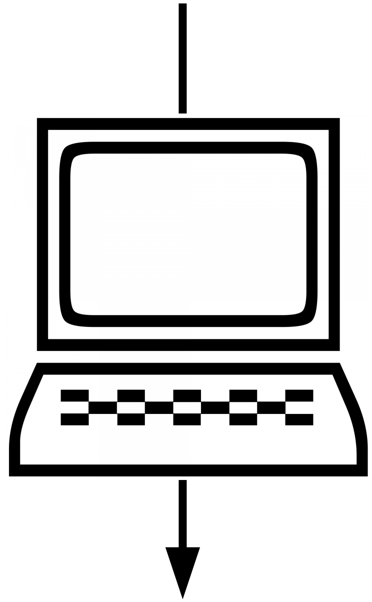
Machine Translation and Sequence-to-sequence Models

Graham Neubig

Carnegie Mellon University
CS 11-731

What is Machine Translation?

kare wa ringo wo tabeta .



He ate an apple .

What are Sequence-to-sequence Models?

Sequence-to-sequence Models

Machine translation:

kare wa ringo wo tabeta → he ate an apple

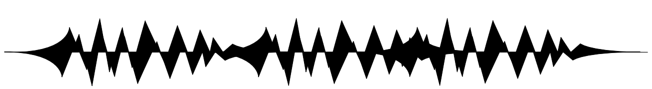
Tagging:

he ate an apple → PRN VBD DET PP

Dialog:

he ate an apple → good, he needs to slim down

Speech Recognition:

 → he ate an apple

And just about anything...:

1010000111101 → 00011010001101

Why MT as a Representative?

Useful!

KUSHINIKIZA! Google Translate SAVES BABY in Irish roadside birth

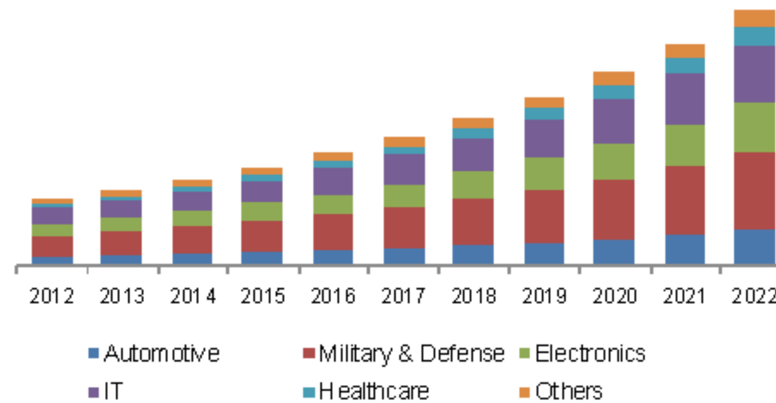
Do no evil? We literally save lives now

13 Feb 2015 at 12:01, John Leyden 8

Quick-thinking Irish paramedics turned to Google Translate to communicate with a pregnant woman who spoke Swahili, allowing her to safely give birth.

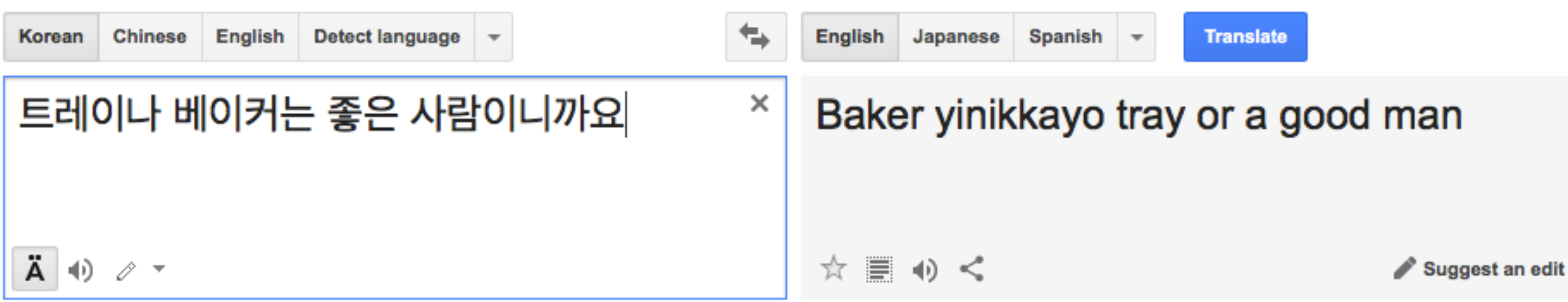
Source: The Register

Global MT Market Expected To Reach $983.3 Million by 2022



Source: Grand View Research

Imperfect...



