**NLP**

**Introduction**

What is NLP?

NLP is the field that studies the processing of human-like languages by computers in order to achieve a range of computational tasks.

**Important about NLP**

* Range of computational techniques
  + Methods to accomplish language analysis
* Naturally occurring texts
  + Any language, written or spoken
* Levels of linguistic analysis
* Human like language processing
  + NLP field inside AI
* Range of tasks or applications
  + Information retrieval, machine translation, question and answer

**Levels of linguistic analysis**

* Phonology
* Morphology
* Lexical
* Syntactic
* Semantic
* Pragmatic
* Discourse

**Regular Expressions**

Main approaches

* Rule based
* Statistical

**Rule based vs statistical method**

Rule based methods

* Manually design a sequence of rules
* Predicts based on the rules
* Made by linguistics specialists.
* Regular expressions
* Pattern Matching

Statistical methods

* Uses machine learning methods
* Makes statistical analysis of numerous texts.
* Builds frequency table
* Able to predict coming words by frequency
* Able to study language trends

**Regular Expressions / Pattern matching**

Simple rule based methods that allow us to process documents and make morphological analysis out of them.

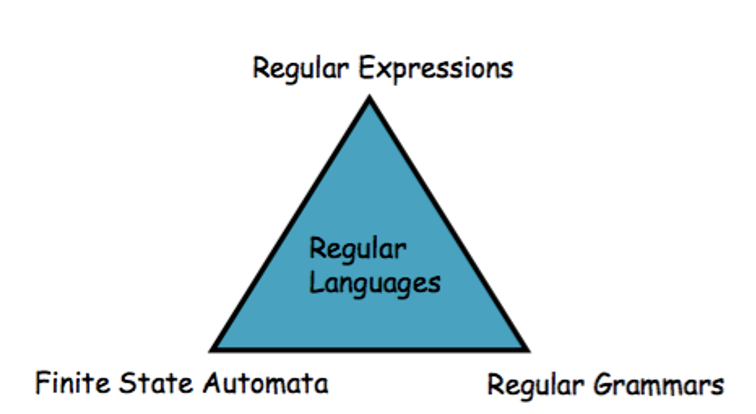


Figure 1 Three views of regular expressions

**Final State Automata (FSA)**

Useful tool to recognizing and generating subsets of natural language.

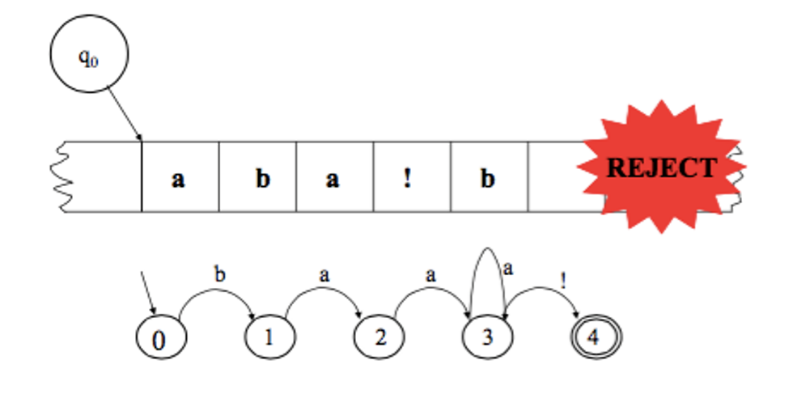
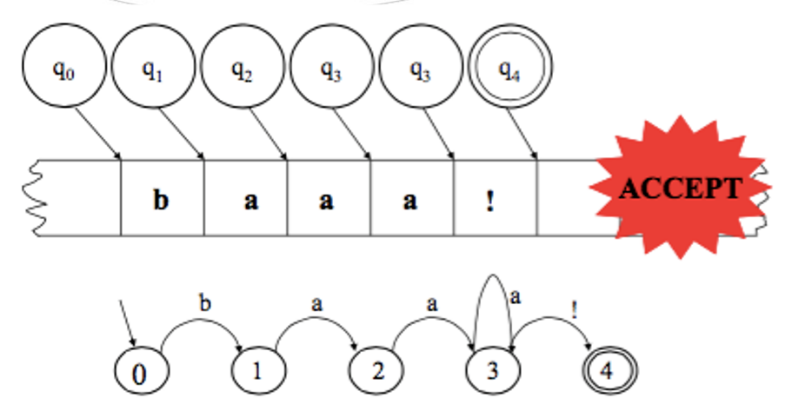
Cannot represent all the NL phenomena.

