BLAB Reporter: Automated journalism covering the Blue Amazon

Yan Vianna Sym
Escola Politécnica
Universidade de São Paulo
São Paulo, Brazil
yan.sym@usp.br

João Gabriel Moura Campos
Escola Politécnica
Universidade de São Paulo
São Paulo, Brazil
joaogcampos@usp.br

Fabio Gagliardi Cozman
Escola Politécnica
Universidade de São Paulo
São Paulo, Brazil
fgcozman@usp.br

Abstract

This demo paper introduces BLAB Reporter, a robot-journalist system covering the Brazilian Blue Amazon. The application is based on a pipeline architecture for Natural Language Generation, which offers daily reports, news summaries and curious facts in Brazilian Portuguese. By collecting, storing and analysing structured data from publicly available sources, the robot-journalist uses domain knowledge to generate, validate and publish texts in Twitter. Code and corpus are publicly available [1].

1 Introduction

Data-to-text Natural Language Generation (NLG) is the computational process of generating meaningful and coherent natural language in the form of text or speech to describe non-linguistic input data (Reiter and Dale, 2000). Successful examples of data-to-text systems can be found in both academia and industry, with applications in weather forecasting (Belz, 2008), image captions and chatbots (Adamopoulou and Moussiades, 2020). Within the range of NLG applications, robot-journalism is one of the most prominent endeavors thanks to the high volume of structured data streams available, which enables automated systems to report recurrent information with high-fidelity and lexical variety (Teixeira et al., 2020).

An interesting domain for data-to-text generation is ocean monitoring. For instance, global attention was drawn in 2021 to a container ship that obstructed the Suez Canal for six consecutive days. The result was a global shortage of essential commodities, including medical supplies and medicines, which were essential during the coronavirus pandemic (COVID-19). Accurate and low latency information reports can be very helpful in these situations, but communicating to general audiences in a accessible way usually demands coverage by specialized human journalists. To address this issue, we present our robot-journalist named *BLAB Reporter*, a NLG system based on a pipeline architecture that generates daily reports, news, content summarization and curious facts about the Blue Amazon and publishes them on Twitter in Brazilian Portuguese [2]. The Blue Amazon is the exclusive economic zone (EEZ) of Brazil, with an offshore area of 3.6 million square kilometers along the Brazilian coast, an area rich in marine biodiversity and energy resources (Wiesebron, 2013). The BLue Amazon Brain (BLAB) is a project aiming to address complex questions about the marine ecosystem, and integrates a number of services aimed at disseminating information about the Blue Amazon region and its importance.

2 System overview

Our system follows a pipeline architecture that converts non-linguistic data into text in 6 steps: *Content Selection, Discourse Ordering, Text Structuring, Lexicalization, Referring Expression Generation and Textual Realization* (Ferreira et al., 2019). Our system also comprises two additional steps: *Data Acquisition*, responsible for extracting and storing information from multiple data streams in a structured format, and *Summarization*, responsible for summarizing news in the form of small consecutive tweets. This kind of architecture, depicted in Figure 1, allows for trustworthy output as well as easy access to and maintenance of sub-modules.

