# ATILF-LLF v.2: Transition-based verbal multiword expression analyser

Hazem AL SAIED*, Marie CANDITO**, Mathieu CONSTANT*

January 22, 2018

ATILF - Université de Lorraine*, LLF - université Paris-diderot**

# Table of contents

# Introduction

- Transition-based analyzer for identifying and categorizing VMWEs.
- Extension of the ATILF-LLF 1 system [Al Saied et al., 2017].
- Robust, multi-lingual, data-driven system, with limited language-specific tuning.
- Some cases of embedded and non-contiguous VMWEs.
- Developed and evaluated using PARSEME shared task datasets [Savary et al., 2017].

# Datasets

Include 18 languages, and consist of tokenized sentences in which VMWEs are annotated.

## Accompanying resources

- 4 languages have none (BG, ES, HE, LT)
- 4 languages have morphological information (CS, MT, RO, SL)
- 10 languages have full dependency parses (DE, EL, FR, HU, IT, PL, PT, SL, SV, TR)

# Datasets

## VMWE instance could be:

- Set of several tokens, potentially non-contiguous.
- Embedded in another longer one.
- Overlaps with another one.
- Multi-word token (MWT).

## VMWE categories

1. Light Verb Constructions (LVC);
2. IDioms (ID);
3. Inherently REFLexive Verbs (IReflV);
4. Verb-Particle Constructions (VPC);
5. OTHer verbal MWEs (OTH).

# System description

# Transition-based systems

- a configuration in our system consists of a triplet $c = (\sigma, \beta, L)$:
  - $\sigma$ : Stack containing units under processing.
  - $\beta$ : Buffer containing the remaining input tokens.
  - $A$ : Set of output VMWEs.

| | |
|---|---|
| Initial | $c_s = (S = [], B = [x_1, , x_n], A = \{\})$ |
| Intermediate | $C_i = (S = [s_m, .., s_0], B = [b_0, , b_n], A = \{e_1, .., e_k\})$ |
| Terminal | $C_t = (S = [], B = [], A = \{e_1, .., e_m\})$ |

**Figure 1:** The possible types of configurations.

# Transition set

- Transitions predicted by a classifier given the current *configuration*.

| | |
|---|---|
| SHIFT | $(S, x\|B, A) \Rightarrow (S\|x, B, A)$ |
| REDUCE | $(S\|x, B, A) \Rightarrow (S, B, A)$ |
| WHITE MERGE | $(S\|x, y, B, A) \Rightarrow (S\|(x, y), B, A)$ |
| MERGE AS C | $(S\|x, y, B, A) \Rightarrow (S\|(x, y), B, A \cup (x, y)_C)$ |
| MARK AS C | $(S\|x, B, A) \Rightarrow (S\|(x), B, A \cup (x)_C)$ |

**Figure 2:** The transitions used in our system.

- Applies a sequence of *transitions* to incrementally build the output structure in a bottom-up manner.

# Example

| Transition | | Configuration |
|---|---|---|
| | | [ ], [Damit, müsste, ..], [ ] |
| Shift | $\Rightarrow$ | [Damit], [müsste, man, ..], [ ] |
| Reduce | $\Rightarrow$ | [ ], [müsste, man, ..], [ ] |
| ... | | |
| Shift | $\Rightarrow$ | [sich], [nun, herumschlagen], [ ] |
| Shift | $\Rightarrow$ | [sich, nun], [herumschlagen], [ ] |
| Reduce | $\Rightarrow$ | [sich], [herumschlagen], [ ] |
| Shift | $\Rightarrow$ | [sich, herumschlagen], [ ], [ ] |
| Mark as VPC | $\Rightarrow$ | [sich, herumschlagen$_{VPC}$], [ ], [herumschlagen$_{VPC}$] |
| Merge as IreflV | $\Rightarrow$ | [(sich, herumschlagen$_{VPC}$)$_{IReflV}$], [ ], [herumschlagen$_{VPC}$, sich, herumschlagen$_{VPC}$)$_{IReflV}$] |
| Reduce | $\Rightarrow$ | [ ], [ ], [herumschlagen$_{VPC}$, (sich, herumschlagen$_{VPC}$)$_{IReflV}$] |

**Figure 3:** Transition sequence for tagging the German sentence DAMIT MÜSSTE MAN **SICH** NUN **HERUMSCHLAGEN**, *(One would have to struggle with that)*, containing **two VMWEs**: (1) IReflV: **sich herumschlagen**; (2) VPC: **herumschlagen**.

