ERNIE 2.0

A CONTINUAL PRE-TRAINING FRAMEWORK FOR LANGUAGE UNDERSTANDING

by TAEU

CONTENTS

0. Motivation

1. ERNIE 2.0 Abstract

2. ERNIE 2.0 Detail

3. Conclusion

We hope our model learns real beauty of language

Many existing models only focus on co-occurrence of token and seq

But, we want to learn more valuable things that language has.

- EX) lexical, syntactic, semantic information so on..

HOW ??

ERNIE 2.0 CAN DO !

Abstract

Incrementally multi pre-training tasks

3 Main Pre-training Tasks : Word , Structure , Semantic

Outperforms BERT and XLNET on 16 Tasks

Model Architecture

ERNIE 2.0：A Continual Pre-training framework for Language Understanding

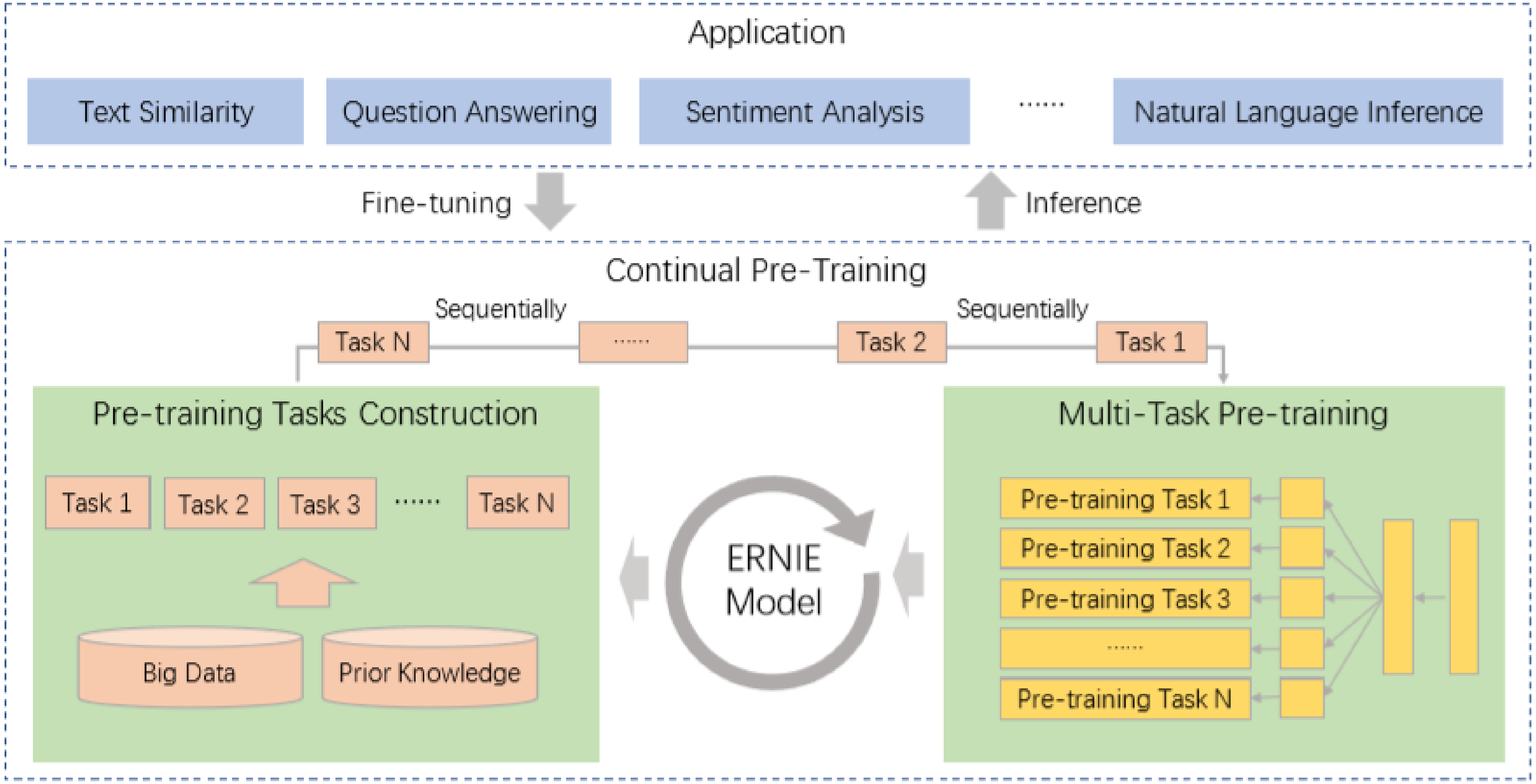


Figure 1: The framework of ERNIE 2.0, where the pre-training tasks can be incrementally constructed, the models are pre-trained through multi-task learning, and the pre-trained model is fine-tuned to adapt to various language understanding tasks.

