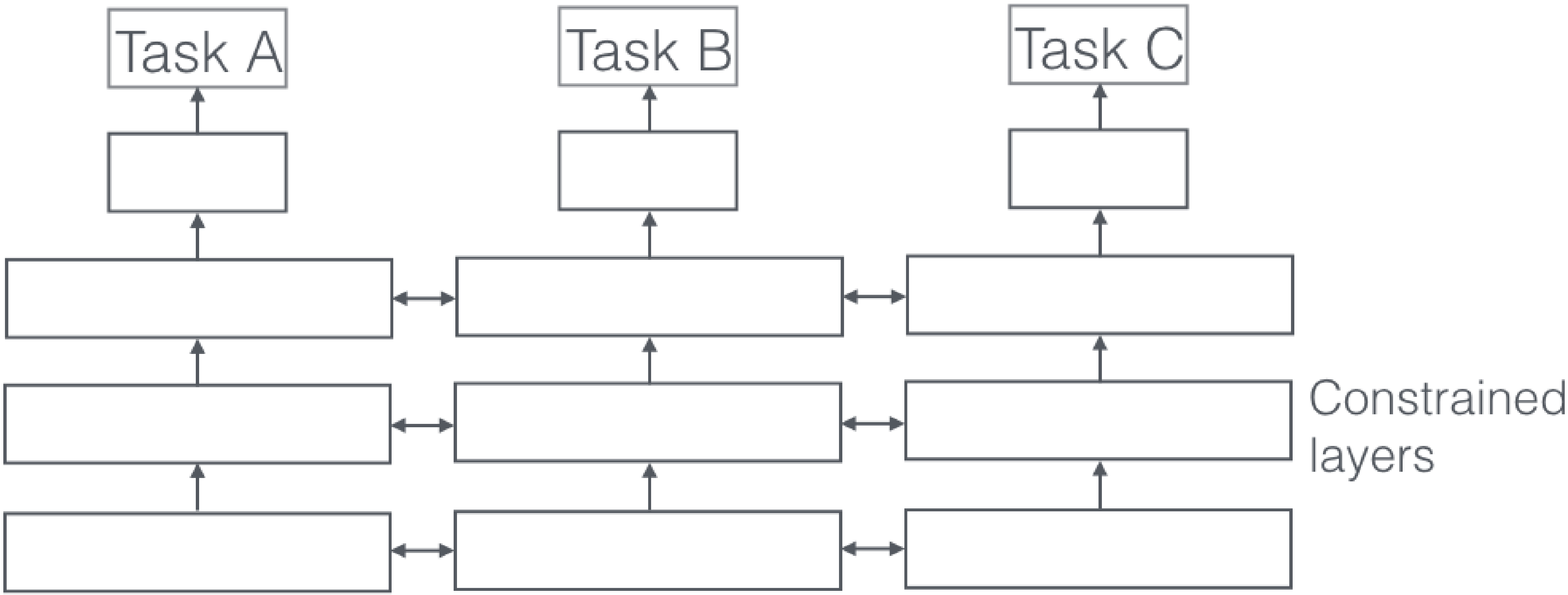
An Overview of Multi-task Learning

Sebastian Ruder



Speaker: Junfeng, Tian

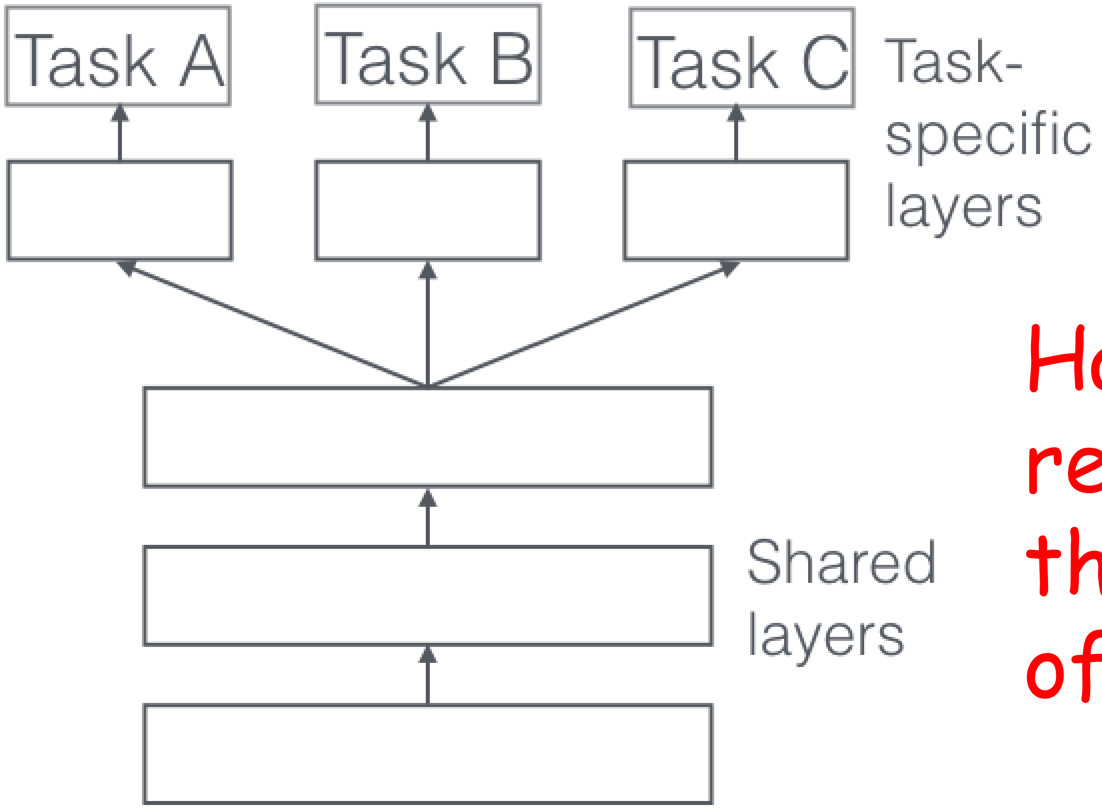
Motivation

- from a biological view:
- from a pedagogical view:
- from a machine learning point of view: bias

Pedagogical： 教学

Two MTL methods for DL

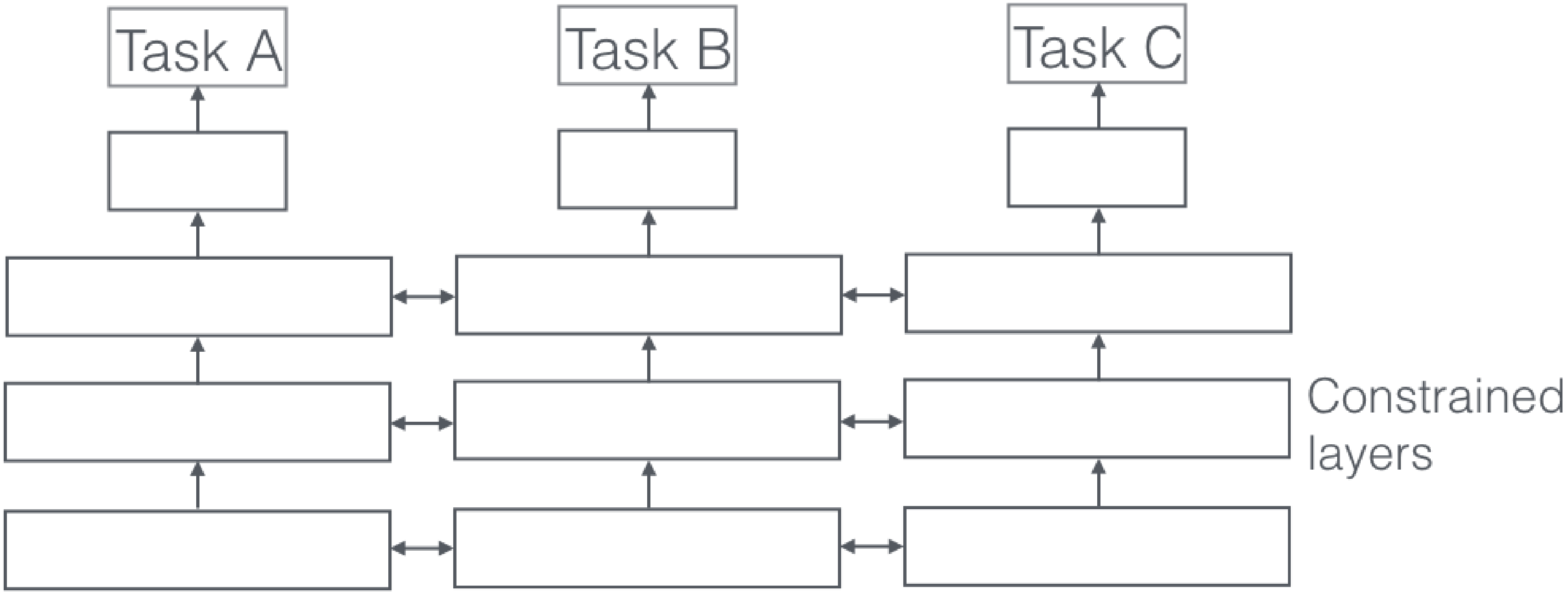
- hard parameter sharing



Hard to find a representations that captures all of the tasks.

Two MTL methods for DL

- soft parameter sharing



Encourage the parameters to be similar, e.g., L2 norm

Artificial auxiliary objectives

- Language modelling
- Adversarial loss
- Predicting what should be there
- Learning the inverse
- Conditioning the initial state
- Predicting data statistics

Language modelling

e.g., CoVe: NIPS17 Learned in Translation: Contextualized Word Vectors

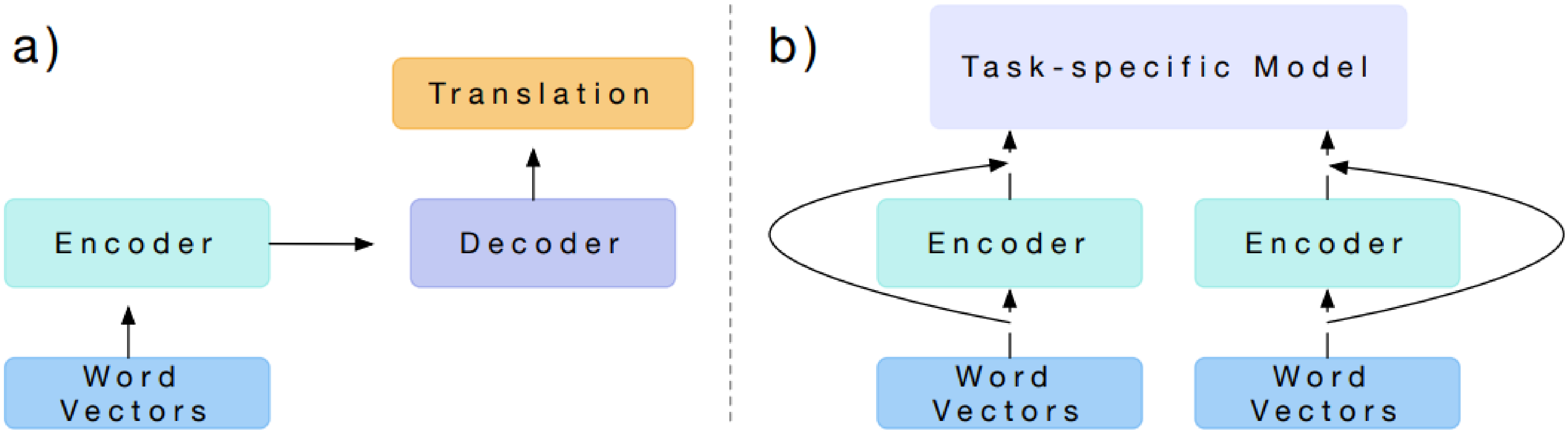


Figure 1: We a) train a two-layer, bidirectional LSTM as the encoder of an attentional sequence-to-sequence model for machine translation and b) use it to provide more context for other NLP models.

