

VisuaLLM : Playground for Seq2Seq Generation

František Trebuňa
ferotre@gmail.com

Ondřej Dušek
odusek@ufal.mff.cuni.cz

Plug in your LM & dataset and see what's going on inside!

Set Up a Generation Playground

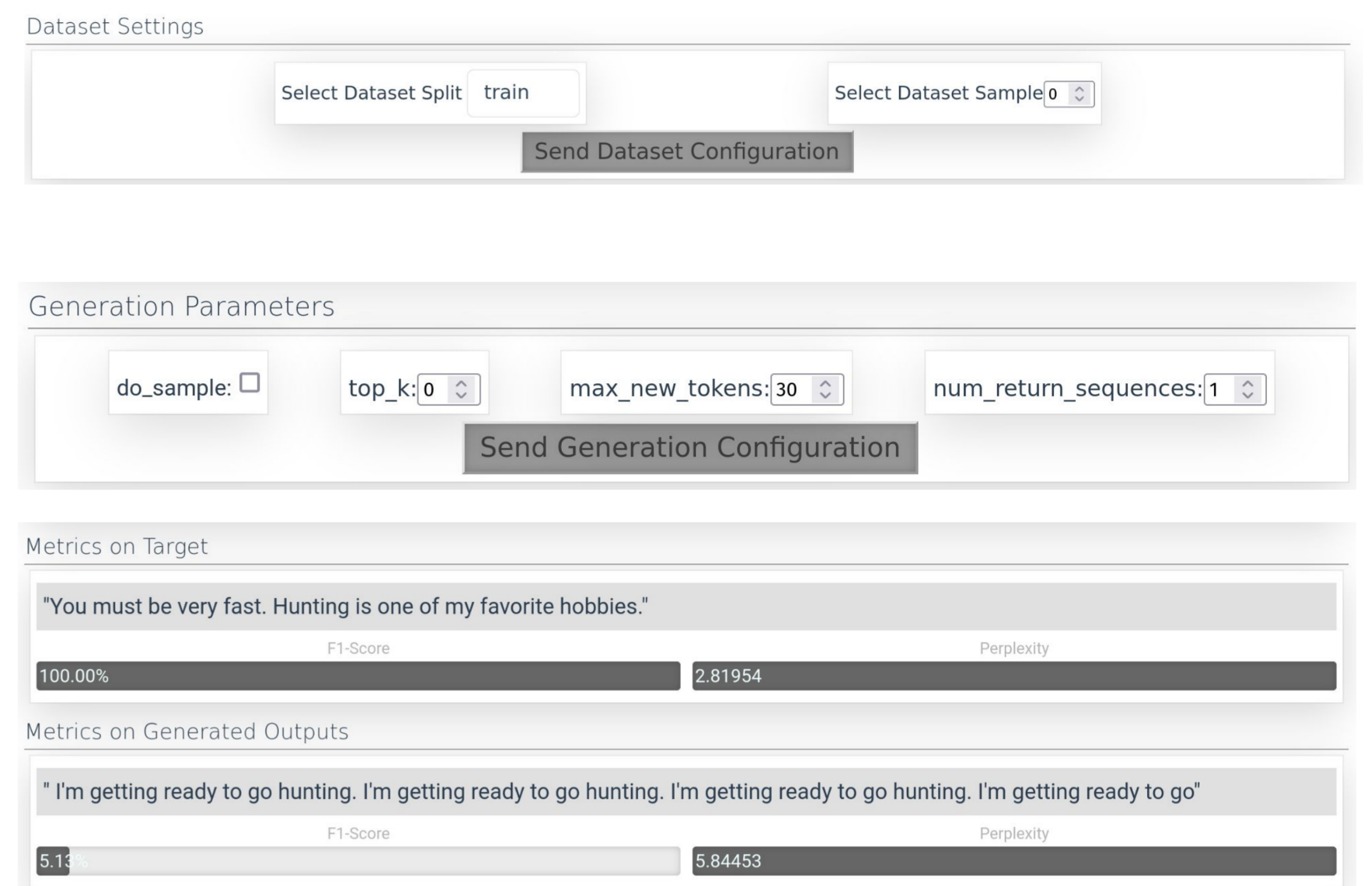
```
Generation(  
  dataset=dataset,  
  model=model,  
  tokenizer=tokenizer,  
  selectors={  
    "do_sample": CheckBoxSelectorType(False),  
    "top_k": MinMaxSelectorType(0, 1000),  
    "max_new_tokens": MinMaxSelectorType(10, 100, default_value=30),  
    "num_return_sequences": MinMaxSelectorType(1, 20),  
  },  
  metrics_on_probs={  
    "Perplexity": ProbsMetric("{:.5f}", False, Perplexity())  
  },  
  metrics_on_generated_text={  
    "F1-Score": GeneratedTextMetric("{:.2%}", True, F1Score())  
  },  
)
```

