



BRNO UNIVERSITY OF TECHNOLOGY

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

FACULTY OF INFORMATION TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ

DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA

ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ

STYLIZED NATURAL LANGUAGE GENERATION IN DIALOGUE SYSTEMS

GENEROVÁNÍ STYLIZOVANÉHO LIDSKÉHO JAZYKA V DIALOGOVÝCH SYSTÉMECH

BACHELOR'S THESIS

BAKALÁŘSKÁ PRÁCE

AUTHOR

AUTOR PRÁCE

KSENIA BOLSHAKOVA

SUPERVISOR

VEDOUCÍ PRÁCE

Ing. MARTIN FAJČÍK

BRNO 2020

Abstract

Do tohoto odstavce bude zapsán výtah (abstrakt) práce v anglickém jazyce.

Abstrakt

Do tohoto odstavce bude zapsán výtah (abstrakt) práce v českém (slovenském) jazyce.

Keywords

Sem budou zapsána jednotlivá klíčová slova v anglickém jazyce, oddělená čárkami.

Klíčová slova

Sem budou zapsána jednotlivá klíčová slova v českém (slovenském) jazyce, oddělená čárkami.

Reference

BOLSHAKOVA, Ksenia. *Stylized Natural Language Generation in Dialogue Systems*. Brno, 2020. Bachelor's thesis. Brno University of Technology, Faculty of Information Technology. Supervisor Ing. Martin Fajčík

Stylized Natural Language Generation in Dialogue Systems

Declaration

I hereby declare that this Bachelor's thesis was prepared as an original work by the author under the supervision of Mr. X The supplementary information was provided by Mr. Y I have listed all the literary sources, publications and other sources, which were used during the preparation of this thesis.

.....

Ksenia Bolshakova

November 21, 2019

Acknowledgements

Here it is possible to express thanks to the supervisor and to the people which provided professional help (external submitter, consultant, etc.).

Contents

1	Introduction	2
2	Dialogue systems	3
3	Natural Language Generation	5
3.1	Architecture of NLG	5
3.2	Categories of Natural Language Generation approaches	6
3.3	Models for Natural Language Generation	7
4	Other	9
	Bibliography	10

Chapter 1

Introduction

//TODO Write what is NLP

Chapter 2

Dialogue systems

Dialogue system is a computer system to communicate with a human. Nowadays you meet dialog systems everywhere. A lot of devices have incorporated goal-oriented spoken dialogue systems, such as Yandex's Alisa, Apple's Siri, Microsoft's Cortana, Amazon Alexa, and Google Assistant. Dialogue systems are also used in cars (hands-free car-specific functions, Android Auto, Apple CarPlay, vendor-specific solutions), web (search assistants (IKEA), Facebook Messenger and Telegram chatbots), robots, computer games, research systems (skylar.speech.cs.cmu.edu) etc, because a conversation is a natural way for people to get information.

Basic Dialogue System Types:

- Task-oriented
 - focused on completing a certain task(s)
- Non-task-oriented
 - chitchat
 - gaming the Turing test

Communication Domains:

„Domain“ is a conversation topic or an area of interest

- Single/Closed-domain is one well-defined area
- Multi-domain is joining several single-domain systems
- Open-domain “responds to anything”

Exists several **modes of communication**:

- Text
- Voice
- Multimodal
 - voice/text + graphics
 - additional modalities: video - gestures, mimics; touch

Dialogue initiative:

- system-initiative
 - system asks questions, user must reply in order to progress
 - least natural
 - „form-filling“ („Hello, please enter your e-mail“)
- user-initiative
 - user asks, machine responds („Siri, set the timer for 5 minutes“)
- mixed-initiative
 - system and user both can ask and react to queries
 - most natural

Dialogue system architecture is illustrated in Figure 2.1. This architecture consists from Natural Language Understanding (NLU), dialogue management (DM), and Natural Language Generation (NLG).

NLU extracts the meaning from the user utterance and converts into a structured semantic representation. Natural Language Understanding traditionally consists of domain identification and intent prediction, which are framed as utterance classification problems, and slot filling, framed as a sequence tagging task.

DM plays two roles, tracking the dialogue state and performing the dialogue policy (i.e., telling the agent how to act given the dialogue state.)

NLG transforms structured data into natural language.^[2]

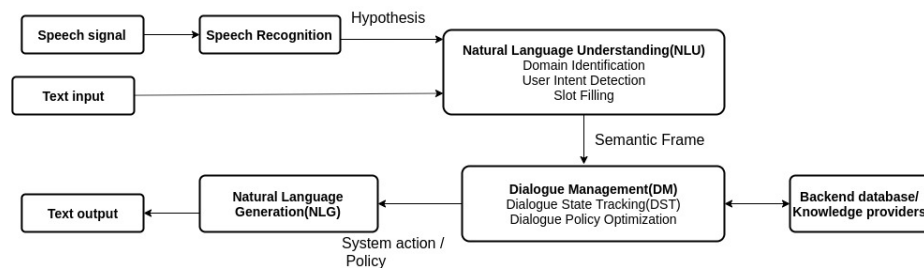


Figure 2.1: Dialogue system architecture.

