	0.826
		Qwen2.5-1.5B-Instruct	26.72	0.588	61.47	0.904	0.851	0.873	0.860	1.052
		salamandra-2b-instruct	20.98	0.499	55.08	0.833	0.812	0.832	0.820	0.907
3	zero	Llama-3.2-1B-Instruct	9.74	0.383	43.03	0.692	0.700	0.785	0.738	1.784
		Qwen2.5-0.5B-Instruct	10.70	0.367	41.73	0.623	0.713	0.765	0.735	2.020
		Qwen2.5-1.5B-Instruct	14.82	0.501	49.51	0.808	0.736	0.827	0.776	2.863
		salamandra-2b-instruct	11.46	0.374	44.00	0.689	0.723	0.769	0.743	1.506
		Llama-3.2-1B-Instruct	20.70	0.471	52.29	0.819	0.810	0.819	0.813	0.787
		Qwen2.5-0.5B-Instruct	20.39	0.472	53.13	0.841	0.819	0.823	0.820	1.031
	one	Qwen2.5-1.5B-Instruct	26.52	0.568	59.99	0.899	0.843	0.858	0.849	1.350
		salamandra-2b-instruct	13.03	0.408	46.94	0.723	0.743	0.780	0.759	1.360
		Llama-3.2-1B-Instruct	23.91	0.520	56.03	0.852	0.825	0.835	0.828	0.780
		Qwen2.5-0.5B-Instruct	23.13	0.517	55.87	0.863	0.825	0.832	0.827	1.105
		Qwen2.5-1.5B-Instruct	27.93	0.570	60.45	0.900	0.846	0.860	0.852	1.318
		salamandra-2b-instruct	17.26	0.458	51.82	0.805	0.791	0.808	0.797	1.071
4	zero	Llama-3.2-1B-Instruct	9.38	0.372	43.65	0.701	0.704	0.781	0.738	2.028
		Qwen2.5-0.5B-Instruct	10.36	0.355	42.77	0.681	0.725	0.761	0.741	2.113
		Qwen2.5-1.5B-Instruct	14.59	0.488	50.71	0.822	0.745	0.822	0.779	3.082
		salamandra-2b-instruct	9.53	0.338	41.53	0.659	0.713	0.751	0.729	1.807
		Llama-3.2-1B-Instruct	19.06	0.431	48.54	0.812	0.805	0.799	0.800	0.859
		Qwen2.5-0.5B-Instruct	16.42	0.419	49.19	0.834	0.806	0.804	0.803	1.298
	one	Qwen2.5-1.5B-Instruct	23.02	0.524	57.15	0.890	0.827	0.841	0.832	1.724
		salamandra-2b-instruct	11.74	0.369	44.64	0.712	0.742	0.765	0.751	1.483
		Llama-3.2-1B-Instruct	20.91	0.460	51.85	0.842	0.814	0.811	0.810	0.902
		Qwen2.5-0.5B-Instruct	19.36	0.473	53.31	0.848	0.805	0.814	0.808	1.457
		Qwen2.5-1.5B-Instruct	24.46	0.539	58.10	0.896	0.832	0.843	0.836	1.686
		salamandra-2b-instruct	14.98	0.415	48.45	0.786	0.774	0.789	0.779	1.356
5	zero	Llama-3.2-1B-Instruct	9.80	0.371	44.15	0.702	0.705	0.771	0.735	2.247
		Qwen2.5-0.5B-Instruct	10.46	0.331	41.61	0.687	0.728	0.750	0.736	2.316
		Qwen2.5-1.5B-Instruct	14.89	0.474	50.78	0.822	0.741	0.813	0.774	3.699
		salamandra-2b-instruct	9.85	0.329	41.39	0.663	0.716	0.740	0.726	1.781
		Llama-3.2-1B-Instruct	16.75	0.377	45.12	0.802	0.804	0.782	0.791	0.899
		Qwen2.5-0.5B-Instruct	14.94	0.397	48.14	0.815	0.791	0.787	0.788	1.695
	one	Qwen2.5-1.5B-Instruct	23.96	0.514	58.08	0.896	0.824	0.833	0.828	2.015
		salamandra-2b-instruct	10.55	0.341	43.73	0.702	0.729	0.751	0.738	1.759
		Llama-3.2-1B-Instruct	18.97	0.421	47.92	0.823	0.811	0.791	0.799	0.937
