

Lecture 5 preview

1. Syntactic Structure: Consititueny and Dependency

- parsing : 문장의 문법적인 구성 또는 구문을 분석하는 과정
 - constituency parsing : 문장의 구성요소를 파악하여 문장구조를 분석하는 방법
 - dependency parsing : 단어 간의 의존관계를 파악하여 문장구조를 분석하는 방법
- constutueny parsing
 - phrase 단위로 문장을 파악한다
 - phrase structure grammer, context free grammer
 - 어순이 비교적 고정적인 언어에서 주로 사용된다
 - 문장을 구성하는 구, 구를 구성하는 문법적 의미를 지닌 단어들 분해
- dependency parsing
 - 단어 간 의존, 수식방향으로 관계를 파악하여 문장구조를 분석하는 방법
 - 자유로운 어순을 가지거나, 문장 성분이 생략 가능한 언어에서 사용되는 방법
 - head(수식을 받는 단어), modifier(수식을 하는 단어)
 - (수식하는 단어) is dependent of (수식받는 단어)
 - 단어 간 관계를 정립하여 문장을 트리구조로 표현할 수 있다
- Why do we need the sentence structure? → 모호성(ambiguity)
 - Phrase Attachment Ambiguity : 형용사구, 동사구, 전치사구 등이 어떤 단어를 수식하는지에 따라 의미가 달라지는 모호성(수식 대상이 명확하지 않아 중의성이 발생)
 - Coordination Scope Ambiguity : 특정 단어가 수식하는 대상의 범위에 따라 의미가 달라지는 모호성

2. Dependency Grammar and Treebanks

- Dependency structure
 - sequence

- tree

두 구조의 수식관계 표현은 동일해야한다

가상의 노드 root : 모든 성분의 최종 head

화살표는 수식받는 단어에서 수식하는 단어로 향한다

화살표 위에는 dependency를 마킹

화살표는 순환하지 않으며 중복의 관계가 형성되지 않음

- dependency conditioning preferences : dependency parsing의 보편적 특징
 - bilexical affinities : 두 단어 사이의 실제 의미가 드러난다
 - dependency distance : 문법적 관계의 거리, 주로 가까운 위치에서 dependent 관계가 형성된다
 - intervening material : 마침표, 세미콜론과 같은 구두점을 넘어 dependent한 관계가 형성되지는 않는다.
 - valency of heads : head의 주변에 위치하는 dependent의 개수와 관련된 특성

3. Transition-based dependency parsing

dependency parsing 의 4가지 방법 : dynamic programming, graph algorithms, constraint satisfaction, transition-based

- transition-based parsing : 두 단어의 의존 여부를 차례대로 결정해나가면서 점진적으로 문법적 구조를 형성
 - 속도는 빠르지만 낮은 정확성
 - 3가지 구조 : BUFFER, STACK, Set of Arcs(과정 좀 더 자세히 공부해볼 것!)
 - 초기 상태 : BUFFER에는 주어진 문장이 토큰으로 입력됨, STACK에는 ROOT만 존재, Set of Arcs에는 parsing 의 결과물이 담긴다.
 - BUFFER에서 STACK로 토큰이 이동, state를 기반으로 decision
 - Shift: BUFFER에서 STACK으로 이동하는 경우
 - Right-Arc: 우측으로 dependency가 결정되는 경우
 - Left-Arc: 좌측으로 dependency가 결정되는 경우
 - decision의 방법 : SVM, Neural Network

- dependency 가 형성되면 Set of Arcs로 이동한다
 - 다시 새로운 단어가 BUFFER에서 STACK로 업데이트 되어 dependency를 판단, Set of Arcs로 이동
- state의 임베딩 과정이 필요 → conventional feature representation
 - notation을 기반으로 indicator features 조건 설정
 - binary&sparse
 - notation 1-3개의 조합
 - 계산 비용이 높고 단어, 태그의 의미를 반영하지는 못한다는 단점

4. Neural Dependency parsing

neural network 가 적용된 dependency parsing

- input : words, POS tag, arc labels
- words feature의 데이터
 - STACK과 BUFFER의 TOP 3 words (6개)
 - STACK TOP 1, 2 words의 첫번째, 두번째 left & right child word (8개)
 - STACK TOP 1,2 words의 left of left & right of right child word (8개)