Unit-I

Unit-I

- Finding the Structure of Words
- Finding the Structure of Documents

Finding the Structure of Words

- Words and their Components
    - Tokens
    - Lexemes
    - Morphemes
    - Typology
- Issues and Challenges
    - Irregularity
    - Ambiguity
    - Productivity
- Morphological Models
    - Dictionary Lookup
    - Finite-State-Morphology
    - Unification-Based Morphology
    - Functional Morphology
    - Morphology Induction

Finding the Structure of Words-Introduction

- Human Language is used to express our thoughts, and through language, we receive information and infer its meaning.
- Linguistic Expressions (words, phrases, sentences) show structure of different kinds and complexity and consist of more elementary components.
- The co-occurrence of linguistic expressions in context refines the notions they refer to in isolation and implies further meaningful relations between them.

Finding the Structure of Words-Introduction

- The whole disciplines that look at languages from different perspectives and at different levels of detail are:
  - **Morphology**- study the variable forms and functions of words.
  - **Syntax**- It is concerned with the arrangement of words into phrases, clauses and sentences.
  - **Phonology**- describes the word structure constraints due to pronunciation.
  - **Orthography**-deals with the conventions for writing in a language.
  - **Etymology** and **Lexicography**- evolution of words and explains the semantic, morphological and other links among them.

Finding the Structure of Words-Morphological Parsing

- Here we discuss about:
  - How to identify words of distinct types in human languages?
  - How the internal structure of words can be modelled with respect to grammatical properties and lexical concepts the words should represent?
- This discovery of word structure is called as **morphological parsing**.

Words and their Components

- Words in most languages are the smallest linguistic units that can form a complete utterance by themselves.
- Three important terms which are integral parts of a word are:
  - **Phonemes** – the distinctive units of sound in spoken language.
  - **Graphemes** – the smallest unit of a written language which corresponds to a phoneme.
  - **Morphemes** - the minimal part of a word that delivers aspects of meaning to the word.

Words and their Components

- Tokens
- Lexemes
- Morphemes
- Typology

Tokens

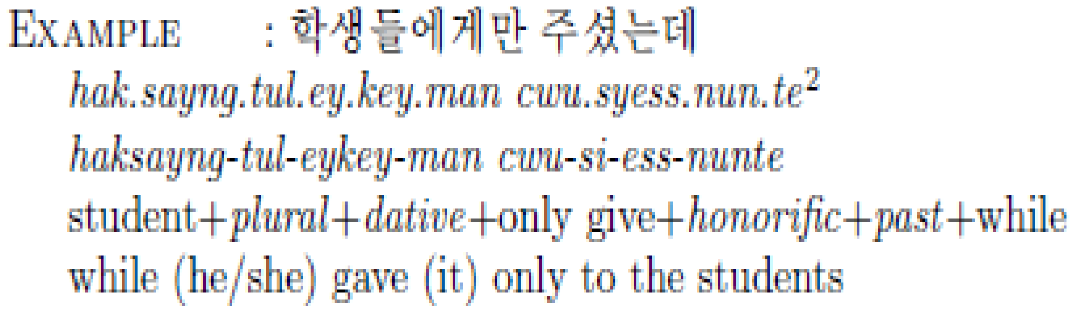
- Let us look at an example in English:

  Will you read the newspaper? Will you read it? I won’t read it.

- Here we see two words **newspaper** and **won’t.**
- In writing, newspaper and its associated concepts are very clear but in speech there are a few issues.
- When it comes to word won’t linguists prefer to analyze it as two words or tokens **will** and **not**.
- This type of analysis is called **tokenization** and **normalization**.

Tokens

- In Arabic or Hebrew certain tokens are concatenated in writing with the preceding or the following words, possibly changing their forms.
- This type of tokens are called clitics (I’m, we’ve).
- In the writing systems of Chinese, Japanese and Thai white space is not used to separate words.
- In Korean character strings are called eojeol ‘word segment’ and correspond to speech or cognitive units which are usually larger than words and smaller than clauses.



EXAMPLE : 학생들에게만 주셨는데

hak.sayng.tul.ey.key.man cwu.syess.nun.te[2]

haksayng-tul-eykey-man cwu-si-ess-nunte

student+*plural*+*dative*+only give+*honorific*+*past*+while

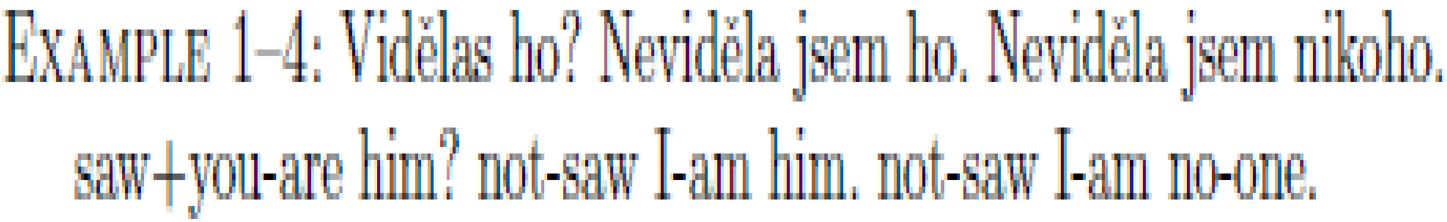
while (he/she) gave (it) only to the students

Lexemes

- There are a lot of alternative forms that can be expressed for a given word.
- Such sets are called lexemes or lexical items.
- They constitute the lexicon of a language.
- Lexemes are divided by their lexical categories such as verb, noun, adjective, adverb etc.
- The citation form of a lexeme by which it is identified is called lemma.
- In the conversion of singular mouse to plural mice we **inflect** the lexeme.
- In the case of receiver and reception we **derive** the words from the verb to receive.

Lexemes

- Example: Did you see him? I didn’t see him? I didn’t see anyone.
- Example in Czech



EXAMPLE 1–4: Vidělas ho? Neviděla jsem ho. Neviděla jsem nikoho.
saw+you-are him? not-saw I-am him. not-saw I-am no-one.

- Example in Telugu:

vAlYlYu aMxamEna wotalo neVmmaxigA naduswunnAru.
They beautiful garden slowly walking

Morphemes

- The structural components that associate the properties of word forms are called morphs.
- The morphs that by themselves represent some aspect of the meaning of a word are called **morphemes** of some function.
- Example : dis-agree-ment-s where agree is a free lexical morpheme and other elements are bound grammatical morphemes.
- Morphs when interact with each other undergo additional phonological and orthographic changes.
- These alternative forms are called allomorphs.
- Example: the past tense morphemes, plural morphemes etc.

Typology

- Morphological typology divides languages in groups. Here we outline the typology that is based on quantitative relations between words, their morphemes and their features:
- **Isolating**, or **analytic**, languages include no or relatively few words that would comprise more than one morpheme (typical members are Chinese, Vietnamese, and Thai; analytic tendencies are also found in English).
- **Synthetic** languages can combine more morphemes in one word and are further divided into agglutinative and fusional languages.
    - **Agglutinative** languages have morphemes associated with only a single function at a time (as in Korean, Japanese, Finnish, and Tamil, etc.).
    - **Fusional** languages are defined by their feature-per-morpheme ratio higher than one (as in Arabic, Czech, Latin, Sanskrit, German, etc.).