Chapter 3

Natural Language Generation

Natural Language Generation is a subsection of Natural Language Processing (NLP). NLG uses different methods and tricks to convert data to human language.

3.1 Architecture of NLG

Reiter and Dale in the article [4] proposed architecture splitted into 3 stages, which is illustrated in Figure 3.1.

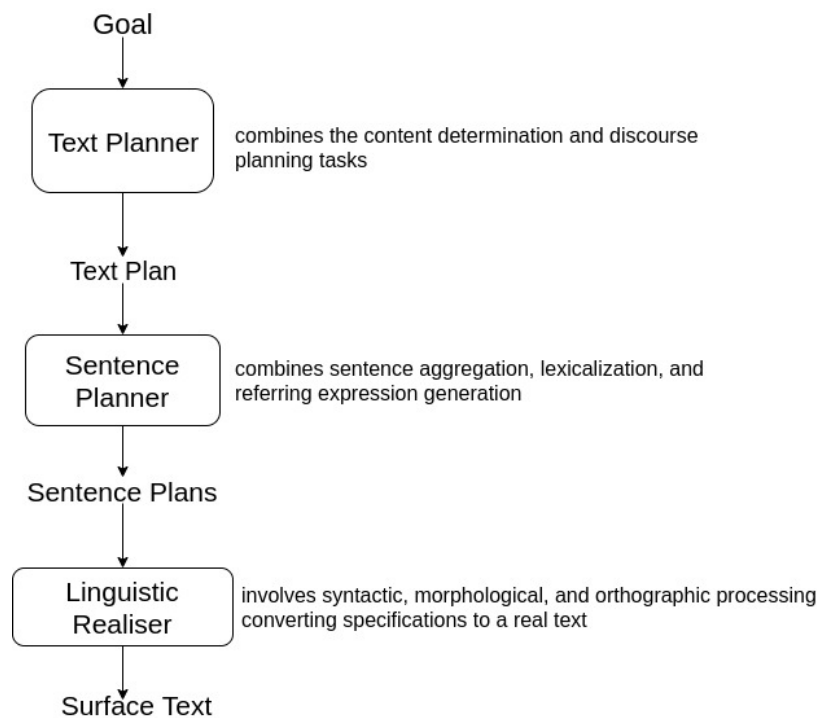


Figure 3.1: NLG architecture.

This pipeline shows the milestones of natural language generation, however, specific steps and approaches, as well as the models used, can vary significantly with the technology development.

NLG approaches can be grouped into two categories, one focuses on generating text using templates or rules (linguistic) methods, the other uses corpus-based statistical/dynamic methods, where **corpus** is a collection of texts.

3.2 Categories of Natural Language Generation approaches

Template-based systems map their non-linguistic input directly to the linguistic surface structure. This linguistic structure may contain gaps. Well-formed outputs does not contain gaps. The template-based system selects a proper response for the current conversation from a repository with response selection algorithms.

This approach has a long development way from simple gap-filling to word-level grammatical functions:

Gap-Filling Approach is one of the oldest approaches, that can automatically fill gaps in text, that have a predefined structure, with a small amount of data retrieved from a spreadsheet row, database table entry, etc. This approach is a quite limited.

Scripts or Rules-Producing Text is an approach, that expands basic gap-filling systems with general-purpose programming constructs via a scripting language or by using business rules. These systems still lack linguistic capabilities and cannot generate high-quality text reliably.

Word-Level Grammatical Functions was adding to template-based systems to deal with morphology, morphophonology and orthography. These functions made it easier to generate grammatically correct texts and to write complex template systems.

Advantages of template-based systems:

- Robust and reliable
- The output produced by this modules is likely to be grammatically correct and not contain unexpected generation errors
- Generation process is fully controlled

Disadvantages of template-based systems:

- Number of templates grows quickly (different templates for singular and plural versions)
- Expensive and time-consuming to deploy a real dialogue system
- Limits its usage to other domains
- Not much variation in output (no planning of text; simple concatenation of strings)
- Not able to handle unknown inputs
- Templates often sound unnatural due to their generic structures
- Not able to learn and not able to adapt to the user

Example: „The 306 train leaves Aberdeen at 11:20 am“

Semantic representation:

Departure(train₃₀₆, location_{abdn}, time₁₁₂₀)

Template:
[train] *is leaving* [town] *now*

The gaps represented by [train] and [town] are filled by looking up the relevant information in a table. More information you can find in [4].

Corpus-based/Dynamic approach dominates the NLG community, special in the case of open-domain tasks, where it is almost impossible to hand-craft the templates for all possible combinations of semantic units and their respective surface realization. In corpus-based systems include statistical and machine learning approaches.

