

# Lekta Tutorial and Documentation

*Document Type:* Tutorial

*Date:* 2016/09/26

*Version:* 0.1

## 1 Possible index of Lekta tutorial

1. Introduction
  - (a) What is Lekta?
  - (b) Features of Lekta Programming Language
2. Your first Lekta project
  - (a) Setting up a lekta project
  - (b) Folder and file structure
3. Automatic Speech Recognition
4. Natural Language Understanding
  - (a) Lexicons
  - (b) Grammars
5. Dialogue Manager
  - (a) Colligo
  - (b) Senso
  - (c) MindBoard
  - (d) Respondo
  - (e) Locutio
6. Natural Language Generation
  - (a) Templates
7. Text to Speech Synthesis
8. Programmer Reference
  - (a) Metatypes
  - (b) Declaring new types in Lekta
  - (c) Basic programming structures
  - (d) Lekta own commands
  - (e) Built-in functions

## 2 Introduction

Language: Set of conventional spoken or written symbols used for communication between entities. So we can see language as the linking between meaning (semantic aspect) and expression (syntactic aspect)

We have two types of languages:

- \* Natural languages: used for the communication between human beings (indeed the most natural form of communication of humans).

- \* Formal languages: used by computers and in mathematical areas.

So we have some distinctions between these two types of languages:

- \* Computers don't understand natural language. (Normal) humans don't understand computer languages.

- \* Formal languages mustn't have ambiguities but natural languages do have (some examples in a minute).

Natural Language Processing (NLP) and, by the way, what we call Language Technologies want to create some software that have some knowledge about natural languages.

NLP started with the techniques used to study formal languages and adopted (also adapted) its techniques to be able to deal with natural language.

Applications of LT's:

- \* Machine Translation.

- \* Q&A = Question Answering (Siri, Cortana).

- \* Automatic text classification (topic extraction).

- \* Automatic text summarization.

- \* Social analytics of comments in Facebook and Twitter. This topic is in fashion due to their successful predictions in some events like Eurovision song contest and opinion trending media in US president election of Donald Trump.

- \* Dialogue Systems (Spoken and Written). Our main field of action. Also known as conversational interfaces or chatbots.

Applications of conversational interfaces:

- \* Information providers. Account balance, search of available flights, ...

- \* Transactional providers. Banking transactions, flight, hotel or medical appointments booking.

- \* Educational. Tutoring in the learning of some subjects (or indeed language learning like ELEna application).

The most thrilling challenge in the design of conversational interfaces are related with the highly ambiguous nature of natural language. These ambiguities are present in almost every stage of natural language processing.

Examples:

2 syntactic forms have 1 semantic form

Peter come yesterday.

Yesterday Peter come.

1 syntactic form has 2 semantic forms

Peter said John came yesterday.

Was it yesterday when Peter said that ...?

Was it yesterday that John came ...?

Humans can deal with this kind of ambiguities applying psycholinguistic preferences and, sometimes, logic and reasoning. For example, the next sentence is as same as ambiguous than the previous one from the point of view of a computer, but everybody knows, without effort, what occurred yesterday.

Peter said John will come yesterday

We will be centered in conversational interfaces. The most used and important diagram of the architecture is this:

asr - nlu - dm - nlg - tts - database - backoffice - web services

### 3 Natural Language Understanding (NLU)

The goal of SLU stage is to transform an input string, possibly an user proference, in an abstract representation of the meaning of that proference.

This kind of representation must be computationally affordable by the computer in order to execute some kind of reasoning.

For example:

John came yesterday

SUBJECT: John

ACTION: come

TENSE: past

OFFSETDATE: -1 day

John will talk in two days

SUBJECT: John

ACTION: talk

TENSE: future

OFFSETDATE: +2 days

The NLU stage can be divided into some smaller parts:

- \* Tokenization.

- \* Speller checker.

- \* POS Tagging.

- \* Parsing.

- \* Unifier.

Tokenizer

Convert a sequence of characters into a sequence of tokens (this is aka lexical analysis). In this part we must take into account:

- \* Separators ( -\_)

- \* Punctuation marks (.,,:)

- \* Special symbols (\$€%?!)

- \* Numbers (1234)

- \* Alphanumeric codes

Example:

tk1 tk2 tk3 tk4 tk5 tk6

· The · dog · is · in · the · park ·

0 1 2 3 4 5 6 Speller Checker