Multi-task Sequence Prediction for Natural Language Processing

Lyheang Ung

November 18th, 2021

Content

1 Introduction
2 Related Work
3 Datasets & Methods
4 Results
5 Conclusion

1

Introduction

1.1 Background

Multi-task Learning (MTL) is a learning paradigm in machine learning that aims to leverage applicable information in multiple related tasks to help improve the generalization performance of all the tasks.

Transformer is a novel encoder-decoder architecture that based on attention-mechanism for transforming one sequence into another without the use of recurrent networks.

1.2 Problems

LSTM, a variant of **RNN** that can not be trained in parallel due to sequential processing behavior of recurrent networks.

This results in higher demand of resources to train and longer computation times than attention-based models.

1.3 Objectives

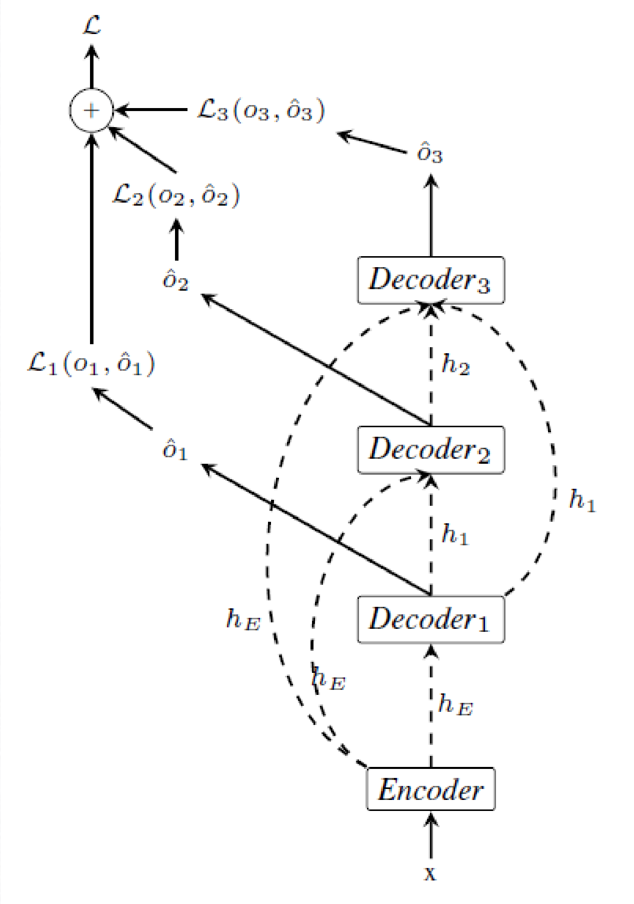
- ❖ Extend the exisiting **cascade multi-task system** based on **LSTM** architecture with a sophisitcated architecture, **Transformer**.

- ❖ Study the behavior of **cascaded multi-task learning** on different **NLP problems** :
  - ➢ Morpho-Syntactic Tagging
  - ➢ Machine Translation
  - ➢ Constituent Syntactic Analysis

2

Related Work

2.1 Multi-Task Sequence Prediction System



Cascaded One-to-Many setting
[Elisa et al. 2020]

3

Datasets & Methods

3.1 Datasets

3.1.1 TIGER

	Training		Dev		Test	
# sentences	40 472		5 000		5 000	
	Words	Labels	Words	Labels	Words	Labels
# tokens	719 530	-	76 704	-	92 004	-
dictionary	77 220	681	15 852	501	20 149	537
OOV%	-	-	30,90	0,01	37,18	0,015

TIGER was annotated with rich **morpho-syntactic information** including PoS tags, gender, numbers, cases, conjugation information for verbs and other inflection information.

