Fundaments of Virtual Agents – My Transcom Experience

Eduardo Valdelomar

To illustrate how NLP works, I will use the following sentence as an example:

"*John Smith was wondering about the weather on March 15th 2018 in Stockholm*"

Applying some algorithms of the **Stanford CoreNLP software**[1] to this sentence may help to understand the different NLP perspectives:

- **Phonology**: this first level of NLP is widely used in speech recognition, segmentation and analysis. This discipline is extremely important for virtual agents, and it deserves a specific post, so I will not go into it here.
- **Morphology**: focused on the word and its composition, morphology is the basis for syntactic analysis. One interesting application of morphology is the **lemmatization**: a way of reducing each word to a canonical form. E.g. in our sample sentence, lemmatization produces the following result:

Word	Lemma
Was	*Be*
Wondering	*Wonder*

  Lemmatization retrieves the infinitive form of the verbs from their original past and gerund forms, which simplifies their utilization in further processes like decision trees.

- **Lexicography**: focused on the meaning of the words, the NLP systems require a machine-readable dictionary (MRD) to implement the lexicographical analysis. Stanford CoreNLP has gone one step further and has integrated the Wikipedia as a dictionary, marking those terms that can be found in its database. For our example, we get "*John Smith of Jamestown*", "*2018-03-15*" and "*Stockholm*" as references existing in Wikipedia that may be related to our sentence.

- **Syntax**: the syntactic analysis deals with the break-down of the speech in different constituents (sentences, phrases, words) and the links between them. There are different algorithms related to syntax analysis:

    - **Parts of speech**: this type of algorithm tag each word with its function in the sentence. CoreNLP tags the words in our sentence like this:

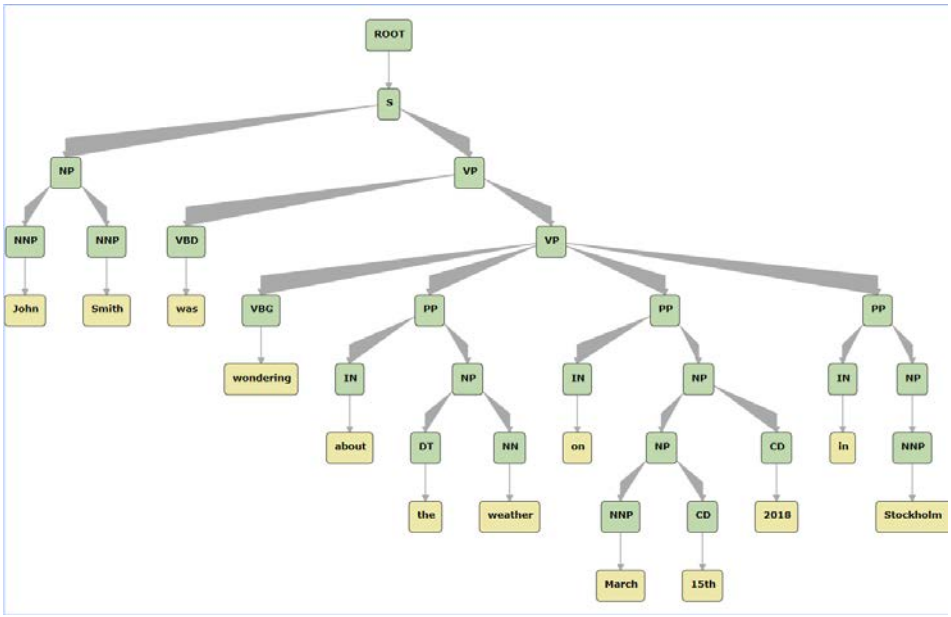
Word	Tag	Description
John	NNP	Proper noun, singular
Smith	NNP	Proper noun, singular
Was	VBD	Verb, past tense
Wondering	VBG	Verb, gerund or present participle

[1] Stanford CoreNLP is licensed under the GNU General Public License (see license details is https://stanfordnlp.github.io/CoreNLP/#license)

About	IN	Preposition or subordinating conjunction
The	DT	Determiner
Weather	NN	Noun, singular or mass
On	IN	Preposition or subordinating conjunction
March	NNP	Proper noun, singular
15th	CD	Cardinal number
2018	CD	Cardinal number
In	IN	Preposition or subordinating conjunction
Stockholm	NNP	Proper noun, singular

We must consider that one single word can have different meanings depending upon the context. E.g. March, as a proper noun, means a month, but as a verb would mean "to walk somewhere quickly and in a determined way" (Cambridge Dictionary). Therefore, correct tagging of the words is essential to properly understand the meaning of the speech.

