Multiword expression identification: how far have we got?

CENTAL, Louvain-la-Neuve, Oct 19 2018

Carlos Ramisch
Aix Marseille University, LIS, France

Carlos.ramisch@lis-lab.fr
First things first  'important matters must be treated first'

- Slides adapted from ESSLLI 2018 slides created in collaboration with:
  ○ Agata Savary
  ○ Aline Villavicencio

- Includes material created in collaboration by:
  ○ Silvio Cordeiro
  ○ Manon Scholivet
  ○ Nicolas Zampieri
  ○ Veronika Vincze
  ○ Benoit Favre
  ○ The PARSEME community
This presentation **in a nutshell** 'summarised'

1. **Starting the ball rolling** with multiword expressions
2. **Putting** our **finger** on MWE token identification
3. The PARSEME shared task **at a glance**
4. **Getting** our **hands dirty** with MWE identification models
5. A **pain in the neck** or a **piece of cake**?
1

Starting the ball rolling with multiword expressions
What's in a multiword? Definitions 1/3

- idiosyncratic interpretations that cross word boundaries (or spaces) (Sag et al., 2002)
  - throw beans vs. spill lentils vs. spill beans
- sequence of words that acts as a single unit at some level of linguistic analysis (Calzolari et al., 2002)
- no unified phenomenon to describe but rather a complex of features that interact in various, often untidy, ways and represent a broad continuum between non-compositional (or idiomatic) and compositional groups of words (Moon, 1998)
  - kick the bucket vs. police car
Any linguistic expression involving more than one word that requires an interpreter – human or machine – to have more than the abilities of an Innocent Speaker-Hearer (ISH) who has knowledge only about unitary words and word-to-word relations (Fillmore, 2003)

- deep learning, make ends meet, copy of

Distinction between what a speaker can compute automatically from language and what he must explicitly store (Fillmore, Kay and Connor, 1988)

- ideally most of the knowledge about how to use a language should be computable
  - tight clothes vs. tight lips 'reticent, taciturn person; close-mouthed'
What's in a multiword? Definitions 3/3

- Any kind of linguistic unit that has been considered *formulaic* in any research field (Wray, 1999)
  - object oriented programming, relational databases, artificial intelligence
- Habitual *recurrent word combinations* of everyday language (Firth, 1957)
  - salt and pepper, freak out
My favourite definition

- Multiword expressions are lexical items that:
  - (a) can be decomposed into multiple lexemes; and
  - (b) display lexical, [morphological], syntactic, semantic, pragmatic and/or statistical idiomaticity

(Baldwin and Kim, 2010)

- Idiomaticity: deviation from usual composition rules

- Exceptions that occur when words come together
Multiword expressions are composed of multiple *words*

What counts as a word?
- tricky question with no consensual answer

Multiword expressions are composed of multiple *lexemes*

**Lexemes**: lexical items (or units) or elementary units of meaning that represent basic blocks of a language’s lexicon
- Mostly equivalent to "words"
- Affixes (-*ing*) and standalone morphemes (‘*s*) are not lexemes
Words and tokens: a rat's nest 'mess' 2/2

- **Tokens** are the result of a computational process of tokenization, that is, splitting the text into minimal units for further processing (e.g. parsing).

- **Lexemes** are a linguistic notion, **tokens** are a computational notion.

- Ideally, lexemes = tokens, but:
  - Compounds: *whitespace*
  - Contractions: *don't*
  - Orthography conventions: *pre-existing, part-of-speech tag*
  - Languages use spaces differently (or not at all!): 五岳之西岳华山

- **Multiword tokens** can be MWEs (*whitespace*)
- **Multi-token words** (*John's*) cannot be MWEs
Lexicalized components and open slots

- Some elements are mandatory, otherwise the MWE cannot occur
  - throw the beans, spill the lentils, spill royal beans, spill the beans
- We will call them *lexicalized components*
- They will appear in **bold** in our examples

- Some components can be omitted or replaced in MWEs
  - take a/some/an important **decision**
- It may be useful to mention them to provide disambiguating context
- We will call them **open slots**
MWEs come in all shapes and sizes 1/2
'MWEs are diverse'