Fine-tuning

+

Pre-training

For pre-training tasks

1. Contruct unsupervised pre-training tasks with big data and prior knowledge involved

2. Training ERNIE model via multi-task learning

What’s the difference

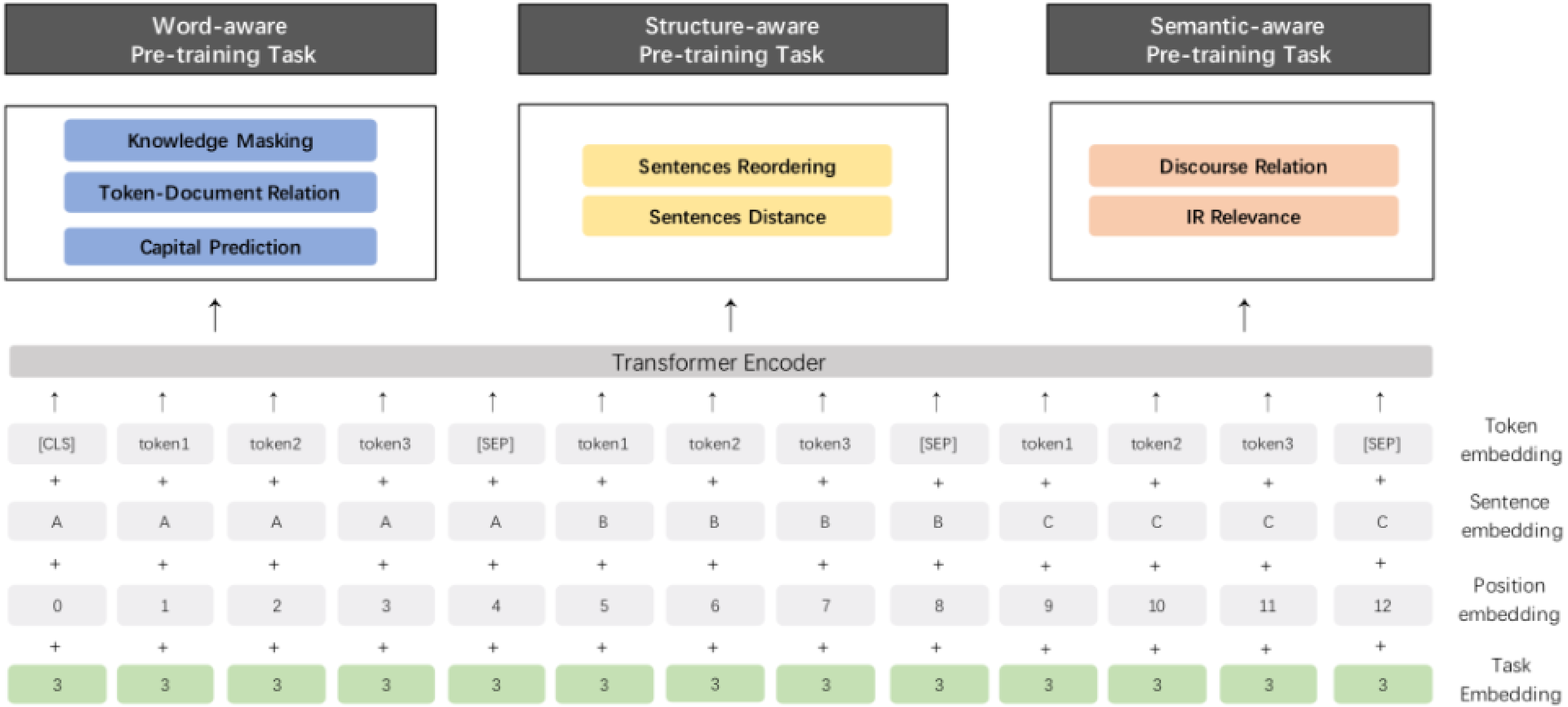
small number of pre-training objectives → A large variety of pre-training tasks

Word-aware Tasks: to handle the lexical information

Structure-aware Tasks: to capture the syntactic information

Semantic-aware Tasks: in charge of semantic signals

Pre-training Tasks



Word-aware Pre-training Task

- Knowledge Masking
- Token-Document Relation
- Capital Prediction

Structure-aware Pre-training Task

- Sentences Reordering
- Sentences Distance

Semantic-aware Pre-training Task

- Discourse Relation
- IR Relevance

Transformer Encoder

[CLS]	token1	token2	token3	[SEP]	token1	token2	token3	[SEP]	token1	token2	token3	[SEP]	Token embedding
+	+	+	+	+	+	+	+	+	+	+	+	+	
A	A	A	A	A	B	B	B	B	C	C	C	C	Sentence embedding
+	+	+	+	+	+	+	+	+	+	+	+	+	
0	1	2	3	4	5	6	7	8	9	10	11	12	Position embedding
+	+	+	+	+	+	+	+	+	+	+	+	+	
3	3	3	3	3	3	3	3	3	3	3	3	3	Task Embedding

Pre-training Tasks

Word-aware Tasks

Knowledge Masking Task : ERNIE 1.0 introduced phrase and named entity masking strategies to help the model learn the dependency information in both local contexts and global contexts.

Ex) James was [MASK] by Jeremy

Ex) [MASK] [MASK] was written by George R. R. Martin

Capitalization Prediction Task : Capitalized words usually have certain specific semantic value compared to other words in sentences. we add a task to predict whether the word is capitalized or not.

Ex) james was kidnapped by jeremy

Token-Document Relation Prediction Task : A task to predict whether the token in a segment appears in other segments of the original document. (check..)

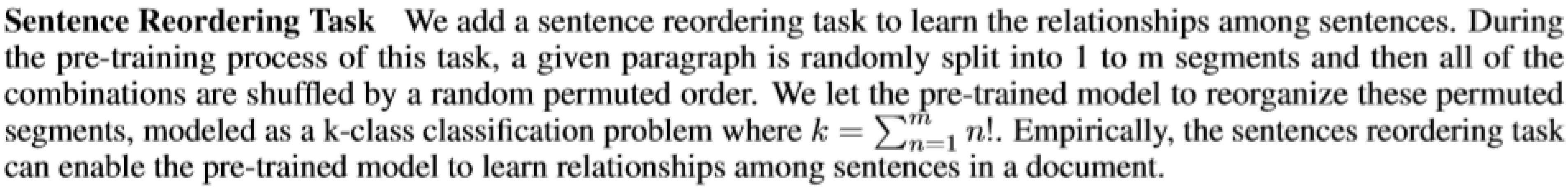
Ex) A meme is an idea, behavior ~.. // (paper) the key words of a document appearing in the segment

Pre-training Tasks

Structure-aware Tasks

Sentence Reordering Task : This task try to learn the relationships among sentences by randomly spliting a given paragraph into 1 to m segments and reorganizing these permuted segments as a standard classification task.

(check..)



Sentence Reordering Task We add a sentence reordering task to learn the relationships among sentences. During the pre-training process of this task, a given paragraph is randomly split into 1 to m segments and then all of the combinations are shuffled by a random permuted order. We let the pre-trained model to reorganize these permuted segments, modeled as a k-class classification problem where $k = \sum_{n=1}^{m} n!$. Empirically, the sentences reordering task can enable the pre-trained model to learn relationships among sentences in a document.

Sentence Distance Task : This task handles the distance between sentences as a 3-class classification problem.

- 0 : Two sentences are adjacent in the same document
- 1 : Two sentences are in the same document (not adjacent)
- 2 : Two sentences are from two different documents

Pre-training Tasks

Semantic-aware Tasks

Discourse Relation Task : A task try to predict the semantic or rhetorical relation between two sentences.

