## PromptSource: IDE + repo for natural language prompts

**Paper:** S. Bach et al., "PromptSource: An Integrated Development Environment and Repository for Natural Language Prompts", ACL Demo (2022)

## Motivation

Prompting: represent task as an utterance

E.g. text classification prompt: "This video class label



describes PromptSource. It is about..." ~ "NLP"

Prompt applications enables adapting LMs to ad hoc tasks

sample efficiency in low data regimes

Prompt engineering can have a major influence

Training on prompts can enable task generalisation

**References:** 

T. Brown et al., "Language models are few-shot learners", NeurIPS (2020)

T. Schick et al., "It's Not Just Size That Matters: Small Language Models Are Also Few-Shot Learners", NAACL-HLT (2021)

Le Scao et al., "How many data points is a prompt worth?", NAACL-HLT (2021)

What it is Why it is needed





# Prompt creation $\neq$ traditional NLP annotation

What makes prompt creation different?

Functions, not labels

Prompts are functions that map examples to input/target pairs - how expressive should format be?

Datasets, not examples

Unlike labels, prompt design must consider all dataset examples - what interface will support this?

Variation is desirable

Label variation generally undesirable but prompt variation has benefits - how can this be supported?



# The PromptSource workflow

**Example Task:** design prompt query for SNLI Require answers that can map to SNLI classes

	S1: Exploration	S2, S3,
<ul> <li>The second sec</li></ul>	<pre>The SNLI corpus (version 1.0) is a collection of 570k human-written English sentence pairs manually labeled for the task of NLL  { premise: "A person", hypothesis: "A person", label: 1 }  { premise: "The kids", hypothesis: "All kids", label: 2 }</pre>	Sourcing SNLI SNLI SNLI SVLI SVLI SVLI SVLI SVLI SVLI SVLI SV
	S1: Exploration	S2: Writing

Image credits/Reference:

S. Bach et al., "PromptSource: An Integrated Development Environment and Repository for Natural Language Prompts", ACL Demo (2022)

S. Bowman et al., "A large annotated corpus for learning natural language inference", EMNLP (2015)



# PromptSource plays nicely with Source plays nicely with

## **Import** libraries

Practical usage

## Fetch prompt by name

p = prompts[prompt key]

### Load example

from promptsource.templates import DatasetTemplates from datasets import load dataset prompts = DatasetTemplates("snli") prompt\_key = "based on the previous passage" dataset = load\_dataset("snli", split="train") result = p.apply(example) print("INPUT: ", result[0]) A person ....? Yes, no, or maybe?

example = dataset[0]

### Apply to an example

print("TARGET: ", result[1]) Maybe

Image credits/Reference:

S. Bach et al., "PromptSource: An Integrated Development Environment and Repository for Natural Language Prompts", ACL Demo (2022) Q. Lhoest et al., "Datasets: A community library for natural language processing", EMNLP Demo (2021)



# Prompt Template Engine: Jinja2

Jinja2 template engine used for prompts

- More flexible vs rule-based generation
- Simpler than pure Python code

Example prompt:

If {{premise}} holds, does {{hypothesis}} also hold? ||| {{entailed}}

Placeholder refer to fields in example dict

Separator

between condition and completion

Jinja2 enables some fancy string manipulation In practice, simple manipulations suffice

References

S. Bach et al., "PromptSource: An Integrated Development Environment and Repository for Natural Language Prompts", ACL Demo (2022) (Example dict format) Q. Lhoest et al., "Datasets: A community library for natural language processing", EMNLP Demo (2021) (Guidelines for prompt authors) https://github.com/bigscience-workshop/promptsource/blob/main/CONTRIBUTING.md



## Useful idioms

Template may not be applicable for all examples

Conditionals can be used to skip examples (empty)

Examples can generate multiple training instances

Elements can be selected with the choice function

Some examples may have multiple valid completions These can be specified as a separate field



## User Interface: Dataset Browsing



Github - Promptsource

X

Choose a mode

Prompted dataset viewer

#### Promptsource 🌸 -**Prompted dataset viewer**

Dataset ③			
ag_n&ws	•		
Split			
train	•		
No of prompts created for ag_news	: 8		
Prompt name			
Hello	•		
Select the example index (Size = 120000)	?		
0 –	+		
Dataset Schema			
<pre>"label" : [</pre>			
0 : "World"			
1 : "Sports"			

#### Dataset: ag\_news

Homepage: http://groups.di.unipi.it/~gulli/AG corpus of news articles.html

Dataset: https://github.com/huggingface/datasets/blob/master/datasets/ag\_news/ag\_news.py

AG is a collection of more than 1 million news articles. News articles have been gathered from more than 2000 news sources by ComeToMyHead in more than 1 year of activity. ComeToMyHead is an academic news search engine which has been running since July, 2004. The dataset is provided by the academic comunity for research purposes in data mining (clustering, classification, etc), information retrieval (ranking, search, etc), xml, data compression, data streaming, and any other non-commercial activity. For more information, please refer to the link http://www.di.unipi.it/~gulli/AG corpus of news articles.html

The AG's news topic classification dataset is constructed by Xiang Zhang (xiang.zhang@nyu.edu) from the dataset above. It is used as a text classification benchmark in the following paper: Xiang Zhang, Junbo Zhao, Yann LeCun. Character-level Convolutional Networks for Text Classification. Advances in Neural Information Processing Systems 28 (NIPS 2015).