Typology

- In accordance with the notions about word formation processes mentioned earlier, we can also discern:
  - **Concatenative** languages linking morphs and morphemes one after another.
  - **Nonlinear** languages allowing structural components to merge non-sequentially to apply tonal morphemes or change the consonantal or vocalic templates of words.

Issues and Challenges

- Irregularity
- Ambiguity
- Productivity

Issues and Challenges- Introduction

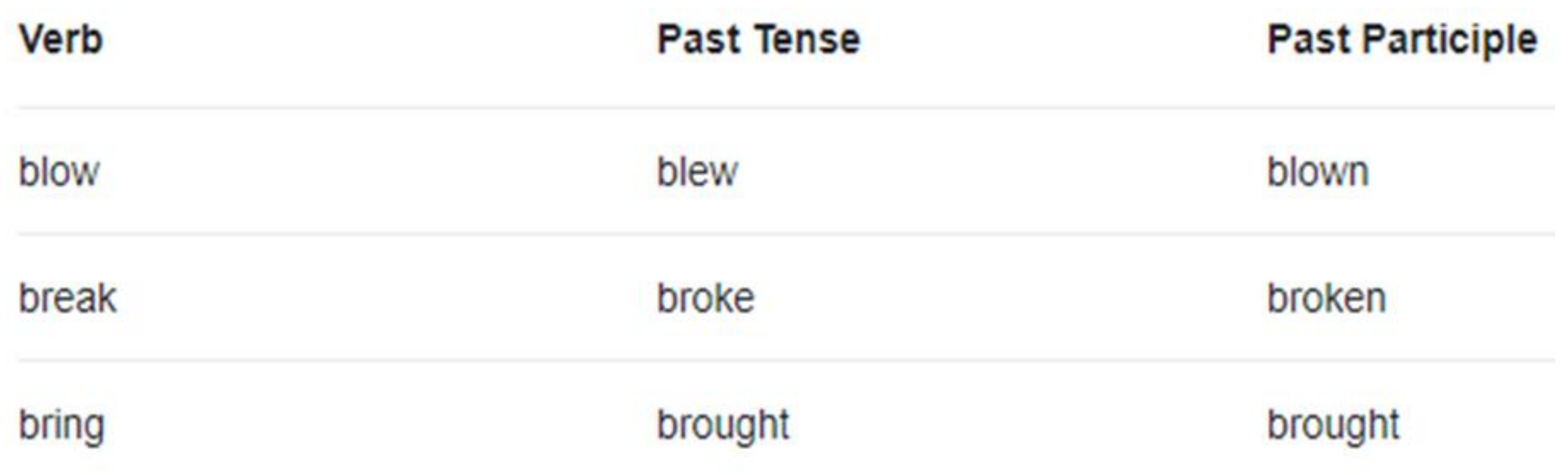
- Morphological parsing tries to remove unnecessary **irregularities** and give limits to **ambiguity** both of which exist in natural languages.
- Irregularity is all about forms and structures that are not described appropriately by a prototypical linguistic model.
- Ambiguity is indeterminacy in interpretation of expressions of language.
- In addition to ambiguity we need to deal with the issues of **syncretism**, or systematic ambiguity. (bet)
- In addition to the above morphological modelling also faces the problem of **productivity** and creativity in language, by which new but perfectly meaningful new words or new senses are coined.

Irregularity

- Morphological parsing provides generalization and abstraction in the world of words.
- Irregular morphology can be seen as enforcing some extended rules the nature of which is phonological, over the underlying or prototypical regular word forms.
- In English the general past form occurs by adding –ed or –t.(accepted and built)
- The irregular verbs in English tend to take different forms in the past or in the present participle depending on the origin of the word.

Irregularity

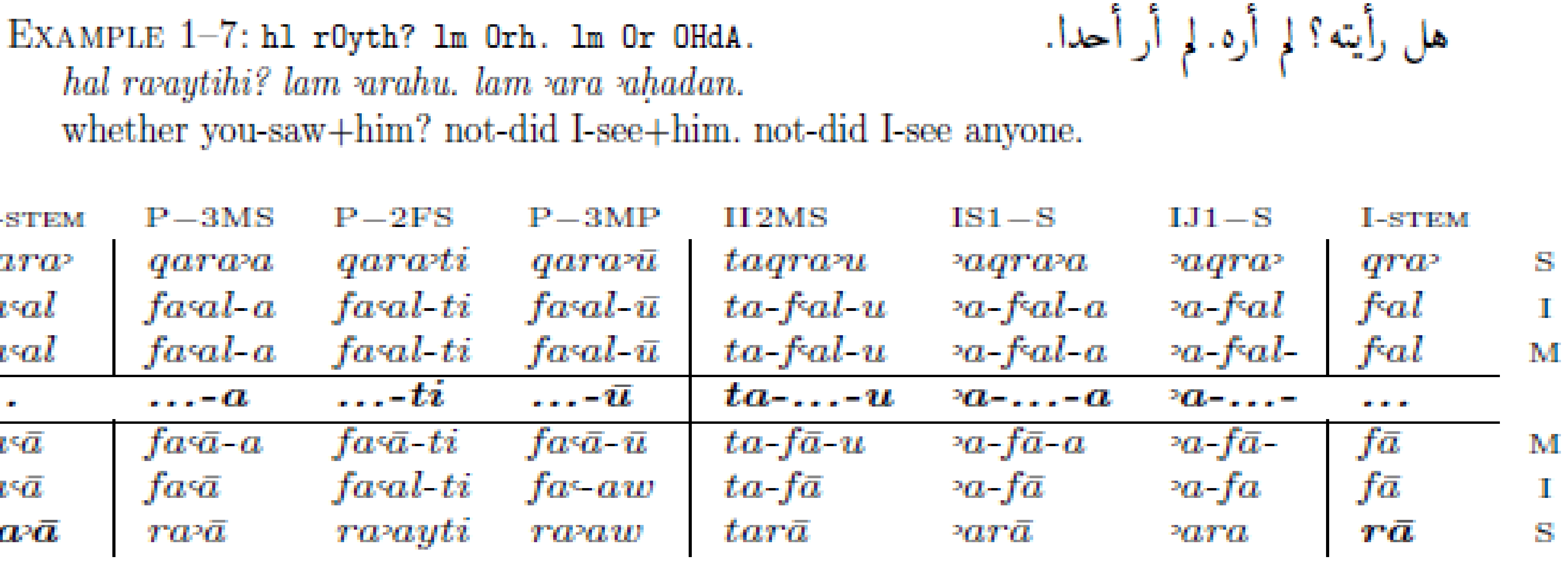
- A few examples:



Verb	Past Tense	Past Participle
blow	blew	blown
break	broke	broken
bring	brought	brought

Irregularity

- Example in Arabic:



EXAMPLE 1–7: `hl rOyth? lm Orh. lm Or OHdA.` هل رأيته؟ لم أره. لم أر أحدا.

hal raʾaytihi? lam ʾarahu. lam ʾara ʾaḥadan.

whether you-saw+him? not-did I-see+him. not-did I-see anyone.

