

Unit 2

Word and Morphology

Morphology, Word Construction

Natural Language Processing (NLP)
MDS 555



Objective

- Morphology
- Word Construction
- Use of FST in NLP tasks
- Lexicon
- Further Study
 - Chapter 2 , 3 of Text book



Finite State Transducers (FST)

- A finite state transducer essentially is a finite state automaton that works on two (or more) tapes.
 - The most common way to think about transducers is as a kind of “**translating machine**”.
- They read from one of the tapes and write onto the other.
 - This, for instance, is a transducer that translates a into b



FST: Formal Defination

Finite State Transducer (FST) is a 6-tuple $T = (Q, \Sigma, \Gamma, \delta, s, \gamma)$ where

Q is a finite set of states,

Σ is a finite set of input symbols,

Γ is a finite set of output symbols,

$\delta: Q \times \Sigma \rightarrow Q$ is the transition function,

$s \in Q$ is the start state.

$\gamma: Q \rightarrow \Gamma^*$ is the output function.



FST: Formal Defination

Our definition of FST is similar to that of a DFA, with the following differences:

- The FST includes not only an input alphabet Σ , but also an **output alphabet** Γ . Using different alphabets for input and output may be used to define transducers that convert between different alphabets.
- Instead of a set of accepting states F , an FST as an output function **$\gamma: Q \rightarrow \Gamma^*$**



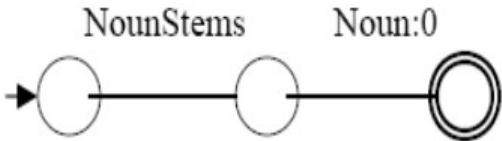
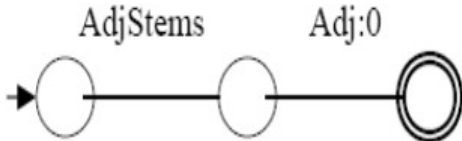
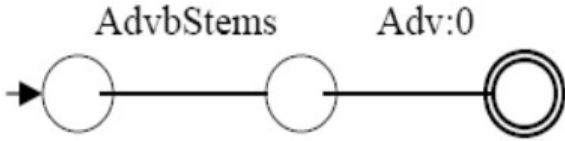
FST

- The important property of FSTs is that they are in **principle bi-directional**, meaning that they can also be applied backwards.
- The bi-directionality feature of the FST can be applied to the morphological analysis and generation



FST – Operations (Union)

- There are three FSTs for nouns, adjectives and adverbs

 <p>The diagram shows a finite state transducer for nouns. It consists of three states connected in a sequence. The first state is the start state, indicated by an incoming arrow from the left. The second state is an intermediate state. The third state is the final state, represented by a double circle. The transition from the first to the second state is labeled 'NounStems'. The transition from the second to the third state is labeled 'Noun:0'.</p>	 <p>The diagram shows a finite state transducer for adjectives. It consists of three states connected in a sequence. The first state is the start state, indicated by an incoming arrow from the left. The second state is an intermediate state. The third state is the final state, represented by a double circle. The transition from the first to the second state is labeled 'AdjStems'. The transition from the second to the third state is labeled 'Adj:0'.</p>	 <p>The diagram shows a finite state transducer for adverbs. It consists of three states connected in a sequence. The first state is the start state, indicated by an incoming arrow from the left. The second state is an intermediate state. The third state is the final state, represented by a double circle. The transition from the first to the second state is labeled 'AdvbStems'. The transition from the second to the third state is labeled 'Adv:0'.</p>
FST for nouns	FST for adjectives	FST for adverbs



FST – Operations (Union)

- When operation union is performed on these three FSTs, it results into a single FST,

