Interpretation Support System for Classification Patterns Using HMM in Deep Learning with Texts

- Masayuki Ando (The University of Shiga Prefecture, Japan)
  email: oh23mandou@ec.usp.ac.jp

Yoshinobu Kawahara (Kyushu University; RIKEN, Japan)
Wataru Sunayama (The University of Shiga Prefecture, Japan)
Yuji Hatanaka (Oita University , Japan)

滋賀県立大学
THE UNIVERSITY OF SHIGA PREFECTURE

IARIA

Short resume of the presenter

- Masayuki Ando

  Masayuki Ando graduated from the Department of Electronic Systems Engineering, Faculty of Engineering, University of Shiga Prefecture in 2017. In 2019, he completed the master's course at the same university. He is currently enrolled in the doctoral course of the same university.

  His research interests include deep learning, human-computer interaction, and text mining.

Introduction

- In recent years, deep learning has become popular and is used for text classification.

On the other hand, there are problems...

- Interpreting the content of learning results (learning networks) is difficult.

→ Classification criteria are not understood by humans.

→ Need to understand the rationale for deep learning decisions：XAI（Explainable AI）

Related research:
studies focusing on classification criteria

- Recently, research using a method called Attention [1][2] has become a hot topic.

These research estimate important features from the relationship between input and output, rather than from the content of the learning results.

↓

Learned NetworkIt is difficult to understand what rules have been learned.

Attention.
A method for learning by paying attention to some inputs that have a particularly strong relationship with the output (e.g., highlighting important parts in an image or words that contribute to the output in a sentence).

[1] A.Vaswani,N.Shazeer,N.Parmar,J.Uszkoreit,L.Jones,A.N. Gomez, L. Kaiser, and I. Polosukhin, “Attention Is All You Need,” CoRR, vol. abs/1706.03762, 2017
[2] Saumya Jetley, Nicholas A. Lord, Namhoon Lee, Philip H.S. Torr, “Learn to pay attention.” , arXiv preprint arXiv:1804.02391, 2018

Related research:
Focusing on the contents of the network

- Research the interpretation of networks by highlighting images that contribute to the output [3][4]
  Inferring the learning process of deep learning networks by representing intermediate layer nodes with images based on learning to classify images using deep learning.

Since the input was an image, the intermediate nodes could also be represented by images.

↓

If the input is text, for example, it cannot be represented by an image, and the intermediate nodes are difficult to interpret.

---

[3]J. Wagner, J. M. Kohler, T. Gindele, L. Hetzel, J. T. Wiedemer, S. Behnke, `Interpretable and fine-grained visual explanations for convolutional neural networks', Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 9097-9107, 2019.

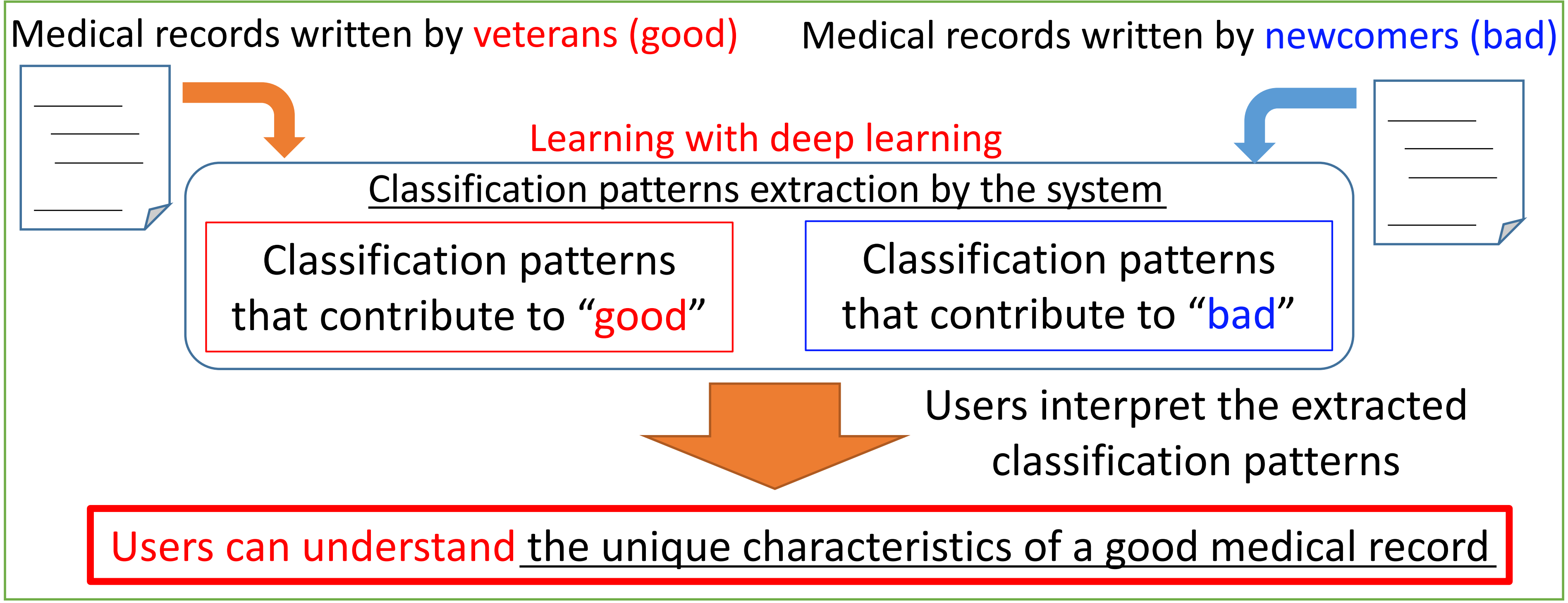
[4]C. Panigutti, A. Perotti, D. Pedreschi, `Doctor XAI: an ontology-based approach to black-box sequential data classification explanations', Proceedings of the 2020 Conference on Fairness Accountability and Transparency, pp.629-639, 2020.