- Nominal compounds
  - busy bee, whitespace, machine translation
- Nominal idioms
  - piece of cake, pain in the neck
- Verbal idioms
  - to do the trick, to make sense, to give rise, to take into account
- Light-verb constructions
  - to make a decision
MWEs come in all shapes and sizes 2/2

'MWEs are diverse'

- Verb-particle constructions
  - to break up, to figure out
- Multiword adverbials
  - more often than not, in turn, on the one/other hand, not only [...] but also
- Multiword prepositions and conjunctions
  - in order to, as well as, when it comes to, such as, so much that
- Multiword terms
  - lexical unit, natural language processing, multiword expression
- ...
MWEs vs. collocations

Collocations - many **points of view** on the phenomenon:

- *(Sag et al., 2002): statistically significant word co-occurrences*

- *(Baldwin & Kim, 2009): statistically idiomatic MWEs*

- *(Mel’čuk, 2010): binary, semantically compositional combinations of words subject to **lexical selection** constraints*

- *(Savary et al., 2018): word co-occurrences whose idiosyncrasy is of **pragmatic** or **statistical** nature only*
2

Putting our finger on MWE token identification
Computational processing of MWEs involves different tasks
- Finding new MWEs in raw text for inclusion in lexicons ←most popular
- Identifying MWEs in running text
- Dealing with MWEs in a parser
- Translating MWEs automatically in machine translation
- ...

Names matter:
- Discovery, identification, extraction, learning, recognition, detection, dictionary induction,...
MWE processing 2/2

- "MWE processing is composed of two main subtasks that are often confused in the literature: MWE discovery and MWE identification"
MWE discovery

- "MWE discovery* is concerned with finding new MWEs (types) in text corpora, and storing them for future use in a repository of some kind such as a lexicon" (Constant et al 2017)
  - **INPUT**: a text
  - **OUTPUT**: a repository (list, lexicon) of MWEs

* We sometimes use the term MWE type discovery to emphasize that it is performed out of context.
MWE identification

- "MWE identification* is the process of automatically annotating MWEs (tokens) in running text" (Constant et al 2017)
  - **INPUT**: a text
  - **OUTPUT**: a text annotated with MWEs

* We sometimes use the term MWE token identification to emphasize that it is performed in context.
MWEs processing interactions

- NLP tasks and applications require MWE processing
  - MWE identification ↔ parsing
  - MWE discovery ↔ machine translation
  - ...
- And vice-versa!
Practice: give it a try

⇒ Identify the MWEs in the following excerpt:

Automatically breaking a sentence up into minimal lexical units may look like a piece of cake, especially in languages like English where whitespace is used to delimit tokens. More often than not, however, it is not so straightforward to figure out how to make segmentation decisions, in order to split sentences into lexical units that make sense. While, on the one hand, good tokenization rules do the trick for simple words, on the other hand, lexico-semantic segmentation is a pain in the neck when it comes to lexical units composed of more than one lexeme.

Adapted from Ramisch and Villavicencio (2018)
Practice: give it a try - correction

⇒ Identify the MWEs in the following excerpt:

Automatically breaking a sentence up into minimal lexical units may look like a piece of cake, especially in languages like English where whitespace is used to delimit tokens. More often than not, however, it is not so straightforward to figure out how to make segmentation decisions, in order to split sentences into lexical units that make sense. While, on the one hand, good tokenization rules do the trick for simple words, on the other hand, lexico-semantic segmentation is a pain in the neck when it comes to lexical units composed of more than one lexeme.

Adapted from Ramisch and Villavicencio (2018)
Why is it useful in NLP?

- Treat MWEs as a whole at some level of processing
- Prevent errors in NLP tasks related to word-by-word modelling
- Guide downstream tasks and applications:
  - Tokenisation: whitespace
  - Syntactic parsing: by and large 'mainly'
  - Word sense disambiguation: dry run 'rehearsal'
  - Semantic parsing/analysis: to make a presentation
  - ...

...
Why is it useful in the real world?

Identify an MWE before machine translation can avoid catastrophic translations
Resources: what is needed?