**Recognition**

Process of recognizing if a string should be accepted by the machine or there’s any regular expression that matches the string.

**Steps**

1. Start in the start state
2. Examine current input
3. Consult the table
4. Go to a new state and update the table pointer
5. Until run out of tape

Examples:

**Deterministic** FSA

Deterministic means that at every point in processing there is only one thing to do.

**Non deterministic** FSA

At any choise point we might follow the wrong arc.

Solutions

* Save backup states at each point
* Look ahead on input before making a choice
* Pursue alternatives in parallel
* Determine the NFSA and minimize

**Morphology**

What is morphology?

It’s the study of the words, they’re formation and relationship with other words.

Why is morphology important?

* Spelling correction
* Machine translation
* Hyphenation algorithms
* Part of speech analysis
* Text-to-speech
* Allows us to guess meaning

**Text processing**

* **Inflection –** Various forms of each word (singular, plural, past tense, etc.)
* **Derivation –** New words formed from pre-existing words often from different syntactic categories.

Morphology is finite resulting in a terminal node that lists all the possible forms of every word in the language.

**Techniques to implement morphology**

* List all the forms of every word
* **Tokenization** – breaking the text into tokens

**Complex word composition**

* Morphs – smallest meaningful segment of the word
* Morphemes - Constituted by several morphs and form a meaningful word.
  + Stems: core meaning
  + Affixes: bits and pieces that adhere to stems
* Allomorphs – Various forms of a morpheme that have the same syntactic and semantic effect.

Examples

Dogs = dog + s

dishes = dish + es

**Morphological Processes**

In English a word is constituted by zero or more prefixes followed by a root and 0 or more suffixes.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Word | Prefix | Prefix | Root | Suffix | Suffix |
| Untouchables |  | Un | Touch | able | S |
| Maintaining |  |  | Maintain | ing |  |

Other morphological processes

* Vowel change -> wr**o**te and wr**i**ting
* Zero morphs -> plural of sheep is sheep

**English inflection morphology**

Rules can be generated from an inflection morphology but only implemented on verbs.

1. Verbs
   1. Regular verbs
      1. Plain form (fido will bark)
      2. S form, singular (fido barks)
      3. Ing form, adjective or noun (a barking dog)/(his loud barking)
      4. ed form, past tense (barked)
   2. Irregular verbs
      1. Vowel change
      2. Fifth past tense. Uses has and another auxiliary verb (fido has eaten)
2. Adjectives and adverbs
   1. Suffix er or est (happier, happiest)
3. Noun possessive
   1. Regular
      1. s or es (dogs/ dishes)
   2. Irregular
      1. Child: children
      2. Plural has the same singular
      3. Borrowed from another language for morphology ( bacterium: bacteria)
4. Possessive ‘s
   1. Syntactically separates a word that is pronounced has part of another word

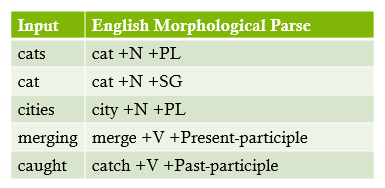
**Morphophonemic Rules**

Rules that change the spellings of various morphemes depending on the context.

1. Final E deletion (rating = rate + ing)
2. Y to I rule (carried = carry + ed)
3. S to ES rule ( dishes = dish + es)
   1. Appears as es after s, z , ch, sh and i.
4. Final consonant doubling (robbed = rob + ed)

What is required?

Something that will do the mapping



**Finite state transducer**

**Objective**

Use machinery with FSAs to capture facts about morphology

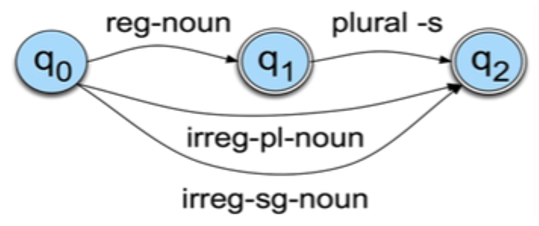
* Accept strings that are in the language
* Reject strings that are not
* No need to list all the words of the language
* Make morphological analysis

