The CACAPO Dataset: A Multilingual, Multi-Domain Dataset for Neural Pipeline and End-to-End Data-to-Text Generation

Chris van der Lee
Tilburg University
C.VdrLee@uvt.nl

Chris Emmery
Tilburg University
C.D.Emmery@uvt.nl

Sander Wubben
Tilburg University
S.Wubben@uvt.nl

Emiel Krahmer
Tilburg University
E.J.Krahmer@uvt.nl

Abstract

This paper describes the CACAPO dataset, built for training both neural pipeline and end-to-end data-to-text language generation systems. The dataset is multilingual (Dutch and English), and contains almost 10,000 sentences from human-written news texts in the sports, weather, stocks, and incidents domain, together with aligned attribute-value paired data. The dataset is unique in that the linguistic variation and indirect ways of expressing data in these texts reflect the challenges of real world NLG tasks.

1 Introduction

The current paper presents the **C**ombinations of **A**ligned Data-Senten**c**es from N**a**turally **Pro**duced Texts (hereafter: CACAPO) dataset; a dataset for data-to-text generation (the task of producing adequate, fluent natural language text from non-linguistic structured data, such as database records, spreadsheets, knowledge graphs, tables, etc., Gatt and Krahmer, 2018). The dataset contains sentences from automatically scraped news texts for the sports, weather, stock, and incidents domain in English and Dutch, aligned with relevant attribute-value paired data (see Figure 1 and Appendix A for examples). To our knowledge, this is the first dataset based on 'naturally occurring' human-written texts (i.e., texts that were not collected in a task-based setting), that covers various domains, as well as multiple languages.

Neural Natural Language Generation (NLG) is a promising technique, as neural NLG systems are not bound by any special-purpose mechanisms, and hence are argued to be easily adaptable to other domains and languages (Oraby et al., 2019; Puduppully et al., 2019; van der Lee et al., 2018). Yet despite this advantage, it can still be challenging to create a neural NLG system that achieves the same rich and detailed output as a well-designed traditional rule-based pipeline system (Novikova et al., 2017; van der Lee et al., 2018; Moryossef et al., 2019b). This is because a large-scale parallel dataset (i.e., a dataset with aligned texts and relevant data) is required for training neural NLG systems, and such datasets are not a common natural occurrence. This limitation is especially persistent in neural pipeline architectures: neural architectures modeled after the 'traditional' pipeline architecture (Reiter and Dale, 2000) that sequentially performs tasks related to document planning, sentence planning and linguistic realization (Castro Ferreira et al., 2019). These architectures require an explicit representation for every intermediate step. The (enriched) WebNLG dataset (Gardent et al., 2017a,b; Castro Ferreira et al., 2018) is presently the only other dataset viable for both end-to-end, as well as neural pipeline architectures.

The present paper thus presents a new automatically scraped dataset that can be used for end-to-end, as well as neural pipeline architectures. Furthermore, it describes a collection process inspired by Oraby et al. (2019), where collection starts with the news reports and attribute-value datapoints are constructed from them, which also enables relatively low-effort extension and adaptation of the current dataset (Section 3). Characteristics of the dataset are described based on the methodology by Perez-Beltrachini and Gardent (2017) (Section 4). Finally, a baseline is developed for the dataset using TGen (Dušek and Jurčíček, 2015) (Section 5).

The full dataset is freely available for research purposes upon request, licensed under AusGoal Restrictive Licence. A 'thin' version of the dataset that contains the annotated data in combination with the URLs of the scraped texts and the scraping tools is publicly available via https://github.com/TallChris91/CACAPO-Dataset, licensed under CC BY-NC-SA.

Proceedings of The 13th International Conference on Natural Language Generation, pages 68–79,
Dublin, Ireland, 15-18 December, 2020.

Atribute	Value
pitcherName	CC Sabathia
teamName	Blue Jays
teamName	Yankees
hitNumber	hitless
inningsPitched	five

↓

All CC Sabathia did was hold the Blue Jays hitless over the final five innings to give the Yankees a chance to rally.

Figure 1: Example of a set of attribute-value pairs (top) and corresponding text (bottom).

Tag	Entity
PATIENT-1	CC Sabathia
PATIENT-2	Blue Jays
PATIENT-3	hitless
PATIENT-4	five
PATIENT-5	Yankees

↓

*All **PATIENT-1** did was hold the **PATIENT-2 PATIENT-3** over the final **PATIENT-4** innings to give the **PATIENT-5** a chance to rally.*

Figure 2: Mapping between tags and entities for the corresponding delexicalized template.

2 Related work

Neural data-to-text NLG models have the ability to produce texts without requiring handwritten rules and templates, generating texts in a completely data-driven way. However, neural data-to-text NLG is struggling to overcome two critical bottlenecks, identified by Oraby et al. (2019), that hamper the performance of the models: (1) a **data bottleneck**, a lack of (high quality, large scale) parallel data-text datasets; and (2) a **control bottleneck**, which they describe as an inability to control stylistic variation, but can be more broadly described as the inability to systematically control the generation process and the generated output of a neural system.

2.1 Data bottleneck

The field has started to address the data bottleneck issue recently, exhibited by an increase of parallel data-text corpora. E2E (Novikova et al., 2017), and WebNLG (Gardent et al., 2017a,b) are two prime examples of this. Crowdsourcing techniques were employed for the creation of these datasets, meaning that humans were asked to write a text given an input meaning representation (MR). This makes it feasible to collect ample texts of good quality, but such techniques can quickly become a financial burden, and require significant effort from the researchers to design and assemble (Oraby et al., 2019). This amount of time can be reduced as is shown by the construction of the ToTTo dataset (Parikh et al., 2020), where humans edited existing Wikipedia phrases to reflect a given input MR, rather than writing text phrases from scratch. This still requires significant resources, however.

Compiling a dataset via crowdsourcing usually ensures that texts are a direct verbalization of the aligned data, which limits the amount of noise and inaccuracies present in the datasets. However, people have increasingly started to criticize the realism of these datasets as they are usually not representative of real world scenarios and language use.[1] Verbalizations by crowdsource workers are different from how data is usually verbalized by professional journalists, for instance, whose focus—besides high fidelity—is also on producing fluent and enjoyable texts. Such a focus can result in more indirect descriptions of data or superfluous information. Generating texts from these indirect descriptions may be more challenging as the NLG systems need to learn how to abstract from the 'noise' in these datasets. A different but related problem is that the presence of superfluous descriptions in input make these neural systems more prone to 'hallucinations', i.e., producing output information that is not present in the input data (Reiter, 2018a). However, having a system that performs well on such unedited texts might make it more attainable to develop systems that can be deployed in non-academic settings, as these texts are representative of such settings. Companies for which data-to-text systems may be especially relevant (i.e., press agencies, publishers, weather institutes, etc.), oftentimes have an extensive archive of historical data and human-written texts, that would contain similar types of 'noise' in their data representation.

