Contextual Embeddings: When Are They Worth It? (ACL 2020)

Simran Arora,[*] Avner May,[*] Jian Zhang, Christopher Ré
Stanford Univeristy
{simarora, avnermay, zjian, chrismre}@cs.stanford.edu

Pretrained contextual embeddings:BERT Base(768d)

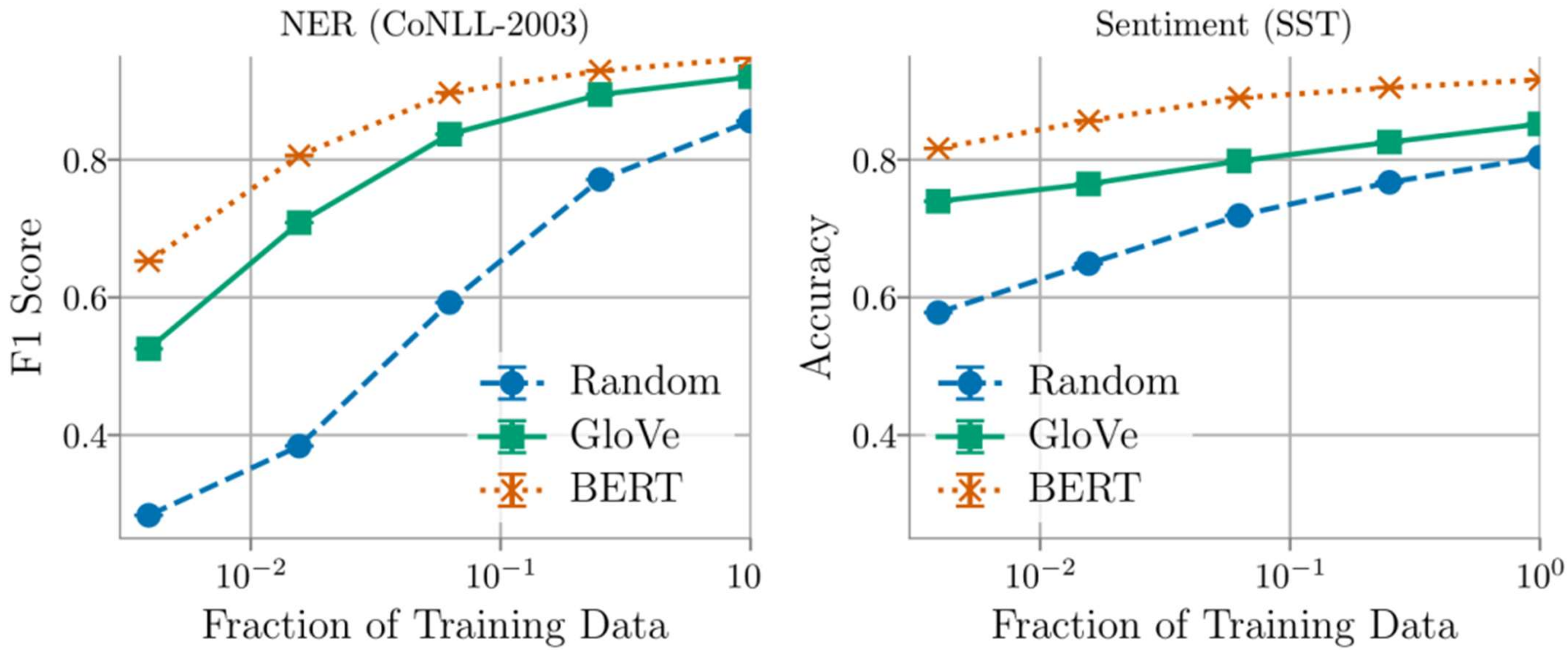
Non-contextual embeddings: GloVe(300d)

