ACL 2019

ACL – general information

2019 - The 57th Annual Meeting of the Association for Computational Linguistics (ACL)

Location: around the world; this time in Florence

Core rank A* (200 MNiSW points)

Best NLP conference in the world

Very hard to get into - acceptance rate in 2019 was 22.7%

Focus is on NLP in various applications

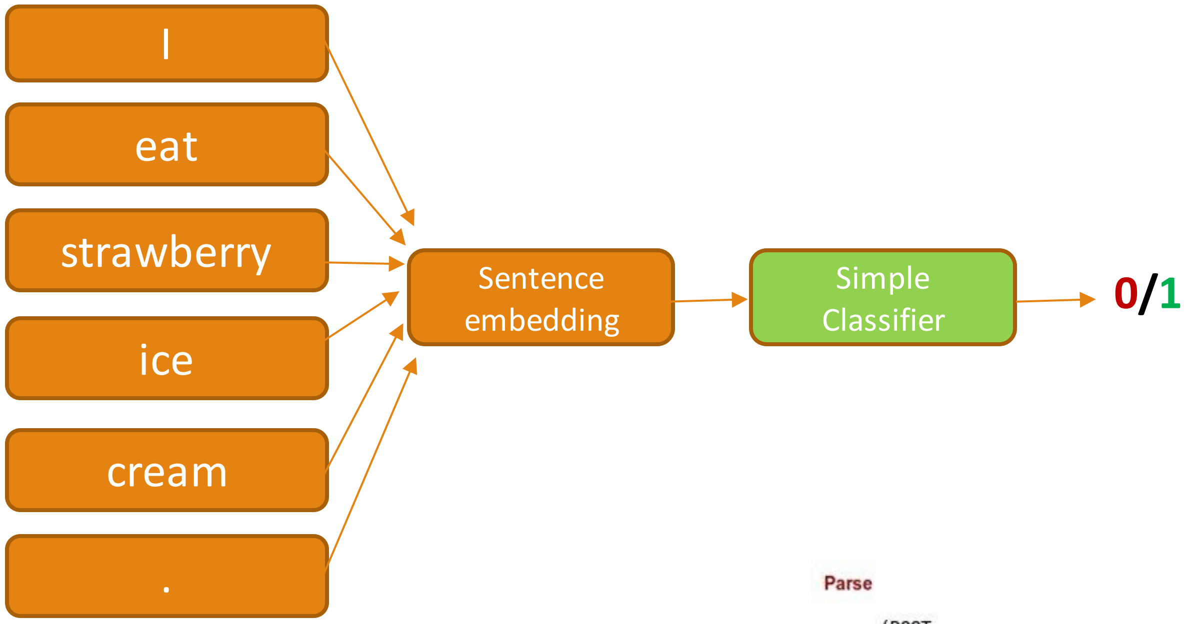
~3000 submissions in 2019

Always a lot of interesting workshops

ACL 2019 word cloud

Probing tasks

Conneau et al. **What you can cram into a single $&!#* vector: Probing sentence embeddings for linguistic properties** (ICLR 2019)



Parse

```
(ROOT
  (S
    (NP (PRP I))
    (VP (VBP eat)
      (NP (JJ strawberry) (NN ice) (NN cream)))
    (. .)))
```

- **Bigram shift** - distinguish intact sentences from sentences where we inverted two random words — *I **strawberry eat** ice cream.*
- **Tree depth** - group sentences by the depth of the longest path from root to any leaf. — *4*
- **Tense** - infer the tense of the main verb — *I <present>eat strawberry ice cream.*
- **Subject number** – infer the number of the subject of the main — *I eat strawberry <single>ice cream.*
- **Odd man out** - recognize sentences with replaced nouns — *I **spoonful** strawberry ice cream.*

Probing tasks

Conneau et al. **What you can cram into a single $&!#* vector: Probing sentence embeddings for linguistic properties** (ICLR 2019)

