Data-To-Text Generation for Domain-Specific Purposes

Tetiana Drobot[0000–0003–4172–2846]

Taras Shevchenko National University of Kyiv,
Institute of Philology Taras Shevchenko Blvd, 14, Kyiv, Ukraine

tetyanadrobot33@gmail.com

Abstract. The first commercial implementation of Natural Language Generation (NLG) system dates back to the turn of the XXI century. Since then two main methods of NLG – text-to-text generation and data-to-text generation – have grown more complex in order to solve new business challenges. This research project focuses on the full cycle of template-based generation of hotel descriptions from linguistic and non-linguistic input: starting with data scraping and preparation up to rendering the whole text. Also, several improvements to the template- based approach were suggested.

Keywords: Natural Language Generation, data-to-text generation, template-based approach.

Nowadays many industries (e.g., tourism, meteorology, sports journalism, etc.) face a problem of having thousands of data to process and quickly write about. The task is tremendously time-consuming for professional writers. So eventually the choice will fall on data-to-text generation[1] when a computer program converts the incoming data into a text by filling the gaps in a predefined template. The process mentioned above describes a template-based approach to natural language generation (NLG). This method is quite popular due to its simplicity (i.e., no specialized knowledge needed to develop), flexibility (i.e., the ability to be customized to any domain) and good quality of output texts. There are also some drawbacks, such as the possibility to add only handcrafted tem- plates. Another one, little variation in style, can be considered as an advantage if we have synonymized the templates richly. Therefore, a customer gets the feeling that all texts are written by the same qualified author. With the high attention to machine learning (ML) and neural networks (NN), one more question has to be asked: can the NLG system be trainable? Yes, but it takes too much time and resources to train an ML algorithm and, even more, retrain it if need be.

There are a few ways to enhance template-based text generation:

– expand templates to contain information needed to generate more complicated utterances. This task can be done automatically with the help of the WordNet ontology and unannotated corpora of domain-specific texts;
– add linguistic rules to manipulate and maintain templates.

The attempt of their implementation in order to generate hotel descriptions will be shown in the next paragraphs.

According to Reiter and Dale (2000)[2], an NLG system can be decomposed into distinct modules that form a pipeline process. These modules are: document planner with a tree where internal nodes represent structure and leaf nodes represent content as an output; microplanner, also a tree output with internal nodes as structural elements of the document and the leaf nodes as sentences; surface realiser which transforms the sentence representations into text[3].

A template-based approach erases the boundaries between the mentioned modules. But the same tasks to perform are left. The first thing that any NLG system has to take as input is a communication goal. It describes the desired output of the system, e.g., the communication goal of a system that generates hotel information will be expressed in terms of the data it stores, such as "What features does the specific hotel have (nearby attractions, facilities, on-site restaurant, etc.)?". These features can be scraped from travel websites (Trivago, Hotels.com, TripAdvisor, Booking.com, etc.).

The next step is data preparation. Some changes are applied to the raw data before "filling" the templates' gaps: asserting conjunctions to the lists, synonymization of some phrases, combining two or more expressions together (e.g., "flat-screen TV", "cable channels" and "satellite channels" to "flat-screen

TV with cable and satellite channels") etc. The modified data are then stored as dictionary objects.

Using the data sources, the document planning module will decide what information should be included in the produced text (content determination)[2]. As the template-based generator is the object of interest, the order of the sentences is fixed. They are also structured into text sections. A section will be added to the final text only if it includes two or more sentences. For this task, the rules for sentence rendering were written. They work section-by-section, activating the templates of specific sentences in accordance with the given data.

The last step is sections' rendering. It is primarily concerned with the selection of the appropriate synonyms by searching of the word, chosen by the system, in the previous three sentences (or less, depending on the number of sentences that have already been added to the text).

Thus this work offers a solution to domain-specific tasks of text generation for small businesses and start-ups which have restricted hardware resources or a short time for development and implementation.

References

1. Gatt, A., Krahmer, E.: Survey of the state of the art in natural language generation: Core tasks, applications and evaluation. Journal of Artificial Intelligence Research, 61, pp. 65–170 (2018)
1. Reiter, E., Dale, R.: Building natural language generation systems. Cambridge University Press, Cambridge, UK (2000)
2. Learning to tell tales: automatic story generation from Corpora, https://urlzs.com/GUcj. Last accessed 10 Apr 2019