#### Prompt

Name Hello

Reference

**Original Task?** None

**Choices in template?** None

Metrics None

Prompt Languages None

**Answer Choices** None

#### **References:**

S. Bach et al., "PromptSource: An Integrated Development Environment and Repository for Natural Language Prompts", ACL Demo (2022) https://github.com/streamlit

## useful for verifying prompts

### across many examples

 $\equiv$ 

### Dataset browsing



## User Interface: Sourcing

×	<ul> <li>Task description localization: where is the task description blended with the inputs? In the beginning, in the middle, a end?</li> </ul>	t the
Github - Promptsource	<ul> <li>Implicit situation or contextualization: how explicit is the query? For instance, Given this review, would you buy this priss an indirect way to ask whether the review is positive.</li> </ul>	roduct?
Choose a mode		
Sourcing -	Name	
	based on the previous concise and insightful passage	
Promptsource 🔅 - Sourcing	Prompt Reference	?
Dataset ③	Adapted (and extended) from the BoolQ prompts in Schick & Schütze 2021.	
snli -	✓ Original Task? ③	
Split	Choices in Template? ②	
train	Metrics	?
tiani	Accuracy ×	0 -
No of prompts created for snli:16		
Colored Free marks	Prompt Languages	?
Select Example	en (English) ×	0 -
0	Answer Choices	(?)
0 550151	Vee III Meybe III Ne	
<b>▼</b> ξ	fes    Maybe    No	
"premise" : "A person on a horse jumps over a	Template	
broken down airplane." "hypothesis" : "A person is training his horse for a competition."	{{premise}} Based on the previous concise and insightful passage, is it true that "{{hypothesis}}"? Yes, no, or maybe?     answer_choices[label] }}	.{{
"label" : 1 }	Save	
Dataset Schema		
τ. {		
"premise" : "string"		

**References:** 

S. Bach et al., "PromptSource: An Integrated Development Environment and Repository for Natural Language Prompts", ACL Demo (2022)

#### Input

A person on a horse jumps over a broken down airplane. Based on the previous concise and insightful passage, is it true that "A person is training his horse for a competition."? Yes, no, or maybe?

 $\equiv$ 

#### Target

Maybe

#### preview

K

### Prompt sourcing

## User Interface: Helicopter view



**References:** 

S. Bach et al., "PromptSource: An Integrated Development Environment and Repository for Natural Language Prompts", ACL Demo (2022)

	≡
g.	
4	
in logscale	
S.	
	view

# **Community Guidelines**

Describing what makes a good prompt is hard Community guidelines evolved through iteration Key objectives for guidelines:

- Standardised vocab & minimum requirements
- Highlight common errors & best practices
- Gather useful metadata for future research The guidelines cover templates and metadata

**References:** 

S. Bach et al., "PromptSource: An Integrated Development Environment and Repository for Natural Language Prompts", ACL Demo (2022) (Guidelines for prompt authors) https://github.com/bigscience-workshop/promptsource/blob/main/CONTRIBUTING.md

### **Encourage:**

- explicitly state possible completions
- remove spurious ambiguity from targets
- creation of multiple prompt variants

### **Require**:

- only natural language prompts allowed
- inclusion of metadata (e.g. reference paper)



## Case Studies with PromptSource

### Massively multitask prompted training

TO (Sanh et al., 2021) uses a

multitask mixture of prompts for

training to boost generalisation

Training and evaluation use P3



**References:** 

S. Bach et al., "PromptSource: An Integrated Development Environment and Repository for Natural Language Prompts", ACL Demo (2022)

V. Sanh et al., "Multitask Prompted Training Enables Zero-Shot Task Generalization", ICLR (2021)

V. Lin et al., "Few-shot learning with multilingual language models", arXiv (2021)

S. Min et al., "MetalCL: Learning to learn in context", arXiv (2021)

## Multilingual prompting

- XGLM (Lin et al., 2021) train on
- 30 languages to study cross-
- P3 quality English prompts



## Prior work



**References:** 

S. Bach et al., "PromptSource: An Integrated Development Environment and Repository for Natural Language Prompts", AC (GPT-3) T. Brown et al., "Language models are few-shot learners", NeurIPS (2020)

(PET) T. Schick et al., "Exploiting Cloze-Questions for Few-Shot Text Classification and Natural Language Inference", EACL ( (T0) V. Sanh et al., "Multitask Prompted Training Enables Zero-Shot Task Generalization", ICLR (2021)

(FLAN) J. Wei et al., "Finetuned Language Models are Zero-Shot Learners", ICLR (2021)

(XGLM) V. Lin et al., "Few-shot learning with multilingual language models", arXiv (2021)

(MetaICL) S. Min et al., "MetaICL: Learning to learn in context", arXiv (2021)



1 Demo (2022)	(GATE) H. Cunningham, "GATE, a general architecture for text engineering." Computers and the Humanities (2002) (BRAT) P. Stenetorp et al. "BRAT: a web-based tool for NLP-assisted text annotation". EACL (2012)
	(MyMiner) D. Salgado et al., "MyMiner: a web application for computer-assisted biocuration and text annotation",
(2021)	Bioinformatics (2012)
	(YEDDA) J. Yang et al., "YEDDA: A Lightweight Collaborative Text Span Annotation Tool", ACL (2018)
	(AlpacaTag) B. Lin et al., "AlpacaTag: an active learning-based crowd annotation framework for sequence tagging"
	(2019)
	(TreeAnnotator) P. Helfrich et al., "TreeAnnotator: versatile visual annotation of hierarchical text relations", LREC (20

#### ce tagging", ACL s", LREC (2018)