The grammar used by the model was built by first running the content selection step in previous data and generating 30 non-linguistic reports. These non-linguistic reports were then manually verbalized and the input and output representations for each pipeline module were manually annotated. When deployed, each module draws on the selected combination of templates using rule-based approaches. Because we deal with a sensitive do-
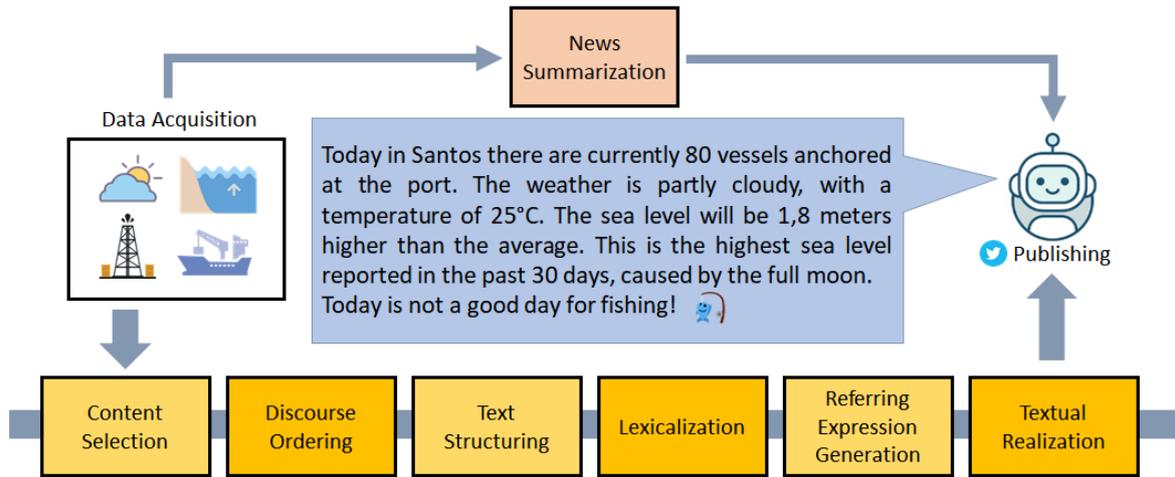

Figure 1: Robot-journalist system architecture

main, we opted to use the pipeline architecture instead of the novel end-to-end systems, which sometimes hallucinates content (Ji et al., 2022). The following sections describe each module.

Data Acquisition The first step of our system performs data gathering, filtering and cleaning before it is put in a data warehouse. In our application, this module consists of a web scraping framework for extracting information from public websites and storing it on a structured format. Our system currently collects information about weather, tide charts, marine vessel traffic and eventual earthquakes on the Brazilian coast, and stores data using MongoDB, a source-available cross-platform document-oriented database program (Győrödi et al., 2015).

Content Selection This module decides which relevant information should be verbalized in the text. The content selection process generally consists of applying domain specific knowledge to create a rule-based approach. The following text is an example of the content selection module output:

```
CURRENT WEATHER AND
TEMPERATURE (weather="partly
cloudy",temperature="25°C",city="Santos",
timestamp="May 22, 2022); FISHING
CONDITION (condition="good",event="sea
level is high";height of the sea:"1.8
meters";days since last peak="30");
CAUSE(earthquake:"no",moon
calendar:"yes");
```

Discourse Ordering and **Text Structuring** Once the relevant content has been selected, our application constructs a logical timeline of events and sorts the intent messages in sentences and paragraphs in order to enhance reader comprehension (Heilbron et al., 2019). This combined module is based on a list of possible intent orderings collected from the corpus and decides what is the most optimal way to sort the sentences, bearing in mind the 280 character limit on Twitter. For example, for the messages related to weather conditions, a possible outcome order would be:

```
WEATHER ALERT → CAUSES → DAYS
SINCE LAST PEAK
```

Lexicalization At this step of the pipeline, lexical choices are made in order to verbalize the intents, finding the proper words to generate proper sentences. We applied a template-based lexicalization with plenty of options to choose from, providing for more inflections and variety of text in comparison to the fill-template approach (Stede, 1994). The templates provide for gender and number inflection, for example: *"No Rio de Janeiro foi registrada a maior temperatura da última semana"* vs. *"Em São Paulo foi registrado o maior vento dos últimos 10 dias"*.

Referring Expression Generation In order to replace entity tags throughout the template, this module generates the appropriate references using a list of possible expressions for each entity (Krahmer and Van Deemter, 2012). For the first reference to an entity in the text, a full description is used (e.g., INSTITUTE → "The Seismological Center at the University of São Paulo (USP)"), whereas for subsequent references a random referring expression to the entity is chosen (e.g., INSTITUTE →

"The Seismological Center at USP"; "the Institute"; "sismoUSP"; "it"; etc.).