P-STEM	P–3MS	P–2FS	P–3MP	II2MS	IS1–S	IJ1–S	I-STEM	
qaraʾ	*qaraʾa*	*qaraʾti*	*qaraʾū*	*taqraʾu*	*ʾaqraʾa*	*ʾaqraʾ*	*qraʾ*	S
faʿal	*faʿal-a*	*faʿal-ti*	*faʿal-ū*	*ta-fʿal-u*	*ʾa-fʿal-a*	*ʾa-fʿal*	*fʿal*	I
faʿal	*faʿal-a*	*faʿal-ti*	*faʿal-ū*	*ta-fʿal-u*	*ʾa-fʿal-a*	*ʾa-fʿal-*	*fʿal*	M
...	**...-*a***	**...-*ti***	**...-*ū***	***ta-...-u***	***ʾa-...-a***	***ʾa-...-***	...	
faʿā	*faʿā-a*	*faʿā-ti*	*faʿā-ū*	*ta-fā-u*	*ʾa-fā-a*	*ʾa-fā-*	*fā*	M
faʿā	*faʿā*	*faʿal-ti*	*faʿ-aw*	*ta-fā*	*ʾa-fā*	*ʾa-fa*	*fā*	I
raʾā	*raʾā*	*raʾayti*	*raʾaw*	*tarā*	*ʾarā*	*ʾara*	***rā***	S

Irregularity-Telugu

- Roots like telus- in Telugu inflect differently in the past.
- Examples

- The verb un- has two complementary 1) un- and 2) undu
- Examples

Ambiguity

- Words forms that look the same but have distinct functions or meaning are called homonyms. (Example: kind, ring, right, rose)
- Ambiguity is present in all aspects of morphological processing and language processing at large.
- Morphological parsing is not concerned with complete disambiguation of words in their context, however; it can effectively restrict the set of valid interpretations of a given word form.

Ambiguity

- Example in Korean:

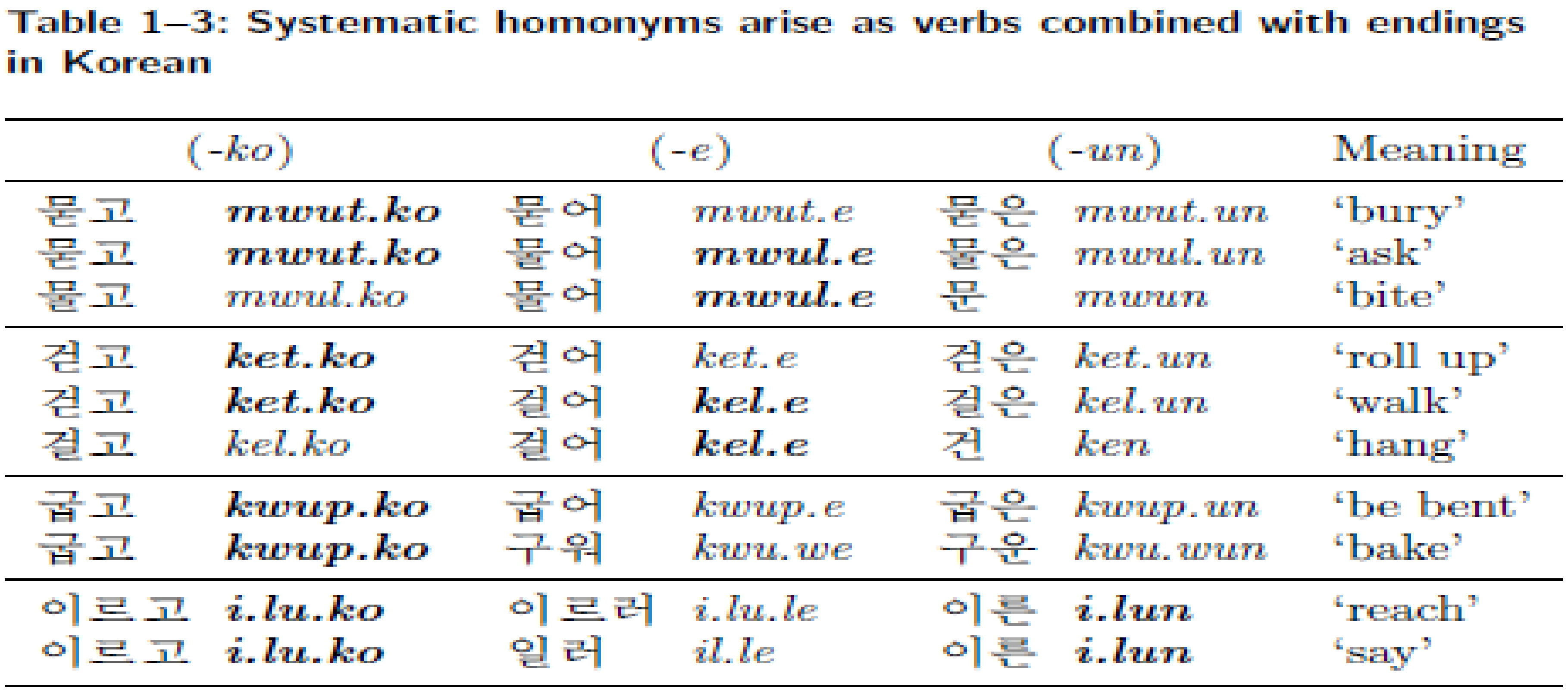


Table 1–3: Systematic homonyms arise as verbs combined with endings in Korean

(*-ko*)		(*-e*)		(*-un*)		Meaning
묻고	***mwut.ko***	묻어	*mwut.e*	묻은	*mwut.un*	'bury'
묻고	***mwut.ko***	물어	***mwul.e***	물은	*mwul.un*	'ask'
물고	*mwul.ko*	물어	***mwul.e***	문	*mwun*	'bite'
걷고	***ket.ko***	걷어	*ket.e*	걷은	*ket.un*	'roll up'
걷고	***ket.ko***	걸어	***kel.e***	걸은	*kel.un*	'walk'
걸고	*kel.ko*	걸어	***kel.e***	건	*ken*	'hang'
굽고	***kwup.ko***	굽어	*kwup.e*	굽은	*kwup.un*	'be bent'
굽고	***kwup.ko***	구워	*kwu.we*	구운	*kwu.wun*	'bake'
이르고	***i.lu.ko***	이르러	*i.lu.le*	이른	***i.lun***	'reach'
이르고	***i.lu.ko***	일러	*il.le*	이른	***i.lun***	'say'

Ambiguity

- The morphological disambiguation of a few languages like (Arabic) encompass not only the resolution of the structural components of words and their actual morpho-syntactic properties but also tokenization and normalization etc.
- Inverting sandhi during tokenization can provide multiple solutions to the problem of ambiguity in Indian languages. (na asatah vidyate bhavah which means the unreal has no existence)

Ambiguity

- Example in Czech:

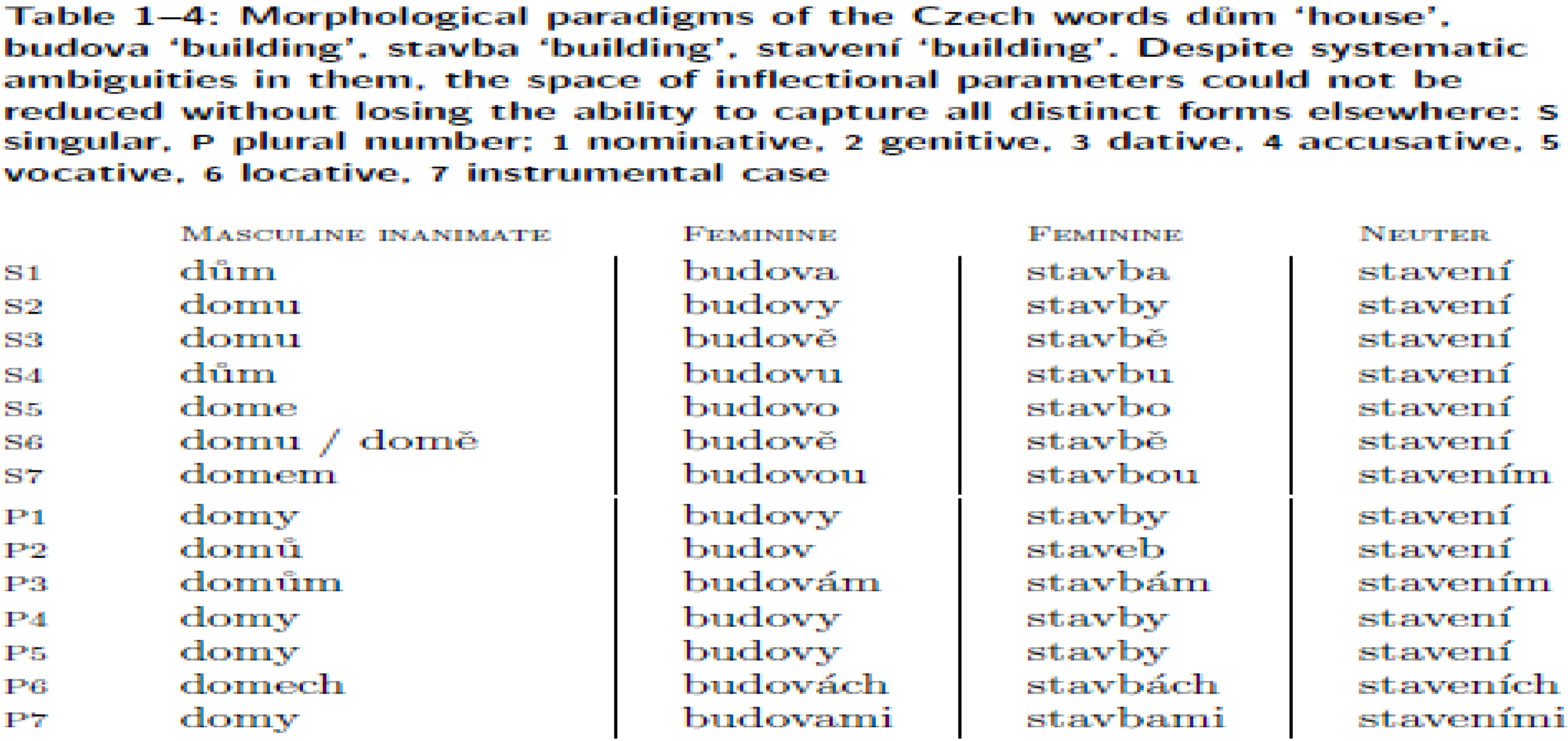


Table 1–4: Morphological paradigms of the Czech words dům 'house', budova 'building', stavba 'building', stavení 'building'. Despite systematic ambiguities in them, the space of inflectional parameters could not be reduced without losing the ability to capture all distinct forms elsewhere: S singular, P plural number; 1 nominative, 2 genitive, 3 dative, 4 accusative, 5 vocative, 6 locative, 7 instrumental case

	MASCULINE INANIMATE	FEMININE	FEMININE	NEUTER
S1	dům	budova	stavba	stavení
S2	domu	budovy	stavby	stavení
S3	domu	budově	stavbě	stavení
S4	dům	budovu	stavbu	stavení
S5	dome	budovo	stavbo	stavení
S6	domu / domě	budově	stavbě	stavení
S7	domem	budovou	stavbou	stavením
P1	domy	budovy	stavby	stavení
P2	domů	budov	staveb	stavení
P3	domům	budovám	stavbám	stavením
P4	domy	budovy	stavby	stavení
P5	domy	budovy	stavby	stavení
P6	domech	budovách	stavbách	staveních
P7	domy	budovami	stavbami	staveními

Ambiguity in Telugu

- Examples:

Ambiguity

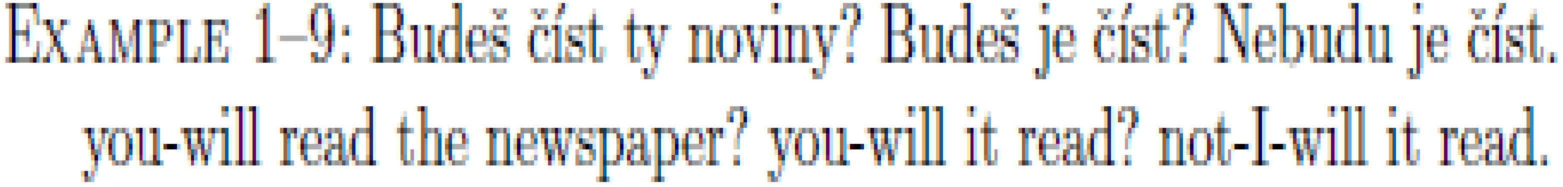
- The morphological phenomenon that some words or word classes show instances of systematic homonymy is called syncretism.
- In particular, homonymy can occur due to **neutralization** and **un-inflectedness** with respect to some morpho-syntactic parameters.
- **neutralization** is about syntactic irrelevance as reflected in morphology
- **uninflectedness** is about morphology being unresponsive to a feature that is syntactically relevant.
- In English the gender category is syntactically neutralized in the case of pronouns.
- The difference between he and she, him and her are only semantic.

Productivity

- An important question to be answered- Is the inventory of words in a language finite or infinite?
- In one view, language can be seen as simply a collection of utterances actually pronounced or written.
- This data set can be the linguistic corpora, a finite collection of linguistic data.
- If we consider language as a system, we discover structural devices like recursion (great-great), iteration or compounding (in-side) that allow to produce an infinite set of concrete linguistic utterances.
- This process is called as morphological productivity.

Productivity

- The members of the corpus are the word types.
- The original instances of the word form is the word token.
- The distribution of words or other elements of language follow the "80/20" rule also know as the law of the vital few.
- The negation is a productive morphological operation in some languages. Examples of English are dis-, non- , un-
- Example in Indian Languages a-samardh
- Example in Czech:



EXAMPLE 1–9: Budeš číst ty noviny? Budeš je číst? Nebudu je číst.
you-will read the newspaper? you-will it read? not-I-will it read.

Productivity

- Let us look at an example where creativity, productivity and the issue of unknown words meet nicely.
- According to Wikipedia the word ‘googol’ is a made-up word denoting a number “one followed by hundred zeros”.
- The name of the company Google is actually a misspelled word.
- Now both of these words entered the English lexicon.