MT and Machine Learning

Big Data! Billions of words for major languages
… but little for others

Well-defined, Difficult Problem!

Use for algorithms, math, etc.

Algorithms Widely Applicable!

MT and Linguistics

트레이나 베이커는 좋은 사람이니까요

Baker yinikkayo tray or a good man

Trina Baker is a good person

Morphology! 이니까요 is a variant of 이다 (to be)

Syntax! should keep subject together

Semantics! "Trina" is probably not a man...

… and so much more!

Class Organization

Class Format

- **Before class:**
  - Read the assigned material
  - Ask questions via web (piazza/email)
- **In class:**
  - Take a small quiz about material
  - Discussion/questions
  - Pseudo-code walk
  - Programming (TAs/Instructor will supervise)

Assignments

- **Assignment 1:** Create a neural sequence-to-sequence modeling system. Turn in code to run it, and a short 1-2 page report.
- **Assignment 2:** Create a symbolic sequence-to-sequence modeling system. Similarly turn in code/report.
- **Final project:** Come up with an interesting new idea and test it.

Assignment Instructions

- Bring your computer to every class and make a Github account.
- We recommend you implement in the following libraries:
  - DyNet: for neural networks (C++ or Python)
  - OpenFST: for transducers, if you use them (C++)
  - pyfst: for transducers in Python
- It is OK to work in small groups up to 3, particularly for the final project. If you do so, please use a shared git repository and commit the code that you write, and in reports note who did what part of the project.

Class Grading

- **Short quizzes:** 20%
- **Assignment 1:** 20%
- **Assignment 2:** 20%
- **Final Project:** 40%

Class Plan

1. Introduction (Today): 1 class
2. Language Models: 4 classes
3. Neural MT: 2 classes
3. Evaluation/Data: 2 classes — #1 Due
4. Symbolic MT: 4 classes
5. Algorithms: 2 classes — #2 Due
7. Advanced Topics: 11 classes
8. Final Project Discussion: 2 classes — Project Due

Guest Lectures

- Bob Frederking:
  Knowledge-based Translation
- LP Morency:
  Something Multi-modal

  (Date TBD)

Statistical Machine Translation

Statistical Machine Translation

F = *kare wa ringo wo tabeta .*

E = He ate an apple .

Probability model: $P(E|F;\Theta)$

Parameters

Problems in SMT

- **Modeling:** How do we define $P(E|F;\Theta)$?
- **Learning:** How do we learn Θ?
- **Search:** Given F, how do we find the highest scoring translation?

$$E' = \operatorname{argmax}_E P(E|F;\Theta)$$

- **Evaluation:** Given E' and a human reference E, how do we determine how good E' is?

Part 1: Neural Models

Language Models 1: n-gram Language Models

Given multiple candidates, which is most likely as an English sentence?