Random embeddings(baseline):circulant random matrices(800d)

sentiment analysis、 NER

- **Size of training set**
- **characteristics of language**

Size of training set



characteristics of language

TASK	Complexity	Ambiguity	Unseen
NER(CoNLL)	+4.6	+7.7	+5.0
Sent.(MR)	-5.4	+3.3	+1.2
Sent.(SUBJ)	-1.8	+6.7	+0.9
Sent.(CR)	+0.6	+3.0	+4.1
Sent.(SST)	+7.4	+8.7	+2.3
Sent.(TREC)	+5.1	+5.9	+4.4
Sent.(MPQA)	+7.9	+7.1	+1.3

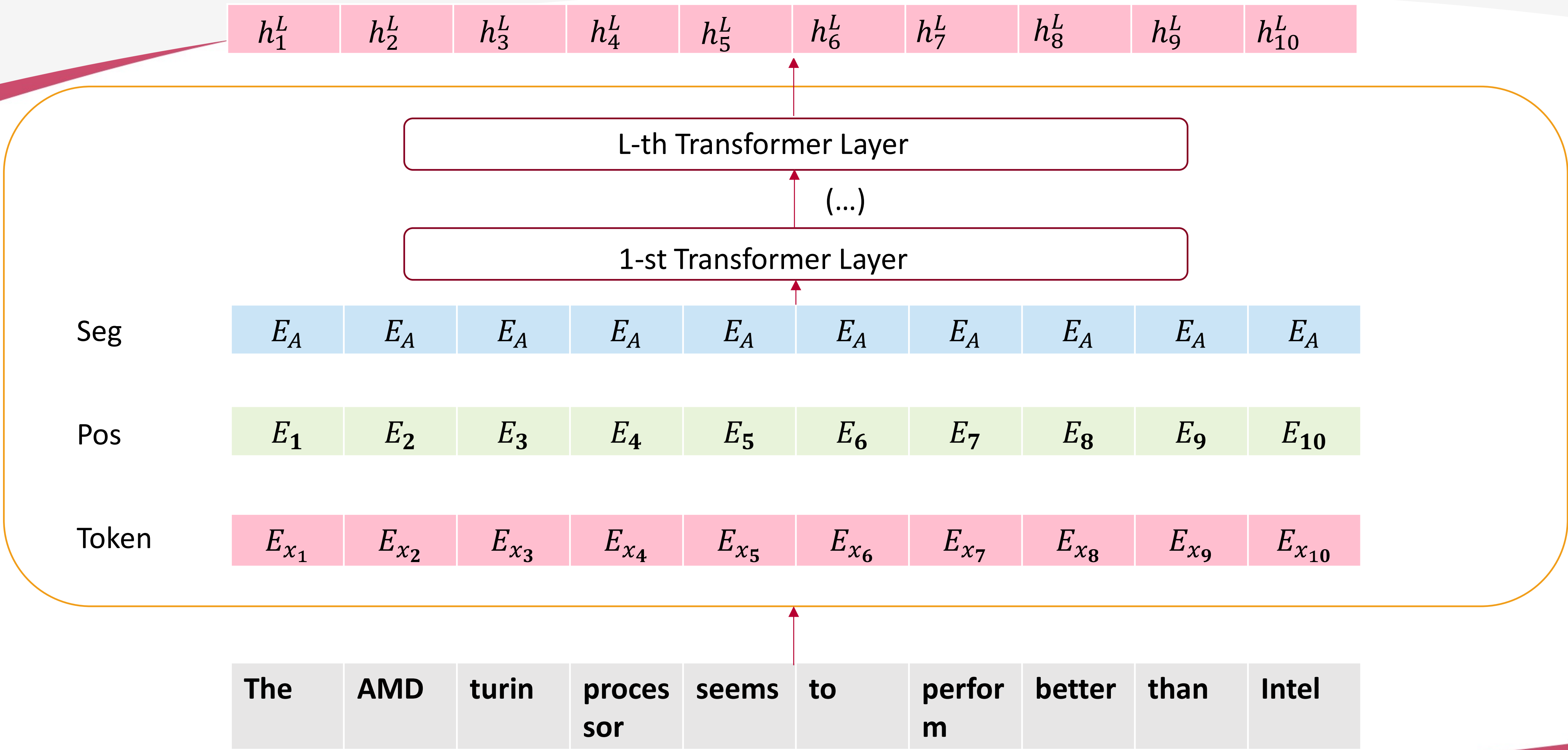
vs random

TASK	Complexity	Ambiguity	Unseen
NER(CoNLL)	+6.7	+5.9	-1.4
Sent.(MR)	-0.6	+6.5	-1.0
Sent.(SUBJ)	-1.8	+4.4	-1.3
Sent.(CR)	+1.2	-2.4	0.0
Sent.(SST)	+7.8	+6.0	-2.8
Sent.(TREC)	+2.2	+8.1	+3.7
Sent.(MPQA)	+6.6	+2.9	+0.4

vs glove

Exploiting BERT for End-to-End Aspect-based Sentiment Analysis* （EMNLP2019）