Purpose

We will develop a system that extracts the contents of learning networks by deep learning as classification patterns and supports their interpretation

Classification patterns: features and combinations of features that contribute to classification

Concrete example: A system obtains classification criteria for good electronic medical records from a learning network



Medical records written by veterans (good)

Medical records written by newcomers (bad)

Learning with deep learning

Classification patterns extraction by the system

Classification patterns that contribute to "good"

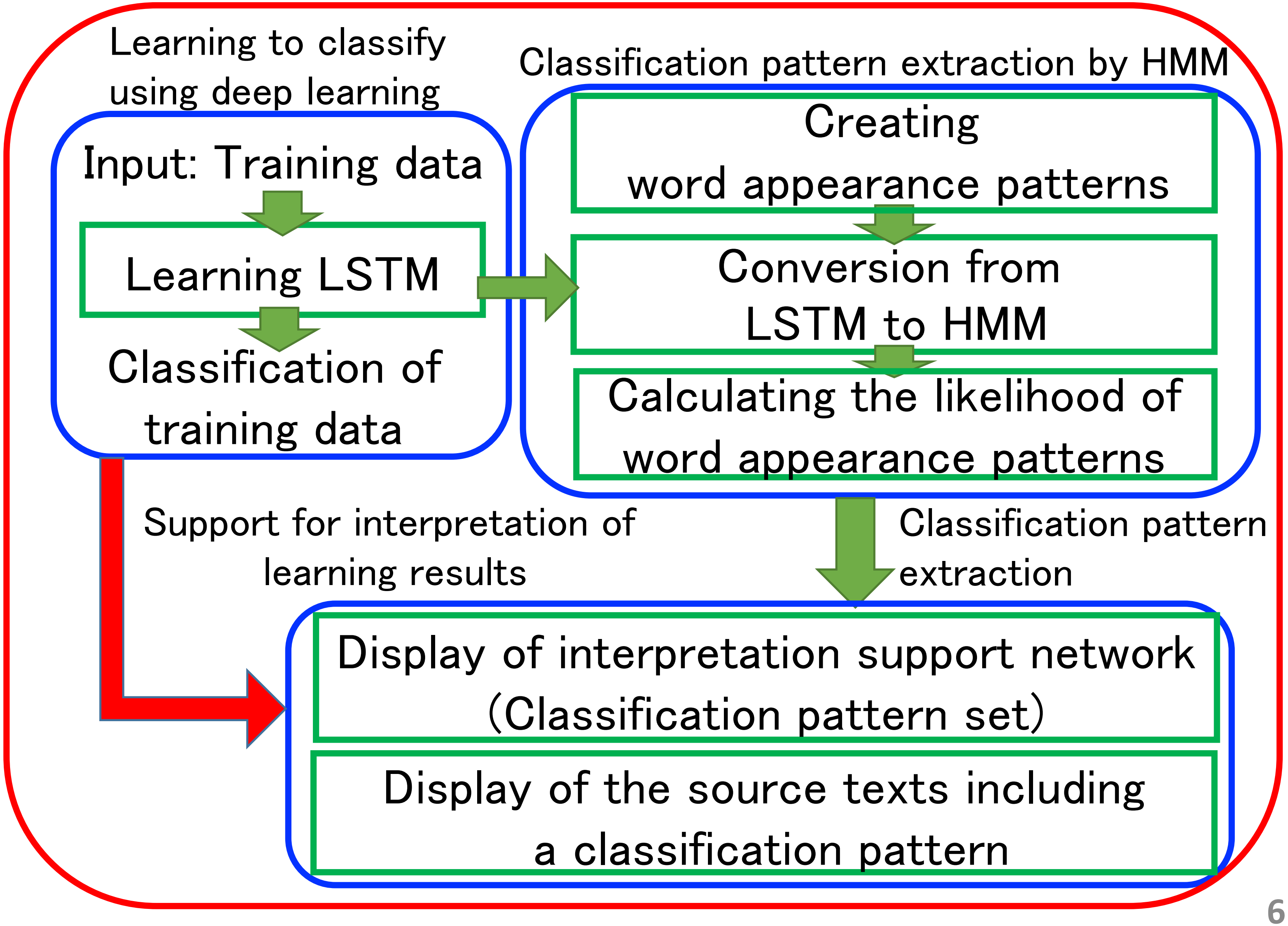
Classification patterns that contribute to "bad"

Users interpret the extracted classification patterns

Users can understand the unique characteristics of a good medical record

Lead to skill acquisition and new knowledge creation

System configuration



Learning to classify using deep learning

Input: Training data

Learning LSTM

Classification of training data

Classification pattern extraction by HMM

Creating word appearance patterns

Conversion from LSTM to HMM

Calculating the likelihood of word appearance patterns

Support for interpretation of learning results

Classification pattern extraction

Display of interpretation support network (Classification pattern set)

Display of the source texts including a classification pattern

Shaping of input data

Input data: A set of sentences for which we want to learn classification patterns

・LSTM is often used in sentences where word order relationships are important (e.g., procedures, synopsis)

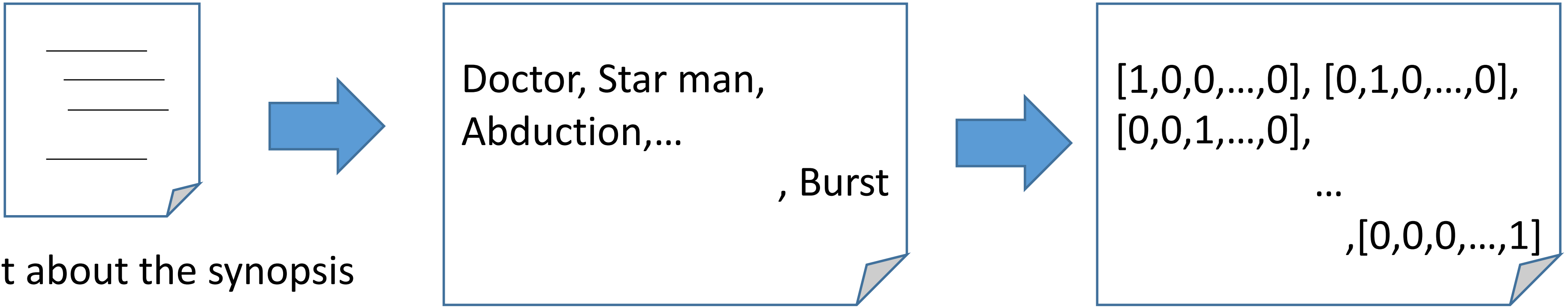
Word vectorization (One hot method)