● Lexicons
  ○ Expert knowledge
  ○ Corpora (MWE discovery)

● (Learned) models and/or rules
  ○ Expert knowledge
  ○ Annotated corpora
In short, it all boils down to...
Challenges for MWE identification

1. Statistical idiosyncrasy
2. Discontinuity
3. Ambiguity and coincidental cooccurrence
4. Variability (flexibility)
5. Overlaps
Challenges: a pain in the neck 'hard problem' 1/2

1. Statistical idiosyncrasy
   ○ We gave it up
   ○ We gave it back → frequent but not idiosyncratic
   ○ It fulfills the requirements

2. Discontinuity
   ○ breaking a sentence up
   ○ make segmentation decisions

3. Ambiguity and coincidental cooccurrence
   ○ MWE identification is a piece of cake
   ○ I never eat more than one piece of cake → ambiguity
   ○ he was making cookies when the decision was announced → coincidental cooc.  

(Constant et al 2017, Savary and Cordeiro 2017)
Challenges: a pain in the neck 'hard problem' 2/2

4. Variability (flexibility) ⇒ dictionary forms and corpus forms do not match
   ○ put/puts/putting a/his/her/my/our finger on
   ○ decisions which we made
   ○ crocodile tear (?) → crocodile tears

5. Overlaps: factorisation and nesting
   ○ Should I take a shower or a bath? → factorisation
   ○ We must take the dry run into account → nesting
Orchestration: before, during or after?

- Identification is typically seen as a preprocessing step
- Other scheduling choices are possible
- **Orchestration** refers to the decision of performing MWE identification:
  - before,
  - during, or
  - after

another NLP task or application such as machine translation or parsing

(Constant et al 2017)
Orchestration: identification and parsing

- Before:
  - Text: abc a bbb
  - MWE identification
  - Parser

- During:
  - Parser
  - MWE identification
  - MWE-annotated text + structure

- After:
  - Text: abc a bbb
  - Parser
  - Text + Structure
  - MWE identification

○ We will consider identification before/after another process, but not during
Evaluation: **checking results out** 1/3

- Quantify the overlap between:
  - test corpus annotated by the system: **predictions**
  - reference annotations provided by humans: **gold standard**
Evaluation: checking results out 2/3

- Precision: TP / P
  - proportion of correctly identified MWEs wrt all predicted MWEs
- Recall: TP / T
  - proportion of correctly identified MWEs wrt all gold MWEs
- F-measure: F = 2PR / (P + R)
  - harmonic average between precision and recall
How do we compare identified MWEs to decide what counts as a TP?

- **Exact match (MWE-based):** full MWEs
- **Link-based:** pairs of consecutively identified tokens
  - DiMSUM shared task
  - No single-token MWE
- **Token-based:** tokens in the best bijection between prediction and gold
  - PARSEME shared tasks

- **Phenomenon-specific:** same as above, but focused on some type of MWE:
  - Discontinuous vs. continuous
  - Single-token vs. Multi-token
  - Variant vs. identical
  - Unseen vs. seen

Evaluation: **checking results out** 3/3
Evaluation examples

- **Gold:** \[t_1 \ t_2 \ t_3\]
- **System1:** \[t_1 \ t_2 \ t_3\]
  - MWE-based: Precision = 0, Recall = 0
  - Token-based: Precision = \(\frac{2}{3}\), Recall = \(\frac{2}{3}\)
- **System2:** \[t_1 \ t_2 \ t_3\]
  - MWE-based: Precision = \(\frac{1}{3}\), Recall = \(\frac{1}{2}\)
  - Token-based: Precision = \(\frac{2}{3}\), Recall = \(\frac{2}{3}\)

(Savary et al 2017)
Exercise: evaluation

Gold:  *make* segmentation *decisions*, *in order to* split sentences into *lexical units*

System: *make* segmentation *decisions*, *in order to* *split sentences* into lexical *units*

- MWE-based (exact) precision and recall?
- Token-based precision and recall?
Exercise: evaluation - correction

Gold: **make** segmentation **decisions**, *in order to* split sentences into **lexical units**

System: **make** segmentation **decisions**, *in order* to **split sentences** into lexical **units**

- MWE-based (exact) precision and recall
  \[ TP = 1, \quad P = \frac{1}{4}, \quad R = \frac{1}{3}, \quad F = \frac{2}{7} \approx 0.28 \]

- Token-based precision and recall
  \[ TP = 5, \quad P = \frac{5}{7}, \quad R = \frac{5}{7}, \quad F = \frac{5}{7} \approx 0.71 \]
3