Task	SentLen	WC	TreeDepth	TopConst	BShift	Tense	SubjNum	ObjNum	SOMO	CoordInv
Baseline representations										
Majority vote	20.0	0.5	17.9	5.0	50.0	50.0	50.0	50.0	50.0	50.0
Hum. Eval.	100	100	84.0	84.0	98.0	85.0	88.0	86.5	81.2	85.0
Length	**100**	0.2	18.1	9.3	50.6	56.5	50.3	50.1	50.2	50.0
NB-uni-tfidf	22.7	**97.8**	24.1	41.9	49.5	77.7	68.9	64.0	38.0	50.5
NB-bi-tfidf	23.0	95.0	24.6	53.0	**63.8**	75.9	69.1	65.4	39.9	**55.7**
BoV-fastText	66.6	91.6	**37.1**	**68.1**	50.8	**89.1**	**82.1**	**79.8**	**54.2**	54.8
BiLSTM-last encoder										
Untrained	36.7	43.8	28.5	76.3	49.8	84.9	84.7	74.7	51.1	64.3
AutoEncoder	**99.3**	23.3	35.6	78.2	62.0	84.3	84.7	82.1	49.9	65.1
NMT En-Fr	83.5	**55.6**	42.4	81.6	62.3	88.1	89.7	89.5	52.0	71.2
NMT En-De	83.8	53.1	42.1	81.8	60.6	88.6	89.3	87.3	51.5	**71.3**
NMT En-Fi	82.4	52.6	40.8	81.3	58.8	88.4	86.8	85.3	52.1	71.0
Seq2Tree	94.0	14.0	**59.6**	**89.4**	**78.6**	**89.9**	**94.4**	**94.7**	49.6	67.8
SkipThought	68.1	35.9	33.5	75.4	60.1	89.1	80.5	77.1	**55.6**	67.7
NLI	75.9	47.3	32.7	70.5	54.5	79.7	79.3	71.3	53.3	66.5
BiLSTM-max encoder										
Untrained	73.3	**88.8**	46.2	71.8	70.6	89.2	85.8	81.9	73.3	68.3
AutoEncoder	**99.1**	17.5	45.5	74.9	71.9	86.4	87.0	83.5	73.4	71.7
NMT En-Fr	80.1	58.3	51.7	81.9	73.7	89.5	90.3	89.1	73.2	75.4
NMT En-De	79.9	56.0	52.3	82.2	72.1	90.5	90.9	89.5	73.4	**76.2**
NMT En-Fi	78.5	58.3	50.9	82.5	71.7	90.0	90.3	88.0	73.2	75.4
Seq2Tree	93.3	10.3	**63.8**	**89.6**	**82.1**	**90.9**	**95.1**	**95.1**	73.2	71.9
SkipThought	66.0	35.7	44.6	72.5	73.8	90.3	85.0	80.6	**73.6**	71.0
NLI	71.7	87.3	41.6	70.5	65.1	86.7	80.7	80.3	62.1	66.8
GatedConvNet encoder										
Untrained	90.3	17.1	30.3	47.5	62.0	78.2	72.2	70.9	61.4	59.6
AutoEncoder	**99.4**	16.8	46.3	75.2	71.9	87.7	88.5	86.5	**73.5**	72.4
NMT En-Fr	84.8	41.3	44.6	77.6	67.9	87.9	88.8	86.6	66.1	72.0
NMT En-De	89.6	49.0	50.5	81.7	72.3	90.4	91.4	89.7	72.8	**75.1**
NMT En-Fi	89.3	**51.5**	49.6	81.8	70.9	90.4	90.9	89.4	72.4	**75.1**
Seq2Tree	96.5	8.7	**62.0**	**88.9**	**83.6**	**91.5**	**94.5**	**94.3**	**73.5**	73.8
SkipThought	79.1	48.4	45.7	79.2	73.4	90.7	86.6	81.7	72.4	72.3
NLI	73.8	29.2	43.2	63.9	70.7	81.3	77.5	74.4	73.3	71.0

Table 2: **Probing task accuracies.** Classification performed by a MLP with sigmoid nonlinearity, taking pre-learned sentence embeddings as input (see Appendix for details and logistic regression results).

