1. Motivation

- **Topic**: Compositional distributional models of phrase/sentence meaning.
- **What**: Apply the Practical Lexical Function (PLF) model (Paperno et al. 2014) to Croatian, a free word order language.
- **Why**: PLF is built on observations of predicate-argument combinations that seem to work well on English, but are harder to recover in free word order languages.
- **How**: We evaluate the PLF model, together with different variants of the PLF (Gupta et al. 2015) and baseline models, on a newly constructed lexical substitution dataset for Croatian.

2. PLF

- **Idea**: The PLF model represents predicates as (1) one matrix for each argument slot plus (2) one vector for its overall lexical meaning.
- **Advantages**:
  - Efficient model estimation, simple composition (matrix multiplication, vector addition).
  - Recursive composition applied on longer phrases:
    \[
    P(\text{big window}) = \underbrace{\text{big}} + \underbrace{\text{big} \times \text{window}}
    \]
  - Training the model: Ridge regression with corpus-extracted vectors for arguments as input and vectors for bigram phrases as output:
    \[
    \langle \hat{d}, \hat{v} \rangle = \arg \min \sum_{n \in \text{corpus}} \left\| M \times \tilde{n} - \tilde{d} \right\|^2
    \]
- **PLF variants**: Two variants proposed by Gupta et al. (2015) alter (1) the way matrices are trained ("PLF-train") and (2) used in the computing the phrase vectors in testing phase ("PLF-test").

3. PLF for Croatian

- **Corpus**: hrWaC (Ljubešić and Erjavec, 2011)
- **Versions**: Two bigram extraction (BE) methods for extracting predicate-argument pairs from text:
  - **dependency-based**: pairs adjacent in a dependency tree
  - **surface-based**: pairs adjacent at the surface
- **Advantages**:
  - Idea: The PLF model represents predicates as (1) one matrix for each argument slot plus (2) one vector for its overall lexical meaning.
  - Recursive composition applied on longer phrases:
    \[
    P(\text{big window}) = \underbrace{\text{big}} + \underbrace{\text{big} \times \text{window}}
    \]
  - Training the model: Ridge regression with corpus-extracted vectors for arguments as input and vectors for bigram phrases as output:
    \[
    \langle \hat{d}, \hat{v} \rangle = \arg \min \sum_{n \in \text{corpus}} \left\| M \times \tilde{n} - \tilde{d} \right\|^2
    \]
  - **PLF variants**: Two variants proposed by Gupta et al. (2015) alter (1) the way matrices are trained ("PLF-train") and (2) used in the computing the phrase vectors in testing phase ("PLF-test").

4. Novel Evaluation

- **Motivation**: Semantic similarity (as used so far) is not a reasonable evaluation criteria for cases in which one or both of two phrases are ungrammatical or nonsensical.
- **Setup**: Word-choice tasks in a lexical substitution evaluation setup (see Table 1), composed of ANVAN (adjective-noun-verb-adjective-noun) phrase, a position in the phrase (A1, N1, V, A2, or N2), a correct substitute and three randomly chosen distractors.
- **Prediction**: For each word choice item, compute original phrase vector and 4 substitute phrase vectors.
- **Metric**: Count the number of items where the correct substitute phrase vector is most similar to the original phrase vector.
- **Benefit**: Enables a detailed analysis of model performance at each word in the phrase.

5. Dataset

- **Construction**: We chose 6 highly polysemous verbs and selected 3 subjects and 3 objects that often appear with each of them (using the distributional memory for Croatian). Next, for each subject and object we chose a single adjective that appears often with them.
- **Size**: Total of 18 plausible ANVAN phrases.
- **Annotation**: Three annotators proposed up to three substitutes for each word in a phrase, while ensuring that the grammaticality and meaning of the original phrase remains preserved.

6. Results

| Model          | BE | A1 | N1 | V  | A2 | N2 | Overall |
|---------------|----|----|----|----|----|----|---------|
| add           | 73.4 | 92.0 | 44.6 | 70.1 | 89.7 | 74.0 |
| mut           | 39.2 | 61.4 | 32.5 | 40.2 | 62.8 | 47.4 |
| PLF           | 74.0 | 85.2 | 66.3 | 67.5 | 85.9 | 76.0 |
| PLF-train     | 58.2 | 89.8 | 49.4 | 51.9 | 83.3 | 66.9 |
| PLF-test      | 72.2 | 85.2 | 60.2 | 67.5 | 84.6 | 74.0 |
| PLF           | 55.7 | 87.5 | 63.9 | 65.4 | 84.6 | 71.7 |
| PLF-train     | 54.4 | 89.8 | 51.8 | 56.4 | 82.1 | 67.2 |
| PLF-test      | 69.6 | 87.5 | 55.4 | 60.3 | 83.3 | 71.4 |

- **Overall**: PLF obtained highest accuracy overall and for ‘V’erbs (in line with the results for English). Potential explanation: a verb has the highest valency of all words in a phrase (two arguments).
- **PLF variants**: Do not work for Croatian as they do for English. Possible explanation: noise arising from dependency-based extraction.
- **Bigram extraction (BE) methods**: Surface-based extraction leads to a drop in performance.

7. Conclusion

- **PLF works as well for Croatian as for English**, although its specific strength lies in modeling verbs.
- **Using the dependency parser helps overcome the issue of free word order, but still affects less robust PLF variant (PLF-test).**