Extract all words in a sentence and vectorize them as [1,0,0,...,0]

・Igo for morphological analysis (MeCab for dictionary)

・Nouns, verbs, and adjectives are used

Doctor → [1,0,0,...,0]
Star man → [0,1,0,...,0]
Abduction → [0,0,1,...,0]
⋮
Burst → [0,0,0,...,1]



Doctor, Star man,
Abduction,...
, Burst

[1,0,0,...,0], [0,1,0,...,0],
[0,0,1,...,0],
...
,[0,0,0,...,1]

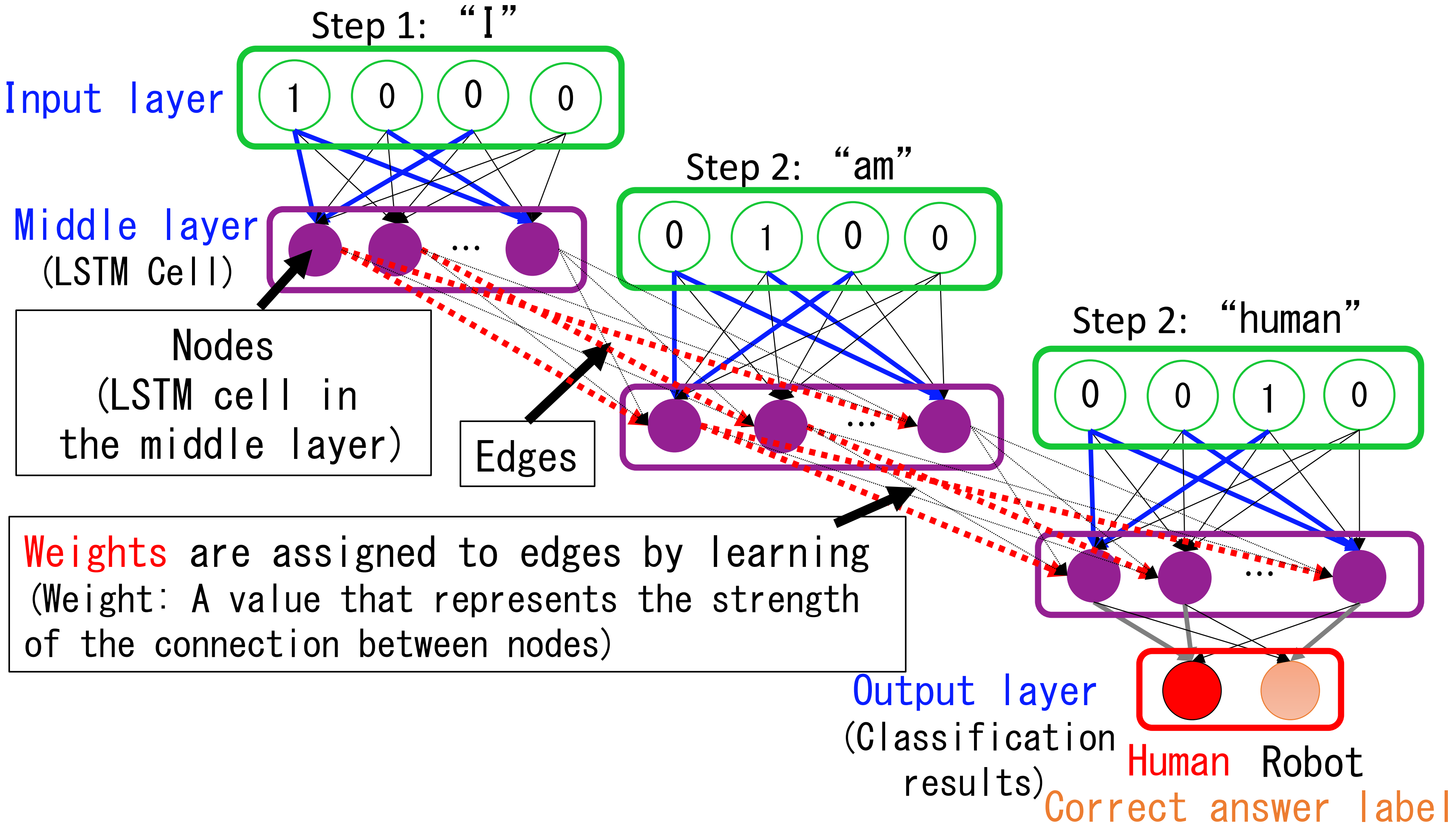
Text about the synopsis of the anime

Sentences with words (nouns) extracted and separated

Sentences with words as vectors

Weighting during classification task using LSTM

Word set for input: “I”, “am”, “human”



Step 1: “I”

Input layer

1 0 0 0

Middle layer
(LSTM Cell)

…

Nodes
(LSTM cell in
the middle layer)

Edges

Step 2: “am”

0 1 0 0

Step 2: “human”

0 0 1 0

Weights are assigned to edges by learning
(Weight: A value that represents the strength
of the connection between nodes)

Output layer
(Classification
results)

Human Robot

Correct answer label

Creating word appearance patterns (observation series)

Create word appearance patterns from combinations of words that appear

① Create combinatorial patterns of arbitrary length from words that appear in the source text

② Keep only word patterns that actually appear in the source text.