Enhancing classification with probing tasks

Vu and Iyyer, Encouraging Paragraph Embeddings to Remember Sentence Identity Improves Classification (ACL 2019)

Model	Yelp	DBPedia	Yahoo
purely supervised w/o external data			
ngrams TFIDF	95.4	98.7	68.5
Large Word ConvNet	95.1	98.3	70.9
Small Word ConvNet	94.5	98.2	70.0
Large Char ConvNet	94.1	98.3	70.5
Small Char ConvNet	93.5	98.0	70.2
SA-LSTM (word level)	NA	98.6	NA
Deep ConvNet	95.7	98.7	73.4
CNN (Zhang et al., 2017)	95.4	98.2	72.6
pre-training + fine-tuning w/o external data			
CNN-R (Zhang et al., 2017)	96.0	98.8	74.2
CNN-SC (ours)	**96.6**	**99.0**	**74.9**

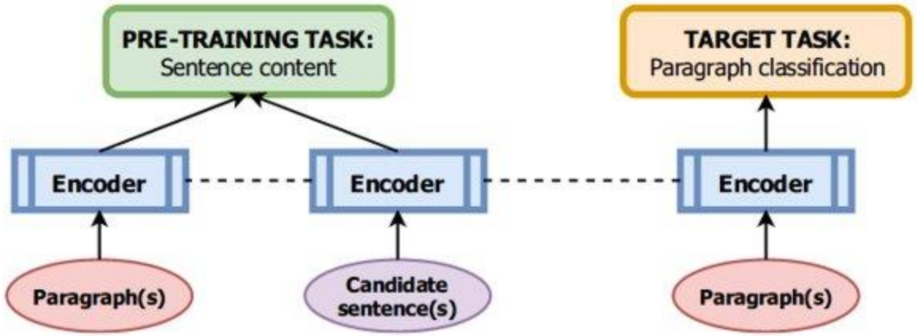


Figure 2: A visualization of our semi-supervised approach. We first train the CNN encoder (shown as two copies with shared parameters) on unlabeled data using our sentence content objective. The encoder is then used for downstream classification tasks.

Polish touch at ACL 2019

Katarzyna Krasnowska-Kieraś and Alina Wróblewska, Empirical Linguistic Study of Sentence Embeddings

(ACL 2019)

Analysis of probing tasks

1. Induced from various embedding methods
2. For 2 languages: En and Pl

	language	measure	FASTTEXT$_{MAX}$	FASTTEXT$_{MEAN}$	BERT$_{MAX}$	BERT$_{MEAN}$	COMBO$_{MAX}$	COMBO$_{MEAN}$	SENT2VEC$_{NS}$	SENT2VEC$_{ORIG}$	LASER	USE
SentLen	**E**	a	52.55	72.27	72.66	82.13	85.03	87.38	71.56	64.76	**85.98**	60.00
	P	a	52.63	67.44	70.79	82.19	84.46	86.31	65.15	—	**86.73**	—
WC	**E**	a	24.44	46.73	35.24	45.53	9.39	11.05	59.96	**79.23**	59.79	43.11
	P	a	19.83	45.84	38.56	43.60	23.04	26.23	**63.85**	—	49.03	—
TreeDepth	**E**	a	29.91	33.00	33.97	38.20	49.08	51.87	33.92	31.03	**39.48**	31.09
	P	a	26.99	30.12	34.43	37.81	44.96	47.35	32.84	—	**40.04**	—
TopDeps	**E**	a	60.49	71.11	78.20	79.33	93.99	93.87	75.77	65.31	**83.33**	63.88
	P	a	65.45	70.67	71.68	75.28	88.16	88.53	73.44	—	**78.84**	—
Passive	**E**	a	84.13	89.47	89.77	92.40	98.48	98.41	88.73	89.04	**92.85**	86.61
	P	a	85.19	91.92	92.16	94.77	98.41	98.71	92.44	—	**95.37**	—
Tense	**E**	a	75.04	84.47	89.32	90.89	96.65	96.64	83.19	85.25	**92.19**	85.64
	P	a	81.56	88.89	93.73	96.09	97.35	97.47	87.36	—	**96.87**	—
SubjNum	**E**	a	73.87	81.43	88.43	90.75	93.19	93.37	82.27	80.88	**94.21**	81.65
	P	a	76.73	87.01	89.89	91.51	94.20	95.03	87.84	—	**93.79**	—
ObjNum	**E**	a	71.75	79.24	85.16	86.89	93.23	94.71	77.23	80.12	**89.33**	79.61
	P	a	69.41	76.05	80.24	**82.64**	90.27	90.31	74.77	—	82.53	—
SentType	**E**	a	96.23	96.20	97.39	97.76	96.85	96.04	97.17	93.76	**97.84**	85.25
	P	a	90.61	96.09	98.36	**98.57**	98.53	98.56	98.09	—	98.39	—
Relatedness	**E**	p	75.71	76.02	74.23	76.54	58.94	59.38	73.43	79.81	84.54	**86.86**
		s	69.35	69.20	68.61	69.54	58.35	58.59	67.97	70.64	79.03	**80.80**
	P	p	76.10	78.06	78.46	83.08	77.40	77.44	76.53	—	**88.09**	—
		s	77.01	79.31	78.91	83.65	77.81	77.98	76.72	—	**89.30**	—
Entailment	**E**	a	76.72	76.86	77.71	77.11	72.82	72.58	78.59	78.26	**83.26**	81.77
	P	a	86.10	87.40	86.70	83.90	84.70	86.10	83.80	—	**87.80**	—