Therefore, it seems imperative to also pursue other dataset collection techniques—such as text and data collection—by scraping publicly available sources. Datasets that were compiled via this method have also seen a surge recently, with YelpNLG (Oraby et al., 2019), `RotoWire` (Wiseman et al., 2017), and `RotoWire`-inspired datasets like `RotoWire-FG` (Wang, 2019) and MLB (Puduppully et al., 2019). Using this method

[1] See, for instance, the discussion at https://twitter.com/yoavgo/status/1281971375029325824.

enables data and text collection without having to spend as much time and resources as would be necessary with crowdsourcing techniques. However, most of the current automatically scraped datasets are for document-level texts, which generally requires different architectural approaches than the shorter sentence-level or phrase-level texts that are most commonly found in the datasets compiled via crowdsourcing. Furthermore, they are limited to one domain (restaurants, basketball, and baseball, for YelpNLG, RotoWire, and MLB respectively), and one language (English). This makes it difficult to train domain-invariant systems.

2.2 Control bottleneck

Furthermore, most existing datasets are constructed for end-to-end architectures, where the non-lingustic input is converted into natural language without explicit intermediate representations in between (Castro Ferreira et al., 2019). By contrast, researchers have started to experiment with neural pipeline methods, in which the data conversion process happens via one or more explicit intermediate transformations (see, for instance, Castro Ferreira et al., 2019; Jiang et al., 2020; Moryossef et al., 2019a,b). These methods enable the control over parts of the data-to-text conversion process, making it possible to develop hybrid (e.g. rule-based and neural) systems. Additionally, a direct comparison between end-to-end and pipeline approaches suggests that pipeline approaches lead to improved output quality, and decreases data hallucination and data omission; two challenges for datasets compiled using unedited texts from publicly available sources (Castro Ferreira et al., 2019). However, pipeline architectures require a training dataset containing the intermediate representations in order to be trained. And, with the exception of the Enriched WebNLG dataset (Castro Ferreira et al., 2018), there are currently no datasets facilitating such an approach.[2]

2.3 Current work

The current work introduces the CACAPO dataset which addresses the aforementioned limitations of the existing datasets: it contains intermediate representations for discourse ordering, text structuring, lexicalization, referring expression generation, and textual realization for pipeline approaches such as the one by Castro Ferreira et al. (2019). Furthermore, it is a sentence-level dataset containing unedited sentences from news articles written by professional journalists and meteorologists (see Section 3 for details).

Finally, many of the datasets that are commonly used currently lack domain diversity (Radev et al., 2020) and are solely constructed for the English language (with the exception of WebNLG, see Castro Ferreira et al., 2018; Shimorina et al., 2019). The CACAPO dataset contains texts from the sports, weather, stocks, and incidents domain for both Dutch and English.

3 The CACAPO dataset

3.1 Collection methods

Both the Dutch and English version of the CACAPO dataset contain the same four domains (sports, weather, stocks, and incidents) albeit with different events and hence also some topical variety between both languages. For each domain a scraping tool was used or custom built that either fully automatically collected relevant texts, or collected these texts with as little human effort as possible (e.g humans needed to copy the URLs, website source code, or needed to copy some aspects to a custom-built tool).[3] The following texts were collected:

- **Dutch sports** domain texts cover soccer match reports from the 15/16 and 16/17 season of the Dutch *Eredivisie*, the highest professional soccer league in The Netherlands. Texts were scraped from 10 professional news websites using Google search queries for all matches played during the 15/16 and 16/17 seasons (teams and play date). In total, 6,600 texts were scraped (2,101,338 tokens; 27,619 types).

- **Dutch stocks** domain texts cover daily reports on stock exchanges, company stock listings, (crypto)currency exchange rates, and oil prices. These reports were collected from 49 different newspapers using Nexis Uni,[4] covering all reports from January 2019-January 2020. A total of 4,280 texts were collected (1,211,842 tokens; 22,685 types).

- **Dutch weather** domain texts cover several-daily short-term weather forecasts for The

[2] At least, datasets that start from data. Surface realization datasets such as the one employed in (Mille et al., 2019) can be seen as facilitating the pipeline approach.

[3] See https://github.com/TallChris91/CACAPO-Dataset for the collection tools.

[4] http://www.nexisuni.com/

Netherlands from the Royal Netherlands Meteorological Institute (KNMI); the Dutch national weather service. These texts originate from the "complete weather report" prognosis, found on the KNMI website[5]. The weather reports were obtained for all of 2019, totalling 5,897 texts (1,099,556 tokens; 1,076 types).

- **Dutch incidents** domain texts originate from Hendriks (2019) who collected data from https://www.hetongeluk.nl/; an online database for news articles about traffic incidents, which in total contains traffic incident reports from 139 websites from 2013 to 2019. This collection contains 1,600 texts (154,596 tokens; 8,919 types).

- **English sports** domain texts cover baseball reports from the American MLB League, the top league in American professional baseball. The baseball reports were obtained using the scraper made available by Puduppully et al. (2019) to collect their MLB dataset covering 2007-2018. The texts originate from ESPN; an American sports website.[6] A total of 26,393 (12,852,342 tokens; 34,123 types) were collected for this domain.

- **English stocks** domain texts cover the same topics as the Dutch stocks domain texts. The texts were obtained using Google News by searching news items containing "stock index" and "stock market" in the period of January 2019-January 2020. 1,109 texts from 182 websites were collected (621,997 tokens; 23,216 types).

- **English weather** domain texts cover weather forecasts for several countries (e.g., Canada, United States, India, Ireland). The weather forecasts were collected using Google News by searching news items containing "weather forecast" in the period of January 2019-2020. This resulted in a collection of 926 texts from 215 websites (341,622 tokens; 11,426 types).

- **English incidents** domain texts cover gun violence incidents from the Gun Violence Archive,[7] a database on gun violence incidents, which in total contains 3,180 incident reports from 596 websites ranging from 2012 to 2019 (1,105,567 tokens; 26,968 types).

Thus, in total 51,575 texts were collected via these different methods. For the CACAPO dataset, all texts above 325 words were discarded as most basic news reports typically do not exceed that amount of words (Asbreuk et al., 2017), leaving 20,630 texts.[8] From this sample, 200 texts were randomly selected for each language and domain (a total of 1,600 texts; 12.89% of the text selection) to obtain a representative number of sentences while keeping the annotation load reasonable (see Section 3.2). These texts were automatically split into sentences using a sentence tokenizer. SpaCy (Honnibal and Montani, 2017) was used as a tokenizer for the Dutch part, and SoMaJo for the English part (Proisl and Uhrig, 2016). Finally, the sentences were assigned to training, validation, and testing sets in a 76.5, 8.5, 15 ratio—the same ratio that (Novikova et al., 2017) used. All sentences occurring in the selected texts are part of the CACAPO dataset and the order of occurrence of the sentences in a text was preserved in the dataset.

