Paper Seminar

Syntax and Semantics in Language Model Representation

Myeongsup Kim

Integrated M.S./Ph.D. Student
Data Science & Business Analytics Lab.
School of Industrial Management Engineering
Korea University

Myeongsup_kim@korea.ac.kr

A Structural Probe for Finding Syntax in Word Representation

Hewitt and Manning, 2019, NAACL

Visualizing and Measuring the Geometry of BERT

Coenen et al., 2019, NIPS

Introduction

- Concept of Language Model

<What This Seminar Does Not Cover>

- **Details of ELMo**

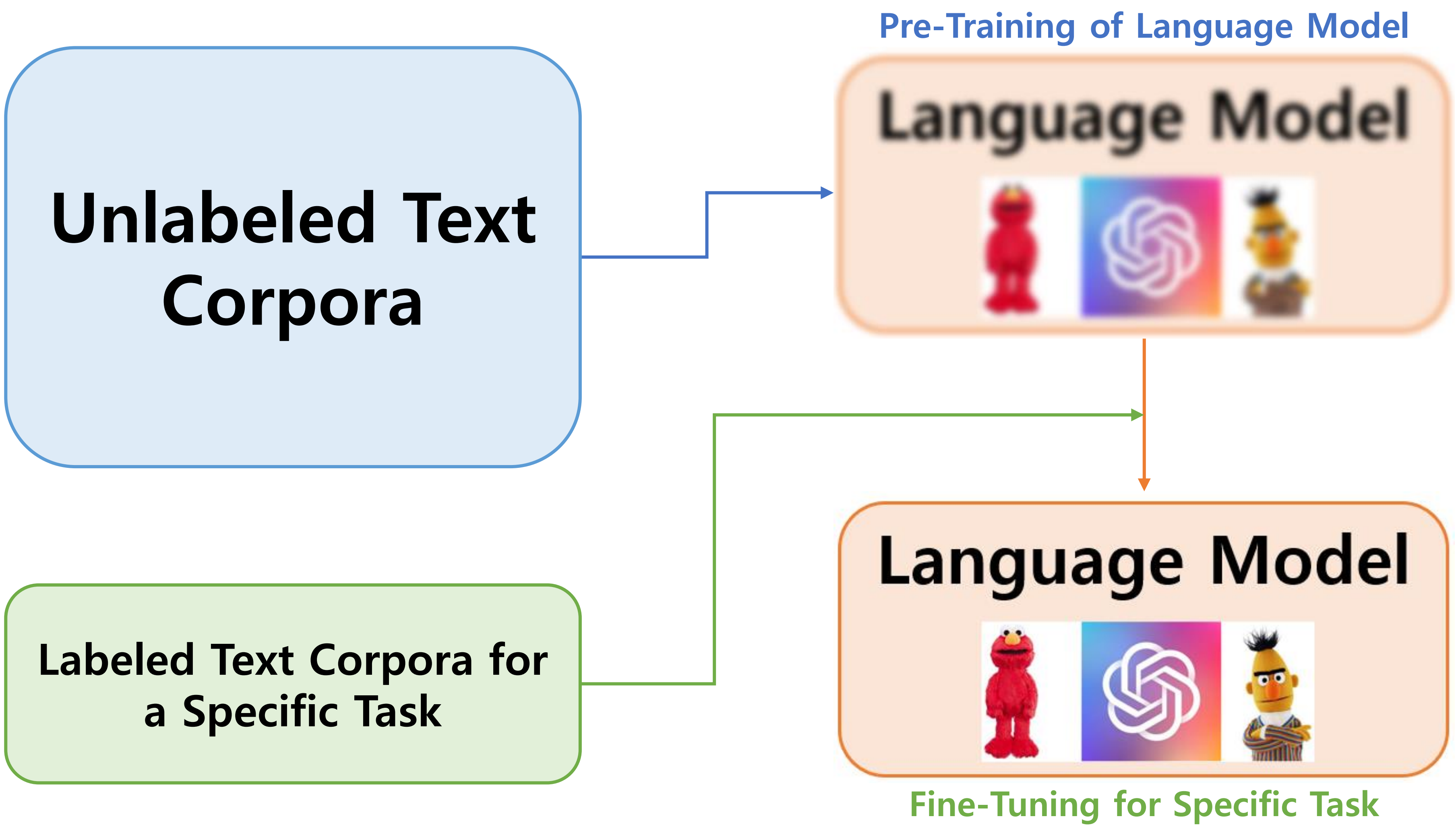
  Peters et al., 2018, Deep Contextualized Word Representations, NAACL

- **Details of BERT**

  Devlin et al., 2019, BERT: Pre-Training of Deep Bidirectional Transformers for Language Understanding, NAACL

From Text Analytics (Pilsung Kang, 2020)

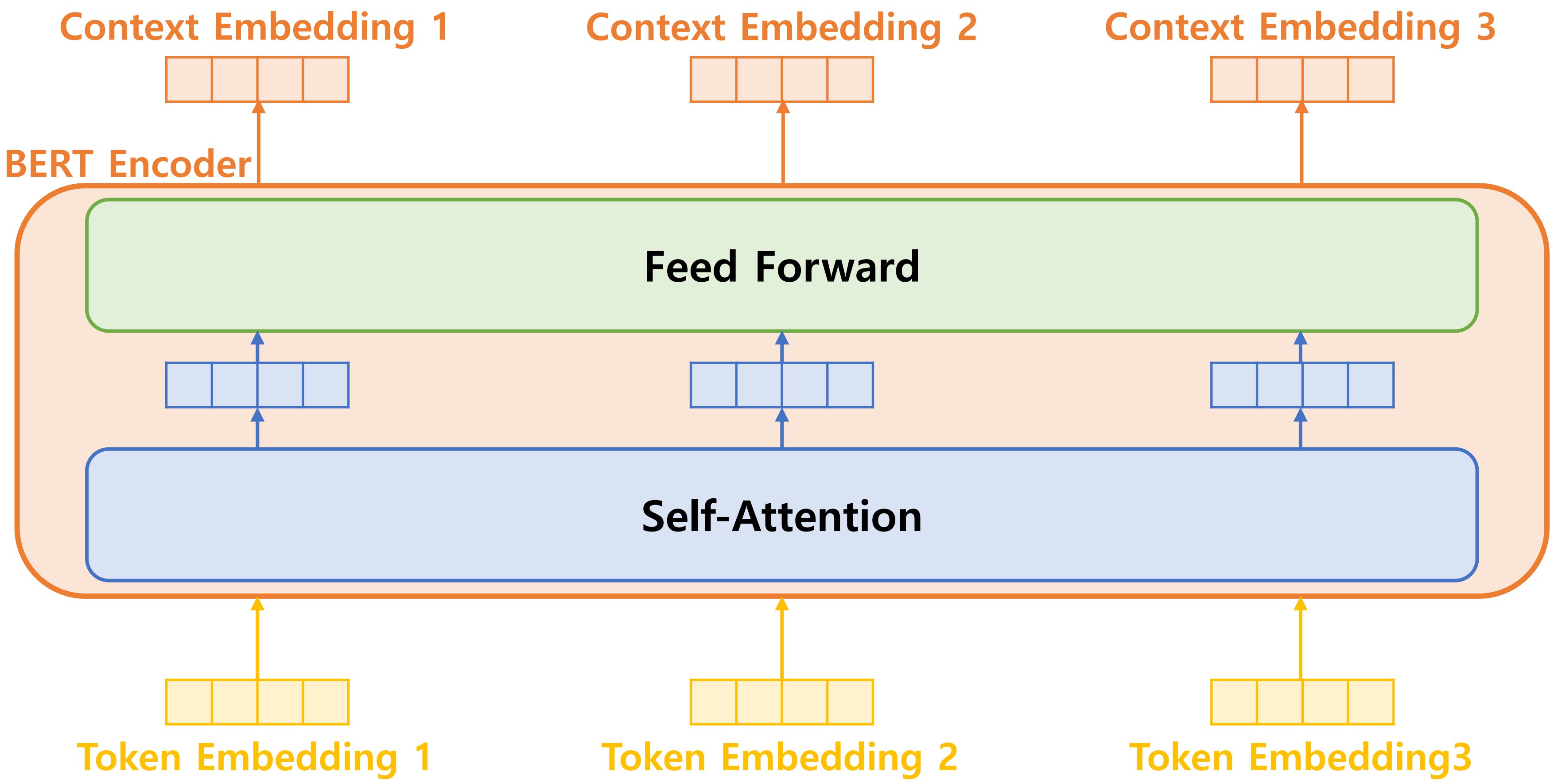
<Concept of Language Model : Motivation>



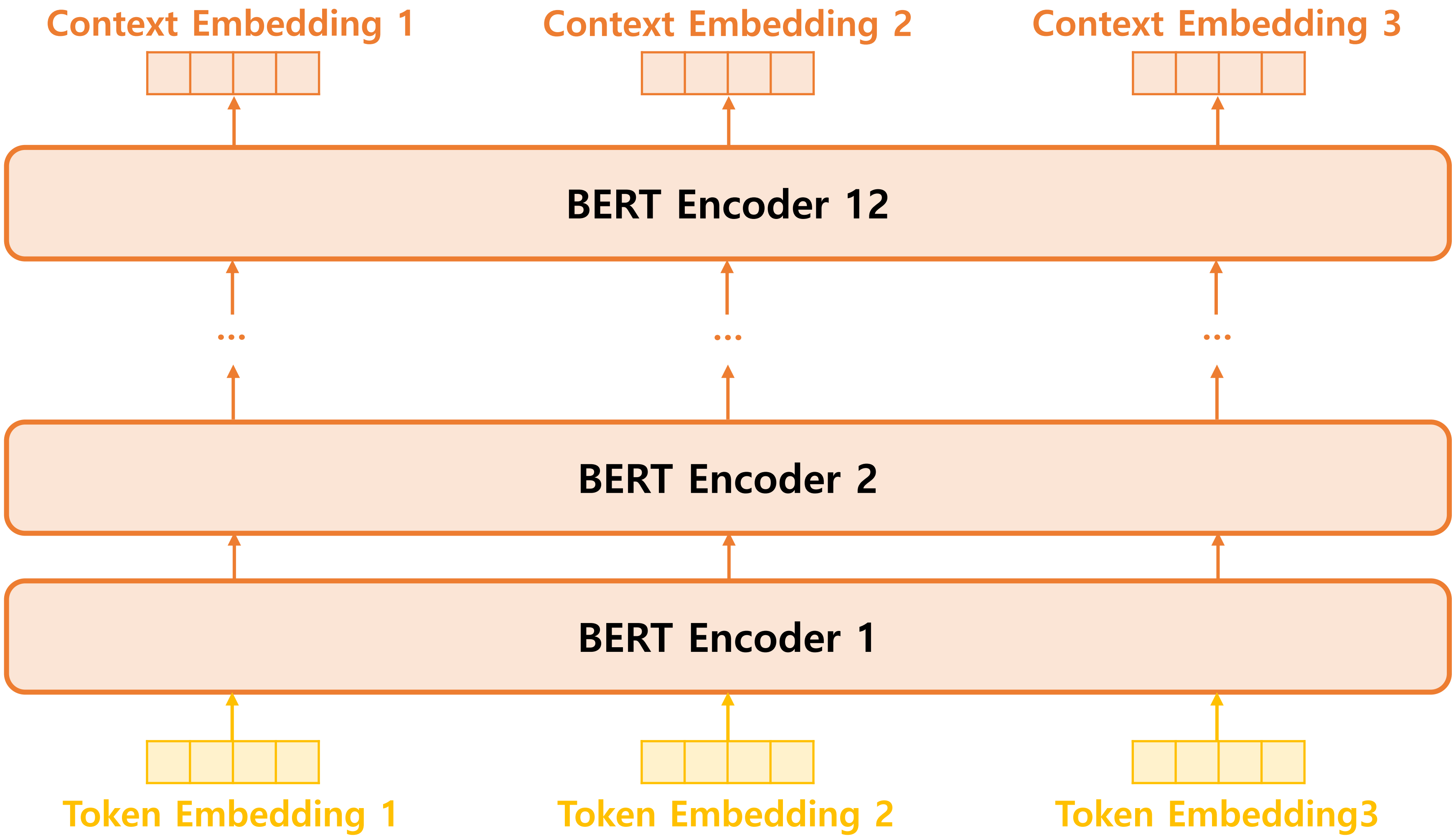
Introduction

-Concept of Language Model

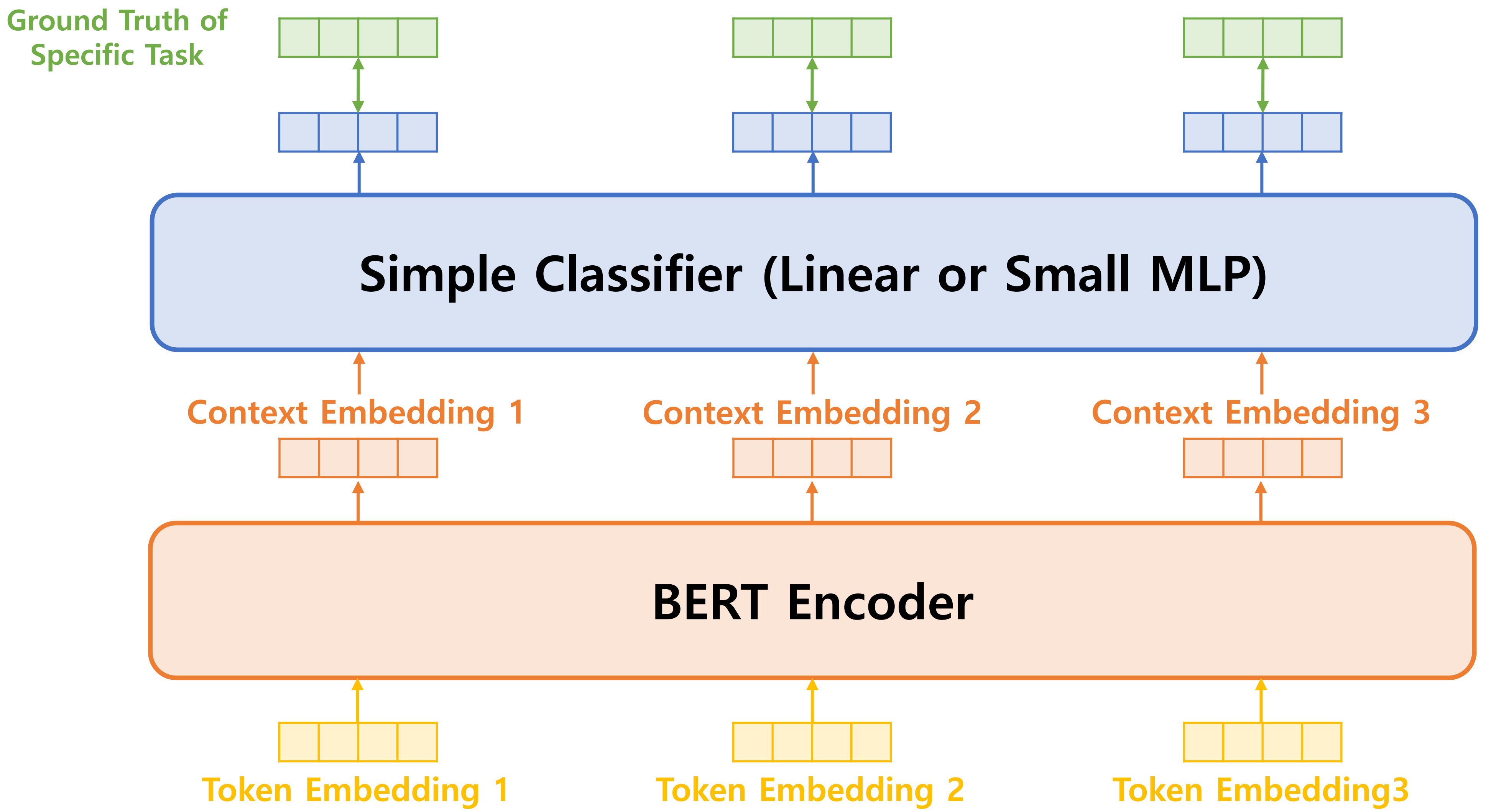
<Concept of Language Model : Pre-Training>