Ex: German sentence : Ehemaliger Angestellter in Untersuchungshaft
POS : ADJA NN APPR NN
Morpho : ADJA.Pos.Nom.Sg.Masc NN.Nom.Sg.Masc APPR NN.Dat.Sg.Fem

3.1 Datasets

3.1.2 WMT14 Europarl v7

	Training	Test	Dev
# Sentences	532 940	1 503	3 003

	Training	Test	Dev
English	15 190 616	44 350	85 010
Czech	14 222 136	79 482	47 275
French	17 042 729	95 365	52 868
German	15 645 951	49 639	87 525

Ex: En -> Cz -> Fr -> De
En ->Cz -> De -> Fr
En -> Fr -> De -> Cz

3.1 Datasets

3.1.3 WSJ

	Training	Test	Dev
#Sentences	39 832	2 416	1 700

[Marco and Loïc, 2019]

WSJ Base

	Training	Test	Dev
English Sentences	950 028	56 684	40 117
PoS	950 028	56 684	40 117
Chunk	1 895 952	113 348	79 227
Tree	6 145 107	366 812	257 467

WSJ BPE 16000

	Training	Test	Dev
English Sentences	1 090 514	65 546	46 349
PoS	950 028	56 684	40 117
Chunk	2 036 438	122 210	85 459
Tree	6 285 565	375 643	263 693

WSJ BPE 32000

	Training	Test	Dev
English Sentences	1 767 146	103 836	72 198
PoS	950 152	56 697	40 119
Chunk	2 713 070	160 500	111 308
Tree	6 962 325	413 948	289 552

3.1 Datasets

3.1.3 WSJ

Ex: Sentence : We 're about to see if advertising works .

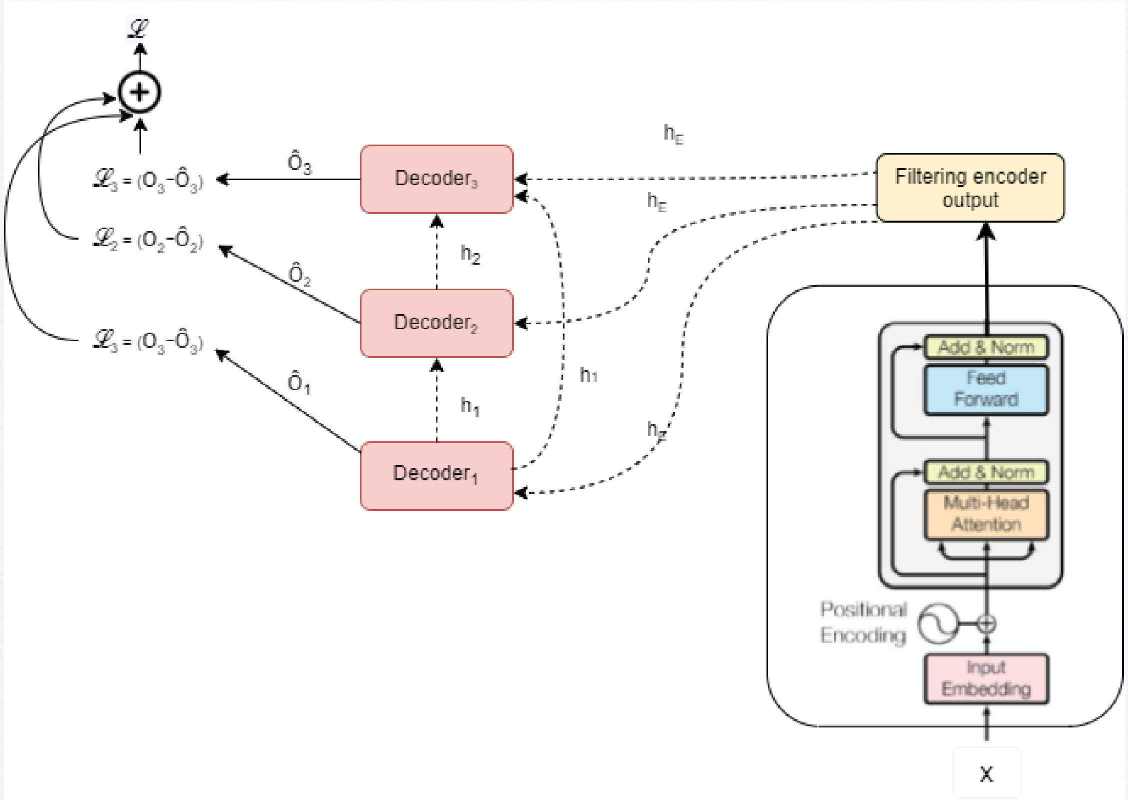
POS : PRP VBP IN TO VB IN NN VBZ .