The PARSEME shared task
at a glance
The **shared tasks** 'evaluation campaign'

- **DiMSUM (2016)**
  - English corpus, 3 domains, annotated for strong/weak MWEs and supersenses
  - 9 submissions, best system had 56.67 link-based F1 (MWE identification only)

- **PARSEME (2017 and 2018)**
  - Corpora in 18 languages (2017) and 20 languages (2018)
  - Centralised annotation guidelines and interface
  - Focus on **verbal MWEs**
PARSEME shared task 1.0 (2017)

- Multilingual guidelines with examples
- Annotation methodology and teams (PARSEME)
- Corpora in 18 languages under free licenses
- Train/test corpora with 52,724/9,494 VMWEs
- New evaluation measures (MWE-/Token-based)
- 7 participating systems

(Savary et al 2017, Ramisch et al 2018)
Guidelines 1.1 (2018)

- [http://parsemefr.lif.univ-mrs.fr/parseme-st-guidelines/1.1](http://parsemefr.lif.univ-mrs.fr/parseme-st-guidelines/1.1)
- Initially written by experts
- Improved/amended by the community via Gitlab issues
- Based on decision trees and linguistic tests
- Flexibility as a proxy for semantic compositionality
Apply test VID.1 (prev. 1) - [CRAN: Candidate contains cranberry word?]

- **YES** ⇒ It is a VID, exit.
- **NO** ⇒ Apply test VID.2 (prev. 2) - [LEX: Regular replacement of a component ⇒ unexpected meaning shift?]
  - **YES** ⇒ It is a VID, exit.
  - **NO** ⇒ Apply test VID.3 (prev. 3) - [MORPH: Regular morphological change ⇒ unexpected meaning shift?]
    - **YES** ⇒ It is a VID, exit.
    - **NO** ⇒ Apply test VID.4 (prev. 4) - [MORPHSYNT: Regular morphosyntactic change ⇒ unexpected meaning shift?]
      - **YES** ⇒ It is a VID, exit.
      - **NO** ⇒ Apply test VID.5 (prev. 5) - [SYNT: Regular syntactic change ⇒ unexpected meaning shift?]
        - **YES** ⇒ It is a VID, exit.
        - **NO** ⇒ It is not a VID, exit

**Test VID.1 (prev. 1) - [CRAN] - Cranberry word**

Does the candidate expression contain a cranberry word?

- **YES** ⇒ it is a VID
  - *(EN)* to go *astray* → astray is not a stand-alone word
  - *(FR)* prendre la poudre d'*escampette* → escampette is not a stand-alone word
- **NO** ⇒ further tests are required
  - *(EN)* to go *away* → go and away are stand-alone words
VMWE categories

- **Universal categories**
  - Verbal idioms (VID)
    - to call it a day 'to stop working'
  - Light-veb constructions (LVC)
    - to give a lecture (LVC.full)
    - to grant rights (LVC.cause)

- **Quasi-universal categories**
  - Inherently reflexive verbs (IRV)
    - to help oneself 'to take something freely'
  - Verb-particle constructions (VPC)
    - to do in 'to kill' (VPC.full)
    - to eat up (VPC.semi)
  - Multi-verb constructions (MVC)
    - to make do 'to suffice'

- **Optional categories (LS.ICV, IAV)**
A whole bunch of languages

- **Balto-slavic**: Bulgarian (BG), Croatian (HR), Lithuanian (LT), Polish (PL), Slovene (SL), Czech (CZ)
- **Germanic**: German (DE), English (EN), Swedish (SV)
- **Romance**: French (FR), Italian (IT), Romanian (RO), Spanish (ES), Brazilian Portuguese (PT)
- **Others**: Arabic (AR), Greek (EL), Basque (EL), Farsi (FA), Hebrew (HE), Hindi (HI), Hungarian (HU), Turkish (TR), Maltese (MT)
Corpora

- **Training**
  - 4,5M tokens, 59460 VMWEs

- **Development**
  - 672K tokens, 9250 VMWEs (except EN, HI, LT)

- **Test**
  - 847K tokens, 10616 VMWEs (at least 500 VMWEs per language)