		Qwen2.5-0.5B-Instruct	17.60	0.438	51.42	0.842	0.794	0.797	0.794	1.773
		Qwen2.5-1.5B-Instruct	24.29	0.516	57.45	0.892	0.826	0.834	0.829	2.020
		salamandra-2b-instruct	14.39	0.385	47.19	0.778	0.761	0.777	0.767	1.660
6	zero	Llama-3.2-1B-Instruct	10.34	0.390	45.82	0.701	0.704	0.775	0.736	2.669
		Qwen2.5-0.5B-Instruct	10.82	0.348	45.09	0.744	0.742	0.758	0.748	2.570
		Qwen2.5-1.5B-Instruct	16.51	0.482	52.82	0.838	0.749	0.812	0.778	4.023
		salamandra-2b-instruct	11.10	0.347	43.98	0.706	0.724	0.750	0.735	2.098
		Llama-3.2-1B-Instruct	16.88	0.378	44.36	0.805	0.808	0.777	0.791	1.045
		Qwen2.5-0.5B-Instruct	15.03	0.398	48.19	0.821	0.779	0.778	0.777	1.994
	one	Qwen2.5-1.5B-Instruct	22.28	0.501	56.46	0.891	0.814	0.826	0.819	2.475
		salamandra-2b-instruct	12.05	0.353	44.27	0.738	0.738	0.755	0.744	2.019
		Llama-3.2-1B-Instruct	17.78	0.400	46.37	0.834	0.823	0.785	0.802	1.027
		Qwen2.5-0.5B-Instruct	18.76	0.442	51.84	0.840	0.793	0.795	0.793	2.153
		Qwen2.5-1.5B-Instruct	24.34	0.496	56.60	0.895	0.822	0.828	0.824	2.413
		salamandra-2b-instruct	13.97	0.386	47.30	0.789	0.767	0.771	0.767	1.696
7	zero	Llama-3.2-1B-Instruct	11.33	0.375	46.32	0.687	0.711	0.766	0.736	2.778
		Qwen2.5-0.5B-Instruct	13.36	0.360	46.86	0.727	0.744	0.757	0.749	2.650
		Qwen2.5-1.5B-Instruct	18.25	0.496	55.63	0.846	0.759	0.815	0.785	4.133
		salamandra-2b-instruct	10.85	0.331	42.57	0.702	0.714	0.740	0.726	2.514
		Llama-3.2-1B-Instruct	16.40	0.360	42.44	0.801	0.814	0.770	0.790	1.037
		Qwen2.5-0.5B-Instruct	15.31	0.384	48.05	0.812	0.777	0.774	0.774	2.241
	one	Qwen2.5-1.5B-Instruct	23.85	0.506	57.81	0.890	0.813	0.825	0.818	2.771
		salamandra-2b-instruct	13.06	0.353	46.84	0.737	0.742	0.751	0.745	1.986
		Llama-3.2-1B-Instruct	16.46	0.359	43.05	0.804	0.809	0.768	0.787	1.086
		Qwen2.5-0.5B-Instruct	20.96	0.450	53.32	0.859	0.800	0.794	0.796	2.230
		Qwen2.5-1.5B-Instruct	24.29	0.495	57.91	0.893	0.817	0.824	0.820	2.739
		salamandra-2b-instruct	13.51	0.348	44.05	0.752	0.751	0.751	0.749	1.831

Table 13
 Prompt-based learning English test 2 full evaluation results

Triple size	Prompt type	Model	Metrics							Average time per triple set in seconds
			BLEU	METEOR	CHRFP++	Cosine Similarity	BERTScore Precision	BERTScore Recall	BERTScore F1	
1	zero	Llama-3.2-1B-Instruct	10.22	0.377	36.87	0.564	0.853	0.898	0.873	1.015
		Qwen2.5-0.5B-Instruct	13.92	0.541	49.73	0.808	0.882	0.936	0.907	1.111
		Qwen2.5-1.5B-Instruct	23.18	0.588	53.62	0.867	0.894	0.944	0.917	1.656
		salamandra-2b-instruct	17.09	0.502	48.91	0.770	0.865	0.920	0.890	1.607
	one	Llama-3.2-1B-Instruct	33.32	0.726	67.62	0.887	0.925	0.949	0.936	0.440
