ScatterShot: Interactive In-context Example Curation for Text Transformation

Sherry Tongshuang Wu, Carnegie Mellon University
Hua Shen, PennState University
Daniel S. Weld, University of Washington
Jeffrey Heer, University of Washington
Marco Tulio Ribeiro, Microsoft
What is prompt-based learning with LLMs?

Encourages a **pre-trained** Large Language Model (LLM) to make **particular predictions** by providing a **“prompt”** specifying the task to be done.

Figure from: Liu Pengfei, et al. "Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing." arXiv 2021
What is prompt-based learning with LLMs?

Encourages a **pre-trained** Large Language Model (LLM) to make **particular predictions** by providing a "prompt" specifying the task to be done.
What is in-context learning?

The input to the model describes a new task with some possible examples, **in natural language**. Effective on **very large models** (173B GPT-3)

In-context learning: Prompt types

Zero-shot
Natural language descriptions only

Find the nationality of people:

1. Find the nationality of people:
   2. Marie Curie =>

One-shot
Description + one example

Find the nationality of people:

1. Find the nationality of people:
   2. Albert Einstein => German
   3. Marie Curie =>

Few-shot
Description + a few example (3-100) [5-10 is most common]

Find the nationality of people:

1. Find the nationality of people:
   2. Albert Einstein => German
   3. Alan Turing => English
   4. Mahatma Gandhi => Indian
   5. Marie Curie =>

How to make?

Challenge: which sets of examples?

Let’s assume users are given a training data set to choose prompt examples.

**Different** few-shot example sets lead to very different results.

**Different ordering** of the same set also lead to different results!!

Challenge: when “enough” examples?

The model performs better when the test input is similar to some training input. But it’s hard to get coverage in 30 examples.

We present, ScatterShot, to help users interactively and iteratively find high-quality demonstrative examples to build effective in-context functions.
Scattershot principles

1. Handle common patterns
   Help the user discover previously unexplored patterns.

2. Not neglect unusual ones
   Help the user prioritize the most informative examples.

3. Cost effective
   Minimize annotation cost.
User interface

Extract all the mentioned dates as detailed as possible, in the ISO format of YEAR-MONTH-DAY.

How can we use the least examples to cover most prompt patterns?
**Scattershot algorithm**

**Input-output pairs, iteration 1 to i − 1**

nineteen ninety-six == 1996

today == 2000-01-05

[Posted: 2000-01-06] *He was plucked on Thanksgiving Day.*
Thanksgiving == 1999-11-25

---

**Existing prompt examples**
Scattershot algorithm

Input-output pairs, iteration 1 to i − 1

nineteen ninety-six == 1996

today == 2000-01-05

[Posted: 2000-01-06] He was plucked on Thanksgiving Day.
Thanksgiving == 1999-11-25

Key phrase templates

- PRON (Halloween, Thanksgiving)
- DATE (today, Oct. 23, 1999)
- NUM years ago (24 years ago)

Extract key phrases & slices
## Slice-based Sampling

### Input-output pairs, iteration 1 to \(i - 1\)

- [Posted: 2000-01-05] Photo: on today . today \(==\) 2000-01-05
- [Posted: 2000-01-06] He was plucked on Thanksgiving Day. Thanksgiving \(==\) 1999-11-25

### Key phrases & data slices, iteration \(i\)

<table>
<thead>
<tr>
<th>Key phrases &amp; data slices, iteration (i)</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>B</td>
</tr>
<tr>
<td><strong>Key phrases templates</strong></td>
<td></td>
</tr>
<tr>
<td>PRON  (Halloween, Thanksgiving)</td>
<td></td>
</tr>
<tr>
<td>DATE  (today, Oct. 23, 1999)</td>
<td></td>
</tr>
<tr>
<td>NUM years ago (24 years ago)</td>
<td></td>
</tr>
</tbody>
</table>

**Extract key phrases & slices**

- ❌ [Posted: 1989-10-31] It hopes to control 5% of jewelry business.
- ❌ [Posted: 2014-10-12] HALLOWEEN SHOW FOR HSBC FAMILY…
- ❌ [Posted: 2000-01-06] He was plucked on Thanksgiving Day.
- ❗️ [Posted: 2014-12-25] @viereedom Merry Christmas!
- ❌ [Posted: 2015-01-02] Are you going to yoga today?
- ❌ [Posted: 2015-04-20] But it’s already 10 months ago!!
- ❖ [Posted: 2015-01-02] Are you going to yoga today?
- ✔️ [Posted: 2015-03-21] Her last run was 24 years ago.
Prioritize sampled examples