- Including morphological/syntactic information (mostly UD)
- Macro-averages IAA on a sample: 0.691 for span, 0.836 for category
- CUPT format, in collaboration with UD
- Released under creative commons licenses on LINDAT
  - [http://hdl.handle.net/11372/LRT-2842](http://hdl.handle.net/11372/LRT-2842)
The shared task

- Two tracks
  - Closed track: only provided training/dev data
  - Open track: provided data + any external resource

- Evaluation
  - Identification only
  - Token-based and MWE-based P/R/F1
  - Phenomenon-specific scores
    - Continuous vs discontinuous
    - Multi-token vs single-token
    - Seen vs unseen (wrt training corpus)
    - Identical vs variants (wrt training corpus)
Systems

- 12 teams, 17 system submissions
  - 13 closed track + 4 open track
  - 16/17 submissions cover 3 or more languages
  - 11/17 submissions cover 19 languages

- Techniques:
  - Neural networks
  - Parsing
  - CRF
  - Association measures
  - NaiveBayes classifier
Some results

Average MWE-based scores

<table>
<thead>
<tr>
<th>submission</th>
<th>track</th>
<th>P</th>
<th>R</th>
<th>F1</th>
</tr>
</thead>
<tbody>
<tr>
<td>TRAVERSAL</td>
<td>closed</td>
<td>67.58</td>
<td>44.97</td>
<td>54.00</td>
</tr>
<tr>
<td>TRAPACC-S</td>
<td>closed</td>
<td>62.28</td>
<td>41.40</td>
<td>49.74</td>
</tr>
<tr>
<td>TRAPACC</td>
<td>closed</td>
<td>55.68</td>
<td>44.67</td>
<td>49.57</td>
</tr>
<tr>
<td>CRF-Seq-nocategs</td>
<td>closed</td>
<td>56.13</td>
<td>39.12</td>
<td>46.11</td>
</tr>
<tr>
<td>SHOMA</td>
<td>open</td>
<td>66.08</td>
<td>51.82</td>
<td>58.09</td>
</tr>
</tbody>
</table>

More details: http://multiword.sourceforge.net/sharedtaskresults2018
So, what next?

- Shared task 2.0 in 2020
- More languages, better quality
- Cover nominal MWEs
- Account for lexical nature of MWEs (unseen)
- Synergies with other initiatives (e.g. UD)
4

Getting our hands dirty with MWE identification models
How can we do it?

- **Case 1:** We **do** have a lexicon of known MWEs to identify:
  - Lexicon matching/projection methods based on rules, transducers,…

- **Case 2:** We **do not** have a lexicon of known MWEs to identify, but:
  - We have a raw corpus and we know the target MWEs' characteristics:
    - Perform MWE discovery to obtain a lexicon, then go to Case 1
  - We have an annotated training corpus:
    - Extract a lexicon from the corpus, then go to Case 1 (only for known MWEs)
    - Train a machine learning model using
      - Local classifiers
      - Sequence models
      - Parsers
How can we do it?

- **Case 1**: We **do** have a lexicon of known MWEs to identify:
  - Lexicon matching/projection methods based on rules, transducers,...

- **Case 2**: We **do not** have a lexicon of known MWEs to identify, but:
  - We have a raw corpus and we know the target MWEs' characteristics:
    - Perform MWE discovery to obtain a lexicon, then go to Case 1
  - We have an annotated training corpus:
    - **Extract a lexicon from the corpus, then go to Case 1 (only for known MWEs)**
    - Train a machine learning model using
      - Local classifiers
      - **Sequence models**
      - Parsers

- Exercises with the mwetoolkit: [http://mwetoolkit.sf.net](http://mwetoolkit.sf.net)
The lexicon

- **Simple**: a list of more fixed MWEs to identify (compounds, adverbials,...)
- Due to inflection, surface forms may not suffice
  - MWE components' lemmas and POS tags (more on day 4)
    - `dead/ADJ end/NOUN`
    - `crocodile/NOUN tears/NOUN`
- **Structured**: lexicalised components and open slots (gaps)
- For more flexible MWEs, additional constraints may be needed:
  - Allowed/forbidden syntactic relations between components:
    - `figure/VERB -compound:prt → out/ADP`
    - `take/VERB -obl → account/NOUN -case → into/ADP`
  - Allowed/forbidden modifications (e.g. alternations):
    - Passivisation, pronominalisation of components, etc.
  - Distance between components, POS of intervening words, etc.
Lexicon matching