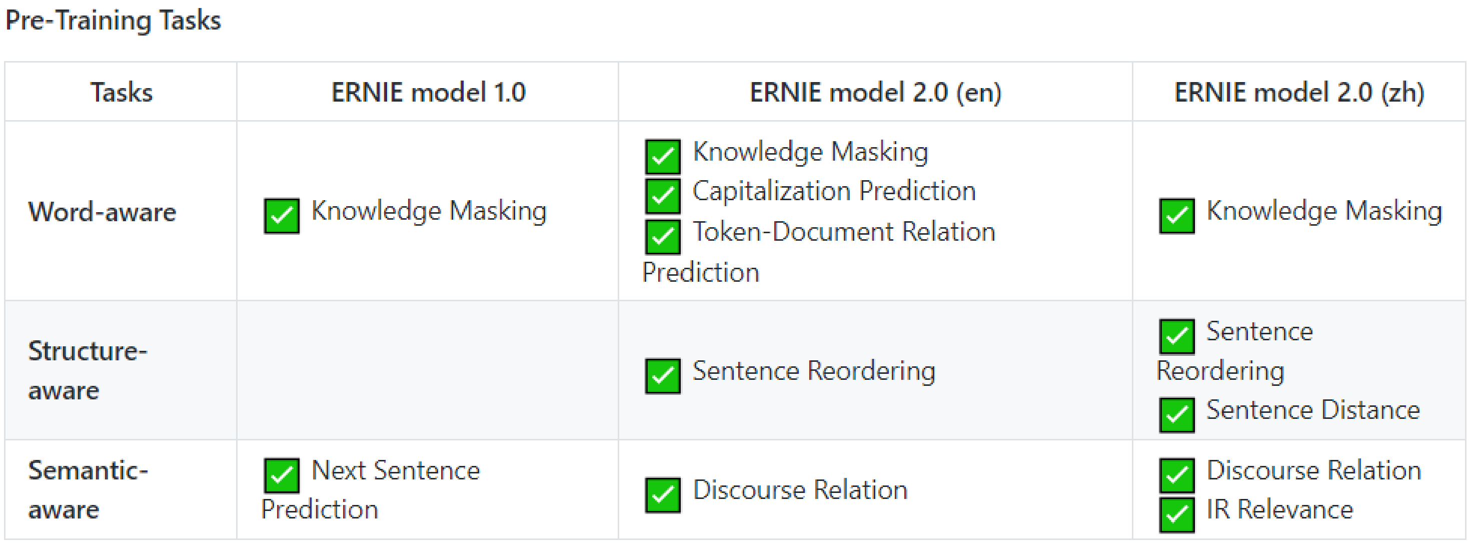
Damien Sileo, Tim Van-De-Cruys, Camille Pradel, and Philippe Muller. Mining discourse markers for unsupervised sentence representation learning. arXiv preprint arXiv:1903.11850, 2019.

Ex) I took my umbrella this morning. [because] The forecast was rain in the afternoon.

IR Relevance Task : A 3-class classification task which predicts the relationship between a query and a title.

- 0 : Strong relevance(user query and click title) ,
- 1 : weak r. (user query, not click title(just appear)),
- 2 : irr.

Pre-training Tasks



Pre-Training Tasks

Tasks	ERNIE model 1.0	ERNIE model 2.0 (en)	ERNIE model 2.0 (zh)
Word-aware	✅ Knowledge Masking	✅ Knowledge Masking ✅ Capitalization Prediction ✅ Token-Document Relation Prediction	✅ Knowledge Masking
Structure-aware		✅ Sentence Reordering	✅ Sentence Reordering ✅ Sentence Distance
Semantic-aware	✅ Next Sentence Prediction	✅ Discourse Relation	✅ Discourse Relation ✅ IR Relevance

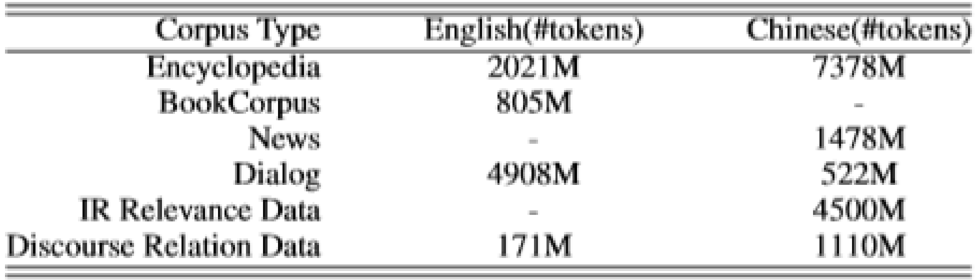
encyclopedia, news, dialogue, information retrieval and discourse relation data from Baidu Search Engine.

Wikipedia

BookCorpus

Reddit

Discovery data (for discourcse relation)



Corpus Type	English(#tokens)	Chinese(#tokens)
Encyclopedia	2021M	7378M
BookCorpus	805M	-
News	-	1478M
Dialog	4908M	522M
IR Relevance Data	-	4500M
Discourse Relation Data	171M	1110M

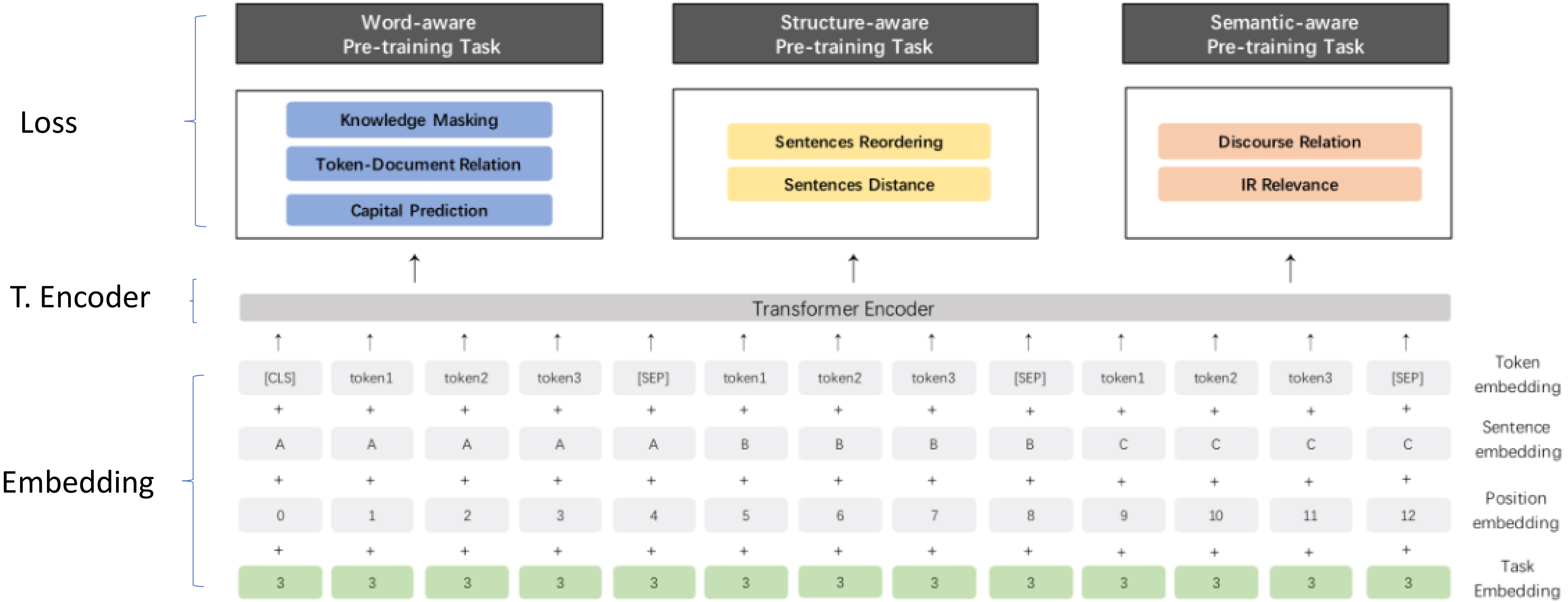
Table 1: The size of pre-training datasets.