# Training oracle and parsing algorithm I

- Sentence $\Rightarrow$ [configuration, optimum transition] pairs.

## Training time

- **Optimum trans**: legal + compatible with golden annotations.
- **Compatibility**: greedy filtering algorithm.

## Analysis time

- **Optimum trans**: predicted by the trained classifier.
- Predicted transition not legal => optimum = first legal transition.

# Training oracle and parsing algorithm II

| training data production | Analysis |
| --- | --- |
| $c \leftarrow c_s$;<br>**while** $c \notin C_t$ **do**<br>    $t \leftarrow O(c)$;<br>    $c \leftarrow t(c)$;<br>**end** | $c \leftarrow c_s$;<br>**while** $c \notin C_t$ **do**<br>    $t \leftarrow$<br>      $ArgMax_{t \in L(c)} CLF(c, t)$;<br>    $c \leftarrow t(c)$;<br>**end** |

Legality:

- SHIFT: iff $|B| \neq 0$.
- REDUCE: iff $|S| \neq 0$.
- MARK AS: iff $|S| \neq 0$ and $s_0$ is token.
- WHITE MERGE, MERGE AS: iff $|S| \geq 0$.

Priority order: MARK AS, MERGE AS, WHITE MERGE, REDUCE, SHIFT.

# Expressive power

- ATILF-LLF 2 vs ATILF-LLF 1:
  - Categorization.
  - Some cases of embedded VMWEs.
  - Both cannot analyze interleaving VMWEs.
- ATILF-LLF 1's transitions:
  - SHIFT.
  - WHITE MERGE.
  - MERGE AS C+REDUCE.
  - MARK AS C+REDUCE: hard-coded procedures.

# Experimental setup

# Feature groups I - Basic Linguistic Features

- **Focused elements:** $S_1$, $S_0$, $B_0$ and sometimes $B_1$.
- **Bi-grams:** $S_1S_0$, $S_0B_0$, $S_1B_0$, and sometimes $S_0B_1$, $S_0B_2$.
  - For a bi-gram $XY$: $X_wY_w$, $X_pY_p$, $X_lY_l$, $X_pY_l$ and $X_lY_p$
- **Trigrams:** $S_1S_0B_0$
  - For a trigram $XYZ$: $X_wY_wZ_w$, $X_lY_lZ_l$, $X_pY_pZ_p$, $X_lY_pZ_p$, $X_pY_lZ_p$, $X_pY_pZ_l$, $X_lY_lZ_p$, $X_lY_pZ_l$, $X_pY_lZ_l$
- Languages without morphological information
  - using the last two and last three letters as suffixes.

# Feature groups II - Syntax-based Features

- $B_i$ having syntactic dependency L on $S_0$.
  - RightDep($S_0$,$B_i$)=True
  - RightDepLab($S_0$,$B_i$)=L
- $B_i$ is $S_0$'s syntactic governor with label L:
  - LeftDep($S_0$,$B_i$)=True
  - LeftDepLab($S_0$,$B_i$)=L
- There is a syntactic relation *l* between $S_0$,$S_1$
  - syntacticRelation($S_0$,$S_1$) = $\pm$ L

# Feature groups III - Misc. Features

- History-based features
  - Represents the sequence of previous transitions
- Distance-based features
  - Represents the distance between $S_0$ and $S_1$ and between $S_0$ and $B_0$
- dictionary-based features
  - $S_0$ belongs to the **MWT dictionary**
  - $S_0$, $S_1$, $B_0$, $B_1$ or $B_2$ belong to an entry of **VMWE dictionary**
- Stack-length-based features

# Results

# Identification results

- Heterogeneous results across languages:
  - Size of corpora.
  - Availability and the quality of annotations.
  - Most common VMWE categories in train and test sets.

- **Positive correlation**: the F-score and the training set size.
- **Linear negative correlation**: VMWE-based F-score and the proportion of unknown VMWE occurrences in test sets.