Input-output pairs, iteration 1 to \( i - 1 \)

  - nineteen ninety-six \( \Rightarrow 1996 \)

  - today \( \Rightarrow 2000-01-05 \)

- [Posted: 2000-01-06] He was plucked on Thanksgiving Day.
  - Thanksgiving \( \Rightarrow 1999-11-25 \)

- [Posted: 2014-10-19] Lunch at Agnes B Cafe
  - Lunch at Agnes B Cafe

Key phrases & data slices, iteration \( i \)

|   | \( n=449 \) |
| B | [Posted: 1989-10-31] It hopes to control 5% of jewelry business |
|   | \( m=10 \) |
|   | \( k=4 \) |
|   | \( \mu=4.82 \) |
|   | \( m=2 \) |
|   | \( k=0 \) |
|   | \( \mu=4.34 \) |
|   | [Posted: 2014-12-25] @viereedom Merry Christmas! |
|   | \( n=19 \) |

- [Posted: 2014-10-12] HALLOWEEN SHOW FOR HSBC FAMILY...
  - HALLOWEEN SHOW FOR HSBC FAMILY...

|   | [Posted: 2015-01-02] Are you going to yoga today? |
|   | \( n=113 \) |
|   | \( m=3 \) |
|   | \( k=3 \) |
|   | \( \mu=1.14 \) |

Prioritize similar data that has **low performance**, are **large**, and slices that have **not been** sampled many times.

\[
\mu_{i,c} = \left(1 - \frac{k}{m}\right) \cdot \ln n + \sqrt{\frac{\ln t}{m}}
\]

- Error rate
- Size
- Sample Rarity

Slice \( c \) has \( n \) examples, \( m \) are labeled in previous iterations.
Out of \( m \), the current function is correct on \( k \).
How to handle no ground truth labels?

We estimate function quality by re-ordering stability.

[Posted: 2014-12-25] @viereedom Merry Christmas!

- Unanimity voting
  - Christmas == 2014-12-25
  - Christmas == 2014-12-25
  - Christmas == 2014-12-25

- Manual inspection
  - Keep Christmas == 2014-12-25


- Unanimity voting
  - nineteen ninety-six == 1996-01
  - nineteen ninety-six == 1996
  - nineteen ninety-six == 1996

- Manual inspection
  - Edit nineteen ninety-six == 1996
Scattershot evaluation

1. Simulation Experiment
   - Simulate the labeling process

2. Within-subject User Study
   - 10 person evaluation
   - QA-pair rewriting task

Temporal Expression Extraction

  - today == 2000-01-05

  - Oct. 23, 1999 == 1999-10-23

- [Posted: 1989-10-31] It hopes to control 5% of jewelry business
  - N/A

Question-Answer Pair Rewriting

- Q: Where are the buildings? A: in distance
- Q: Are the buildings in distance? A: yes
- Q: Why is it dark? A: twilight
- Q: Is it dark because of the twilight? A: yes
- Q: Is the water warm or cold? A: cold
- Q: Is the water cold? A: yes
## Simulation performance

### Temporal

<table>
<thead>
<tr>
<th>Conditions</th>
<th>Extraction</th>
<th>Normalization</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>F1</td>
<td>Precision</td>
</tr>
<tr>
<td>Random</td>
<td>73.2 ± 4.0</td>
<td>74.0 ± 3.8</td>
</tr>
<tr>
<td>ScatterShot</td>
<td>75.0 ± 2.9</td>
<td>75.6 ± 2.8</td>
</tr>
</tbody>
</table>

### QA-Pair

<table>
<thead>
<tr>
<th>Conditions</th>
<th>ROUGE-L</th>
<th>BLEU-4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rule-based</td>
<td>78.4</td>
<td>66.7</td>
</tr>
<tr>
<td>Random</td>
<td>74.3 ± 3.9</td>
<td>65.4 ± 3.5</td>
</tr>
<tr>
<td>ScatterShot</td>
<td>80.0 ± 3.5*</td>
<td>69.1 ± 3.1*</td>
</tr>
</tbody>
</table>

The significant improvements, measured by the student’s t-test are marked with *: p<0.05, and **: p<0.01.