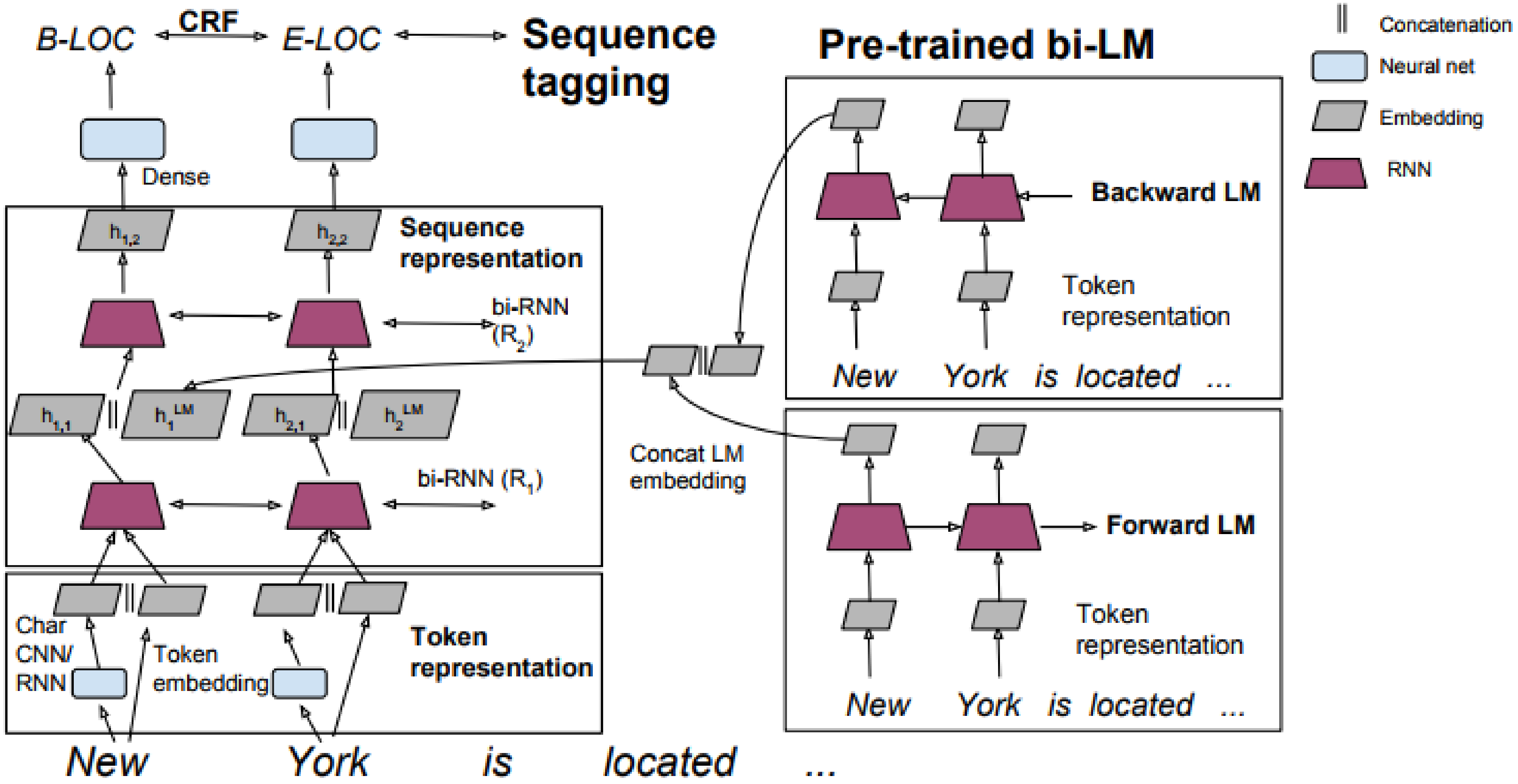
Language modelling

			GloVe+				
Dataset	Random	GloVe	Char	CoVe-S	CoVe-M	CoVe-L	Char+CoVe-L
SST-2	84.2	88.4	90.1	89.0	90.9	91.1	**91.2**
SST-5	48.6	53.5	52.2	54.0	54.7	54.5	**55.2**
IMDb	88.4	91.1	91.3	90.6	91.6	91.7	**92.1**
TREC-6	88.9	94.9	94.7	94.7	95.1	95.8	**95.8**
TREC-50	81.9	89.2	89.8	89.6	89.6	90.5	**91.2**
SNLI	82.3	87.7	87.7	87.3	87.5	87.9	**88.1**
SQuAD	65.4	76.0	78.1	76.5	77.1	79.5	**79.9**

Table 2: CoVe improves validation performance. CoVe has an advantage over character n-gram embeddings, but using both improves performance further. Models benefit most by using an MT-LSTM trained with MT-Large (CoVe-L). Accuracy reported for classification tasks; F1 for SQuAD.

Language modelling

e.g., ACL17 Semi-supervised Model



Language modelling

Model	External resources	F_1 Without	F_1 With	Δ
Yang et al. (2017)	transfer from CoNLL 2000/PTB-POS	91.2	91.26	+0.06
Chiu and Nichols (2016)	with gazetteers	90.91	91.62	+0.71
Collobert et al. (2011)	with gazetteers	88.67	89.59	+0.92
Luo et al. (2015)	joint with entity linking	89.9	91.2	+1.3
Ours	no LM vs TagLM *unlabeled data only*	90.87	**91.93**	+1.06

Table 3: Improvements in test set F_1 in CoNLL 2003 NER when including additional labeled data or task specific gazetteers (except the case of TagLM where we do not use additional labeled resources).

Model	External resources	F_1 Without	F_1 With	Δ
Yang et al. (2017)	transfer from CoNLL 2003/PTB-POS	94.66	95.41	+0.75
Hashimoto et al. (2016)	jointly trained with PTB-POS	95.02	95.77	+0.75
Søgaard and Goldberg (2016)	jointly trained with PTB-POS	95.28	95.56	+0.28
Ours	no LM vs TagLM *unlabeled data only*	95.00	**96.37**	**+1.37**

Table 4: Improvements in test set F_1 in CoNLL 2000 Chunking when including additional labeled data (except the case of TagLM where we do not use additional labeled data).

Language modelling

e.g., ACL17 Semi-supervised Multitask Model

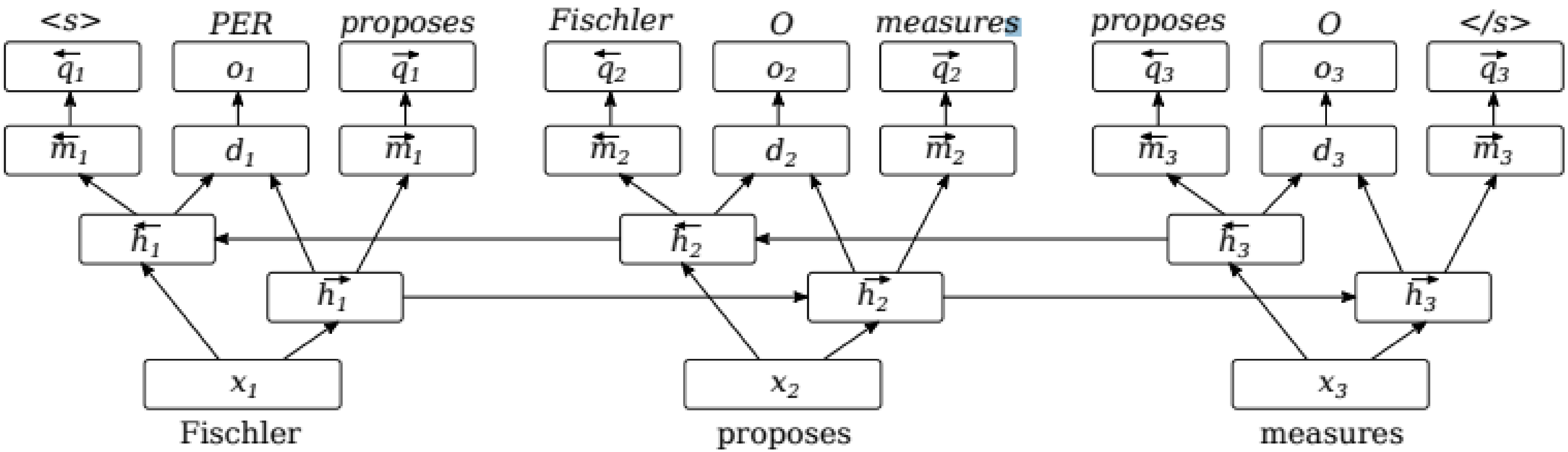


Figure 1: The unfolded network structure for a sequence labeling model with an additional language modeling objective, performing NER on the sentence *"Fischler proposes measures"*. The input tokens are shown at the bottom, the expected output labels are at the top. Arrows above variables indicate the directionality of the component (forward or backward).

