命名实体识别
Named Entity Recognition (NER)

李辰

2024年10月

Outline

1. Introduction to IE and NER
2. Rule-based Approaches
3. Feature-based Approaches
4. Deep Learning Methods

Outline

1. Introduction to IE and NER
2. Rule-based Approaches
3. Feature-based Approaches
4. Deep Learning Methods

Introduction

- **Information Extraction**

Information extraction (IE) systems

- Find and understand limited relevant parts of texts
- Gather information from many pieces of text
- Produce a structured representation of relevant information:
  - *relations* (in the database sense), a.k.a.,
  - a *knowledge base*

Introduction

● Information Extraction

Information extraction (IE) systems

- Find and understand limited relevant parts of texts
- Gather information from many pieces of text
- Produce a structured representation of relevant information:
  - *relations* (in the database sense), a.k.a.,
  - a *knowledge base*
- Goals:
  1. Organize information so that it is useful to people
  2. Put information in a semantically precise form that allows further inferences to be made by computer algorithms

Introduction

● Information Extraction

IE systems extract clear, factual information

- Roughly: *Who did what to whom when?*

Introduction

● Information Extraction

IE systems extract clear, factual information

- Roughly: *Who did what to whom when?*

E.g.,

- Gathering earnings, profits, board members, headquarters, etc. from company reports

Introduction

- **Information Extraction**

IE systems extract clear, factual information

- Roughly: *Who did what to whom when?*

E.g.,

- Gathering earnings, profits, board members, headquarters, etc. from company reports
  - The headquarters of BHP Billiton Limited, and the global headquarters of the combined BHP Billiton Group, are located in Melbourne, Australia.
  - headquarters("BHP Biliton Limited", "Melbourne, Australia")

Introduction

- **Information Extraction**

IE systems extract clear, factual information

- Roughly: *Who did what to whom when?*

E.g.,

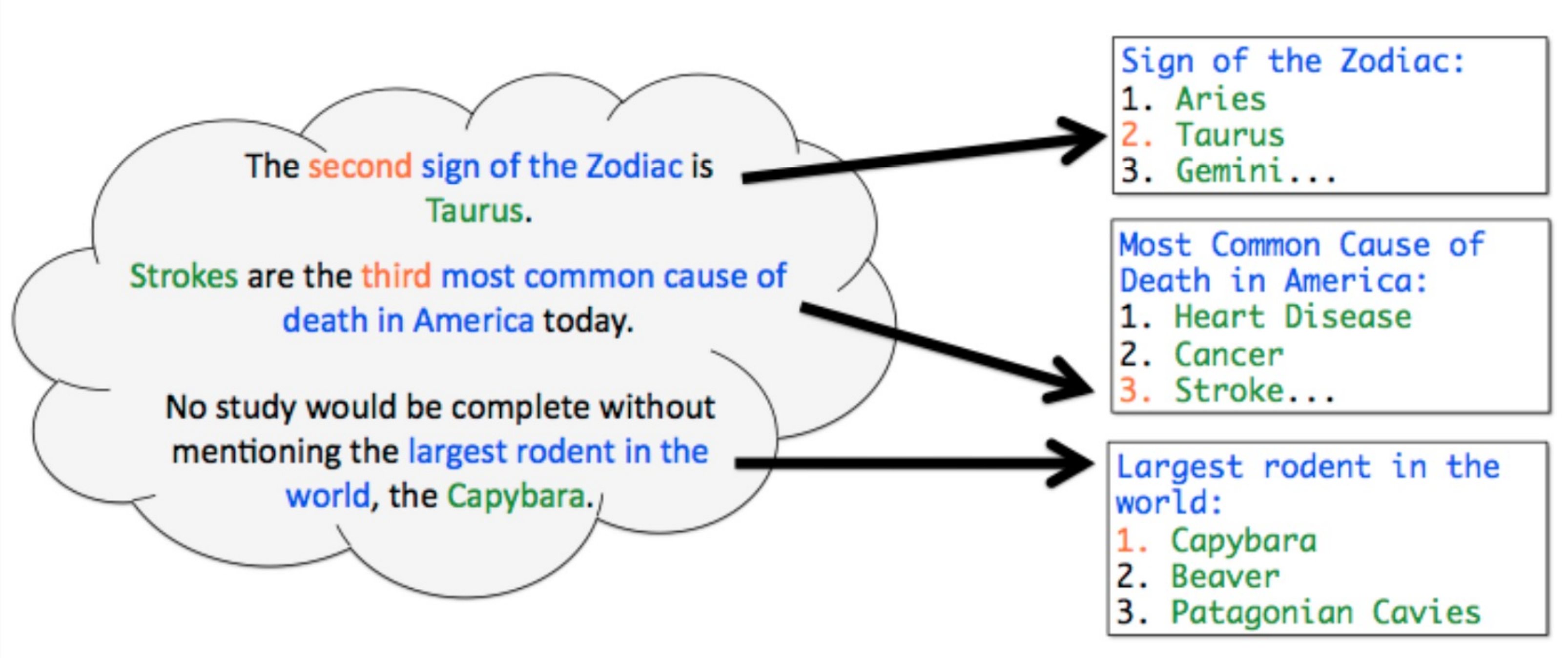
- Gathering earnings, profits, board members, headquarters, etc. from company reports
  - The headquarters of BHP Billiton Limited, and the global headquarters of the combined BHP Billiton Group, are located in Melbourne, Australia.
  - headquarters("BHP Biliton Limited", "Melbourne, Australia")
- Learn and categorize entity information in text.

Introduction

- **Information Extraction**

Automatic identification and classification of instances of user-specified types of entities, relations, and events from text.

Unstructured text ⟶ **Structured sequences**



Introduction

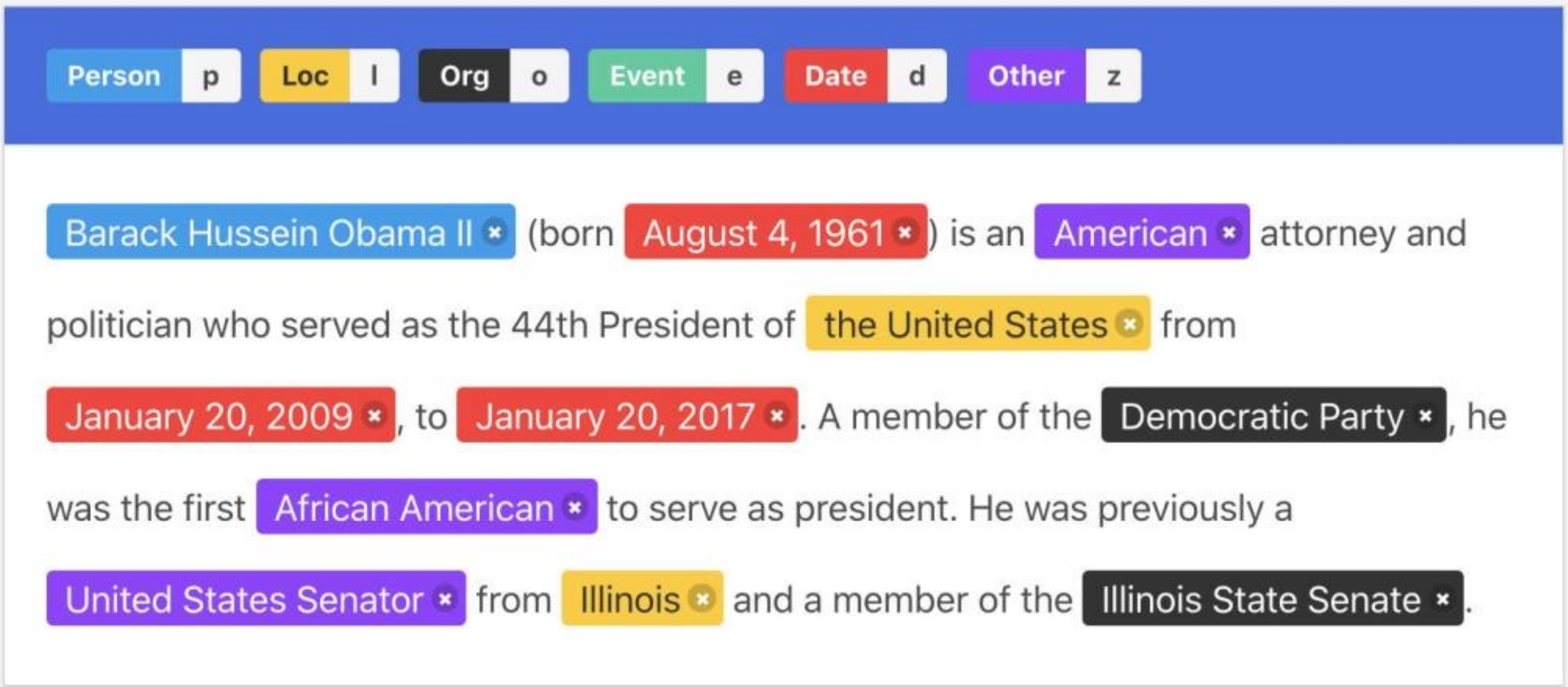
- **Named-entity Recognition (NER)**

A method of recognizing and classifying essential pieces of information from within larger unstructured text-based data into predefined categories such as person names, organizations, locations.

Introduction

- **Named-entity Recognition (NER)**

A method of recognizing and classifying essential pieces of information from within larger unstructured text-based data into predefined categories such as person names, organizations, locations.



Introduction

- **Pattern-based NER**

Dictionary-based NER:

Match the words in the input text against the words in the dictionary. If a match is found, then the corresponding entity type is assigned to the text segment that contains the matched word.

Introduction

- **Pattern-based NER**

Dictionary-based NER:

There are several instances:

- Stanford Named Entity Recognizer(Stanford NER): Stanford NER is a widely used dictionary-based NER system that supports several entity types, including persons, organizations, locations, and miscellaneous entities.

Introduction

- **Pattern-based NER**

Dictionary-based NER:

There are several instances:

- Stanford Named Entity Recognizer(Stanford NER)
- SpaCy: SpaCy is a popular NLP library that includes a built-in named entity recognizer based on a combination of rule-based and dictionary-based approaches. It supports several entity types, including persons, organizations, locations, and products, and can be trained on custom entity dictionaries

Introduction

- **Pattern-based NER**

Dictionary-based NER:

There are several instances:

- Stanford Named Entity Recognizer(Stanford NER)
- SpaCy
- NLTK: The Natural Language Toolkit (NLTK) is a popular Python library for NLP that includes a dictionary-based named entity recognizer

Introduction

- **Pattern-based NER**

Dictionary-based NER:

There are several instances:

- Stanford Named Entity Recognizer(Stanford NER)
- SpaCy
- NLTK
- GATE:The General Architecture for Text Engineering (GATE) is a comprehensive NLP framework that includes a dictionary-based named entity recognizer

Introduction

- **Pattern-based NER**

Dictionary-based NER:

There are several instances:

- Stanford Named Entity Recognizer(Stanford NER)
- SpaCy
- NLTK
- GATE

The choice of system depends on the specific task, data, and resources available

Outline

1. Introduction to IE and NER
2. Rule-based Approaches
3. Feature-based Approaches
4. Deep Learning Methods

Introduction

- **Pattern-based NER**

Syntax parsing tree:

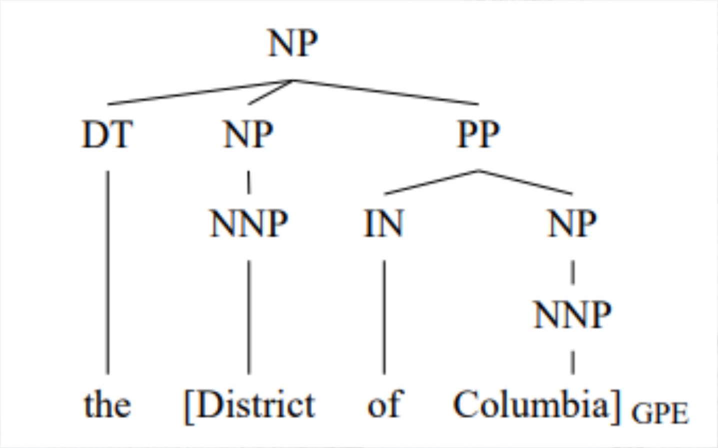
We can parse natural language sentences into syntax trees based on language patterns, and perform named entity recognition using the syntax trees.