<Concept of Language Model : Pre-Training>



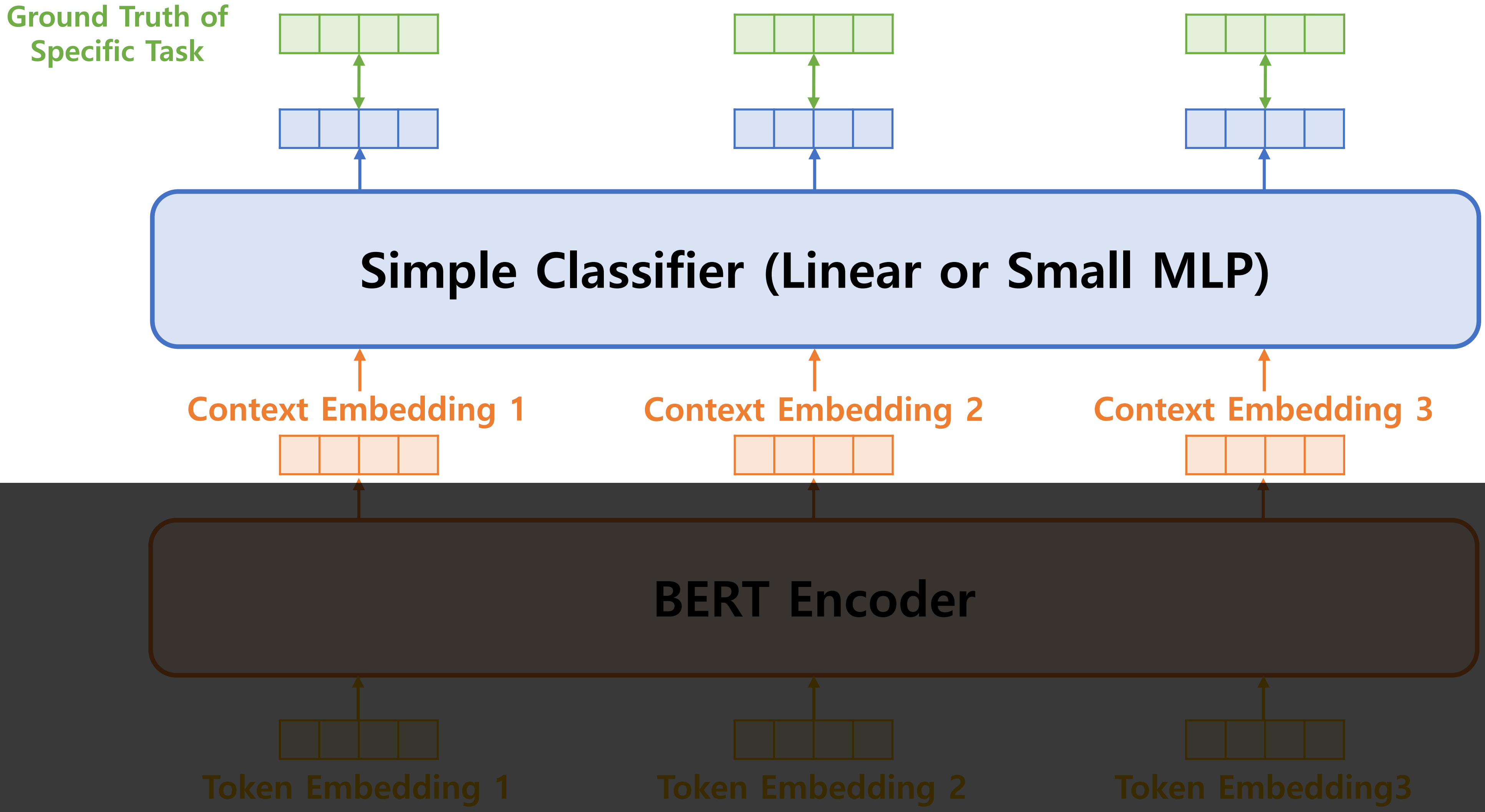
<Concept of Language Model : Fine-tuning>



Introduction

-Concept of Language Model

<Concept of Language Model : Fine-tuning>



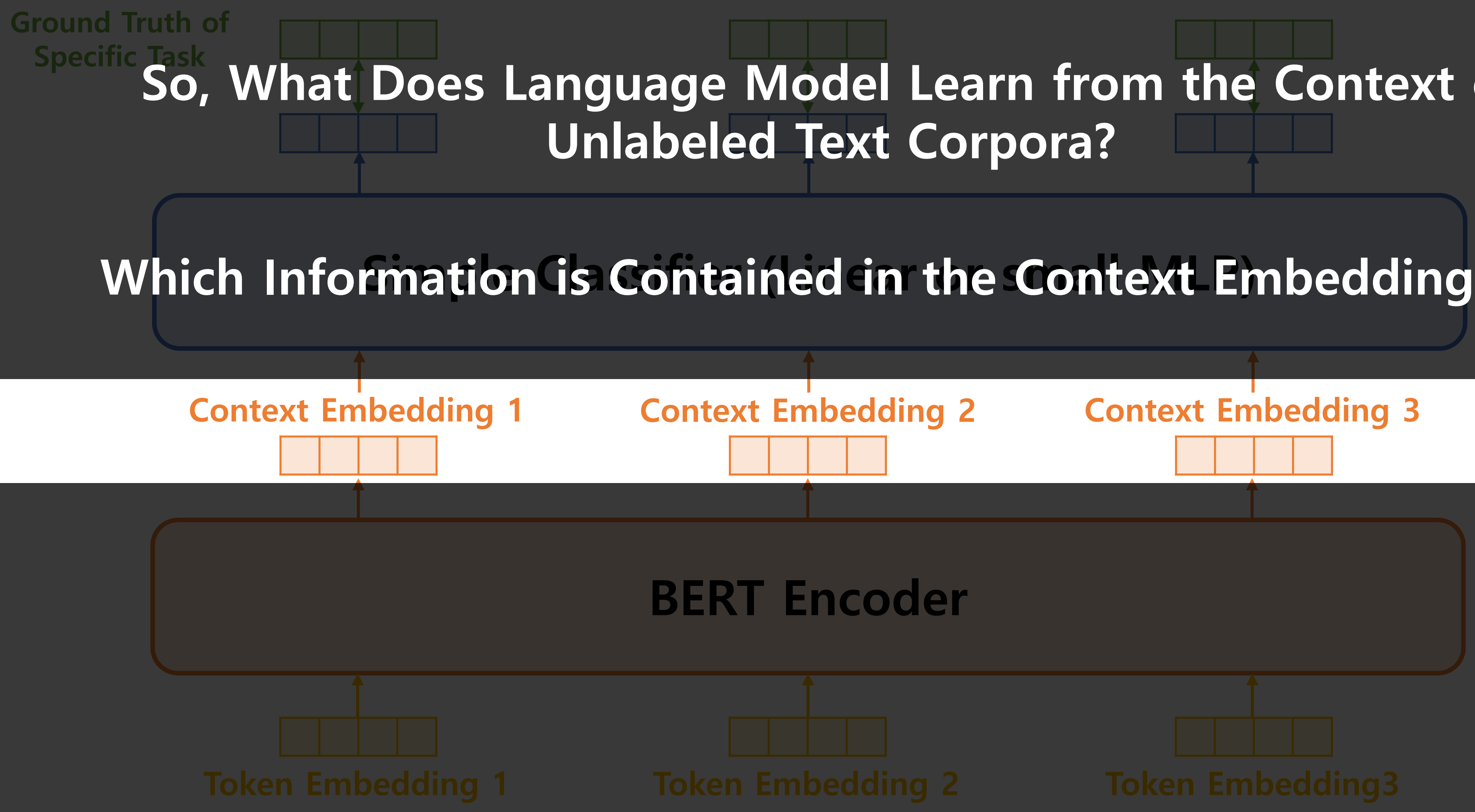
Introduction

-Concept of Language Model

<Concept of Language Model : Fine-tuning>

So, What Does Language Model Learn from the Context of Unlabeled Text Corpora?

Which Information is Contained in the Context Embeddings?



A Structural Probe for Finding Syntax in Word Representation

Hewitt and Manning, 2019, NAACL

A Structural Probe for Finding Syntax in Word Representation

Hewitt and Manning, 2019, NAACL

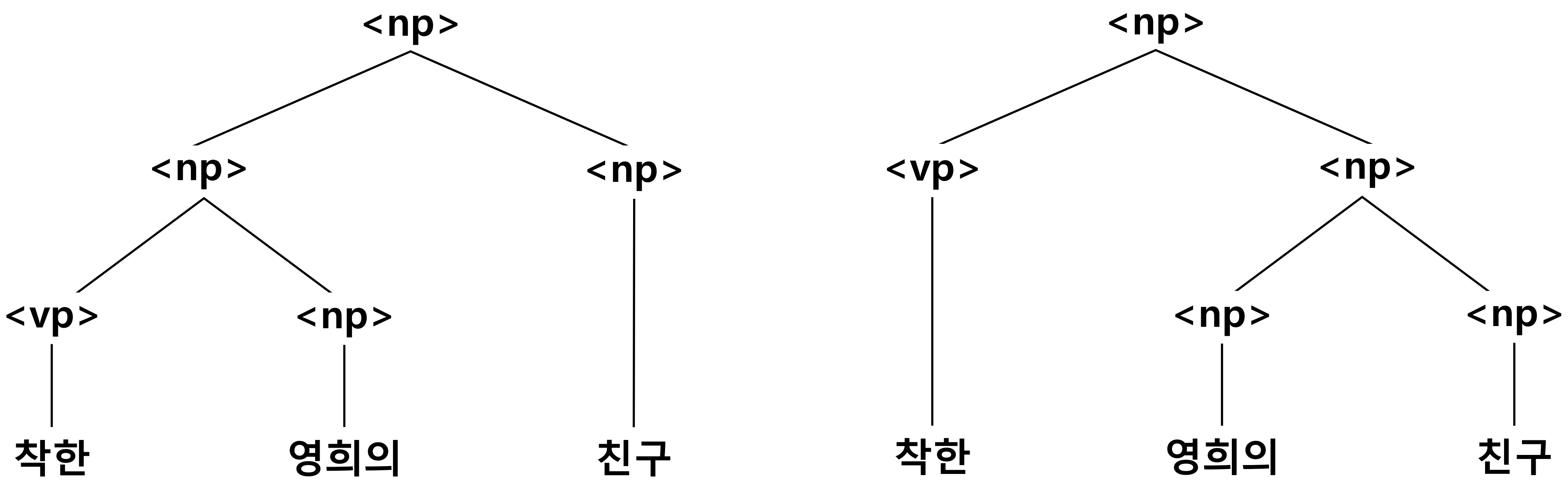
Pre-requisites

- Dependency Parse Tree

Pre-requisites

-Dependency Parse Tree

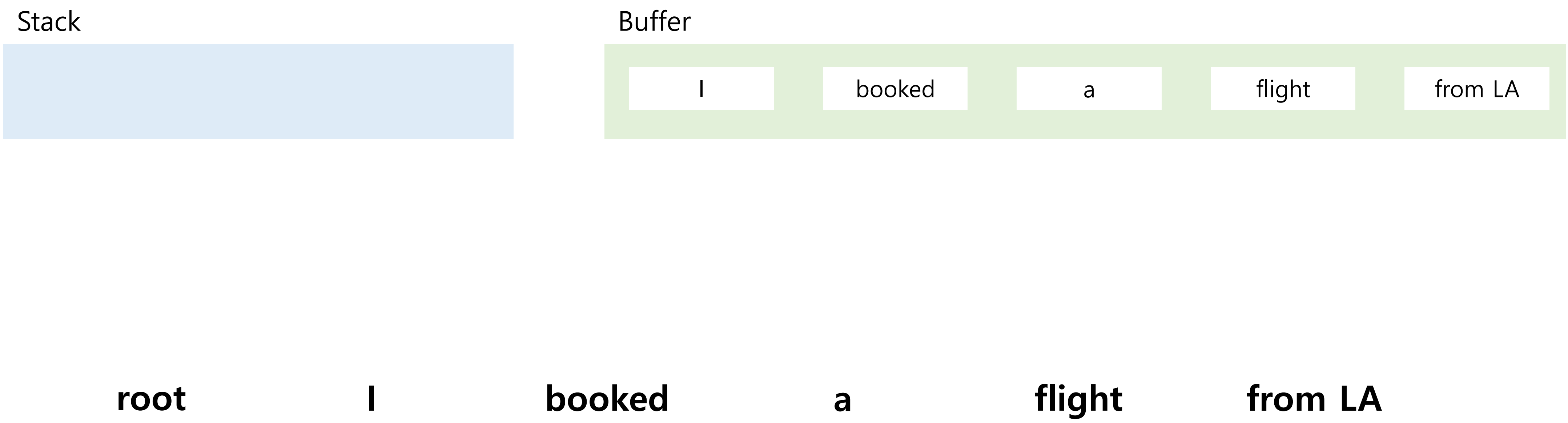
<Dependency Parse Tree>



Pre-requisites
-Dependency Parse Tree

<Transition-based Dependency Parsing>

- Buffer의 제일 앞에 있는 Word 하나를 Stack으로 옮긴 후(Shift)
- Stack에 있는 단어에 대해서 문법 규칙을 적용하고 하나의 Word만 남기는 행동(Reduce)을
- Stack에 하나의 Word만 남을 때 까지 반복하는 방식으로 Parsing
    - Buffer : 토크나이징 한 단어들 중에 Parsing을 기다리고 있는 단어들
    - Stack : 토크나이징 한 단어들 중에 파싱 작업을 진행 중인 단어들



Pre-requisites
-Dependency Parse Tree

<Transition-based Dependency Parsing>

- Buffer의 제일 앞에 있는 Word 하나를 Stack으로 옮긴 후(Shift)
- Stack에 있는 단어에 대해서 문법 규칙을 적용하고 하나의 Word만 남기는 행동(Reduce)을
- Stack에 하나의 Word만 남을 때 까지 반복하는 방식으로 Parsing
  - Buffer : 토크나이징 한 단어들 중에 Parsing을 기다리고 있는 단어들
  - Stack : 토크나이징 한 단어들 중에 파싱 작업을 진행 중인 단어들

Stack

I

Buffer

booked a flight from LA

Shift

root I booked a flight from LA

https://songys.github.io/2019LangCon/data/Parser.pdf

<Transition-based Dependency Parsing>

- Buffer의 제일 앞에 있는 Word 하나를 Stack으로 옮긴 후(Shift)
- Stack에 있는 단어에 대해서 문법 규칙을 적용하고 하나의 Word만 남기는 행동(Reduce)을
- Stack에 하나의 Word만 남을 때 까지 반복하는 방식으로 Parsing
  - Buffer : 토크나이징 한 단어들 중에 Parsing을 기다리고 있는 단어들
  - Stack : 토크나이징 한 단어들 중에 파싱 작업을 진행 중인 단어들

Stack		Buffer		
I	booked	a	flight	from LA

Shift

root **I** **booked** **a** **flight** **from LA**

https://songys.github.io/2019LangCon/data/Parser.pdf

Pre-requisites

-Dependency Parse Tree

<Transition-based Dependency Parsing>

- Buffer의 제일 앞에 있는 Word 하나를 Stack으로 옮긴 후(Shift)
- Stack에 있는 단어에 대해서 문법 규칙을 적용하고 하나의 Word만 남기는 행동(Reduce)을
- Stack에 하나의 Word만 남을 때 까지 반복하는 방식으로 Parsing
    - Buffer : 토크나이징 한 단어들 중에 Parsing을 기다리고 있는 단어들
    - Stack : 토크나이징 한 단어들 중에 파싱 작업을 진행 중인 단어들

Stack

booked

Buffer

a flight from LA

Left-Reduce

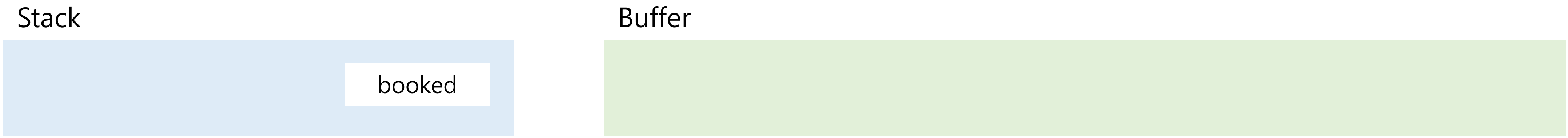
Subj

root I booked a flight from LA

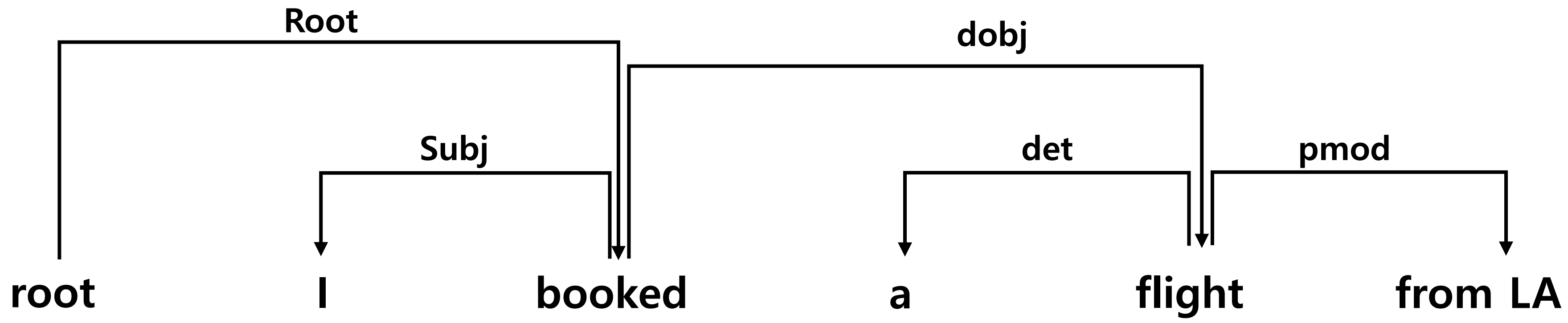
https://songys.github.io/2019LangCon/data/Parser.pdf