Language modelling

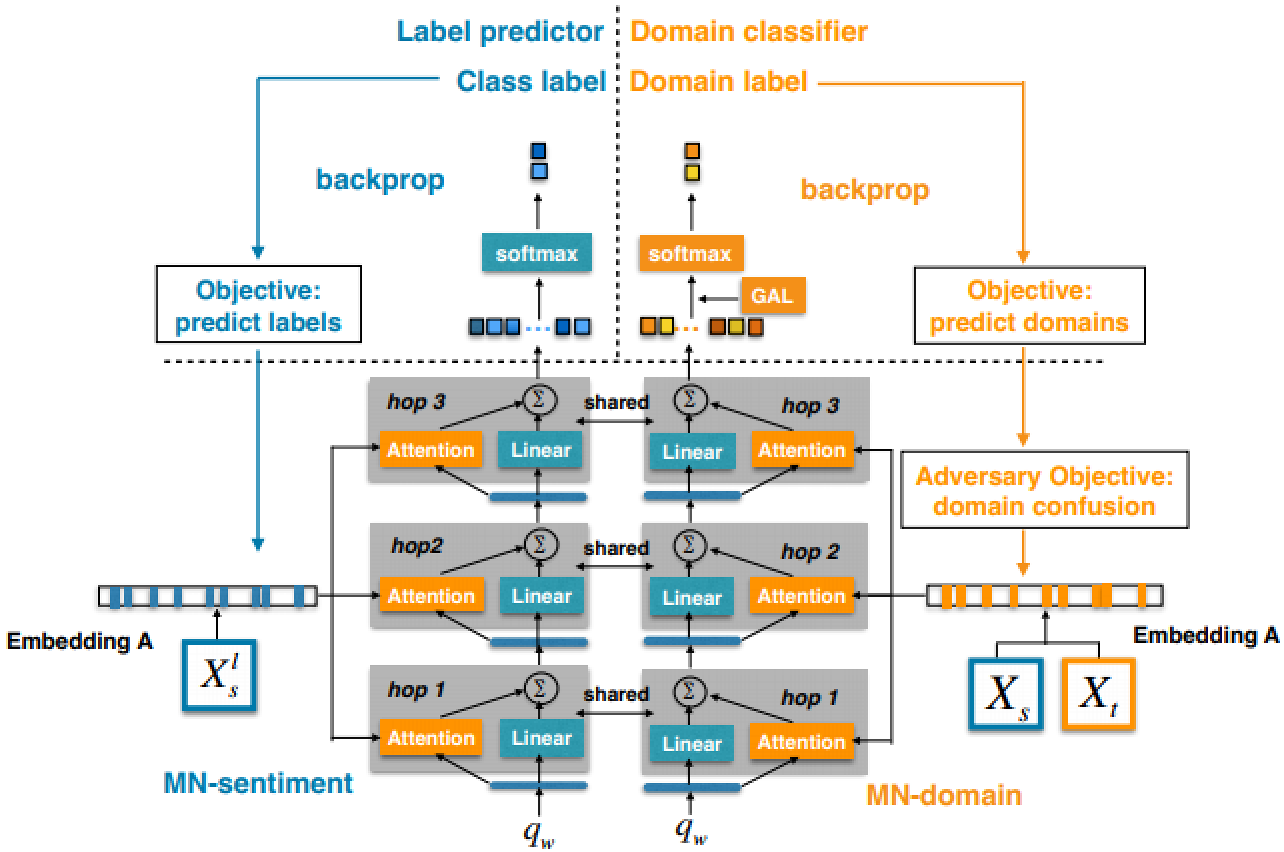
e.g., ACL17 Semi-supervised Multitask Model

	CoNLL-00		CoNLL-03		CHEMDNER		JNLPBA	
	DEV	TEST	DEV	TEST	DEV	TEST	DEV	TEST
Baseline	92.92	92.67	90.85	85.63	83.63	84.51	77.13	72.79
+ dropout	93.40	93.15	91.14	86.00	84.78	85.67	77.61	73.16
+ LMcost	**94.22**	**93.88**	**91.48**	**86.26**	**85.45**	**86.27**	**78.51**	**73.83**

Table 2: Performance of alternative sequence labeling architectures on NER and chunking datasets, measured using CoNLL standard entity-level F_1 score.

Adversarial loss

e.g., domain adaptation: IJCAI17 End-to-End Adversarial Memory Network



Adversarial loss

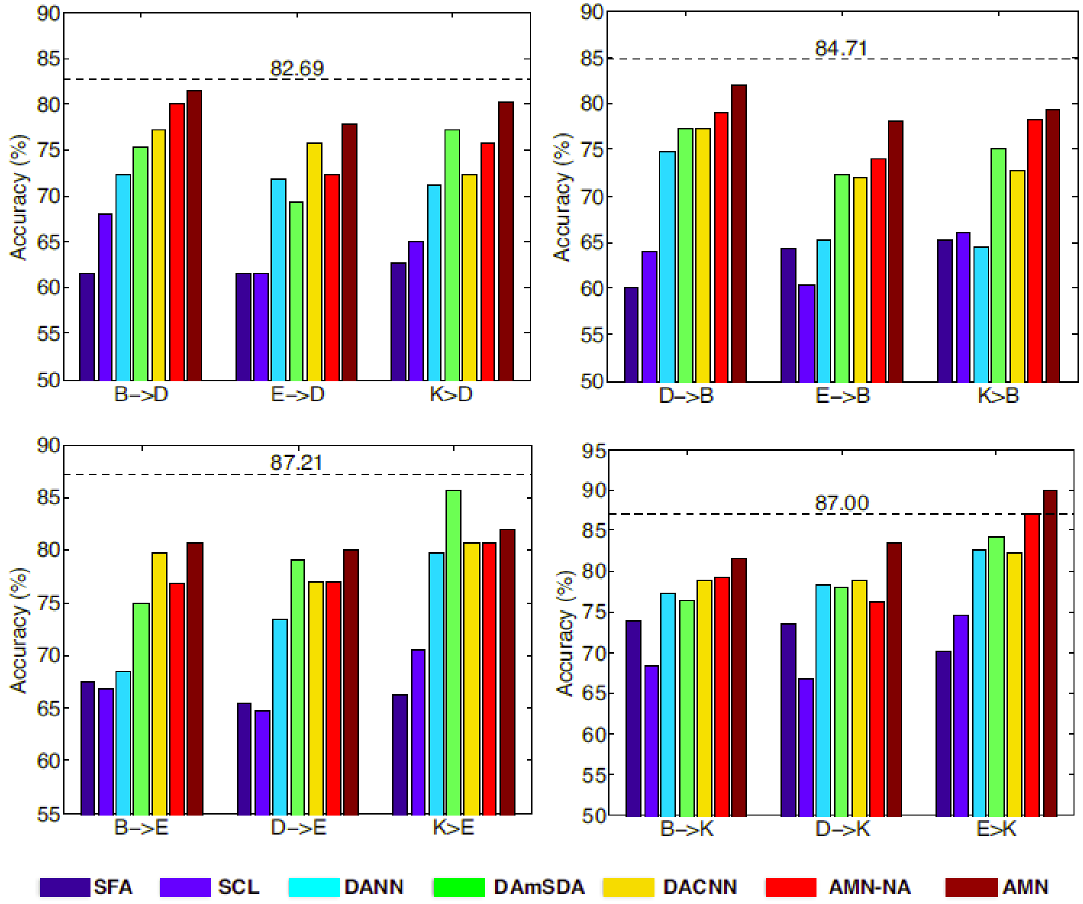
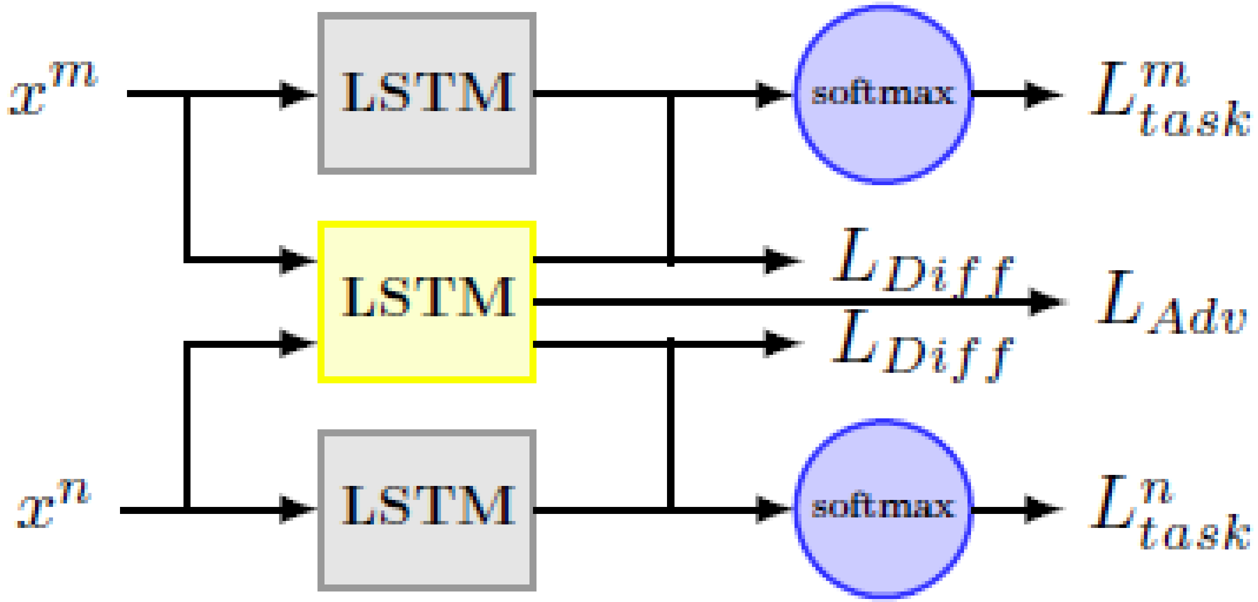


Figure 2: Average results for cross-domain sentiment classification on the Amazon reviews dataset.

Adversarial loss

e.g., learn task-independent representation:
ACL17 – Liu Adversarial Multi-task Learning for Text Classification



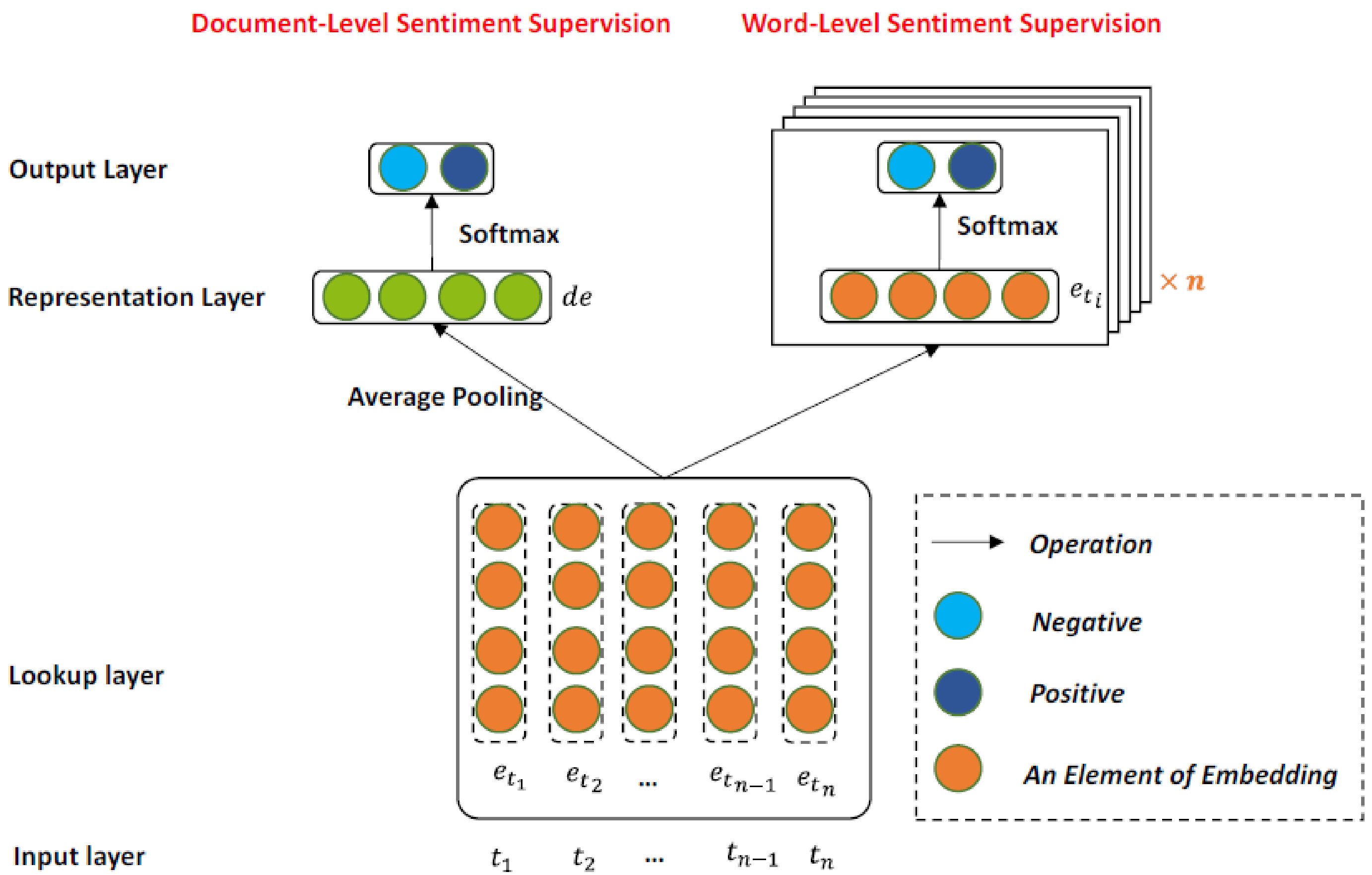
Adversarial Multi-task Learning for Text Classification