Morphological Models

- Dictionary Lookup
- Finite-State Morphology
- Unification-Based Morphology
- Functional Morphology
- Morphology Induction

Morphological Models-Introduction

- Morphological parsing is a process by which word forms of a language are associated with corresponding linguistic descriptions.
- There are many approaches to designing and implementing morphological models.
- A lot of domain specific programming languages have been created that can be very useful in implementing theoretical problems with minimal programming effort.
- There are also a few approaches that do not resort to the domain specific programming.
- We now discuss a few prominent types of computational approaches to morphology.

Dictionary Lookup

- A dictionary is a data structure that directly enables obtaining precomputed word analysis.
- Dictionaries can be implemented as lists, binary search trees, tries, hash tables etc.
- Dictionaries enumerate the set of associations between word forms and their descriptions.
- The generative power of the language is not exploited when implemented in the form of a dictionary.
- The problem with dictionary based approach is how the associated annotations are constructed and how informative and accurate they are.

Finite-State Morphology

- Finite-State morphological models are directly compiled into finite-state transducers.
- Examples of finite-state transducers are Xerox Finite-state tool (XFST) and LEX tool.
- The set of possible sequences accepted by the transducer defines the input language and the set of possible sequences emitted by the transducer defines the output language.
- In finite state computational morphology the input word forms are referred to as **surface strings** and the output strings are referred to as **lexical strings**. (the finite string children can be converted to lexical string child [+ plural])

Finite-State Morphology

- Let us have a relation R, and let us denote it by [∑], the set of all sequences over some set of symbols ∑ so that the domain and range of R are subsets of [∑].
- Now R is a function mapping input string to a set of output strings.

  R: [∑]→{[∑]} which can be written as

  R: string→{string}
- Morphological operations and processes in human languages can be expressed in finite-state terms.
- A theoretical limitation of finite state models of morphology is the problem of reduplication of words found in many natural languages.

Finite-State Morphology

- Finite-state technology can be applied to the morphological modeling of isolating and agglutinative languages very easily.
- Finite-state tools can be used to a limited extent in morphological analyzers or generators.

Unification-Based Morphology

- Unification based approaches to morphology are inspired by two things:
  - The formal linguistic frameworks like head-driven phase structure grammar(HPSG).
  - Languages for lexical knowledge representation like (DATR)
- The concepts and methodologies of these formalisms are closely connected to logic programming. (Prolog)
- In higher level approaches linguistic information is expressed by more appropriate data structures that can include complex values unlike finite state models.

Unification-Based Morphology

- Morphological parsing P associates linear forms Ø with alternatives of structured content ψ

  P: Ø→ {ψ}

  P:form→{content}

- For morphological modelling word forms are best captured by regular expressions, while the linguistic content is best described through typed **feature structures** (can be viewed as directed acyclic graphs).
- Unification is the key operation by which feature structures can be merged into a more informative feature structure.

Unification-Based Morphology

- Morphological models of this kind are typically formulated as logic programs and unification is used to solve the system of constraints imposed by the model.

Functional-Morphology

- Functional morphology defines its models using principles of functional programming and type theory.
- It treats morphological operations and processes as pure mathematical functions and organizes the linguistic as well as abstract elements of a model into distinct types of values and type classes.
- Functional morphology is not limited to modeling particular types of morphologies in human languages but is useful for fusional morphology.

Functional Morphology

- Functional Morphology can be used for the implementation of:
- Morphological parsing
- Morphological generation
- Lexicon browsing etc
- Along with parsing described in the previous section we can also describe inflection I, derivation D and lookup L as functions of these generic types:

I: lexeme→{parameter}→{form}

D: lexeme→{parameter}→{lexeme}

L : content→{lexeme}

Morphology Induction

- Until now the focus is on finding the structure of words in diverse languages supposing we know what we are looking for.
- We now consider the problem of discovering and inducing word structure without the human insight.(unsupervised or semi-supervised).
- Automated acquisition of morphological and lexical information, even if not perfect, can be reused for bootstrapping and improving the classical morphological models, too.
- There are several challenging issues about deducing the word structure just from the forms and their context.

Finding the Structure of Documents

- Introduction
- Methods
- Complexity of the Approaches
- Performances of the Approaches

Finding the Structure of Documents- Introduction

- Sentence Boundary Detection
- Topic Boundary detection

Finding the Structure of Documents-Introduction

- As we all know words form sentences.
- Sentences can be related to each other by explicit discourse connectives such as **therefore**.
- Sentences form paragraphs.
- Paragraphs are self contained units of discourse about a particular point or idea.
- Automatic extraction of structure of documents help in:
  - Parsing
  - Machine Translation
  - Semantic Role Labelling

Finding the Structure of Documents- Introduction

- Sentence boundary annotation is important for human readability of the output of the **automatic speech recognition** (ASR) system.
- Chunking the input text or speech into topically coherent blocks provides better organization and indexing of the data.
- For simplicity we consider sentence and group of sentences related to a topic as the structure elements.
- Here we discuss about two topics:
  - **Sentence boundary detection:** the task of deciding where sentences start and end given a sequence of characters.
  - **Topic Segmentation:** the task of determining when a topic starts and ends in a sequence of sentences.

Finding the Structure of Documents-Introduction

- Here we discuss about statistical classification approaches which base their predictions on features of the input.
- Features of the input are local characteristics that give evidence toward the presence or absence of a sentence or a topic boundary such as:
    - Punctuation marks
    - A pause in a speech
    - A new word in a document
- Careful design and selection of features is required in order to be successful and prevent overfitting and noise problems.

Finding the Structure of Documents-Introduction

- The statistical approaches we discuss here are language independent but every language is a challenge in itself. For example:
    - In Chinese documents words are not separated by spaces.
    - In morphologically rich languages word structure may be analyzed to extract additional features.
- Such processing is usually done as a preprocessing step.

Sentence Boundary Detection

- Sentence boundary detection deals with automatically segmenting a sequence of word tokens into sentence units.
- In written text in English and a few languages the beginning of a sentence usually marked with an upper case letter and the end of a sentence is marked with:
- a period(.),a question mark(?) and an exclamation mark(!)
- In addition to the role as sentence boundary markers:
  - **Capitalized letters**- distinguish proper nouns
  - **Periods**- used in abbreviations and numbers
  - **Other punctuation marks**- used inside proper names

Sentence Boundary Detection

- Dr. can be an abbreviation for doctor or drive.
- Examples (from Brown corpus-10% of the periods are abbreviations):
  - I spoke with Dr. Smith.
  - My house is on Mountain Dr.
- Examples (from Wall Street Journal Corpus-47% of the periods are abbreviations):
- “This year has been difficult for both Hertz and Avis,” said Charles Finnie, carrental industry analyst-yes, there is such a profession-at Alex. Brown & sons.

Sentence Boundary Detection

- An automatic method might cut the previous sentences incorrectly.
- If the preceding sentence is spoken then prosodic cues(accent, stress, rhythm, tone, pitch and intonation) mark the structure.
- One more problem of sentence segmentation in written text is spontaneously written texts (SMS and IM) have poorly used punctuation.
- If the input text comes from an automatic system (OCR or ASR) with the aim to translate images of handwritten, typewritten or printed text or spoken utterances then finding of the system boundaries must handle the errors of those systems as well.