**Uses**

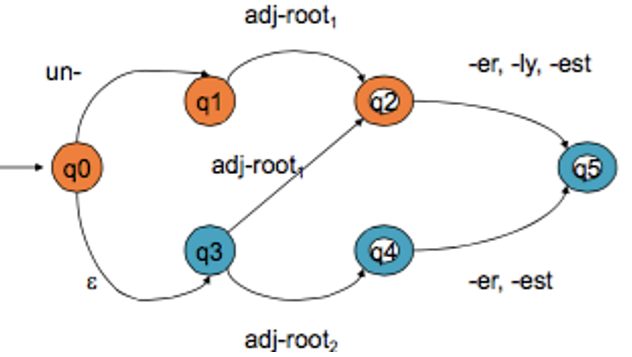
* Can be important standalone in an application
  + Spelling correction, information retrieval
* Simply a link in a chain of process

**Morphological parser**

* **Lexicon**: list stems and affixes (Part of speech).
* **Morphotactics** of the language: model of how can the morphemes be affixed to a stem.
* **sOrtographic** **rules**: spelling modifications that may occur when affixation happens.
* Most of phenomena can be described with regular expressions – finite state tecniques are used to represent the morphological process.

Examples:

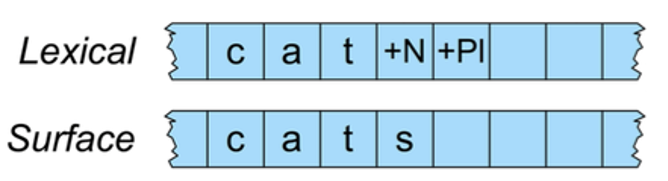
* Regular single nouns are ok
* Regular plural nouns have an s on the end
* Irregulars are ok



* Adj-root1: clear, happi , real
* Adj-root2: big, ready

**Idea**

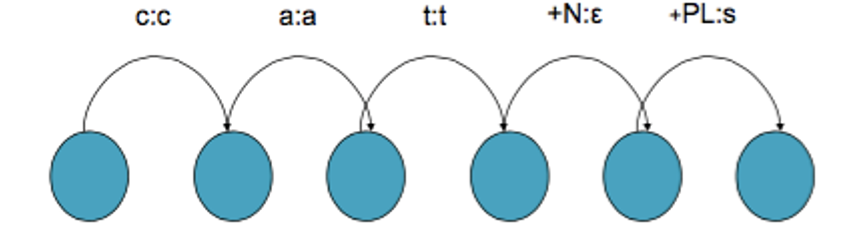
The idea behind an fst is: a word is a relation between the lexical level (morphemes) and the surface level (orthography).



**Transitions**

* C:C means reading a c on a tape and write a c on orthography
* +N: ε means reading a symbol n and writing nothing
* +Pl means reading +PL and writing an s

Example: Cats



N-grams and Corpus Linguistics

**What are N-grams?**

N-grams are Language Models that use N-1 words sequence to predict a sentence. These models are built based on a corpora. Probabilities can capture some syntactic facts and word knowledge of the language.

**Corpora -** Online collections of text and speech.

**Applications**

* Predict words
* Calculate likelihood of a sentence
  + Text generation
  + Machine translation

**Terminology**

**Sentence**: Unit of a written language.

**Utterance:** Unit of spoken language.

**Words Form**: The inflected form as it appears on the corpus

**Lemma**: Words dictionary form. Stands for the class of words with a stem.

**Types**: Number of distinct words in a corpus. (Vocabulary size)

**Tokens**: Total number of words.

**How N-Grams work**

* Assuming that a language has T types in its lexicon, how likely is X followed by Z?
  + Probability 1/T.
  + Alternative: estimate likelihood of x appearing in a text based on its general frequency of occurrence in a corpus.
    - Ex: popcorn is more likely than unicorn.
* **Alternative** **2**: Probability of X happen given the previous words.
  + - Ex: mythical unicorn is more likely than mythical popcorn.

**Computing the probability of a word sequence**

* The probability of a word depends only on the probability of n previous words.
* The longer the sequence, the less probable.
* Approximate using N-grams.
  + P(Wn|W1n-1) -> P(Wn | Wn-1)
* **Generalization**: The probability of a word depends on the probability of the previous n words.
  + Bigram, Trigram, 4-grams…
  + The higher the n the more we need to train.
* For **bigrams**:
  + P(the,mythical,unicorn) = P(unicorn|mythical) \* P(mythical|the) \* P(the|<start>)

**Training and testing**

* N-Gram probabilities come from a training corpus.
  + Overly narrow corpus: probabilities don’t generalize.
  + Overly general corpus: probabilities don’t reflect the task or domain.
* Separate test corpus is used to evaluate the model
  + Cross validation
  + Statistical significance
  + Test set

**Notes**

* Has we increase the value of N, the accuracy of an N-gram increases since the choice of the next word increasingly constraints.
* Small number of events occur with high frequency.
* Large number of events occur lower frequency.
* We can quickly collect statistics on high frequency event.
* Low frequency events will take much longer to get valid statistics.
* Some values in the table are really zeros but others are simply low frequency events.