Introduction

- **Pattern-based NER**

Syntax parsing tree:

We can parse natural language sentences into syntax trees based on language patterns, and perform named entity recognition using the syntax trees.



For instance, the District of Columbia can be parsed like image above.
We set a rule that NP(noun phrase) followed by PP(prepositional phrase) is a GPE entity. After recognizing this pattern, we set this span to GPE entity.

Introduction

- **Pattern-based NER**

Regular expression-based NER:

We can use common patterns for using regular expressions to match a certain type of entity.

Introduction

- **Pattern-based NER**

Regular expression-based NER:

We can use common patterns for using regular expressions to match a certain type of entity.

	regex
location	(?:[A-Z][a-z]*)*[A-Z][a-z]+
organization	[A-Z][a-z]+(?: [A-Z][a-z]+)* (?:Co\.|Inc\.|Ltd\.)?
person	[A-Z][a-z]+ [A-Z][a-z]+

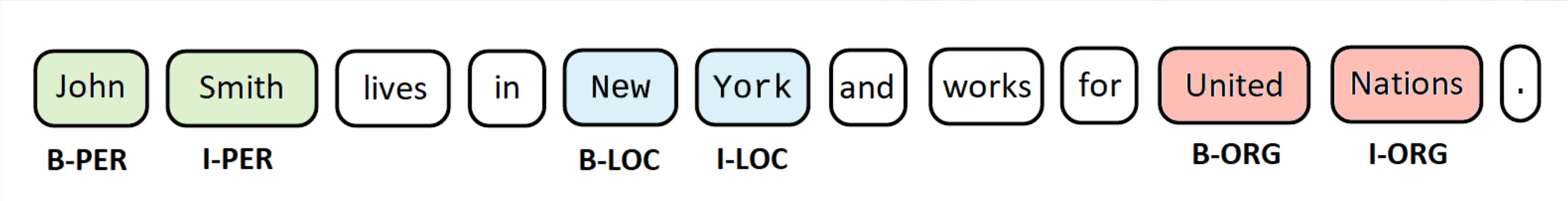
Outline

1. Introduction to IE and NER
2. Rule-based Approaches
3. Feature-based Approaches
4. Deep Learning Methods

Feature-based NER

- **(Conditional Random Fields) CRF**

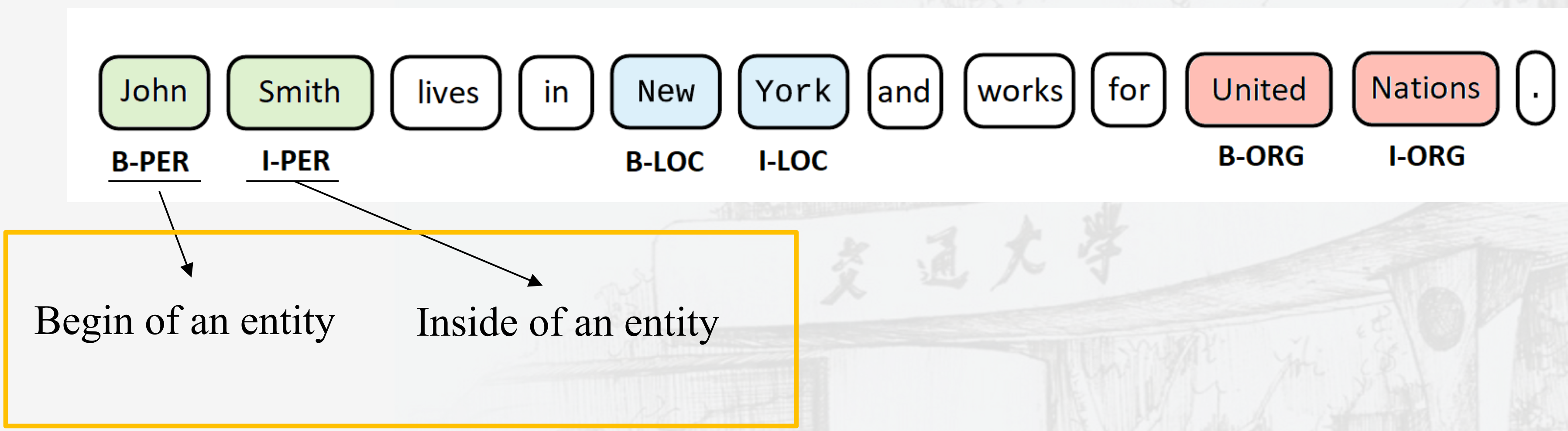
Modeling NER as a sequence labeling task, assigning a label to each word, and identifying entities through the labels.



Feature-based NER

- **(Conditional Random Fields) CRF**

Modeling NER as a sequence labeling task, assigning a label to each word, and identifying entities through the labels.



An entity

Feature-based NER

- **(Conditional Random Fields) CRF**

A linear chain CRF confers to a labeler in which tag assignment(for present word, denoted as y_i) depends only on the tag of just one previous word(denoted by y_{i-1}).

Feature Functions:

Functions that extract multiple features per word of the sentence that will be assisting in recognizing a Named Entity

Feature-based NER

- **(Conditional Random Fields) CRF**

A linear chain CRF confers to a labeler in which tag assignment(for present word, denoted as y_i) depends only on the tag of just one previous word(denoted by y_{i-1}).

Feature Functions:

Functions that extract multiple features per word of the sentence that will be assisting in recognizing a Named Entity

Feature Functions take four parameters:

- Index of current word i, which represents the position of a word
- Label of current word y_i, which represents the sequence label of a word
- Label of previous word y_{i-1}, which represents the sequence label of previous word
- Original sentence x, which represents sentence to be extracted

So the format of the Feature Functions is $f_j(x, y_i, y_{i-1}, i)$, where j represents jth Feature Functions, we don't only have one **Feature Function** for each task.

Feature-based NER

- **(Conditional Random Fields) CRF**

Consider ‘Ram is cool’ as our example with Named Entity Labels as [PER O O] where we have Ram: PER, is:O, cool:O

Consider a **Feature function** $f_j(x, y_i, y_{i-1}, i)$ with the definition: The i-th word in ‘x’ is capitalized return 1 else 0

Feature-based NER

- **(Conditional Random Fields) CRF**

Consider 'Ram is cool' as our example with Named Entity Labels as [PER O O] where we have Ram: PER, is:O, cool:O

Consider a **Feature function** $f_j(x, y_i, y_{i-1}, i)$ with the definition: The i-th word in 'x' is capitalized return 1 else 0

If i=1, hence we are calculating feature for 'is', the above feature function is demonstrated below:

$$f_j(\textbf{'Ram is cool'}, \textbf{'O'}, \textbf{'PER'}, 1) = 0$$

as 'is' isn't capitalized

Feature-based NER

- **(Conditional Random Fields) CRF**

Consider 'Ram is cool' as our example with Named Entity Labels as [PER O O] where we have Ram: PER, is:O, cool:O

Consider a **Feature function** $f_j(x, y_i, y_{i-1}, i)$ with the definition: The i-th word in 'x' is capitalized return 1 else 0

If i=1, hence we are calculating feature for 'is', the above feature function is demonstrated below:

$$f_j(\textbf{'Ram is cool'}, \textbf{'O'}, \textbf{'PER'}, 1) = 0$$

as 'is' isn't capitalized

We can define various **Feature function,** there is no limitation for definition. The main goal of the **Feature function** is find a better way to represent the information in sentence.

Feature-based NER

- **(Conditional Random Fields) CRF**

The CRF is defined below:

$$p_\theta(y|x) = \frac{\exp(\sum_j w_j F_j(x,y))}{\sum_{y'} \exp(\sum_j w_j F_j(x,y'))}$$

Where

$$F_j(x,y) = \sum_{i=1}^{L} f_j(x, y_i, y_{i-1}, i)$$

Feature-based NER

- **(Conditional Random Fields) CRF**

The CRF is defined below:

$$p_\theta(y|x) = \frac{\exp(\sum_j w_j F_j(x,y))}{\sum_{y'} \exp(\sum_j w_j F_j(x,y'))}$$

Where

$$F_j(x,y) = \sum_{i=1}^{L} f_j(x, y_i, y_{i-1}, i)$$

Here, L is the sequence length of the sentence, w_j represent weight of the function

- $p_\theta(y|x)$ refers to the probability of calculating a Label sequence(y) given a word sequence(x)
- $F_j(x,y)$ is summation of values of a feature function for all words.

Feature-based NER

- **(Conditional Random Fields) CRF**

The CRF is defined below:

$$p_\theta(y|x) = \frac{\exp(\sum_j w_j F_j(x,y))}{\sum_{y'} \exp(\sum_j w_j F_j(x,y'))}$$

Where

$$F_j(x,y) = \sum_{i=1}^{L} f_j(x, y_i, y_{i-1}, i)$$

Understanding each term one by one:

- The inner summation goes from i=1 to i=length of sentence 'L'. Hence we are summating the value of any feature function for all words of the sentence

if we have a sentence 'Ram is cool', the inner summation will add values of the output of the j^{th} feature function for all 3 words of the sentence

Feature-based NER

- **(Conditional Random Fields) CRF**

The CRF is defined below:

$$p_\theta(y|x) = \frac{\exp(\sum_j w_j F_j(x,y))}{\sum_{y'} \exp(\sum_j w_j F_j(x,y'))}$$

Where

$$F_j(x,y) = \sum_{i=1}^{L} f_j(x, y_i, y_{i-1}, i)$$

Understanding each term one by one:

- The inner summation goes from i=1 to i=length of sentence 'L'. Hence we are summating the value of any feature function for all words of the sentence
- The outer summation goes from j=1 to the total number of feature functions. It is like W_1*Σfeature_function$_1$+W_2*Σfeature_function$_2$

Feature-based NER

- **(Conditional Random Fields) CRF**

The CRF is defined below:

$$p_\theta(y|x) = \frac{\exp(\sum_j w_j F_j(x,y))}{\sum_{y'} \exp(\sum_j w_j F_j(x,y'))}$$

Where

$$F_j(x,y) = \sum_{i=1}^{L} f_j(x, y_i, y_{i-1}, i)$$

Understanding each term one by one:

- The inner summation goes from i=1 to i=length of sentence 'L'. Hence we are summating the value of any feature function for all words of the sentence
- The outer summation goes from j=1 to the total number of feature functions. It is like W_1*Σfeature_function$_1$+W_2*Σfeature_function$_2$
- y' refers to all the possible Label Sequence that can be assigned to a word sequence (sentence)

Outline

1. Introduction to IE and NER
2. Rule-based Approaches
3. Feature-based Approaches
4. Deep Learning Methods

Deep Learning Method

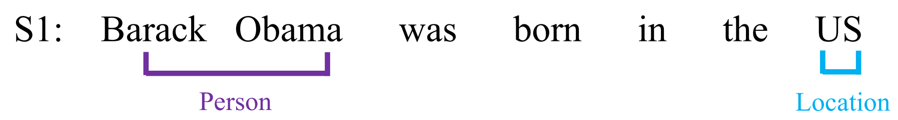
- **NER: The three NER subtasks**

There are three different subtasks of NER: flat NER (simple entity extraction), nested NER, and discontinuous NER.