E_1 = he ate an apple

E_2 = he ate an apples

E_3 = he insulted an apple

E_4 = preliminary orange orange

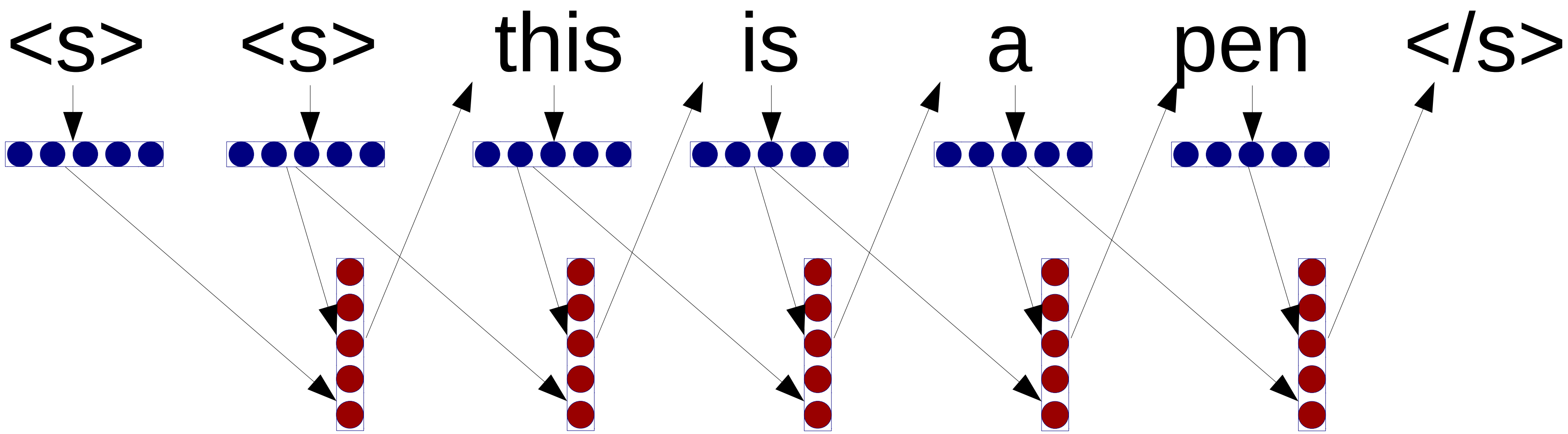
- Definition of language modeling
- Count-based n-gram language models
- Evaluating language models
- **Implement:** n-gram language model

Language Models 2: Log-linear Language Models

$$\begin{array}{l} \text{a} \\ \text{the} \\ \text{talk} \\ \text{gift} \\ \text{hat} \\ \ldots \end{array} \quad \boldsymbol{b} = \begin{pmatrix} 3.0 \\ 2.5 \\ -0.2 \\ 0.1 \\ 1.2 \\ \ldots \end{pmatrix} \quad \boldsymbol{w}_{1,a} = \begin{pmatrix} -6.0 \\ -5.1 \\ 0.2 \\ 0.1 \\ 0.6 \\ \ldots \end{pmatrix} \quad \boldsymbol{w}_{2,giving} = \begin{pmatrix} -0.2 \\ -0.3 \\ 1.0 \\ 2.0 \\ -1.2 \\ \ldots \end{pmatrix} \quad \boldsymbol{s} = \begin{pmatrix} -3.2 \\ -2.9 \\ 1.0 \\ 2.2 \\ 0.6 \\ \ldots \end{pmatrix}$$

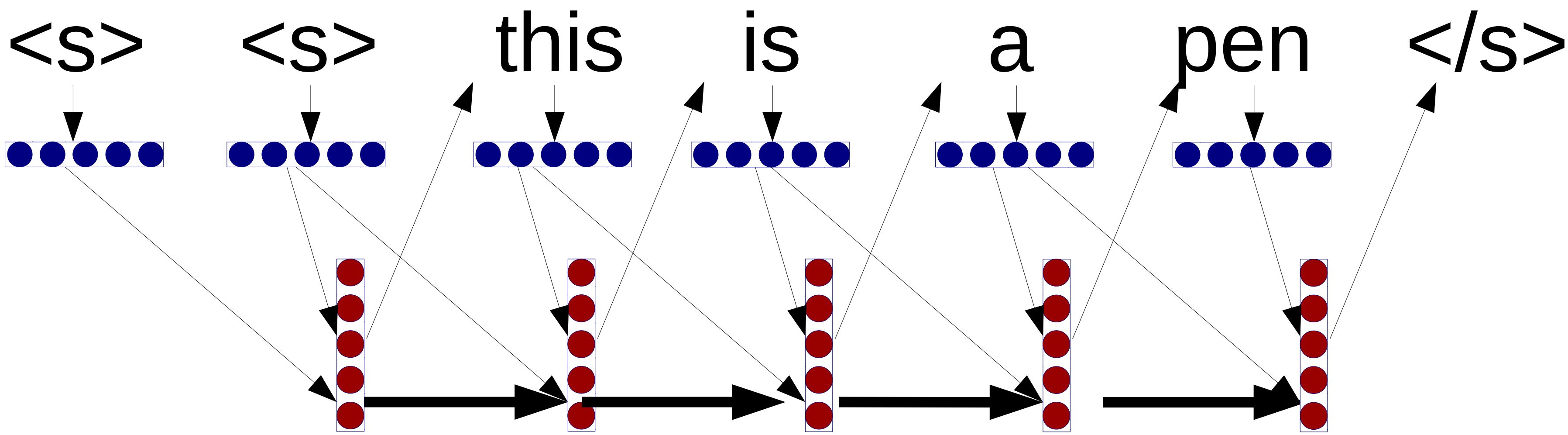
- Log-linear language models
- Stochastic gradient descent
- Features for language modeling
- **Implement:** Log-linear language model