Sentence Boundary Detection

- OCR systems confuse periods and commas and can result in meaningless sentences.
- ASR transcripts lack punctuation marks and are mono-case.
- Human participants when tried to re-punctuate mono-case texts performed at an F1-measure of about 80% which shows how difficult the task is.
- Let us look at the utterance “okay no problem” in a conversation.
- It is not clear whether there is a single sentence or two in the above utterance.
- This problem is redefined for the conversational domain as the task of dialogue act segmentation.

Sentence Boundary Detection

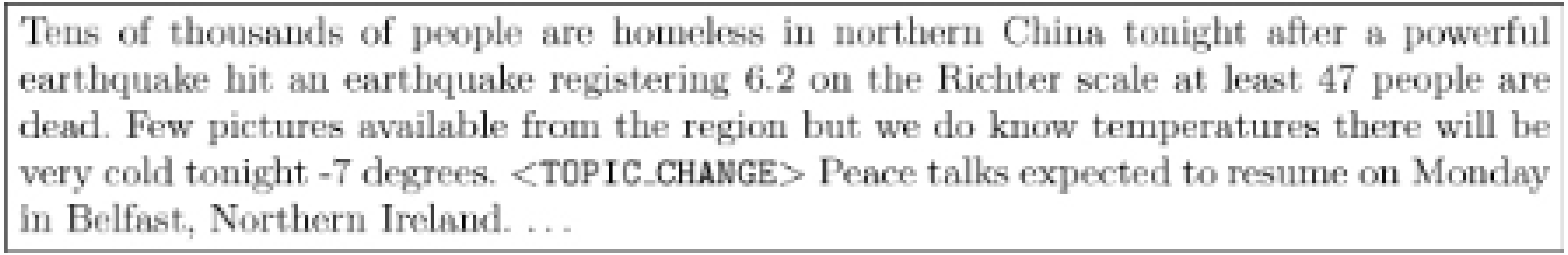
- Dialogue acts are better defined for conversational speech using a number of markup standards.
- Dialogue Act Markup in Several Layers (DAMSL) and Meeting Recorder Dialogue Act (MRDA) are two examples of such markup standards.
- According to these standards the utterance “okay no problem” consists of two sentential units (dialogue act units) okay and no problem.
- In the sentence “I think so but you should also ask him”, according to the segmentation standards there are two dialogue act tags.

Sentence Boundary Detection

- Code switch(sentences from multiple languages by multilingual speakers) is another problem that can affect the characteristics of sentences.
- Code switch also affects technical texts for which the punctuation signs can be redefined.
- Conventional rule-based sentence segmentation systems in well-formed texts rely on patterns to identify potential ends of sentences and lists of abbreviations for disambiguating them.
- Sentence segmentation can be stated as classification problem.

Topic Boundary Detection

- Topic segmentation is the task of automatically dividing a stream of text or speech into topically homogeneous blocks.
- Example:



Tens of thousands of people are homeless in northern China tonight after a powerful earthquake hit an earthquake registering 6.2 on the Richter scale at least 47 people are dead. Few pictures available from the region but we do know temperatures there will be very cold tonight -7 degrees. <TOPIC_CHANGE> Peace talks expected to resume on Monday in Belfast, Northern Ireland. . . .

Figure 2–1. Example of a topic boundary in a news article

Topic Boundary Detection

- Topic segmentation is an important task for applications like:
  - Information extraction and retrieval
  - Text Summarization
- In the 1990s Defense Advanced Research Projects Agency(DARPA) initiated the topic detection and Tracking (TDT) program.
- The objective was to segment a news stream into individual stories.
- Topic segmentation is a significant problem and requires a good definition of topic categories and their granularities.

Topic Boundary Detection

- Topics are not typically flat but occur in a semantic hierarchy.
- When a statement about soccer is followed by a statement about cricket should the annotator mark a topic change??
- It is difficult to segment the text into a predefined number of topics.
- In the case of TDT corpus high inter-annotator agreement (Cohen’s kappa value of 0.7 to 0.9) was achieved.
- In text, topic boundaries are usually marked with distinct segmentation cues like headlines and paragraph breaks.
- Speech provides other cues such as pause duration and speaker change.

Methods

- Generative Sequence Classification Methods
- Discriminative Local Classification Methods
- Discriminative Sequence Classification Methods
- Hybrid Approach
- Extensions for Global Modeling for Sentence Segmentation

Methods-Introduction

- Given a boundary candidate the goal is to predict whether or not the candidate is an actual boundary.
- Let x ϵ X be the vector of features associated with a candidate and y ϵ Y be the label predicted for that candidate.
- The label y can be b for boundary and b’ for non-boundary.
- Given a set of training examples $\{x,y\}_{train}$ we need to find a function that will assign the most accurate possible label y of unseen examples x_{unseen}.

Methods-Introduction

- Sentence segmentation in text can be framed as a three class problem: sentence boundary with an abbreviation b^{a}, without abbreviation $b^{a'}$ and abbreviation not at boundary b^{-a}.
- In spoken language a three way classification can be made between non-boundaries b^{-1}, statement boundaries b^{s} and question boundaries b^{q}.
- For sentence or topic segmentation the problem is defined as finding the most probable sentence or topic boundaries.

Methods-Introduction

- The classification can be done at each potential boundary I (**local modeling**) then the aim is to estimate the most probable boundary type y’ for each candidate example xi :

$$y_i’= \text{argmax}_{yi \text{ in } Y} P(y_i/x_i)$$

- Here y_i’ is the estimated category and y_i is the possible categories.
- If we look at the candidate boundaries as a **sequence** then

$Y’=y_1’,\ldots y_n’$ that have

The maximum probability given the candidate examples $X=x_1,\ldots.,x_n$

$$Y’=\text{argmax}_Y P(Y/X)$$

Methods-Introduction

- The methods used are categorized into:
  - Local
  - Sequence
- Another categorization is done according to the type of the machine learning algorithm:
  - Generative
  - Discriminative
- Generative sequence models estimate the joint distribution of the observations, P(X,Y) and the labels which requires specific assumptions and have good generalization properties.
- Discriminative sequence models focus on features that characterize the differences between labeling of the examples.

Methods-Introduction

- These methods can be used for both sentence and topic segmentation in both text and speech.
- The only problem is that in the case of text if end-of-sentence delimiters are not included it will be difficult to categorize.

Generative Sequence Classification Methods

- The most commonly used generative sequence classification method for topic and sentence segmentation is the Hidden Markov Model(HMM).

$$\hat{Y} = \underset{Y}{\operatorname{argmax}}\, P(Y|X) = \underset{Y}{\operatorname{argmax}}\, \frac{P(X|Y)P(Y)}{P(X)} = \underset{Y}{\operatorname{argmax}}\, P(X|Y)P(Y)$$

- P(X) is dropped because it is the same for different Y.
- P(X/Y) and P(Y) can be estimated as:

$$P(X|Y) = \prod_{i=1}^{n} P(\mathbf{x}_i|y_1, \ldots, y_i)$$

$$P(Y) = \prod_{i=1}^{n} P(y_i|y_1, \ldots y_{i-1})$$

Generative Sequence Classification Methods

- We assume that:

$$P(\mathbf{x}_i|y_1, \ldots, y_i) \approx P(\mathbf{x}_i|y_i)$$

- A bigram model can be assumed for modeling output categories:

$$P(y_i|y_1, \ldots, y_{i-1}) \approx P(y_i|y_{i-1})$$

- The bigram case is modeled by a fully connected m-state Markov model, where m is the number of boundary categories.
- The states emit words for sentence (topic) segmentation, and the state sequence that most likely generated the word sequence is estimated.