Table 1: Probing and downstream task results. Languages: **P**=Polish, **E**=English, measures: a=accuracy, p=Pearson's r, s=Spearman's ρ. All measures are expressed in %.

Societally engaged NLP

Saeideh Shahrokh Esfahani et al. **Context-specific Language Modeling for Human Trafficking Detection from Online Advertisements (ACL 2019)**

- Crawled ads on adult sites
- Matched phone numbers on ads to known trafficking victims with help from LE
- Obtained ~5000 victim ads and ~5000 'normal' ads

Close your eyes and imagine sliding into a warm flowing river of relaxation as I slowly pull and push your worries away. I want you here with me. Satisfy my need to please you now.

Call Lisa xxx-xxxx-xxxx

(a)

Hi gentlemen,
Meet xxxx beauty Annie, She is 5\'8, very slim, honey blonde hair, gorgeous long legs. Very sexy, friendly and engaging.
Call xxx-xxxx-xxxx to schedule your visit. Xo Xo,
See u soon

(b)

Figure 1: Two examples of online sex ads describing (a) a trafficking victim and (b) a non-trafficked provider, selected from our labeled ads.

Societally engaged NLP

Saeideh Shahrokh Esfahani et al. Context-specific Language Modeling for Human Trafficking Detection from Online Advertisements (ACL 2019)

Interesting differences in ad characteristics:

- T-ads median length was 538 vs Non-T-ads length was 401
- Non-T-ads included 24,000 distinct unigrams and T-ads contained 9,662 distinct unigrams.

3 feartures:

- **LDA**: LDA model assigns a score based on the importance of representation of the words within each topic.
- **Mean Vector**: Mean of FastText embeddings
- **BERT**: document encoding with BERT Base