### Quantitative Results:

- ✔️ Compared with the Random condition, ScatterShot outperformed the baseline on all metrics.
### Coverage
Transforms more forms of inputs.

<table>
<thead>
<tr>
<th>Input</th>
<th>Rule-based</th>
<th>Random</th>
<th>ScatterShot</th>
</tr>
</thead>
<tbody>
<tr>
<td>Q: Are there more girls or boys? A: equal</td>
<td>(No generation)</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Q: How many hairs does the sheep in front have? A: infinite</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
</tbody>
</table>

### Fluency
Generates outputs that sound natural.

<table>
<thead>
<tr>
<th>Input</th>
<th>Rule-based</th>
<th>Random</th>
<th>ScatterShot</th>
</tr>
</thead>
<tbody>
<tr>
<td>Q: What make is the phone? A: vtech</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Q: What does the woman have on her face? A: headband</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
</tbody>
</table>

### Correctness
Produces desired outputs (the new question-pair are logically equivalent to the original pair).

<table>
<thead>
<tr>
<th>Input</th>
<th>Rule-based</th>
<th>Random</th>
<th>ScatterShot</th>
</tr>
</thead>
<tbody>
<tr>
<td>Q: What monument are they next to? A: unknown</td>
<td>(No generation)</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Q: What type of motorcycle is in the picture? A: mountain</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
</tbody>
</table>

Compared with the Random condition, and a **Rule-based** system: ScatterShot functions tend to have better coverage, fluency, and correctness.
We evaluate the held-out test set every time we add five more examples to the in-context bucket until the stop condition is satisfied. ScatterShot tends to frequently outperform Random, and tends to have better performance.
User Study Performance

Active learning is effective for humans (More holistic view)!

I went through several rounds of pretty similar examples in Step 2 (Random), thinking the function is behaving quite decently, and didn’t realize the function needed more diverse and edge cases until I reached Step 3.
Performance of user created function

### Table 3: The performances of participants' in-context functions after each step. +/- represents the average performance change

<table>
<thead>
<tr>
<th>Condition</th>
<th>Step 1</th>
<th>Step 2</th>
<th>Step 3</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>M-R-S</td>
<td>/ (59.3)</td>
<td>+17.4 (74.7)</td>
<td>+3.2 (77.8)</td>
<td></td>
</tr>
<tr>
<td>M-S-R</td>
<td>/ (61.8)</td>
<td>+18.1 (75.4)</td>
<td>-0.4 (74.9)</td>
<td></td>
</tr>
</tbody>
</table>

(a) ROUGE-L

<table>
<thead>
<tr>
<th>Condition</th>
<th>Step 1</th>
<th>→ Step 2</th>
<th>→ Step 3</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>M-R-S</td>
<td>/ (63.9)</td>
<td>+10.1 (74.0)</td>
<td>+3.1 (76.9)</td>
<td></td>
</tr>
<tr>
<td>M-S-R</td>
<td>/ (65.3)</td>
<td>+8.9 (74.2)</td>
<td>-0.6 (73.6)</td>
<td></td>
</tr>
</tbody>
</table>

(b) BLEU-4

+/­: represents the average performance change compared to the prior step, (number) are the absolute performance.

**M-R-S**: users build in-context functions using methods of “Manual - Random - ScatterShot” in sequence.

**M-S-R**: users use “Manual - ScatterShot - Random” methods in sequence.

---

**M-R-S** users were able to keep adding useful examples, whereas **M-S-R** users decreased the function performance by 0.6 in Step 3 (ScatterShot -> Random), indicating that these efforts were wasted.
What’s more?

✓ Slice-based sampling can increase **data space coverage**

✗ Random sampling performs less

✓ Interacting with the latest function for users is essential for in-context learning.

✓ Human-AI collaborative labeling for building better functions results in better quality and better task definition.
Takeaways

**ScatterShot** helps users find *informative input examples* in the unlabeled data, *improves* the annotator’s awareness and handling of diverse patterns, and ultimately, the *in-context function performance*. 
The full user study instructions, and the detailed exit survey, are at:

GitHub: https://github.com/tongshuangwu/scattershot

Thank You!

Sherry Tongshuang Wu
sherryw@cs.cmu.edu
@tongshuangwu

Hua Shen
huashen218@psu.edu
@huashen218

Daniel S. Weld
weld@cs.uw.edu

Jeffrey Heer
jheer@cs.uw.edu

Marco Tulio Ribeiro
marcotcr@microsoft.com