		Qwen2.5-0.5B-Instruct	19.97	0.634	58.60	0.848	0.902	0.945	0.922	0.776
		Qwen2.5-1.5B-Instruct	49.53	0.798	78.05	0.955	0.958	0.964	0.960	0.543
		salamandra-2b-instruct	19.81	0.574	55.87	0.780	0.890	0.926	0.907	0.945
	few	Llama-3.2-1B-Instruct	40.70	0.770	73.50	0.928	0.944	0.955	0.949	0.365
		Qwen2.5-0.5B-Instruct	28.08	0.685	65.49	0.880	0.919	0.948	0.933	0.654
		Qwen2.5-1.5B-Instruct	44.93	0.769	76.48	0.951	0.954	0.961	0.957	0.553
		salamandra-2b-instruct	23.23	0.607	59.08	0.821	0.905	0.933	0.918	0.879
2	zero	Llama-3.2-1B-Instruct	12.60	0.484	47.88	0.698	0.862	0.913	0.886	1.218
		Qwen2.5-0.5B-Instruct	16.50	0.552	53.25	0.828	0.897	0.933	0.914	1.339
		Qwen2.5-1.5B-Instruct	28.07	0.685	63.99	0.906	0.920	0.950	0.934	1.323
		salamandra-2b-instruct	13.81	0.492	48.55	0.758	0.877	0.912	0.893	1.660
	one	Llama-3.2-1B-Instruct	28.25	0.630	62.02	0.885	0.934	0.934	0.933	0.529
		Qwen2.5-0.5B-Instruct	23.96	0.638	61.96	0.878	0.922	0.942	0.931	0.922
		Qwen2.5-1.5B-Instruct	40.07	0.758	73.27	0.949	0.951	0.957	0.953	0.863
		salamandra-2b-instruct	19.87	0.570	56.87	0.816	0.899	0.922	0.909	1.066
	few	Llama-3.2-1B-Instruct	33.10	0.680	66.54	0.916	0.945	0.942	0.943	0.508
		Qwen2.5-0.5B-Instruct	30.13	0.674	66.17	0.903	0.931	0.945	0.937	0.834
		Qwen2.5-1.5B-Instruct	39.73	0.752	73.16	0.949	0.952	0.957	0.954	0.875
		salamandra-2b-instruct	23.73	0.588	60.36	0.846	0.913	0.928	0.919	0.901
3	zero	Llama-3.2-1B-Instruct	12.17	0.510	51.59	0.733	0.857	0.918	0.886	1.544
		Qwen2.5-0.5B-Instruct	16.07	0.534	53.20	0.841	0.898	0.924	0.910	1.495
		Qwen2.5-1.5B-Instruct	25.05	0.666	62.73	0.908	0.917	0.944	0.929	1.647
		salamandra-2b-instruct	12.35	0.470	47.89	0.775	0.874	0.906	0.889	1.907
	one	Llama-3.2-1B-Instruct	27.09	0.605	60.39	0.887	0.930	0.929	0.929	0.656
		Qwen2.5-0.5B-Instruct	23.18	0.594	59.18	0.877	0.919	0.931	0.924	1.142
		Qwen2.5-1.5B-Instruct	34.16	0.714	68.96	0.940	0.943	0.949	0.945	1.147
		salamandra-2b-instruct	15.17	0.521	53.28	0.813	0.891	0.913	0.901	1.444
	few	Llama-3.2-1B-Instruct	29.25	0.621	62.16	0.900	0.939	0.931	0.934	0.625
		Qwen2.5-0.5B-Instruct	25.45	0.628	62.19	0.902	0.926	0.936	0.930	1.105
		Qwen2.5-1.5B-Instruct	34.18	0.708	68.59	0.942	0.945	0.949	0.946	1.144
		salamandra-2b-instruct	17.66	0.526	54.98	0.825	0.901	0.916	0.908	1.243
4	zero	Llama-3.2-1B-Instruct	12.02	0.498	51.96	0.747	0.859	0.911	0.884	1.782
		Qwen2.5-0.5B-Instruct	16.20	0.514	53.69	0.850	0.900	0.917	0.908	1.728
		Qwen2.5-1.5B-Instruct	22.92	0.632	61.84	0.904	0.914	0.938	0.925	1.983
		salamandra-2b-instruct	10.13	0.431	46.24	0.765	0.872	0.896	0.883	2.157