<Transition-based Dependency Parsing>

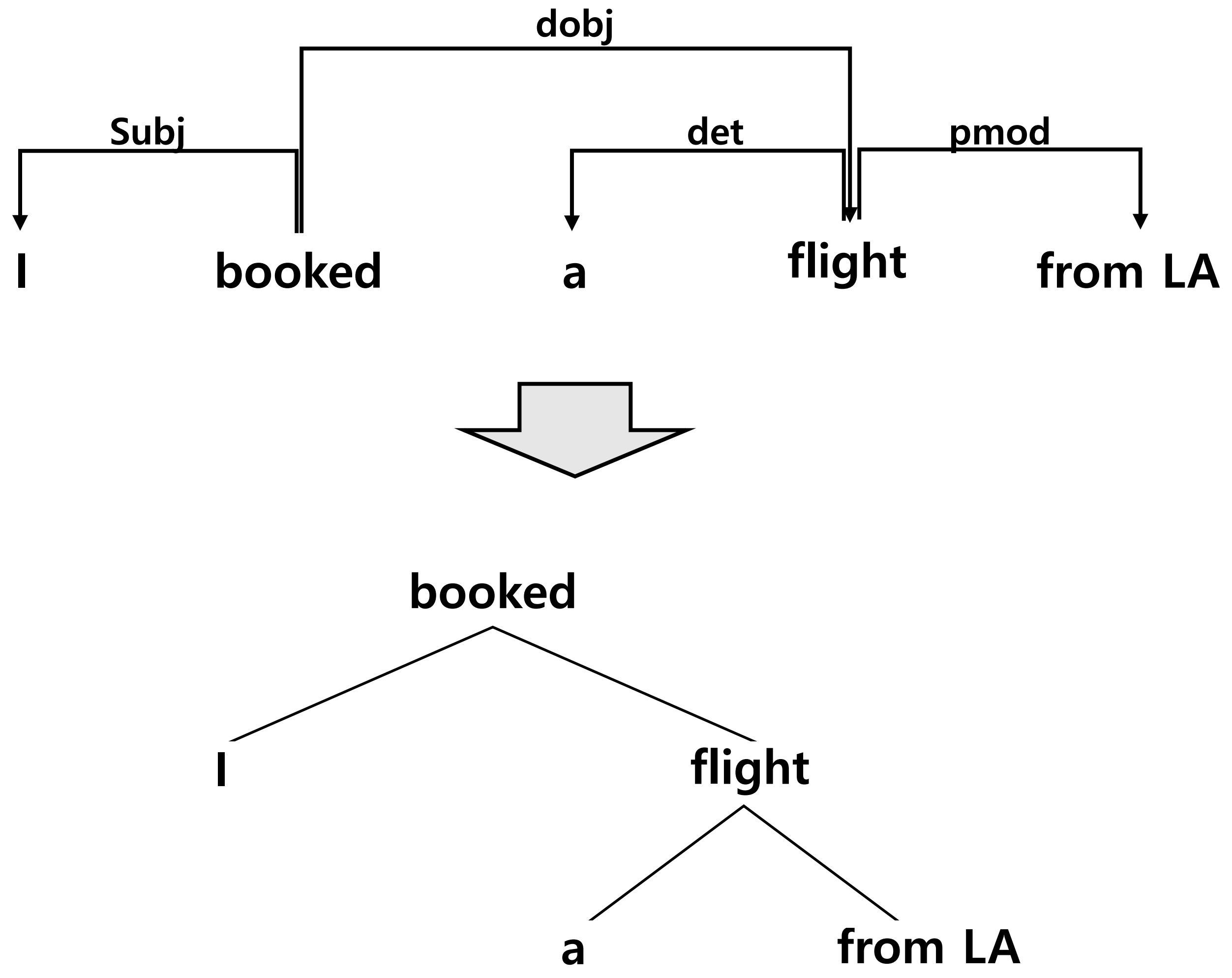
- Buffer의 제일 앞에 있는 Word 하나를 Stack으로 옮긴 후(Shift)
- Stack에 있는 단어에 대해서 문법 규칙을 적용하고 하나의 Word만 남기는 행동(Reduce)을
- Stack에 하나의 Word만 남을 때 까지 반복하는 방식으로 Parsing
  - Buffer : 토크나이징 한 단어들 중에 Parsing을 기다리고 있는 단어들
  - Stack : 토크나이징 한 단어들 중에 파싱 작업을 진행 중인 단어들



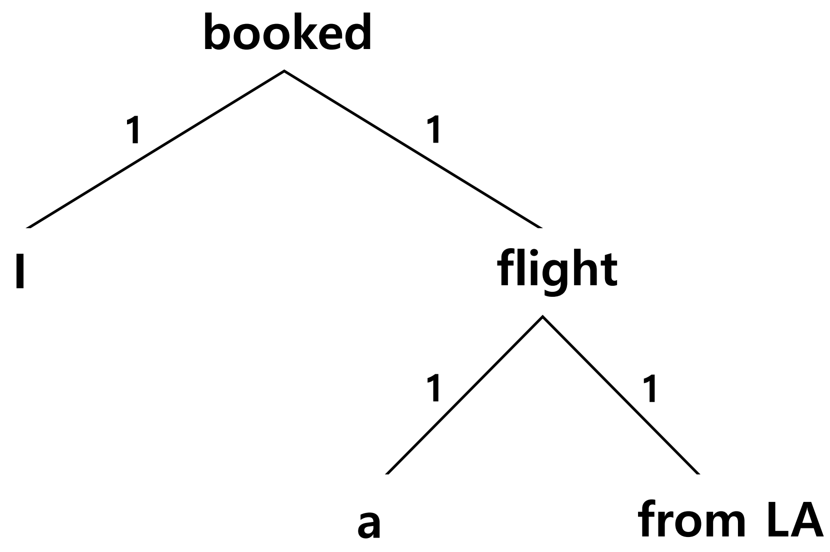
Final State



<Dependency Parse Tree>



<Dependency Parse Tree>



$$d_T(booked, I) = \ 1$$

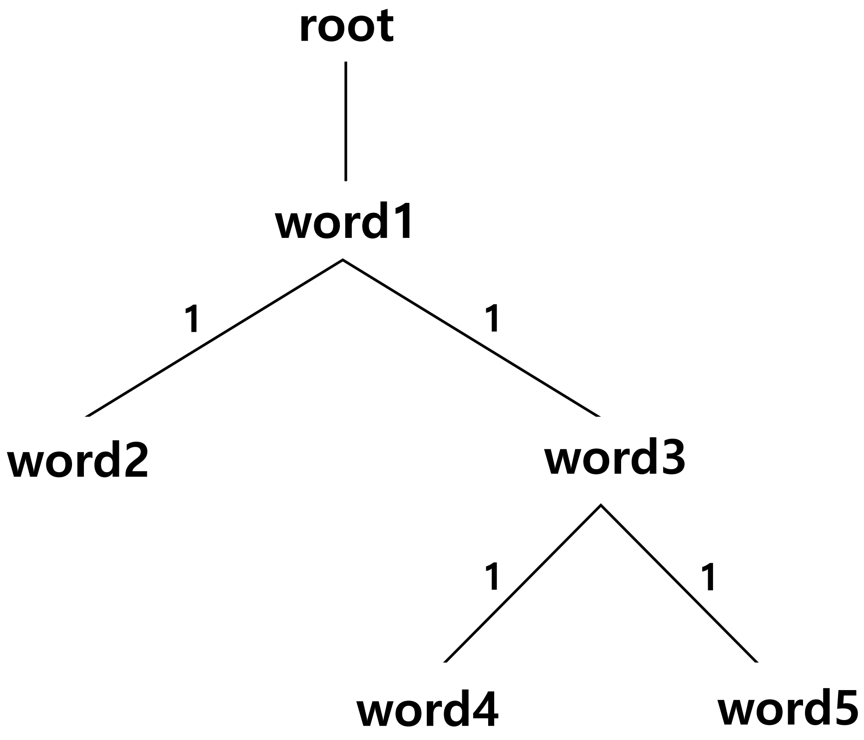
$$d_T(I, a) = 3$$

$$d_T(booked, flight) = 1$$

<Dependency Parse Tree>

$$for\ i,j:\ d_T(word_i, word_j)$$

$$for\ all\ i:\ d_T(root, word_i)$$



<Unique Tree>

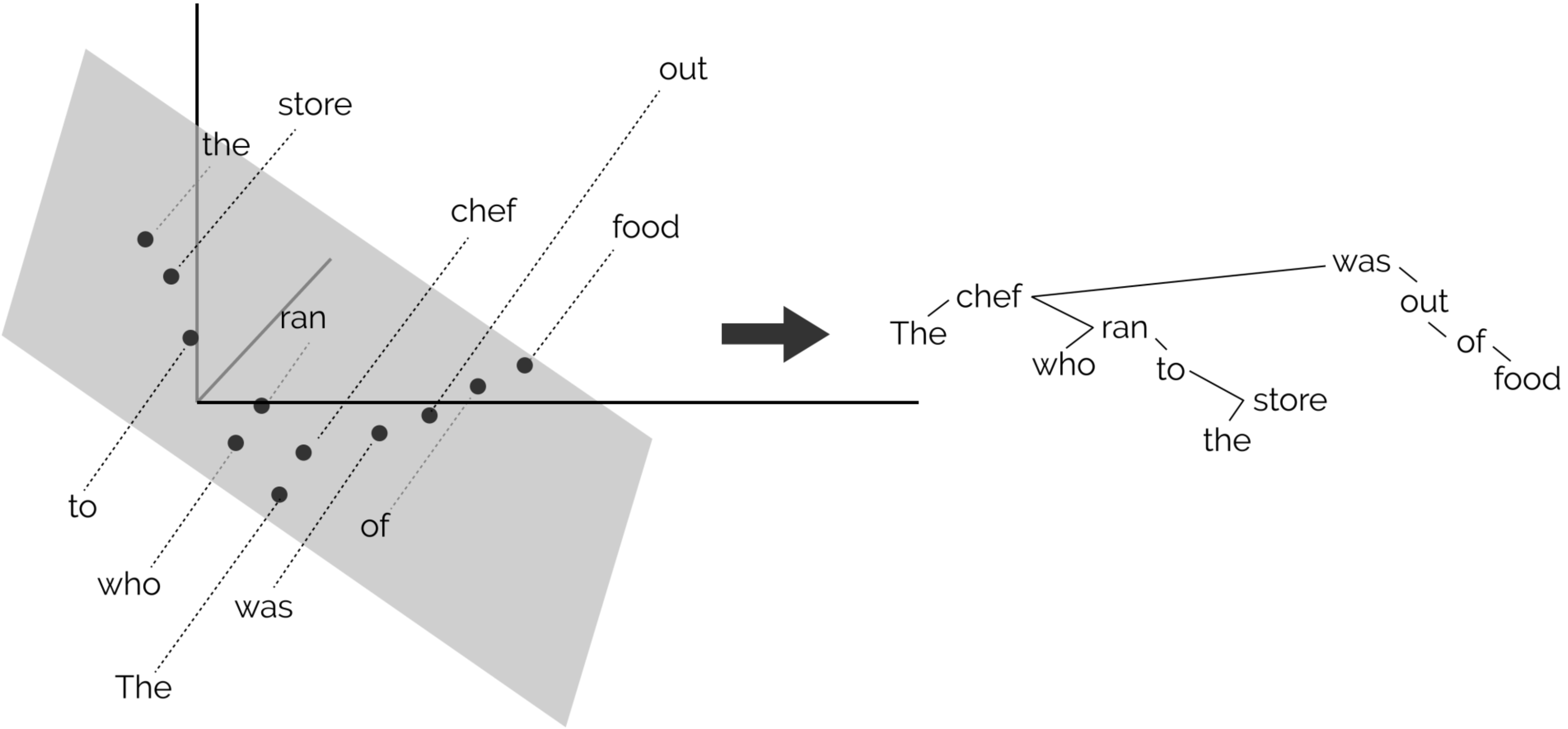
Method

- Structural Probe

<Structural Probe Overview>

There is a Linear Transformation that Transforms the Embeddings of Language Model into Dependency Parse Tree

So, Language Model Embeds Dependency Relation (Syntactic Information) of Text



Method

-Structural Probe

<Structural Probe Overview>

Linear Transformation

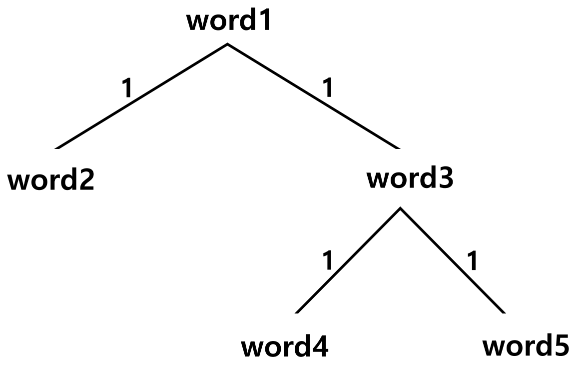
Context Embedding

Language Model

$$for\ i, j\ (i \neq j):\ d_T(word_i, word_j)$$
$$for\ all\ i:\ d_T(root, word_i)$$

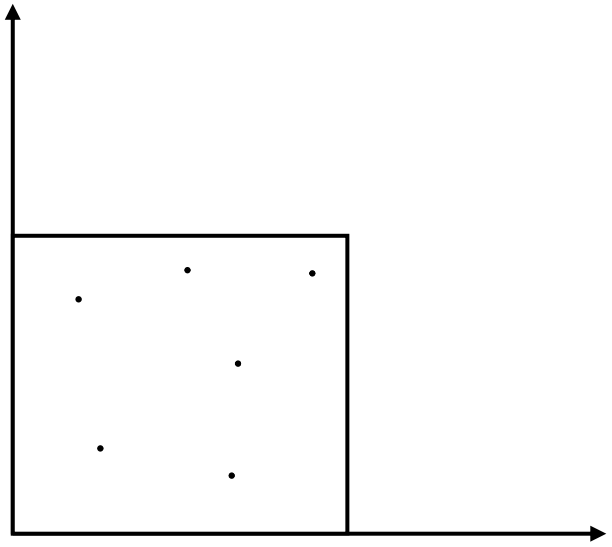
Squared Distance of Nodes in Transformed Space

$\approx$

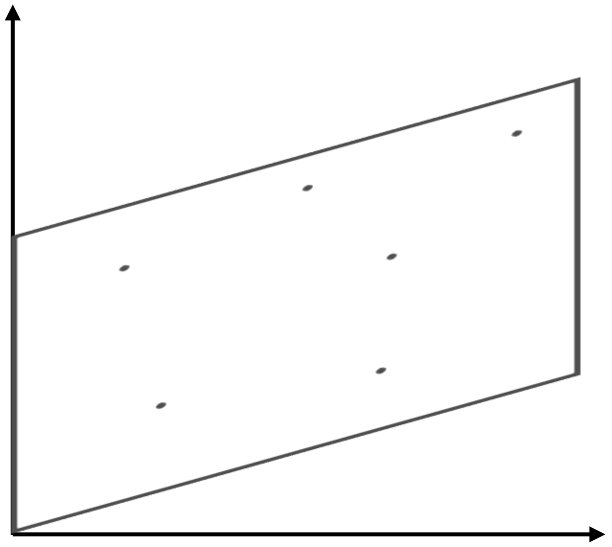


Distance of Nodes in Dependency Parse Tree

<Geometric Meaning of Linear Transformation>



$$v = \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{bmatrix}$$



$$Av = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{bmatrix}$$

<Structural Probe>

<Notation>

$w_{1:n}^{l}$: $words\ in\ sequnce_{l}$

$h_{1:n}^{l}$: $sequence\ of\ verctor\ representation$

A : $positive\ semi\ definite, symmetric\ matrix, A \in \mathbb{S}_{+}^{m\times m}$

$h^{T}Ah$: $family\ of\ inner\ product$

B : $linear\ transformation, B \in \mathbb{R}^{k\times m}, such\ that\ A = B^{T}B$

d_m:embedding dimension

$$h = \begin{bmatrix} d_1 \\ d_2 \\ \vdots \\ d_m \end{bmatrix} (m \times 1) \quad A = \begin{bmatrix} a_{11} & a_{21} & \cdots & a_{1k} & \cdots & a_{1m} \\ a_{21} & a_{22} & \cdots & a_{2k} & \cdots & a_{2m} \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ a_{k1} & a_{k2} & \cdots & a_{kk} & \cdots & a_{km} \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mk} & \cdots & a_{mm} \end{bmatrix} (m \times m) \quad B = \begin{bmatrix} b_{11} & b_{12} & \cdots & b_{1m} \\ b_{21} & b_{22} & \cdots & b_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ b_{k1} & b_{k2} & \cdots & b_{km} \end{bmatrix} (k \times m)$$

$h^{T}Ah$: $scalar\ (1 \times 1)$

<Linear Transformation>

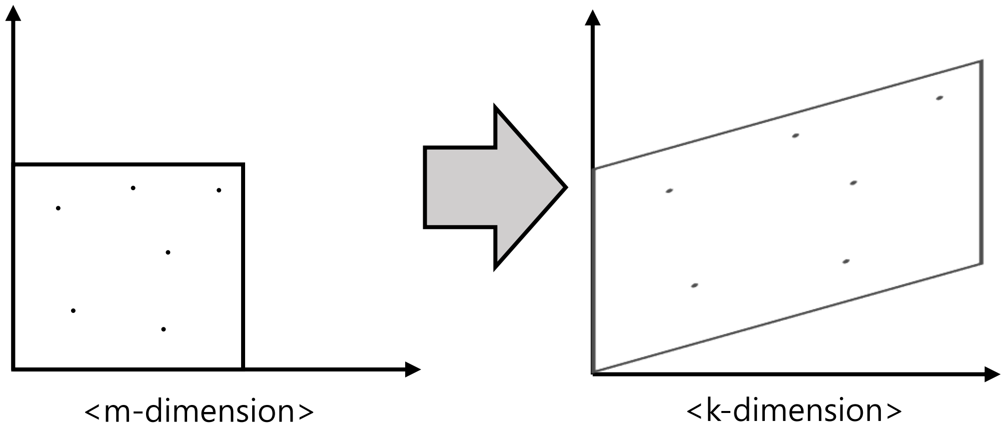
$$h^T A h : family\ of\ inner\ product$$

$$B : linear\ transformation, B \in \mathbb{R}^{k \times m}, such\ that\ A = B^T B$$

$$\therefore h^T A h = (Bh)^T (Bh) : inner\ product\ of\ transformed\ vector$$

nd_k: new dimension

$$B = \begin{bmatrix} b_{11} & b_{12} & \cdots & b_{1m} \\ b_{21} & b_{22} & \cdots & b_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ b_{k1} & b_{k2} & \cdots & b_{km} \end{bmatrix} (k \times m) \qquad h = \begin{bmatrix} d_1 \\ d_2 \\ \vdots \\ d_m \end{bmatrix} (m \times 1) \qquad Bh = \begin{bmatrix} nd_1 \\ nd_2 \\ \vdots \\ nd_k \end{bmatrix} (k \times 1), k \leq m$$