For pre-training tasks

1. Contruct unsupervised pre-training tasks with big data and prior knowledge involved

2. Training ERNIE model via multi-task learning

Model Structure



Loss

T. Encoder

Embedding

Word-aware Pre-training Task

- Knowledge Masking
- Token-Document Relation
- Capital Prediction

Structure-aware Pre-training Task

- Sentences Reordering
- Sentences Distance

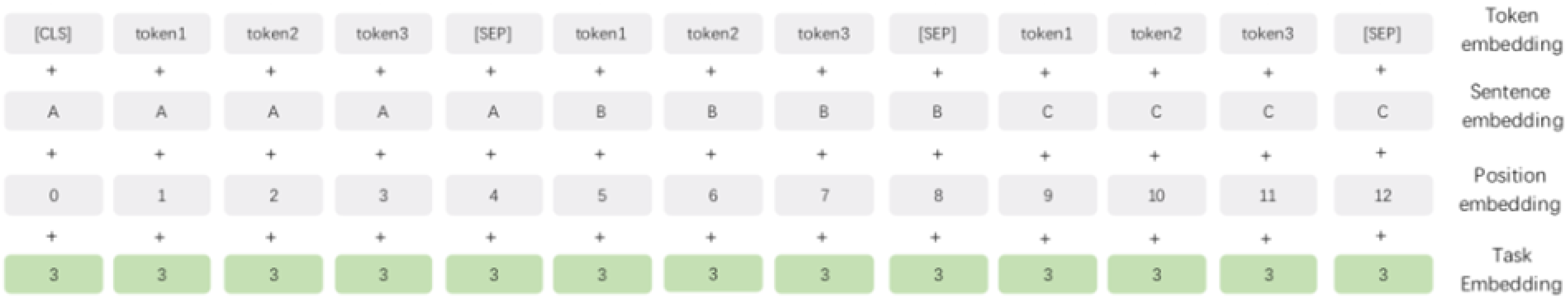
Semantic-aware Pre-training Task

- Discourse Relation
- IR Relevance

Transformer Encoder

[CLS]	token1	token2	token3	[SEP]	token1	token2	token3	[SEP]	token1	token2	token3	[SEP]	Token embedding
+	+	+	+	+	+	+	+	+	+	+	+	+	
A	A	A	A	A	B	B	B	B	C	C	C	C	Sentence embedding
+	+	+	+	+	+	+	+	+	+	+	+	+	
0	1	2	3	4	5	6	7	8	9	10	11	12	Position embedding
+	+	+	+	+	+	+	+	+	+	+	+	+	
3	3	3	3	3	3	3	3	3	3	3	3	3	Task Embedding

Model settings of Embedding



[CLS]	token1	token2	token3	[SEP]	token1	token2	token3	[SEP]	token1	token2	token3	[SEP]	Token embedding
+	+	+	+	+	+	+	+	+	+	+	+	+	
A	A	A	A	A	B	B	B	B	C	C	C	C	Sentence embedding
+	+	+	+	+	+	+	+	+	+	+	+	+	
0	1	2	3	4	5	6	7	8	9	10	11	12	Position embedding
+	+	+	+	+	+	+	+	+	+	+	+	+	
3	3	3	3	3	3	3	3	3	3	3	3	3	Task Embedding

[CLS] is added to the first place of the sequence

[SEP] is added as the separator in the intervals of the segments for the multiple input segment tasks

Task Embedding

Different tasks with an id ranging from 0 to N

We can use any task id to initialize our model in the fine-tuning process

Model settings of transformer

ERNIE 2.0 Base = BERT Base

12 layers, 12 self-attention heads and 768-dimensional of hidden size

48 NVidia v100 GPU

ERNIE 2.0 Large = Bert Large

24 layers, 16 self-attention heads and 1024-dimensional of hidden size

64 NVidia v100 GPU

Loss

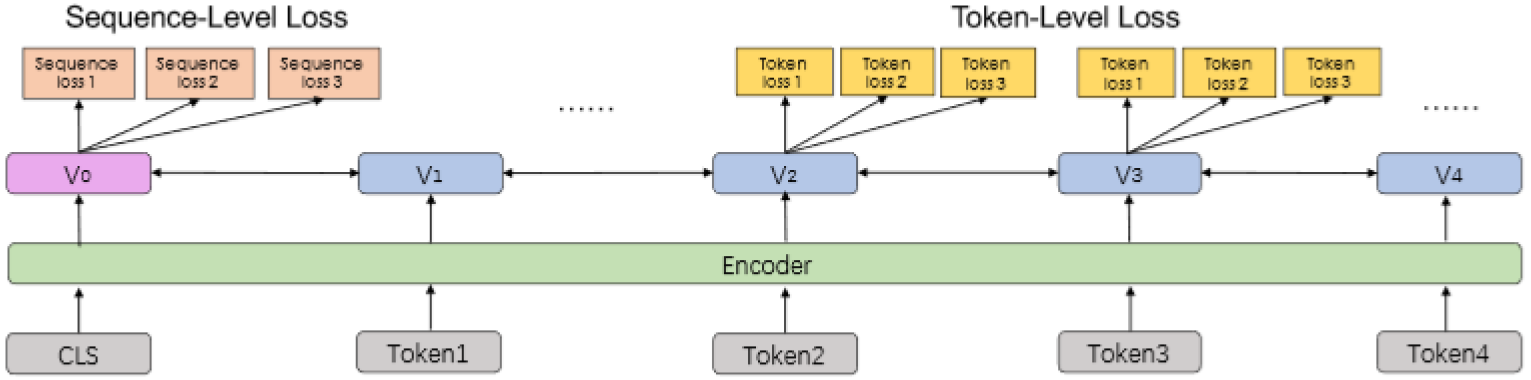
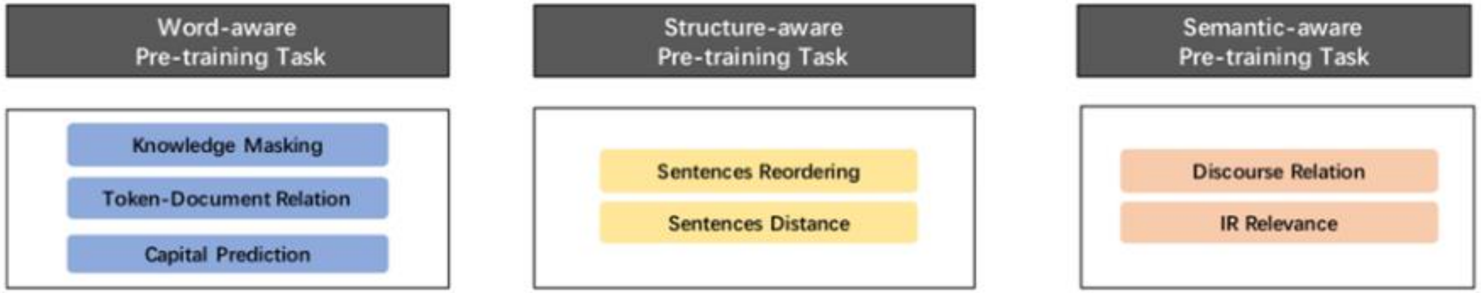