Generative Sequence Classification Methods

- State transition probabilities P(y_i/y_{i-1}), and state observation likelihoods, P(x_i/y_i), are estimated using the training data.
- The most probable boundary sequence is obtained by dynamic programming.
- Example:

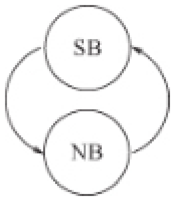
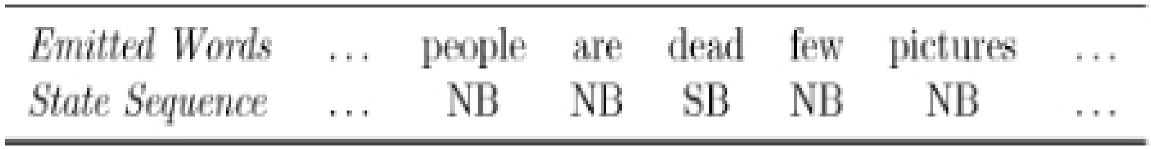


Figure 2–2. Conceptual hidden Markov model for segmentation with two states:

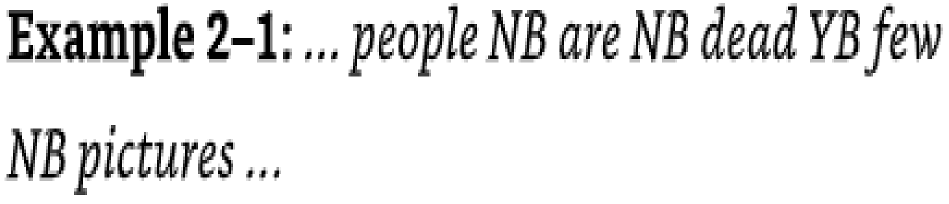
Table 2–1. Sentence segmentation with simple two-state Markov model



Emitted Words	...	people	are	dead	few	pictures	...
State Sequence	...	NB	NB	SB	NB	NB	...

Generative Sequence Classification Methods

- For topic segmentation we can use n states where n is the number of topics.
- Obtaining state observation likelihoods without knowing the topic categories is the main challenge.
- An imaginary token is inserted between all consecutive words in case the word preceding the boundary is not part of a disfluency.
- Example:



Example 2–1: *... people NB are NB dead YB few NB pictures ...*

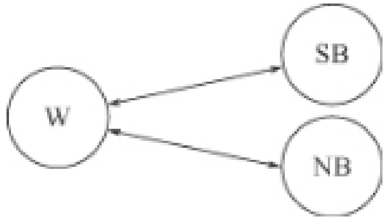


Figure 2–3. Conceptual hidden event language model for segmentation

Generative Sequence Classification Models

- The extra boundary tokens can be used to capture other meta information.
- The most commonly used meta information is the feedback obtained from other classifiers.
- For topic segmentation the same idea can be used to model topic start and topic final sections explicitly which help for broadcast news topic segmentation.
- One more extension is to capture not only words but also morphological syntactic and other information.

Discriminative Local Classification Methods

- These models aim to model P(y_i/x_i) directly.
- In generative approaches algorithms like naïve Bayes are used.
- In discriminative methods, discriminant functions of the feature space define the model.
- The machine learning algorithms used for discriminative classification approaches are:
    - Support Vector Machines
    - Boosting
    - Maximum Entropy
    - Regression

Discriminative Local Classification Methods

- Discriminative approaches have outperformed generative methods in many speech and language processing tasks.
- Training for these approaches require iterative optimization.
- In discriminative local classification each boundary is processed separately with local and contextual features.
- No global optimization (sentence or document wide) is performed unlike in sequence classification models.
- For sentence segmentation supervised learning methods have been applied.

Discriminative Local Classification Methods

- Transformation based learning (TBL) is used to infer rules in the supervised learning.
- The classifiers used for the task are:
  - Regression trees
  - Neural networks
  - C 4.5 classification tree
  - Maximum Entropy classifiers
  - Support vector machines

Discriminative Local Classification Methods

- A few techniques treated sentence segmentation problem as a subtask of POS tagging by assigning a tag to punctuation similar to other tokens.
- For tagging a combination of HMM and maximum entropy approaches have been used.
- For topic segmentation a method called as TextTiling method is used.
- It uses a lexical cohesion metric in a word vector space as an indicator of topic similarity.
- TextTiling can be seen as a local classification method with single feature of similarity.