● Gapping possibilities:
  ○ Contiguous: sequences of words from a list of MWEs
  ○ Gappy: words with up to a limit number of gaps in between

● Match distances:
  ○ Shortest: shortest possible candidate
  ○ Longest: longest possible candidate
  ○ All: all possible candidates, including longest and shortest

● Match modes:
  ○ Non-overlapping: at most one MWE per word in the corpus
  ○ Overlapping: words can be part of more than one MWE
Lexicon matching: example

● Regular expressions:
  ○ Compound nouns: Noun Noun+
  ○ Phrasal verbs: Verb AnyWord{0,5} (Preposition|Adverb)

● Matching heuristics:
The mwetoolkit

- Command-line scripts in Python for text processing
  - Corpus preprocessing
  - MWE discovery
  - MWE identification
    - Using lexicon and rules
    - Using sequence models (CRF)
- [http://multiword.sf.net](http://multiword.sf.net)
How to obtain the lexicon?

- Existing resources
- MWE discovery (e.g. using mwetoolkit)
- Extract from MWE-annotated corpus
Isn't this too simple to work?

- Submission to DiMSUM 2016 shared task
  - Shamelessly simple – ranked 2nd among 9 submissions
- Heuristics must be more sophisticated to deal with variants
- Depends on the quality of the underlying annotation
- Memorisation is not too bad for MWEs
  - What can be really generalised from seen to unseen MWEs?
Machine learning for MWE identification

- We do not have a lexicon of known MWEs to identify, but:
  - We have an annotated training corpus
- Use a machine learning approach:
  - Local classifiers
  - Sequence models
  - Parsers
  - ...
- During training
  - learn a model that represents recurrent occurrence patterns of annotated MWEs
- During test
  - apply the model on unseen text and verify if the model can identify new MWEs
MWE identification as tagging

- Tag each word as belonging to an MWE or not
- Use extended BIO encoding (Schneider et al 2014):
  - B - Token is the Beginning of an MWE
  - I - Token is Inside an MWE
  - O - Token is Outside any MWE
  - G - Token is a Gap in a MWE
- Cannot represent overlaps (nesting and factorisation)
- Example:

```
he  took  my  remarks  into  account  before  the  dry  run
O   B    G    G    I    I    O    O    B    I
```

- Exercise:

```
make  segmentation  decisions  in  oder  to  split  sentences  into  lexical  units
```
MWE identification as tagging - correction

- Tag each word as belonging to an MWE or not
- Use extended BIO encoding (Schneider et al. 2014):
  - B - Token is the Beginning of an MWE
  - I - Token is Inside an MWE
  - O - Token is Outside any MWE
  - G - Token is a Gap in a MWE
- Cannot represent overlaps (nesting and factorisation)
- Example:

<table>
<thead>
<tr>
<th>he</th>
<th>took</th>
<th>my</th>
<th>remarks</th>
<th>into</th>
<th>account</th>
<th>before</th>
<th>the</th>
<th>dry</th>
<th>run</th>
</tr>
</thead>
<tbody>
<tr>
<td>O</td>
<td>B</td>
<td>G</td>
<td>G</td>
<td>I</td>
<td>I</td>
<td>O</td>
<td>O</td>
<td>B</td>
<td>I</td>
</tr>
</tbody>
</table>

- Exercise:
Probabilistic tagging

- Predict the most likely tag $t_i$ for token $w_i$ at each position $i$
- Learn the probability of possible tags ($B$, $I$, $O$, $G$) in conjunction with:
  - The input token(s) and their properties (lemma, POS, syntactic head, etc.)
  - The previously predicted tag(s)
- **Example:** Probability of tag $I$ given that the current word is *into* and the previous tag was $B$?
  - $P(t_i=I,w_i=\text{into},t_{i-1}=B) = ?$

- Probabilities are estimated based on counts in the training corpus
- **Example:** Proportion of times the word *into* was tagged $I$ and the previous tag was $B$:
  - $P(t_i=I,w_i=\text{into},t_{i-1}=B) = nb(t_i=I,w_i=\text{into},t_{i-1}=B) / nb(t_i=*,w_i=*,t_{i-1}=*)$
Conditional random fields (CRFs)