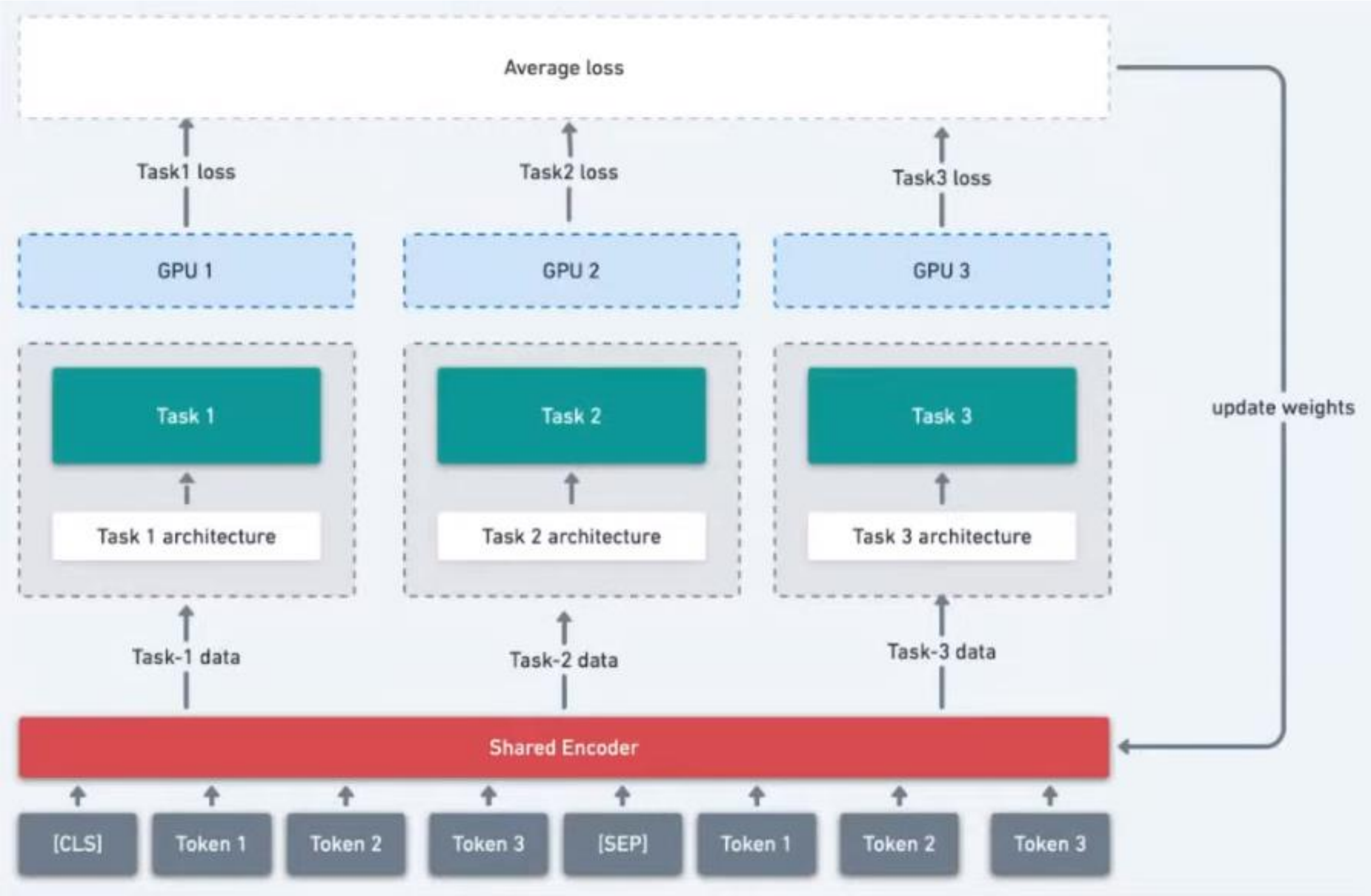
Figure 2: The architecture of multi-task pre-training in the ERNIE 2.0 framework, in which the encoder can be recurrent neural networks or a deep transformer.