3.2 Data annotation

The data annotation process followed after the sample sentences were tokenized. Sentences were manually aligned with data using Prodigy[9], a data annotation tool (Montani and Honnibal, 2018), in a attribute-value pair format, done by two expert annotators. The annotators annotated a part of the dataset jointly (1,755 sentences), resulting in Cohen's $\kappa = 0.67$ (substantial agreement; Landis and Koch, 1977) and a 70.92% agreement. This agreement was deemed high enough for a single coder per item approach for the rest of the dataset. One of the annotators developed the guidelines with a definition of each category and examples of passages belonging to that category resulting in relatively quick acquisition of the categories. Annotation took between 5 and 15 minutes per text on average.

All annotated attributes can be found in Appendix C. The amount of types that were annotated varied between 10 (Dutch/English stocks domain) and 76 (English sports domain). Which labels to annotate was decided upon by doing a practice set of

[5] https://www.knmi.nl/nederland-nu/weer/verwachtingen

[6] http://www.espn.com/

[7] http://www.gunviolencearchive.org/

[8] The full collection of unlabeled texts and the selection of unlabeled texts is freely available upon request—licensed under AusGoal Restrictive Licence—to facilitate extension of the dataset as well as other tasks, such as information extraction.

[9] https://prodi.gy/

10 texts. All data labels are based on the 5 Ws and 1 H questions (Who, What, When, Where, Why, and How). As most journalism schools teach students to write news articles that focus on answers to the 5 Ws and 1 H questions (Canavilhas, 2007; Kussendrager et al., 2018).

'Who' data is for instance player and referee information for the sports domain (*assistName, goalName, goalkeeperName, pitcherName, pitcherRecord, umpireName*) and suspect and victim information for the incidents domain (*suspectAge, suspectGender, victimBased, victimName, victimOccupation*). Examples of 'What' data are stock price increases and decreases for the stocks domain (*stockChange, stockChangePercentage, stockPoints*), and information about cloudiness, wind, and weather type for the weather domain (*cloudAmount, gustChange, temperatureCelsius, weatherType*). 'When' data types are the (next) match date for the sports domain (*matchDate, matchTime, nextMatchDate*), and the date/time that an incident occurred for the incidents domain (*dateTime, accidentDate*). 'Where' data is the stadium where a match is played for the sports domain (*stadiumPlayed, locationPlayed*), or where weather events will happen for the weather domain (*locationArea*). 'Why' data is for example the cause of a traffic incident for the incidents domain (*incidentCause*). And 'How' data can be information about the way a goal was scored or a ball was hit for the sports domain (*goalType, strikingType)*, and how a traffic/shooting incident took place for the incidents domain (*incidentType, shootingType)*.

3.3 Intermediate representations

After the data annotation process was completed, the annotated data and collected texts were then used to create explicit intermediate representations suitable for neural pipeline architectures. The CACAPO dataset is saved in a similar XML format as the Enriched WebNLG dataset (Castro Ferreira et al., 2018) (see Figure 3 for an example) to enable effortless testing of systems designed for this dataset. This also means that the CACAPO dataset is suitable for pipeline systems that convert data into text using the same 5 sequential steps as Castro Ferreira et al. (2019), which follows the original pipeline architecture of (Reiter and Dale, 2000):

1. **Discourse Ordering** is the task of determining in which order to present the data that should be verbalized in the target text. This can be trained using the MRs found in the (alphabetically ordered) <originaldataset>, and the <sorteddataset> that is ordered based on the appearance of the MR in the sentence. This ordering is determined based on the string position information provided by Prodigy.

2. **Text Structuring** is the task of organizing the ordered triples into paragraphs and sentences. The <sorteddataset> tag also contains sentence information relevant for the Text Structuring step. As the CACAPO dataset is a sentence-level dataset, Text Structuring is not a directly relevant step. Although the sentence information in the <sorteddataset> tag allows for extensions to phrase-level or paragraph-level instances.

3. **Lexicalization** is the task of finding the words and phrases that describe the input data correctly (Reiter and Dale, 2000). This means using the information found in <sorteddataset> to (ideally) generate the string in <lexicalization>, for this dataset. The string found in this tag is a delexicalized version of the original sentence (found in <text>). This <lexicalization> tag not only contains information to se-

```
<entry category="EnglishIncidents" eid="Id2" size="3">
  <originaldataset>
        <odata>victimAge | 22-year-old</odata>
        <odata>victimStatus | grazed in the thigh</odata>
  </originaldataset>
  <lex comment="good" lid="Id1">
        <sorteddataset>
          <sentence ID="1">
                <sdata>victimAge | 22-year-old</sdata>
                <sdata>victimStatus | grazed in the
                    ↪ thigh</sdata>
          </sentence>
        </sorteddataset>
        <references>
          <reference entity="22-year-old" number="1"
                ↪ tag="ENTITY-1"
                ↪ type="description">22-year-old
                ↪ </reference>
          <reference entity="grazed in the thigh"
                ↪ number="2" tag="ENTITY-2"
                ↪ type="description">grazed in the
                ↪ thigh</reference>
        </references>
        <text>A 22-year-old was grazed in the thigh .</text>
        <template>A ENTITY-1 was ENTITY-2 .</template>
        <lexicalization>DT[form=undefined] A ENTITY-1
              ↪ VP[aspect=simple,tense=past,
              ↪ voice=active,person=null, number=singular]
              ↪ be ENTITY-2 .</lexicalization>
  </lex>
</entry>
```

Figure 3: Example of an XML formatted data instance in the CACAPO dataset.

lect accurate words and phrases to describe an MR, but also contains information for the two steps further ahead in the pipeline. The ENTITY-[0-9] placeholders indicate where MRs should be realized. The entity number indicates which MR to realize based on the order in <sorteddataset>. Furthermore, the delexicalized string contains syntactical information. For (lemmatized) verbs, it stores aspect, mood, tense, voice and number in a VP tag. And it stores the form of determiners in a DT tag.

Delexicalization was done by a script that matches the annotated data with the original string, using the string location information provided by Prodigy. The annotation of syntactical information and lemmatization was done using CoreNLP (Manning et al., 2014) for English, and DeepFrog[10] for Dutch.

4. **Referring Expression Generation** is the task of generating the correct entities in a text (Krahmer and van Deemter, 2012). In this step, a system can be trained to fill the ENTITY-[0-9] placeholders found in the <lexicalization> string with the data found in the <references> tag.