Deep Learning Method

- **NER: The three NER subtasks**

There are three different subtasks of NER: flat NER (simple entity extraction), nested NER, and discontinuous NER.

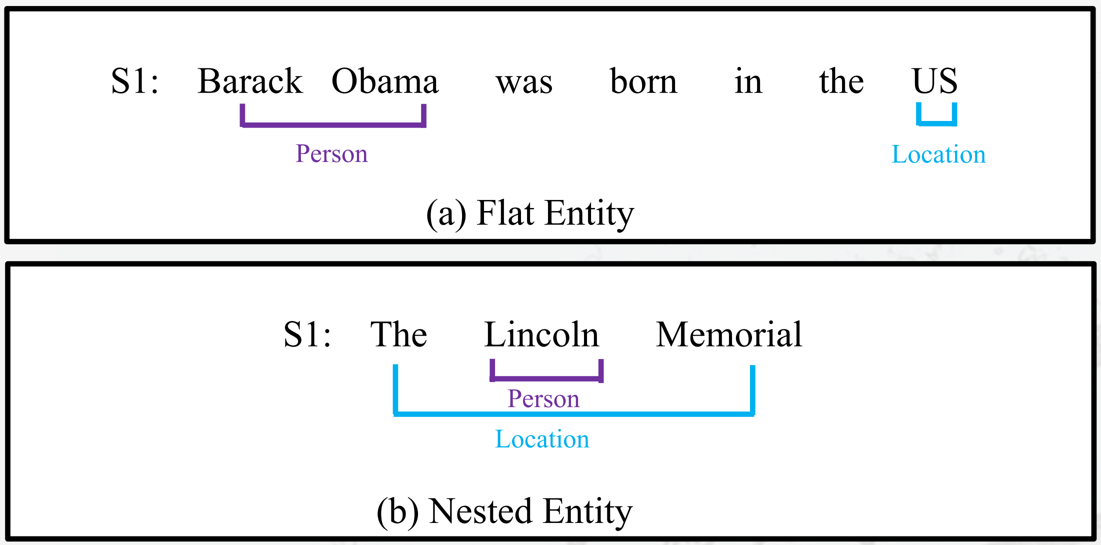


(a) Flat Entity

Deep Learning Method

- **NER: The three NER subtasks**

There are three different subtasks of NER: flat NER (simple entity extraction), nested NER, and discontinuous NER.



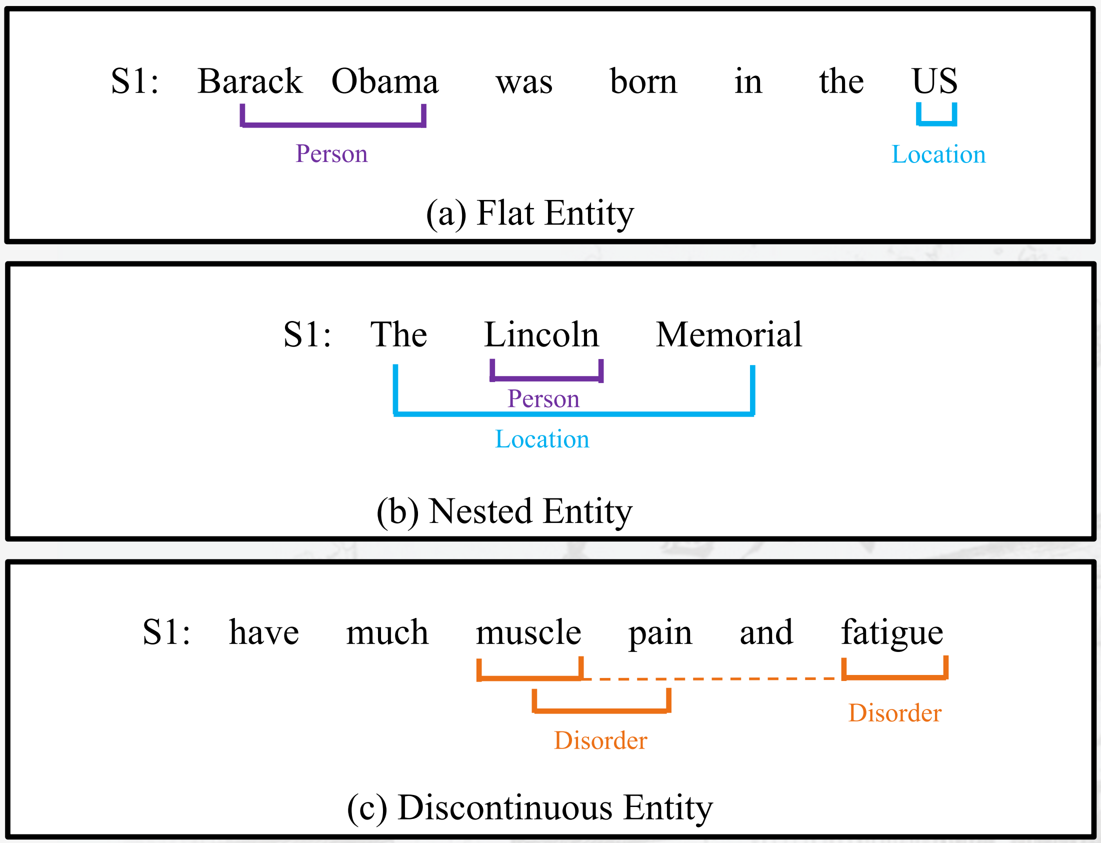
(a) Flat Entity

(b) Nested Entity

Deep Learning Method

- **NER: The three NER subtasks**

There are three different subtasks of NER: flat NER (simple entity extraction), nested NER, and discontinuous NER.



Deep Learning Method

- **NER: The four paradigms with Deep Learning**

  - **Sequence Labeling Methods**

    Each word in the text is treated as an input in the sequence, and a label is assigned to each word to indicate which type of entity it belongs to.

Deep Learning Method

- **NER: The four paradigms with Deep Learning**

- **Sequence Labeling Methods**

  Each word in the text is treated as an input in the sequence, and a label is assigned to each word to indicate which type of entity it belongs to.

- **Machine Reading Comprehension (MRC) Methods**

  This paradigm transforms the NER task into a reading comprehension problem by extracting key information from the text to answer a series of questions about entity types.

Deep Learning Method

- **NER: The four paradigms with Deep Learning**

  - **Sequence Labeling Methods**

    Each word in the text is treated as an input in the sequence, and a label is assigned to each word to indicate which type of entity it belongs to.

  - **Machine Reading Comprehension (MRC) Methods**

    This paradigm transforms the NER task into a reading comprehension problem by extracting key information from the text to answer a series of questions about entity types.

  - **Token-Pair Methods**

    Identify named entities by building relationships between tokens. In this method, there is a prediction target between every two tokens in the text, indicating whether there is an entity boundary between them.

Deep Learning Method

- **NER: The four paradigms with Deep Learning**

  - **Sequence Labeling Methods**

    Each word in the text is treated as an input in the sequence, and a label is assigned to each word to indicate which type of entity it belongs to.

  - **Machine Reading Comprehension (MRC) Methods**

    This paradigm transforms the NER task into a reading comprehension problem by extracting key information from the text to answer a series of questions about entity types.

  - **Token-Pair Methods**

    Identify named entities by building relationships between tokens. In this method, there is a prediction target between every two tokens in the text, indicating whether there is an entity boundary between them.

  - **Generative Methods**

    Treating named entity recognition as a text generation task, these methods utilize the power of generative pre-training models to handle three different types of NER tasks in a unified framework.

Deep Learning Method

• NER: The four paradigms with Deep Learning

- **Sequence Labeling Methods**

  Each word in the text is treated as an input in the sequence, and a label is assigned to each word to indicate which type of entity it belongs to.

- **Machine Reading Comprehension (MRC) Methods**

  This paradigm transforms the NER task into a reading comprehension problem by extracting key information from the text to answer a series of questions about entity types.

- **Token-Pair Methods**

  Identify named entities by building relationships between tokens. In this method, there is a prediction target between every two tokens in the text, indicating whether there is an entity boundary between them.

- **Generative Methods**

  Treating named entity recognition as a text generation task, these methods utilize the power of generative pre-training models to handle three different types of NER tasks in a unified framework.

Using deep neural networks (such as multi-layer LSTM or pre-trained language models) as the basic model, encode the input text into contextual representations, and then model NER through different paradigms.

Deep Learning Method

- **NER: Sequence Labeling Methods**

For a sequence labeling task, the input is a sentence X:

$$X = X_1, \mathrm{X}_2, \dots, \mathrm{Xn}$$

We need label each token with a corresponding tag y_i by a special tagging scheme M:

$$Y = y_1, \mathrm{y}_2, \dots, \mathrm{y}_\mathrm{n}$$

Deep Learning Method

● NER: Sequence Labeling Methods

For a sequence labeling task, the input is a sentence X:

$$X = X_1, \mathrm{X}_2, \dots, \mathrm{Xn}$$

We need label each token with a corresponding tag y_i by a special tagging scheme M:

$$Y = y_1, \mathrm{y}_2, \dots, \mathrm{y}_\mathrm{n}$$

Tagging Schemes

- BIO format (Beginning, Inside, Outside)

  Every token is labeled as B-label if the token is the beginning of a named entity, I-label if it is inside a named entity but not the first token within the named entity, or O otherwise.

- BIESO format (Beginning, Inside, Ending, Single, Outside,)

  singleton entities (S)
  end of named entities (E)

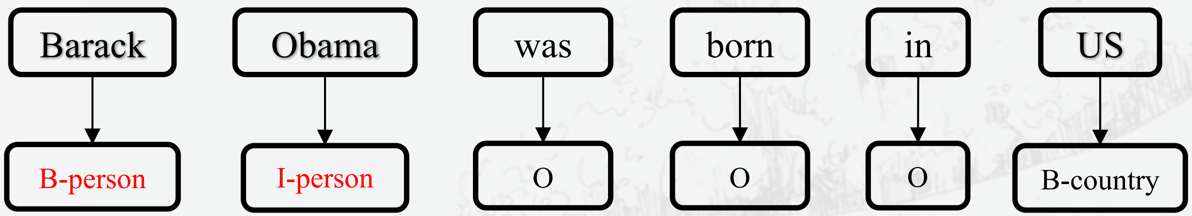
Deep Learning Method

• NER: Sequence Labeling Methods

Tagging Schemes

- BIO format (Beginning, Inside, Outside)

Every token is labeled as B-label if the token is the beginning of a named entity, I-label if it is inside a named entity but not the first token within the named entity, or O otherwise.