Output “carnivore”
Appearance words

Dog
Sheep
Lion
Carnivore

Create
combination patterns

Combination pattern (length 2)

“Dog → sheep”
“Dog → lion”
“Dog → carnivore”
“Sheep → carnivore”
“Lion → carnivore”
⋮

Check if it appears
in the source text

Word appearance pattern

“Dog → sheep”
“Dog → carnivore”
“Lion → carnivore”
⋮

Conversion of LSTM to HMM

Overview of LSTM Conversion to HMM

① Obtain the weight vector for a single node from the weight matrix between layers of LSTM

② Normalize the weight vector and treat it as a probability distribution

③ Treat the probability distribution created from the weights between each layer as the probability of the HMM according to the following

$$\text{LSTM}:(W_i, W_r, W_o)$$
$$\downarrow$$
$$\text{HMM}:(A, B, \pi)$$

→

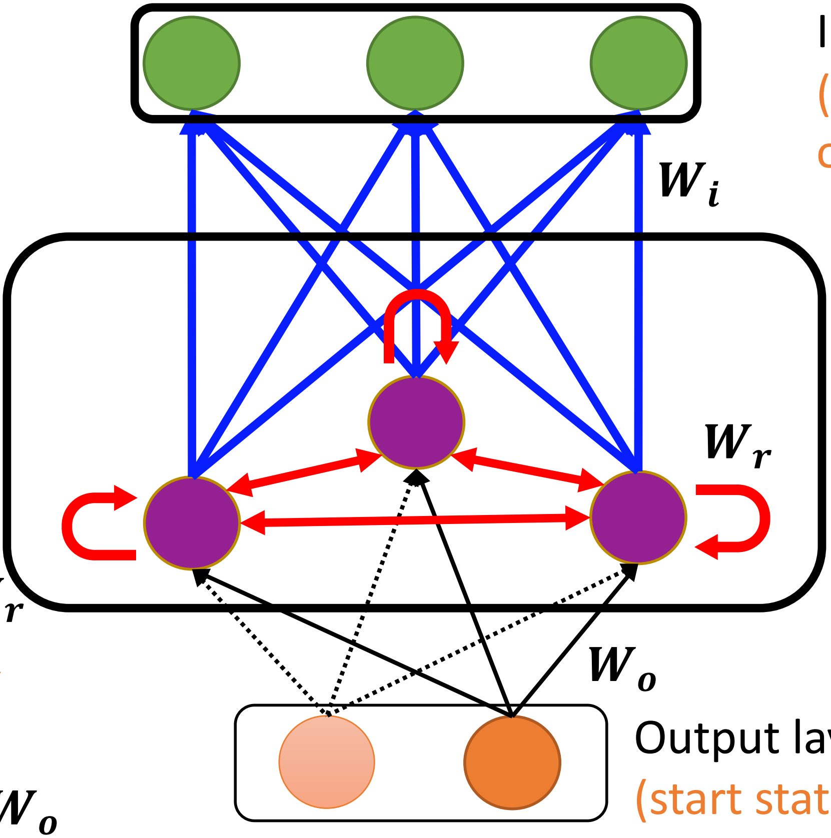
Input ～ middle layer weights $\boldsymbol{W_i}$
→ Observation probabilities B

↔

Weights between middle layers $\boldsymbol{W_r}$
→ State transition probabilities A

→

Output ～ middle layer weights $\boldsymbol{W_o}$
→ Start state probabilities π



Input layer
(Observation symbol of HMM)

Middle layer
(state of HMM)

Output layer
(start state of HMM)

Calculating the likelihood of word appearance patterns

Calculate the likelihood (generation probability) of word appearance patterns

This is the image of going backward from the final output to the intermediate layer nodes at each step

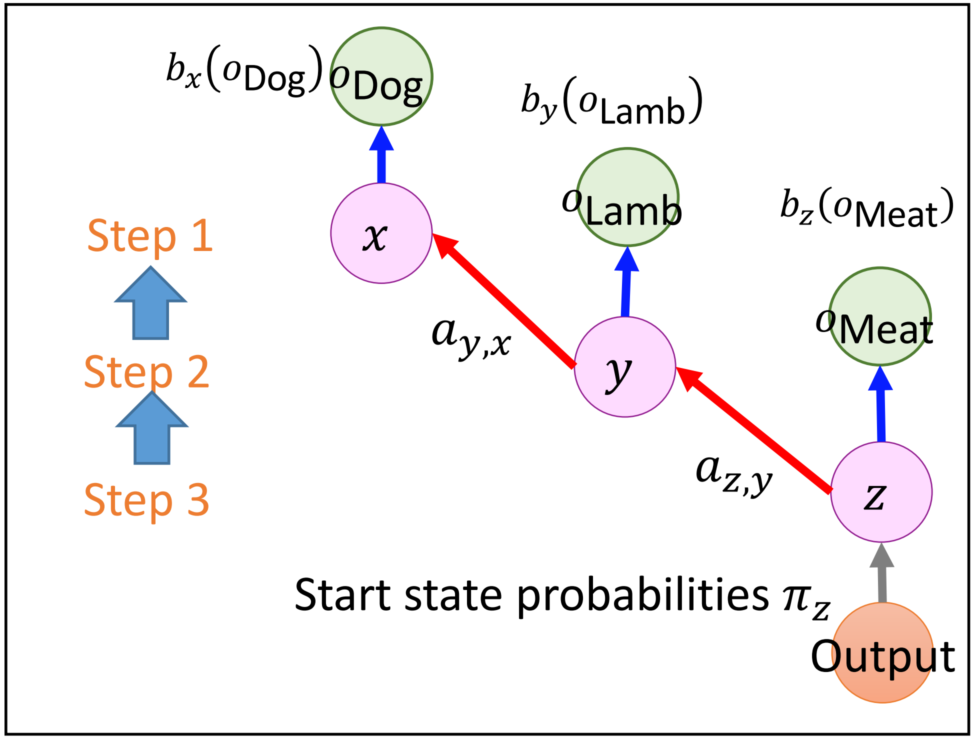
Given HMM : $\lambda = (A, B, \pi)$

Calculation example:
Likelihood of the word appearance pattern
“dog→lamb→meat”

$$P\left(O_{\text{Meat,Lamb,Dog}} \middle| \lambda\right)$$

$$= \sum_{all\, s_0 \ldots s_3} a_{s_0,s_1} b_{s_1}(o_{\text{Meat}}) \cdot a_{s_1,s_2} b_{s_2}(o_{\text{Lamb}}) \cdot a_{s_2,s_3} b_{s_3}(o_{\text{Dog}})$$