Language Models 3: Neural Networks and Feed-forward LMs



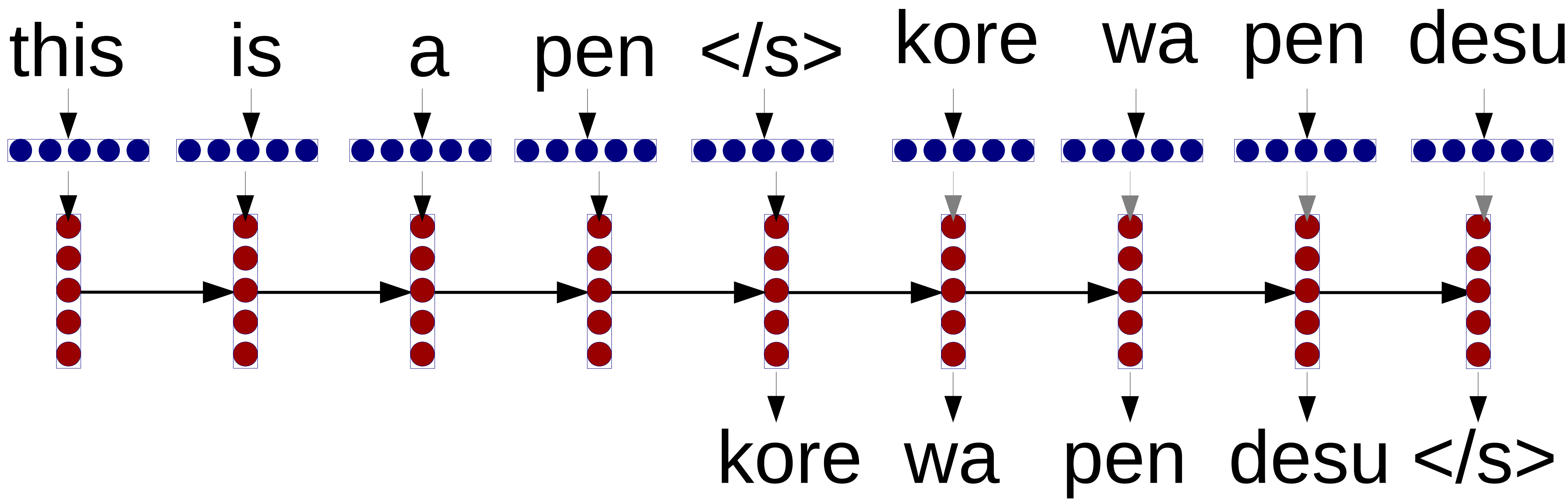
- Neural networks and back-propagation
- Feed-forward neural language models
- Mini-batch training
- **Implement:** Feed-forward LM

Language Models 4: Recurrent LMs



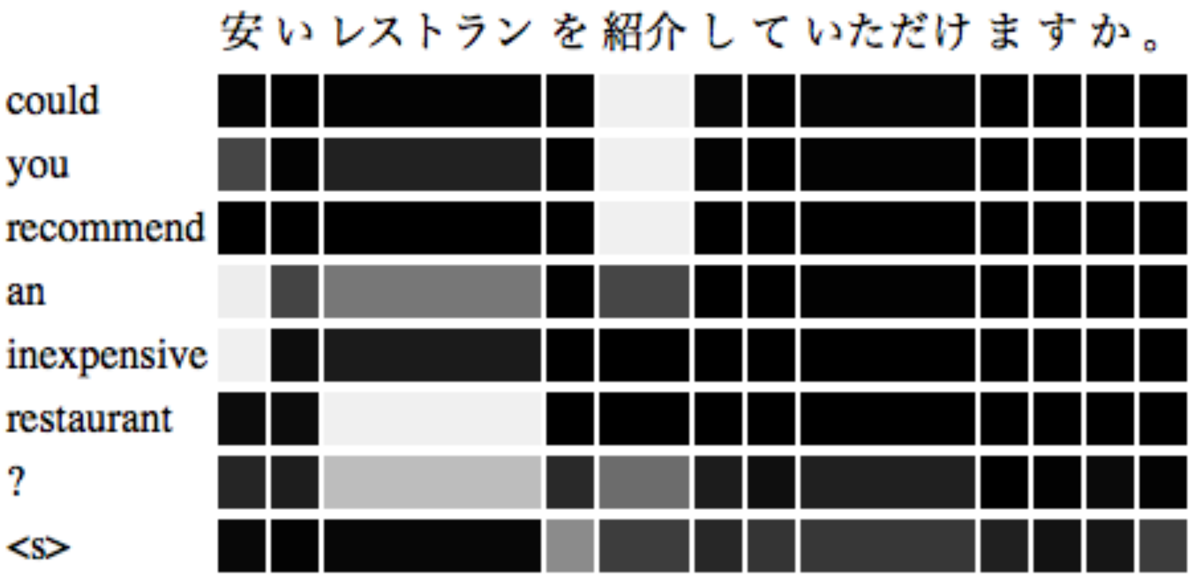
- Recurrent neural networks
- Vanishing Gradient and LSTMs/GRUs
- Regularization and dropout
- **Implement:** Recurrent neural network LM