	one	Llama-3.2-1B-Instruct	23.45	0.525	54.46	0.862	0.928	0.911	0.919	0.714
		Qwen2.5-0.5B-Instruct	20.31	0.554	57.14	0.875	0.915	0.923	0.918	1.418
		Qwen2.5-1.5B-Instruct	30.47	0.657	65.44	0.928	0.939	0.941	0.939	1.413
		salamandra-2b-instruct	13.79	0.477	51.29	0.809	0.891	0.904	0.897	1.593
	few	Llama-3.2-1B-Instruct	25.36	0.546	56.37	0.881	0.933	0.914	0.923	0.700
		Qwen2.5-0.5B-Instruct	22.82	0.584	59.86	0.892	0.920	0.929	0.924	1.417
		Qwen2.5-1.5B-Instruct	31.62	0.660	66.21	0.930	0.940	0.941	0.940	1.400
		salamandra-2b-instruct	14.34	0.468	51.80	0.825	0.896	0.906	0.900	1.438
5	zero	Llama-3.2-1B-Instruct	12.53	0.493	52.55	0.757	0.857	0.908	0.882	2.008
		Qwen2.5-0.5B-Instruct	16.91	0.509	53.90	0.861	0.901	0.915	0.907	1.870
		Qwen2.5-1.5B-Instruct	23.36	0.617	61.27	0.905	0.912	0.935	0.923	2.367
		salamandra-2b-instruct	11.19	0.422	47.09	0.766	0.874	0.893	0.883	2.191
	one	Llama-3.2-1B-Instruct	20.14	0.472	50.54	0.858	0.926	0.902	0.913	0.776
		Qwen2.5-0.5B-Instruct	18.23	0.524	55.01	0.874	0.911	0.916	0.912	1.680
		Qwen2.5-1.5B-Instruct	29.13	0.626	63.57	0.927	0.936	0.936	0.935	1.680
		salamandra-2b-instruct	12.65	0.446	49.73	0.802	0.884	0.899	0.891	1.940
	few	Llama-3.2-1B-Instruct	21.27	0.480	51.41	0.862	0.930	0.904	0.916	0.750
		Qwen2.5-0.5B-Instruct	22.24	0.556	58.50	0.890	0.917	0.922	0.919	1.657
		Qwen2.5-1.5B-Instruct	29.11	0.619	63.50	0.927	0.937	0.936	0.936	1.675
		salamandra-2b-instruct	14.33	0.455	51.44	0.832	0.893	0.901	0.896	1.666
6	zero	Llama-3.2-1B-Instruct	12.59	0.497	54.48	0.771	0.858	0.905	0.880	2.267
		Qwen2.5-0.5B-Instruct	16.34	0.482	53.27	0.874	0.903	0.909	0.905	2.136
		Qwen2.5-1.5B-Instruct	20.67	0.594	60.16	0.907	0.909	0.933	0.921	2.882
		salamandra-2b-instruct	10.31	0.395	45.73	0.793	0.875	0.887	0.880	2.373
	one	Llama-3.2-1B-Instruct	17.90	0.434	48.00	0.859	0.925	0.894	0.909	0.819
		Qwen2.5-0.5B-Instruct	19.37	0.517	55.21	0.879	0.912	0.912	0.911	1.904
		Qwen2.5-1.5B-Instruct	27.61	0.617	63.06	0.926	0.934	0.934	0.934	2.034
		salamandra-2b-instruct	12.09	0.432	48.84	0.815	0.887	0.895	0.890	2.081
	few	Llama-3.2-1B-Instruct	17.60	0.432	47.35	0.852	0.929	0.891	0.909	0.791
		Qwen2.5-0.5B-Instruct	21.12	0.535	56.73	0.890	0.915	0.917	0.915	1.950
		Qwen2.5-1.5B-Instruct	28.24	0.618	62.88	0.923	0.934	0.933	0.933	2.051
		salamandra-2b-instruct	12.85	0.424	49.17	0.816	0.887	0.895	0.890	2.060
7	zero	Llama-3.2-1B-Instruct	14.29	0.495	56.13	0.782	0.861	0.901	0.880	2.415
		Qwen2.5-0.5B-Instruct	19.30	0.510	55.66	0.886	0.904	0.908	0.905	2.476
		Qwen2.5-1.5B-Instruct	23.90	0.617	63.08	0.909	0.913	0.931	0.921	3.058