Entity： 【country】US 【label】[B-country]
【person】Barack Obama 【label】[B-person, I-person]

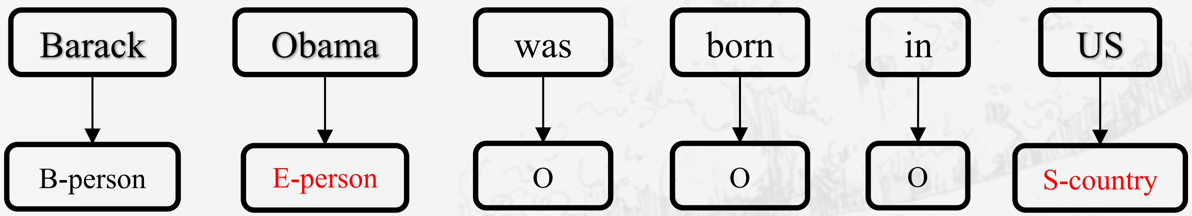
Deep Learning Method

- **NER: Sequence Labeling Methods**

Tagging Schemes

- BIESO format (Beginning, Inside, Ending, Single, Outside,)

  singleton entities (S)
  end of named entities (E)



Entity: 【country】US 【label】[S-country]

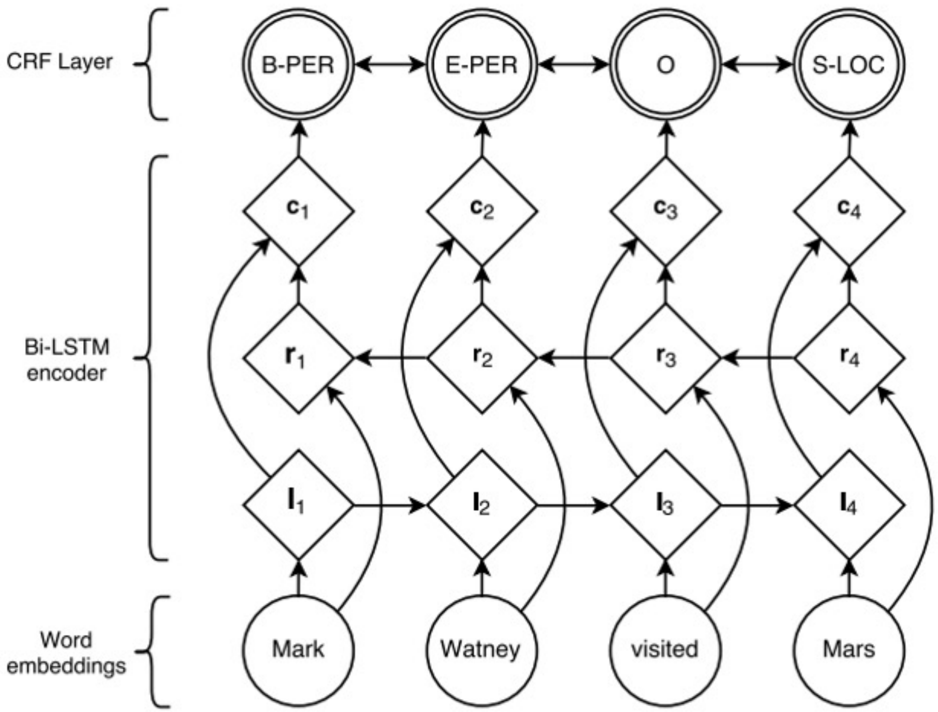
【person】Barack Obama 【label】[B-person, E-person]

Deep Learning Method

• NER: Sequence Labeling Methods

LSTM-CRF

CRF is a kind of discriminative probability model, which is often used to annotate or analyze sequence data, such as natural language text or biological sequence.

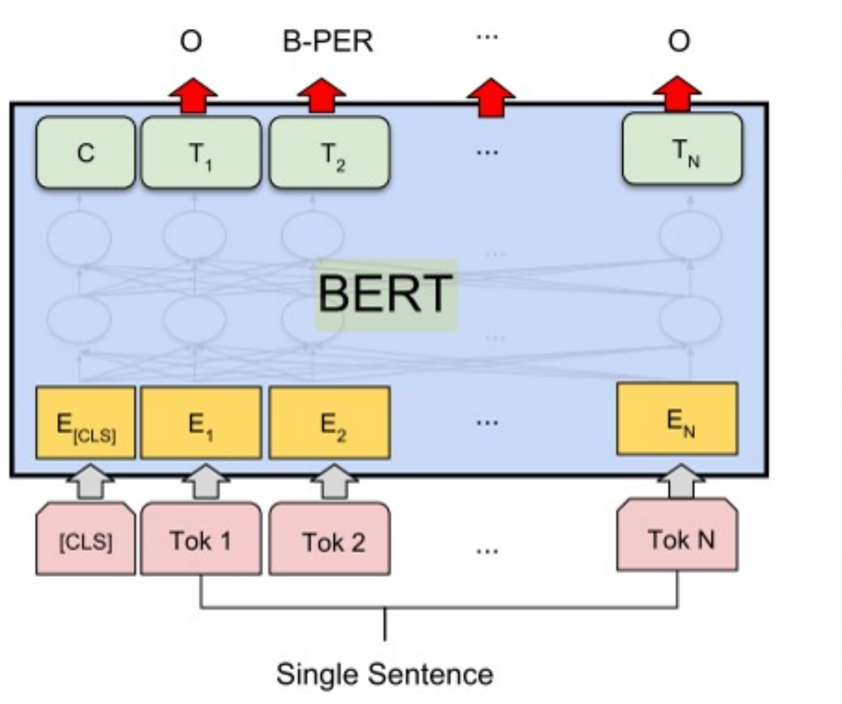


Deep Learning Method

- **NER: Sequence Labeling Methods**

LSTM Encoder ⇨ **BERT Encoder**

CRF Classifier ⇨ **Linear Classifier**



Deep Learning Method

• NER: Sequence Labeling Methods

Advantage

- Simple and effective.

Disadvantage

- Unable to model nested entities and non-continuous entities.

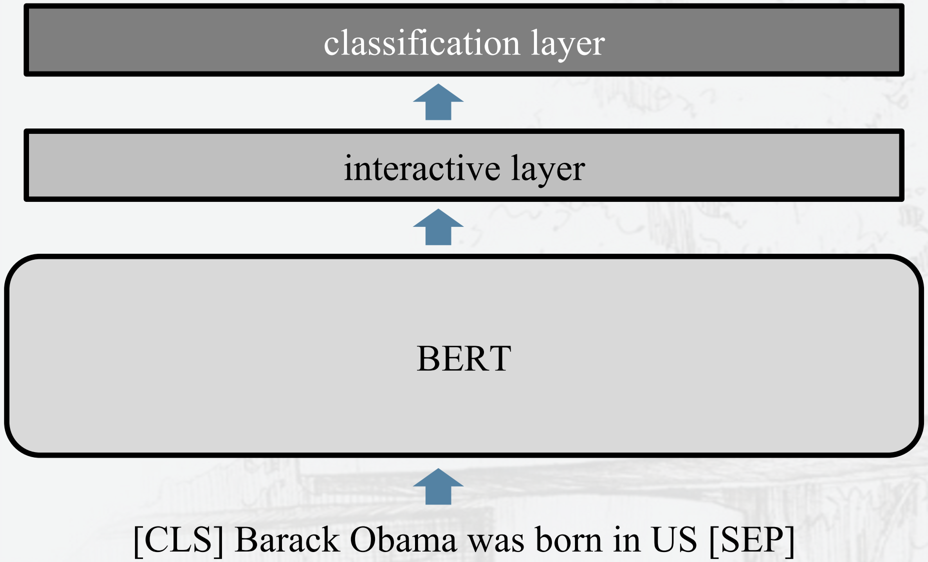
Reference

- Bidirectional LSTM-CRF for named entity recognition
- Chinese named entity recognition method based on BERT

Deep Learning Method

- **NER: Token-pair Methods**

Use the head token and the tail token to represent a span, and then connect with the interactive layer and classification layer to get the Multi-head matrix.



Token-pair Methods with BERT

Deep Learning Method

• NER: Token-pair Methods

Use the head token and the tail token to represent a span (so called token pairs), and then connect with the interactive layer and classification layer to get the Multi-head matrix.

The final result is a $[L, L, N]$ prediction matrix, where L is the length of the input text and N is the number of entity types.

The mathematical expression is $f(h_i, h_j, t) = score_{ij,t}$, where h represents the output of Encoder, and $score_{ij,t}$ represent the scores of the text span $(start = i, end = j)$ on the t-th entity class.

Deep Learning Method

• NER: Token-pair Methods

The mathematical expression is $f(h_i, h_j, t) = score_{ij,t}$, where h represents the output of Encoder, and $score_{ij,t}$ represent the scores of the text span $(start = i, end = j)$ on the t-th entity class.

start	1	0	0	0	0	0	0
end	0	1	0	0	0	0	0
	Barack	Obama	was	born	in	the	us

	Barack	Obama	was	born	in	the	us
Barack	0	1	0	0	0	0	0
Obama		0	0	0	0	0	0
was			0	0	0	0	0
born				0	0	0	0
in					0	0	0
the						0	0
us							0

The person head matrix is located at position (0,1) with a value of 1, indicating that ”Barack Obama" is the person.

Deep Learning Method

- **NER: Token-pair Methods**

Token-pair can naturally solve the problems of nested NER.

	The	Lincoln	Memorial
The	0	0	1
Lincoln		0	0
Memorial			0

Detect the location name: The Lincoln Memorial

	The	Lincoln	Memorial
The	0	0	0
Lincoln		1	0
Memorial			0

Detect the person name: Lincoln

Deep Learning Method

• NER: Token-pair Methods

The problem of discontinuity can also be solved by setting the head.

For example, here we set an additional "product name middle" head to extract the product name middle as a separate entity. Then, subtracting the overlapping part of the product name in index from the extracted product name middle will give us the discontinuous product name.

	have	much	muscle	pain	and	fatigue
have	0	0	0	0	0	0
much		0	0	0	0	0
muscle			0	0	0	1
pain				0	0	0
and					0	0
fatigue						0

The disorder head

	have	much	muscle	pain	and	fatigue
have	0	0	0	0	0	0
much		0	0	0	0	0
muscle			0	0	0	0
pain				0	1	0
and					0	0
fatigue						0

The middle of the disorder head

Disorder name : muscle pain and fatigue

\-

Middle of Disorder name: pain and

=

Disorder Entity: muscle fatigue

NER: Token-pair Methods

Three methods to model the interactive layer and classification layer.

Multiplicative

- "GlobalPointer: Handling Nested and Non-Nested NER in a Unified Framework"
- Calculation formula: $f(h_i, h_j, t) = q_{i,t}^T k_{j,t}$, where $q_{i,t} = W_{i,t} h_i + b_{i,t}$, $k_{j,t} = W_{j,t} h_j + b_{j,t}$

Additive

- Paper: "Joint entity recognition and relation extraction as a multi-head selection problem"
- Calculation formula: $f(h_i, h_j, t) = W_t h_{ij} + b_t$, where $h_{ij} = \tanh(W_h [h_i \oplus h_j] + b_h)$, $\oplus$ represents concatenation operation.

Bi-affine

- "Named Entity Recognition as Dependency Parsing"
- Calculation formula: $f(h_i, h_j, t) = h_i^T U_t h_j + W_t [h_i \oplus h_j] + b_t$

Deep Learning Method

- **NER: Token-pair Methods**

Advantage

- **Flat Entities, Nested Entities and discontinuous entities can be handled flexibly.**

Disadvantage

- **Need to generate a header for each entity category, modeling is complex.**

Reference

- GlobalPointer: Handling Nested and Non-Nested NER in a Unified Framework
- Joint entity recognition and relation extraction as a multi-head selection problem
- Named Entity Recognition as Dependency Parsing

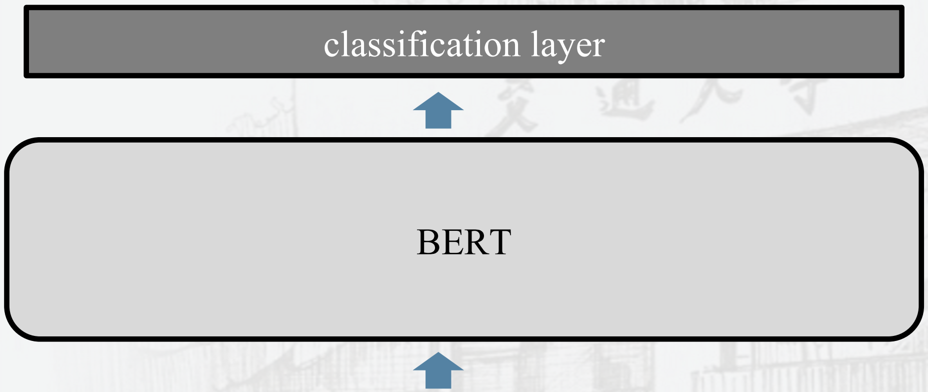
• NER: MRC Methods

Convert NER tasks into MRC tasks, each entity type is represented by natural language queries, and entities are extracted by answering these queries.