Xin Li[1], Lidong Bing[2], Wenxuan Zhang[1] and Wai Lam[1]
[1]Department of Systems Engineering and Engineering Management
The Chinese University of Hong Kong, Hong Kong
[2]R&D Center Singapore, Machine Intelligence Technology, Alibaba DAMO Academy
{lixin,wxzhang,wlam}@se.cuhk.edu.hk
l.bing@alibaba-inc.com

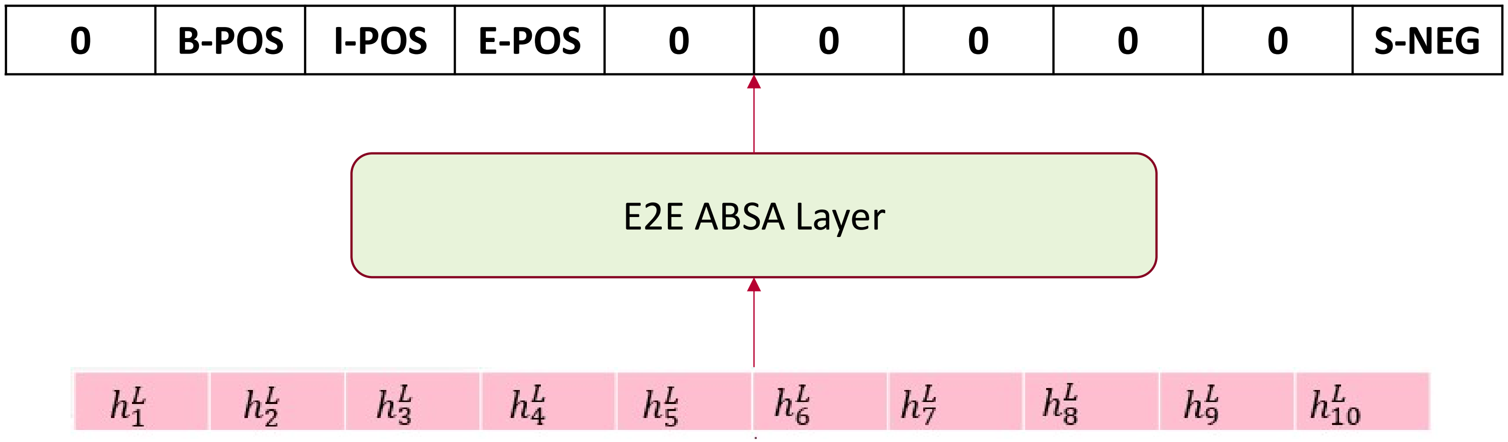


h_1^L h_2^L h_3^L h_4^L h_5^L h_6^L h_7^L h_8^L h_9^L h_{10}^L
L-th Transformer Layer
(...)
1-st Transformer Layer
Seg
E_A E_A E_A E_A E_A E_A E_A E_A E_A E_A
Pos
E_1 E_2 E_3 E_4 E_5 E_6 E_7 E_8 E_9 E_{10}
Token
E_{x_1} E_{x_2} E_{x_3} E_{x_4} E_{x_5} E_{x_6} E_{x_7} E_{x_8} E_{x_9} $E_{x_{10}}$
The AMD turin proces sor seems to perfor m better than Intel