Method

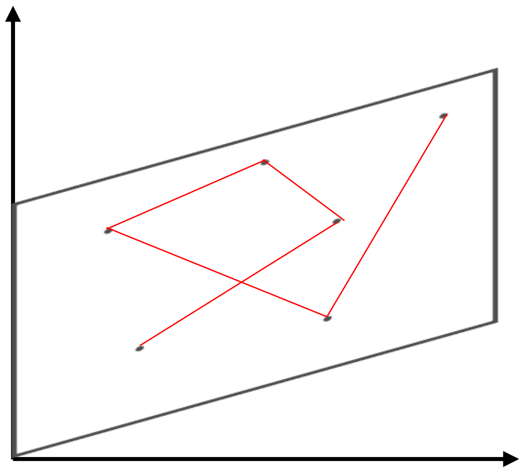
-Structural Probe

<Calculate Distance of Nodes>

$$d_B(h_i^l, h_j^l)^2 = \left(B(h_i^l - h_j^l)\right)^T (B(h_i^l - h_j^l))$$

$$d_B(h_i^l, h_j^l)^2 = (nd_{i1} - nd_{j1})^2 + (nd_{i2} - nd_{j2})^2 + \cdots + (nd_{ik} - nd_{jk})^2$$

$$\therefore d_B\,(h_i^l, h_j^l)^2 : squared\ euclidean\ distance\ of\ nodes$$



<k-dimension>

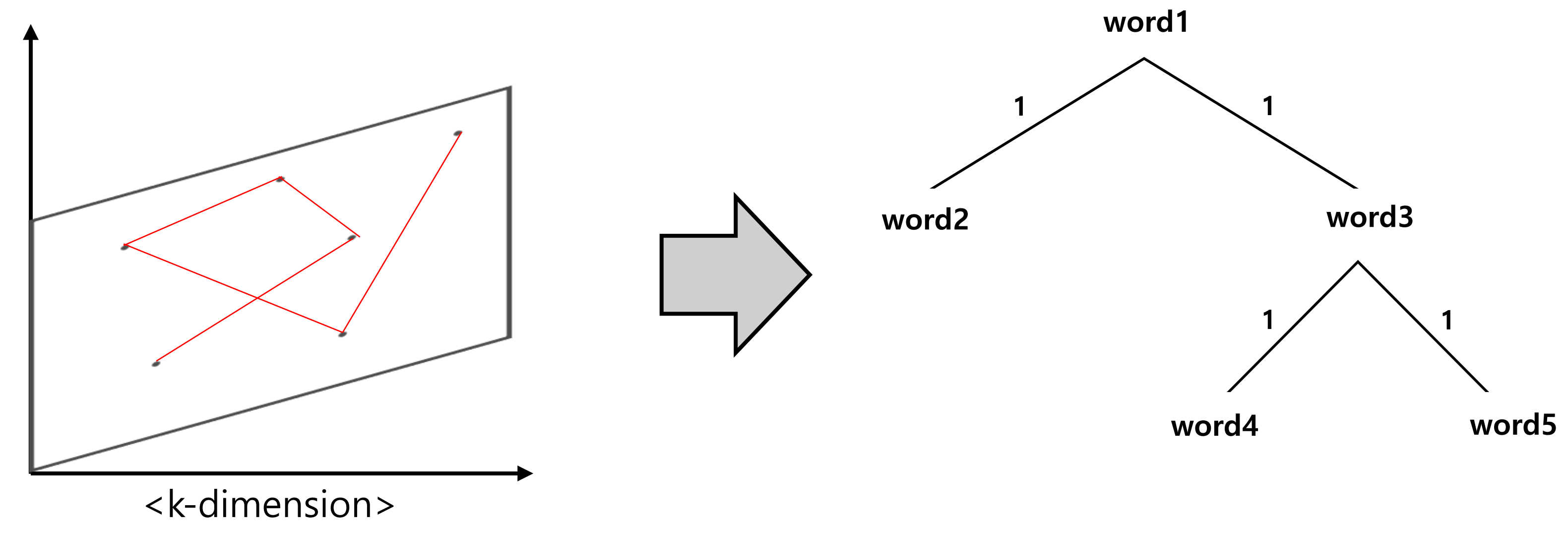
Method

-Structural Probe

<Training>

$$\min_{B} \sum_{l} \frac{1}{|s^{l}|^{2}} \sum_{i,j} |d_{T^{l}}(w_{i}^{l}, w_{j}^{l}) - d_{B}(h_{i}^{l}, d_{j}^{l})^{2}|$$

$$\left|s^{l}\right|: length\ of\ the\ sentence, sentence\ has\ \left|s^{l}\right|^{2} word\ pairs$$



Method

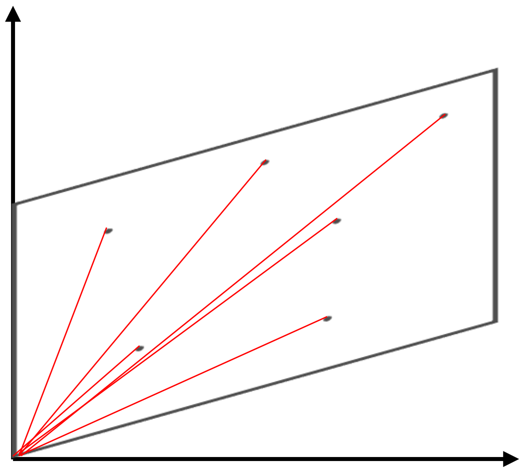
-Structural Probe

<Calculate Distance from Root Node>

$$\left|\left|h_i\right|\right|_B{}^2 = (Bh_i)^T(Bh_i)$$

$$\left|\left|h_i\right|\right|_B{}^2 = (nd_{i1})^2 + (nd_{i2})^2 + \cdots + (nd_{ik})^2$$

$$\therefore \left|\left|h_i\right|\right|_B{}^2 : norm\ of\ node\ i\ in\ transformed\ space$$

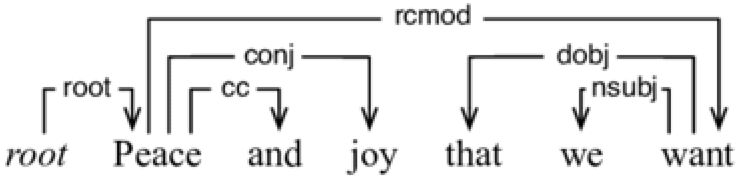


<k-dimension>

Experiments

<Penn Treebank Dataset>

"Peace and joy that we want"



<Sentence>

<Tree Structure of Given Sentence>
<Dependency Relations of Tokens in Given Sentence>

<Representation Models>

<ELMo>

5.5B-word
Pre-trained ELMo
1024-dim Embedding

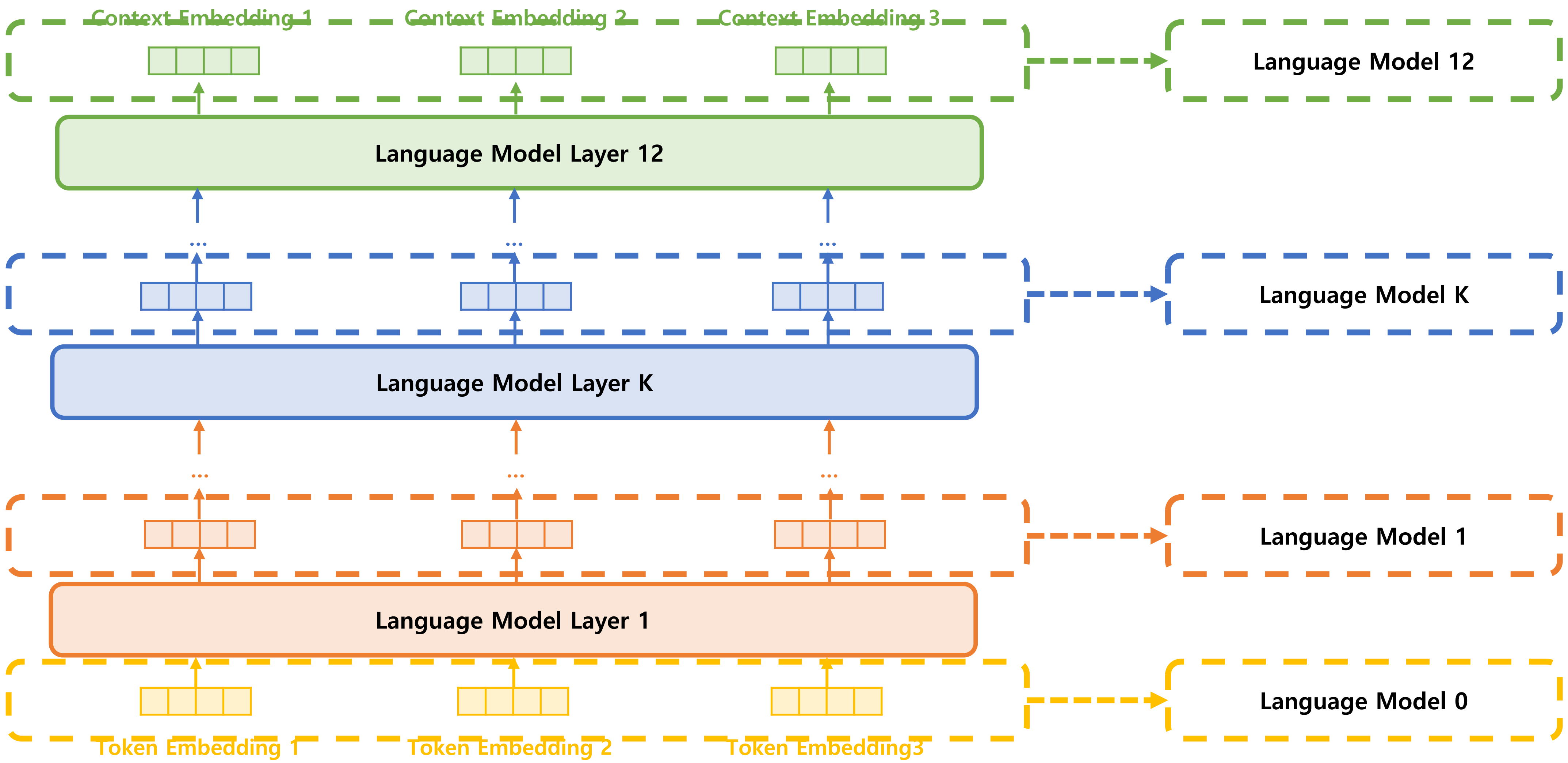
<BERTBase>

Pre-trained BERTBase
(cased)
768-dim Embedding

<BERTLarge>

Pre-trained BERTLarge
(cased)
1024-dim Embedding

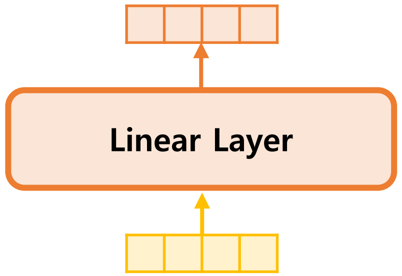
<Representation Models>



Experiments

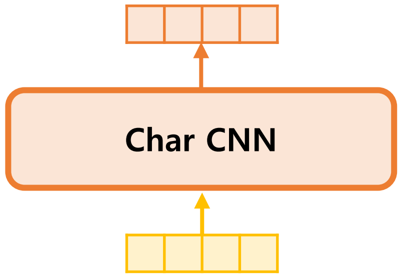
-Baseline

<Baseline>



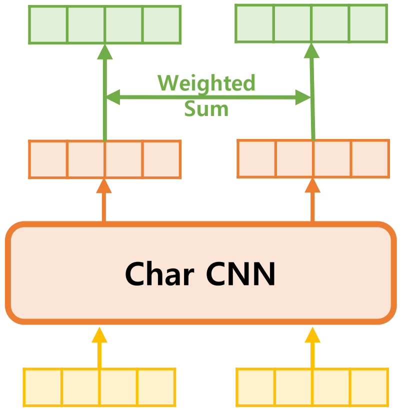
<Linear>

Token Embedding
No Context Information
No Sequential Information



<ELMo0>

Character Embedding
No Context Information
No Sequential Information

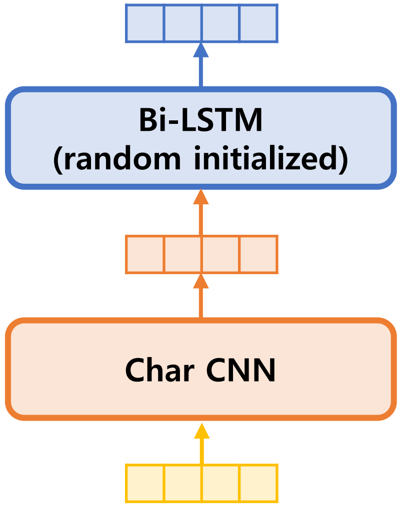


<Decay0>

ELMo0 with Weighted Sum of
Other Tokens in Same Sentence

$$Weight = \frac{1}{2^d}, d: distance\ of\ words$$

Contain Context Information
No Sequential Information



<Proj0>

ELMo0 with Randomly Initialized
Bi-LSTM (1024-dim)
Contain Sequential and Context
Information

<Evaluation Metric>

<Distance Metric>

<UUAS>

$$\frac{\boldsymbol{\#\ of\ Correct\ Nodes}}{\boldsymbol{\#\ of\ All\ Groud\ Truth\ Nodes}}$$

<DSpr.>

$$\frac{\sum_{i=5}^{50} \frac{avg(spearman\ Correlation\ of\ Each\ (Predicted, True)\ word\ in\ lenght\ \boldsymbol{i}\ sentence)}{\#\ of\ length\ i\ sentence}}{\sum_{i=5}^{50} \boldsymbol{i}}$$

$Ordered\ by\ Distance$

<Depth Metric>

<Root%>

$$\frac{\boldsymbol{\#\ of\ Correct\ Root}}{\boldsymbol{\#\ of\ Root\ Node\ of\ Ground\ Truth}}$$

$Root$: $Node\ with\ Least\ Depth$

<NSpr.>

$$\frac{\sum_{i=5}^{50} \frac{avg(spearman\ Correlation\ of\ Each\ (Predicted, True)\ word\ in\ lenght\ \boldsymbol{i}\ sentence)}{\#\ of\ length\ i\ sentence}}{\sum_{i=5}^{50} \boldsymbol{i}}$$

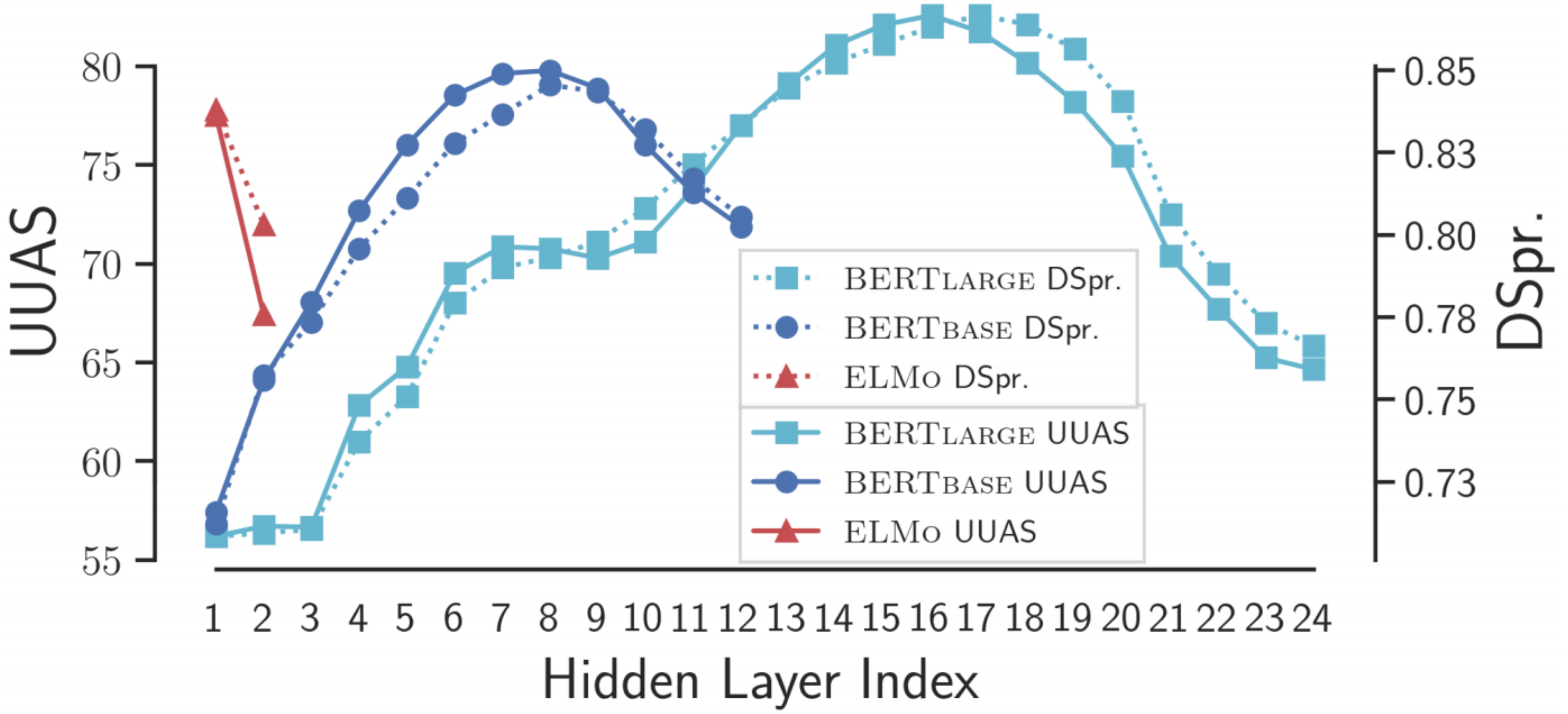
$Ordered\ by\ Norm$

<Result>

	Distance		Depth	
Method	**UUAS**	**DSpr.**	**Root%**	**NSpr.**
LINEAR	48.9	0.58	2.9	0.27
ELMO0	26.8	0.44	54.3	0.56
DECAY0	51.7	0.61	54.3	0.56
PROJ0	59.8	0.73	64.4	0.75
ELMO1	77.0	0.83	86.5	0.87
BERT$_{BASE}$7	79.8	0.85	88.0	0.87
BERT$_{LARGE}$15	**82.5**	0.86	89.4	0.87
BERT$_{LARGE}$16	81.7	**0.87**	**90.1**	**0.89**

