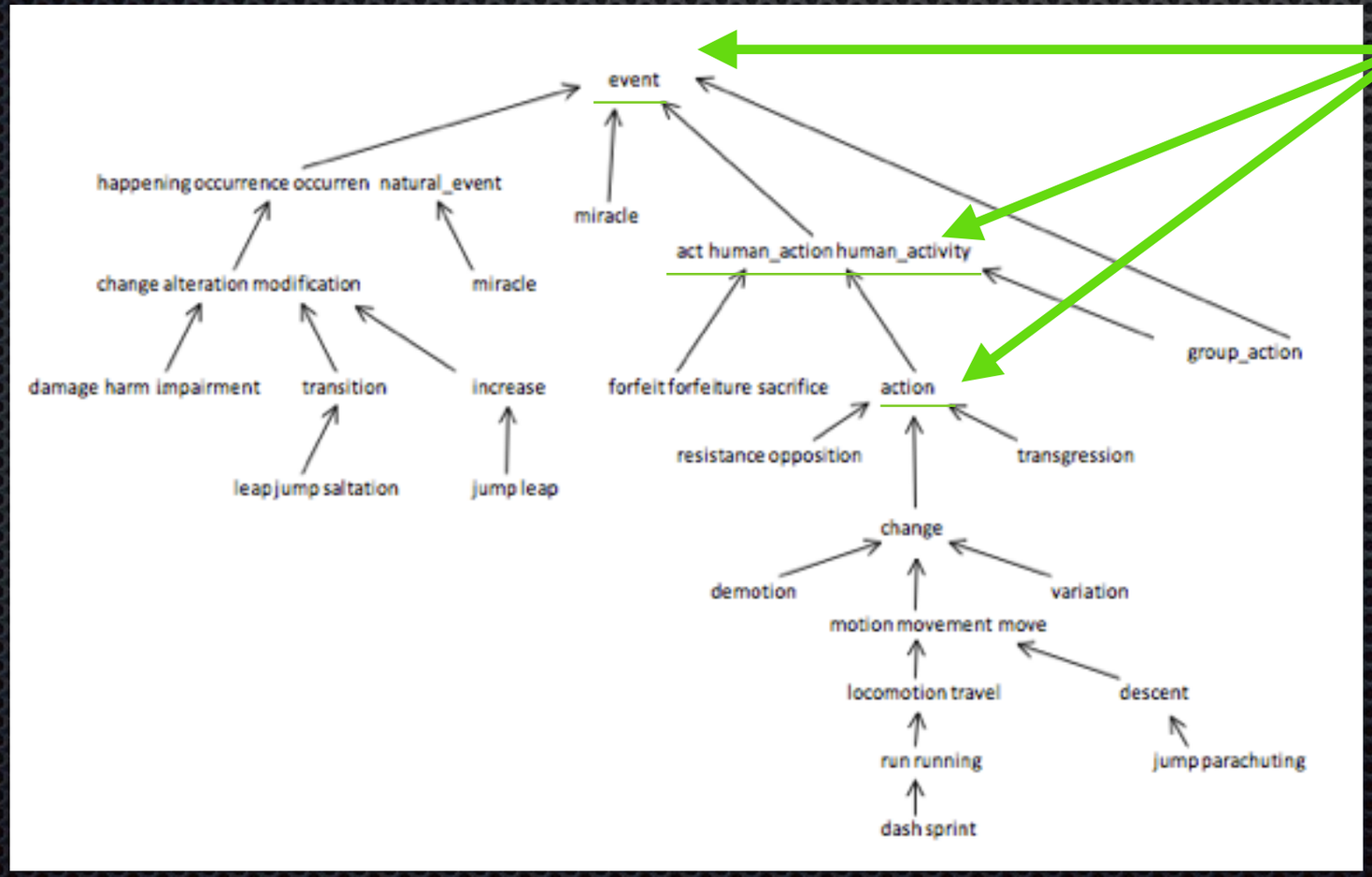
NLP Project : First Presentation

Article Analysis using hypernym tree

20130690 Sumin Han
20112034 Joohee Park

Intro: Hypernym Tree

- A **hyponym** is a word or phrase whose semantic field is included within its **hypernym**.
- Since many words can have a common **hypernym**, the tree structure can be made as figure below.