Chunk : (We) ('re) (about) (to) (see) (if) (advertising works) (.)

Tree : (TOP (S (NP (PRP We)) (VP (VBP 're) (VP (IN about) (S (VP (TO to) (VP (VB see) (SBAR (IN if) (S (NP (NN advertising)) (VP (VBZ works))))))))) (. .)))

3.2 Methods

3.2.1 Extending Transformer to Multi-task System



4

Results

4.1 Training

The default hyperparameters use for experiment [Elissa et al, 2020] :

- Learning rate = 0.005
- Dropout = 0.5
- Clipnorm = 5.0
- Weight decay = 0.0001
- Batch size = 5
- Model size = 256
- Attention heads = 4
- Encoder and Decoder layers = 4
- Optimizer = Adam

4.2 Post-processing

4.2.1 WMT14

```
S-1775  Let us try to avoid that .
T-1775  Sna@@ ž@@ me se tomu zabránit .
H-1775  -0.7514205574989319     Sna@@ ž@@ me se to vyhnout .
```

```
S-2791  We are counting on this .
T-2791  Wir zählen darauf ! _SEQ_SEP_ Nous comptons là @-@ dessus . _SEQ_SEP_ Po@@ čít@@ áme s tím .
H-2791  -0.7300655990839005     Wir zählen darauf . _SEQ_SEP_ Nous comptons sur ce point . _SEQ_SEP_ Na to spolé@@ háme .
```

4.2 Post-processing

4.2.2 WSJ

```
S-294   P@@ A@@ R@@ I@@ S :
T-294   NNS : _SEQ_SEP_ ( TOP ( NP ( NNS P@@ A@@ R@@ I@@ S ) ( : : ) ) )
H-294   -0.0015219401320791803  NNS : _SEQ_SEP_ ( TOP ( NP ( NNS P@@ A@@ R@@ I@@ S ) ( : : ) ) )
```

```
S-991   There were no new issues .
T-991   EX VBD DT JJ NNS . _SEQ_SEP_ ( There ) ( were no new issues ) ( . ) _SEQ_SEP_ ( TOP ( S ( NP
( EX There ) ) ( VP ( VBD were ) ( NP ( DT no ) ( JJ new ) ( NNS issues ) ) ) ( . . ) ) )
H-991   -0.00023972347662493l2  EX VBD DT JJ NNS . _SEQ_SEP_ ( There ) ( were no new issues ) ( . ) _
SEQ_SEP_ ( TOP ( S ( NP ( EX There ) ) ( VP ( VBD were ) ( NP ( DT no ) ( JJ new ) ( NNS issues ) ) )
 ( . . ) ) )
```

4.3 Results

4.3.1 TIGER

	lr	dropout	Loss	
			Training Loss	Validation Loss
LSTM	0.0005	0.5	**0.173**	**0.257**
Transformer	0.0005	0.5	1.973	1.887
	0.001	0.12	**1.531**	**1.464**
	0.001	0.25	1.747	1.785
	0.001	0.37	1.837	1.837
	0.000125	0.12	2.067	1.775
	0.000125	0.25	2.152	1.938
	0.000125	0.37	2.266	2.077

Losses of **LSTM** VS **Transformer**

	lr	dropout	Tasks	
			PoS Tags	Morpho
LSTM	0.0005	0.5	**98.19**	**93.92**
Transformer	0.0005	0.5	19.46	9.31
	0.001	0.12	**34.25**	**22.08**
	0.001	0.25	23.02	12.68
	0.001	0.37	21.25	10.96
	0.000125	0.12	24.40	14.08
	0.000125	0.25	20.12	9.93
	0.000125	0.37	16.77	7.50

Accuracy(%) on **POS** and **Morpho**

4.3 Results

4.3.2 WMT14

Batch Size	Learning Rate	Model Size	En-Cs	En-De
10	0.0005	256	24.96	27.32
16	0.0005	256	27.07	28.40
32	0.0005	256	27.07	29.22
32	0.001	512	**29.65**	**31.49**
32	0.005	512	0.26	3.86
32	0.01	512	0	0.59
32	0.001	768	27.31	29.92
32	0.001	1024	28.26	29.61
32	0.005	1024	0	0
32	0.01	1024	0	0