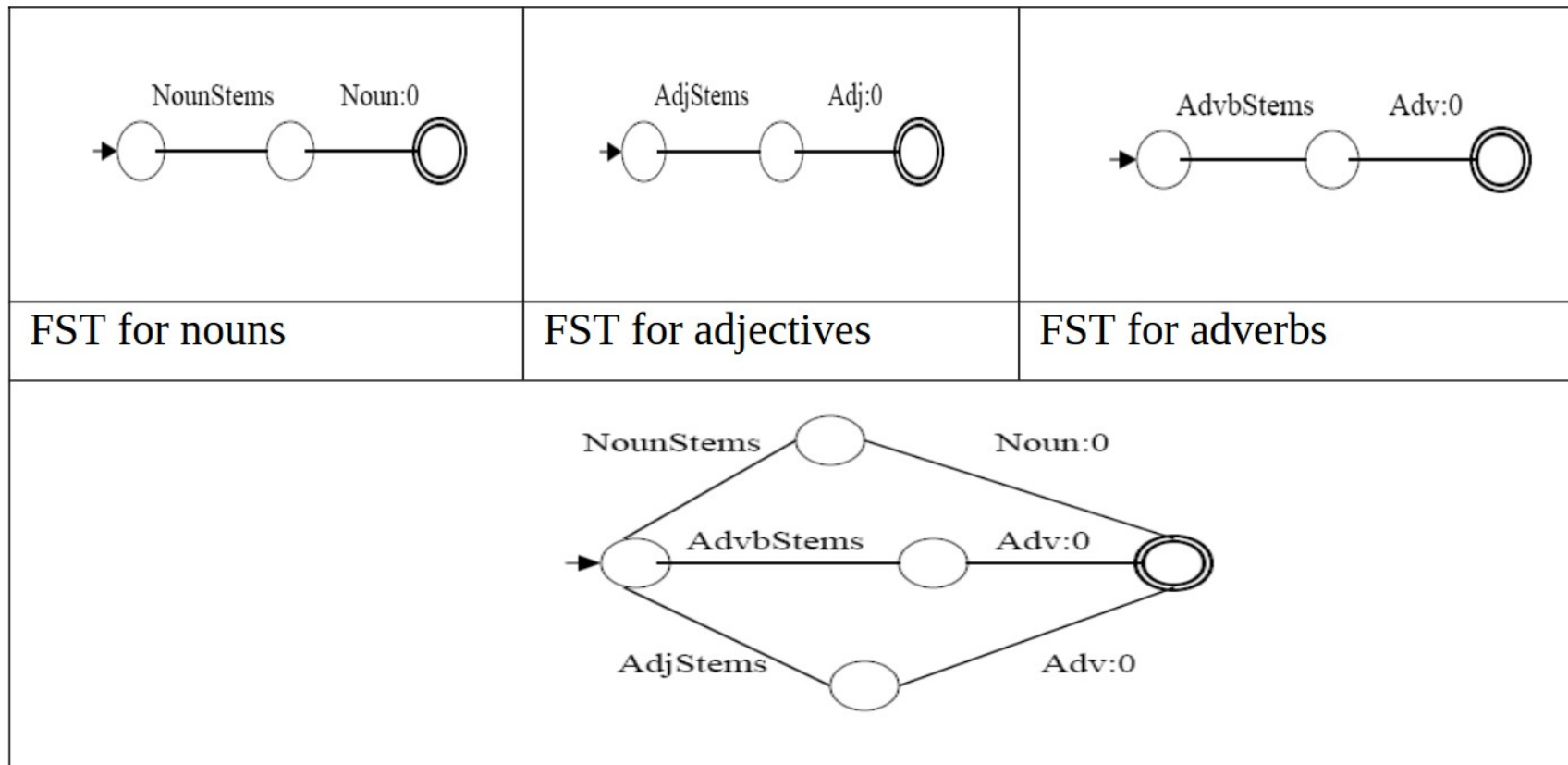


Figure 2.3: FST unioned from three FSTs for nouns, adjectives and adverbs



FST - Concatenation

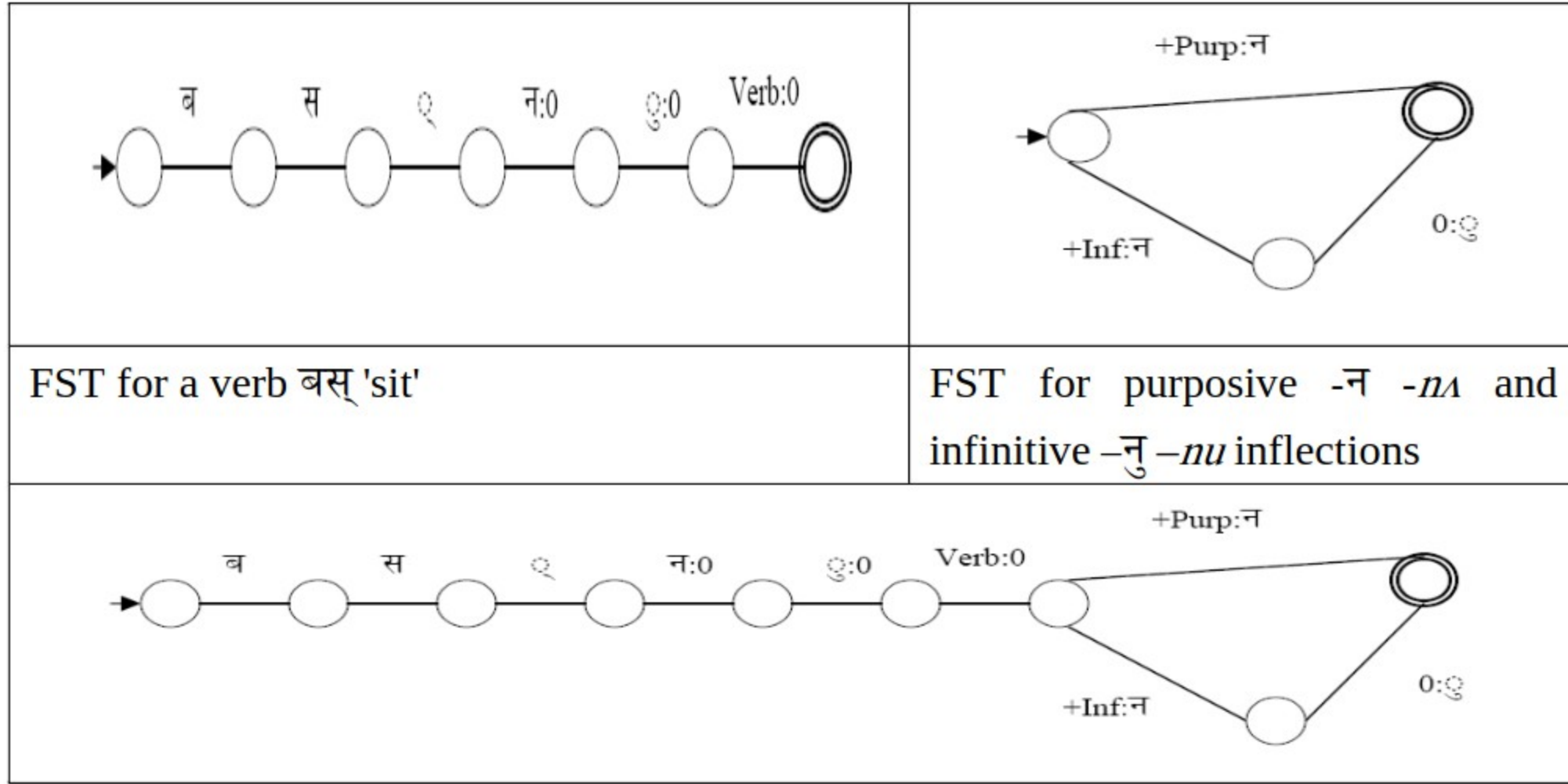


Figure 2.4: A finite state transducer concatenated from two FSTs above

FST - Concatenation

- In Figure 2.4, there are two FSTs in the upper part of the Figure 2.4, one for a Nepali verb बस् *bas* 'sit' and another for **purposive** - न - *na* 'PURP' and **infinitive** - नु - *nu* 'INF' suffixes.
- And in the lower part of the Figure 2.4, there is an FST resulted from concatenating two FSTs, which can **analyze and generate purposive and infinitive forms** of the verb बस् *bas* 'sit'.
- This **concatenation operation** is useful in handling the verb stems and inflectional and derivational suffixes.