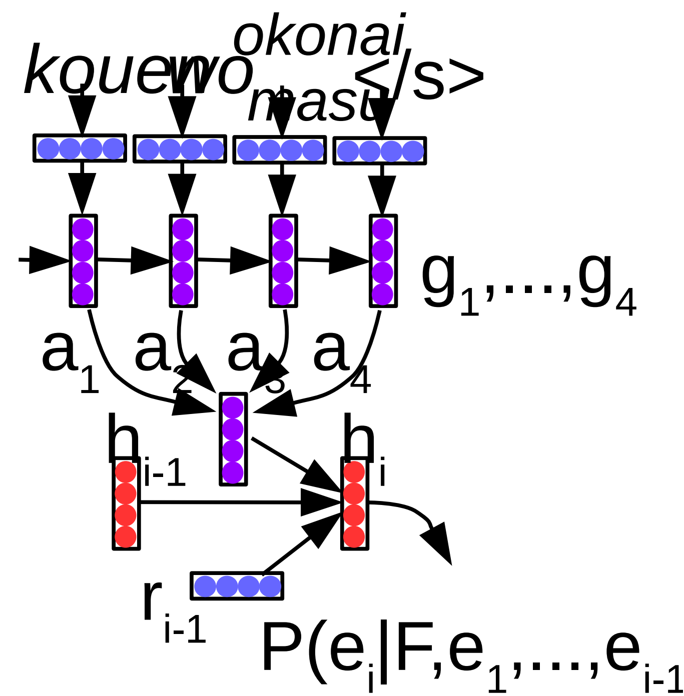
Neural MT 1: Encoder-decoder Models



- Encoder-decoder Models
- Searching for hypotheses
- Mini-batched training
- **Implement:** Encoder-decoder model

Neural MT 2: Attentional Models





- Attention in its various varieties
- Unknown word replacement
- Attention improvements, coverage models
- **Implement:** Attentional model

Data and Evaluation

Creating Data

- Preprocessing
- Document harvesting and crowdsourcing
- Other tasks: dialog, captioning
- **Implement:** Find/preprocess data

Evaluation

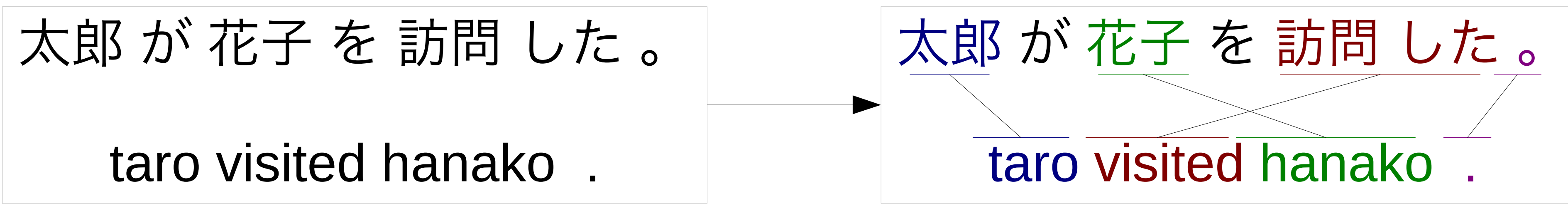
taro ga hanako wo otozureta

	Taro visited Hanako	the Taro visited the Hanako	Hanako visited Taro
Adequate?	○	○	X
Fluent?	○	X	○
Better?	B, C	C	