Algo: logistic regression

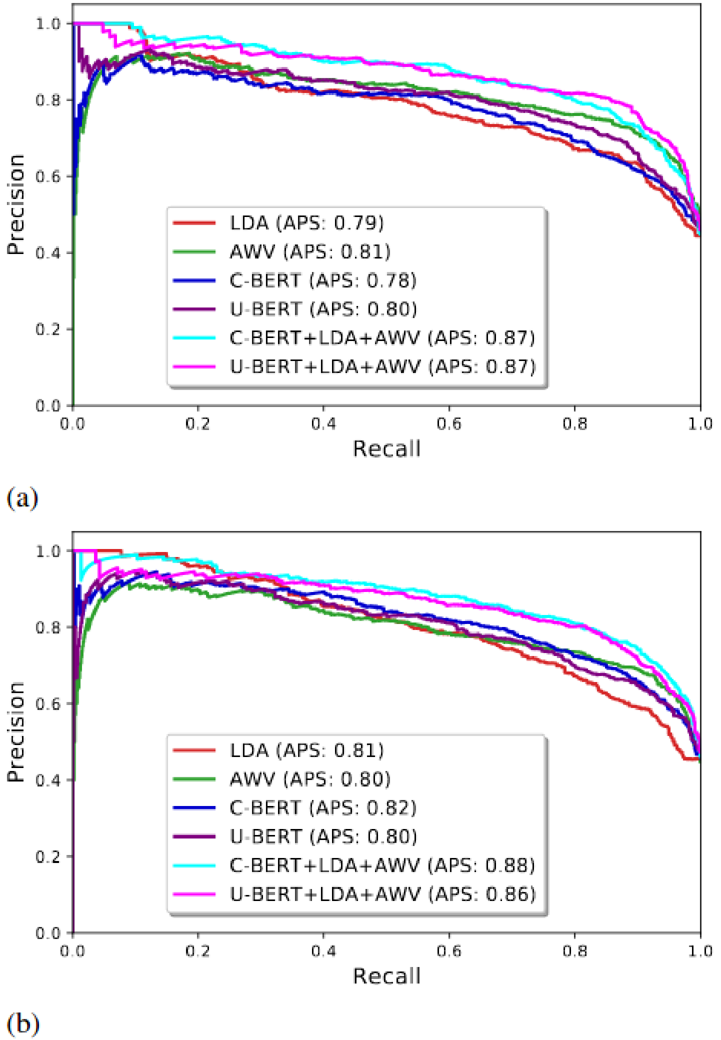


Figure 2: Precision and Recall curves (PRCs) and their corresponding APS values: (a) pure text, (b) text without emojis and punctuation.

Gender Bias Evaluation in NMT

Stanovsky et al. Evaluating Gender Bias in Machine Translation

- Evaluation benchmark for gender bias in NMT
- Correlation with human annotators = 87%
- Procedure:
  - Translate all benchmark examples
  - Align between source and target with *fast_align*
  - Map annotated entity to its translation
  - Figure out gender of the entity using some hardcoded heuristics

	Winogender	WinoBias	WinoMT
Male	240	1582	1826
Female	240	1586	1822
Neutral	240	0	240
Total	**720**	**3168**	**3888**

Table 1: The coreference test sets and resulting WinoMT corpus statistics (in number of instances).

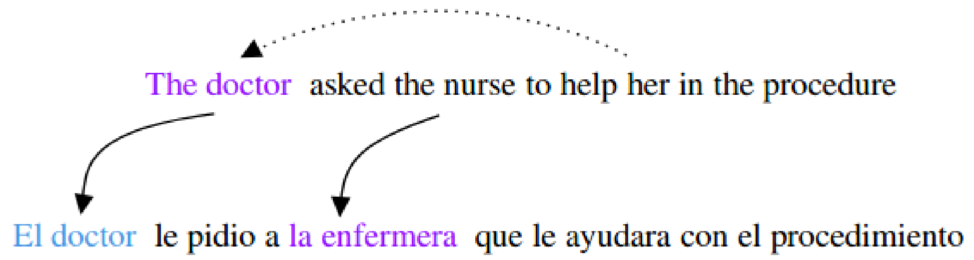


Figure 1: An example of gender bias in machine translation from English (top) to Spanish (bottom). In the English source sentence, the nurse's gender is unknown, while the coreference link with "her" identifies the "doctor" as a female. On the other hand, the Spanish target sentence uses morphological features for gender: "*el* doctor" (male), versus "*la* enfermer-*a*" (female). Aligning between source and target sentences reveals that a stereotypical assignment of gender roles changed the meaning of the translated sentence by changing the doctor's gender.

Gender Bias Evaluation in NMT

Stanovsky et al. Evaluating Gender Bias in Machine Translation

	Google Translate			Microsoft Translator			Amazon Translate*			SYSTRAN		
	Acc	Δ_G	Δ_S	Acc	Δ_G	Δ_S	Acc	Δ_G	Δ_S	Acc	Δ_G	Δ_S
ES	53.1	23.4	21.3	47.3	36.8	23.2	**59.4**	15.4	22.3	45.6	46.3	15.0
FR	**<u>63.6</u>**	6.4	26.7	44.7	36.4	29.7	55.2	17.7	24.9	45.0	44.0	9.4
IT	39.6	32.9	21.5	39.8	39.8	17.0	**42.4**	27.8	18.5	38.9	47.5	9.4
RU	37.7	36.8	11.4	36.8	42.1	8.5	**39.7**	34.7	9.2	37.3	44.1	9.3
UK	38.4	43.6	10.8	**41.3**	46.9	11.8	–	–	–	28.9	22.4	12.9
HE	**53.7**	7.9	37.8	48.1	14.9	32.9	50.5	10.3	47.3	46.6	20.5	24.5
AR	48.5	43.7	16.1	47.3	48.3	13.4	**49.8**	38.5	19.0	47.0	49.4	5.3
DE	59.4	12.5	12.5	**<u>74.1</u>**	0.0	30.2	<u>62.4</u>	12.0	16.7	<u>48.6</u>	34.5	10.3