As shown in the figure, the predicted start and end are the start and end of the corresponding entity.

start				1	0	0	0	0	0	
end				0	1	0	0	0	0	
	[CLS]	What is the **person** name in the text?	[SEP]	Barack	Obama	was	born	in	US	[SEP]

Detect the person name: Barack Obama



[CLS] What is the person name in the text? [SEP] Barack Obama was born in US [SEP]

Deep Learning Method

• NER: MRC Methods

Convert NER tasks into MRC tasks, each entity type is represented by natural language queries, and entities are extracted by answering these queries.

As shown in the figure, the predicted start and end are the start and end of the corresponding entity.

start				1	0	0	0	0	0	
end				0	1	0	0	0	0	
	[CLS]	What is the **person** name in the text?	[SEP]	Barack	Obama	was	born	in	US	[SEP]

Detect the person name: Barack Obama

Unlike Token-pair, there is only one head here, and it is independent of the entity type, so it is also transferable.

NER: MRC Methods

MRC Methods can naturally solve the problem of nested NER.

start				0	1	0	
end				0	1	0	
	[CLS]	What is in the **person** name in the text?	[SEP]	The	Lincoln	Memorial	[SEP]

Detect the person name: Lincoln

start				1	0	0	
end				0	0	1	
	[CLS]	What is the **location** name in the text?	[SEP]	The	Lincoln	Memorial	[SEP]

Detect the location name: The Lincoln Memorial

Deep Learning Method

• NER: MRC Methods

Dealing with discontinuous problems can be solved by setting up questions.

start				0	0	1	0	0	0	
end				0	0	0	0	0	1	
		What is disorder name in the text?		have	much	muscle	pain	and	fatigue	[SEP]

Detect the disorder name

start				0	0	0	1	0	0	
end				0	0	0	0	1	0	
	[CLS]	What is in the middle of the disorder name in the text?	[SEP]	have	much	muscle	pain	and	fatigue	[SEP]

Detect the middle of the disorder name

Disorder name : muscle pain and fatigue

\-

Middle of Disorder name: pain and

=

Disorder Entity: muscle fatigue

Deep Learning Method

- **NER: MRC Methods**

Advantage

- **Three different types of entities can be handled flexibly.**
- **It can handle unseen entity categories.**

Disadvantage

- **Need to add a query to each category to increase the inference complexity.**
- **The choice of query affects the effect of prediction.**

Reference

- Entity-relation extraction as multi-turn question answering
- A Unified MRC Framework for Named Entity Recognition

Deep Learning Method

- **NER: Generative methods**

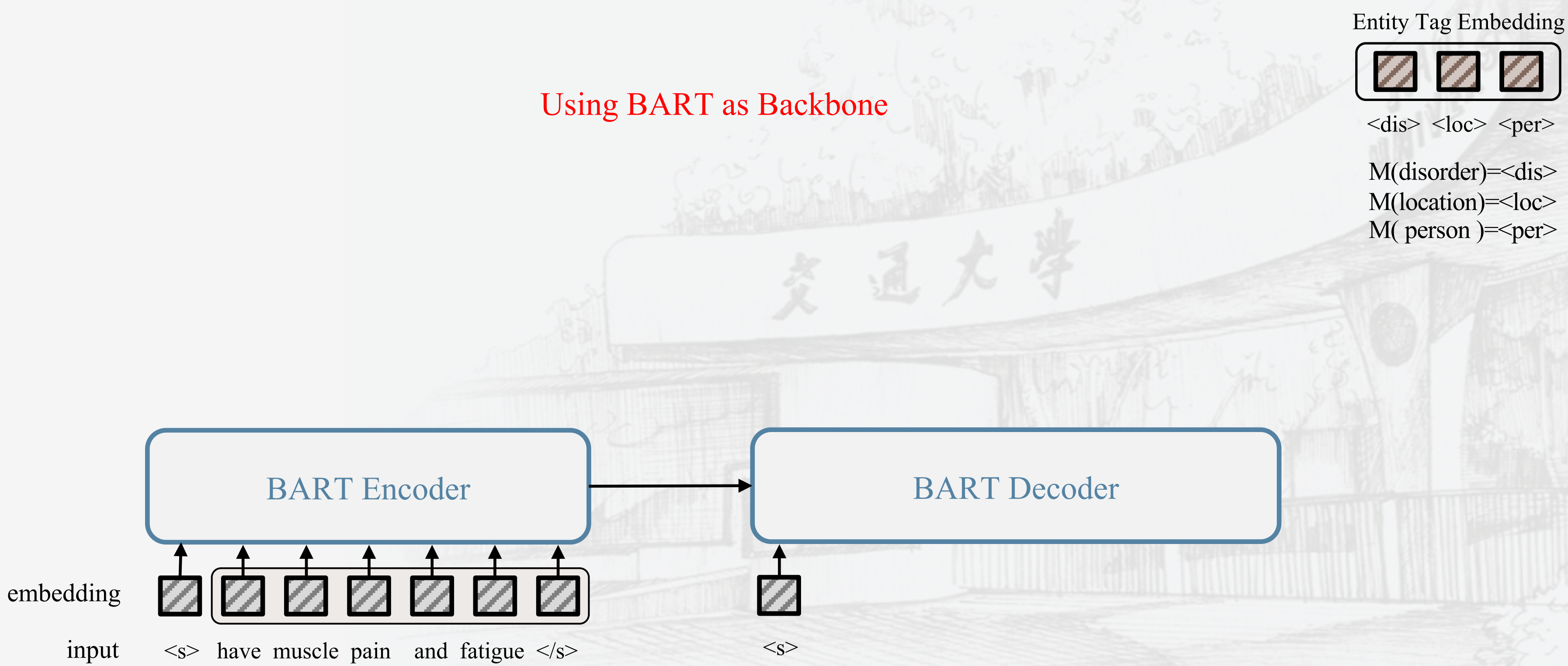
Using the pointer method, the labeling task is transformed into a sequence generation task, and the seq2seq paradigm is used for generation.

Deep Learning Method

- ## NER: Generative methods

Using the pointer method, the labeling task is transformed into a sequence generation task, and the seq2seq paradigm is used for generation.

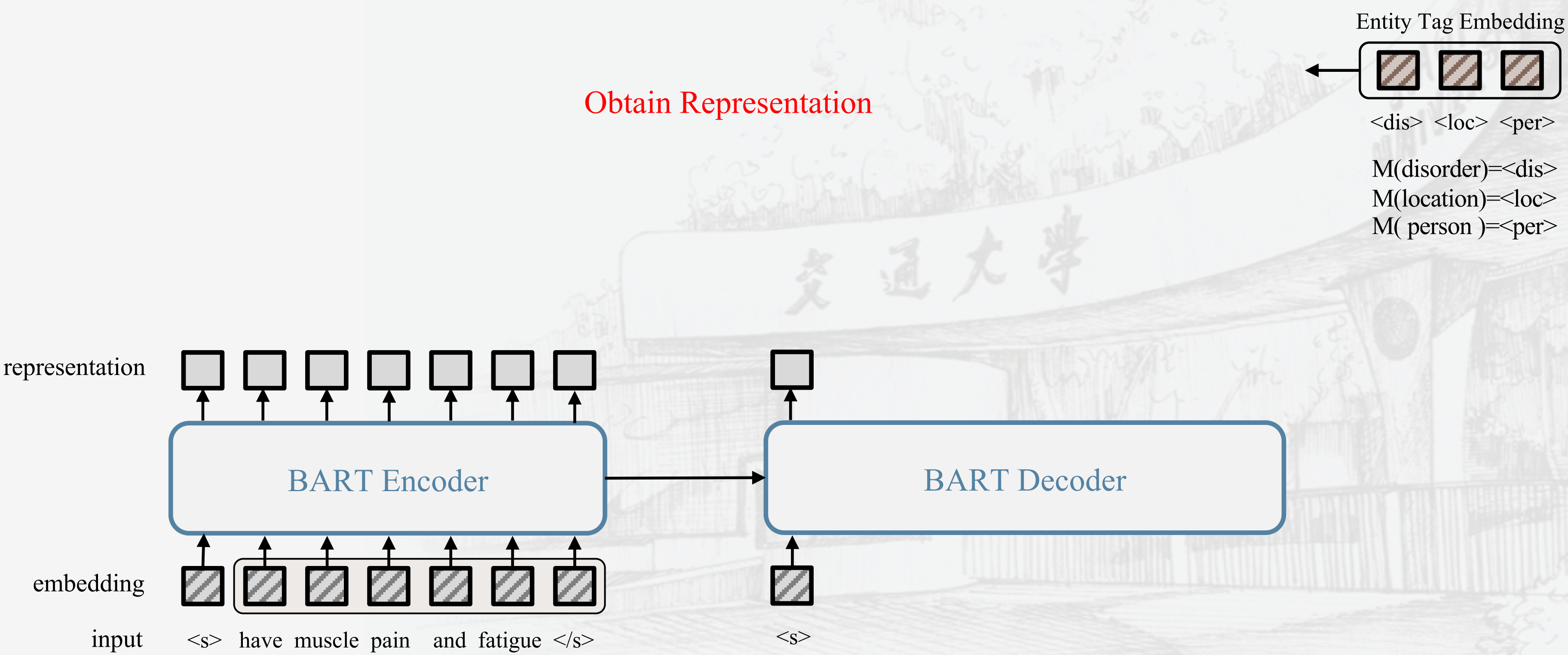
Input: <s> have muscle pain and fatigue </s>
Output:



- **NER: Generative methods**

Using the pointer method, the labeling task is transformed into a sequence generation task, and the seq2seq paradigm is used for generation.