※ $\pi_{s_1} = a_{s_0,s_1}$


$b_x(o_{\text{Dog}})$
o_{Dog}
$b_y(o_{\text{Lamb}})$
Step 1
Step 2
Step 3
x
o_{Lamb}
$b_z(o_{\text{Meat}})$
$a_{y,x}$
y
o_{Meat}
$a_{z,y}$
z
Start state probabilities π_z
Output

$a_{i,j}$: Transition probability from state i to j
$b_i(o_t)$: Observation probability from state i to symbol o_t
s_t : State at time t ($1 \leq t \leq 3$ in this example)

Determine the classification patterns to be extracted

① Arrange word appearance patterns in order of calculated likelihood

1st place “Dog→Goat→Meat” : likelihood 1.89
2nd place “Cat→Dog→Enemy” : likelihood 1.45
3rd place “Sheep→Dog→Flee” : likelihood 1.23
4th place “Cat→Human→Whiting” : likelihood 1.10
⋮

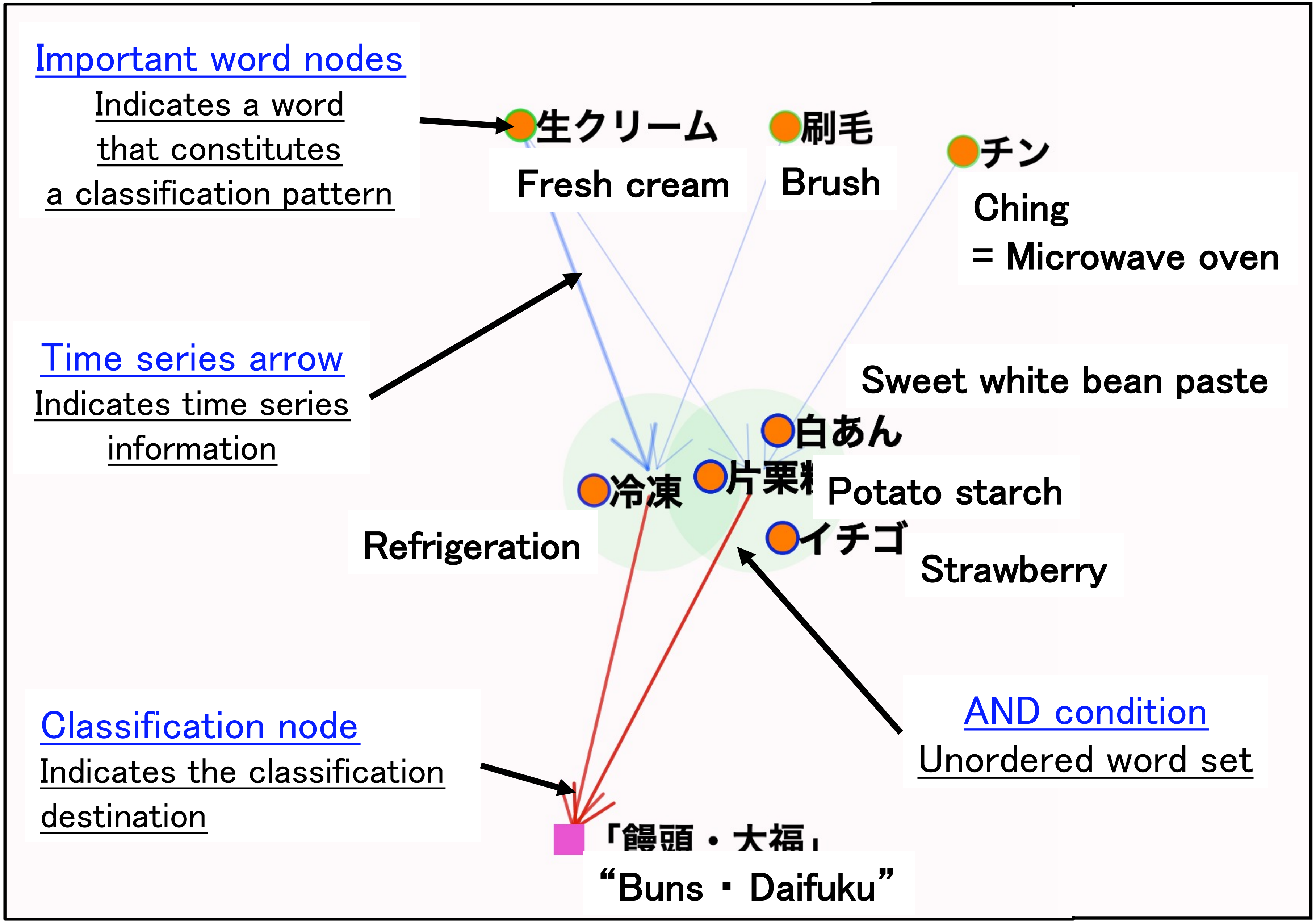
② Extraction of top word appearance patterns as classification patterns

1st place Classification pattern “Dog→Goat→Meat”
2nd place Classification pattern “Cat→Dog→Enemy”
3rd place Classification pattern “Sheep→Dog→Flee”

Output "animal" classification patterns

※ It is up to the system user to decide how much of a pattern to classify

Interpretation support network (visualization of extracted classification patterns)


生クリーム
Fresh cream
刷毛
Brush
チン
Ching
= Microwave oven
Sweet white bean paste
白あん
冷凍
片栗
Potato starch
Refrigeration
イチゴ
Strawberry
「饅頭・大福」
“Buns ・ Daifuku”

