



## Text to Image

Abhishek Sah (B15CS001), Akash Gupta (B15CS003)

Mentor: Dr. Chiranjoy Chattopadhyay

*Department of Computer Science and Engineering*

*Indian Institute of Technology, Jodhpur*

**Motivation:** To gain hands on experience with how to design a Natural Language Processing system which solves some well-defined problem in CS. The project addresses a real world scenario which can be further extended for large uses in text to graphics fields or AIs.

**Objective:** To create an application which takes as input a textual description of a geometric scene and in natural language produces the image. (Field: NLP and Graphics)

### Challenges/Research Issues:

- Finding the attributes of the shapes and mapping those attributes to shapes.
- POS Tagging, Getting a correct chunking grammar to extract the chunks to the best accuracy.
- Drawing the actual shapes relative to each other and to the scale.

**Methodology/Algorithm:** There were 2 parts of the problem. The first part was parsing the input text to find out what shapes are there and their attributes and relative position. The second part was to draw those shapes to the scale.

For part one, we employed 4 methods for parsing (Python NLTK) and analysed their results,

1. String Matching
2. POS Tagging (without context)
3. POS Tagging (with context)
4. **String Matching + POS Tagging:**

For part two, we used the python turtle library to sketch the shapes, fill the color and place them at correct relative positioning.

### Results:

- String Matching algorithm was pretty much rule based. Very strict assumption on input text.
- POS Tagging relaxed a few assumptions on input text but it failed on ambiguous POS chunks.
- String matching on top of POS Tagging gave the **best accuracy**. It had very fewer assumptions on the input text and resolved ambiguities as well.

### Conclusion:

- The complexity of the problem of parsing text and extracting useful information grows as the scope of the problem. String matching where gives best results restricts the input text to be of some fixed format.
- No POS tagger is “the best”. It all depends on the dataset on which it is being trained on.
- In the domain of our problem statement, String matching on top of POS tagging worked quite efficient.

### References:

1. <http://www.nltk.org/book/>
2. <https://docs.python.org/2/library/turtle.html>

Date:

Mentor's signature