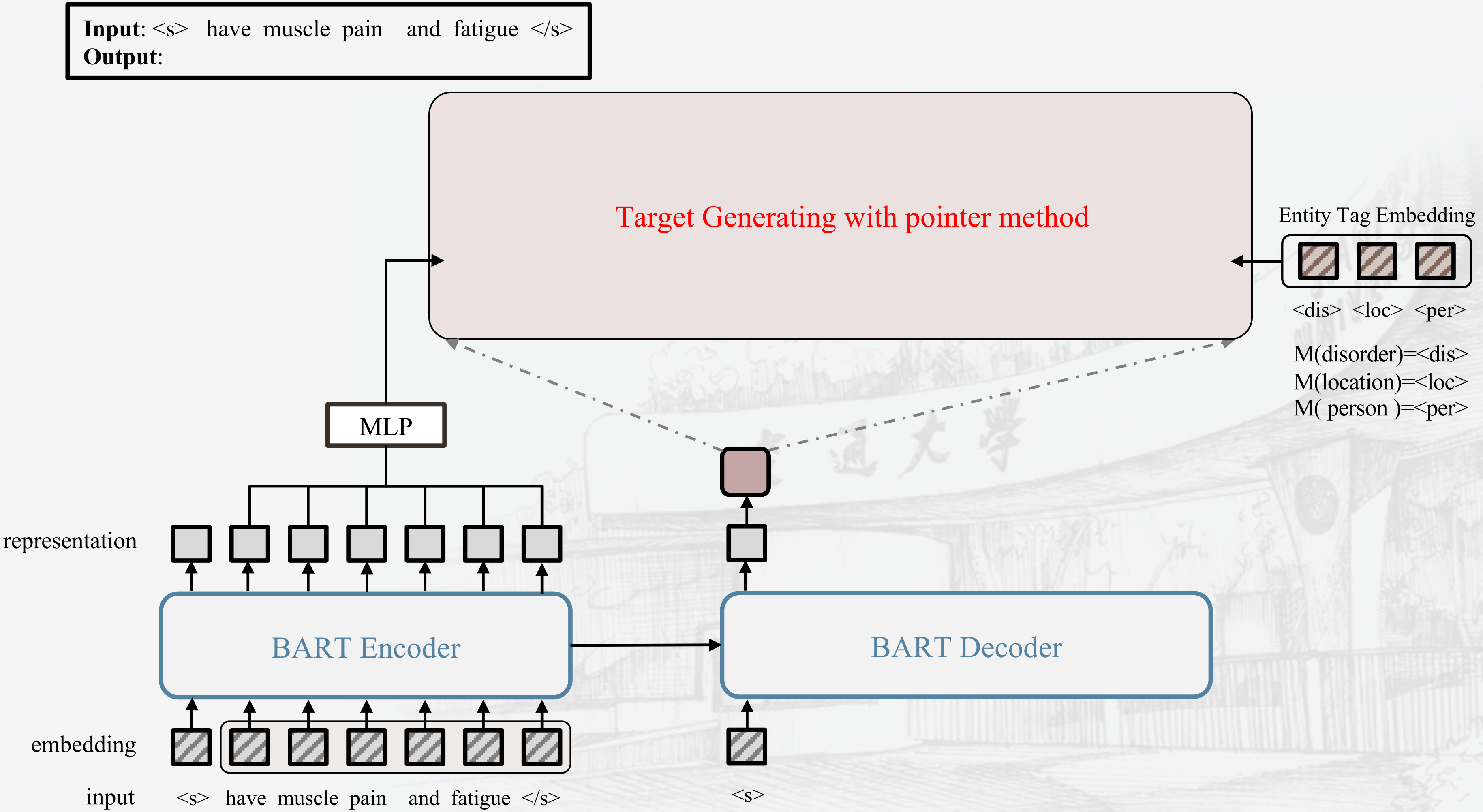
Input: <s> have muscle pain and fatigue </s>
Output:



Deep Learning Method

• NER: Generative methods

Using the pointer method, the labeling task is transformed into a sequence generation task, and the seq2seq paradigm is used for generation.

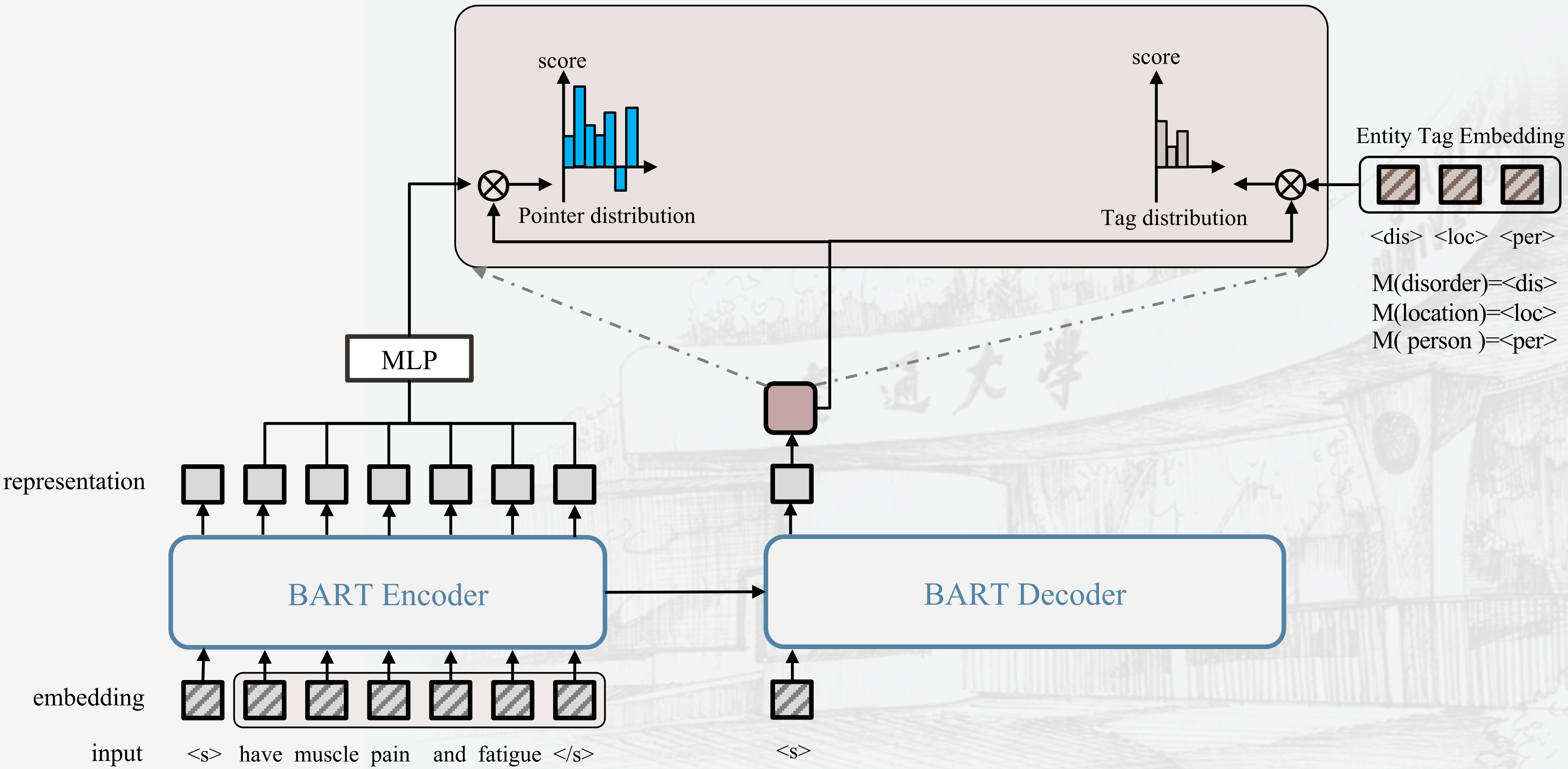


Deep Learning Method

• NER: Generative methods

Using the pointer method, the labeling task is transformed into a sequence generation task, and the seq2seq paradigm is used for generation.

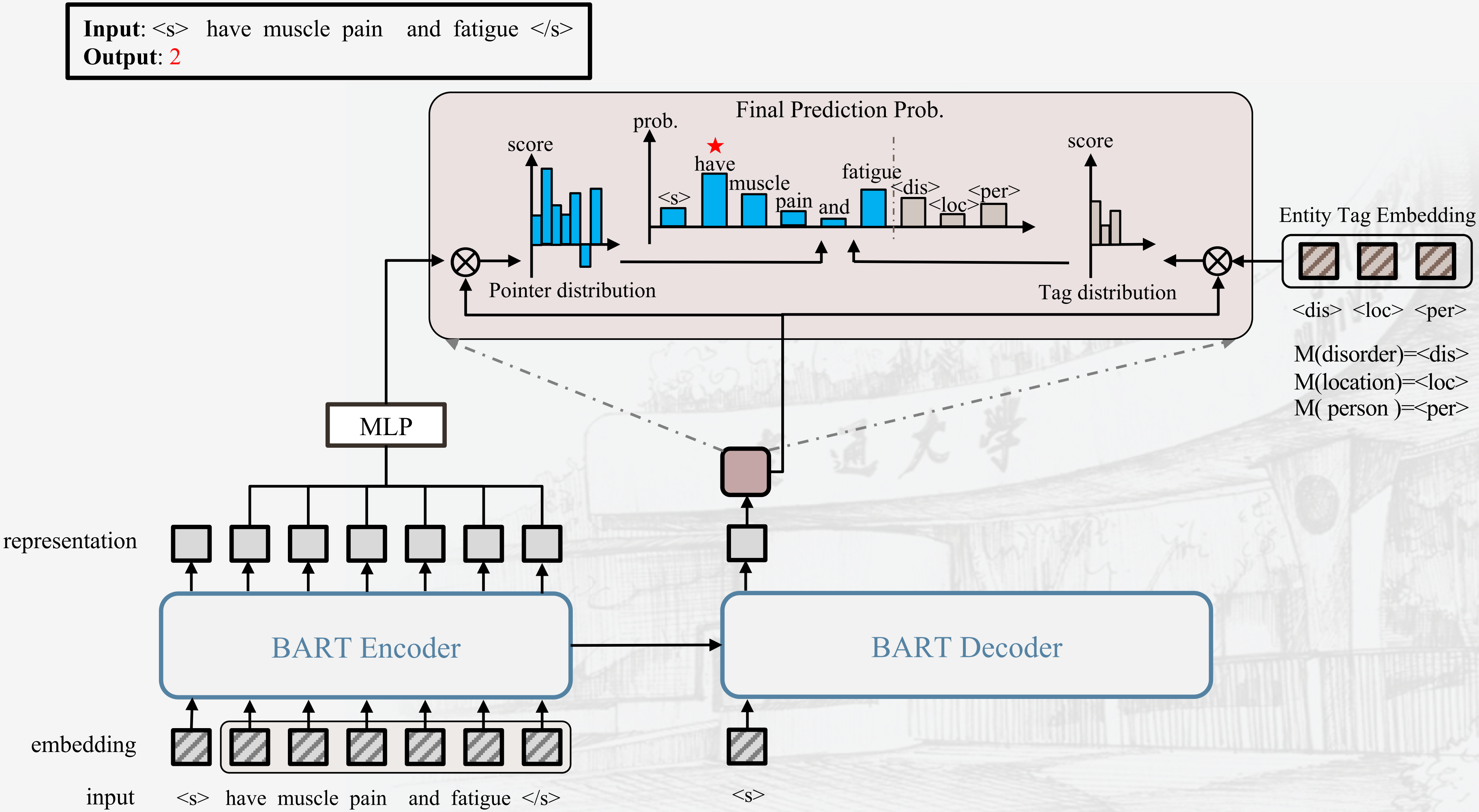
Input: <s> have muscle pain and fatigue </s>
Output:



Deep Learning Method

• NER: Generative methods

Using the pointer method, the labeling task is transformed into a sequence generation task, and the seq2seq paradigm is used for generation.