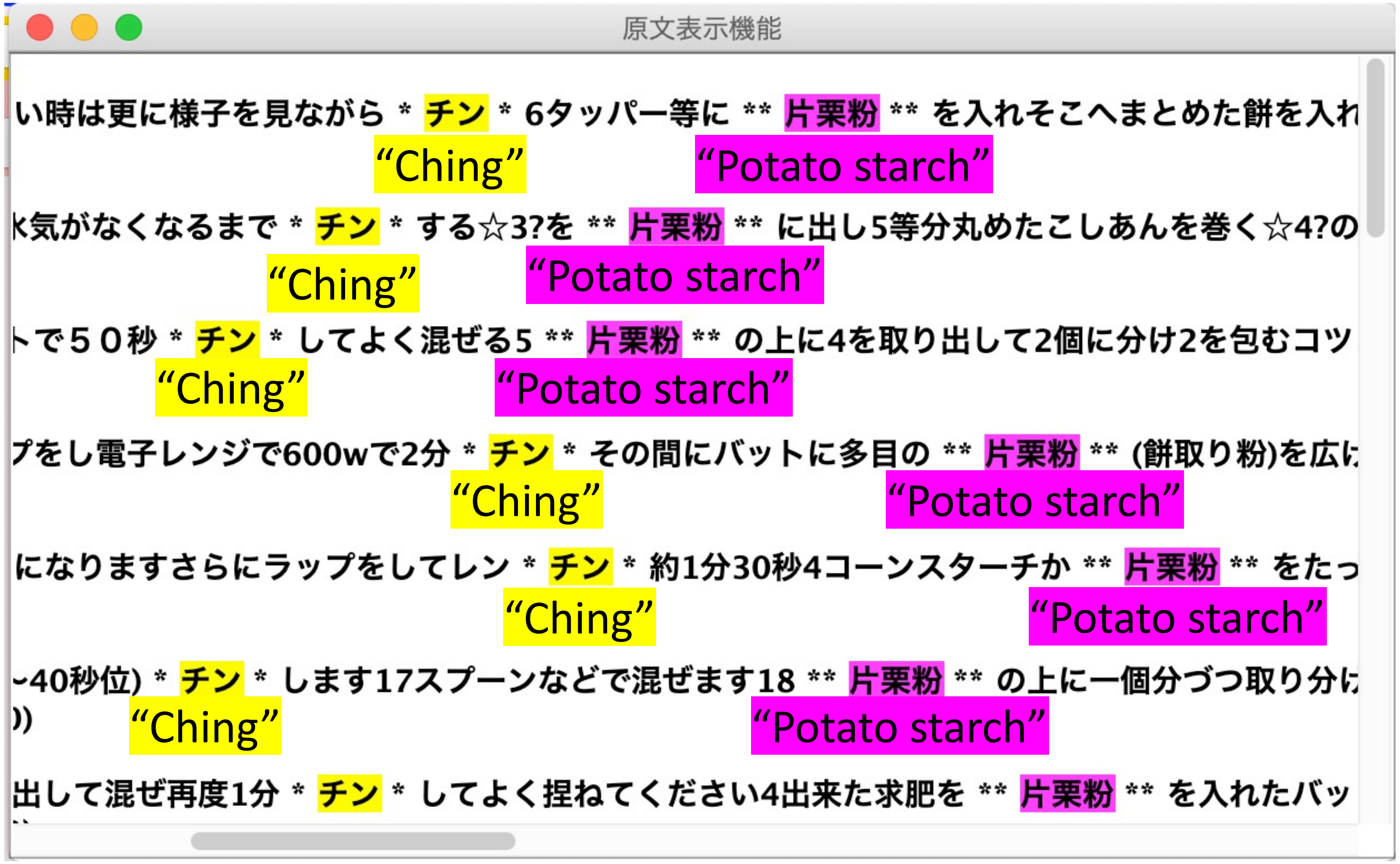
Important word nodes
Indicates a word that constitutes a classification pattern

Time series arrow
Indicates time series information

AND condition
Unordered word set

Classification node
Indicates the classification destination

Interpretation support function: Source text display function



い時は更に様子を見ながら * チン * 6タッパー等に ** 片栗粉 ** をそこへまとめた餅を入れ

“Ching” “Potato starch”

気がなくなるまで * チン * する☆3?を ** 片栗粉 ** に出し5等分丸めたこしあんを巻く☆4?の

“Ching” “Potato starch”

で５０秒 * チン * してよく混ぜる5 ** 片栗粉 ** の上に4を取り出して2個に分け2を包むコツ

“Ching” “Potato starch”

プをし電子レンジで600wで2分 * チン * その間にバットに多目の ** 片栗粉 ** (餅取り粉)を広

“Ching” “Potato starch”

になりますさらにラップをしてレン * チン * 約1分30秒4コーンスターチか ** 片栗粉 ** をた

“Ching” “Potato starch”

~40秒位) * チン * します17スプーンなどで混ぜます18 ** 片栗粉 ** の上に一個分づつ取り分
))

“Ching” “Potato starch”

出して混ぜ再度1分 * チン * してよく捏ねてください4出来た求肥を ** 片栗粉 ** を入れたバッ

The display range is up to 50 characters before and after the word

If there is more than one sentence, all of them will be displayed

Display of the source text display function (when “Chin” → “Potato starch” is selected)

- Click on a word in the Interpretation Network to select it (multiple selections are possible)
  → A part of the source text in which the selected word is used is displayed.