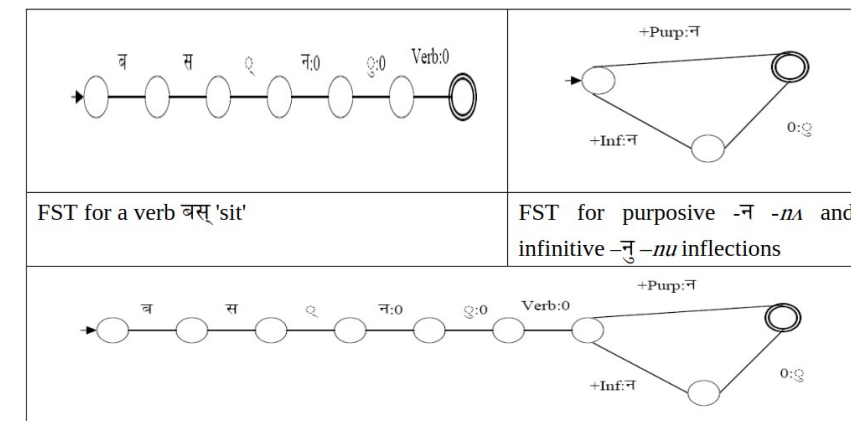


Figure 2.4: A finite state transducer concatenated from two FSTs above

FST - Composition

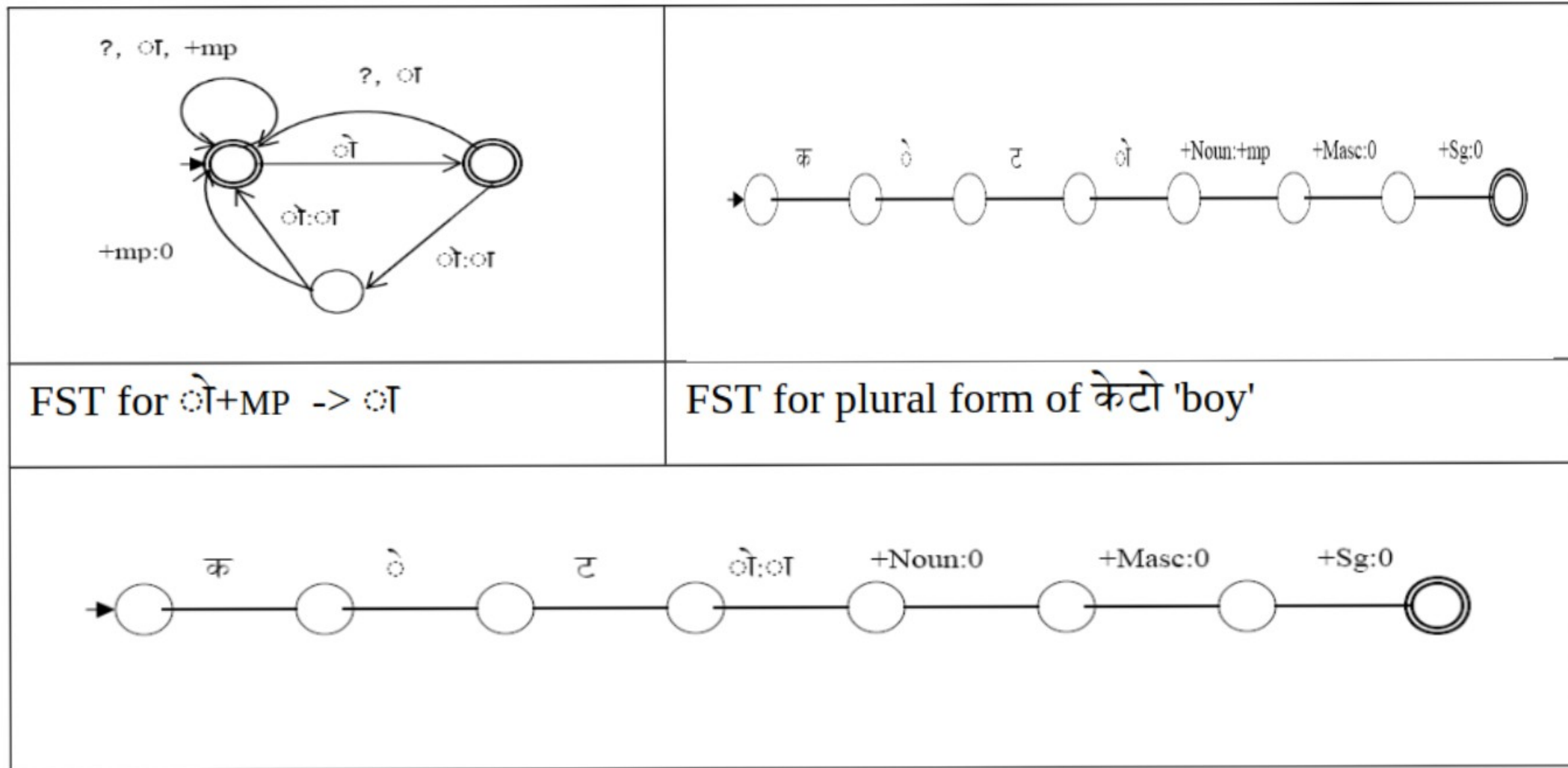
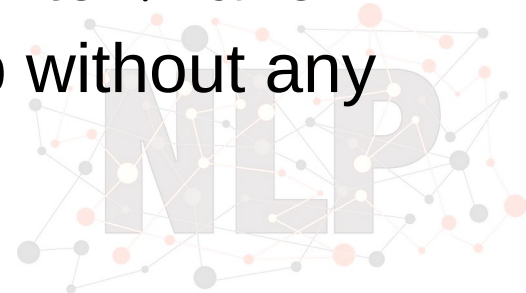


Figure 2.5: A finite state transducer from composing two FSTs above



FST - Composition

- There is a rule FST at the left top in Figure 2.5 which changes ो into ोT a for plural feature.
 - An arbitrary symbol + MP is used for creating the environment so that the rule can be applied to specific group of nouns.
- At the right top of the Figure 2.5, there is an FST for केटो 'boy' with +mp symbol.
 - When these FSTs are composed, it results into a single FST in lower part of the Figure 2.5 which is capable of changing ो o into ोT a for plural feature and also removes the arbitray symbol +mp without any intermediate FSTs.



FST - Composition

- In fact, composition operation forms a sequence of transducers.
 - It builds a cascade of FSTs into a single one by eliminating the common intermediate outputs, so, it allows working for a **modular structure**.
 - Because of this feature of composition, it has been very much useful for composing rules with lexicon to obtain the correct surface forms.



Study of Nepali Morphology

- COMPUTATIONAL ANALYSIS OF NEPALI MORPHOLOGY: A MODEL FOR NATURAL LANGUAGE PROCESSING
- <https://ojs.ub.uni-konstanz.de/jsal/dissertations/diss-balaram.pdf>
- Nepali Grammer Structure
 - https://www.researchgate.net/publication/237261579_Structure_of_Nepali_Grammar



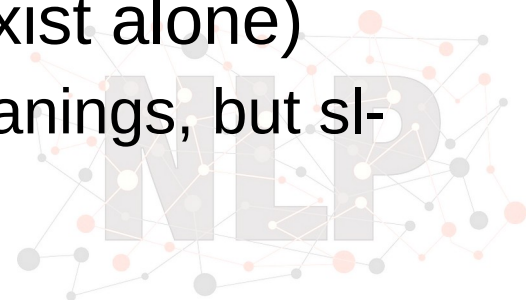
Morphology

- **Morphology** is the study of words, **how they are formed**, and **their relationship to other words** in the same language.
- **Internal structure** of the word
- It analyzes the structure of words and parts of words such as **stems**, **root words**, **prefixes**, and **suffixes**.