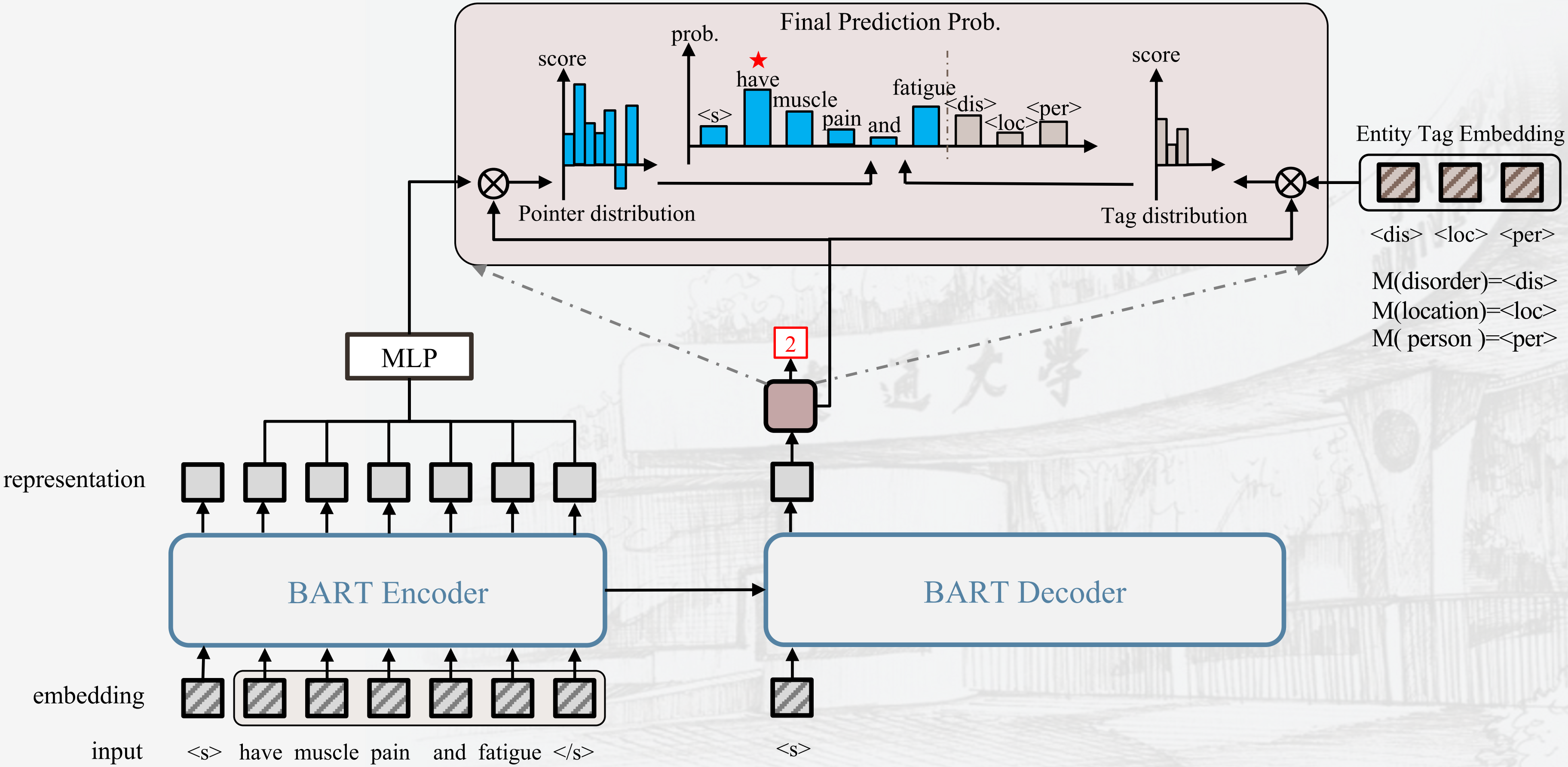


Deep Learning Method

- ## NER: Generative methods

Using the pointer method, the labeling task is transformed into a sequence generation task, and the seq2seq paradigm is used for generation.

Input: <s> have muscle pain and fatigue </s>
Output: 2

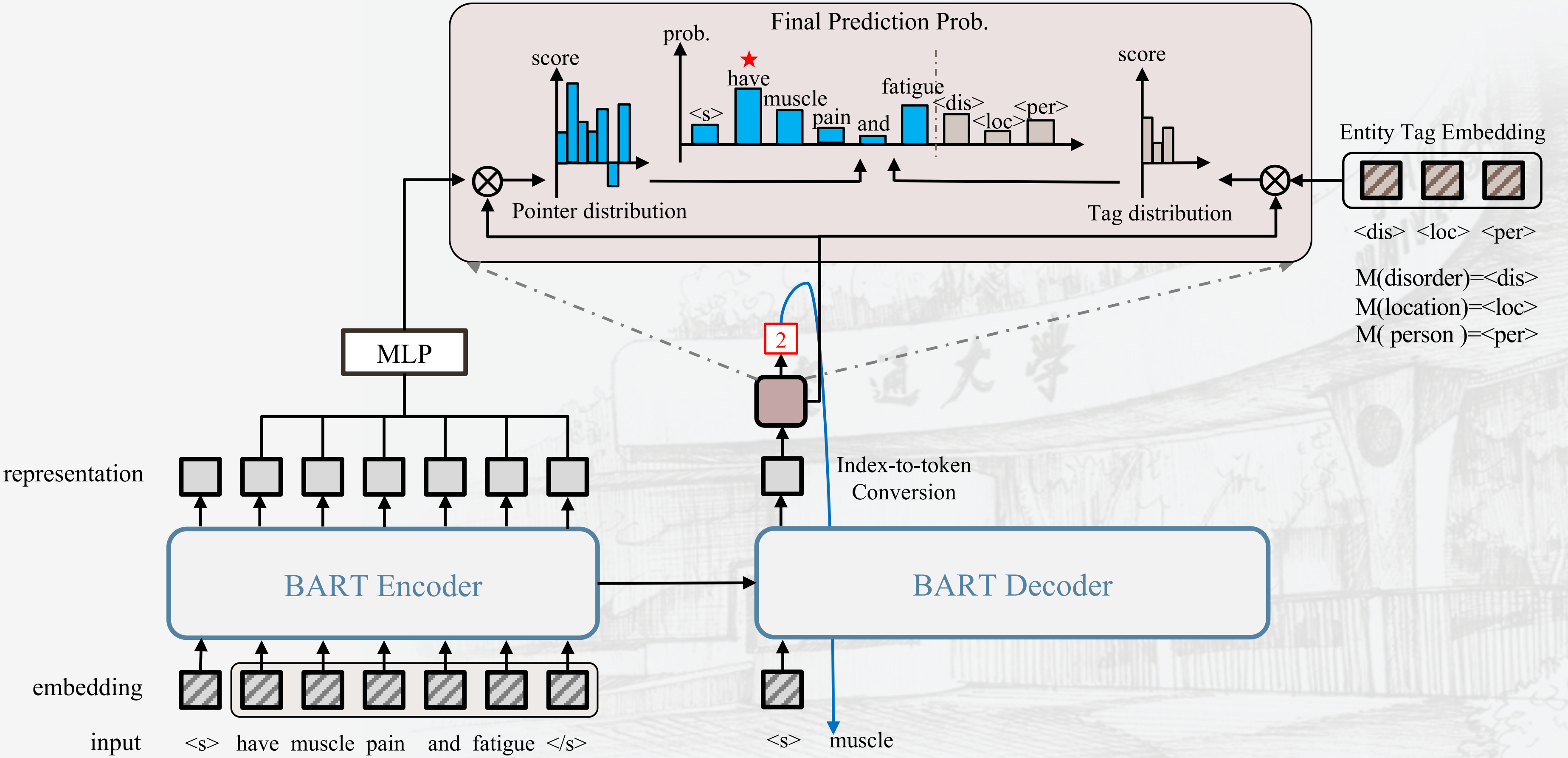


Deep Learning Method

• NER: Generative methods

Using the pointer method, the labeling task is transformed into a sequence generation task, and the seq2seq paradigm is used for generation.

Input: <s> have muscle pain and fatigue </s>
Output: 2

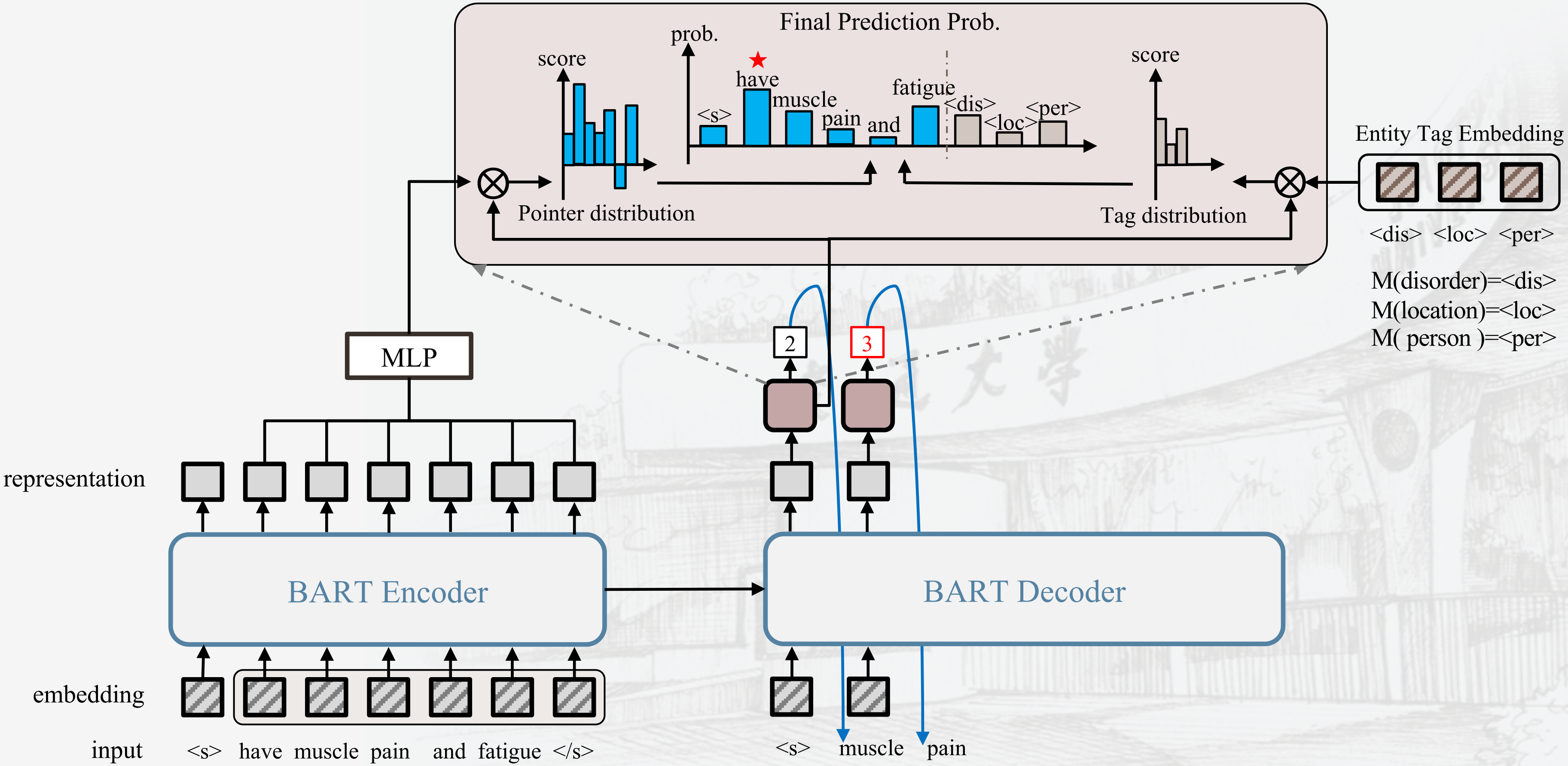


Deep Learning Method

• NER: Generative methods

Using the pointer method, the labeling task is transformed into a sequence generation task, and the seq2seq paradigm is used for generation.