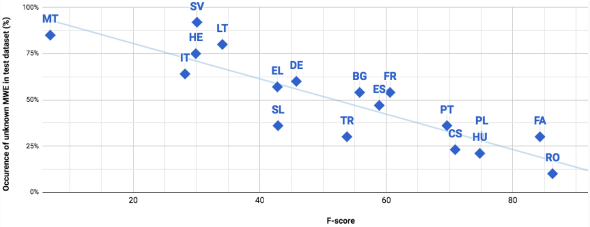


**Figure 4:** VMWE-based F-score and the proportion of unknown VMWE occurrences in test sets.

# French?

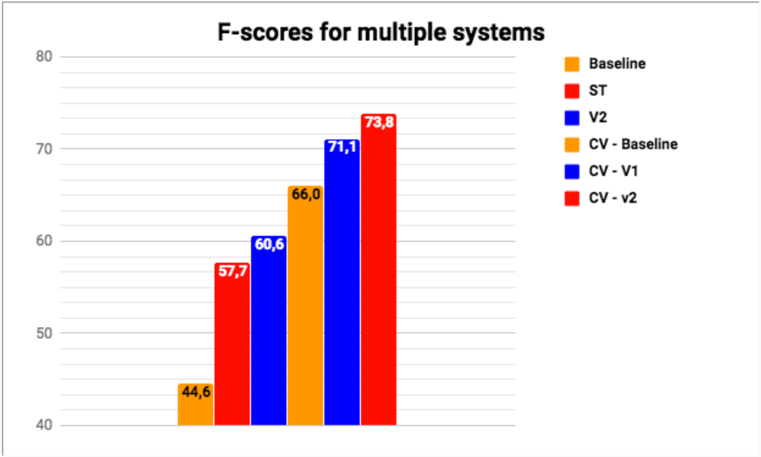


Figure 5: VMWE-based F-scores for multiple experiments on French.

# Identification results - ATILF-LLF 2 vs ATILF-LLF 1

- ATILF-LLF 1 reached best scores for all languages (except HU and RO).
- Test sets: 56.5 vs 56.7.
- Cross-validation:
    - ATILF-LLF 2 beats ATILF-LLF 1 (10/18 languages).
    - Average gain: 4.2-point.
- Good results?
    - Categorization => more transitions.
    - Extended expressive power.
    - Elegant architecture .

# Categorization results

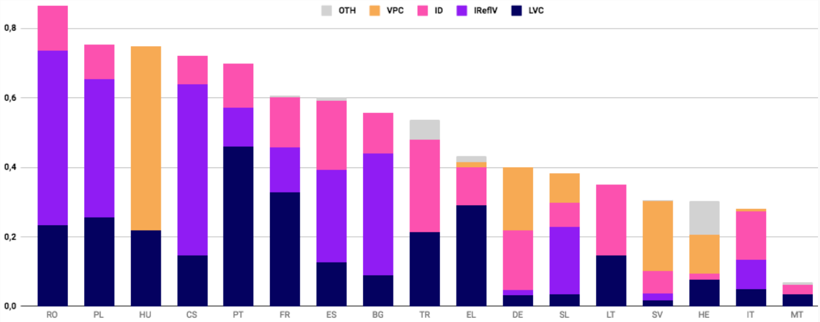


**Figure 6:** languages according to their F-scores on test set.

- ATILF-LLF 2 reaches high performance on categorization too
- Performance varies greatly across categories.
- General trend: higher performance for IReflV, then LVC, then ID

# Conclusion

# Conclusion

- Simple transition-based system.
- Very competitive scores.
- Quite robust across languages.
- Linear time complexity.
- Capable of handling discontinuity and embedding.

- Apply more sophisticated features!
- design deep models!

# Acknowledgement

# Acknowledgement

- This work was partially funded by the French National Research Agency (PARSEME-FR ANR-14-CERA-0001).

# Source code

Get the source of this tagger from

`github.com/hazemalsaied/IdenSys`

The theme *itself* is licensed under a MIT License.

Questions?

# References

Al Saied, H., Candito, M., and Constant, M. (2017).
**The ATILF-LLF system for parseme shared task: a transition-based verbal multiword expression tagger.**
In *Proceedings of the 13th Workshop on Multiword Expressions (MWE 2017)*, pages 127–132, Valencia, Spain. Association for Computational Linguistics.

Savary, A., Ramisch, C., Cordeiro, S., Sangati, F., Vincze, V., QasemiZadeh, B., Candito, M., Cap, F., Giouli, V., Stoyanova, I., and Doucet, A. (2017).
**The PARSEME Shared Task on Automatic Identification of Verbal Multi-word Expressions.**
In *Proceedings of the 13th Workshop on Multiword Expressions (MWE 2017)*, Valencia, Spain.