5. **Textual Realization** is the task of performing the final steps to convert the non-linguistic data into natural language text. For this dataset, this means converting the lemmatized verbs and determiners to a form that is congruent with the MR, using the VP and DT tags found in <lexicalization>.

Of course, the dataset also lends itself for data-to-text generation in an end-to-end fashion. For this, a system can be trained on the information in <originaldataset> and <text>.

4 Statistics

We compare the CACAPO dataset to the Enriched WebNLG dataset (Castro Ferreira et al., 2018; Gardent et al., 2017a,b), as these datasets are comparable in the sense that both are multilingual, multidomain, and contain explicit intermediate steps that allow for neural pipeline architectures to be employed. However, they are different in the fact that WebNLG is constructed using crowdsourcing, while CACAPO is constructed using unedited texts scraped from publicly available sources. Similar to Novikova et al. (2017) we compare the two datasets on size, lexical richness, and sentence complexity.

4.1 Size

Based on, Novikova et al. (2017) and Perez-Beltrachini and Gardent (2017), we employ the following size metrics to compare the Enriched WebNLG dataset (Castro Ferreira et al., 2018) to our dataset (see Table 1):

- **Number of instances**: Absolute number of texts in the dataset (single sentences for CACAPO, single sentences and multi-sentence phrases for WebNLG). This gives a direct indication of the dataset size.

- **Number of unique MRs**: Number of different MRs appearing in the dataset (set of attribute-value paired data for CACAPO, set of RDF-triple data for WebNLG aligned to a text). Besides dataset size, this also gives an indication of training difficulty: more unique MRs means a greater challenge to train models on the data.

- **Instances per MR**: Average number of verbalizations for one MR. The more references for an MR appear in the training set, the better models can be trained to learn how to verbalize this MR.

- **Slots per MR**: Average number of data points (single attribute-value paired data for CACAPO, single RDF-triples for WebNLG) that compose an MR.

- **Words per instance**: Average number of words appearing in an instance (single sentences for CACAPO, single sentences and multi-sentence phrases for WebNLG).

- **Words per sentence**: Average number of words appearing in a sentence.

- **Sentences per instance**: Average number of sentences appearing in an instance.

The metrics in Table 1 show that the CACAPO dataset and Enriched WebNLG dataset are very similar in size, as displayed by the number of instances and number of unique MRs, with the CACAPO dataset being slightly bigger. Also, in terms of slots per MR, and words per reference,

[10] https://github.com/proycon/deepfrog

	No. of instances	No. of unique MRs	Refs/MR	Slots/MR	W/Inst	W/Sent	Sents/Inst
CACAPO (Dutch)	10,486	8,833	1.19 (1-285)	2.74	15.19	15.19	1 (1-1)
CACAPO (English)	10,566	9,352	1.17 (1-290)	2.83	18.52	18.52	1 (1-1)
WebNLG (English)	9,674	9,604	2.63 (1-12)	2.95	20.03	14.26	1.4 (1-6)
WebNLG (German)	7,812	7,753	2.63 (1-12)	2.96	19.22	13.64	1.4 (1-6)

Table 1: Descriptive statistics for various size-related dimensions.

the CACAPO dataset and the Enriched WebNLG dataset seem comparable. However, on average, there are fewer references for MRs in the CACAPO dataset compared to the WebNLG dataset. This indicates that it would be more challenging for data-to-text generation systems to learn alignments between MRs and text for the CACAPO dataset compared to the WebNLG dataset.

4.2 Lexical Richness

Following Novikova et al. (2017), we investigate various aspects of lexical richness by looking at traditional measures, such as the number of tokens, and types, and type-token ratio (TTR; see Table 2). And we include the more robust mean segmental TTR (MSTTR), which divides the dataset into equal segments of a given token length (here: 25 tokens) and calculates the average TTR of all these segments. Finally, we also include Lexical Sophistication (LS). Also known as Guiraud Advanced (Daller et al., 2003) which gauges the number of unique words in a dataset; another way to measure lexical richness. We calculate the Guiraud Advanced metric by taking the proportion of word types that are not in the top 2,000 most frequent words in large and diverse corpora for each language: the British National Corpus (British National Corpus, 2007), the SoNaR 500 corpus (Oostdijk et al., 2013), and the German Internet corpus (Sharoff, 2006), for the English and Dutch CACAPO dataset, with additional statistics for English and German WebNLG added for comparison, respectively. Each of these corpora contains a large amount of texts and covers a wide array of topics and domains. Therefore, we believe that their top 2,000 most frequent words are representative of the language.

The number of tokens in Table 2 show that the texts of the CACAPO dataset are somewhat smaller than those found in the WebNLG dataset. However, supporting our expectations, the CACAPO dataset is the more lexically varied dataset of the two, as illustrated by the higher TTR and MSTTR scores, and

	Tokens	Types	LS	TTR	MSTTR
CACAPO (NL)	147,770	10,152	0.87	0.07	0.87
CACAPO (EN)	175,860	11,485	0.87	0.07	0.89
WebNLG (EN)	491,731	5,521	0.84	0.01	0.75
WebNLG (DE)	376,184	6,433	0.86	0.02	0.78

Table 2: Size and lexical diversity metrics.

the higher absolute number of types. The higher amount of lexical diversity found in the CACAPO dataset is a further indication that training a data-to-text generation system to produce high quality output may be more challenging for this dataset. The lexical sophistication metric shows a similar proportion of infrequent words in the CACAPO and WebNLG dataset, which suggests that both datasets are similarly diverse in terms of the amount of non-standard language found in the dataset.

Also similar to Novikova et al. (2017), we have analyzed the appearance of bigrams and trigrams in the dataset. Focusing on (1) the proportion of bigrams and trigrams appearing only once in the CACAPO dataset and WebNLG dataset; and (2) on the average frequency of bigrams and trigrams of those that appear more than once. These metrics give further indication of lexical richness: a high amount of unique bigrams and trigrams, and a low average frequency for non-unique bigrams and trigrams makes it more challenging to train a neural data-to-text system.

The results in Table 3 show further evidence that the English and Dutch versions of the CACAPO dataset are more lexically rich compared to the English and German versions of the WebNLG dataset. The CACAPO dataset has a much larger proportion of bigrams and trigrams that appear only once. Furthermore, of the bigrams and trigrams appearing more than once, the average frequency of bigrams and trigrams in the CACAPO dataset is much lower than for the WebNLG dataset.

4.3 Sentence complexity

To assess the complexity of sentences in the WebNLG and CACAPO datasets, we look at the

	2-grams		3-grams	
	% = 1	$\overline{x} > 1$	% = 1	$\overline{x} > 1$
CACAPO (NL)	73.73%	6.99	85.18%	4.73
CACAPO (EN)	74.77%	5.78	88.32%	3.60
WebNLG (EN)	45.29%	18.92	57.09%	9.41
WebNLG (DE)	49.35%	14.19	61.65%	7.54

Table 3: Proportion of bigrams and trigrams occuring once, and average frequency of bigrams and trigrams that occur more than once.