Morphology - vocabulary

- Morphology is the study of word structure
 - **morpheme**: a minimal information carrying unit
 - **affix**: morpheme which only occurs in conjunction with other morphemes (affixes are **bound** morphemes)
 - words made up of **stem** and zero or more affixes.
 - e.g. dog+**s**
 - **compounds** have more than one stem.
 - e.g. book+shop+**s**
 - **stems** are usually **free morphemes** (meaning they can exist alone)
 - Note that slither, slide, slip etc have somewhat similar meanings, but sl- not a morpheme



Affixes

- suffix: dog+s, truth+ful, किताब + हरू, किताब + मा
- prefix: un+wise, , सु +विचार, सु+पुत्र,
- infix: (maybe) abso-bloody-lutely => **absolutely**
- circumfix: not in English

German ge+kauf+t (stem kauf, affix ge_t)

- Eg: enlightened = en + lighten + ed
- न + काम + को , बे + काम + को



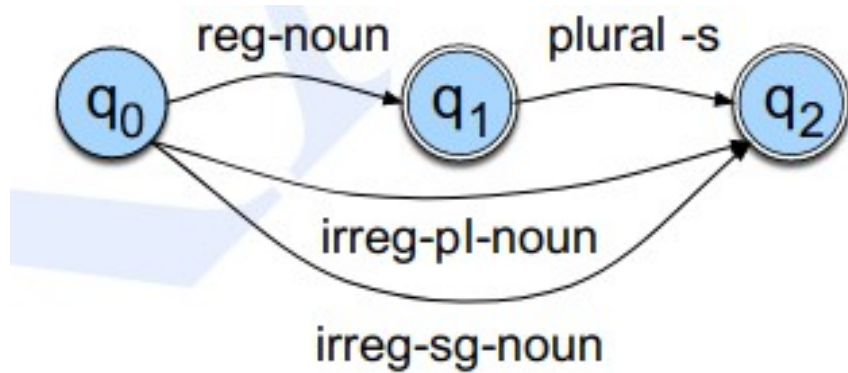
Inflectional morphemes

- Inflectional morphemes carry grammatical information
- Inflectional morphemes can tell us about tense, aspect, number, person, gender, case...
 - e.g., plural suffix +s, past participle +ed



Inflectional morphemes

- A finite-state automaton for English nominal inflection



reg-noun	irreg-pl-noun	irreg-sg-noun	plural
fox	geese	goose	-s
cat	sheep	sheep	
aardvark	mice	mouse	

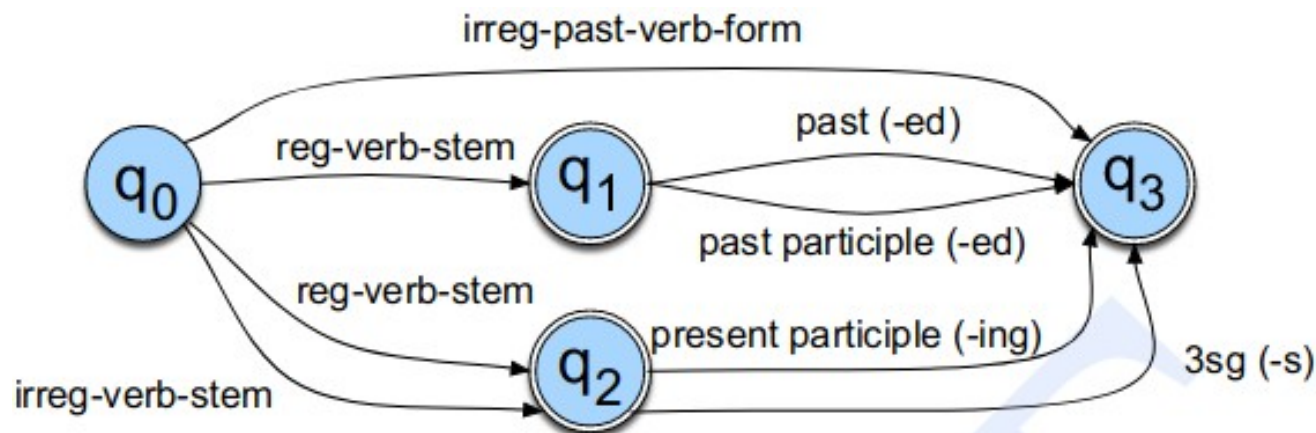
- regular nouns (reg-noun)
- Irreg – irregular
- sg – singular
pl - plural



Inflectional morphemes

- A finite-state automaton for English nominal inflection

reg-verb-stem	irreg-verb-stem	irreg-past-stem	past	past-part	pres-part	3sg
walk	cut	caught	-ed	-ed	-ing	-s
fry	speak	ate				
talk	sing	eaten				
impeach		sang				



Derivational morphemes

- Derivational morphemes change the meaning
 - e.g., un-, re-, anti-, -ism, -ist ...
 - broad range of semantic possibilities
 - may **change part of speech**:
 - help (Verb) → helper (Noun)
- indefinite combinations:
 - antiantidisestablishmentarianism
 - anti-anti-dis-establish-ment-arian-ism