**Part of Speech tagging**

**Smoothing**

* Words follow Zipfian distribution
  + A words frequency is approximately inversely proportional to its rank in the word distribution list.
* Zero probabilities on bigram cause a zero probability in the entire sentence.
* Smoothing, “takes from the rich and gives to the poor”

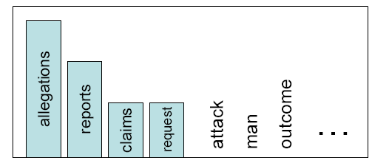
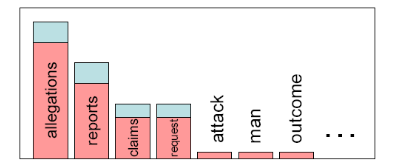


Figure 2 After smoothing

Figure 2 Before smoothing

**Smoothing methods**

* Add- one smoothing (inaccurate)
  + Add one to every word count
* Backoff models
  + When a count for an n-gram is 0, back of to the count of (n-1)-gram
  + Can be weighted
* Class-based smoothing
  + For certain types of n-grams back off to the count of its syntactic class.
  + Ex: count proper nouns in place of names (ex: Obama).
* Good Turing
  + Re-estimate amount of probability mass for zero or low count n-grams by looking at n-grams with higher counts.

**Word class**

Words somehow behave alike:

* Appear similar contexts
* Perform similar functions in sentences
* Undergo similar transformations

9 traditional parts of speech

* Noun, verb, adjective, preposition, adverb, article, interjection, pronoun and conjunction

**POS examples**

N noun chair, bandwidth

V verb study, debate

ADJ adjective tall, beautiful

ADV adverb unfortunately, slowly

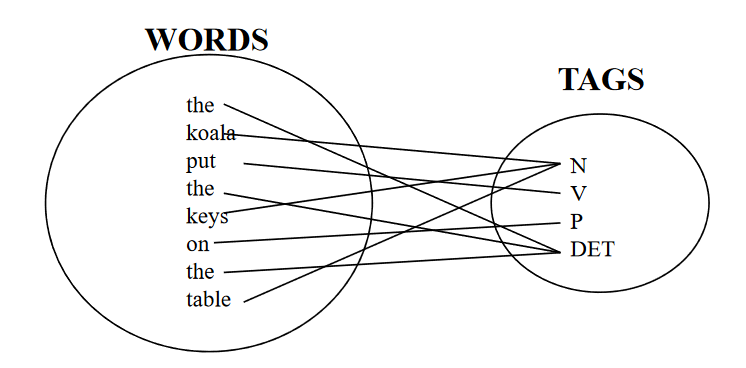
P preposition of, by, to

PRO pronoun I, me, mine

DET determinant the, a, that

**Tagging**

**Definition**: Process of assigning a POS or lexical class marker to each word in a corpus.



**Uses**

* Speech synthesis
  + How to pronounce
    - Ex: OBject obJECT
* Parsing
  + Need to know the tag of the word before parsing
* Word prediction in speech recognition
* Machine Translation

**Open class and closed class**

**Closed class:** relatively fixed membership

* Prepositions: of, in, b…
* Auxiliaries: may, can, will
* Prep, Det, Pron, Conj, Aux, Part, Num
* Idiosyncratic
* Usually function words (short common words that play roles in grammar)

**Open class:** New ones can be created all the time

* In English there’s 4: Nouns, Adverbs, Verbs, and Adjectives.
* Not all languages have 4.

Nouns

* Proper nouns (English capitalizes these)
* Common nouns (Germans capitalize these)
* Count nouns and mass nouns
  + Count: have plurals, gets counted: one goat, two goats…
  + Mass: don’t get counted: salt, water

Adverbs: tend to modify things

* Unfortunately John walked home

Verbs

* Have morphological affixes (eat/eats/eaten)
* Actions (walk, ate) and states (be)

How do we decide which word should go into each class?