	CACAPO (NL)	CACAPO (EN)	WebNLG (EN)
0	49.3%	37.31%	49.27%
1	4.38%	2.57%	0.11%
2	28.13%	10.44%	20.24%
3	6.45%	9.14%	9.62%
4	0.4%	2.12%	0.22%
5	5.89%	9.22%	4.66%
6	3.21%	1.13%	3.94%
7	2.24%	28.08%	11.93%

Table 4: D-Level proportions.

revised Developmental Level scale (Rosenberg and Abbeduto, 1987; Covington et al., 2006), also known as D-Level (similar to Novikova et al., 2017). We used D-Level Analyser (Lu, 2009) to obtain the D-Level proportions for the English datasets, and T-Scan (Pander Maat et al., 2014) to find the D-Level proportions for the Dutch dataset. There are currently no tools to obtain D-Level for German, but it can be assumed that the composition of this dataset is similar to its English WebNLG counterpart, as the German WebNLG dataset is a close translation of that version (Castro Ferreira et al., 2018). The D-Level scale contains 8 levels: level 0 being the simplest, and level 7 the most complex. Complexity is determined by, for instance, complex syntactic structures, subordinate clauses, and referring expressions.

Table 4 shows sizable differences between the datasets in terms of complexity. The Dutch CACAPO dataset predominantly consists of simpler sentences (below level 4), while the English version of the dataset has a large portion of higher level sentences. The WebNLG resides somewhere in between those two in terms of complexity. This would mean that the Dutch version of the CACAPO dataset would be the least challenging for systems to learn the sentence structure of, and the English version of the dataset the most challenging.

5 Baseline system performance

TGen, a sequence-to-sequence model using Attention (Dušek and Jurčíček, 2015), was used to establish a baseline on the CACAPO dataset.[11] The performance of TGen was evaluated on the test data of the CACAPO dataset using BLEU (Papineni et al., 2002), NIST (Doddington, 2002), METEOR (Banerjee and Lavie, 2005), ROUGE-L (Lin, 2004), CIDEr (Vedantam et al., 2015) and BertScore (Zhang et al., 2020) (Table 5).[12]

The results show that the TGen baseline scores vary considerably across domains, as was to be expected. The Dutch Stocks subcorpus offers a positive outlier, which might have to do with the relatively few labels and consistent language of the domain. It should be noted that the same parameters—originally used for the E2E challenge (Novikova et al., 2017)—were applied to all domains, which might mean that the model is too large and complex for some domains (such as Dutch Weather and English Incidents, where the texts are highly consistent translations of the data, and the domain only contains a small number of types), resulting in overfitting. In other cases, the dataset is arguably too small, which—combined with its lexical richness—might make it difficult for a neural NLG model to be trained on. However, in all cases, parameter tuning, application of different models, and tokenization/delexicalization of the training texts (as done by Novikova et al., 2017) is likely to increase the text quality and automatic metrics scores. Additionally, it seems worthwhile to explore ways of semi-automatically extending the training corpora, as we hope to do in future work.

6 Conclusion

This paper described the CACAPO dataset. A multilingual, multi-domain dataset that enables the use of neural pipeline architectures, as well as end-to-end architectures. The dataset is comparable in size to the WebNLG dataset, and its lexical richness—due to the fact that the texts directly originate from journalistic articles—provides interesting challenges. Furthermore, the fact that these texts were derived from 'naturally occurring'

[11] Parameters are provided in Appendix B. It should be noted that the system is only trained in an end-to-end fashion.

[12] METEOR and BertScore were calculated using the authors' provided scripts, while BLEU was calculated using SacreBLEU (Post, 2018), NIST using NLTK (Bird et al., 2009), and ROUGE-L and CIDEr using nlg-eval (Sharma et al., 2017).

Domain	BLEU	NIST	BertScore	METEOR	ROUGE-L	CIDEr
Incidents (Dutch)	4.65	1.13	70.29	10.93	20.02	0.27
Stocks (Dutch)	17.46	2.44	74.13	21.84	27.95	0.95
Sports (Dutch)	1.92	0.86	68.39	7.34	13.85	0.14
Weather (Dutch)	1.66	0.15	64.11	7.11	11.71	0.10
Incidents (English)	0.68	0.36	82.37	5.92	11.89	0.07
Stocks (English)	0.41	0.26	80.08	3.19	5.20	0.01
Sports (English)	1.27	0.64	82.50	5.66	12.64	0.08
Weather (English)	6.80	1.20	86.24	8.74	17.61	0.65

Table 5: TGen results on the CACAPO dataset.

texts means that there may be superfluous information, as well as indirect descriptions of the data in the text. This is challenging for NLG systems, as shown by the system performance scores when performing an end-to-end data-to-text task on the dataset using TGen (Dušek and Jurčíček, 2015). However, the dataset closely mirrors real-world scenarios in which companies oftentimes have large amounts of human-written texts that are not purposefully written for NLG applications, accompanied by corresponding data.

Bias The fact that the CACAPO dataset is based on 'naturally occurring' data addresses the issue of datasets being not representative of real world NLP issues. However, it should also be noticed that having unedited texts in the dataset means that the biases from the original data are still present in the dataset and may lead to further generation of biased texts (Leppänen et al., 2020). Therefore, texts generated with this dataset, as well as the texts in the dataset itself, could warrant more traditional linguistics-oriented text analysis research to investigate biases that might exist.

Evaluation NLG has recently increased its focus on evaluation and multiple researchers have argued that automatic metrics lack interpretability and do not correlate well with human judgments (see, for instance, Reiter, 2018b; van der Lee et al., 2019). This might especially be an issue for this type of dataset, originating from texts that—besides informing—try to provide engaging texts to read, as evidenced by the high lexical richness and sentence complexity. Since journalists try to convey data in diverse ways, reference-based metrics such as BLEU (Papineni et al., 2002), METEOR (Banerjee and Lavie, 2005), and ROUGE (Lin, 2004) might be especially ineffective to measure text quality. Van der Lee et al. (2018), for instance, found that BLEU scores were near zero for a similar dataset, while human evaluation showed the texts to be of reasonable quality. Recent learning-based metrics, such as RUSE (Shimanaka et al., 2018), BertScore (Zhang et al., 2020), MoverScore (Zhao et al., 2019), and BLEURT (Sellam et al., 2020) might be more viable options, since they claim to capture semantic similarity.