- Human evaluation
- Automatic evaluation
- Significance tests and meta-evaluation
- **Implement:** BLEU and measure correlation

Symbolic Translation Models

Symbolic Methods 1: IBM Models



- The IBM/HMM models
- The EM algorithm
- Finding word alignments
- **Implement:** Word alignment

Symbolic Methods 2: Monotonic Symbolic Models

he ate an apple

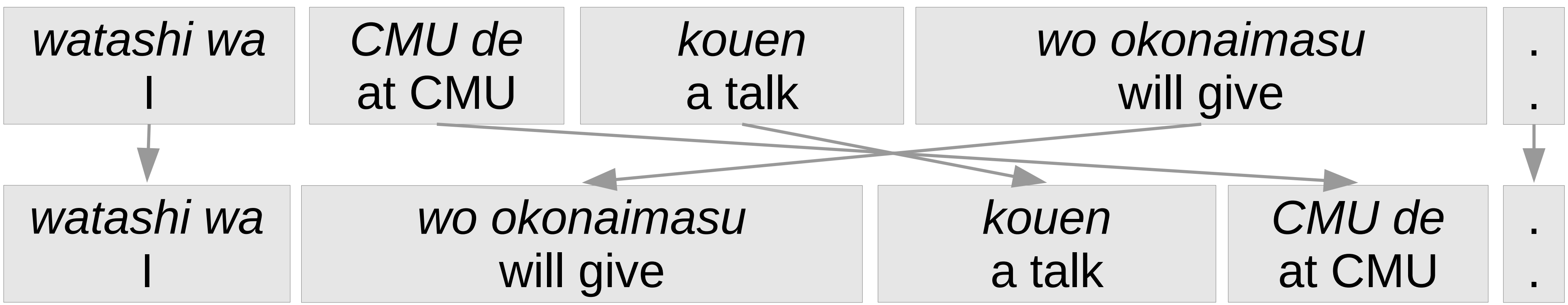
↓

PRN VBD DET PP

- Models for sequence transduction
- The Viterbi algorithm
- Weighted finite-state transducers
- **Implement:** A part-of-speech tagger

Symbolic Methods 3: Phrase-based MT

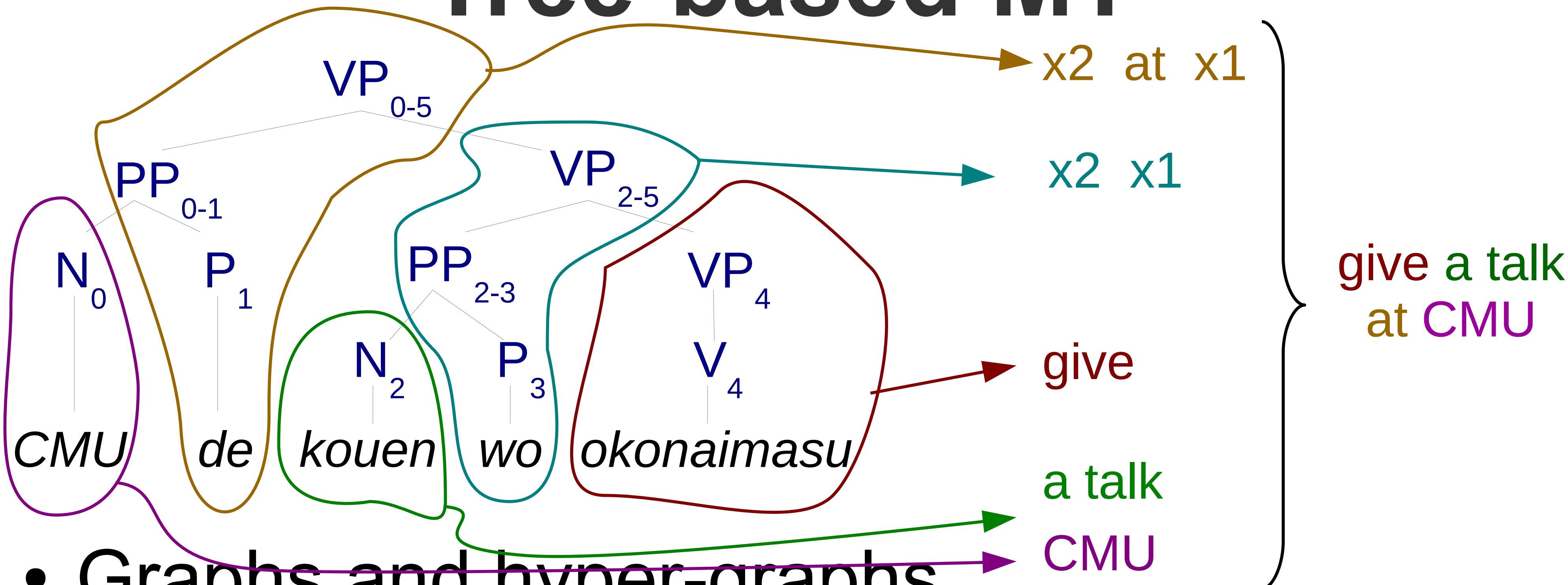
F = *watashi wa CMU de kouen wo okonaimasu .*