Experiment

An experiment was conducted to verify that the proposed system was effective for interpreting classification patterns

Experimental tasks (common to both proposed and comparison systems)

① “Character dialogues” ･Number of texts: 1500 ･Average number of characters per text: 40
Classification of “tsundere”, “deredere”, and “normal” anime character dialogue sets [5]
→ Find 10 characteristics unique to the “tsundere” character and give an interpretation

② “Consumer electronics reviews” ･Number of texts: 3108 ･Average number of characters per text: 244
Classification of “useful” (more than 4 stars and more than 10 “useful people”), “useless” (more than 4 stars and 0 “useful people”), and “low-rated” (less than 2 stars) review sets [6] for popular consumer electronics on Amazon
→ Find 10 characteristics unique to “useful” reviews and give an interpretation

③ “Game reviews” ･Number of texts: 4419 ･Average number of characters per text: 455
Classification of “useful”, “useless”, and “low-rated” review sets [6] for popular game software on Amazon
→ Find 10 characteristics unique to “useful” reviews and give an interpretation

- Comparison system

  For comparison purposes, we have prepared a system that displays the words of the assigned text in order of TFIDF value (with a function to display the source text)

[5]｢Twitter｣ http://twitter.com , [6] ｢amazon｣https://www.amazon.co.jp

Deep learning model (LSTM)

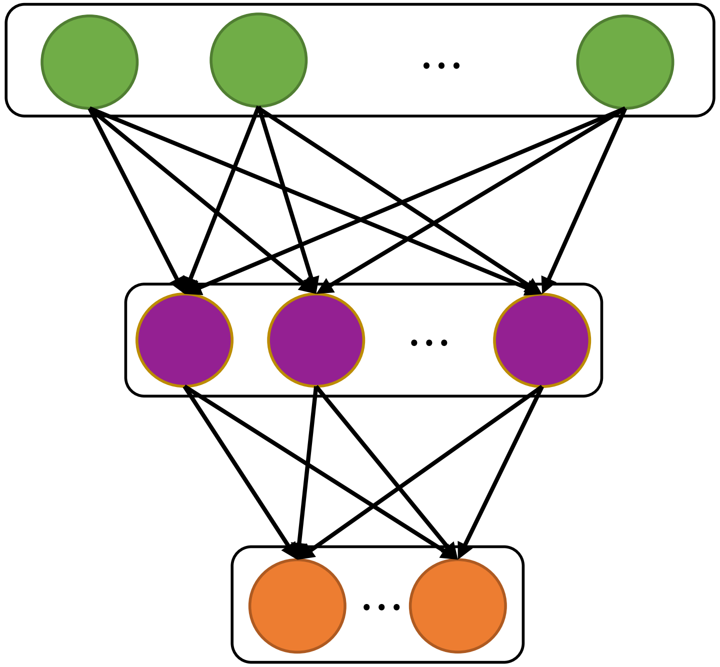
Model used: LSTM with one intermediate layer (3-layer fully coupled model)

Subjects: 16 undergraduate and graduate students

→8 in the proposal system group, 8 in the comparison system group

- Number of training: 10 times ・Activation Function: ReLU ・Learning rate: 0.01
- Classification accuracy: Task 1: 98.7%, Task 2: 99.2%, Task 3: 96.7%

Input: Word vector (one hot)



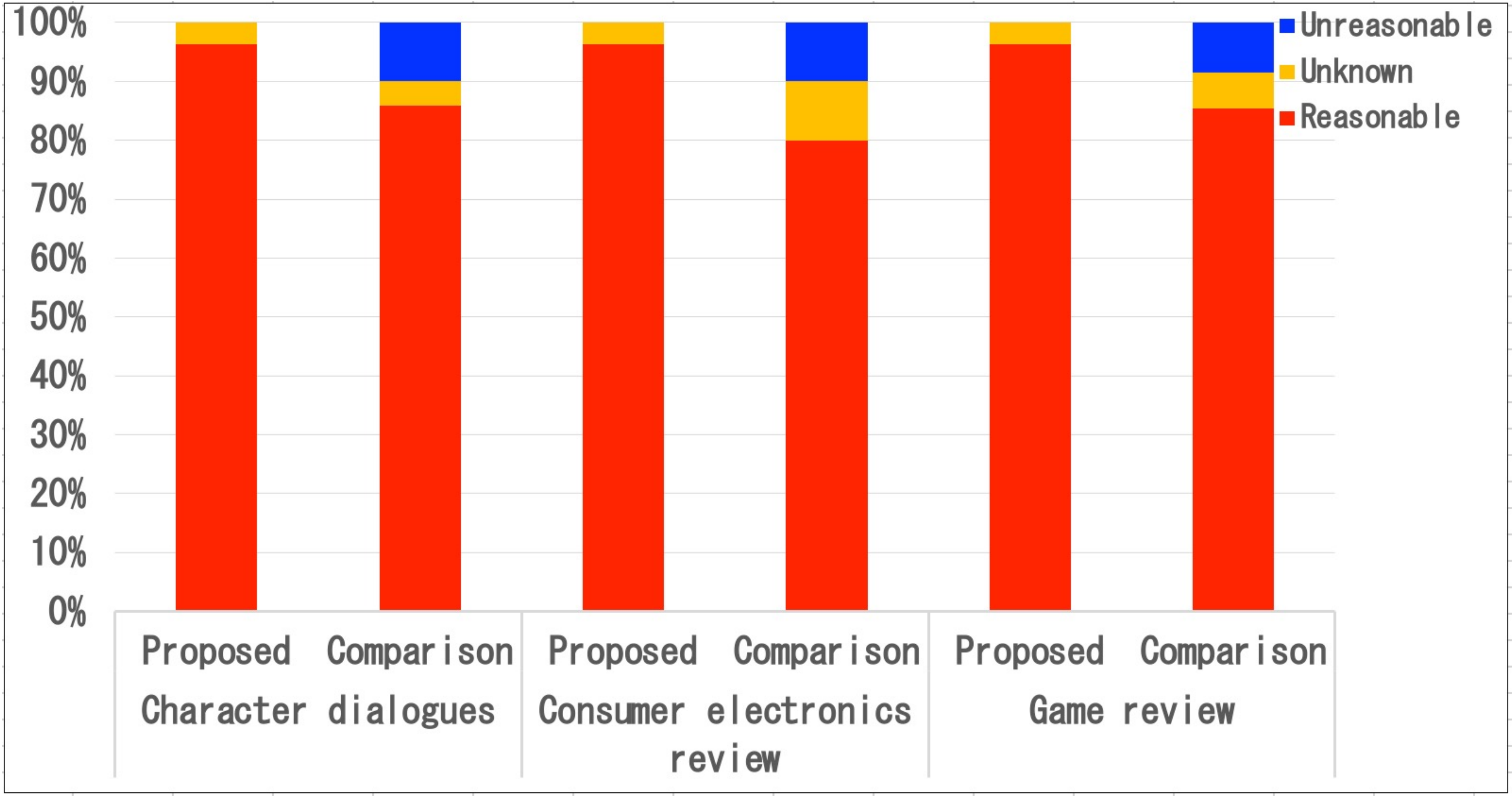
Output: Probability of belonging to each classification destination