Task	Single Task				Multiple Tasks				
	LSTM	BiLSTM	sLSTM	Avg.	MT-DNN	MT-CNN	FS-MTL	SP-MTL	ASP-MTL
Books	20.5	19.0	18.0	19.2	17.8$_{(-1.4)}$	15.5$_{(-3.7)}$	17.5$_{(-1.7)}$	18.8$_{(-0.4)}$	16.0$_{(-3.2)}$
Electronics	19.5	21.5	23.3	21.4	18.3$_{(-3.1)}$	16.8$_{(-4.6)}$	14.3$_{(-7.1)}$	15.3$_{(-6.1)}$	13.2$_{(-8.2)}$
DVD	18.3	19.5	22.0	19.9	15.8$_{(-4.1)}$	16.0$_{(-3.9)}$	16.5$_{(-3.4)}$	16.0$_{(-3.9)}$	14.5$_{(-5.4)}$
Kitchen	22.0	18.8	19.5	20.1	19.3$_{(-0.8)}$	16.8$_{(-3.3)}$	14.0$_{(-6.1)}$	14.8$_{(-5.3)}$	13.8$_{(-6.3)}$
Apparel	16.8	14.0	16.3	15.7	15.0$_{(-0.7)}$	16.3$_{(+0.6)}$	15.5$_{(-0.2)}$	13.5$_{(-2.2)}$	13.0$_{(-2.7)}$
Camera	14.8	14.0	15.0	14.6	13.8$_{(-0.8)}$	14.0$_{(-0.6)}$	13.5$_{(-1.1)}$	12.0$_{(-2.6)}$	10.8$_{(-3.8)}$
Health	15.5	21.3	16.5	17.8	14.3$_{(-3.5)}$	12.8$_{(-5.0)}$	12.0$_{(-5.8)}$	12.8$_{(-5.0)}$	11.8$_{(-6.0)}$
Music	23.3	22.8	23.0	23.0	15.3$_{(-7.7)}$	16.3$_{(-6.7)}$	18.8$_{(-4.2)}$	17.0$_{(-6.0)}$	17.5$_{(-5.5)}$
Toys	16.8	15.3	16.8	16.3	12.3$_{(-4.0)}$	10.8$_{(-5.5)}$	15.5$_{(-0.8)}$	14.8$_{(-1.5)}$	12.0$_{(-4.3)}$
Video	18.5	16.3	16.3	17.0	15.0$_{(-2.0)}$	18.5$_{(+1.5)}$	16.3$_{(-0.7)}$	16.8$_{(-0.2)}$	15.5$_{(-1.5)}$
Baby	15.3	16.5	15.8	15.9	12.0$_{(-3.9)}$	12.3$_{(-3.6)}$	12.0$_{(-3.9)}$	13.3$_{(-2.6)}$	11.8$_{(-4.1)}$
Magazines	10.8	8.5	12.3	10.5	10.5$_{(+0.0)}$	12.3$_{(+1.8)}$	7.5$_{(-3.0)}$	8.0$_{(-2.5)}$	7.8$_{(-2.7)}$
Software	15.3	14.3	14.5	14.7	14.3$_{(-0.4)}$	13.5$_{(-1.2)}$	13.8$_{(-0.9)}$	13.0$_{(-1.7)}$	12.8$_{(-1.9)}$
Sports	18.3	16.0	17.5	17.3	16.8$_{(-0.5)}$	16.0$_{(-1.3)}$	14.5$_{(-2.8)}$	12.8$_{(-4.5)}$	14.3$_{(-3.0)}$
IMDB	18.3	15.0	18.5	17.3	16.8$_{(-0.5)}$	13.8$_{(-3.5)}$	17.5$_{(+0.2)}$	15.3$_{(-2.0)}$	14.5$_{(-2.8)}$
MR	27.3	25.3	28.0	26.9	24.5$_{(-2.4)}$	25.5$_{(-1.4)}$	25.3$_{(-1.6)}$	24.0$_{(-2.9)}$	23.3$_{(-3.6)}$
AVG	18.2	17.4	18.3	18.0	15.7$_{(-2.2)}$	15.5$_{(-2.5)}$	15.3$_{(-2.7)}$	14.9$_{(-3.1)}$	13.9$_{(-4.1)}$

Table 2: Error rates of our models on 16 datasets against typical baselines. The numbers in brackets represent the improvements relative to the average performance (Avg.) of three single task baselines.

Predicting what should be there

e.g., EMNLP17 Sentiment Lexicon Construction



Predicting what should be there

e.g., EMNLP17 Sentiment Lexicon Construction

Lexicon	Semeval2013	Semeval2014	Semeval2015	Semeval2016	Average
Sentiment140	0.7317	0.7271	0.6917	0.6809	0.7079
HIT	0.7181	0.6947	0.6797	0.6928	0.6963
NN	0.7225	0.7115	***0.6970***	0.6887	0.7049
ETSL	0.7104	0.7090	0.6650	0.6862	0.6926
HSSWE	***0.7550***	***0.7424***	0.6921	***0.7097***	***0.7248***

Table 3: Supervised Evaluation for External Comparison (F_1 Score)

Lexicon	Semeval2013	Semeval2014	Semeval2015	Semeval2016	Average
Sentiment140	0.7208	0.7416	0.6935	0.6928	0.7122
HIT	0.7566	0.7922	0.7128	0.7282	0.7474
NN	0.6903	0.7280	0.6507	0.6585	0.6819
ETSL	0.7675	0.8226	0.7505	***0.7365***	0.7693
HSSWE	***0.7734***	***0.8539***	***0.7669***	0.7206	***0.7787***

Table 4: Unsupervised Evaluation for External Comparison (Accuracy)

Predicting what should be there

- predicting whether **certain entities** occur in a sentence might be useful for **relation extraction**;
- predicting whether a headline contains **certain lurid terms** might help for **clickbait detection**;
- predicting whether an **emotion word** occurs in the sentence might benefit **emotion detection**.

In summary, this auxiliary task should be helpful whenever a task includes **certain highly predictive terms or features**.

Why does MTL work?

- Implicit data augmentation
- Attention focusing
- Eavesdropping
- Representation bias
- Regularization

Why does MTL work?

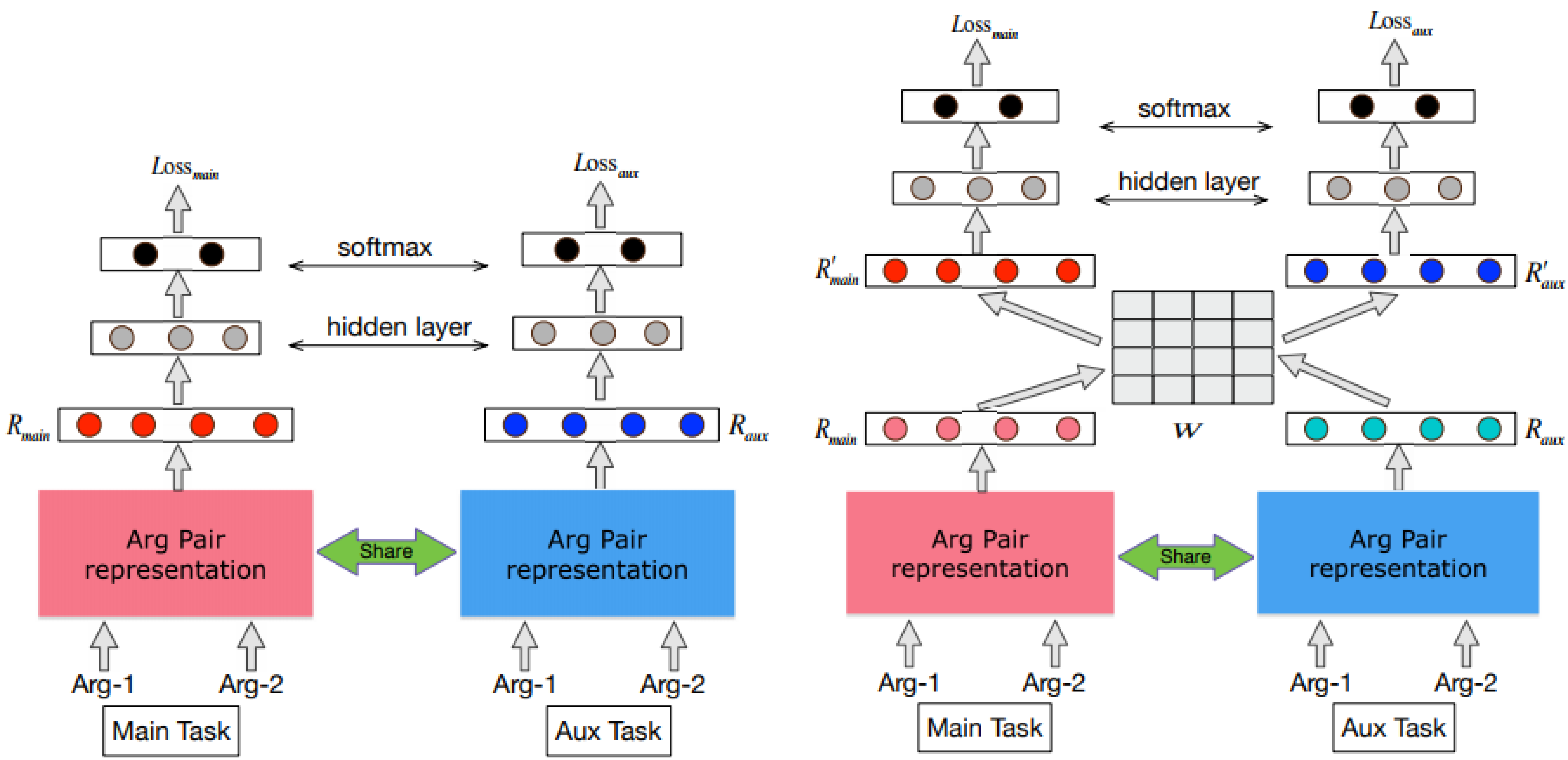
Implicit data augmentation

1. Learning just task A => high risk of **overfitting** in A
2. Learning A and B jointly => better representation F

(since different tasks have different noise patterns)