Input: <s> have muscle pain and fatigue </s>
Output: 2 3

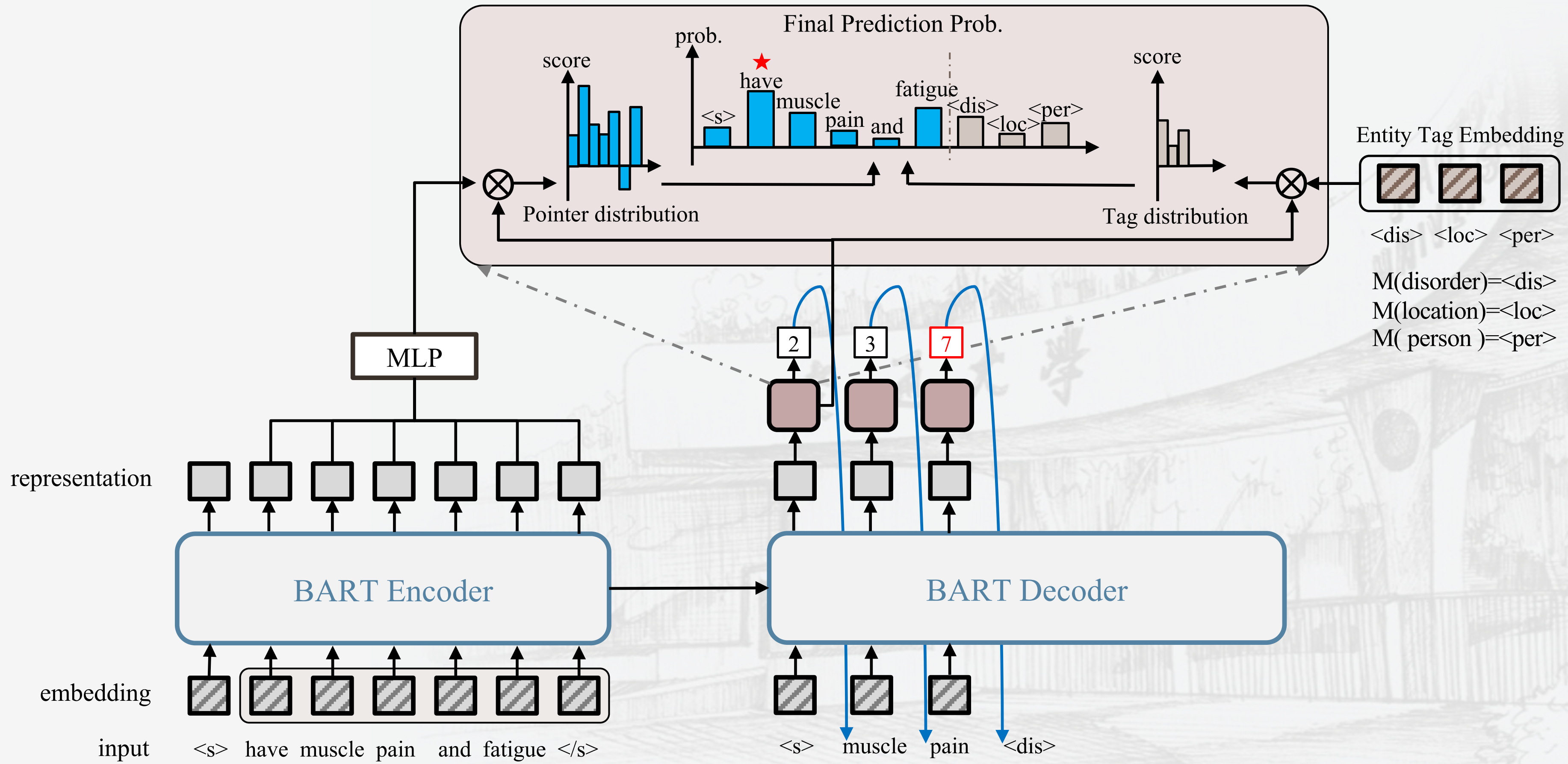


Deep Learning Method

- **NER: Generative methods**

Using the pointer method, the labeling task is transformed into a sequence generation task, and the seq2seq paradigm is used for generation.

Input: <s> have muscle pain and fatigue </s>
Output: 2 3 7

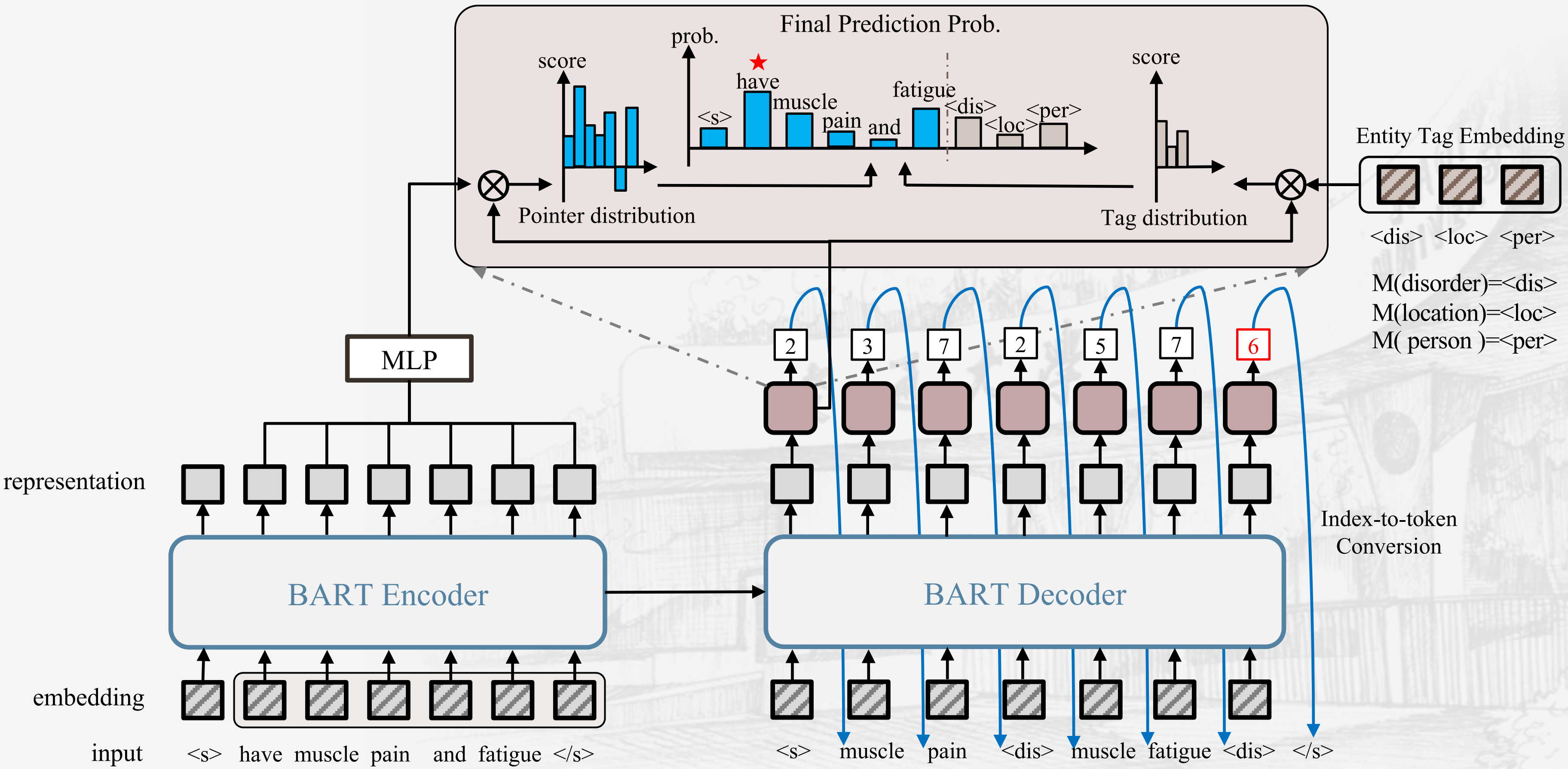


Deep Learning Method

• NER: Generative methods

Using the pointer method, the labeling task is transformed into a sequence generation task, and the seq2seq paradigm is used for generation.

Input: <s> have muscle pain and fatigue </s>
Output: 2 3 7 2 5 7 6



Deep Learning Method

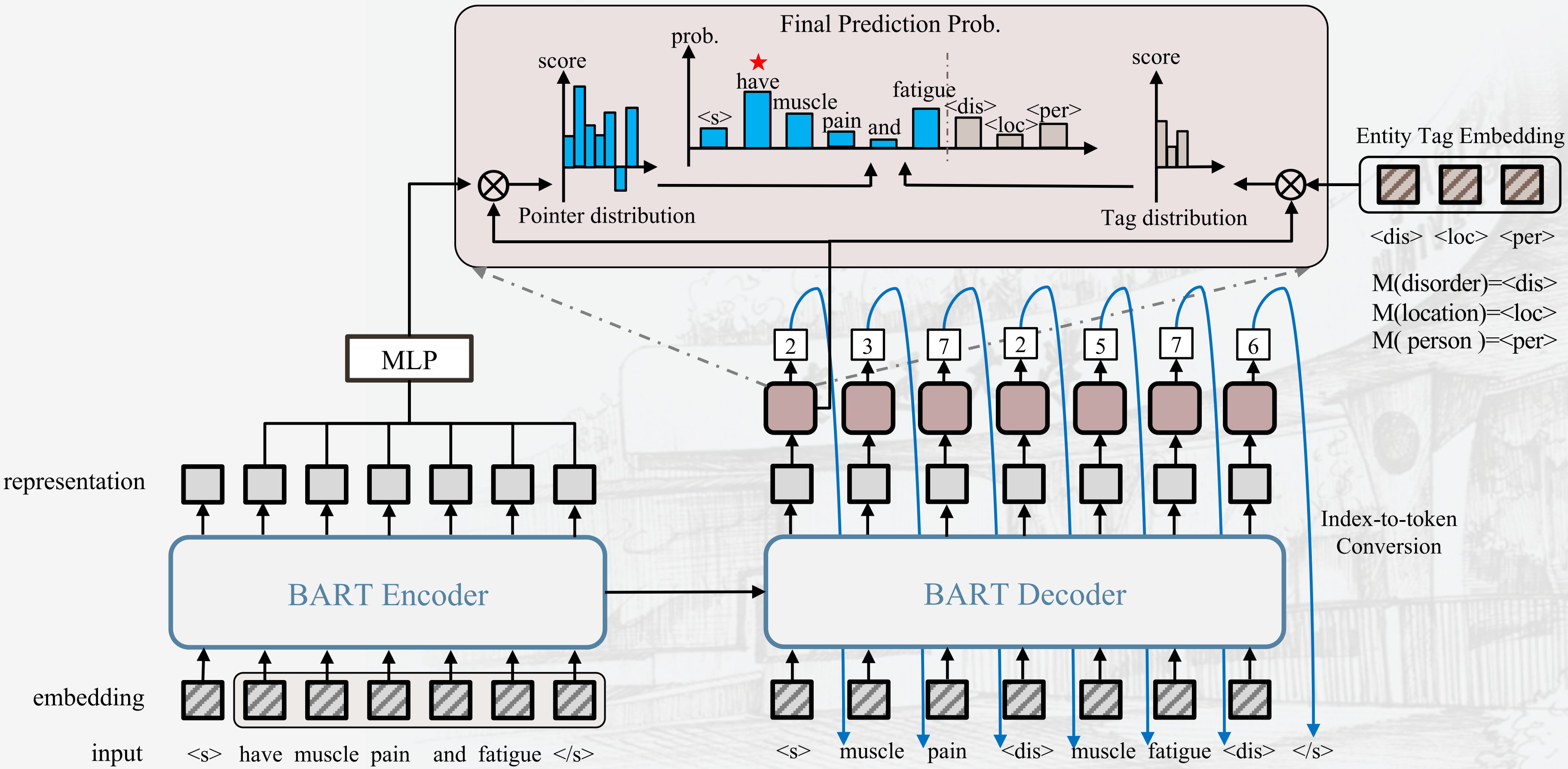
• NER: Generative methods

Using the pointer method, the labeling task is transformed into a sequence generation task, and the seq2seq paradigm is used for generation.

Input: <s> have muscle pain and fatigue </s>
Output: 2 3 7 2 5 7 6

→

Input: <s> have muscle pain and fatigue </s>
Output: disorder: “muscle pain”, “muscle fatigue”



Deep Learning Method

- **NER: Generative methods**

Advantage

- **Three different types of entities can be handled flexibly.**
- **It also has excellent performance when the sample size is small.**

Disadvantage

- **The inference of generative methods is slow.**
- **Modeling is more complicated.**

Reference

- A Unified Generative Framework for Various NER Subtasks
- LightNER: A Lightweight Tuning Paradigm for Low-resource NER via Pluggable Prompting