- Generalised variant of hidden Markov models
- Improved emission model:
  - Features to model current tag given not only the current word, but any input word
- Linear-chain first-order CRF:
CRF features for MWE identification

- What kind of input information can help predicting the current tag?
  - Current word's properties:
    - Surface form
    - Lemma
    - POS tag
    - Capitalization
    - Presence of digits
  - Surrounding words' properties (previous/next 1-2 words)
  - Syntactic head's properties

- Example:
  - $P[0] \rightarrow$ POS of current word
  - $L[-1] \rightarrow$ lemma of previous word
  - $L[0] \ L[1] \rightarrow$ bigram composed of lemmas of the current and next words
  - $P[-2] \ P[-1] \ P[0] \rightarrow$ trigram composed of POS of 2 previous words and current one
External resources in sequence models

- It is easy to include new features in sequence models.
- We can model information from an external lexicon:
  - True: if current word belongs to a sequence that is an MWE in the lexicon
  - False: otherwise
- The external lexicon can be handbuilt or automatically extracted from a large unannotated corpus using some discovery method.
- This can greatly increase the recall of MWE identification with CRFs.

(Constant and Sigogne 2011, Constant and Tellier 2012, Schneider et al 2014, Riedl and Biemann 2016)
Recurrence neural networks (LSTM etc.)

- Similar model to CRF: predict the most probable tag given the input, but:
  - Unbounded history
  - Word embeddings (better generalisation)
  - No need for feature engineering (but what about hyper-parameters?)
The Veyn system

(Zampieri et al. 2018, Scholivet et al. submitted)
Veyn at PARSEME shared task 2018

- Ranked 8th/9th on average
- Focus on Token-based F1
- Best in HE and TR, worst in EN and EL
- Variability: number of epochs
- Average tuning or per-language tuning?
- Factorization and nesting are ignored

https://github.com/zamp13/Veyn
Parsing-based MWE identification

- Sequence taggers cannot represent overlap and have trouble with long gaps
- Parsers can handle long gaps and, sometimes, overlaps
- Idea: perform MWE identification jointly with parsing
  - Add MWE tags to parsing constituents/dependencies
  - Predict an additional tree structure for MWEs simultaneously with parsing

Images source: Constant et al 2018
A pain in the neck or a piece of cake?
Where do we stand?

- Shared tasks help making progress but also bias the research
  - Many systems and techniques
  - More time is needed to analyse results
  - Focus on identification rather than discovery
  - Focus on machine learning rather than lexicons
- Do our systems model semantic compositionality?
- Is deep learning required/useful?
  - Lexical nature of the phenomenon
  - MWEs are exceptions
  - Word embeddings
- Full MWE annotation guidelines in several languages are on the TODO list
Keep an ear to the ground 'keep informed'

- MWE community
  - PARSEME - European network on parsing and MWEs
  - MWE section of SIGLEX (special interest group at the ACL) - join both

- MWE events
  - Yearly MWE workshop collocated with major NLP conferences:
    - 14th edition in 2018: joint event with the Linguistic Annotation Workshop community (LAW-MWE-CxG at COLING 2018)
    - In 2019, joint event planned with the WordNet community
  - PARSEME shared task on automatic identification of MWE
    - New edition planned for 2020 (new languages and MWE categories)
  - Yearly EUROPHRAS conferences
  - MUMTTT workshops (on MWEs in MT)

- Book series: Phraseology and Multiword Expressions, at Language Science Press, Berlin:
  - 1 volume out, 3 others in the pipeline
Keep your nose to the wind 'keep informed'

- MWE resources
  - DIMSUM shared task dataset
  - SIGLEX-MWE resource list
  - PARSEME corpus of verbal MWEs (20 languages in 2018) - open-ended project:
    - new language leaders and annotators are welcome
    - New MWE categories (adverbials, nominals, named entities,...) will be addressed
  - PARSEME annotation guidelines
  - PARSEME surveys
    - On MWE annotation in treebanks
    - On lexical resources of MWEs
    - On multilingual MWE resources
Thanks for **bearing with me!**

That was a **hard nut to crack!**

Carlos Ramisch
Aix Marseille University, LIS, France

[carlos.ramisch@lis-lab.fr](mailto:carlos.ramisch@lis-lab.fr)