Prepared for HuggingFace Datasets 🤗

Prepared for HuggingFace Transformers 🤗

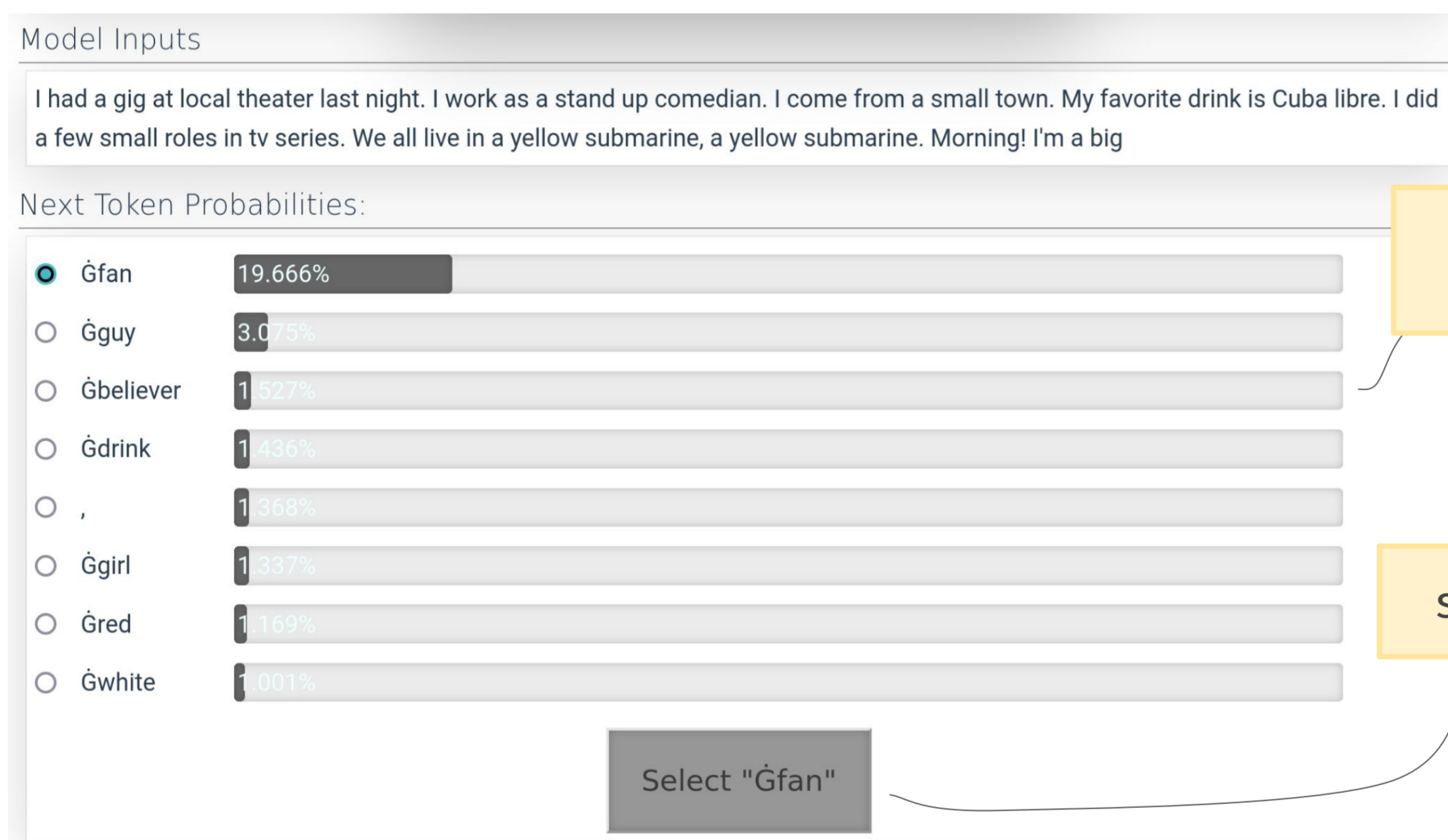
Specify Generation Parameters

Specify Metrics
(Compliant with TorchMetrics)



Go Through Generation Step by Step

```
NextTokenPrediction(model=model, tokenizer=tokenizer, dataset=dataset)
```



