1. An ever-growing predicate paraphrase resource

- Binary verbal predicate paraphrases
- Extracted from Twitter
- Up to 86% accuracy
- Ever-growing resource:
  - Currently ~0.8M paraphrases
  - Complementary to other resources

Get the resource!
https://github.com/vered1986/Chirps

2. Assumptions

- Main assumption: redundant news headlines of the same event are likely to describe it with different words [1].
- This work: propositions extracted from tweets discussing news events, published on the same day, that agree on their arguments, are predicate paraphrases.

![Image of news headlines and tweets]

3. Resource Creation

1. Collecting News Tweets

Using Twitter Search API:

```
get_tweets(lang=en, filter=news) → 
→ clean_tweets()
```

Amazon is buying Whole Foods in $13.7B
Amazon to acquire Whole Foods Market in deal valued at nearly $14 billion...

2. Proposition Extraction

- Extract propositions from the tweets using PropS [2]:
  - https://github.com/gabrielStanovsky/props
- Get binary verbal predicate templates, and apply argument reduction [3].

3. Generating Paraphrase Instances

- We consider two predicates as paraphrases if:
  1. They appear on the same day.
  2. Each of their arguments aligns with a unique argument in the other predicate.
- Two levels of argument matching: strict (exact match / short edit distance) and loose (partial token matching / WordNet synonyms)

<table>
<thead>
<tr>
<th>Predicate 1</th>
<th>Predicate 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>[Amazon] buy [Whole Foods]</td>
<td>[Amazon] acquire [Whole Foods Market]</td>
</tr>
</tbody>
</table>

Heuristic score for a predicate paraphrase type:

\[ s(p_1, p_2) = \frac{\text{count}(p_1, p_2)}{N + \text{days}(p_1, p_2)} \]

- \( \text{count}(p_1, p_2) \) assigns high scores for frequent paraphrases
- \( N \) - number of days since the resource collection begun
- \( \text{days}(p_1, p_2) \) eliminates noise from two arguments participating in different events on the same day, e.g.: 1) Last year when Chuck Berry turned 90; 2) Chuck Berry dies at 90

4. Generating Types

- We release our resource daily, with two files:
  - Instances: predicates, arguments and tweet IDs.
  - Types: predicate paraphrase pair types ranked in a descending order according to the heuristic accuracy score.

5. Resource Release

References