TOKEN LEVEL



SEQ LEVEL

Training – multitask learning



Training – multitask learning

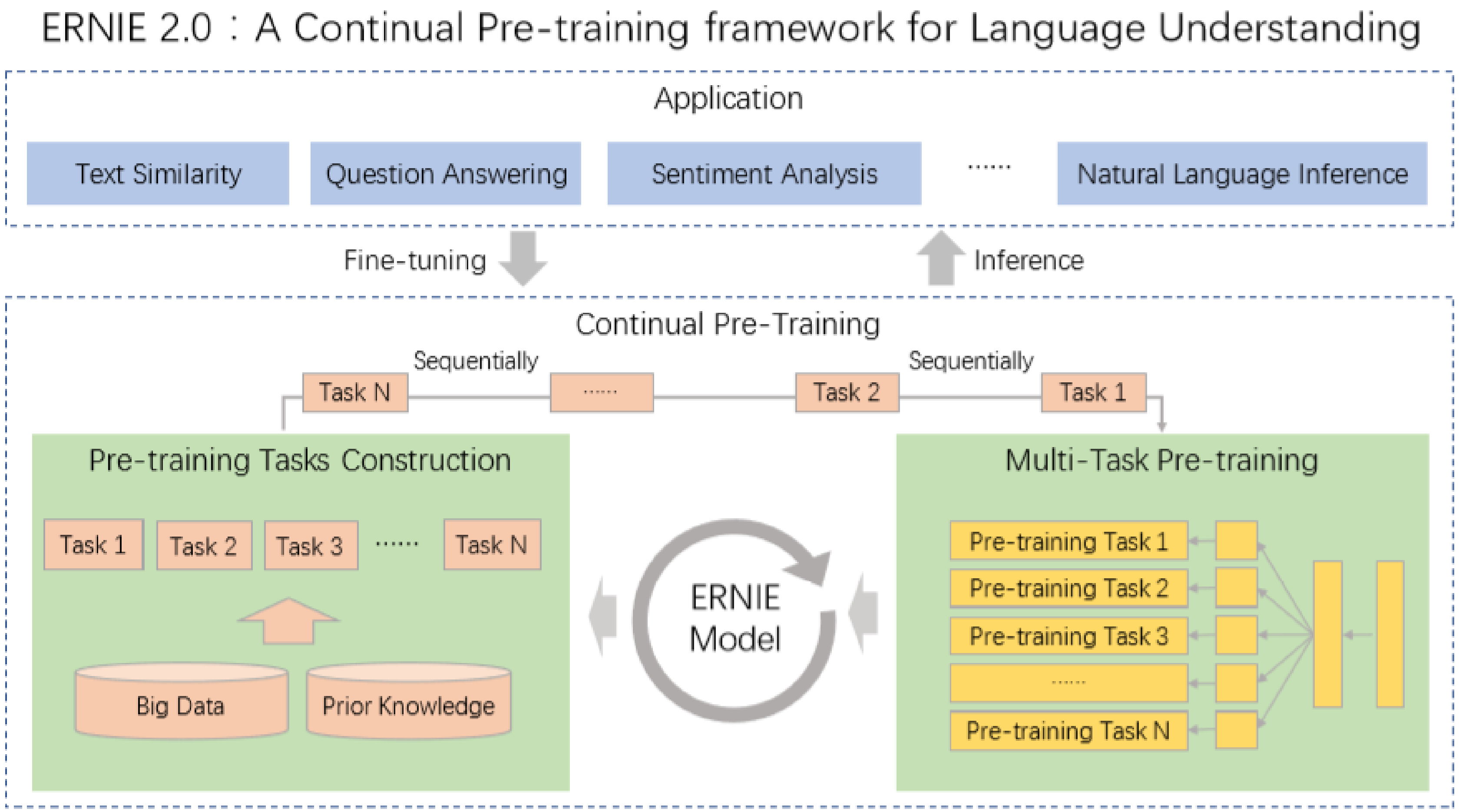


Figure 1: The framework of ERNIE 2.0, where the pre-training tasks can be incrementally constructed, the models are pre-trained through multi-task learning, and the pre-trained model is fine-tuned to adapt to various language understanding tasks.

Sequentially learning means…

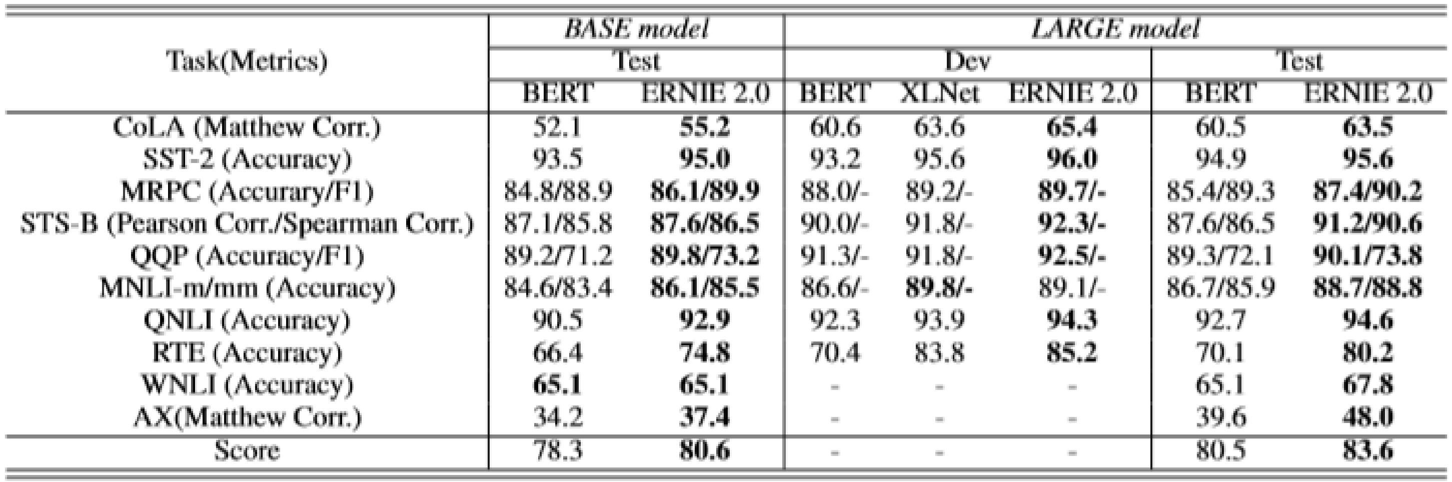
incrementally update
through the tasks

(1) update,
(1,2) update
, … ,
(1, 2, …, N) update

Am I right..?

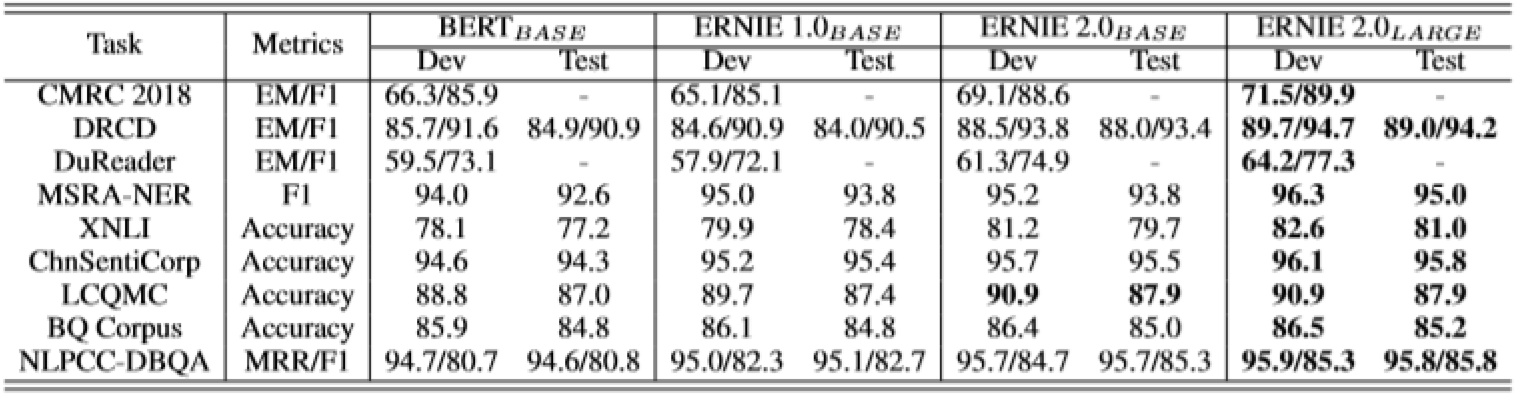
(paper) Whenever a new task is introduced, it would be trained with the previous ones to make sure that the model does not forget the knowledge it has learnt.