		salamandra-2b-instruct	11.85	0.407	47.00	0.768	0.869	0.881	0.874	2.761
	one	Llama-3.2-1B-Instruct	15.56	0.395	43.75	0.824	0.927	0.884	0.904	0.857
		Qwen2.5-0.5B-Instruct	21.08	0.521	56.60	0.890	0.912	0.910	0.911	2.130
		Qwen2.5-1.5B-Instruct	29.91	0.616	64.71	0.920	0.934	0.931	0.932	2.338
		salamandra-2b-instruct	13.20	0.425	48.59	0.806	0.882	0.888	0.884	2.410
	few	Llama-3.2-1B-Instruct	15.94	0.397	43.23	0.817	0.925	0.877	0.900	0.797
		Qwen2.5-0.5B-Instruct	24.40	0.553	59.61	0.897	0.920	0.917	0.918	2.076
		Qwen2.5-1.5B-Instruct	30.05	0.614	64.30	0.926	0.935	0.930	0.932	2.278
		salamandra-2b-instruct	13.64	0.415	49.42	0.828	0.889	0.890	0.888	2.134

Table 14
Fine tuning English test 1 full evaluation results

Triple size	Model	Metrics							Average time per triple set in seconds
		BLEU	METEOR	CHRFP++	Cosine Similarity	BERTScore Precision	BERTScore Recall	BERTScore F1	
1	Llama-3.2-1B-Instruct	56.38	0.819	81.27	0.968	0.966	0.967	0.966	0.264
	Qwen2.5-0.5B-Instruct	49.51	0.779	77.83	0.950	0.958	0.959	0.958	0.404
	Qwen2.5-1.5B-Instruct	47.92	0.789	78.79	0.959	0.957	0.960	0.958	0.482
	salamandra-2b-instruct	40.31	0.743	75.06	0.949	0.947	0.952	0.949	0.436
2	Llama-3.2-1B-Instruct	44.21	0.776	75.61	0.964	0.957	0.958	0.957	0.449
	Qwen2.5-0.5B-Instruct	40.34	0.750	73.96	0.956	0.950	0.951	0.950	0.642
	Qwen2.5-1.5B-Instruct	39.78	0.757	74.19	0.963	0.952	0.955	0.952	0.785
	salamandra-2b-instruct	33.35	0.703	70.43	0.943	0.939	0.943	0.940	0.669
3	Llama-3.2-1B-Instruct	38.89	0.741	72.69	0.950	0.948	0.952	0.950	0.633
	Qwen2.5-0.5B-Instruct	35.16	0.710	69.84	0.941	0.945	0.945	0.945	0.843
	Qwen2.5-1.5B-Instruct	36.40	0.725	71.49	0.952	0.945	0.948	0.946	1.038
	salamandra-2b-instruct	28.32	0.659	66.89	0.934	0.930	0.934	0.932	0.890
4	Llama-3.2-1B-Instruct	34.78	0.701	70.21	0.937	0.941	0.946	0.943	0.841
	Qwen2.5-0.5B-Instruct	32.19	0.670	67.66	0.933	0.940	0.939	0.939	1.113
	Qwen2.5-1.5B-Instruct	32.56	0.677	68.22	0.940	0.940	0.942	0.940	1.312
	salamandra-2b-instruct	24.20	0.609	63.28	0.919	0.926	0.928	0.926	1.126
5	Llama-3.2-1B-Instruct	34.24	0.687	69.54	0.937	0.939	0.943	0.941	1.002
	Qwen2.5-0.5B-Instruct	31.37	0.644	66.06	0.931	0.940	0.936	0.938	1.277
	Qwen2.5-1.5B-Instruct	32.07	0.663	67.38	0.936	0.938	0.940	0.938	1.620
	salamandra-2b-instruct	24.40	0.579	61.11	0.910	0.924	0.925	0.924	1.294
6	Llama-3.2-1B-Instruct	32.93	0.665	68.23	0.933	0.936	0.938	0.936	1.225
	Qwen2.5-0.5B-Instruct	29.81	0.621	64.16	0.920	0.937	0.932	0.934	1.563
	Qwen2.5-1.5B-Instruct	29.86	0.637	65.64	0.930	0.935	0.935	0.934	2.028