E = I will give a talk at CMU .

- Phrase extraction and scoring
- Reordering models
- Phrase-based decoding
- **Implement:** Phrase-based decoding

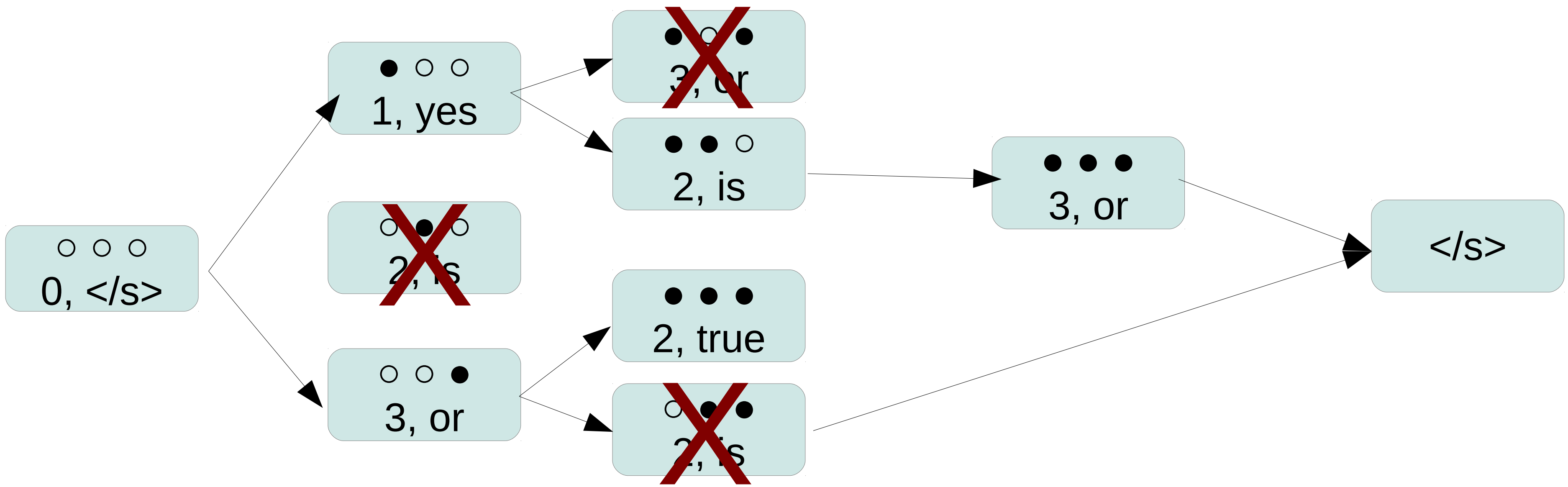
Symbolic Methods 4: Tree-based MT



- Graphs and hyper-graphs
- Synchronous context free grammars
- Tree substitution grammars
- **Implement:** Hierarchical phrase-based MT

Data and Evaluation

Algorithms 1: Search



- Beam search and cube pruning
- Hypothesis recombination
- Future costs, A* search
- **Implement:** Beam search

Algorithms 2: Parameter Optimization

Highest ○

	LM	TM	RM	
○ Taro visited Hanako	0.2*-4	0.3*-3	0.5*-1	-2.2
X the Taro visited the Hanako	0.2*-5	0.3*-4	0.5*-1	-2.7
X Hanako visited Taro	0.2*-2	0.3*-3	0.5*-2	-2.3