e.g., Lan et.al 2017 EMNLP

Multi-task Attention-based Neural Networks



Multi-task Attention-based Neural Networks

		Comp.	Cont.	Exp.	Exp+	Temp
STL	LSTM	33.50	52.09	67.51	76.12	27.88
	Bi-LSTM	33.82	52.30	67.47	76.36	29.01
	Attention	**38.15**	**56.07**	**70.53**	**79.80**	**36.72**
Eshare	*Imp + Exp*	35.07	54.62	69.97	79.15	34.57
	Imp + BLLIP	37.67	56.82	70.81	80.43	35.48
Wshare	*Imp + Exp*	37.51 (w=0.1)	55.83 (w=0.2)	70.37 (w=0.3)	80.22(w=0.2)	35.71 (w=0.3)
	Imp + BLLIP	39.13 (w=0.2)	57.78(w=0.2)	71.88(w=0.1)	80.84 (w=0.3)	37.76(w=0.3)
Gshare	*Imp + Exp*	38.91	56.91	71.41	80.02	36.92
	Imp + BLLIP	**40.73**	**58.96**	**72.47**	**81.36**	**38.50**

Table 2: Performance of multiple binary classification on the top level classes in PDTB corpus in terms of F_1 (%).

e.g., ACL15short Low Resource Dependency Parsing

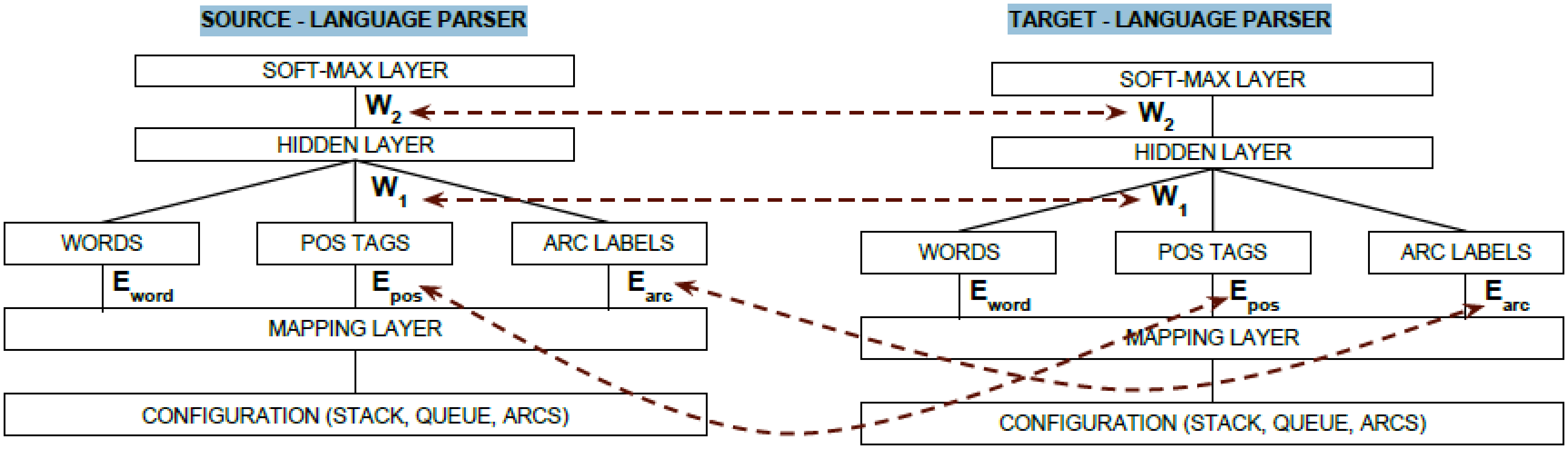


Figure 1: Neural Network Parser Architecture from Chen and Manning (2014) (left). Our model (left and right) with soft parameter sharing between the source and target language shown with dashed lines.

Why does MTL work?

Attention focusing

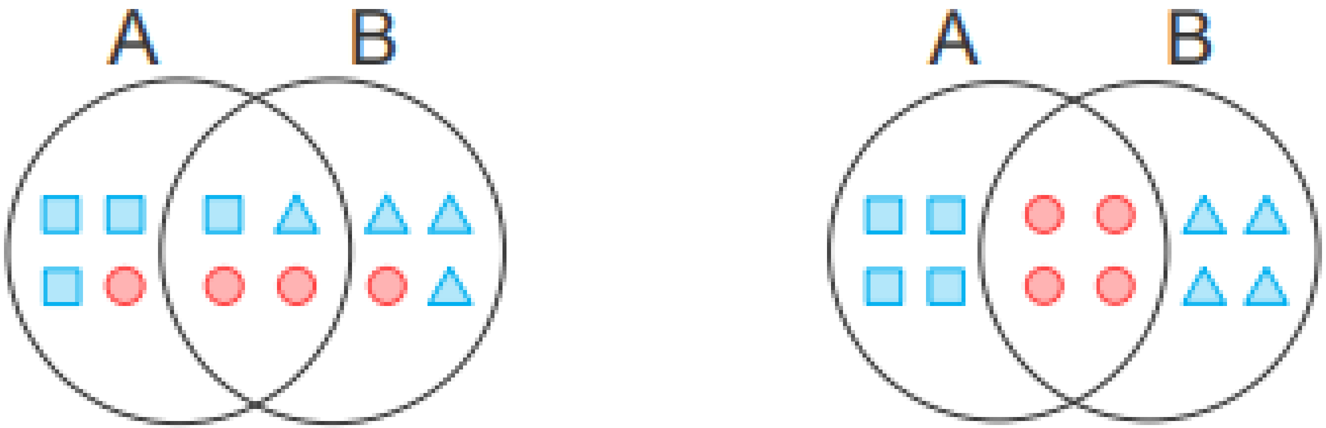
1. it is **difficult** to **differentiate** between **relevant** and **irrelevant** features

2. Auxiliary task provides **additional evidence** for the relevant/irrelevant features

$\Rightarrow$ focus **attention** on those features

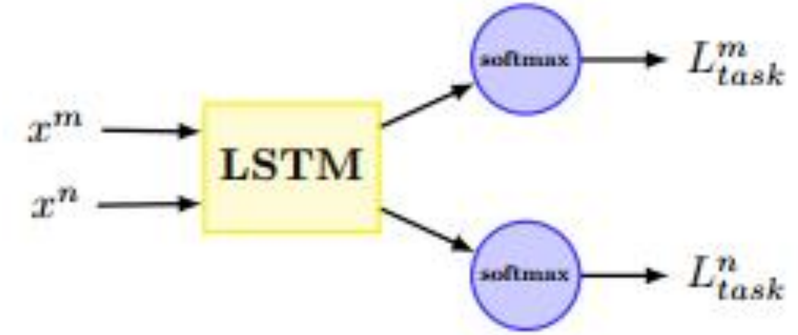
e.g., Liu et. al 2017 ACL

Adversarial Multi-task Learning for Text Classification

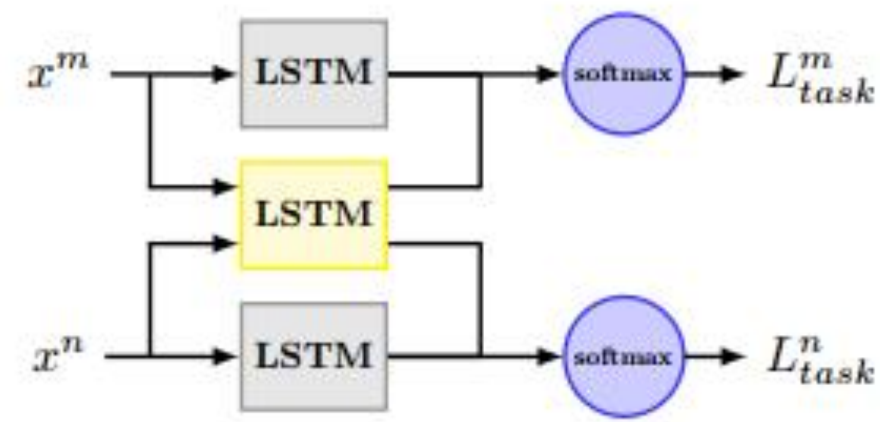


(a) Shared-Private Model (b) Adversarial Shared-Private Model

Adversarial Multi-task Learning for Text Classification



(a) Fully Shared Model (FS-MTL)



(b) Shared-Private Model (SP-MTL)

Adversarial Multi-task Learning for Text Classification

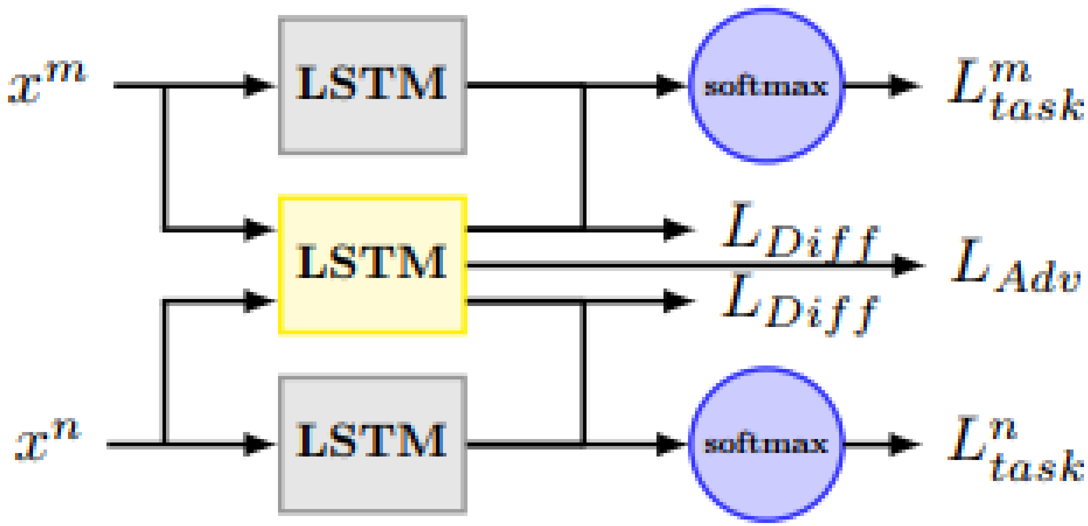


Figure 3: Adversarial shared-private model. Yellow and gray boxes represent shared and private LSTM layers respectively.