However, we discourage using this dataset as a leaderboard chasing game and recommend using various types of evaluation methods to evaluate systems trained on the CACAPO dataset (e.g., evaluating the results on the dataset using human and automatic metrics, and qualitative and quantitative research methods). Variety in evaluation methods ensures that the results obtained on this dataset are put into a broad perspective. This will give valuable insights into the systems trained on the dataset, as well as the characteristics of the dataset itself.

Future work The dataset creation method of this paper, where texts are collected first, and data is subsequently manually annotated for each text (Oraby et al., 2019), also facilitates extensions to the dataset with relative ease. We make the tools to do so publicly available, so that anyone interested can extend the current dataset by annotating a selection of scraped texts that were not used for the definitive dataset. In future work, we would also like to extend the dataset to other languages and other domains (e.g. product reviews, movie descriptions, etc.). Furthermore, we would like to explore the possibility of BERT-based (Devlin et al., 2019) Information Extraction to automatically extend the size of the dataset in a semi-supervised fashion.

Acknowledgements

We received support from RAAK-PRO SIA (2014-01-51PRO) and The Netherlands Organization for Scientific Research (NWO 360-89-050). We also want to thank the anonymous reviewers, Saar Hommes, Annemarie Nanne, Jeroen van de Nieuwenhof, Noa Reijnen, and Tess van der Zanden for their contributions.

References

Henk Asbreuk, Addie de Moor, and Esther van der Meer. 2017. *Basisboek journalistiek schrijven*. Noordhoff Uitgevers.

Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72, Ann Arbor, Michigan, USA. Association for Computational Linguistics.

Steven Bird, Ewan Klein, and Edward Loper. 2009. *Natural language processing with Python: Analyzing text with the natural language toolkit*. "O'Reilly Media, Inc.".

British National Corpus. 2007. *British National Corpus*, BNC XML Edition edition, volume 3. Distributed by Oxford.

João Canavilhas. 2007. Web journalism: from the inverted pyramid to the tumbled pyramid. *Biblioteca on-line de ciências da comunicação*.

Thiago Castro Ferreira, Chris van der Lee, Emiel van Miltenburg, and Emiel Krahmer. 2019. Neural data-to-text generation: A comparison between pipeline and end-to-end architectures. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*, Hong Kong, SAR. Association for Computational Linguistics.

Thiago Castro Ferreira, Diego Moussallem, Emiel Krahmer, and Sander Wubben. 2018. Enriching the WebNLG corpus. In *Proceedings of the 11th International Conference on Natural Language Generation*, pages 171–176, Tilburg University, The Netherlands. Association for Computational Linguistics.

Michael A Covington, Congzhou He, Cati Brown, Lorina Naci, and John Brown. 2006. How complex is that sentence? A proposed revision of the Rosenberg and Abbeduto D-Level Scale. In *CASPR Research Report 2006-01*. University of Georgia Artificial Intelligence Center, Athens, GA.

Helmut Daller, Roeland Van Hout, and Jeanine Treffers-Daller. 2003. Lexical richness in the spontaneous speech of bilinguals. *Applied linguistics*, 24(2):197–222.

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

George Doddington. 2002. Automatic evaluation of machine translation quality using n-gram co-occurrence statistics. In *Proceedings of the Second International Conference on Human Language Technology Research*, pages 138–145. Morgan Kaufmann Publishers Inc.

Ondřej Dušek and Filip Jurčíček. 2015. Training a natural language generator from unaligned data. 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)*, pages 451–461, Beijing, China. Association for Computational Linguistics.

Claire Gardent, Anastasia Shimorina, Shashi Narayan, and Laura Perez-Beltrachini. 2017a. Creating training corpora for NLG micro-planners. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 179–188, Vancouver, Canada. Association for Computational Linguistics.

Claire Gardent, Anastasia Shimorina, Shashi Narayan, and Laura Perez-Beltrachini. 2017b. The WebNLG challenge: Generating text from RDF data. In *Proceedings of the 10th International Conference on Natural Language Generation*, pages 124–133, Santiago de Compostela, Spain. Association for Computational Linguistics.

Albert Gatt and Emiel Krahmer. 2018. Survey of the state of the art in natural language generation: Core tasks, applications and evaluation. *Journal of Artificial Intelligence Research*, 61:65–170.

Barry Hendriks. 2019. Retrieving, cleaning and analysing Dutch news articles about traffic accidents. Master's thesis, University of Amsterdam, The Netherlands.

Matthew Honnibal and Ines Montani. 2017. spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing. *To appear*.

Nan Jiang, Jing Chen, Ri-Gui Zhou, Changxing Wu, Honglong Chen, Jiaqi Zheng, and Tao Wan. 2020. PAN: Pipeline assisted neural networks model for data-to-text generation in social internet of things. *Information Sciences*, 530:167–179.

Emiel Krahmer and Kees van Deemter. 2012. Computational generation of referring expressions: A survey. *Computational Linguistics*, 38(1):173–218.

Nico Kussendrager, Dick Van der Lugt, and Ben Rogmans. 2018. *Basisboek journalistiek*. Noordhoff.

J Richard Landis and Gary G Koch. 1977. The measurement of observer agreement for categorical data. *Biometrics*, pages 159–174.

Chris van der Lee, Albert Gatt, Emiel van Miltenburg, Sander Wubben, and Emiel Krahmer. 2019. Best practices for the human evaluation of automatically generated text. In *Proceedings of the 12th International Conference on Natural Language Generation*, pages 355–368, Tokyo, Japan. Association for Computational Linguistics.

Chris van der Lee, Bart Verduijn, Emiel Krahmer, and Sander Wubben. 2018. Evaluating the text quality, human likeness and tailoring component of PASS: A Dutch data-to-text system for soccer. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 962–972, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Leo Leppänen, Hanna Tuulonen, Stefanie Sirén-Heikel, et al. 2020. Automated journalism as a source of and a diagnostic device for bias in reporting. *Media and Communication*, 8(3):1–11.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Xiaofei Lu. 2009. Automatic measurement of syntactic complexity in child language acquisition. *International Journal of Corpus Linguistics*, 14(1):3–28.

Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven J. Bethard, and David McClosky. 2014. The Stanford CoreNLP natural language processing toolkit. In *Association for Computational Linguistics (ACL) System Demonstrations*, pages 55–60.

Simon Mille, Anja Belz, Bernd Bohnet, Yvette Graham, and Leo Wanner. 2019. The second multilingual surface realisation shared task (SR'19): Overview and evaluation results. In *Proceedings of the 2nd Workshop on Multilingual Surface Realisation (MSR 2019)*, pages 1–17, Hong Kong, China. Association for Computational Linguistics.

Ines Montani and Matthew Honnibal. 2018. Prodigy: A new annotation tool for radically efficient machine teaching. *Artificial Intelligence*.