Input layer
Number of nodes: Task 1: 510
Task 2: 916
Task 3: 1809

Middle layer
Number of nodes (LSTM units): 10

Output layer
Number of nodes: 3

Results (breakdown of the validity of the interpretation)

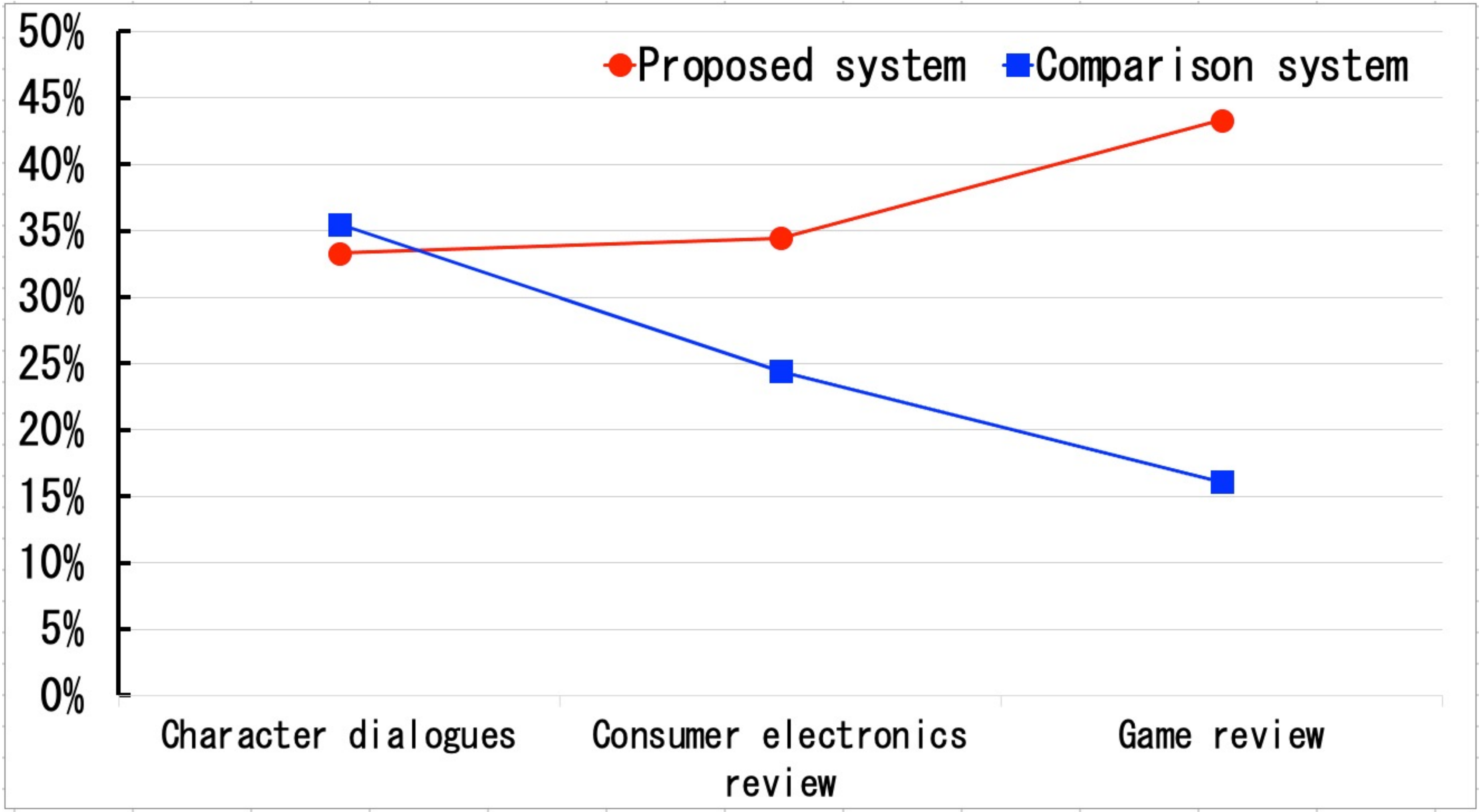


Average of interpretations classified as “reasonable,” “unknown,” and “unreasonable” for both groups

- With the proposed system, more than 97% of the interpretations were classified as reasonable, and there were no unreasonable interpretations

→It can be said that the proposed system enables us to make more reasonable interpretations

Results (percentage of source texts where the interpretation applies)



Percentages of interpretations that fit the source text for "reasonable" interpretations in both groups

・The percentage of the source texts that were applicable to “consumer electronics review” and “game review” was more than 10% higher

→In the proposed system, we can say that the interpretation is more applicable to a wider range of the source text

Conclusion and Future Work

- We developed a system that supports extraction and interpretation of classification patterns from deep learning using network weights from the results of text classification

- Proposes a method for extracting classification patterns that transforms LSTM into HMM
  - ✓ By using HMM, we can easily analyze complex LSTM
  - ✓ Classification patterns can be extracted by considering the learned time series information of LSTM
  - ✓ In an experiment to verify the effectiveness of the proposed system, we concluded that the proposed system can be used to derive reasonable and broadly applicable interpretations of the source text from the classification patterns including the time series information

Challenge: Toward interpretation of classification patterns

① Improving the ease of interpretation

・It is not enough to display the classification patterns that contribute to the classification
→Functions that enable understanding of the source meaning of the classification patterns are necessary

② Changes in deep learning models

・We need to improve the interpretability of attention-based deep learning models such as BERT