Adversarial Multi-task Learning for Text Classification

Task	Single Task				Multiple Tasks				
	LSTM	BiLSTM	sLSTM	Avg.	MT-DNN	MT-CNN	FS-MTL	SP-MTL	ASP-MTL
Books	20.5	19.0	18.0	19.2	17.8$_{(-1.4)}$	15.5$_{(-3.7)}$	17.5$_{(-1.7)}$	18.8$_{(-0.4)}$	16.0$_{(-3.2)}$
Electronics	19.5	21.5	23.3	21.4	18.3$_{(-3.1)}$	16.8$_{(-4.6)}$	14.3$_{(-7.1)}$	15.3$_{(-6.1)}$	13.2$_{(-8.2)}$
DVD	18.3	19.5	22.0	19.9	15.8$_{(-4.1)}$	16.0$_{(-3.9)}$	16.5$_{(-3.4)}$	16.0$_{(-3.9)}$	14.5$_{(-5.4)}$
Kitchen	22.0	18.8	19.5	20.1	19.3$_{(-0.8)}$	16.8$_{(-3.3)}$	14.0$_{(-6.1)}$	14.8$_{(-5.3)}$	13.8$_{(-6.3)}$
Apparel	16.8	14.0	16.3	15.7	15.0$_{(-0.7)}$	16.3$_{(+0.6)}$	15.5$_{(-0.2)}$	13.5$_{(-2.2)}$	13.0$_{(-2.7)}$
Camera	14.8	14.0	15.0	14.6	13.8$_{(-0.8)}$	14.0$_{(-0.6)}$	13.5$_{(-1.1)}$	12.0$_{(-2.6)}$	10.8$_{(-3.8)}$
Health	15.5	21.3	16.5	17.8	14.3$_{(-3.5)}$	12.8$_{(-5.0)}$	12.0$_{(-5.8)}$	12.8$_{(-5.0)}$	11.8$_{(-6.0)}$
Music	23.3	22.8	23.0	23.0	15.3$_{(-7.7)}$	16.3$_{(-6.7)}$	18.8$_{(-4.2)}$	17.0$_{(-6.0)}$	17.5$_{(-5.5)}$
Toys	16.8	15.3	16.8	16.3	12.3$_{(-4.0)}$	10.8$_{(-5.5)}$	15.5$_{(-0.8)}$	14.8$_{(-1.5)}$	12.0$_{(-4.3)}$
Video	18.5	16.3	16.3	17.0	15.0$_{(-2.0)}$	18.5$_{(+1.5)}$	16.3$_{(-0.7)}$	16.8$_{(-0.2)}$	15.5$_{(-1.5)}$
Baby	15.3	16.5	15.8	15.9	12.0$_{(-3.9)}$	12.3$_{(-3.6)}$	12.0$_{(-3.9)}$	13.3$_{(-2.6)}$	11.8$_{(-4.1)}$
Magazines	10.8	8.5	12.3	10.5	10.5$_{(+0.0)}$	12.3$_{(+1.8)}$	7.5$_{(-3.0)}$	8.0$_{(-2.5)}$	7.8$_{(-2.7)}$
Software	15.3	14.3	14.5	14.7	14.3$_{(-0.4)}$	13.5$_{(-1.2)}$	13.8$_{(-0.9)}$	13.0$_{(-1.7)}$	12.8$_{(-1.9)}$
Sports	18.3	16.0	17.5	17.3	16.8$_{(-0.5)}$	16.0$_{(-1.3)}$	14.5$_{(-2.8)}$	12.8$_{(-4.5)}$	14.3$_{(-3.0)}$
IMDB	18.3	15.0	18.5	17.3	16.8$_{(-0.5)}$	13.8$_{(-3.5)}$	17.5$_{(+0.2)}$	15.3$_{(-2.0)}$	14.5$_{(-2.8)}$
MR	27.3	25.3	28.0	26.9	24.5$_{(-2.4)}$	25.5$_{(-1.4)}$	25.3$_{(-1.6)}$	24.0$_{(-2.9)}$	23.3$_{(-3.6)}$
AVG	18.2	17.4	18.3	18.0	15.7$_{(-2.2)}$	15.5$_{(-2.5)}$	15.3$_{(-2.7)}$	14.9$_{(-3.1)}$	13.9$_{(-4.1)}$

Table 2: Error rates of our models on 16 datasets against typical baselines. The numbers in brackets represent the improvements relative to the average performance (Avg.) of three single task baselines.

Adversarial Multi-task Learning for Text Classification

Source Tasks	Single Task				Transfer Models			
	LSTM	BiLSTM	sLSTM	Avg.	SP-MTL-SC	SP-MTL-BC	ASP-MTL-SC	ASP-MTL-BC
ϕ (Books)	20.5	19.0	18.0	19.2	17.8$_{(-1.4)}$	16.3$_{(-2.9)}$	16.8$_{(-2.4)}$	16.3$_{(-2.9)}$
ϕ (Electronics)	19.5	21.5	23.3	21.4	15.3$_{(-6.1)}$	14.8$_{(-6.6)}$	17.8$_{(-3.6)}$	16.8$_{(-4.6)}$
ϕ (DVD)	18.3	19.5	22.0	19.9	14.8$_{(-5.1)}$	15.5$_{(-4.4)}$	14.5$_{(-5.4)}$	14.3$_{(-5.6)}$
ϕ (Kitchen)	22.0	18.8	19.5	20.1	15.0$_{(-5.1)}$	16.3$_{(-3.8)}$	16.3$_{(-3.8)}$	15.0$_{(-5.1)}$
ϕ (Apparel)	16.8	14.0	16.3	15.7	14.8$_{(-0.9)}$	12.0$_{(-3.7)}$	12.5$_{(-3.2)}$	13.8$_{(-1.9)}$
ϕ (Camera)	14.8	14.0	15.0	14.6	13.3$_{(-1.3)}$	12.5$_{(-2.1)}$	11.8$_{(-2.8)}$	10.3$_{(-4.3)}$
ϕ (Health)	15.5	21.3	16.5	17.8	14.5$_{(-3.3)}$	14.3$_{(-3.5)}$	12.3$_{(-5.5)}$	13.5$_{(-4.3)}$
ϕ (Music)	23.3	22.8	23.0	23.0	20.0$_{(-3.0)}$	17.8$_{(-5.2)}$	17.5$_{(-5.5)}$	18.3$_{(-4.7)}$
ϕ (Toys)	16.8	15.3	16.8	16.3	13.8$_{(-2.5)}$	12.5$_{(-3.8)}$	13.0$_{(-3.3)}$	11.8$_{(-4.5)}$
ϕ (Video)	18.5	16.3	16.3	17.0	14.3$_{(-2.7)}$	15.0$_{(-2.0)}$	14.8$_{(-2.2)}$	14.8$_{(-2.2)}$
ϕ (Baby)	15.3	16.5	15.8	15.9	16.5$_{(+0.6)}$	16.8$_{(+0.9)}$	13.5$_{(-2.4)}$	12.0$_{(-3.9)}$
ϕ (Magazines)	10.8	8.5	12.3	10.5	10.5$_{(+0.0)}$	10.3$_{(-0.2)}$	8.8$_{(-1.7)}$	9.5$_{(-1.0)}$
ϕ (Software)	15.3	14.3	14.5	14.7	13.0$_{(-1.7)}$	12.8$_{(-1.9)}$	14.5$_{(-0.2)}$	11.8$_{(-2.9)}$
ϕ (Sports)	18.3	16.0	17.5	17.3	16.3$_{(-1.0)}$	16.3$_{(-1.0)}$	13.3$_{(-4.0)}$	13.5$_{(-3.8)}$
ϕ (IMDB)	18.3	15.0	18.5	17.3	12.8$_{(-4.5)}$	12.8$_{(-4.5)}$	12.5$_{(-4.8)}$	13.3$_{(-4.0)}$
ϕ (MR)	27.3	25.3	28.0	26.9	26.0$_{(-0.9)}$	26.5$_{(-0.4)}$	24.8$_{(-2.1)}$	23.5$_{(-3.4)}$
AVG	18.2	17.4	18.3	18.0	15.6$_{(-2.4)}$	15.2$_{(-2.8)}$	14.7$_{(-3.3)}$	14.3$_{(-3.7)}$

Table 3: Error rates of our models on 16 datasets against vanilla multi-task learning. ϕ (Books) means that we transfer the knowledge of the other 15 tasks to the target task Books.

Why does MTL work?

Eavesdropping

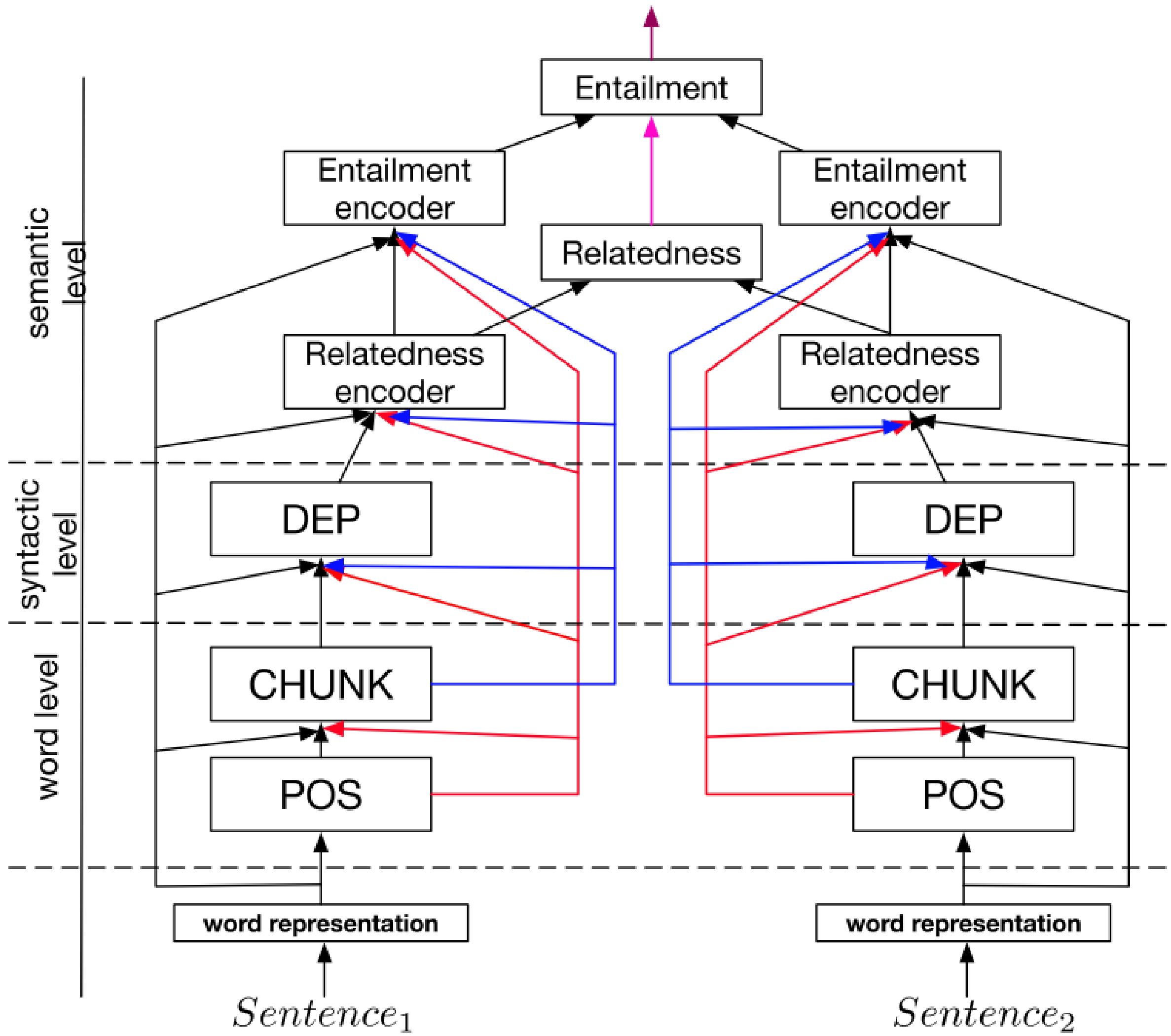
i.e., directly training the model to predict the most important features.