Discriminative Local Classification Methods

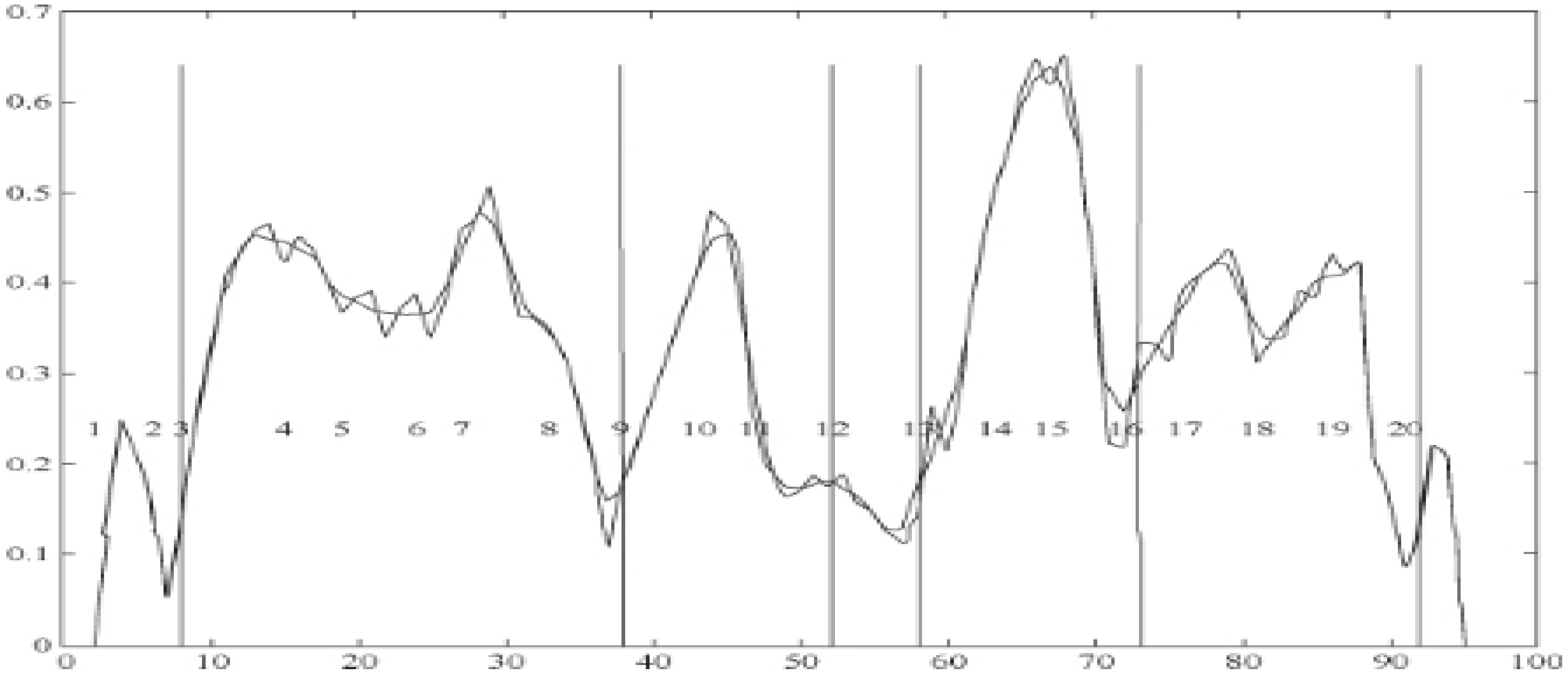


Figure 2–4. Text Tiling example (from [22])

Discriminative Local Classification Methods

- Two methods for computing the similarity score for topic segmentation were proposed:
    - Block Comparison
    - Vocabulary Introduction
- **Block comparison** compares adjacent blocks of text to see how similar they are according to how many words the adjacent blocks have in common.
- Given two blocks b_1 and b2, each having k tokens(sentences or paragraphs) the similarity score is computed by the formula:
- $W_{t,b}$ is the weight assigned to term t in block b.
- The weights may be binary or term frequency.

$$\frac{\sum_t w_{t,b_1} \cdot w_{t,b_2}}{\sqrt{\sum_t w_{t,b_1}^2 \sum_t w_{t,b_2}^2}}$$

Discriminative Local Classification Methods

- The **vocabulary introduction** method assigns a score to a token-sequence gap on the basis of how many new words are seen in the interval in which it is the mid point.
- Given blocks b1 and b2 of equal number of words, w, the topical cohesion score is computed with the following formula:

$$\frac{NumNewTerms(b_1) + NumNewTerms(b_2)}{2 \times w}$$

- Where NumNewTerms(b) is the number of terms in block b seen for the first time in text.
- This method can be extended to exploit latent semantic analysis.

Discriminative Sequence Classification Methods

- Discriminative sequence classification methods are in general extensions of local discriminative models.
- They have additional decoding stages that find the best assignment of labels by looking at neighboring decisions to label an example.
- Conditional Random Fields (CRFs) are a class of log-linear models for labelling structures.
- They model the conditional probability of a sequence of boundary labels (Y = $y_1, y_2, \ldots., y_n$) given the sequence of feature sets extracted from the context in which they occur.

$$P(Y|X) \sim \frac{1}{Z(X)} \exp\left(\sum_{t=1}^{n} \sum_{i=1}^{m} \lambda_i f_i(y_{t-1}, y_t, y_t)\right)$$

$$Z(X) = \sum_{Y} \exp\left(\sum_{t=1}^{n} \sum_{i=1}^{m} \lambda_i f_i(y_{t-1}, y_t, y_t)\right)$$

Discriminative Sequence Classification Methods

- Where $f_i(.)$ are feature functions of the observations and a clique of labels and λ_i are the corresponding weights.
- Z(.) is a normalization function dependent only on the observations.
- CRFs are trained by finding the λ parameters that maximize the likelihood of the training data, usually with a regularization term to avoid overfitting.

Hybrid Approaches

- Non-sequential discriminative classification algorithms ignore the context, which is critical for the segmentation task.
- We can add context as a feature or use CRFs which consider context.
- An alternative is to use a hybrid classification approach.
- In this approach the posterior probabilities, $P_c(y_i/x_i)$ for each candidate obtained from the classifiers such as boosting or CRF are used.
- The posterior probabilities are converted to state observation likelihoods by dividing to their priors using the Bayes rule.

$$\underset{y_i}{\operatorname{argmax}} \frac{P_c(y_i|\mathbf{x}_i)}{P(y_i)} = \underset{y_i}{\operatorname{argmax}} P(\mathbf{x}_i|y_i)$$

Extensions for Global Modeling for Sentence Segmentation

- Most approaches to sentence segmentation have focused on recognizing boundaries rather than sentences in themselves.
- This happened because of the number of sentence hypotheses that must be assessed in comparison to the number of boundaries.
- One approach is to segment the input according to likely sentence boundaries established by a local model.
- Train a re-ranker on the n-best lists of segmentations.
- This approach allows leveraging of sentence-level features such as scores from a syntactic parser or global prosodic features.

Complexity of Approaches

- All the approaches have advantages and disadvantages.
- These approaches can be rated in terms of:
  - Training and prediction algorithms
  - Performance on real world data set
- Training of discriminative approaches is more complex than training of generative ones because they require multiple passes over the training data to adjust for their feature weights.

Complexity of Approaches

- Generative models such as HELMS can handle multiple orders of magnitude larger training sets and benefit from old transcripts.
- They work only with a few features and do not cope well with unseen events.
- Discriminative classifiers allow for a wider variety of features and perform better on smaller training sets.
- Predicting with discriminative classifiers is slower because it is dominated by the cost of extracting more features.

Complexity of Approaches

- In comparison to local approaches sequence approaches have to handle additional complexity.
- They need to handle the complexity of finding the best sequence of decisions which requires evaluating all possible sequences of decisions.
- The assumption of conditional independence in generative sequence algorithms allow the use of dynamic programming to trade time for memory and decode in polynomial time.
- This complexity is measured in:
  - The number of boundary candidates processed together
  - The number of boundary states
- Discriminative sequence classifiers like CRFs need to repeatedly perform inference on the training data which might become expensive.

Performances of the Approaches

- For sentence segmentation in speech performance is evaluated using:
  - Error rate:- ratio of number of errors to the number of examples
  - F1 measure
  - National Institute of standards and Technology (NIST) error rate:- Number of candidates wrongly labeled divided by the number of actual boundaries.
- For sentence segmentation in text the reports on the error rate on a subset of the wall street journal corpus of about 27000 sentences are as follows:
  - A typical rule-based system performs at an error rate of 1.41%
  - An addition of abbreviation list lowers the error rate to 0.45%
  - Combining it with a supervised classifier using POS tag features lead to an error rate of 0.31%
  - An SVM based system obtains an error rate of 0.25%

Performances of the Approaches

- Even though the error rates seem to be very low they might effect activities like extractive summarization which depend on sentence segmentation.
- For sentence segmentation in speech reports on the Mandarin TDT4 Multilingual Broadcast News Speech Corpus are as follows:
  - F1 measure of 69.1% for a MaxEnt classifier, 72.6% with Adaboost, and 72.7% with SVM using the same features.
- On a Turkish Broadcast News Corpus reports are as follows:
  - F1 measure of 78.2% with HELM 86.2% with fHELM with morphology features 86.9% with Adaboost and 89.1% with CRFs.
- Reports show that on the English TDT4 broadcast news corpus Adaboost combined with HELM performs at an F1 measure 67.3%.