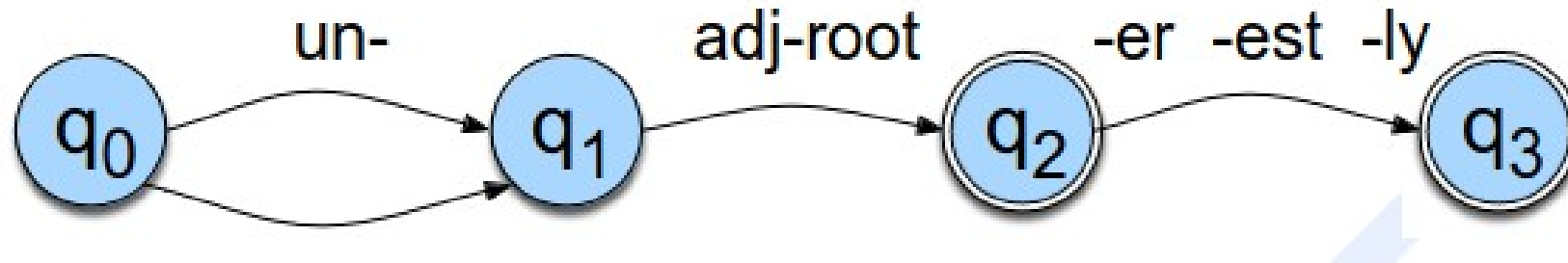


Derivational morphemes

- Adjectives become opposites, comparatives, adverbs

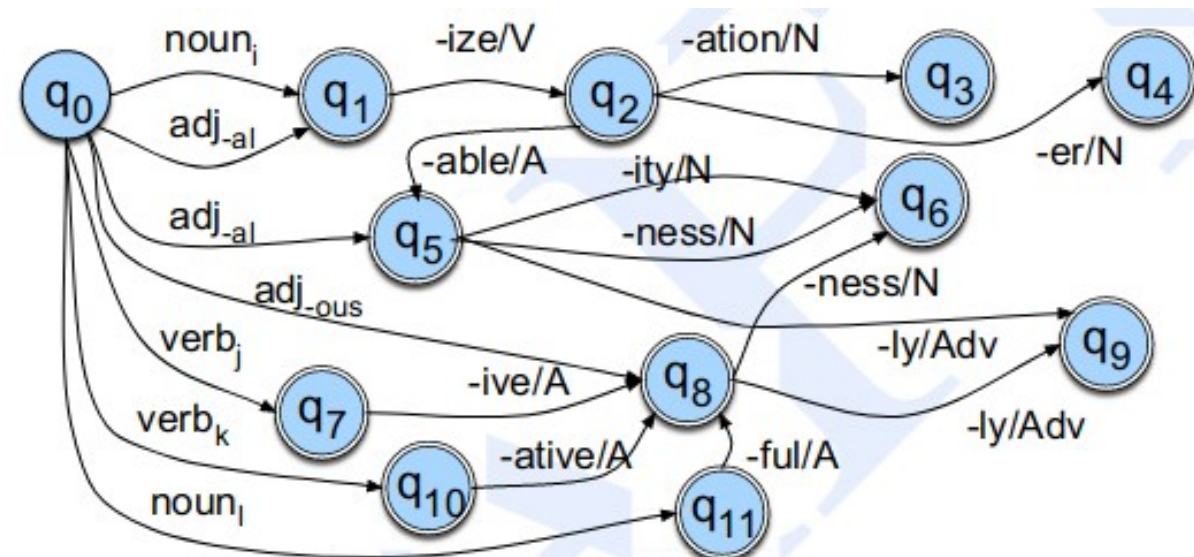
big, bigger, biggest,	cool, cooler, coolest, coolly
happy, happier, happiest, happily	red, redder, reddest
unhappy, unhappier, unhappiest, unhappily	real, unreal, really
clear, clearer, clearest, clearly, unclear, unclearly	

- FSA



Derivational morphemes

- FSA for fragment of English derivational morphology
 - This FSA models a number of derivational facts
 - Such as the well known **generalization** that any verb ending in **-ize** can be followed by the nominalizing suffix **-ation** (Bauer, 1983; Sproat, 1993)
 - Fossilize → fossilization :
 - q0, q1, and q2
 - **Adjectives:** ending in **-al** or **-able** at q5 (equal, formal, realizable) can take the suffix **-ity**
 - **real** => **reality**



Cliticization

- **clitic** is a unit whose status lies in between that of an affix and a word
 - The phonological behavior of clitics is like affixes; they tend to be short and unaccented
 - Their syntactic behavior is more like words, often acting as pronouns, articles, conjunctions, or verbs.
 - **clitics** in English are ambiguous
 - **she's** can mean she **is** or she **has**

Full Form	Clitic	Full Form	Clitic
am	'm	have	've
are	're	has	's
is	's	had	'd
will	'll	would	'd

Recognition vs Parsing

- Morphological recognition
 - `is_past_tense_verb(loved)` --> **TRUE**
 - Finite State Automata can do this
- Morphological parsing: what is its breakdown?
 - `parse(loved)` --> **take/VERB -n/PAST-TENSE**
 - Finite State Transducers can do this



FST - Parsing

- An FST $T = L_{in} \times L_{out}$ defines a relation between two regular languages L_{in} and L_{out} :

$L_{in} = \{\text{cat}, \text{cats}, \text{fox}, \text{foxes}, \dots\}$

$L_{out} = \{\text{cat}+N+sg, \text{cat}+N+pl, \text{fox}+N+sg, \text{fox}+N+PL \dots\}$

$T = \{$
 $\langle \text{cat}, \text{cat}+N+sg \rangle,$
 $\langle \text{cats}, \text{cat}+N+pl \rangle,$
 $\langle \text{fox}, \text{fox}+N+sg \rangle,$
 $\langle \text{foxes}, \text{fox}+N+pl \rangle \}$