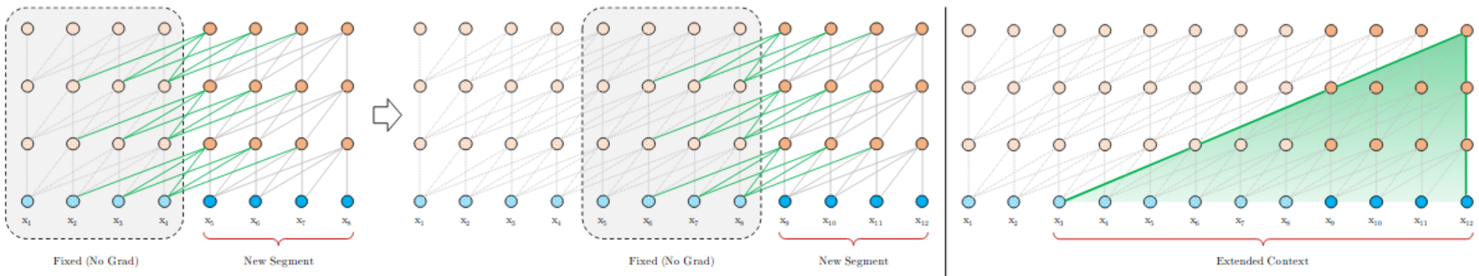
Table 2: Performance of commercial MT systems on the WinoMT corpus on all tested languages, categorized by their family: Spanish, French, Italian, Russian, Ukrainian, Hebrew, Arabic, and German. *Acc* indicates overall gender accuracy (% of instances the translation had the correct gender), Δ_G denotes the difference in performance (F_1 score) between masculine and feminine scores, and Δ_S is the difference in performance (F_1 score) between pro-stereotypical and anti-stereotypical gender role assignments (higher numbers in the two latter metrics indicate stronger biases). Numbers in bold indicate best accuracy for the language across MT systems (row), and underlined numbers indicate best accuracy for the MT system across languages (column). *Amazon Translate does not have a trained model for English to Ukrainian.

Transformer-XL

Dai et al. Transformer-XL: Attentive Language Models Beyond a Fixed-Length Context

Regular Transformer has a window of set length. Transformer-XL enables learning beyond the window restriction.
Some achievements:

- Transformer-XL learns dependency that is 80% longer than RNNs and 450% longer than vanilla Transformers
- Achieves better performance on both short and long sequences
- Is up to 1,800+ times faster than vanilla Transformers during evaluation.



(a) Training phase. (b) Evaluation phase.

Figure 2: Illustration of the Transformer-XL model with a segment length 4.

Transformer-XL

Dai et al. Transformer-XL: Attentive Language Models Beyond a Fixed-Length Context

Nice visual explanation:
https://ai.googleblog.com/2019/01/transformer-xl-unleashing-potential-of.html

Model	#Param	PPL
Grave et al. (2016b) - LSTM	-	48.7
Bai et al. (2018) - TCN	-	45.2
Dauphin et al. (2016) - GCNN-8	-	44.9
Grave et al. (2016b) - LSTM + Neural cache	-	40.8
Dauphin et al. (2016) - GCNN-14	-	37.2
Merity et al. (2018) - QRNN	151M	33.0
Rae et al. (2018) - Hebbian + Cache	-	29.9
Ours - Transformer-XL Standard	151M	**24.0**
Baevski and Auli (2018) - Adaptive Input$^\diamond$	247M	20.5
Ours - Transformer-XL Large	257M	**18.3**

Table 1: Comparison with state-of-the-art results on WikiText-103. $^\diamond$ indicates contemporary work.

Model	#Param	bpc
Ha et al. (2016) - LN HyperNetworks	27M	1.34
Chung et al. (2016) - LN HM-LSTM	35M	1.32
Zilly et al. (2016) - RHN	46M	1.27
Mujika et al. (2017) - FS-LSTM-4	47M	1.25
Krause et al. (2016) - Large mLSTM	46M	1.24
Knol (2017) - cmix v13	-	1.23
Al-Rfou et al. (2018) - 12L Transformer	44M	1.11
Ours - 12L Transformer-XL	41M	**1.06**
Al-Rfou et al. (2018) - 64L Transformer	235M	1.06
Ours - 18L Transformer-XL	88M	1.03
Ours - 24L Transformer-XL	277M	**0.99**

Table 2: Comparison with state-of-the-art results on enwik8.