Result…



Task(Metrics)	*BASE model*		*LARGE model*				
	Test		Dev			Test	
	BERT	ERNIE 2.0	BERT	XLNet	ERNIE 2.0	BERT	ERNIE 2.0
CoLA (Matthew Corr.)	52.1	**55.2**	60.6	63.6	**65.4**	60.5	**63.5**
SST-2 (Accuracy)	93.5	**95.0**	93.2	95.6	**96.0**	94.9	**95.6**
MRPC (Accurary/F1)	84.8/88.9	**86.1/89.9**	88.0/-	89.2/-	**89.7/-**	85.4/89.3	**87.4/90.2**
STS-B (Pearson Corr./Spearman Corr.)	87.1/85.8	**87.6/86.5**	90.0/-	91.8/-	**92.3/-**	87.6/86.5	**91.2/90.6**
QQP (Accuracy/F1)	89.2/71.2	**89.8/73.2**	91.3/-	91.8/-	**92.5/-**	89.3/72.1	**90.1/73.8**
MNLI-m/mm (Accuracy)	84.6/83.4	**86.1/85.5**	86.6/-	**89.8/-**	89.1/-	86.7/85.9	**88.7/88.8**
QNLI (Accuracy)	90.5	**92.9**	92.3	93.9	**94.3**	92.7	**94.6**
RTE (Accuracy)	66.4	**74.8**	70.4	83.8	**85.2**	70.1	**80.2**
WNLI (Accuracy)	**65.1**	**65.1**	-	-	-	65.1	**67.8**
AX(Matthew Corr.)	34.2	**37.4**	-	-	-	39.6	**48.0**
Score	78.3	**80.6**	-	-	-	80.5	**83.6**

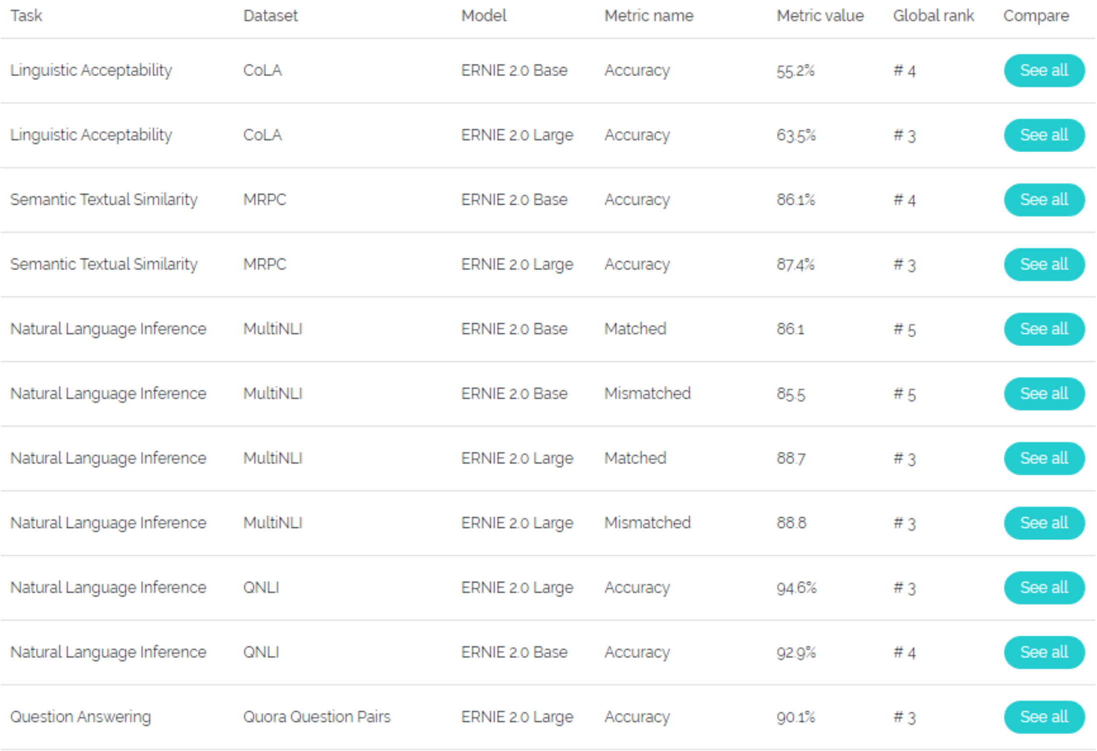
Table 6: The results on GLUE benchmark, where the results on dev set are the median of five experimental results and the results on test set are scored by the GLUE evaluation server (`https://gluebenchmark.com/leaderboard`). The state-of-the-art results are in bold. All of the fine-tuned models of AX is trained by the data of MNLI.