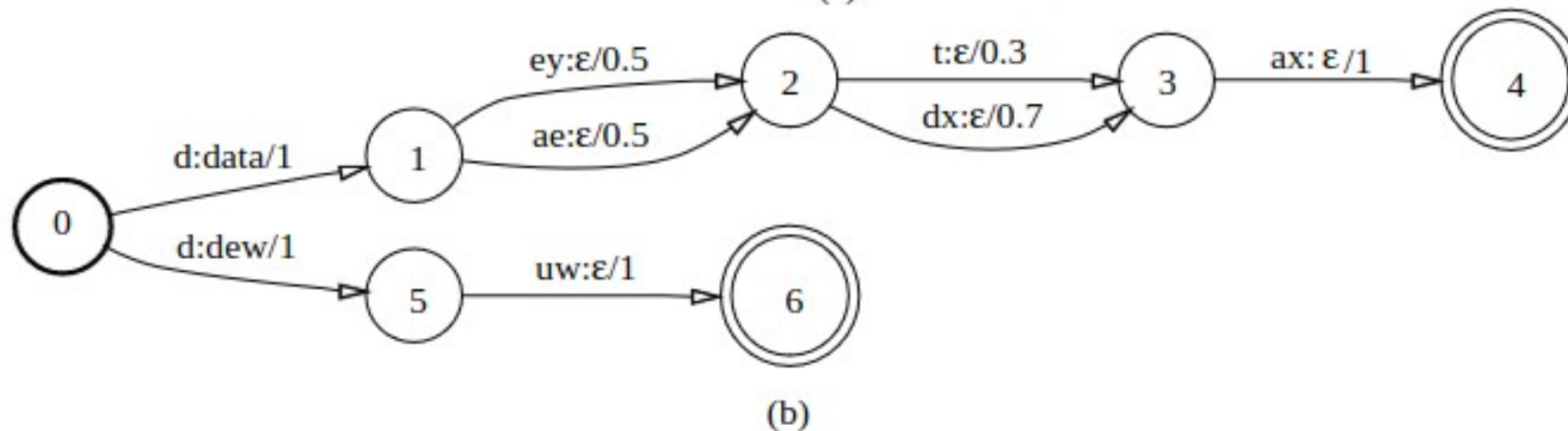
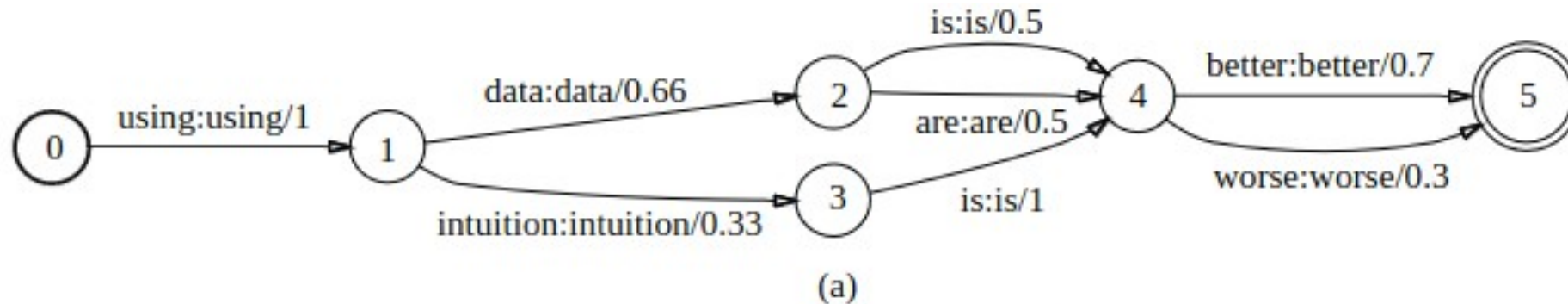
Using FST

- Weighted Finite-State Transducers
- The weights used in speech recognition often represent probabilities; the corresponding semiring is then the **probability semiring**
 - $(\mathbb{R} ; + ; \cdot ; 0 ; 1)$
- Ref:
 - <https://www.openfst.org>
 - <https://www.openfst.org/twiki/pub/FST/FstBackground/csl01.pdf>
 - <https://www.openfst.org/twiki/pub/FST/FstBackground/hbka.pdf>



Weighted FSTs

- Weighted finite state transducer examples



Morphological Analysis with Finite State Transducers

- “**wizard**”, consists of only one morpheme, namely wizard,
- “**wizards**” consists of two morphemes, namely **wizard** and **s** where **s** contributes the plural.
- “**kissed**” also consists of two morphemes, namely **kiss** and the past tense **ed**.



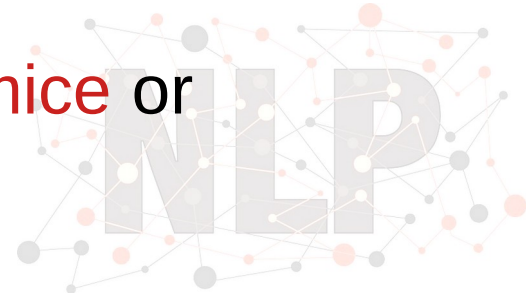
Morphological Analysis with Finite State Transducers

- Morphology is an area of **computational linguistics** where finite state technology has been found to be particularly useful
 - For many languages the rules after which morphemes can be combined to build words can be captured by finite state automata.
 - It is possible to write finite state transducers that map the surface form of a word to a description of the morphemes that constitute that word or vice versa.
- They map, for instance, **wizard+s** to **wizard+PL** or **kiss+ed** to **kiss+PAST**.



