Visualization Synthesis from Natural Language

{ Lijuan Cheng }@UC Santa Barbara

Objective and Goals

Visualizer

Visualizer is a system for synthesizing visualization from natural language. Unlike previous systems for automating data visualization, our approach is:

1. ***Data-driven:*** Rather than using a fixed set of translation rules, Visualizer leverages a *sequence-to-sequence model* that turns natural language into executable code.
2. ***Constraint-based:*** Visualizer uses an off-the-shelf *MAX-SMT solver* to generate the most likely candidates that 1) are consistent with the English description, 2) design guidelines suggested by visualization best-practice, and 3) type-constraints enforced by every input figures. . . .

Future Work

We plan to continue our current work and integrate it into RStudio IDE to reduce the workload of daily program task; We also plan to train a model for large corpus of visualizations to generate more *natural* visualizations. Finally, currently our techniques only support Matplotlib in Python and we plan to extend our framework to support other languages like R and Java.

Acknowledgments

We thank Yanju Chen in the PLSE lab for his unconditional support and guidance.

Challenges

- ***Natural language:*** Real-world natural language descriptions are both noisy and ambiguous
- ***Weak specification:*** Unlike previous program synthesis problems that leverage formal specifications or input-output examples to ensure the correctness of the desired solutions, natural language spec is weaker
- ***Complexity:*** Non-trivial data visualizations typically require a sequence of transformations, which makes it difficult for *pure machine learning* to generate the exact solutions

Our Solution

We combine the advantages of statistical and SMT-based techniques:

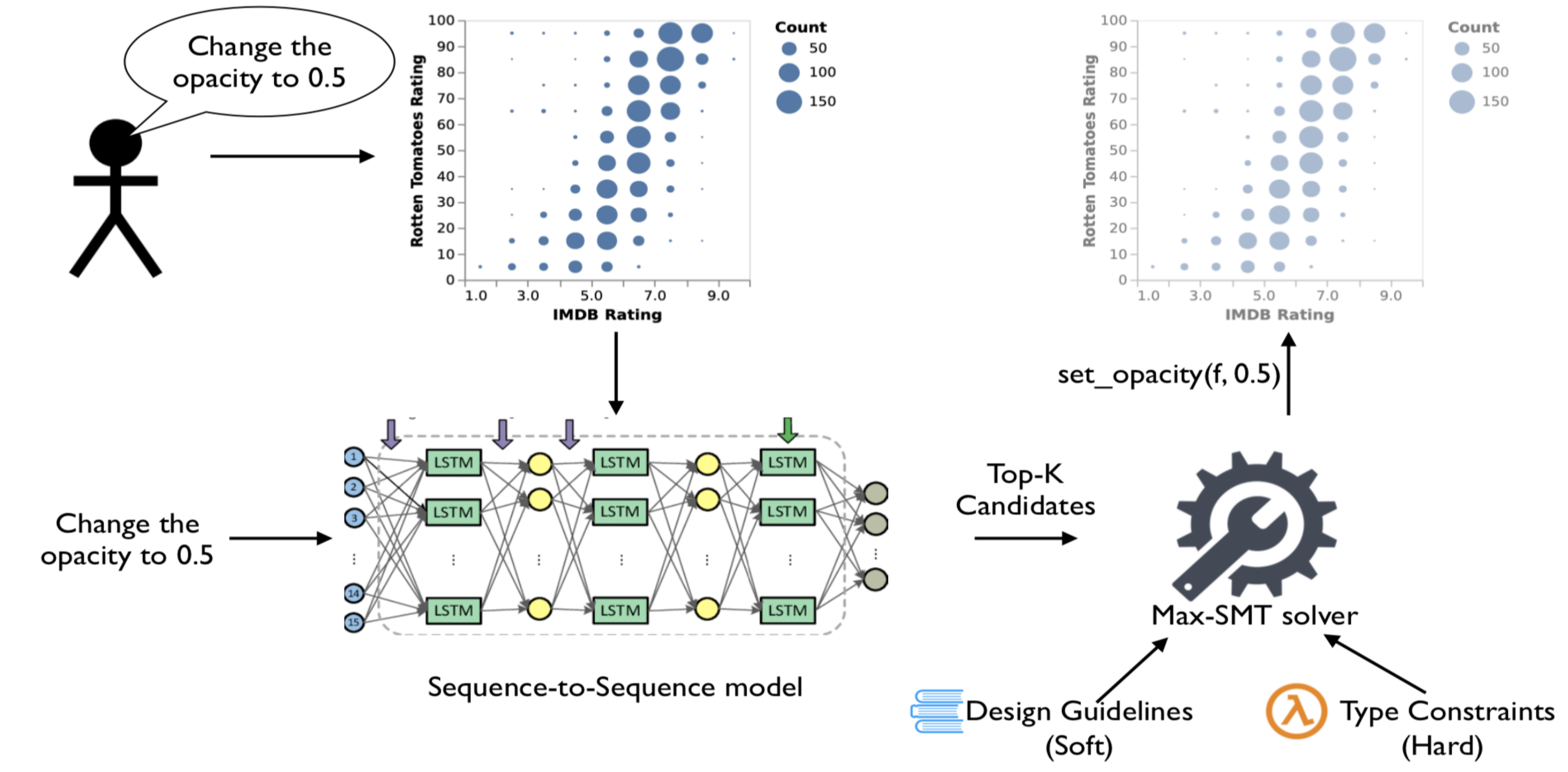
1. Use LSTM to turn a natural language description into a set of candidate programs
2. Use type-directed constraints to encode *valid transformations*
3. Use visualization design guidelines to encode *recommended transformations*
4. Formulate data visualization synthesis as a *MAX-SMT* problem (i.e., find maximum cost matching that satisfies a set of integrity constraints)