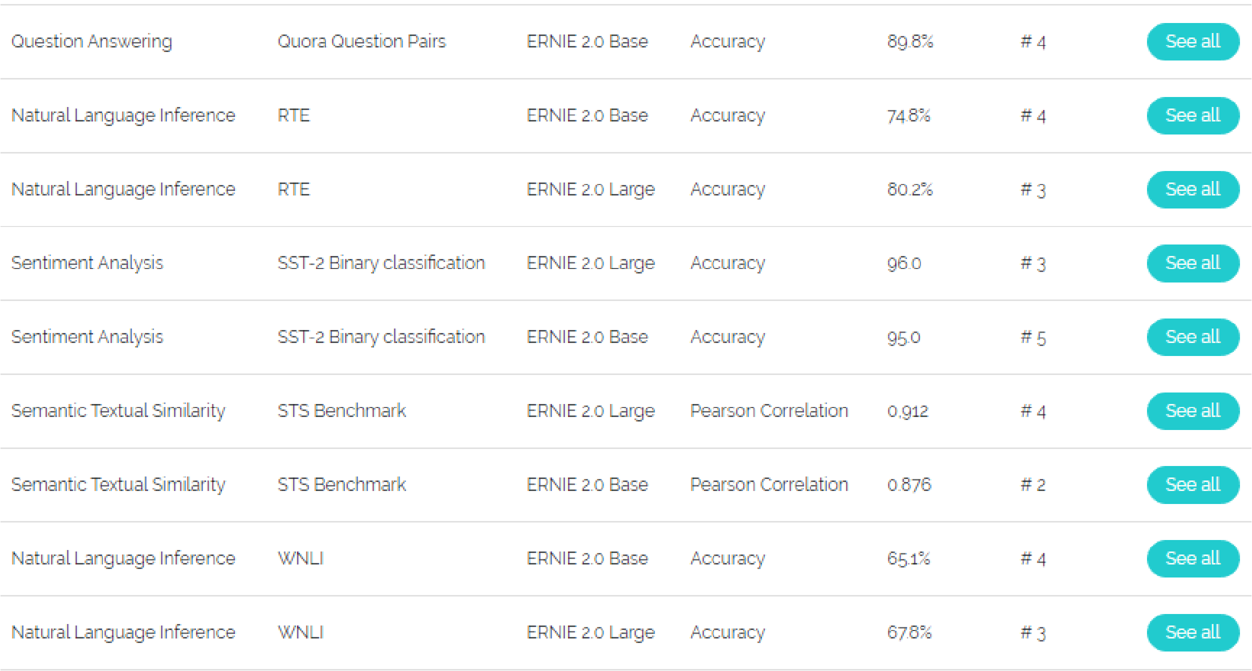
Task	Metrics	BERT_{BASE}		ERNIE 1.0_{BASE}		ERNIE 2.0_{BASE}		ERNIE 2.0_{LARGE}	
		Dev	Test	Dev	Test	Dev	Test	Dev	Test
CMRC 2018	EM/F1	66.3/85.9	-	65.1/85.1	-	69.1/88.6	-	**71.5/89.9**	-
DRCD	EM/F1	85.7/91.6	84.9/90.9	84.6/90.9	84.0/90.5	88.5/93.8	88.0/93.4	**89.7/94.7**	**89.0/94.2**
DuReader	EM/F1	59.5/73.1	-	57.9/72.1	-	61.3/74.9	-	**64.2/77.3**	-
MSRA-NER	F1	94.0	92.6	95.0	93.8	95.2	93.8	**96.3**	**95.0**
XNLI	Accuracy	78.1	77.2	79.9	78.4	81.2	79.7	**82.6**	**81.0**
ChnSentiCorp	Accuracy	94.6	94.3	95.2	95.4	95.7	95.5	**96.1**	**95.8**
LCQMC	Accuracy	88.8	87.0	89.7	87.4	**90.9**	**87.9**	**90.9**	**87.9**
BQ Corpus	Accuracy	85.9	84.8	86.1	84.8	86.4	85.0	**86.5**	**85.2**
NLPCC-DBQA	MRR/F1	94.7/80.7	94.6/80.8	95.0/82.3	95.1/82.7	95.7/84.7	95.7/85.3	**95.9/85.3**	**95.8/85.8**

Table 7: The results of 9 common Chinese NLP tasks. ERNIE 1.0 indicates our previous model ERNIE[4]. The reported results are the average of five experimental results, and the state-of-the-art results are in bold.

But RoBERTa and XLNET are better than ERNIE 2.0



Task	Dataset	Model	Metric name	Metric value	Global rank	Compare
Linguistic Acceptability	CoLA	ERNIE 2.0 Base	Accuracy	55.2%	# 4	See all
Linguistic Acceptability	CoLA	ERNIE 2.0 Large	Accuracy	63.5%	# 3	See all
Semantic Textual Similarity	MRPC	ERNIE 2.0 Base	Accuracy	86.1%	# 4	See all
Semantic Textual Similarity	MRPC	ERNIE 2.0 Large	Accuracy	87.4%	# 3	See all
Natural Language Inference	MultiNLI	ERNIE 2.0 Base	Matched	86.1	# 5	See all
Natural Language Inference	MultiNLI	ERNIE 2.0 Base	Mismatched	85.5	# 5	See all
Natural Language Inference	MultiNLI	ERNIE 2.0 Large	Matched	88.7	# 3	See all
Natural Language Inference	MultiNLI	ERNIE 2.0 Large	Mismatched	88.8	# 3	See all
Natural Language Inference	QNLI	ERNIE 2.0 Large	Accuracy	94.6%	# 3	See all
Natural Language Inference	QNLI	ERNIE 2.0 Base	Accuracy	92.9%	# 4	See all
Question Answering	Quora Question Pairs	ERNIE 2.0 Large	Accuracy	90.1%	# 3	See all



Question Answering	Quora Question Pairs	ERNIE 2.0 Base	Accuracy	89.8%	# 4	See all
Natural Language Inference	RTE	ERNIE 2.0 Base	Accuracy	74.8%	# 4	See all
Natural Language Inference	RTE	ERNIE 2.0 Large	Accuracy	80.2%	# 3	See all
Sentiment Analysis	SST-2 Binary classification	ERNIE 2.0 Large	Accuracy	96.0	# 3	See all
Sentiment Analysis	SST-2 Binary classification	ERNIE 2.0 Base	Accuracy	95.0	# 5	See all
Semantic Textual Similarity	STS Benchmark	ERNIE 2.0 Large	Pearson Correlation	0.912	# 4	See all
Semantic Textual Similarity	STS Benchmark	ERNIE 2.0 Base	Pearson Correlation	0.876	# 2	See all
Natural Language Inference	WNLI	ERNIE 2.0 Base	Accuracy	65.1%	# 4	See all
Natural Language Inference	WNLI	ERNIE 2.0 Large	Accuracy	67.8%	# 3	See all

https://paperswithcode.com/paper/ernie-20-a-continual-pre-training-framework

Discussion Point

0. Is ERNIE 2.0 a scalable approach?
1. Is multi pre-training tasks really working for improvement? And how about sequentially learning ?
2. How much does the order of pre-training tasks affect results?
3. How much improvement come from architecture(multi pre-training tasks) vs size of training data
4. Is there other potential pre-training tasks?
5. What if combine RoBERTa’s method ?
6. Anything will be good point! Tell us!

THANK YOU

REFERENCES

https://arxiv.org/abs/1907.12412

https://github.com/PaddlePaddle/ERNIE

https://paperswithcode.com/paper/ernie-20-a-continual-pre-training-framework

https://www.youtube.com/watch?v=8K1IX7VJ5Fc&t=4027s