Textual Realization The last step of the pipeline is responsible for transforming intermediate representations into human-readable Brazilian Portuguese text. After the content is generated, this step applies a final rule based transformation to the text with the goal to make the texts look more natural, for example adding greetings message and emojis. We also added a validation layer in this step, to ensure there is no offensive content within the text. The output of this module is published using Twitter's API. An example of generated text is shown in Figure 1.

Summarization An extra module was implemented in our system in order to outline public news about the Blue Amazon while also splitting text into small consecutive tweets. Because data hallucination is less critical in this step, this module was implemented using PTT5, a T5 model pretrained in a large collection of web pages in Portuguese, which uses state of the art transformer architecture (Carmo et al., 2020). Key challenges of this approach are interpretation and evaluation of the generated texts (Rao and Gudivada, 2018).

The generated texts are scheduled to be published on specific periods of time. We noticed that weather related content has more user engagement during mornings, while news and curious facts content are usually more viewed in the evenings. More critical messages, for example information related to earthquakes in the Blue Amazon region, are published as soon as the data is collected and stored in the database.

3 Conclusions

This paper presents a data-to-text system based on a pipeline architecture for NLG. Our system applies robot-journalism techniques to generate and publish reports, news and curious facts in Brazilian Portuguese about the Blue Amazon. Due to its rule-base nature, our system provides high-fidelity content by applying a pipeline methodology and obtains lexical variety by drawing from a list of multiple available template options for the same intent. In the future we plan to add more sources of information to the pipeline, for example statistics about oil exploration and reporting of illegal fishing activities in real time. We also plan to utilize user engagement data and apply artificial neural network techniques to improve our system's performance.

References

Eleni Adamopoulou and Lefteris Moussiades. 2020. An overview of chatbot technology. In *IFIP International Conference on Artificial Intelligence Applications and Innovations*, pages 373–383. Springer.

Anja Belz. 2008. Automatic generation of weather forecast texts using comprehensive probabilistic generation-space models. *Natural Language Engineering*, 14(4):431–455.

Diedre Carmo, Marcos Piau, Israel Campiotti, Rodrigo Nogueira, and Roberto Lotufo. 2020. Ptt5: Pretraining and validating the t5 model on brazilian portuguese data. *arXiv preprint arXiv:2008.09144*.

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. *arXiv preprint arXiv:1908.09022*.

Cornelia Győrödi, Robert Győrödi, George Pecherle, and Andrada Olah. 2015. A comparative study: Mongodb vs. mysql. In *2015 13th International Conference on Engineering of Modern Electric Systems (EMES)*, pages 1–6. IEEE.

Micha Heilbron, Benedikt Ehinger, Peter Hagoort, and Floris P De Lange. 2019. Tracking naturalistic linguistic predictions with deep neural language models. *arXiv preprint arXiv:1909.04400*.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Yejin Bang, Andrea Madotto, and Pascale Fung. 2022. Survey of hallucination in natural language generation. *arXiv preprint arXiv:2202.03629*.

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

CR Rao and Venkat N Gudivada. 2018. *Computational analysis and understanding of natural languages: principles, methods and applications*. Elsevier.

Ehud Reiter and Robert Dale. 2000. Building applied natural language generation systems. *Natural Language Engineering*.

Manfred Stede. 1994. Lexicalization in natural language generation: A survey. *Artificial Intelligence Review*, 8(4):309–336.

André Luiz Rosa Teixeira, João Campos, Rossana Cunha, Thiago Castro Ferreira, Adriana Pagano, and Fabio Cozman. 2020. DaMata: A robot-journalist covering the brazilian amazon deforestation. In *Proceedings of the 13th International Conference on Natural Language Generation*, pages 103–106.

Marianne Wiesebron. 2013. Blue Amazon: thinking about the defence of the maritime territory. *Austral: Brazilian Journal of Strategy & International Relations*, 2(3):107–132.