Amit Moryossef, Yoav Goldberg, and Ido Dagan. 2019a. Improving quality and efficiency in plan-based neural data-to-text generation. In *Proceedings of the 12th International Conference on Natural Language Generation*, pages 377–382, Tokyo, Japan. Association for Computational Linguistics.

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

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

Nelleke Oostdijk, Martin Reynaert, Véronique Hoste, and Ineke Schuurman. 2013. The construction of a 500-million-word reference corpus of contemporary written Dutch. In *Essential speech and language technology for Dutch*, pages 219–247. Springer, Berlin, Heidelberg.

Shereen Oraby, Vrindavan Harrison, Abteen Ebrahimi, and Marilyn Walker. 2019. Curate and generate: A corpus and method for joint control of semantics and style in neural NLG. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5938–5951, Florence, Italy. Association for Computational Linguistics.

Henk Pander Maat, Rogier Kraf, Antal van den Bosch, Nick Dekker, M van Gompel, S Kleijn, Ted Sanders, and K van der Sloot. 2014. T-scan: A new tool for analyzing Dutch text. *Computational Linguistics in The Netherlands journal*, 4:53–74.

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

Ankur P. Parikh, Xuezhi Wang, Sebastian Gehrmann, Manaal Faruqui, Bhuwan Dhingra, Diyi Yang, and Dipanjan Das. 2020. ToTTo: A controlled table-to-text generation dataset. *arXiv preprint arXiv:2004.14373*.

Laura Perez-Beltrachini and Claire Gardent. 2017. Analysing data-to-text generation benchmarks. In *Proceedings of the 10th International Conference on Natural Language Generation*, pages 238–242, Santiago de Compostela, Spain. Association for Computational Linguistics.

Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Belgium, Brussels. Association for Computational Linguistics.

Thomas Proisl and Peter Uhrig. 2016. SoMaJo: State-of-the-art tokenization for German web and social media texts. In *Proceedings of the 10th Web as Corpus Workshop*, pages 57–62, Berlin. Association for Computational Linguistics.

Ratish Puduppully, Li Dong, and Mirella Lapata. 2019. Data-to-text generation with entity modeling. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages

2023–2035, Florence, Italy. Association for Computational Linguistics.

Dragomir Radev, Rui Zhang, Amrit Rau, Abhinand Sivaprasad, Chiachun Hsieh, Nazneen Fatema Rajani, Xiangru Tang, Aadit Vyas, Neha Verma, Pranav Krishna, Yangxiaokang Liu, Nadia Irwanto, Jessica Pan, Faiaz Rahman, Ahmad Zaidi, Murori Mutuma, Yasin Tarabar, Ankit Gupta, Tao Yu, Yi Chern Tan, Xi Victoria Lin, Caiming Xiong, and Richard Socher. 2020. DART: Open-domain structured data record to text generation. *arXiv preprint arXiv:2007.02871*.

Ehud Reiter. 2018a. Hallucination in neural nlg. Retrieved from https://ehudreiter.com/2018/11/12/hallucination-in-neural-nlg/ on August 27, 2020.

Ehud Reiter. 2018b. A structured review of the validity of BLEU. *Computational Linguistics*, pages 1–12.

Ehud Reiter and Robert Dale. 2000. *Building Natural Language Generation Systems*. Cambridge University Press, New York, NY, USA.

Sheldon Rosenberg and Leonard Abbeduto. 1987. Indicators of linguistic competence in the peer group conversational behavior of mildly retarded adults. *Applied Psycholinguistics*, 8(1):19–32.

Thibault Sellam, Dipanjan Das, and Ankur Parikh. 2020. BLEURT: Learning robust metrics for text generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7881–7892, Online. Association for Computational Linguistics.

Shikhar Sharma, Layla El Asri, Hannes Schulz, and Jeremie Zumer. 2017. Relevance of unsupervised metrics in task-oriented dialogue for evaluating natural language generation. *CoRR*, abs/1706.09799.

Serge Sharoff. 2006. Creating general-purpose corpora using automated search engine queries. In Marco Baroni and Silvia Bernardini, editors, *WaCky! Working papers on the Web as Corpus*, pages 63–98. GEDIT.

Hiroki Shimanaka, Tomoyuki Kajiwara, and Mamoru Komachi. 2018. RUSE: Regressor Using Sentence Embeddings for Automatic Machine Translation Evaluation. In *Proceedings of the Third Conference on Machine Translation: Shared Task Papers*, pages 751–758, Belgium, Brussels. Association for Computational Linguistics.

Anastasia Shimorina, Elena Khasanova, and Claire Gardent. 2019. Creating a corpus for Russian data-to-text generation using neural machine translation and post-editing. In *Proceedings of the 7th Workshop on Balto-Slavic Natural Language Processing*, pages 44–49, Florence, Italy. Association for Computational Linguistics.

R. Vedantam, C. L. Zitnick, and D. Parikh. 2015. CIDEr: Consensus-based image description evaluation. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4566–4575.

Hongmin Wang. 2019. Revisiting challenges in data-to-text generation with fact grounding. In *Proceedings of the 12th International Conference on Natural Language Generation*, pages 311–322, Tokyo, Japan. Association for Computational Linguistics.

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

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2020. BERTScore: Evaluating text generation with BERT. In *Proceedings of the Eighth International Conference on Learning Representations*, pages 1–43, Ethiopia, Addis Ababa. OpenReview.net.

Wei Zhao, Maxime Peyrard, Fei Liu, Yang Gao, Christian M. Meyer, and Steffen Eger. 2019. MoverScore: Text generation evaluating with contextualized embeddings and earth mover distance. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 563–578, Hong Kong, China. Association for Computational Linguistics.

A Examples

Atribute	Value
incidentCause	nog onbekende oorzaak *unknown causes*
victimVehicle	personenwagen *passenger car*
suspectVehicle	vrachtwagen *truck*
dateTime	omstreeks 15.45 uur *around 3.45 pm*
incidentType	frontaal met elkaar in botsing *collided head-on*

↓

Door nog onbekende oorzaak kwamen een personenwagen en een vrachtwagen omstreeks 15.45 uur frontaal met elkaar in botsing.
Due to unknown causes, a passenger car and a truck collided head-on at around 3.45 pm.

Figure A: Example of a set of attribute-value pairs (top) and corresponding text (bottom) for the Dutch Incidents domain.

Atribute	Value
positionOfPlayer	middenvelder *midfielder*
assistName	Joey Suk *Joey Suk*
assistType	verlengd *extended*
goalName	aanvoerder Ars *captain Ars*
goalType	beheerst binnen schoof *composedly slid it in*

↓

Een geblokt schot werd door middenvelder Joey Suk verlengd tot bij aanvoerder Ars die beheerst binnen schoof.
A blocked shot was extended by midfielder Joey Suk to captain Ars, who composedly slid it in.