<Result of Structural Probe on PTB>

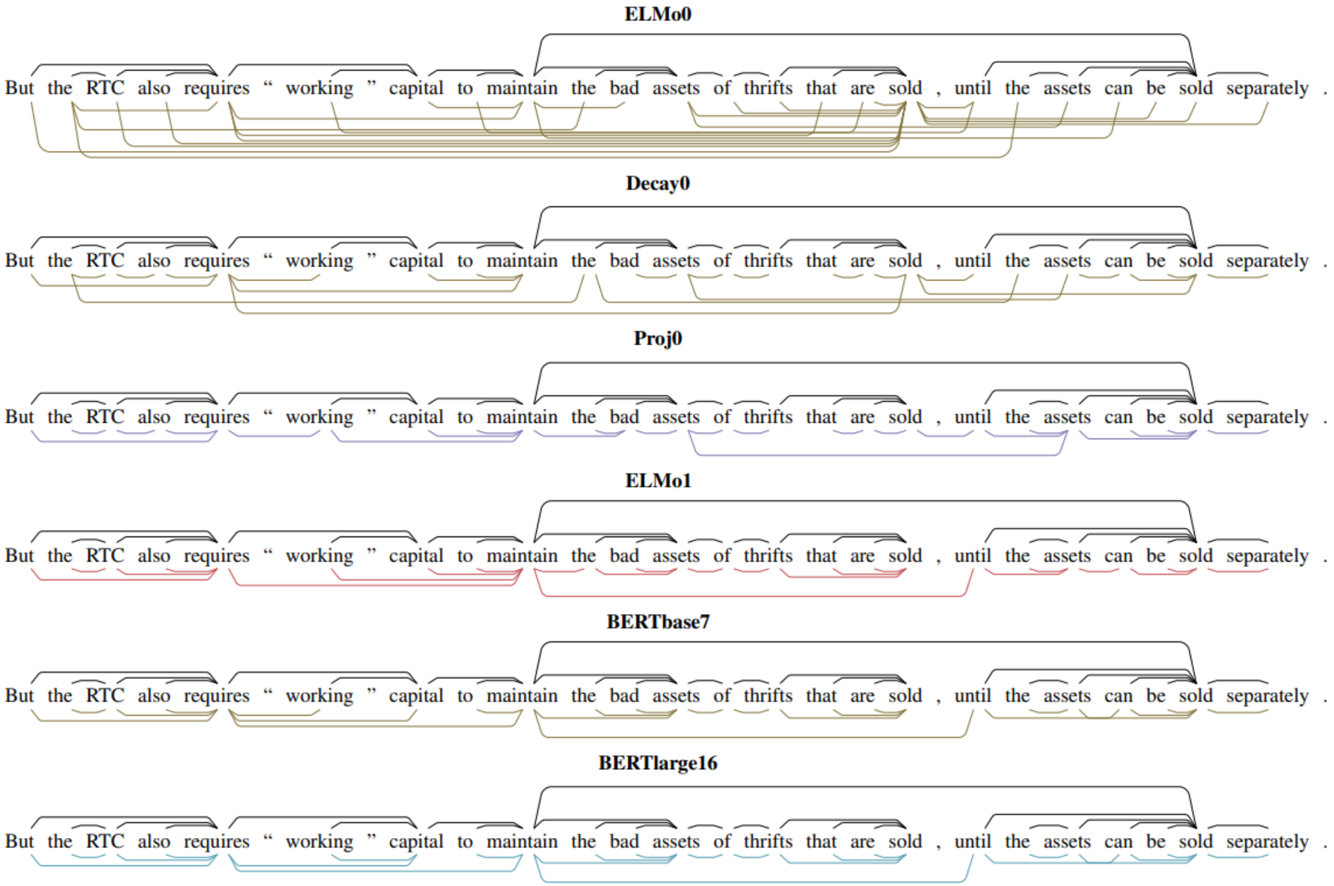
<Result>



<Parse Distance UUAS & Dspr. Across BERT and ELMo Layers>

Experiments
-Result

<Result>

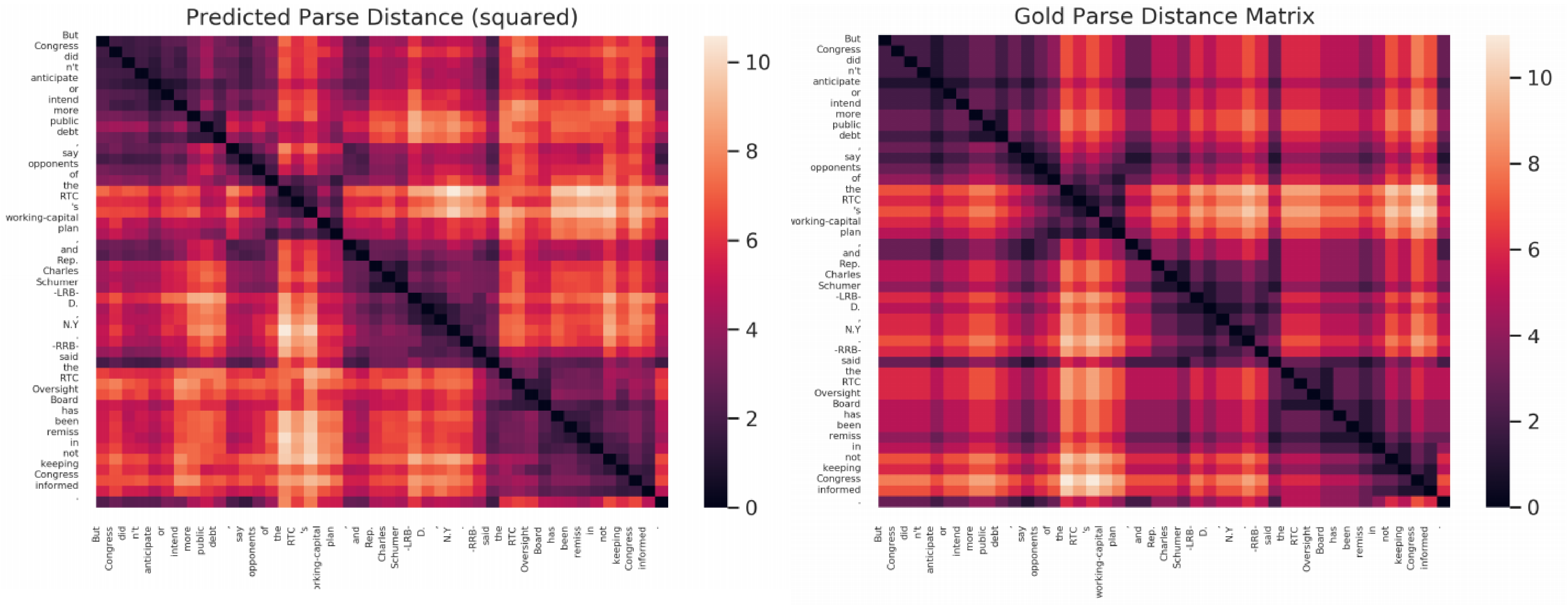


<Minimum Spanning Trees Extracted by Various Models>

Experiments

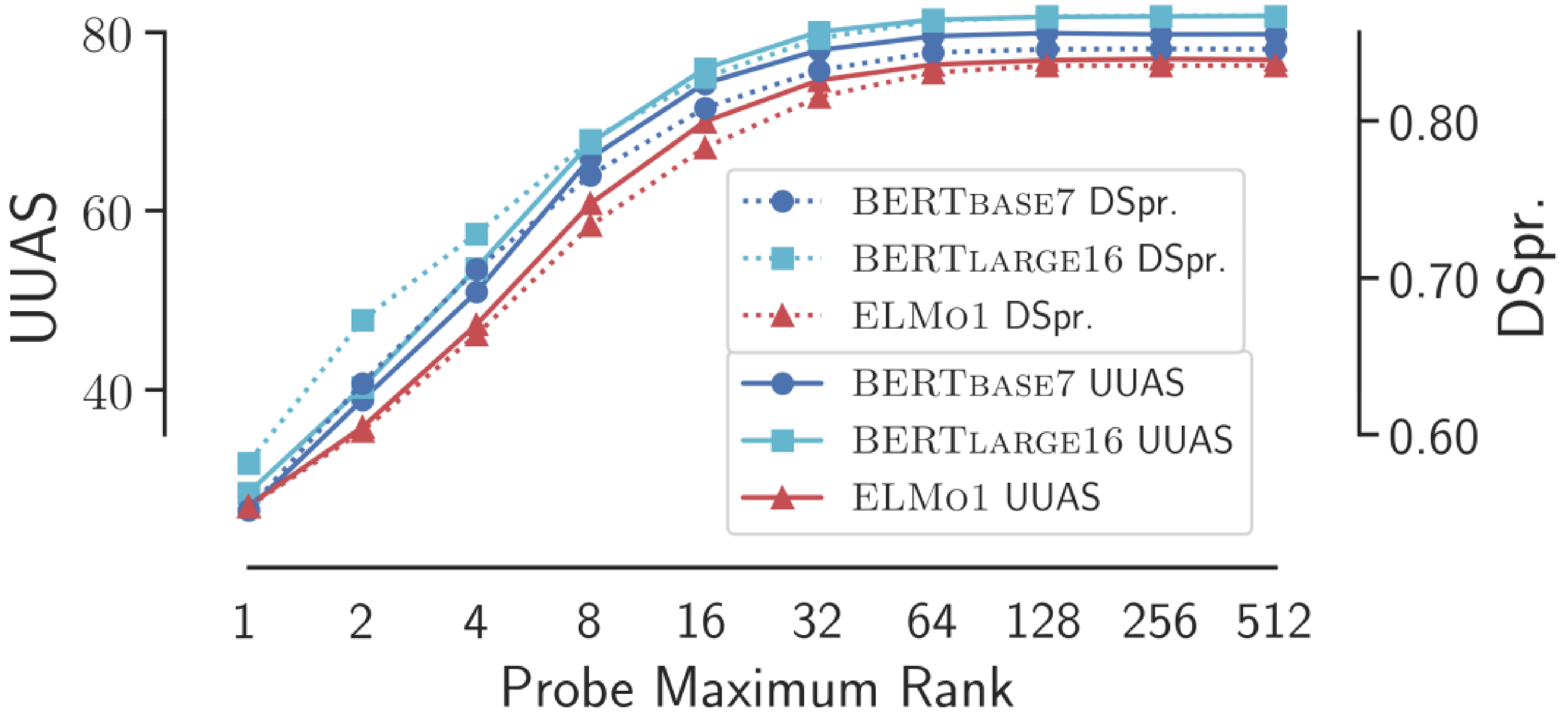
-Result

<Result>



<Parse Distance of Predicted & Gold Tree (from BERTLARGE16)>

<Result>



<Parse Distance Tree Reconstruction Accuracy on Various Probe Maximum Rank(k)>

Future Works

<More Things to do in the Future>

- **Why Does Squared L2 Distance Reconstruct Dependency Parse Tree?**
- **Representation of Language Model Does Not Use Full Dimension to Contain Dependency Relation (Syntactic Information) of Text.**
- **Then, What More Information is Contained in Language Model Embedding?**