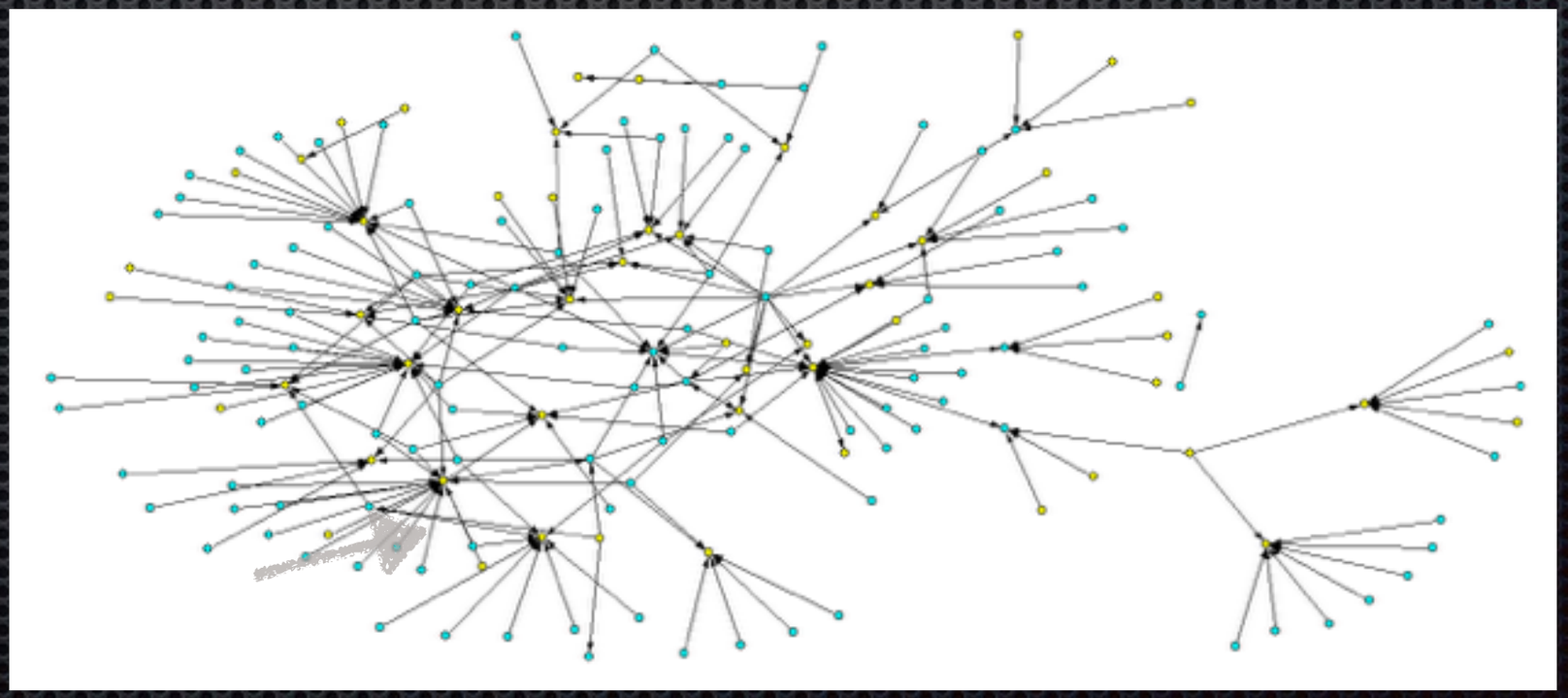


Popular Hypernyms:
Hypernyms that have relatively more hyponyms than others.

Defining the **popular hypernym** not only by the number of hyponyms but also by its level, their children's levels, or the frequency might be the challenging part of our project

Main Goal

- Analyze the main **keywords** of the given article by comparing "**popular hypernyms**" with golden standard, so that the program can **categorize** a given article automatically.
- Analyze the **similarity** of the hypernym tree using the network analysis methods and tools to **identify the author**. (We expect there is relationship between the writing style and the structure of the hypernym tree)



Sample of a network made by Pajek

System Design

- **V**: the set of nodes
- **E**: the set of directed edges: (from hyponym, to hypernym)
- Basic method to find **popular hypernyms**: use FreqDist on the number of edges for each V (we will enhance the algorithm later)

```
 1 V = []        # Node list
 2 E = []        # Edge list
 3 for w in set(file_words):
 4     if w not in stpwd:
 5         syns = wn.synsets(w)
 6         if syns != []:
 7             ws = syns[0] # use the first synset
 8             for path in ws.hypernym_paths():
 9                 for i in range(len(path) - 1):
10                     addToList(V, path[i])
11                     addToList(E, (path[i+1], path[i]))
12                 addToList(V, path[-1]) # adding last node
```

Input and Output definition

1. Categorization of article

Input : Article (Reuter's Corpus)
Output : Candidate words for its category

---

2. Author Identification

Input : Article (Online News article)
Output : Inference of its author

Measurement of performance

1. Categorization of article

```
>>> reuters.categories('training/9865')
['barley', 'corn', 'grain', 'wheat']
```

Compare the result categories to gold standard by using Reuters corpus.
Comparison will be made using distance of words in WordNet

---

2. Author Identification

Use F1 score for the test set. Better than just using accuracy

$$F_1 = 2 \cdot \frac{\mathrm{precision} \cdot \mathrm{recall}}{\mathrm{precision} + \mathrm{recall}}.$$

Measurement of system

- Time complexity and running time
- Performance quality
- Usefulness and Efficiency
- Comparing this with other algorithm (ex LDA)

Thank you

Any Question?