e.g., A Joint Many Task EMNLP17

Reasons:

1. A interacts with the features in a more **complex way**
2. other features are **impeding** the model's ability to learn

A Joint Many-Task Model



A Joint Many-Task Model

		Single	JMT_{all}	JMT_{AB}	JMT_{ABC}	JMT_{DE}	JMT_{CD}	JMT_{CE}
A $\uparrow$	POS	97.45	97.55	97.52	97.54	n/a	n/a	n/a
B $\uparrow$	Chunking	95.02	n/a	95.77	n/a	n/a	n/a	n/a
C $\uparrow$	Dependency UAS	93.35	94.67	n/a	94.71	n/a	93.53	93.57
	Dependency LAS	91.42	92.90	n/a	92.92	n/a	91.62	91.69
D $\downarrow$	Relatedness	0.247	0.233	n/a	n/a	0.238	0.251	n/a
E $\uparrow$	Entailment	81.8	86.2	n/a	n/a	86.8	n/a	82.4

Table 1: Test set results for the five tasks. In the relatedness task, the lower scores are better.

Why does MTL work?

Representation bias

Language Modeling!

MTL biases the model to prefer representations that other tasks also prefer.

e.g., InferSent EMNLP17

Model	dim	NLI dev	NLI test	Transfer micro	Transfer macro
LSTM	2048	81.9	80.7	79.5	78.6
GRU	4096	82.4	81.8	81.7	80.9
BiGRU-last	4096	81.3	80.9	82.9	81.7
BiLSTM-Mean	4096	79.0	78.2	83.1	81.7
Inner-attention	4096	82.3	82.5	82.1	81.0
HConvNet	4096	83.7	83.4	82.0	80.9
BiLSTM-Max	4096	**85.0**	**84.5**	**85.2**	**83.7**

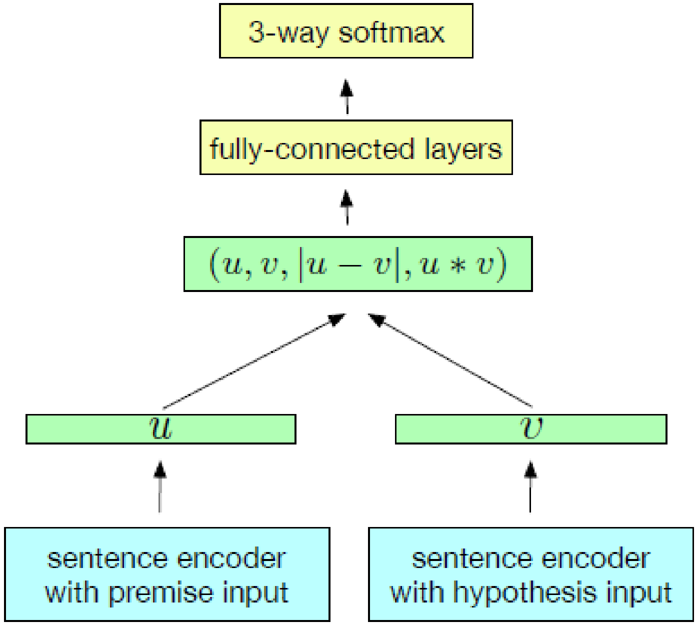


Figure 1: **Generic NLI training scheme.**

e.g., An adversarial joint model for Low-resource

Why does MTL work?

Representation bias

MTL biases the model to prefer representations that other tasks also prefer.

e.g., InferSent EMNLP17

https://github.com/facebookresearch/SentEval

- Binary classification: MR (movie review), CR (product review), SUBJ (subjectivity status), MPQA (opinion-polarity), SST (Stanford sentiment analysis)
- Multi-class classification: TREC (question-type classification), SST (fine-grained Stanford sentiment analysis)
- Entailment (NLI): SNLI (caption-based NLI), MultiNLI (Multi-genre NLI), SICK (Sentences Involving Compositional Knowledge, entailment)
- Semantic Textual Similarity: STS12, STS13 (-SMT), STS14, STS15, STS16
- Semantic Relatedness: STSBenchmark, SICK
- Paraphrase detection: MRPC (Microsoft Research Paraphrase Corpus)
- Caption-Image retrieval: COCO dataset (with ResNet-101 2048d image embeddings)

e.g., An adversarial joint model for Low-resource

Model	MR	CR	SUBJ	MPQA	SST	TREC	MRPC	SICK-R	SICK-E	STS14
Unsupervised representation training (unordered sentences)										
Unigram-TFIDF	73.7	**79.2**	90.3	82.4	-	85.0	73.6/81.7	-	-	.58/.57
ParagraphVec (DBOW)	60.2	66.9	76.3	70.7	-	59.4	72.9/81.1	-	-	.42/.43
SDAE	74.6	78.0	90.8	86.9	-	78.4	**73.7**/80.7	-	-	.37/.38
SIF (GloVe + WR)	-	-	-	-	82.2	-	-	-	**84.6**	**.69**/ -
word2vec BOW†	77.7	79.8	90.9	88.3	79.7	83.6	72.5/81.4	0.803	78.7	.65/.64
fastText BOW†	76.5	78.9	91.6	87.4	78.8	81.8	72.4/81.2	0.800	77.9	.63/.62
GloVe BOW†	**78.7**	78.5	**91.6**	87.6	79.8	83.6	72.1/80.9	0.800	78.6	.54/.56
GloVe Positional Encoding†	78.3	77.4	91.1	87.1	80.6	83.3	72.5/81.2	0.799	77.9	.51/.54
BiLSTM-Max (untrained)†	77.5	**81.3**	89.6	**88.7**	80.7	**85.8**	73.2/81.6	**0.860**	83.4	.39/.48
Unsupervised representation training (ordered sentences)										
FastSent	70.8	78.4	88.7	80.6	-	76.8	72.2/80.3	-	-	**.63/.64**
FastSent+AE	71.8	76.7	88.8	81.5	-	80.4	71.2/79.1	-	-	.62/.62
SkipThought	76.5	80.1	93.6	87.1	82.0	**92.2**	**73.0/82.0**	**0.858**	82.3	.29/.35
SkipThought-LN	**79.4**	**83.1**	**93.7**	**89.3**	82.9	88.4	-	**0.858**	79.5	.44/.45
Supervised representation training										
CaptionRep (bow)	61.9	69.3	77.4	70.8	-	72.2	-	-	-	.46/.42
DictRep (bow)	76.7	78.7	90.7	87.2	-	81.0	68.4/76.8	-	-	**.67/.70**
NMT En-to-Fr	64.7	70.1	84.9	81.5	-	82.8	-	-		.43/.42
Paragram-phrase	-	-	-	-	79.7	-	-	0.849	83.1	**.71**/ -
BiLSTM-Max (on SST)†	(*)	83.7	90.2	89.5	(*)	86.0	72.7/80.9	0.863	83.1	.55/.54
BiLSTM-Max (on SNLI)†	79.9	84.6	92.1	**89.8**	83.3	**88.7**	75.1/82.3	**0.885**	**86.3**	.68/.65
BiLSTM-Max (on AllNLI)†	**81.1**	**86.3**	**92.4**	**90.2**	**84.6**	88.2	**76.2/83.1**	**0.884**	**86.3**	**.70/.67**
Supervised methods (directly trained for each task – no transfer)										
Naive Bayes - SVM	79.4	81.8	93.2	86.3	83.1	-	-	-	-	-
AdaSent	83.1	86.3	95.5	93.3	-	92.4	-	-	-	-
TF-KLD	-	-	-	-	-	-	80.4/85.9	-	-	-
Illinois-LH	-	-	-	-	-	-	-	-	84.5	-
Dependency Tree-LSTM	-	-	-	-	-	-	-	0.868	-	-

Why does MTL work?

Regularization

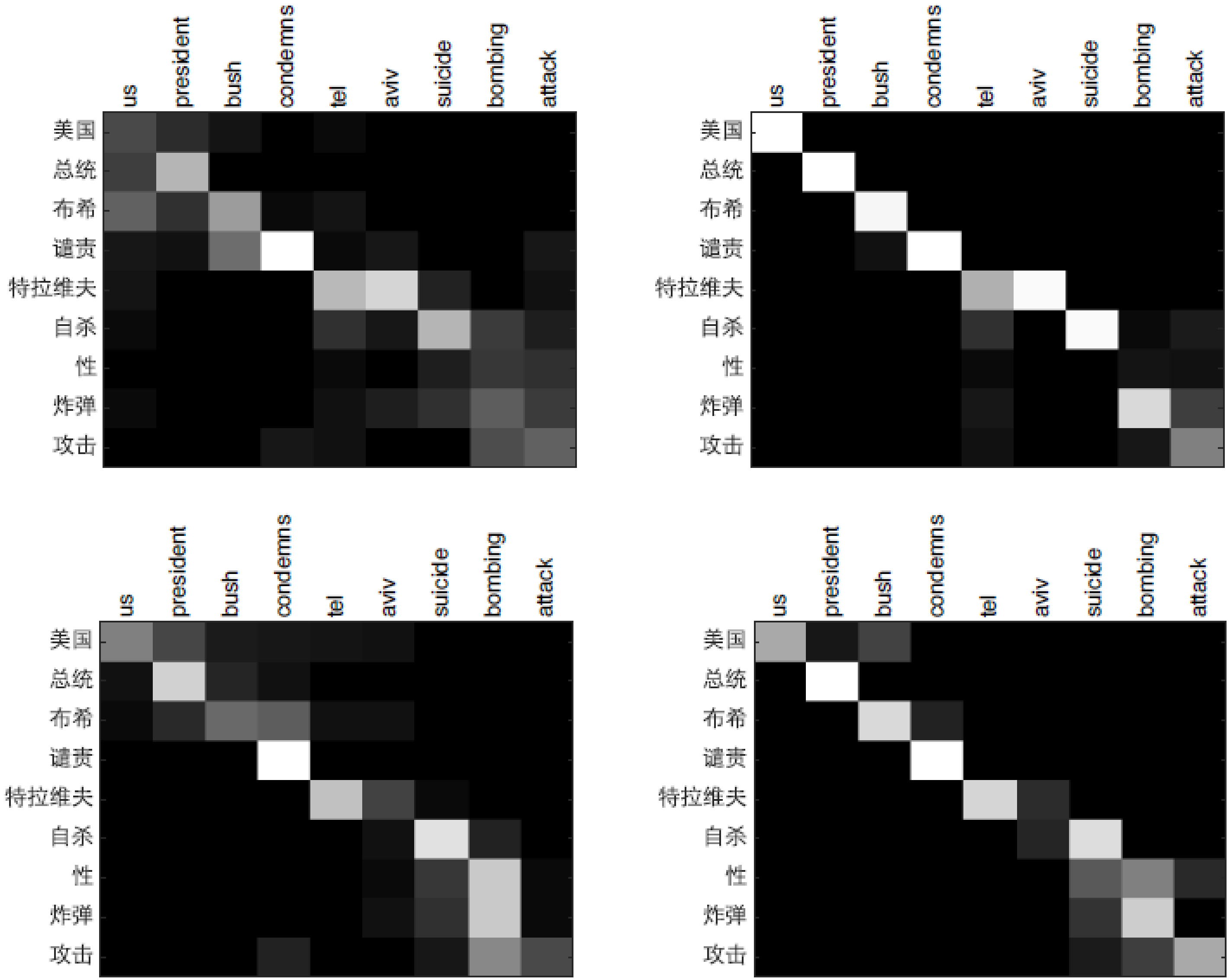
MTL acts as a regularizer by introducing an inductive bias.