- **Constituency parse**: it creates a tree representing the grammatical structure of the speech. In our example, the tree would look like this:



1 Chart provided using the brat visualization / annotation software.

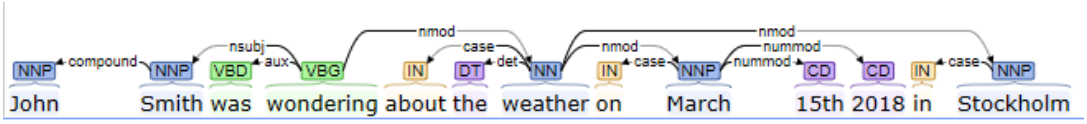
The nodes have the following meanings:

- S: sentence
- NP: Noun phrase
- VP: Verb phrase
- PP: Prepositional phrase

The nodes directly linked to the words are the tags identified in the parts-of-speech process.

Some sentences could have very different meanings, which would drive to different constituency structures. Normally the current algorithms use a **probabilistic context-free grammar** to determine the most probable structure (and consequent meaning), using **machine learning** to optimize the probabilities.

- **Dependency parse**: if constituency parse was based on the grammatical structure, dependency parse is based on how the words are directly connected each other. In our example, the dependency parse of CoreNLP will provide the following links:



2 Chart provided using the brat visualization / annotation software

We can see different types of dependencies, like **compound**, which links two nouns referred to the same object (in this case, "John Smith" is one person identified by two proper nouns), or **nsubj**, that associates the subject of the sentence with the verb (in our example, the compound noun "John Smith" is the subject who "was wondering" about the weather). A list of complete links can be consulted here .

- **Semantic**: the semantic analysis focus on the meaning of the speech, which normally requires previous morphological, lexicographical and syntactical analysis. There are many different types of algorithms depending the requirements, like the **Named Entity Recognition** (NER), which identifies places, persons, organizations, dates, amounts etc. within the sentences. In our example, the NER algorithm in CoreNLP finds the following entities:

Words	Entity type	Info
John Smith	PERSON	"John Smith"
March 15th 2018	DATE	2018-03-15
Stockholm	CITY	"Stockholm"

Also from this perspective, the **sentiment algorithms** are getting more and more importance. These algorithms are able to identify feelings in the speech, which can be extremely useful both on-line, to address the conversation, and off-line, to evaluate the service and improve the expert systems based on the accumulated experience.
CoreNLP also includes a sentiment algorithm; using it in our example, it will consider the sentence as "NEUTRAL", as in fact no feelings can be deducted from that text. But if we try with the sentence "*There is no way to obtain an answer to my question*", it is tagged as "NEGATIVE", and if we try with "That was good, thank you", the result is "POSITIVE".
Sentiment analysis use to be based in machine learning, which allows even to detect sarcasm. E.g. the sentence "*That was useful*" is tagged by CoreNLP as "NEUTRAL", but the sentence "*That was extremely useful*" is tagged as "NEGATIVE", probably because the adjective is more frequently used in a sarcastic sense.

- **Discourse**: this discipline deals with texts including several sentences, focusing on the overall meaning of the speech. Some examples of discourse analysis are **anaphora resolution**, **discourse structure recognition** and **automatic summarization**. Anaphora resolution uses some sentences in the text to solve uncertainties in other sentences. CoreNLP is also able to execute some coreference analysis: in our example, if we extend the text to say: "*John Smith was wondering about the weather on March 15th 2018 in Stockholm. He feared it could be too cold*",

the coreference algorithm tell us that, in the second sentence, "*He*" refers to "*John Smith*", and "*it*" refers to "*the weather on March 15th 2018 in Stockholm*".

- **Pragmatic**: finally, the pragmatic approach involves in the analysis some scope that is not explicitly included in the text under review. The pragmatic algorithms take advantage of wide big data to retrieve the information necessary to create the missing context.