Visualizing and Measuring the Geometry of BERT

Coenen et al., 2019, NIPS

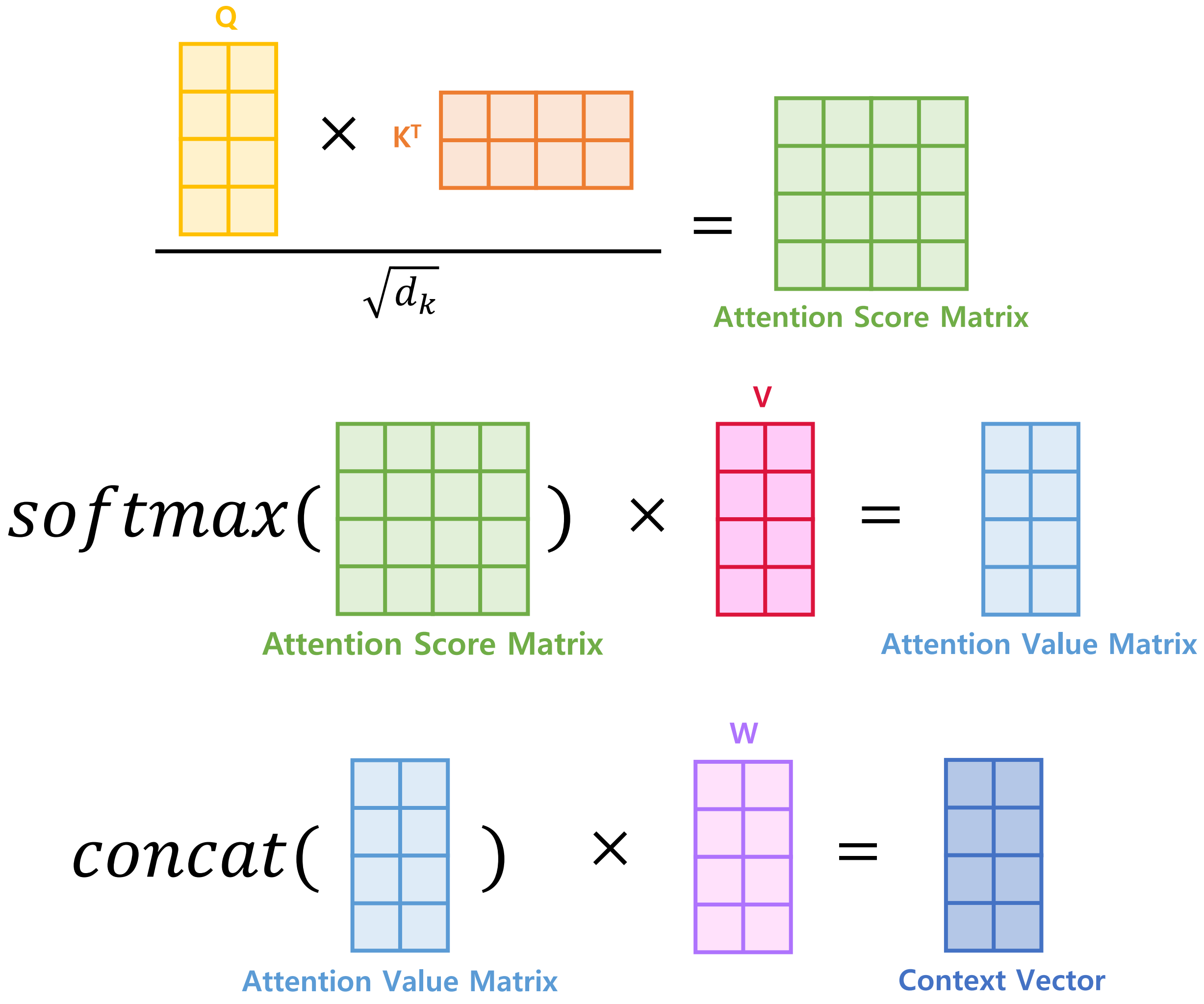
Geometry of Syntax

- Attention Probe
- Geometry of Parse Tree Embedding

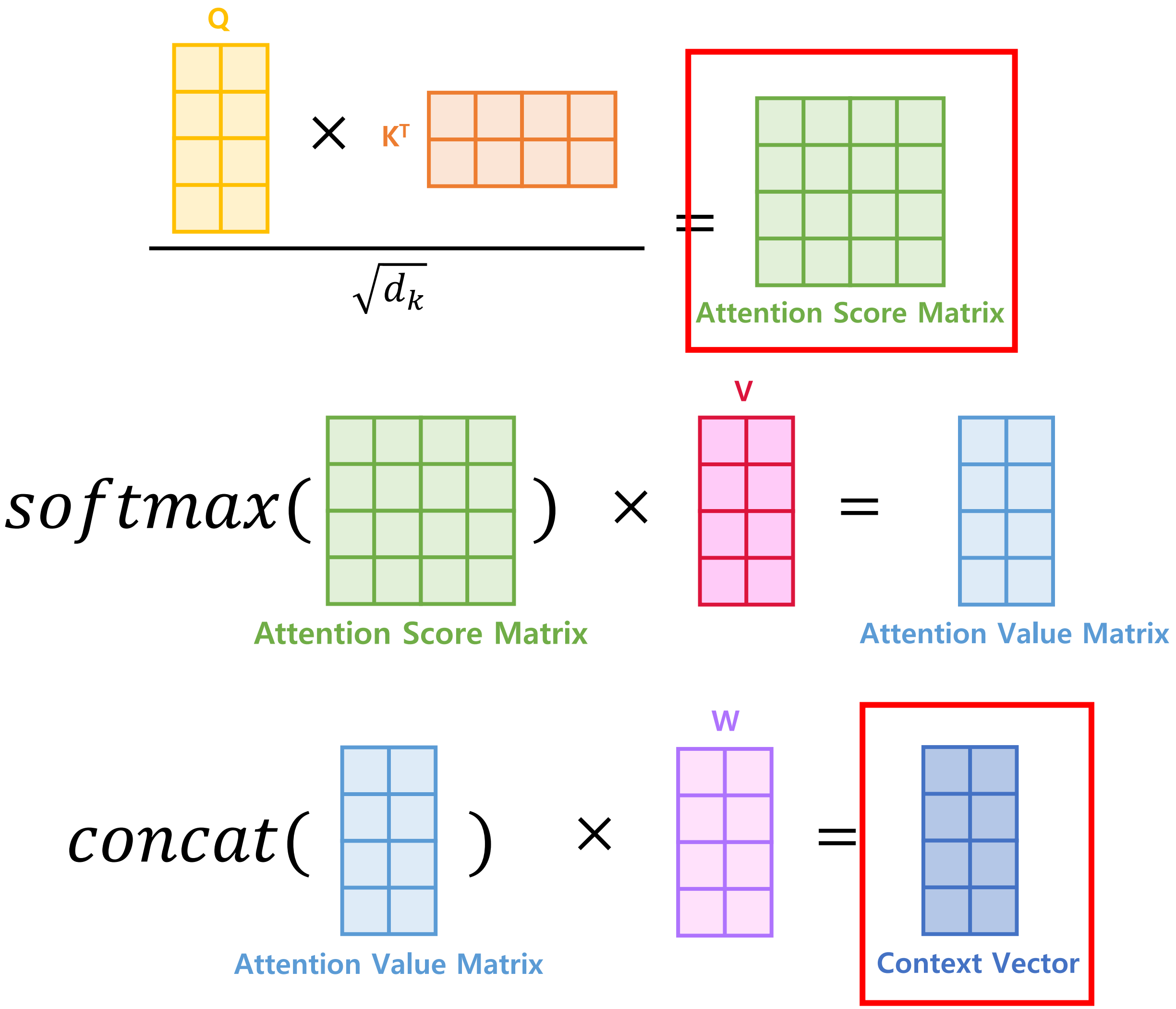
Geometry of Syntax

-Attention Probe

<Multi-head Self Attention>



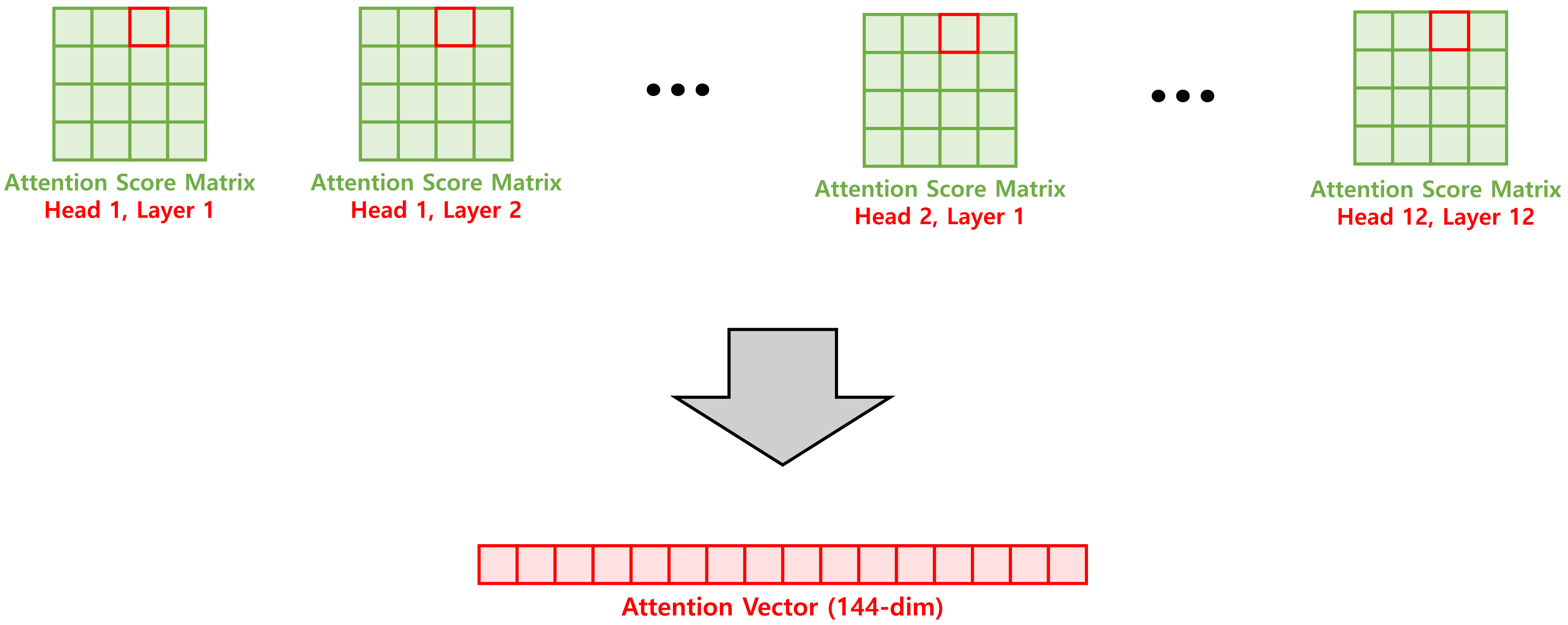
<Attention Probe>



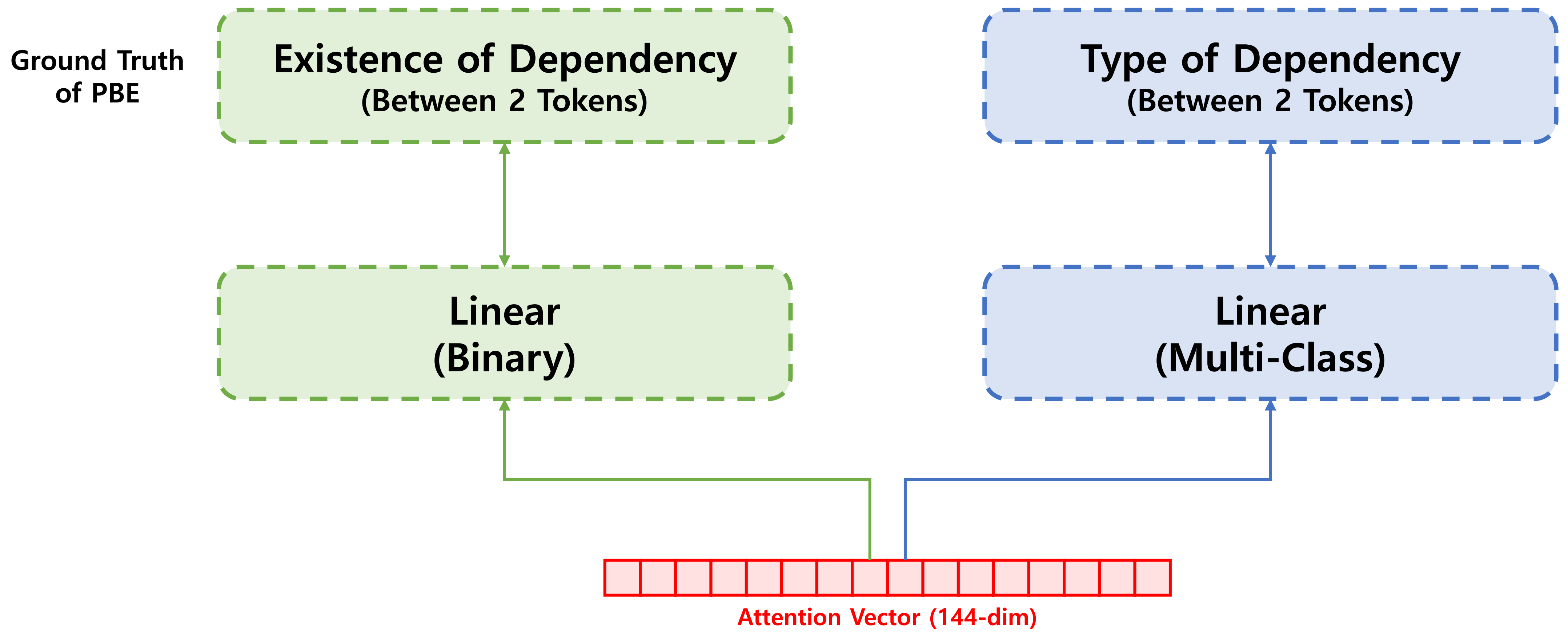
Geometry of Syntax

-Attention Probe

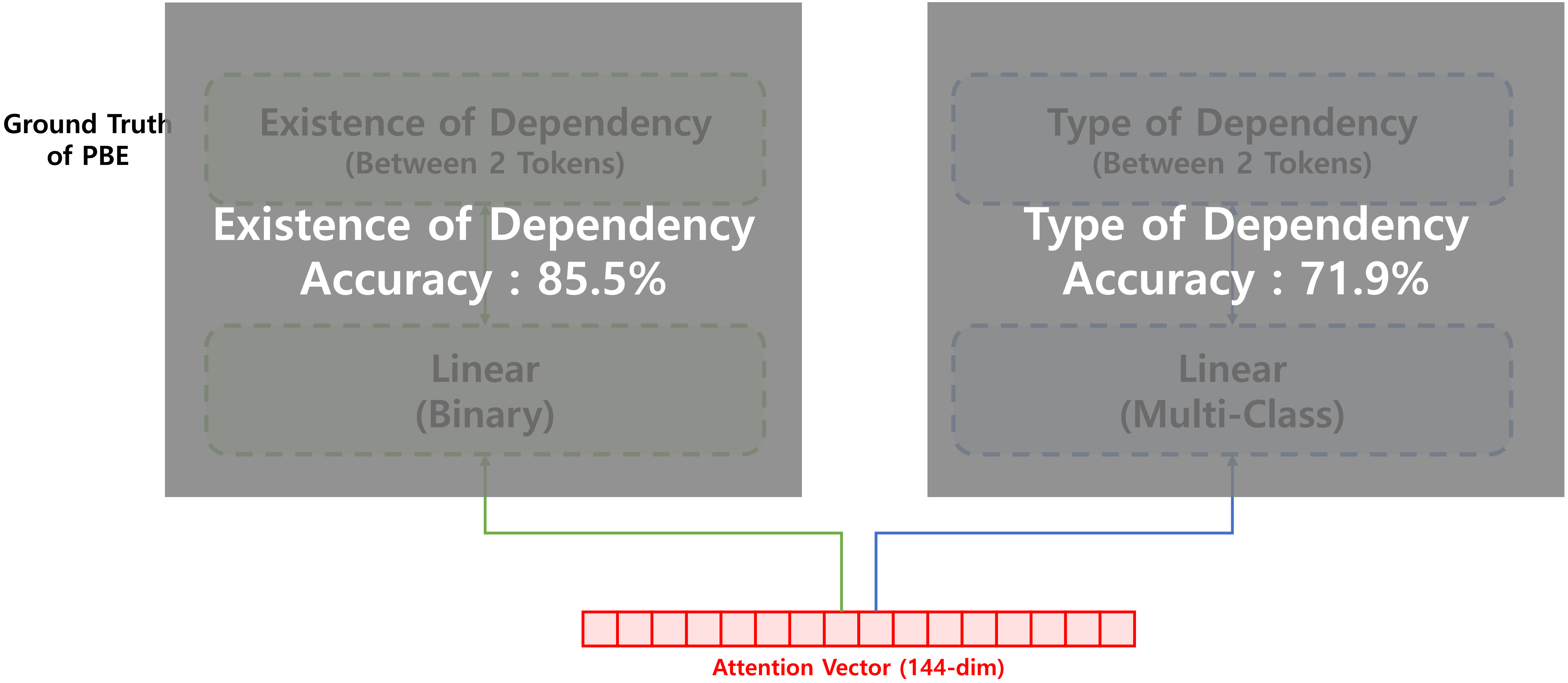
<Attention Probe>



<Attention Probe>

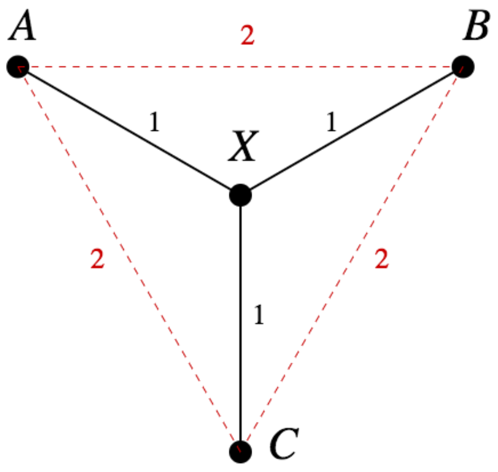


<Attention Probe>



<Pythagorean Embedding>

"One cannot generally embed a tree, with its tree metric d, isometrically into Euclidean space"



$$d(A,B) = d(A,X) + d(X,B)$$

$$\therefore\ A, X \text{ and } B \text{ is collinear}$$

$$d(A,C) = d(A,X) + d(X,C)$$

$$\therefore\ A, X \text{ and } C \text{ is collinear}$$

$$\therefore\ B = C : \text{contradiction}$$

Geometry of Syntax
-Geometry of Parse Tree Embedding

<Pythagorean Embedding>

$$\textit{Tree } T: t_0, \ldots t_{n-1}, \textit{where } t_0: \textit{root node}$$

$$\{e_1, \ldots, e_{n-1}\}: \textit{orthogonal unit basis vectors for } \mathbb{R}^{n-1}$$

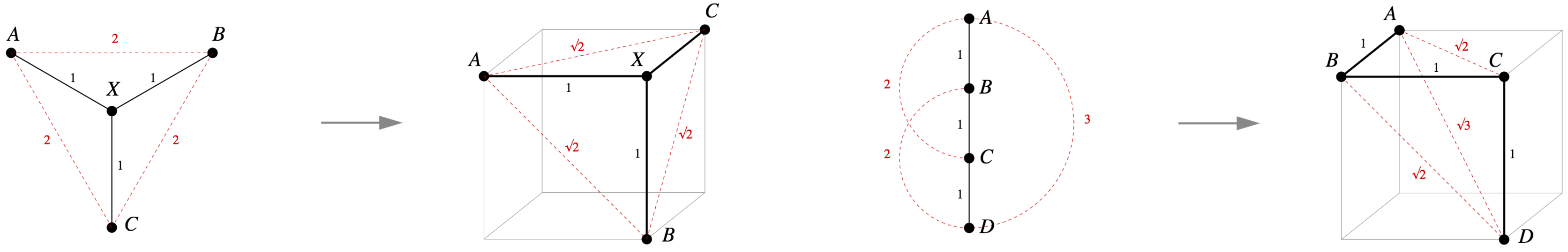
$$f: T \rightarrow \mathbb{R}^{n-1}$$

$$f(t_0) = 0$$

$$f(t_i) = e_i + f(parent(t_i))$$

$$||f(x) - f(y)||^2 = m = d(x, y)$$

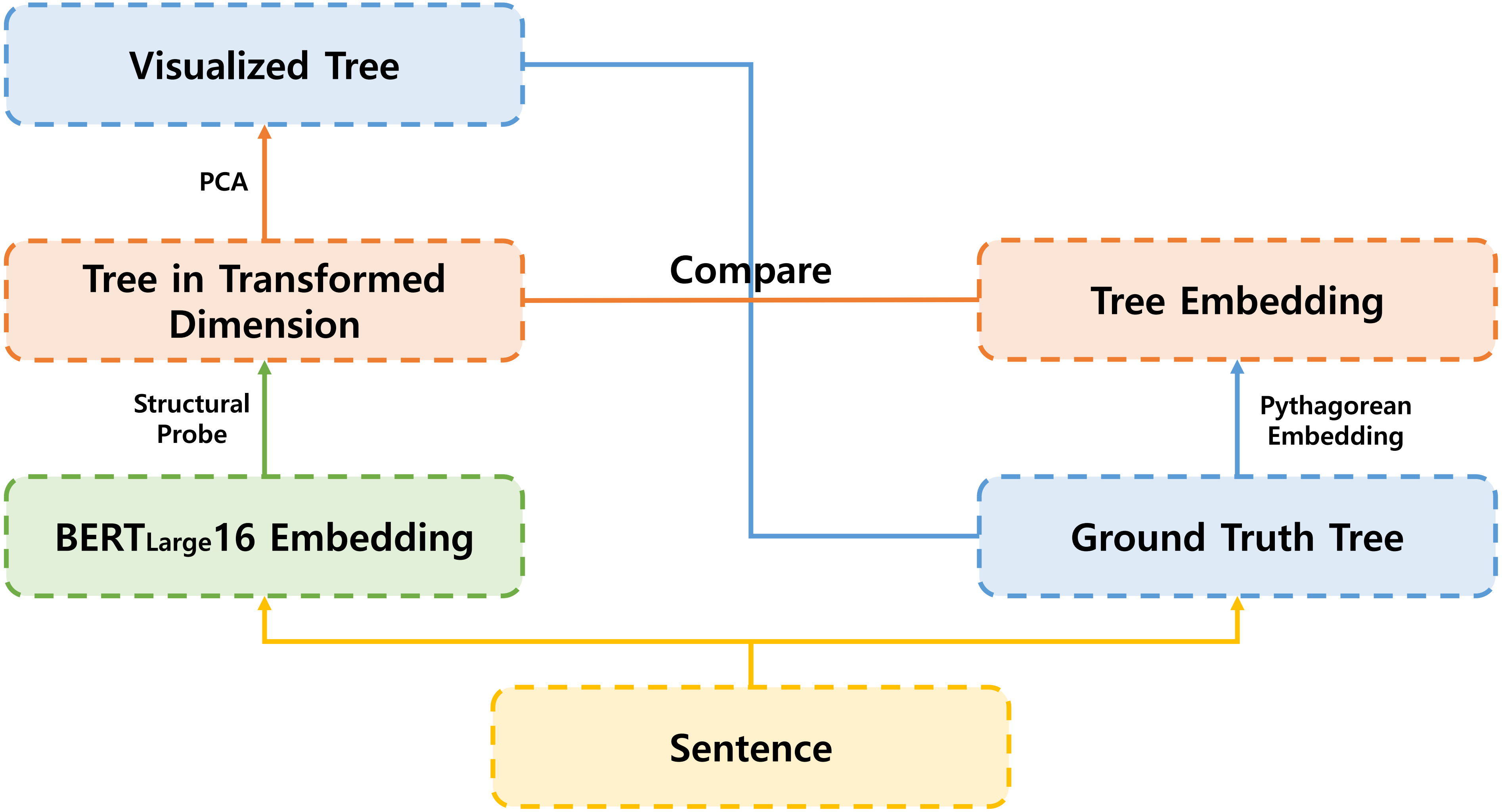
$$\textit{where}, n: \textit{number of nodes}, m = d(x, y): \textit{tree distance}$$