* Nouns: denote people places and things and can be preceded by articles.
* Verbs: Are used to refer actions, processes and states
  + Some are closed class and some are open

Words often have more than one POS: back

* The back door = JJ
* On my back = RB

There is a problem in determining the instance of the word

**Potential sources of disambiguation**

* Most of words have only one POS
* Others have single most likely tag
* Tags also tend to co-occur regularly with other tags
* We can look at POS likelihoods P(t1|tn-1) to disambiguate sentences and to assess sentence likelihoods

**Methods for POS tagging**

1. Rule-based tagging
2. Transformation-based tagging
   1. Statistic and linguistic
3. Stochastic (probabilistic) tagging
   1. HHM tagging

**Rule-based tagging**

* Start with a dictionary
* Assign all possible tags to words from the dictionary
* Write rules by hand to selectively remove tags
* Leaving the correct tag for each word

Transformation-based tagging

* Combination of Rule-based tagging with probabilistic methods
  + Rules are used to specify tags in a certain environment
  + Stochastic because machine learning is used
    - Rules are learned
* Input:
  + Tagged corpus
  + Dictionary
* Basic idea
  + Set the most probable tag for each word as a start value
  + Change the tags according to the rules of type.
* Training is done with a tagged corpus
  + Use set of rules templates
  + Amongst the set of rules, find one with highest score
  + Continue finding the rules until lowest score is passed
  + Keep the ordered set of rules
* Rules make errors that are corrected by later rules

**TBL Rule Application**

* Tagger labels every word with most-likely tag:
  + Race in the Brown corpus:
    - P(NN|race)= .98
    - P(VB|race)= .02
* Transformation rules make changes to tags
  + Ex: Change NN to VB when previous tag is TO

Rule Learning

* Two parts
  + Triggering environment
  + Rewrite rule
* **Problem:** Could apply transformations ad infinitum
* **Solution:** Constraint the set of transformations with templates
* Rules are learned in ordered sequence
* Rules may interact
* Rules are compact and can be inspected by humans

**Computational Semantics**

Natural Language

* Syntax: Determines the structure of language
* Semantics: Determine the meaning of language
  + Answering Questions
  + Following instructions in a software manual
  + Following a recipe
  + Must link linguistic elements to world knowledge
  + Sentences have many entitlements, presuppositions
    - Ex: Instead, the protests turned bloody, as anti-government crowds were confronted by what appeared to be a coordinated group of Mubarak supporters.

Perspectives on meaning

* Meaning and the mind:
  + Meanings are mental constructs mediating between language and the world
* Meaning and action:
  + Meanings map language into actions (robotics)
* Semantics and models
  + Meaning maps onto states in the model theoretic ‘Worlds’
    - Focuses on truth conditions of sentences and their representation

**Challenges in Semantics**

* Semantic representation:
  + What is the appropriate formal language to express prepositions
    - Ex: predicate calculus
      * ∃x.(dog(x) ∧ disappear(x))
* Entailment
  + What are all the valid conclusions that can be drawn?
    - Lincoln was assassinated’ entails ‘Lincoln is dead.
* Reference
  + How do linguistic expressions link objects/ concepts in the real world?
    - ‘the dog’, ‘the President’
* Compositionality
  + How can we derive the meaning of unit from its parts?
    - How do syntactic structure and semantic composition relate?
      * Rubber duck vs rubber chicken
* Semantic analysis
  + How do we derive a representation of the meaning?
  + AyCaramba serves meat
* ∃e Isa(e,serving) ∧ server(e, AyCaramba) ∧ served(e, Meat)

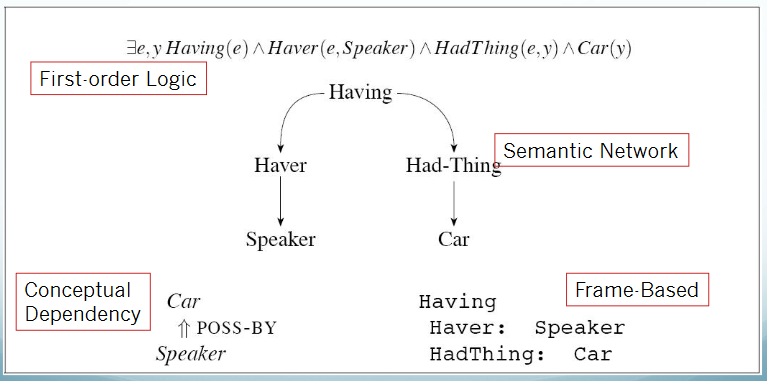
**Tasks**

* Defining a meaning representation
* Developing techniques for semantic analysis to convert NL strings to meaning representations
* Developing methods for reasoning about these representations and perform inference from them