$H^0 = \{e_1, \ldots, e_{\mathrm{T}}\}$

$H^{\mathrm{l}} = Transformer_l(H^{\mathrm{l}-1})$

Use H^{L} for downstream forecasting



E2E ABSA Layer

- Liner
- RNN
- SAN
- CRF

Liner:

$$P(y_t|x_t) = \text{softmax}(W_o h_t^L + b_o)$$

where $W_o \in \mathbb{R}^{\dim_h \times |\mathcal{Y}|}$ is the learnable parameters of the linear layer.

RNN:

$$\begin{bmatrix} r_t \\ z_t \end{bmatrix} = \sigma(\text{LN}(W_x h_t^L) + \text{LN}(W_h h_{t-1}^T))$$

$$n_t = \tanh(\text{LN}(W_{xn} h_t^L) + r_t * \text{LN}(W_{hn} h_{t-1}^T))$$

$$h_t^T = (1 - z_t) * n_t + z_t * h_{t-1}^T$$

$$P(y_t|x_t) = \text{softmax}(W_o h_t^L + b_o)$$

SAN:

$$H^T = \text{LN}(H^L + \text{SLF-ATT}(Q, K, V))$$
$$Q, K, V = H^L W^Q, H^L W^K, H^L W^V$$

CRF:

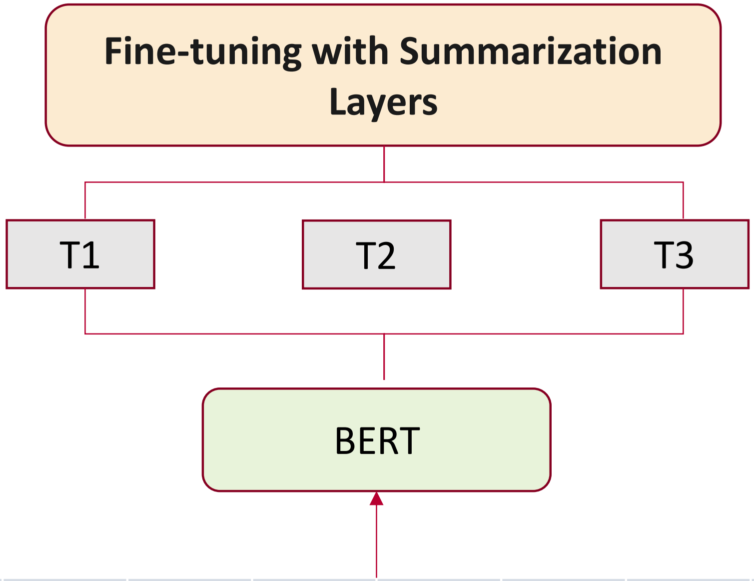
$$s(\mathbf{x}, \mathbf{y}) = \sum_{t=0}^{T} M^A_{y_t, y_{t+1}} + \sum_{t=1}^{T} M^P_{t, y_t}$$
$$p(\mathbf{y}|\mathbf{x}) = \text{softmax}(s(\mathbf{x}, \mathbf{y}))$$
$$\mathbf{y}^* = \arg\max_{\mathbf{y}} s(\mathbf{x}, \mathbf{y})$$

Dataset		Train	Dev	Test	Total
LAPTOP	# sent	2741	304	800	4245
	# aspect	2041	256	634	2931
REST	# sent	3490	387	2158	6035
	# aspect	3893	413	2287	6593