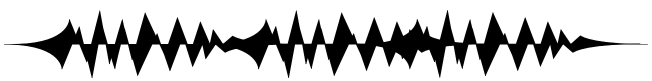
- Loss functions
- Deciding the hypothesis space
- Optimization criteria
- **Implement:** Optimization of NMT or PBMT

Advanced Topics

Other Sequence-to-sequence Tasks

he ate an apple → PRN VBD DET PP

he ate an apple → good, he needs to slim down

 → he ate an apple

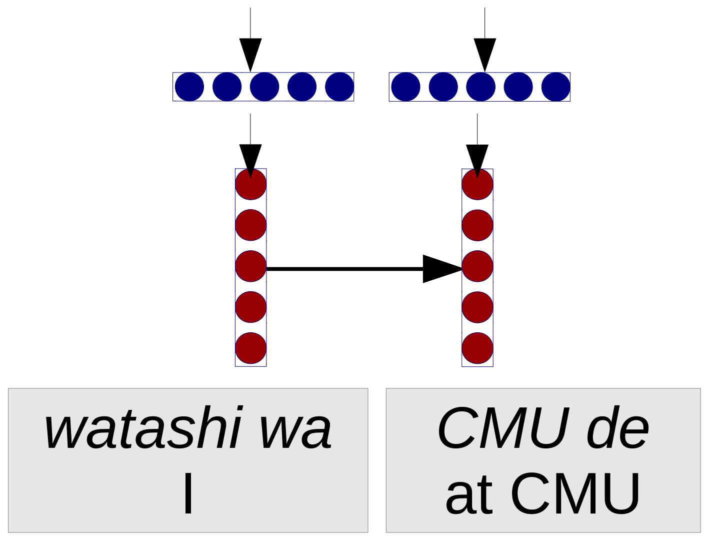
- Case studies about task-specific models
  - Consistency constraints in tagging
  - Diversity objectives in dialog
  - Dynamic programming in speech
- **Implement:** Download and run on datasets

Ensembling/System Combination

Model 1 + Model 2

- Ensembled decoding
- Post-hoc hypothesis combination
- Reranking
- **Implement:** Ensembled decoding

Hybrid Neural-symbolic Models



- Symbolic models with neural components
- Neural models with symbolic components
- **Implement:** Implement lexicons in NMT or neural feature functions

Multi-lingual and Multi-task Learning

hello

こんにちは

hola

- Learning for multiple tasks
- Learning for multiple languages
- **Implement:** Implement a multi-lingual neural system

Subword Models

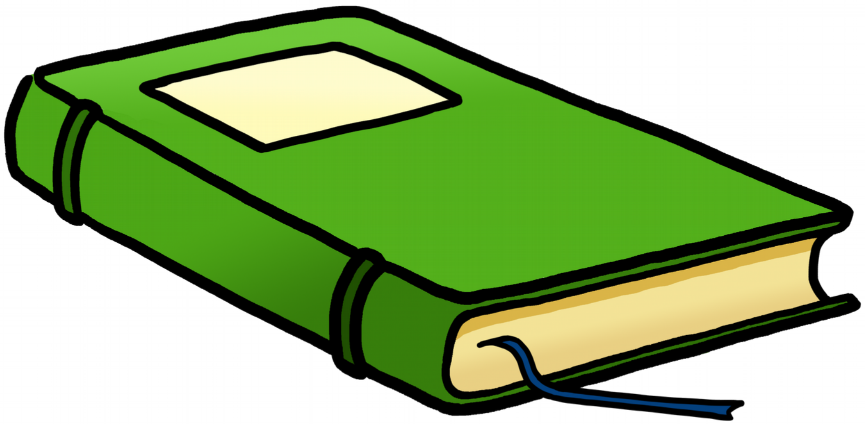
reconstructed

↓

re+ construct+ ed

- Character models
- Subword models
- Morphology models
- **Implement:** Implement subword splitting

Document Level Models



- Document level modeling
- Document level evaluation
- Stream decoding
- **Implement:** Document level measures

For Next Class

Homework

- Read n-gram language modeling materials
- Get software working on your machine (doing all at once may be more efficient?)
  - By Thursday 1/19: Python
  - By Tuesday 1/24: Numpy
  - By Thursday 1/26: DyNet