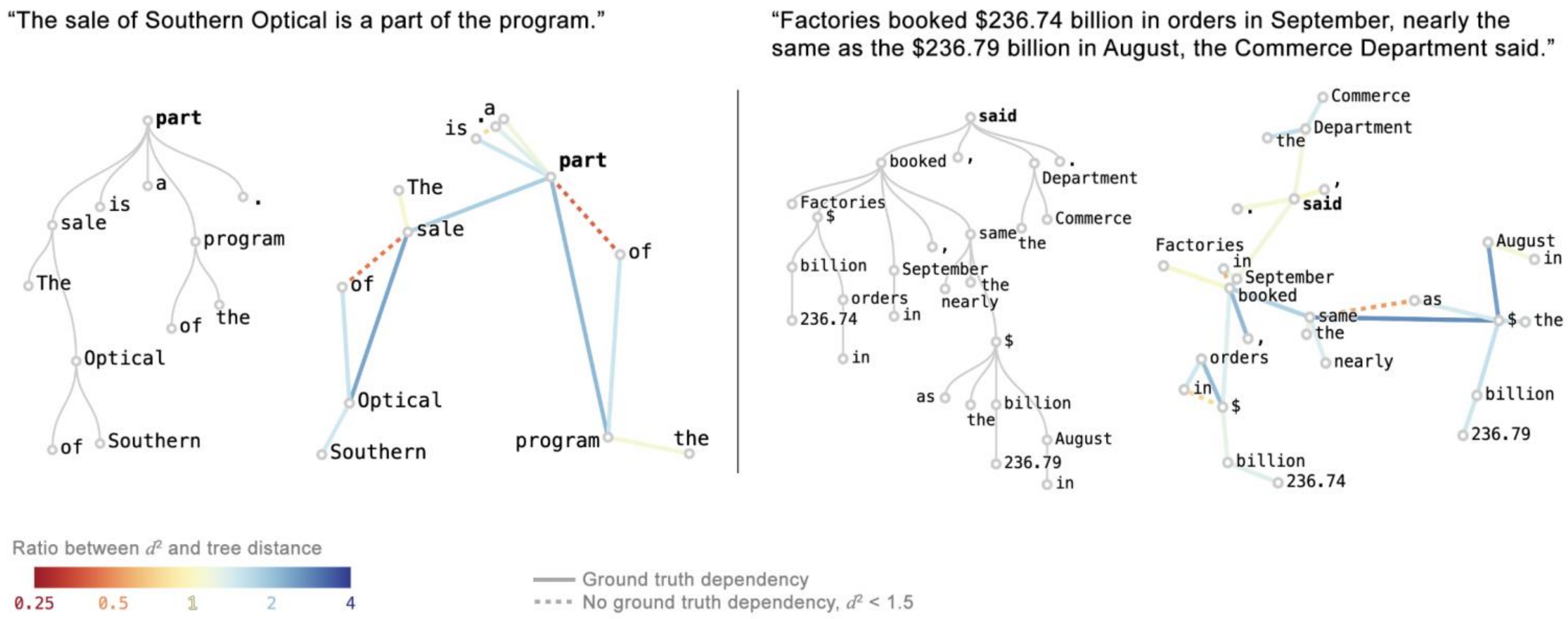
Geometry of Syntax

-Visualization of Parse Tree Embedding

<Visualization of Parse Tree Embedding>

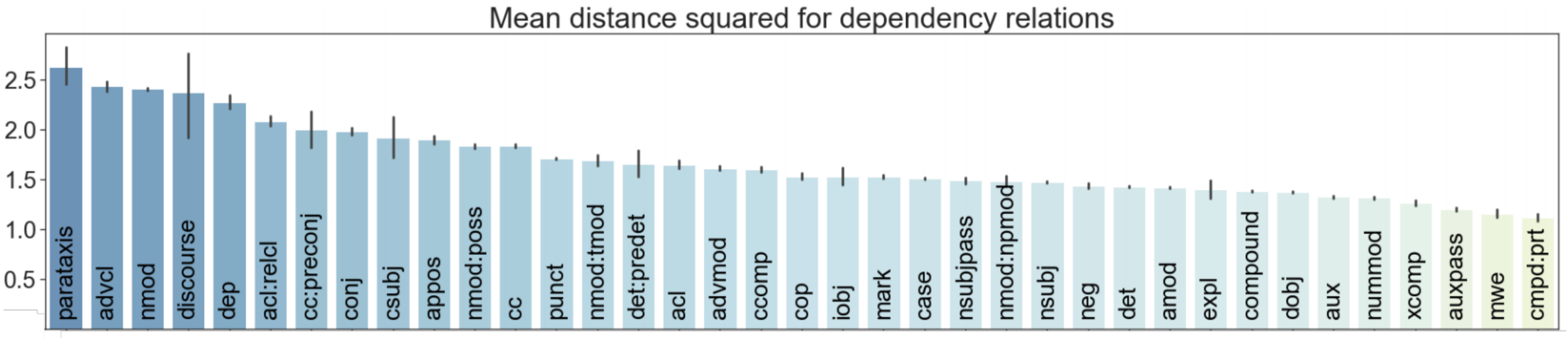


<Visualization of Parse Tree Embedding>



<Visualizations of Tree Embedding>
(Left – Parse Tree, Right – PCA Projection of Context Embedding)

<Visualization of Parse Tree Embedding>



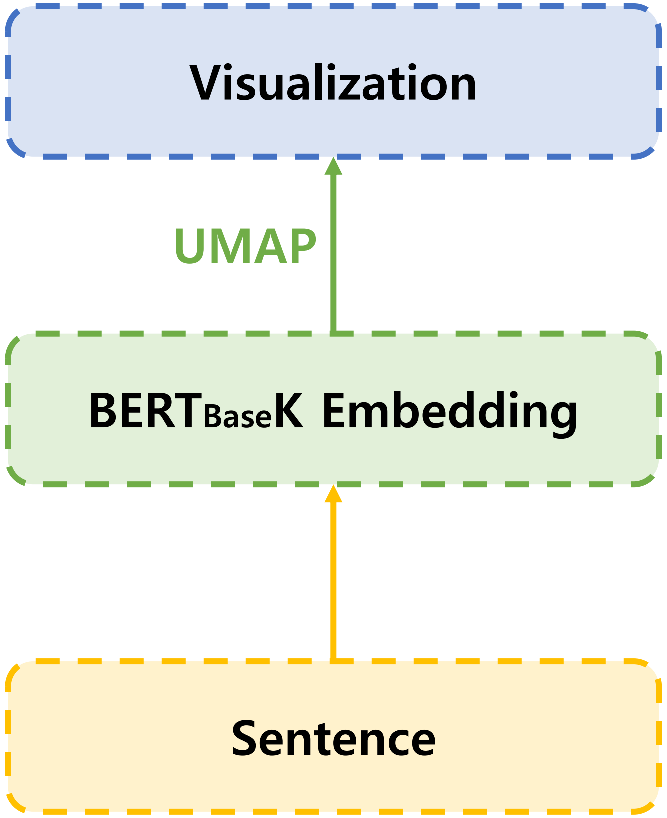
<The Average Squared Edge Length Between Two Words with a Given Dependency>

"BERT`s syntactic representation has an additional quantitative aspect beyond traditional dependency grammar"

Geometry of Word Senses

- Visualization of Word Senses
- Measurement of Word Sens Disambiguation
- Embedding Distance and Context

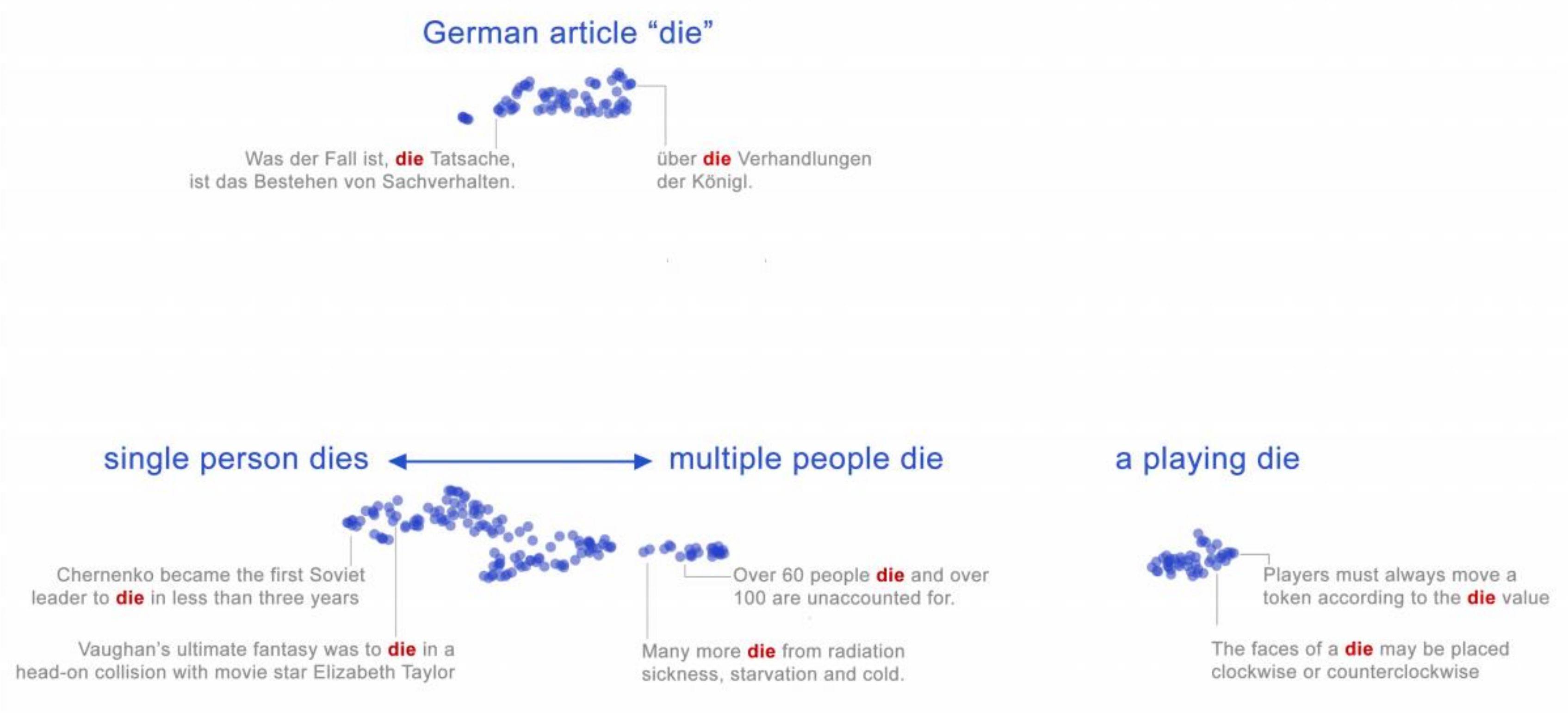
<Visualization of Word Senses>



Geometry of Word Senses

-Visualization of Word Senses

<Visualization of Word Senses>



<Embeddings for the Word "die" in Different Contexts>

<Word Sense Disambiguation>

"Give me an account of what you saw."

"The problem is important on this account."

"You must give in my account once a month."

"In his account it was very excellent."