**Complexity**

**Requires**

* Knowledge of the language, syntax, relationships, structure, meaning and composition procedures
* Knowledge of the world: what are the objects that we refer to, how they relate, what are their properties
* Reasoning: Given a representation and a world, what new conclusions can we infer? (bits of meaning)



**Meaning Representations**

* All consist of structures from set of symbols
  + Representational vocabulary
* Symbol structures correspond to
  + Objects
  + Properties of objects
  + Relations among objects
* Can be viewed as
  + Representation of meaning of linguistic input
  + Representation of state of world
* We focus on literal meaning

**Representational requirements**

Verifiability

* Can a system compare
  + Description of a state given by representation to
  + State of some world modeled by knowledge base?
* Is the preposition encoded by the representation true?
* Example
  + Input: Does Maharani serve vegetarian food?
  + Representation: Serves(Maharani, VegetarianFood)
  + KB: Set of assertions about restaurants
  + If representation matches in KB-> True
  + Else false or don’t know
    - Is KB assumed complete or incomplete?

**Unambiguous Representations**

* Semantics is ambiguous
  + I wanna eat someplace close to UW
    - Eat at someplace or eat the restaurant
* Representation must be unambiguous
  + Ex:
    - E1 = want(I, E2)
    - E2 = eat(I, O1m Loc1)

**Canonical form**

* Inputs can have many meanings and
* Many inputs can have same meaning
  + Flights from Seattle to Chicago
  + Are there any flights from Seattle to Chicago?
  + Do flights go from Seattle to Chicago?
* Could all have different forms
  + Difficult to test in KB
* Single canonical form allows consistent verification
* Issues
  + Pushes ambiguity resolution into semantic analysis
* Different surface forms but same underlying meaning
  + Words:
    - Word senses synonymy
    - Word sense disambiguation
  + Syntactive alternations:
    - Active vs passive
    - Interrogative vs declarative, topicalization, etc.

**Inference**

* Meanings are not identical but
* Linked by facts in the world
  + Can vegetarians eat at Maharani?
  + Does Maharani serve vegetarian food?
* **Inference** allows the system to draw valid conclusions from meaning
  + Serves(Maharani, VegetarianFood) ->
  + CanEat(Vegetarians, AtMaharani)

**Variables**

I want a restaurant that serves vegetarian food

* Can we match this in KB?
  + No restaurant specified, so no simple assertion match
* Solution:
  + Variables
    - Serves(X, VegetarianFood)
  + True if variable can be replaced for some object that resulting preposition can match some assertion in KB

**Meaning Structure of Language**

* Human languages
  + Display basic predicate-argument structure
  + Employ variables
  + Employ quantifiers
  + Exhibit a compositional semantics

**Predicate-Argument Structure**

* Represents concepts and relationships
* Words behave like predicates
  + Verbs, Adj, Adv:
    - Eat(John, VegetarianFood); Red(Ball)
* Some words behave like arguments
* Subcategorization frames indicate:
  + Number, Syntactic category, order of args

**First-Order Logic**

**Terms**

* **Constants:** Specific objects in world:
  + A, B, Maharani
  + Refer exactly one object
* **Functions:** concepts refer to objects
  + LocationOf(Frasca)
  + Refer to objects, avoid using constants
* Variables:
  + x,e

**Representation**

* Predicates
  + Relations among objects
    - Serves(Maharani, VegetarianFood)
* Logical connectives
  + Allow compositionality of meaning
    - Serves(Maharani, VegetarianFood) ∧ Cheap(Maharani)

**Variables & Quantifiers**

Variables refer to:

* Anonymous objects
* All objects in some collection

Quantifiers:

* ∃: existencial quantifier: “there exists”
  + There exists one restaurante that serves vegetarian food
    - ∃xRestaurant(x) ∧ Serves(x,VegetarianFood) ∧ Cheap(x)
* ∀: Universal quantifier: “for all”
  + All vegetarian restaurants serve vegetarian food
    - ∀xVegetarianRestaurant(x) -> Serves(x,VegetarianFood)

**Compositionality**

The meaning of a complex expression is a function of the meaning of its parts and the rules for their combination.

* Formal Languages are compositional
* NL meaning is largely though not fully compositional.