Figure B: Example of a set of attribute-value pairs (top) and corresponding text (bottom) for the Dutch Sports domain.

Atribute	Value
companyName	Ctac *Ctac*
stockChange	daalde *declined*
stockChangePercentage	0,5 procent *0,5 procent*

↓

Ctac daalde 0,5 procent op de lokale markt.
Ctac declined 0.5 percent on the local market.

Figure C: Example of a set of attribute-value pairs (top) and corresponding text (bottom) for the Dutch Stocks domain.

Atribute	Value
windAmount	zwak tot matig *weak to moderate*
windDirection	oost tot zuidoost *east to southeast*

↓

De wind is zwak tot matig en komt uit oost tot zuidoost.
There will be a weak to moderate breeze coming from east to southeast.

Figure D: Example of a set of attribute-value pairs (top) and corresponding text (bottom) for the Dutch Weather domain.

Atribute	Value
victimNumber	several
victimStatus	more serious injuries
takenToHospital	True
hospitalName	UMC

↓

Several with more serious injuries were later transported to UMC.

Figure E: Example of a set of attribute-value pairs (top) and corresponding text (bottom) for the English Incidents domain.

Atribute	Value
winLossRecord	15 of 19
teamName	Milwaukee

↓

But Milwaukee dropped 15 of 19 to begin the regular season's final month.

Figure F: Example of a set of attribute-value pairs (top) and corresponding text (bottom) for the English Sports domain.

Atribute	Value
exchangeName	Dow
stockChange	added
stockPoints	187.86
stockChangePercentage	0.7%

↓

The Dow added 187.86 points, or 0.7%.

Figure G: Example of a set of attribute-value pairs (top) and corresponding text (bottom) for the English Stocks domain.

Atribute	Value
timePoint	Overnight
temperatureCelsius	15C

↓

Overnight temperature of 15C.

Figure H: Example of a set of attribute-value pairs (top) and corresponding text (bottom) for the English Weather domain.

B Baseline Model Parameters

Setting	Value
Adam optimizer learning rate	5e-4
Network cell type	LSTM
Embedding (+cell) size	50
Batch size	20
Encoder length (max. input attribute-value pairs)	10
Decoder length (max. output tokens)	80
Max. training epochs	20
Training instances reserved for validation	2000

TGen training parameters as reported in (Novikova et al., 2017): main-sequence-to-sequence model with attention.

Setting	Value
Adam optimizer learning rate	1e-3
Embedding (+cell) size	50
Batch size	20
Training epochs	20
Encoder length (max. input tokens)	80
Training instances reserved for validation	2000

TGen training parameters as reported in (Novikova et al., 2017): reranker.

Setting	Value
Beam size	10
Reranker misfit penalty	100

TGen decoder parameters as reported in (Novikova et al., 2017).

The parameters are the same as the TGen parameters for the E2E dataset (Novikova et al., 2017). Raw strings are used for training and generation. Validation is performed on the reserved instances after each epoch using BLEU. Early stopping is applied if the top 3 BLEU results do not change for 5 epochs.

C Labels of all data types

Subdomain	Data types
Dutch sports	assistName, assistType, chanceForName, chanceForNationality, chanceForNumber, chanceForType, coachName, defendedName, disallowedGoalName, disallowedGoalType, finalScore, formationTeam, goalName, goalScore, goalType, goalkeeperName, halfTimeScore, hasLostTeam, hasScored, hasTiedTeam, hasWonTeam, homeAway, injuredName, injuryType, matchDate, matchStreakNumber, matchStreakType, matchTime, nextMatchDate, nextMatchHomeAway, nextMatchTeam, numberOfMatchGoals, numberOfMatchesPlayed, numberOfPoints, numberOfSeasonGoals, playerAge, playerName, playerNationality, positionOfPlayer, redCardName, refereeName, stadiumPlayed, substituteName, suspendedName, tackleGiverName, tackleRecipientName, teamName, teamStandings, twiceYellowName
English sports	ERA, RBI, atBatNumber, baseNumber, baseReachedNumber, baseStolen, basesRan, batterHitsTries, batterName, batterScoreNumber, battersFacedNumber, battingAverage, battingLineupNumber, catchType, catcherName, competitionName, earnedRunsNumber, errorNumber, fielderName, fielderPosition, finalScore, gameNumber, gameTally, hasLostTeam, hasScored, hasWonTeam, hitNumber, homeAway, homeRunNumber, injuryType, inningNumber, inningScore, inningsPitched, isOut, leftOnBase, locationPlayed, managerName, matchDate, matchStreakNumber, matchStreakType, numberOfStarts, onBaseNumber, outNumber, pitchCount, pitchNumber, pitchResult, pitchResultNumber, pitchType, pitcherName, pitcherRecord, pitcherSaveRecord, pitchesTotalThrown, presidentName, retireNumber, runAverage, runNumber, scoreNumber, scoreTally, standingsGames, startsNumber, stealNumber, strikeNumber, strikeOutNumber, strikeTrajectory, strikingType, teamName, teamRecord, teamStandings, throwDirection, umpireName, umpireType, unearnedRunsNumber, walkNumber, winLossRecord, winLossType, winningPercentage
Dutch/English stocks	amountNumber, companyName, exchangeName, locationName, moneyAmount, stockChange, stockChangePercentage, stockPoints, tickerName, timePoint
Dutch/English weather	cloudAmount, cloudChange, cloudType, compassDirection, gustAmount, gustChange, gustVelocity, locationArea, maximumTemperature, minimumTemperature, precipitationAmount, snowAmount, temperatureCelsius, temperatureChange, temperatureHotCold, timePoint, weatherArea, weatherChange, weatherFrequency, weatherIntensity, weatherOccurringChance, weatherType, windAmount, windChange, windDirection, windSpeedBft, windTurning, windType
Dutch incidents	dateTime, incidentCause, incidentLocation, incidentType, suspectAddress, suspectAge, suspectAmount, suspectDescription, suspectGender, suspectStatus, suspectVehicle, victimAddress, victimAge, victimAmount, victimDescription, victimGender, victimName, victimStatus, victimVehicle
English incidents	accidentAddress, accidentDate, hospitalName, numberOfRoundsFired, personnelArrivedTime, prisonName, shootingNumber, shootingType, suspectAge, suspectAgeGroup, suspectBased, suspectDescription, suspectGender, suspectHeight, suspectName, suspectNumber, suspectOccupation, suspectRace, suspectStatus, suspectVehicle, suspectWeapon, suspectWeight, takenToHospital, victimAge, victimAgeGroup, victimBased, victimGender, victimName, victimNumber, victimOccupation, victimRace, victimStatus, victimVehicle

Labels of data types used in the CACAPO dataset per subdomain.