<Sentence>

Sense 1: Explanation

Sense 2: Reason

Sense 3: Bank account

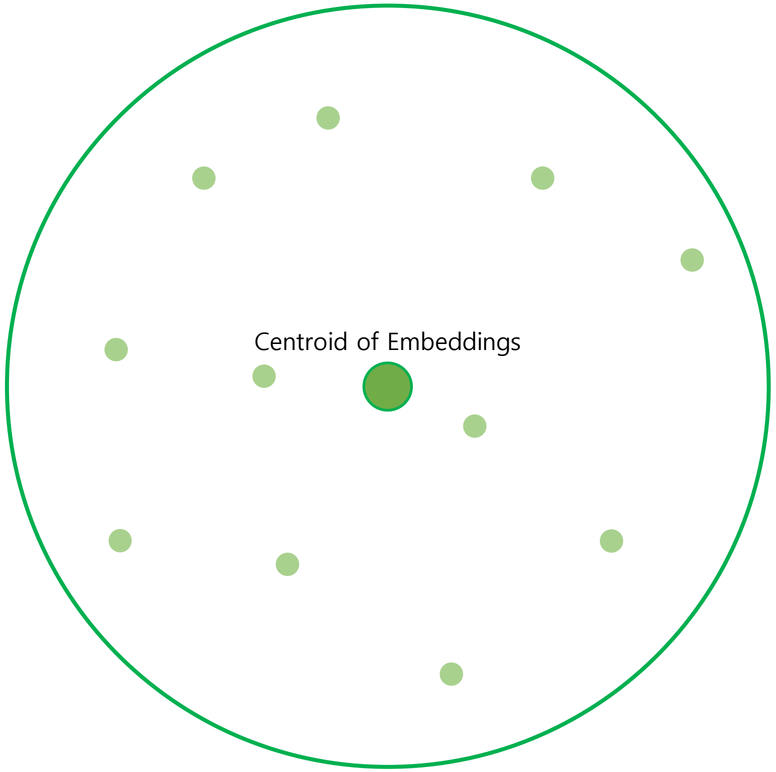
Sense 4: Evaluation

<Senses of all Tokens in Given Sentence>

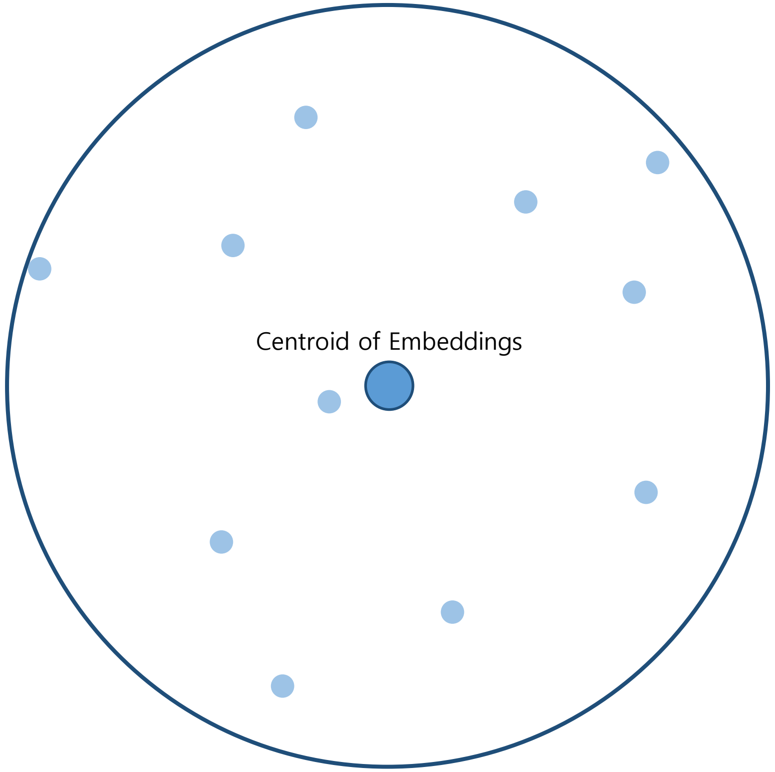
Geometry of Word Senses

-Measurement of Word Sense Disambiguation

<Nearest-neighbor Classification>

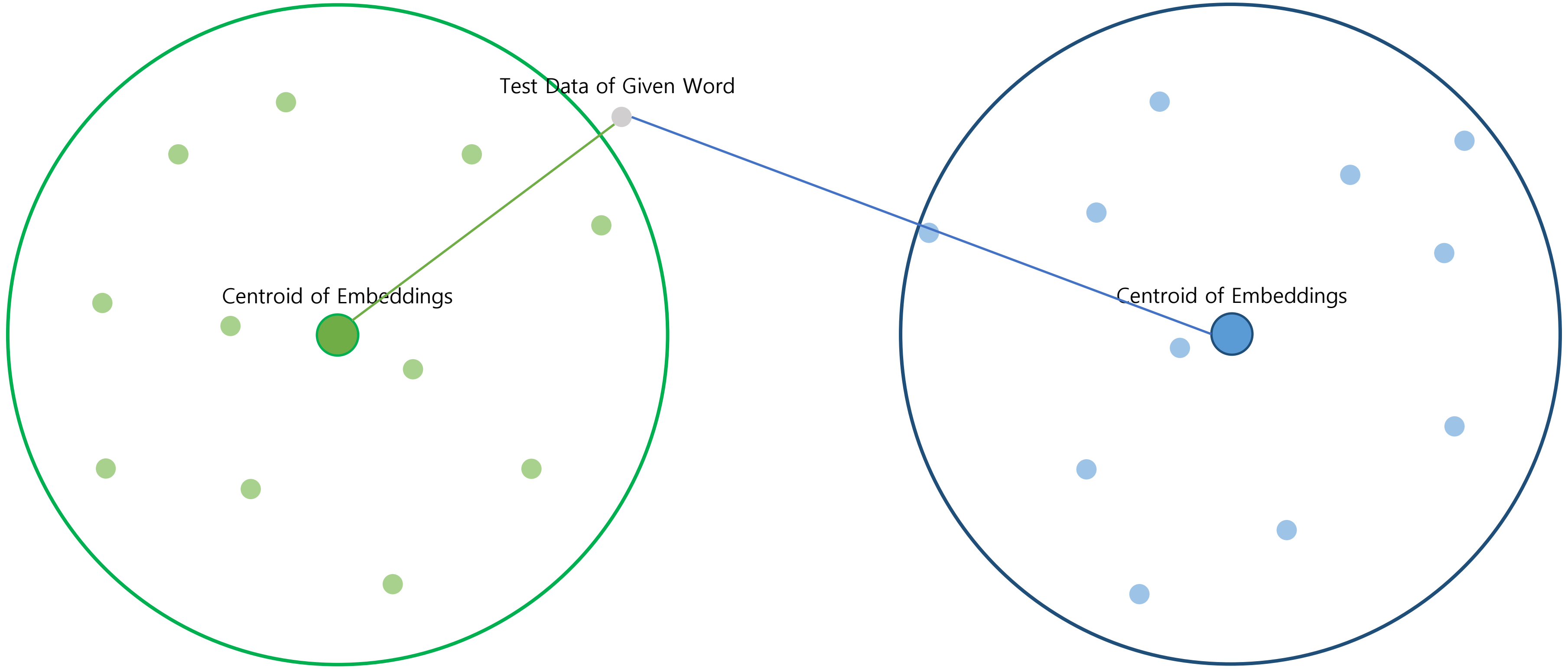


<Embeddings of a Word with Sense 1>



<Embeddings of a Word with Sense 2>

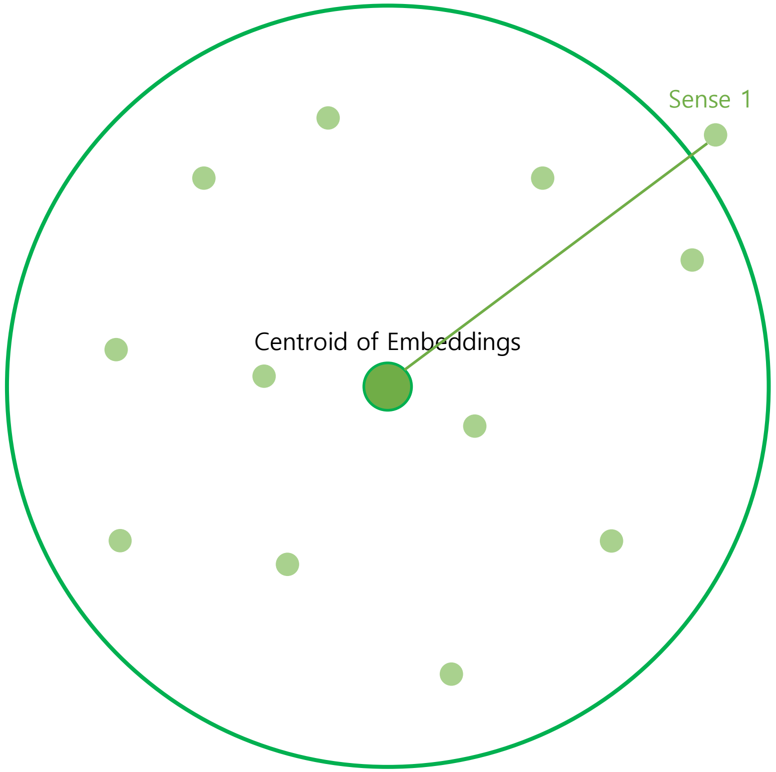
<Nearest-neighbor Classification>



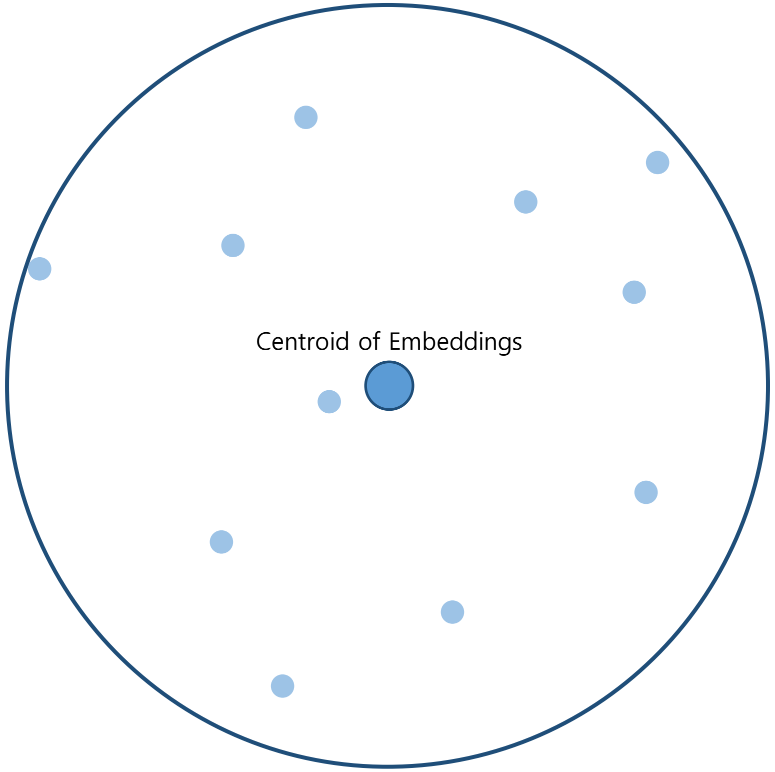
<Embeddings of a Word with Sense 1>

<Embeddings of a Word with Sense 2>

<Nearest-neighbor Classification>



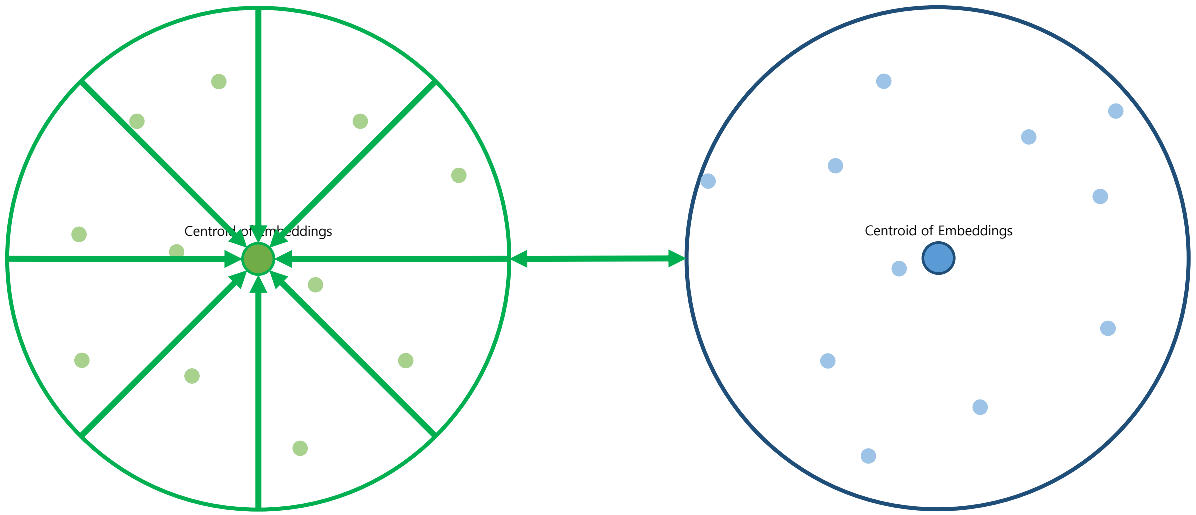
<Embeddings of a Word with Sense 1>



<Embeddings of a Word with Sense 2>

<Structural Probe for Word Sense Disambiguation>

$$\min_{B}(average\ cosine\ similarity\ of\ same\ sense\ - average\ cosine\ similarity\ of\ different\ sense)$$



<Embeddings of a Word with Sense 1> <Embeddings of a Word with Sense 2>

<Result>

Corpus	Method	F1 Score
SemCor	IMS	68.4
	IMS+emb	69.1
	IMS_{-s}+emb	**69.6**
	Context2Vec	69.0
	MFS	64.8

<Raganato et al. 2017>

Corpus	Method	F1 Score
SemCor	Baseline (most frequent sense)	64.8
	ELMo	70.1
	BERT	71.1
	BERT (w/probe)	**71.5**

<This Paper>

<Result>

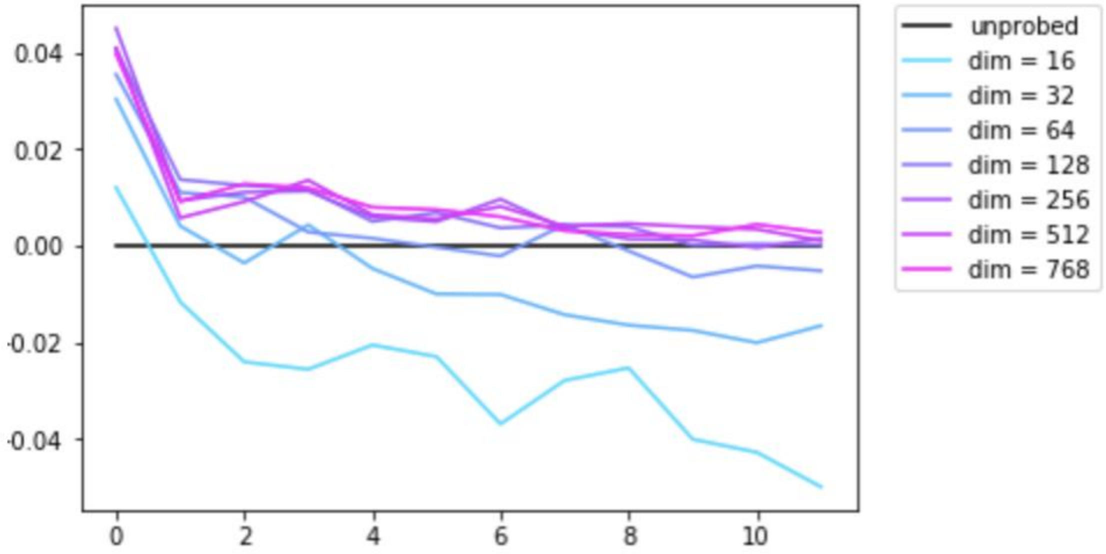
m	Trained probe	Random probe
768 (full)	71.26	70.74
512	71.52	70.51
256	71.29	69.92
128	71.21	69.56
64	70.19	68.00
32	68.01	64.62
16	65.34	61.01

<Semantic Probe % Accuracy on Final-layer BERT-base>

Geometry of Word Senses

-Measurement of Word Sense Disambiguation

<Result>



<Change in Classification Accuracy by Layer of Different Probe Dimensionalities>

<Embedding Distance and Context>

Sentence A: "He thereupon *went* to London and spent the winter talking to men of wealth."

went: to move from one place to another **(Sense 1)**

Sentence B: "He *went* prone on his stomach, the better to pursue his examination."

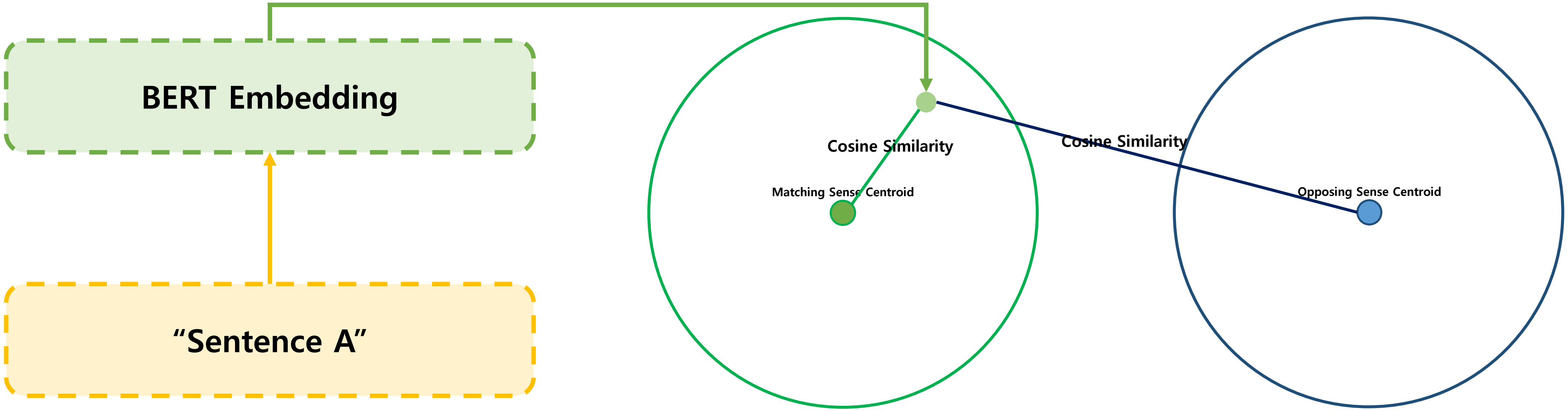
went: to enter into a specified state. **(Sense 2)**

Sense 1: *matching sense* of Sentence A

Sense 2: *opposing sense* of Sentence A

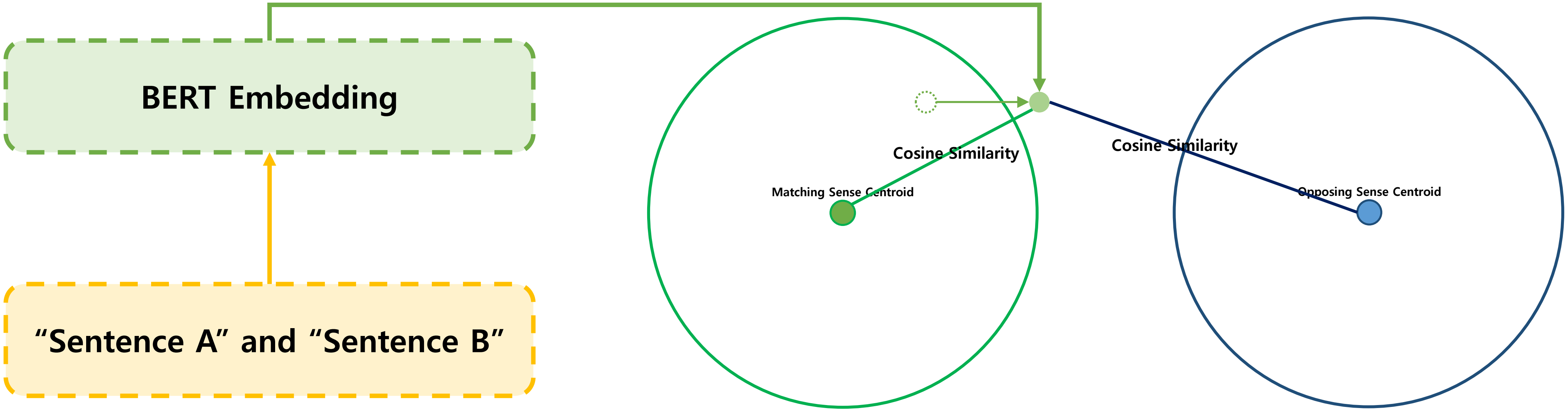
<Embedding Distance and Context>

$$individual\ similarity\ ratio = \frac{matching\ sense\ similarity}{opposing\ sense\ similarity}$$



<Embedding Distance and Context>

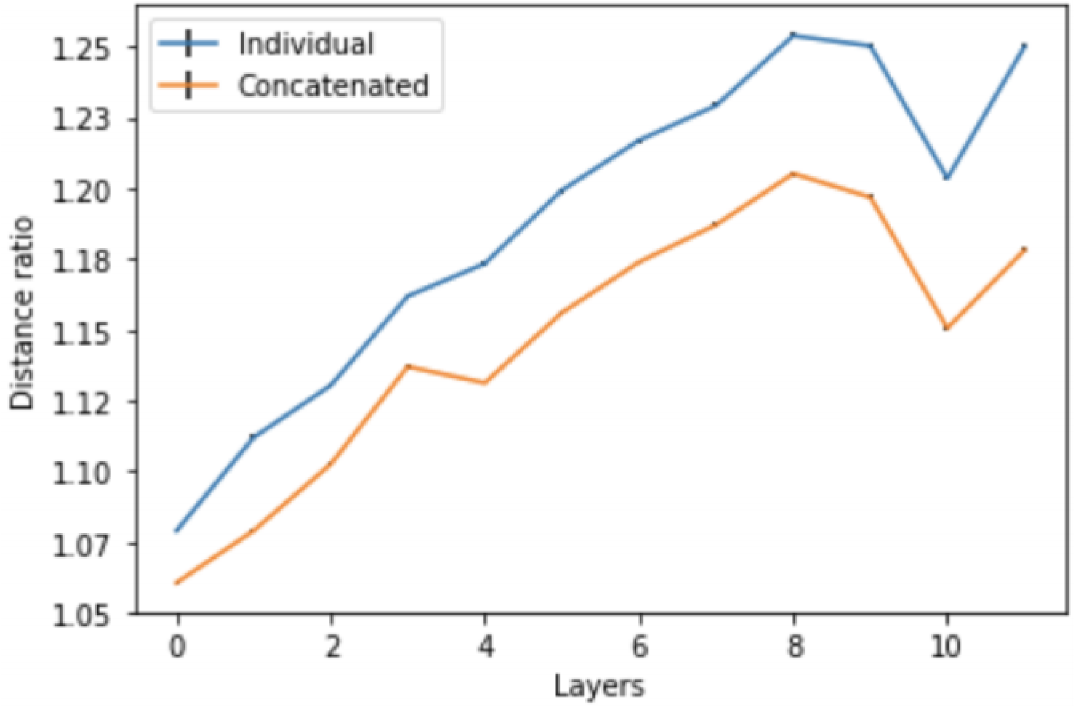
$$\textit{concatenated similarity ratio} = \frac{\textit{matching sense similarity}}{\textit{opposing sense similarity}}$$



Geometry of Word Senses

-Embedding Distance and Context

<Result>



<Average Similarity Ratio: Senses A vs. B>

Conclusion

<Conclusion>

- **There are Subspaces in Language Model Representation that Contain Syntactic and Semantic Information Respectively**

- **There are Limitations of Attention-based Model: Tokens do not Respect Semantic Boundaries, But Absorb Meaning from all Neighbors**

Any Questions?

Thank You