



Automatic Recognition of Noun-Noun-Relations via WordNet

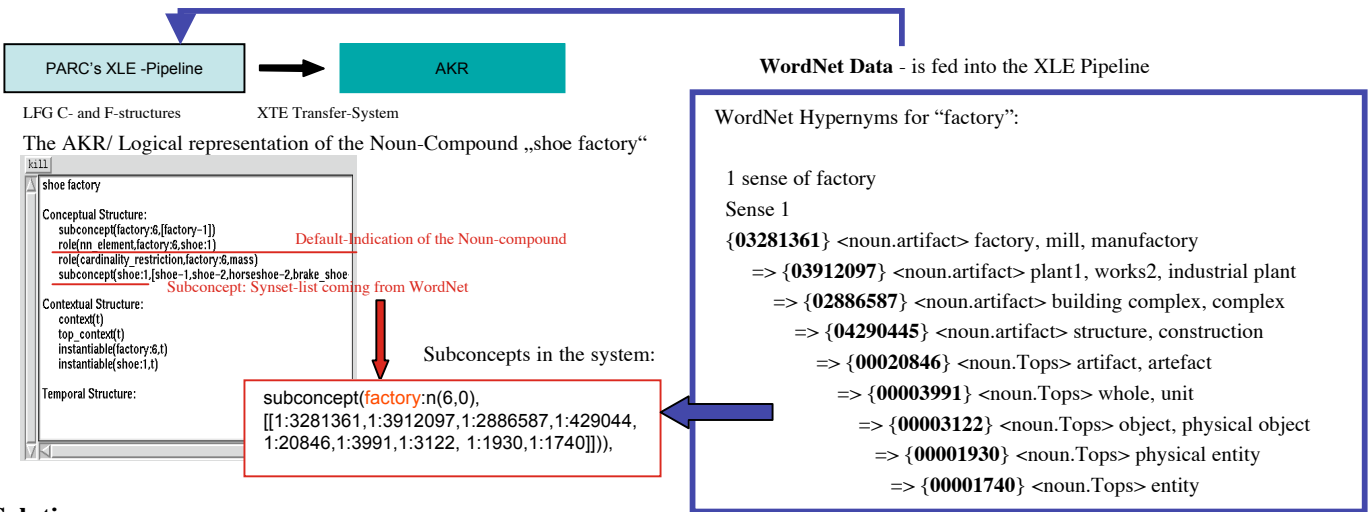
WELL-KNOWN PROBLEM: The single elements of Noun-compounds form relations to each other, which are difficult to predict, e.g.

- Butter knife → knife that is **used to** spread butter
- Steel knife → knife that is **made of** steel
- Pocket knife → knife **for** the pocket

Even though the head noun is the same
a change in the modifying noun
changes the relation within the compound

TASK: Find a way to automatically recognize the relation within compounds so it can be described within PARC's AKR

PARC's XLE (grammar development platform) and AKR (Abstract Knowledge Representation) (http://www.parc.com/ist1/groups/nlxt/xle/xle_toc.html)



Solution:

- 1) Abstract Noun-groups like „artifact“ can be constructed with the help of WordNet Hypernyms. Nouns sharing certain Hypernym number can thus be recognized as a member of an abstract group.

```

| - noun_class(artifact, 00020846).    "shoe", "factory"
| - noun_class(manufactory, 03912097). "factory"
  
```

- 2) By defining the relations between different groups of nouns, the relations within Noun-compounds can be described automatically.

```
@noun_relation(manufactory, artifact, manufacture).
```

- 3) Rule (XTE) that tests all the parameters of the relation and, in case of a positive match, assigns a specific role.

```

noun_relation(%HeadClass, %ModClass, %Role) ::

+nn_element(%Mod,%Head,%%),
+subconcept(%Head, %HeadHypers),
+subconcept(%Mod, %ModHypers),
noun_class(%HeadClass, %HeadNo),
noun_class(%ModClass, %ModNo),
@check_class(%HeadNo, %HeadHypers),
@check_class(%ModNo, %ModHypers)
==>
role(%Role, %Head, %Mod).
  
```

```

check_class(%M, %S) :=
{member(%H, %S), memberchk(%M, %H)}.
  
```

The new AKR now contains
a specific role describing the
relation between the two
compounds more closely.

```

kill
shoe factory
Conceptual Structure:
  subconcept(factory:6,[factory-1])
  role(manufacture,factory:6,shoe:1)
  role(cardinality_restriction,factory:6,shoe:1)
  subconcept(shoe:1,[shoe-1,shoe-2,horseshoe-2,brake_shoe])
Contextual Structure:
  context(t)
  top_context(t)
  instantiable(factory:6,t)
  instantiable(shoe:1,t)
Temporal Structure:
  
```

Summary: --> 124 abstract noun-classes.
--> 115 specific relations to describe the relationship between two nouns.
--> 23 different roles to describe these relations.
--> 1 default-of-role for non-specific relations not covered by the above.

Positive Result: Most of the Noun compounds can be described with this solution,
Extensions: Pertainyms