Like:

- L1 regularization
- L2 regularization
- Orthogonality: independent representations

e.g., Jin'gang Wang et. al 2018

e.g., IJCAI16 Agreement-based Joint Training



(a) independent training

(b) joint training

$$^{s)}(\overrightarrow{\boldsymbol{\theta}}), \overleftarrow{\mathbf{A}}^{(s)}(\overleftarrow{\boldsymbol{\theta}}))$$

$$\left(\overrightarrow{\mathbf{A}}^{(s)}(\overrightarrow{\boldsymbol{\theta}})_{n,m} + \overleftarrow{\mathbf{A}}^{(s)}(\overleftarrow{\boldsymbol{\theta}})_{m,n}\right)^2$$

Recent interesting research!

- Dual Learning

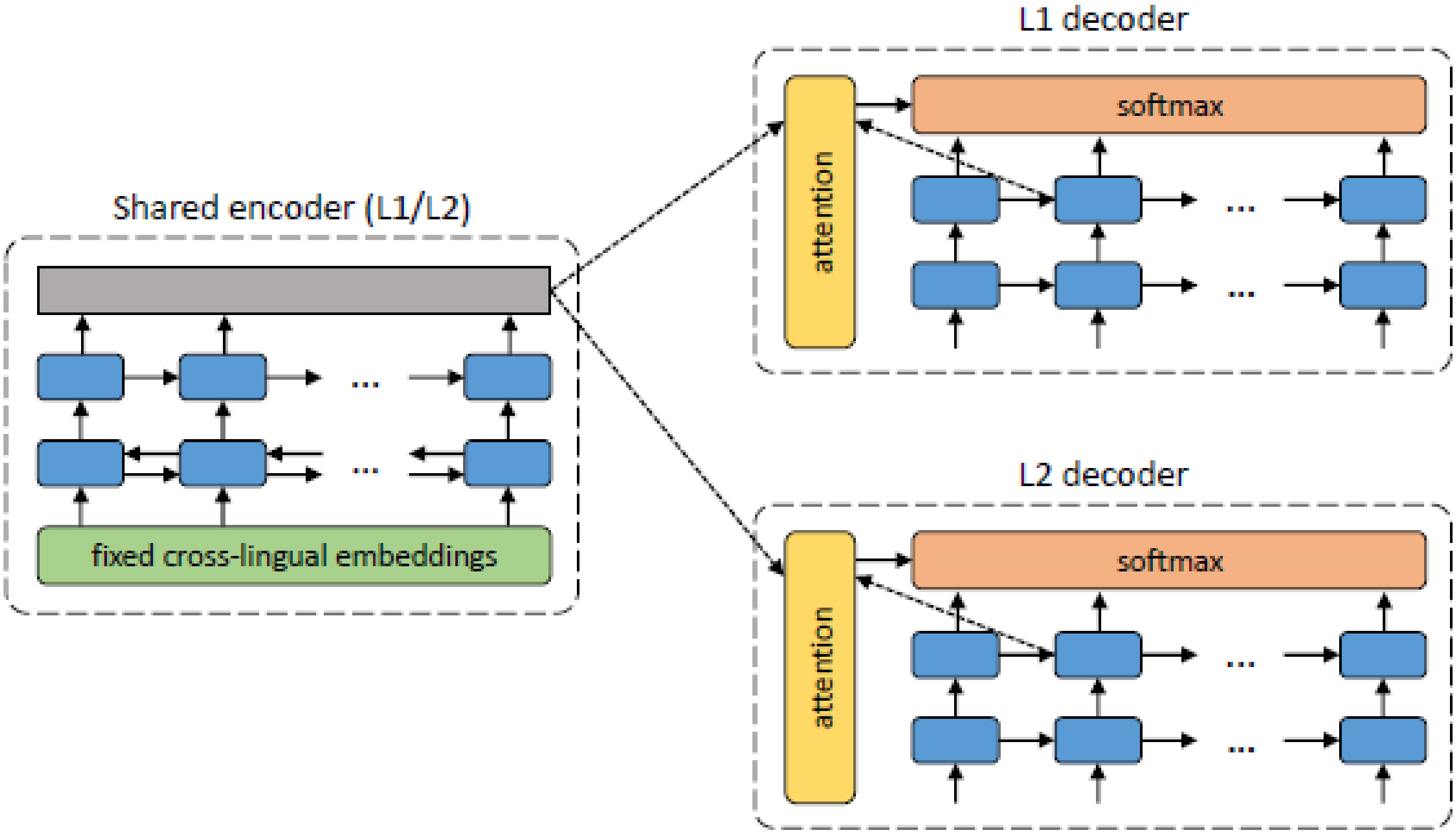
    - MT - Unsupervised Machine Translation

    - QA - Question Generation for Question Answering

- Multilingual

    - A Universal Encoder

Unsupervised Machine Translation



Question Generation for Question Answering

- Select the most relevant answer

$$\widehat{A} = \arg\max_{A} P(A|Q)$$

- Select the most relevant answer

$$\widehat{A} = \arg\max_{A} \{P(A|Q) + \lambda \cdot QQ(Q, Q^{gen}_{max})\}$$

$$QQ(Q, Q^{gen}_{max}) = \arg \max_{i=1,\ldots,10} sim(Q, Q^{gen}_{i}) \cdot p(Q^{gen}_{i})$$

the questions generated from correct answers are more likely to be similar to labeled questions than questions generated from wrong answers.

MULTITASK LEARNING OF MULTILINGUAL SENTENCE REPRESENTATIONS

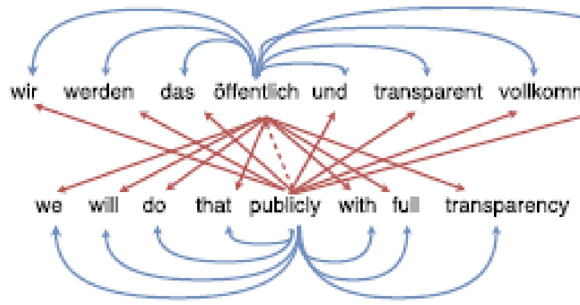
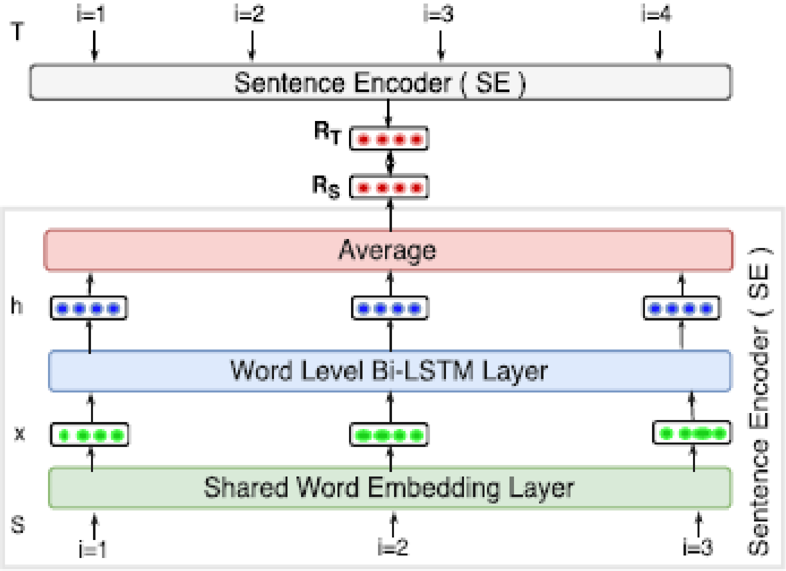


Figure 1: Example context attachments for a bilingual (en-de) skip-gram model.



MULTITASK LEARNING OF MULTILINGUAL SENTENCE REPRESENTATIONS

Model	en → de	de → en
dim=128		
BiCVM-ADD	86.4	74.7
BiCVM-BI	86.1	79.0
BiSkip-UnsupAlign	88.9	77.4
Sent-Avg	88.2	80.0
JMT-Sent-Avg	88.5	80.5
Sent-LSTM	**89.5**	80.4
JMT-Sent-LSTM	89.0	**82.2**
JMT-Sent-Avg*no-mono	88.8	80.3
JMT-Sent-LSTM*no-mono	89.0	81.5
dim=500		
para_doc	**92.7**	**91.5**
BiSkip-UnsupAlign	90.7	80.0
Sent-Avg	91.6	84.8
JMT-Sent-Avg	90.8	83.1
Sent-LSTM	92.0	87.3
JMT-Sent-LSTM	92.3	86.2

What auxiliary tasks are helpful?

What auxiliary tasks are helpful?

Unknown

largely based on the assumption that:

1. the auxiliary task should be related to the main task in some way an
2. it should be helpful for predicting the main task.

What auxiliary tasks are helpful?

Unknown

Like feature-engineering:

- engineering the auxiliary task you optimize.

Why does MTL work?

- Implicit data augmentation
- Attention focusing
- Eavesdropping
- Representation bias
- Regularization

Summary: Auxiliary Task?

- Data is insufficient
- Representations

  - add features: language model / sentence representation

  - task-independent features: agreement / adversarial
- Related task

  - A joint-many task

  - two-steps task

  - predicting what should be there

Summary: How to train?

- Pre-trained / Alternatively train / Combine the loss
- Shared Layer / Constrained Representations (e.g., l2 norm)

Summary

- Low-resource Task
- Two-steps Task
- Multi-lingual Task
- Multi-domain Task
- Dual Task
- Etc.

Reference