References

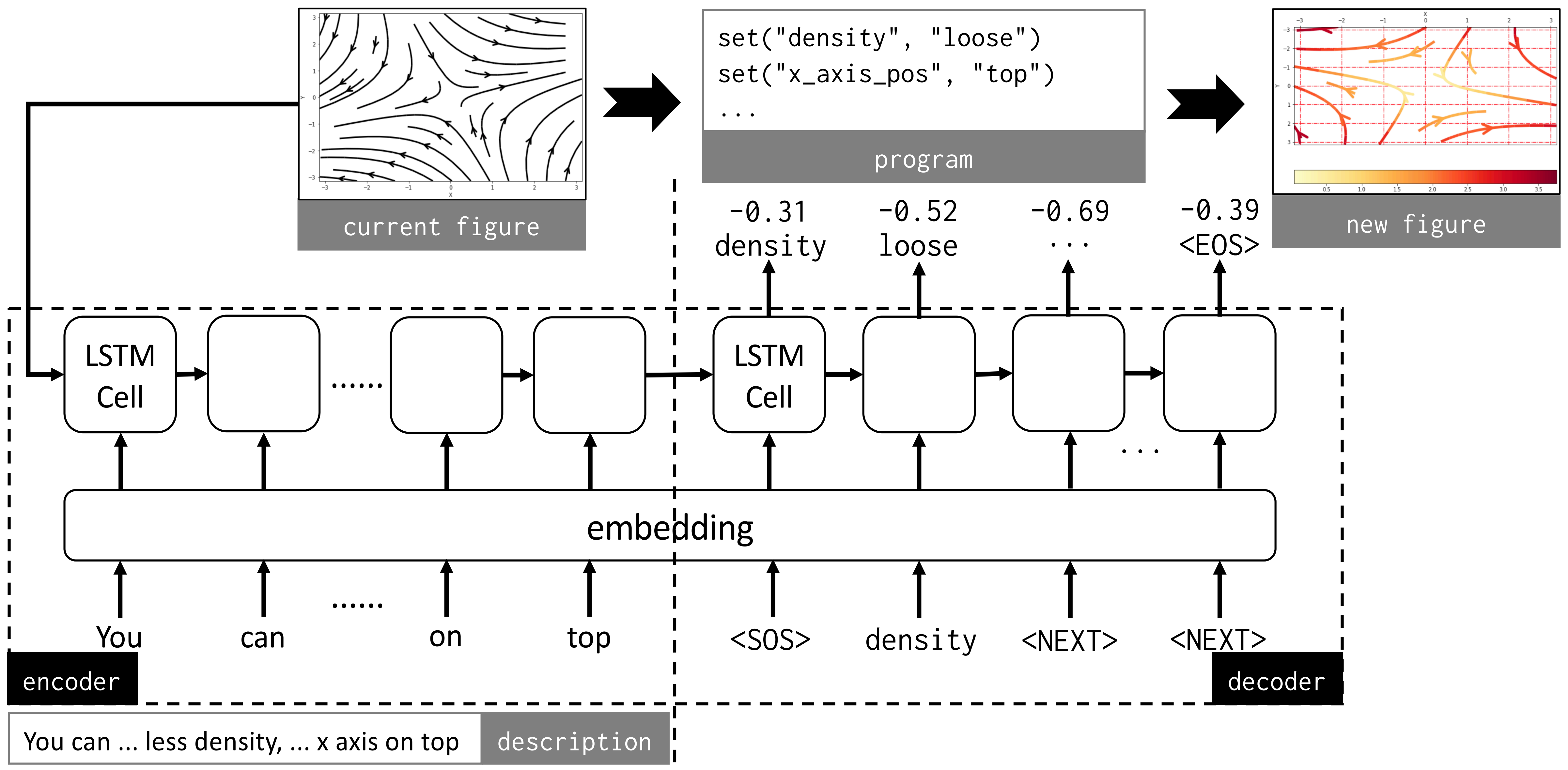
[1] Yutong Shao and Ndapa Nakashole. CHARTDIALOGS: Plotting from Natural Language Instructions In *ACL,2020*

[2] Chenglong Wang, Yu Feng, Ras Bodik, Alvin Cheung, Isil Dillig. Visualization by Example. In *POPL 2020*

Overview of our approach



Neural Architecture



```
set("density", "loose")
set("x_axis_pos", "top")
...
```

Given a *question-solution* pair (D, S), where a *question* is a user description composed by word tokens d: $D = (d_1, d_2, \ldots, d_n)$, and a *solution* is a symbolic program composed by a sequence of functions s_i: $S = (s_1, s_2, \ldots, s_m)$, the *seq2seq* model is used to estimate the probability of $P(S|D)$, which is then given by: $P(S|D) = P(s_1, s_2, \ldots, s_m | d_1, d_2, \ldots, d_n) = \prod_{t=1}^{m} P(s_t | v, s_1, s_2, \ldots, s_{t-1})$.