Model		LAPTOP			REST		
		P	R	F1	P	R	F1
Existing Models	(Li et al., 2019a)[♮]	61.27	54.89	57.90	68.64	71.01	69.80
	(Luo et al., 2019)[♮]	-	-	60.35	-	-	72.78
	(He et al., 2019)[♮]	-	-	58.37	-	-	-
LSTM-CRF	(Lample et al., 2016)[♯]	58.61	50.47	54.24	66.10	66.30	66.20
	(Ma and Hovy, 2016)[♯]	58.66	51.26	54.71	61.56	67.26	64.29
	(Liu et al., 2018)[♯]	53.31	59.40	56.19	68.46	64.43	66.38
BERT Models	BERT+Linear	62.16	58.90	60.43	71.42	75.25	73.22
	BERT+GRU	61.88	60.47	**61.12**	70.61	76.20	73.24
	BERT+SAN	62.42	58.71	60.49	72.92	76.72	**74.72**
	BERT+TFM	63.23	58.64	60.80	72.39	76.64	74.41
	BERT+CRF	62.22	59.49	60.78	71.88	76.48	74.06

Fine-tune BERT for Extractive Summarization (ACL 2019)

Yang Liu
Institute for Language, Cognition and Computation
School of Informatics, University of Edinburgh
10 Crichton Street, Edinburgh EH8 9AB
yang.liu2@ed.ac.uk



Seg	E_A	E_A	E_A	E_A	E_B	E_B	E_B	E_B	E_A	E_A	E_A	E_A
Pos	E_1	E_2	E_3	E_4	E_5	E_7	E_8	E_9	E_{10}	E_{12}	E_{13}	E_{14}
Token	E_{cls}	E_{x_1}	E_{x_2}	E_{sep}	E_{cls}	E_{x_3}	E_{x_4}	E_{sep}	E_{cls}	E_{x_5}	E_{x_6}	E_{sep}
Input	[cls]	sent	one	[sep]	[cls]	sent	two	[sep]	[cls]	sent	three	[sep]

Fine-tuning with Summarization Layers $\left\{ \begin{array}{l} \text{Liner} \\ \text{Inter-sentence Transformer} \\ \text{RNN} \end{array} \right\}$

Transformer：

$$\tilde{h}^l = \mathrm{LN}(h^{l-1} + \mathrm{MHAtt}(h^{l-1}))$$
$$h^l = \mathrm{LN}(\tilde{h}^l + \mathrm{FFN}(\tilde{h}^l))$$

$$h^0 = \mathrm{PosEmb}(T)$$

$$\hat{Y}_i = \sigma(W_o h_i^L + b_o)$$

RNN:

$$\begin{pmatrix} F_i \\ I_i \\ O_i \\ G_i \end{pmatrix} = \mathrm{LN}_h(W_h h_{i-1}) + \mathrm{LN}_x(W_x T_i)$$

$$\begin{aligned} C_i = &\sigma(F_i) \odot C_{i-1} \\ &+ \sigma(I_i) \odot \tanh(G_{i-1}) \\ h_i = &\sigma(O_t) \odot \tanh(\mathrm{LN}_c(C_t)) \end{aligned}$$

Each sentence is sorted according to the probability of reservation, and the top-3 is selected

Trigram Block:

Given the selected summary s and candidate sentence C, we will skip C if there is a trigram in C in the selected summary

Model	ROUGE-1	ROUGE-2	ROUGE-L
PGN*	39.53	17.28	37.98
DCA*	41.69	19.47	37.92
LEAD	40.42	17.62	36.67
ORACLE	52.59	31.24	48.87
REFRESH*	41.0	18.8	37.7
NEUSUM*	41.59	19.01	37.98
Transformer	40.90	18.02	37.17
BERTSUM+Classifier	43.23	20.22	39.60
BERTSUM+Transformer	**43.25**	**20.24**	**39.63**
BERTSUM+LSTM	43.22	20.17	39.59