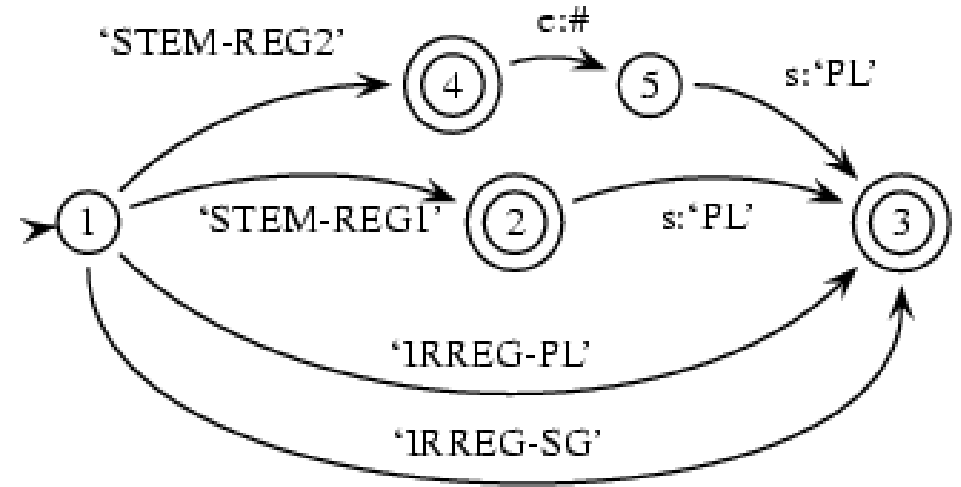
Morphology – Plural Noun

- Plural nouns in English.
 - The default rule is of course to **just add** an **s** as in **wizard+s**.
 - There are some stems which **take es** to form the plural, like **witch** e.g.
 - This can be explained by **morpho-phonological** rules that insert an e whenever the **morpheme preceding the s ends in s, x, ch** or another **fricative**.
 - For simplicity, we will assume here that there are two types of regular stems: those that take an s to form the plural and those that take an es.
 - Finally there are clearly irregular forms like **mouse and mice** or **automaton and automata**.



FST - Plural Noun

- Transducer that translates
 - wizard+s into wizard+PL
 - witch+es into witch+PL,
 - mice, into mouse+PL
 - automata into automaton+PL.
- Lexicon
 - `lex(wizard:wizard, 'STEM-REG1')`.
 - `lex(witch:witch, 'STEM-REG2')`.
 - `lex(automaton:automaton, 'IRREG-SG')`.
 - `lex(automata:'automaton-PL', 'IRREG-PL')`.
 - `lex(mouse:mouse, 'IRREG-SG')`.
 - `lex(mice:'mouse-PL', 'IRREG-PL')`.



Lexicon

- A lexicon refers to the **vocabulary** or **dictionary** of a language.
 - It is a **collection of words, phrases, and other linguistic units** along with their meanings, pronunciations, and grammatical properties.
 - The lexicon is a fundamental component of a language, serving as a repository of the building blocks used for communication.



Lexicon

- In linguistics and natural language processing, the lexicon includes not only individual words but also
 - various multi-word expressions,
 - Idioms
 - grammatical morphemes (prefixes, suffixes, etc.),
 - information about how these elements are used in context.
- It's important to note that a **lexicon is not just a static list of words**; it also encompasses information about the
 - **relationships between words**, such as synonyms, antonyms, homonyms
 - various **semantic associations**



Lexicon

- For example, consider the word "run."
 - In a lexicon, you would find information about its
 - meaning (to move swiftly on foot),
 - grammatical properties (a verb),
 - various forms (running, ran),
 - its pronunciation.
- Additionally,
 - the lexicon might also include information about related words like "runner" (a person who runs) and "running shoes" (footwear designed for running)



Uses of Lexicon

- In NLP, lexicons are crucial for tasks like
 - part-of-speech tagging
 - word sense disambiguation
 - sentiment analysis
- Lexical resources play a vital role in **training machine learning** models to understand and generate human language



Further reading

- Book “ Speech and Language Processing – Jurafky and Martin”
 - Chapter 2 and 3 of
- <https://brilliant.org/wiki/finite-state-machines/>
- <https://nepalishabdakosh.com/>
- Nepali Examples are taken from: A COMPUTATIONAL ANALYSIS OF NEPALI MORPHOLOGY: A MODEL FOR NATURAL LANGUAGE PROCESSING <https://ojs.ub.uni-konstanz.de/jsal/dissertations/diss-balaram.pdf>
- <https://cs.union.edu/~striegnk/courses/nlp-with-prolog/html/index.html>
- Lexicon
 - Blog article of Lexicon: <https://mohamedbakrey094.medium.com/all-about-lexicons-in-nlp-12ada00c2821>
 - The Role of Lexicon in NLP: <https://dl.acm.org/doi/pdf/10.1145/234173.234204>



Thank you

