How can we capture multiword expressions?

[*] Seongmin Mun , [†] Guillaume Desagulier , [‡] Kyungwon Lee

[*] Lifemedia Interdisciplinary Program, Ajou University [*] UMR 7114 MoDyCo - CNRS, University Paris Nanterre

[†] UMR 7114 MoDyCo - University Paris 8, CNRS, University Paris Nanterre

[‡] Department of Digital Media, Ajou University

[*] stat34@ajou.ac.kr, [†] guillaume.desagulier@univ-paris8.fr, [‡] kwlee@ajou.ac.kr

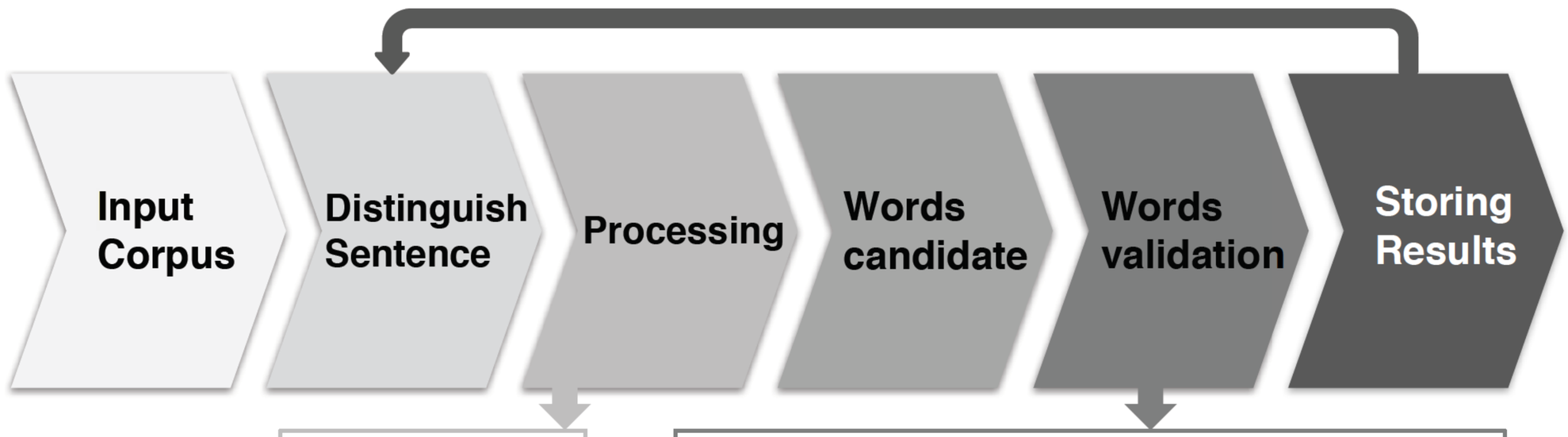

Figure 1: Data processing structure. Framework for topic acquisition from corpus data.

Introduction

Topics in a text corpus include features and information. Analyzing these topics can improve a user's understanding of the corpus. These topics can be divided into two types: those whose meaning can be described in one word and those whose meaning in expressed through a recurring combination of words, also known as multiword expressions (MWE). Out of context, the MWE 'she sets the bar high' is ambiguous between a literal and a metaphorical reading. Ambiguity resolution is needed to extract accurate topics. Several well-known techniques have been proposed for topic extraction: TF*PDF (Khoo Khyou Bun et al., 2002), Topic Detection and Tracking (Kuan-Yu Chen et al., 2007), LDA (T. L. Griffiths and M. Steyvers, 2004), inter alia. However, most of these techniques target single words, not MWEs. In this paper, we propose a system that extracts MWE-based topics accurately. Our algorithm breaks down into six steps: Recognition, Pre-Processing, Processing, Candidate Extraction, Topic Validation, and Storing. We benchmark the Evaluation step using ambiguous sentences. Results show that the algorithm identifies MWEs faster and more accurately. This is because it detects problematic expressions, parses them in the light of a repository of resolved MWEs, and manages to provide a correct interpretation. Compiling a repository of MWEs that are correctly parsed and interpreted is time consuming. We show how this can be solved in the near future.

Case study

We present a data processing architecture for extracting MWEs from corpus (Figure 1). In the processing, candidate words were extracted from a combination of tokenized words with N-grams and reference relations of words with Dependency structure. Dependency is the notion that linguistic units, e.g. words, are connected to each other by directed links (Mel'čuk, Igor A., 2012). Figure 2 illustrates how we can extract multiword candidates with dependency structure from the example sentence : **'Shall I wake him up?'**

```
Result of dependency graph below

dependency graph:
-> wake/VBP (root)
  -> Shall/NNP (nsubj)
    -> I/PRP (dep)
  -> him/PRP (dobj)
  -> up/RP (compound:prt)
  -> ?/. (punct)
```

```
Result of multiword candidates

wake Shall
Shall I
wake Shall I
wake him
wake up
wake ?
```

Figure 2: Words candidates from Dependency structure

Figure 3 shows the results of extracting meaningful words using only N-grams and extracting meaningful words using both dependency tags and N-grams. The result shows that more meaningful words are returned when both methods are used.

```
Final result below

0. wake is meaningful : wake
1. shall is meaningful : shall
2. i is meaningful : i
3. up is meaningful : up
4. shall i is meaningful : shall i
5. him is meaningful : him
```

N-gram

```
Final result below

0. wake is meaningful : wake
1. shall i is meaningful : shall i
2. i is meaningful : i
3. wake up is meaningful : wake up
4. up is meaningful : up
5. him is meaningful : him
6. shall is meaningful : shall
```

N-gram & Dependency parser

Figure 3: Comparing the MWEs to N-gram and Dependency tag

Conclusion

Often, MWEs cause problems; Google translation, Stanford CoreNLP, etc. Those problems can be solved if the algorithm can extract and recognize MWEs correctly. For this reason, we made a parsing algorithm to extract MWEs from the corpus. The case study shows how to extract MWEs. In the near future, we will create a web application based on our algorithm to visualize the results and integrate the user's input on MWEs.