One of the first approaches in corpus-based methods was **Dynamic Sentence Generation**, that dynamically creates sentences from representations of the meaning to be conveyed by the sentence and/or its desired linguistic structure. It allows do not write code for every boundary case and includes aggregation, reference, ordering and connectives to optimise sentences.

Next level of corpus-based approaches was **Dynamic Document Creation**, what can produce relevant and well-structured document.

Advantages of corpus-based approach:

- Possibility to mimic the language of a real domain expert
- Ability to generate more proper responses that could have never appeared in the corpus
- Possibility to use for open-domain dialogue systems
- A lot of possible variations of output
- Ability to learn
- Ability to handle unknown inputs

Disadvantages of corpus-based approach:

- It is necessary to have a good corpus
- Generation process is not fully controlled
- The output can be grammatically incorrect

[3] [1]

3.3 Models for Natural Language Generation

NLG evolution from templates to dynamic generation of sentences took a lot of time and models developed along with it.

One of the first models was **Markov chain**. It is a stochastic model, that is used to describe the next event in a sequence given the previous event only by probabilistic weighting. It does not track all the previous states in a sequence to conclude what is the next possible state. In our case state is an n-gram(contiguous sequence of n items from text or speech).

The next model, what is used in NLG, is **Recurrent neural network (RNN)**. Neural networks are computing systems that are inspired by biological neural networks. A „neuron“

is a mathematical function that classifies and collects information according to a specific architecture. A neural network contains layers of interconnected „neurons“. RNN takes series of input with no predetermined limit on size. Recurrent neural network passes each item of the sequence through a feedforward network and stores the information from the previous step. In each iteration this model calculates the probability of the next item with using information in the memory.

The architecture of RNN is illustrated in Figure 3.2, where coefficient \mathbf{h} is a hidden layer, \mathbf{x} is an input and \mathbf{y} is an output. Coefficient \mathbf{w} is a weight, what is transformed to produce a sensible output.

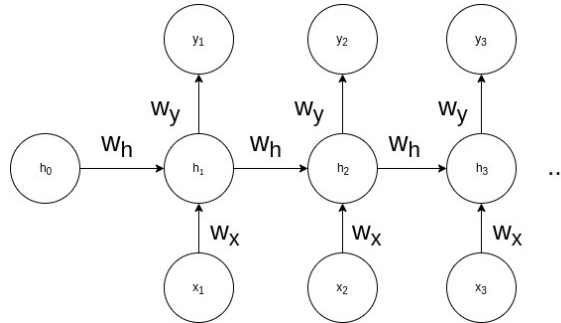


Figure 3.2: Architecture of recurrent neural network.

LSTM
Transformer

NLG problems

The NLG component converts an abstract dialogue action into natural language surface utterances. In NLG it is necessary to control not only correctness of output but also if output is appropriate or felicitous in a given context. As noticed in [5], a good generator usually relies on several factors: adequacy (similarity in meaning), fluency (syntactic correctness), readability (efficacy in context), and variation.

Chapter 4

Other

Oh and Rudnicky showed that stochastic generation benefits from two factors:

- it takes advantage of the practical language of a domain expert instead of the developer
- it restates the problem in terms of classification and labeling, where expertise is not required for developing a rule-based generation system

Non-task-oriented dialogue system

The aim of task-oriented dialogue systems is to complete specific tasks for user, non-task-oriented dialogue systems focus on conversing with human on open domains.

//TODO: more information in article [] //TODO: Info about datasets //TODO: NLP vs Computation linguistic

//TODO: In non task oriented dialogue systems it is very difficult to use template-based generation [3]

Bibliography

- [1] CHEN, H., LIU, X., YIN, D. and TANG, J. *A Survey on Dialogue Systems: Recent Advances and New Frontiers*. [Online; visited 17.11.2019]. Available at: <https://arxiv.org/pdf/1711.01731.pdf>.
- [2] CHEN, Y.-N. and GAO, J. *Open-Domain Neural Dialogue Systems*. [Online; visited 10.11.2019]. Available at: <https://www.aclweb.org/anthology/I17-5003.pdf>.
- [3] OH, A. H. and RUDNICKY, A. I. *Dialog Annotation for Stochastic Generation*. [Online; visited 10.11.2019]. Available at: <https://acl-arc.comp.nus.edu.sg/~antho/W/W00/W00-0306.pdf>.
- [4] REITER, E. and DALE, R. *Building Applied Natural Language Generation*. [Online; visited 17.11.2019]. Available at: <https://pdfs.semanticscholar.org/728e/18fbf00f5a80e9a070db4f4416d66c7b28f4.pdf>.
- [5] STENT, A., MARGE, M. and SINGHAI, M. *Evaluating Evaluation Methods for Generation in the Presence of Variation*. [Online; visited 17.11.2019]. Available at: <http://www.cs.cmu.edu/~mrmarge/cicling04eval.pdf>.