	salamandra-2b-instruct	23.15	0.564	59.99	0.904	0.924	0.921	0.922	1.498
7	Llama-3.2-1B-Instruct	34.33	0.664	69.31	0.932	0.936	0.936	0.936	1.436
	Qwen2.5-0.5B-Instruct	30.59	0.621	65.24	0.918	0.936	0.929	0.932	1.730
	Qwen2.5-1.5B-Instruct	33.16	0.646	67.70	0.931	0.941	0.937	0.939	2.198
	salamandra-2b-instruct	25.56	0.557	60.57	0.902	0.924	0.918	0.921	1.761

Table 15
Fine tuning English test 2 full evaluation results

Triple size	Model	Metrics							Average time per triple set in seconds
		BLEU	METEOR	CHRFP++	Cosine Similarity	BERTScore Precision	BERTScore Recall	BERTScore F1	
1	Llama-3.2-1B-Instruct	56.63	0.815	81.41	0.970	0.965	0.966	0.965	0.272
	Qwen2.5-0.5B-Instruct	48.26	0.776	77.54	0.956	0.958	0.959	0.958	0.404
	Qwen2.5-1.5B-Instruct	49.09	0.796	79.31	0.960	0.958	0.961	0.959	0.474
	salamandra-2b-instruct	39.73	0.735	74.34	0.940	0.945	0.950	0.947	0.431
2	Llama-3.2-1B-Instruct	43.46	0.779	76.00	0.964	0.957	0.959	0.957	0.458
	Qwen2.5-0.5B-Instruct	39.46	0.743	73.28	0.952	0.949	0.951	0.949	0.653
	Qwen2.5-1.5B-Instruct	40.05	0.764	74.55	0.961	0.951	0.954	0.952	0.769
	salamandra-2b-instruct	33.04	0.703	70.28	0.938	0.938	0.943	0.940	0.680
3	Llama-3.2-1B-Instruct	38.75	0.741	72.45	0.950	0.947	0.951	0.948	0.670
	Qwen2.5-0.5B-Instruct	35.72	0.720	70.32	0.944	0.945	0.946	0.945	0.884
	Qwen2.5-1.5B-Instruct	37.41	0.731	72.15	0.955	0.946	0.949	0.947	1.051
	salamandra-2b-instruct	27.66	0.649	65.95	0.933	0.932	0.935	0.933	0.879
4	Llama-3.2-1B-Instruct	35.77	0.711	70.47	0.938	0.942	0.946	0.943	0.859
	Qwen2.5-0.5B-Instruct	31.51	0.669	67.67	0.931	0.939	0.939	0.939	1.133
	Qwen2.5-1.5B-Instruct	33.03	0.685	68.79	0.944	0.942	0.943	0.942	1.329
	salamandra-2b-instruct	25.37	0.621	63.74	0.920	0.926	0.928	0.926	1.105
5	Llama-3.2-1B-Instruct	34.24	0.681	69.07	0.934	0.938	0.941	0.939	1.020
	Qwen2.5-0.5B-Instruct	30.17	0.644	65.65	0.928	0.937	0.935	0.935	1.324
	Qwen2.5-1.5B-Instruct	31.21	0.652	66.78	0.935	0.939	0.938	0.938	1.580
	salamandra-2b-instruct	24.01	0.570	61.09	0.906	0.924	0.924	0.923	1.309
6	Llama-3.2-1B-Instruct	32.45	0.661	67.91	0.933	0.936	0.938	0.936	1.207
	Qwen2.5-0.5B-Instruct	28.26	0.614	64.08	0.924	0.935	0.932	0.933	1.603
	Qwen2.5-1.5B-Instruct	29.88	0.651	66.06	0.933	0.936	0.937	0.936	1.946
	salamandra-2b-instruct	23.18	0.565	59.81	0.900	0.923	0.920	0.921	1.538
7	Llama-3.2-1B-Instruct	35.19	0.672	69.70	0.929	0.939	0.937	0.937	1.346
	Qwen2.5-0.5B-Instruct	31.38	0.623	66.06	0.922	0.937	0.931	0.934	1.827
	Qwen2.5-1.5B-Instruct	32.71	0.635	67.61	0.931	0.939	0.936	0.937	2.187
	salamandra-2b-instruct	24.87	0.543	60.05	0.892	0.925	0.918	0.921	1.759