BLEU scores on single task translation :

- English -> Czech
- English -> German

Batch Size	Learning Rate	Model Size	En-Cs-De		En-De-Cs	
			Cs	De	De	Cs
10	0.0005	256	22.67	24.68	24.95	23.45
16	0.0005	256	25.41	26.53	26.42	24.57
32	0.0005	256	25.89	27.46	26.03	24.61
32	0.001	512	**28.56**	**27.56**	**30.21**	**26.59**
32	0.005	512	**0.64**	**0.34**	**0.60**	**0**
32	0.01	512	0	0	0	0
32	0.001	768	**3.20**	**4.17**	28.01	26.57
32	0.001	1024				
32	0.005	1024	0	0		
32	0.01	1024				

BLEU scores on multi-task translations (2 Tasks) :

- English -> Czech -> German
- English -> German -> Czech

4.3 Results

4.3.2 WMT14

Batch size	lr	Model Size	Loss	
			Training Loss	Validation Loss
16	0.0005	256	8.204	6.458
16	0.001	256	**7.674**	**5.978**
16	0.001	512	OOM	OOM
16	0.005	256		
16	0.01	256		
32	0.001	128	OOM	OOM
32	0.001	256	OOM	OOM
32	0.001	512	OOM	OOM

Losses on Multi-task Translation (3 Tasks)

	Cs	Fr	De
En-Cs-Fr-De	26.95	37.93	25.62
En-De-Cs-Fr	25.98	37.20	27.06
En-De-Fr-Cs	25.70	36.89	26.95

BLEU scores on Multi-task Translation (3 Tasks)

4.3 Results

4.3.3 WSJ

4.3.3.1 Training Losses and Validation Losses

	Base	BPE16000	BPE32000
Training loss	0.3	0.251	**0.138**
Validation loss	0.291	0.137	**0.107**

Losses on **two tasks** model

	Base	BPE16000	BPE32000
Training loss	0.318	0.358	**0.177**
Validation loss	0.335	0.185	**0.124**

Losses on **three tasks** model

4.3 Results

4.3.3 WSJ

4.3.3.2 Evaluation Using Multi-task System

	Base	BPE16000	BPE32000
PoS Tags	97.09	97.02	97.48
Parse Tree	60.08	56.62	59.42

Accuracy(%) for **two tasks** model

	Base	BPE16000	BPE32000
PoS Tags	**96.86**	92.29	96.82
Chunk	**57.86**	52.90	56.89
Parse Tree	**52.30**	47.17	52.03

Accuracy(%) for **three tasks** model

4.3 Results

4.3.3 WSJ

4.3.3.3 Evaluation on Parse Tree Using Evalb

	Base	BPE16000	BPE32000
Recall	**86.83**	82.06	84.22
Precision	**87.21**	83.54	85.82
FMeasure	**87.02**	82.80	85.01

Two tasks model

	Base	BPE16000	BPE32000
Recall	**81.51**	76.20	79.47
Precision	**85.08**	81.51	83.52
FMeasure	**83.25**	78.77	81.44

Three tasks model

Evalb is a bracket scoring program. It reports precision, recall, F-measure, non crossing and tagging accuracy for given data (parse tree).

5

Conclusion

5.1 Conclusion

The proposed Transformer architecture unexpectedly produced low results on the TIGER corpus. We conclude that problem is more related to **hyperparameters optimal choice** not the implementation, and that is not a trivial problem as long as the **Transformer encoder** and **LSTM decoder** hybrid is kept.

The experiments on **multi-task translation** using **cascading multi-task system** based on **LSTM** architecture proved that jointly learn to translate multiple languages does not perform better than the single task counterpart.

The same conclusion also applied to the constituent parse tree, based on the outcomes of the **two tasks** and **three tasks** models. We believe that the **chunks** are actually not improving the results of **PoS tags** and **parse trees**.

Thank You For Your Attention