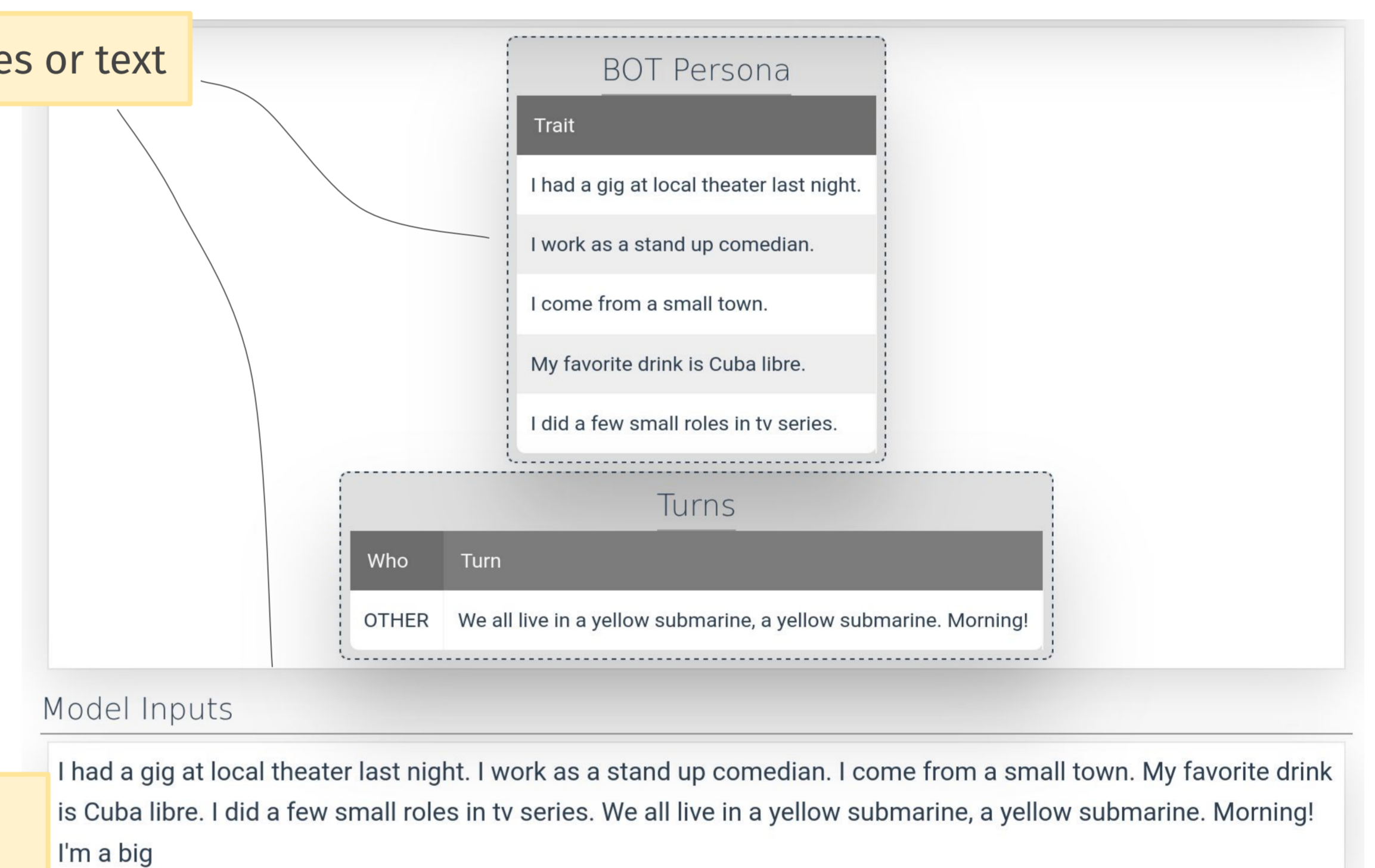
see model inputs as tables or text

see each token's probability distribution

steer the generation manually

modular: you have total control over appearance

Visualize Dataset Samples



Customize to your liking

each Python element has its Vue.js counterpart

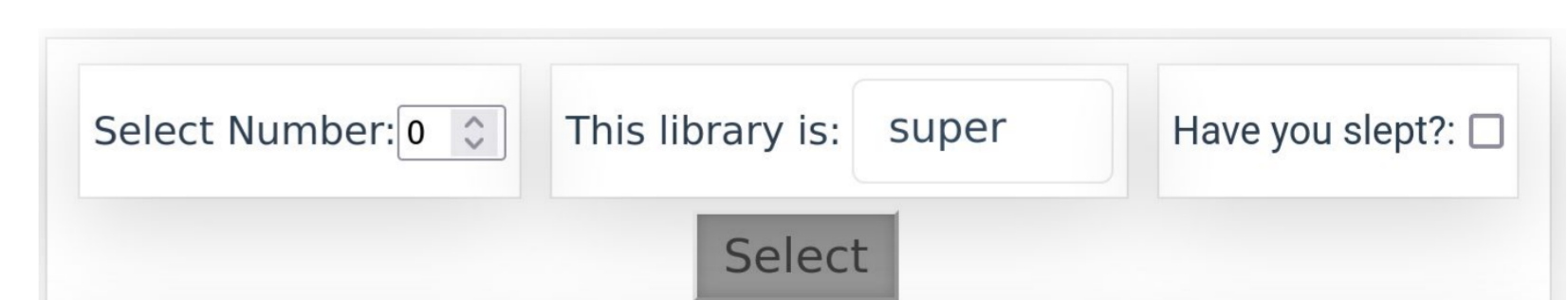
frontend automatically recreates itself based on schema sent by the library

```
table_element = TableElement()  
headers = ["No.", "Turn"]  
rows = [  
  [i, x]  
  for i, x in enumerate(  
    [  
      "This is first row",  
      "This is second row",  
      "This is third row",  
      "This is fourth row",  
      "This is fifth row",  
    ]  
  )  
]  
TABLE_NAME = "Table1 is a Great Table"  
table_element.add_table(TABLE_NAME, headers, rows)
```

No.	Turn
0	This is first row
1	This is second row
2	This is third row
3	This is fourth row
4	This is fifth row

```
number_selector_element = MinMaxSubElement(  
  sample_min=0, sample_max=10, text="Select Number:"  
)  
choices_element = ChoicesSubElement(  
  choices=["super", "magnificent", "incredible"], text="This library is:"  
)  
checkbox_element = CheckBoxSubElement(text="Have you slept?:")  
button_element = ButtonElement(  
  processing_callback=on_button_clicked,  
  subelements=[  
    number_selector_element,  
    choices_element,  
    checkbox_element,  
  ],  
)
```

you can extend components without touching the frontend



Try it!

> pip install visuallm

<https://github.com/gortibaldik/visuallm>

Presented at INLG 2023, Prague.

Supported by the project TL05000236 AI asistent pro žáky a učitele co-financed by the Technological Agency of the Czech Republic and by the ERC (No. 101039303 NG-NLG). Resources provided by the LINDAT/CLARIAH-CZ Research Infrastructure.



CHARLES UNIVERSITY