Model	#Param	bpc
Cooijmans et al. (2016) - BN-LSTM	-	1.36
Chung et al. (2016) - LN HM-LSTM	35M	1.29
Zilly et al. (2016) - RHN	45M	1.27
Krause et al. (2016) - Large mLSTM	45M	1.27
Al-Rfou et al. (2018) - 12L Transformer	44M	1.18
Al-Rfou et al. (2018) - 64L Transformer	235M	1.13
Ours - 24L Transformer-XL	277M	**1.08**

Table 3: Comparison with state-of-the-art results on text8.

Model	#Param	PPL
Shazeer et al. (2014) - Sparse Non-Negative	33B	52.9
Chelba et al. (2013) - RNN-1024 + 9 Gram	20B	51.3
Kuchaiev and Ginsburg (2017) - G-LSTM-2	-	36.0
Dauphin et al. (2016) - GCNN-14 bottleneck	-	31.9
Jozefowicz et al. (2016) - LSTM	1.8B	30.6
Jozefowicz et al. (2016) - LSTM + CNN Input	1.04B	30.0
Shazeer et al. (2017) - Low-Budget MoE	~5B	34.1
Shazeer et al. (2017) - High-Budget MoE	~5B	28.0
Shazeer et al. (2018) - Mesh Tensorflow	4.9B	24.0
Baevski and Auli (2018) - Adaptive Input$^\diamond$	0.46B	24.1
Baevski and Auli (2018) - Adaptive Input$^\diamond$	1.0B	23.7
Ours - Transformer-XL Base	0.46B	23.5
Ours - Transformer-XL Large	0.8B	**21.8**

Table 4: Comparison with state-of-the-art results on One Billion Word. $^\diamond$ indicates contemporary work.

ICONIP 2019

ICONIP – general information

2019 - 26th *International Conference on Neural Information Processing* of the Asia-Pacific Neural Network Society

Location: always somewhere in Asia/Pacific, this time in Sydney

Core rank A (140 MNiSW points)

Rather good on the *difficulty vs gain* scale (considerably easy to get into, considerably high rank)

But, it's not a specialist and highly revered conference, at least in NLP domain.

Conference Topics

Proceedings are open

- https://link.springer.com/conference/iconip
- Our paper is in Vol 3

Conference Tracks:

- **Text Computing using Neural Techniques**
- Spiking Neuron and Related Models
- Adversarial Networks and Learning
- Semantic and Graph Based Approaches
- Convolutional Neural Networks
- Time-series and Related Models
- Image Processing by Neural Techniques
- Model Compression and Optimization
- ….

UJ at ICONIP

Set aggregation network as a trainable pooling layer
Łukasz Maziarka, Marek Śmieja, Aleksandra Nowak , Jacek Tabor, Łukasz Struski, and Przemysław Spurek

-We introduce a Set Aggregation Network (SAN) as an alternative global pooling layer.
-In contrast to typical pooling operators, SAN allows to embed a given set of features to a vector representation of arbitrary size.
-By adjusting the size of embedding, SAN is capable of preserving the whole information from the input.
-It leads to the improvement of classification accuracy. Moreover, it is less prone to overfitting and can be used as a regularizer

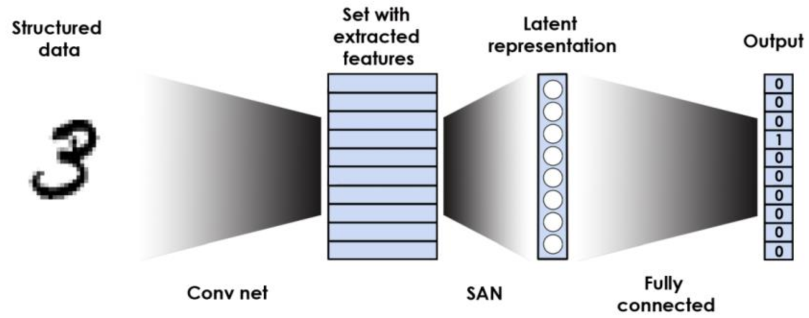


Fig. 1: SAN is an intermediate network which is responsible for learning a vector representation using a set of features extracted from of structured data.

UJ at ICONIP

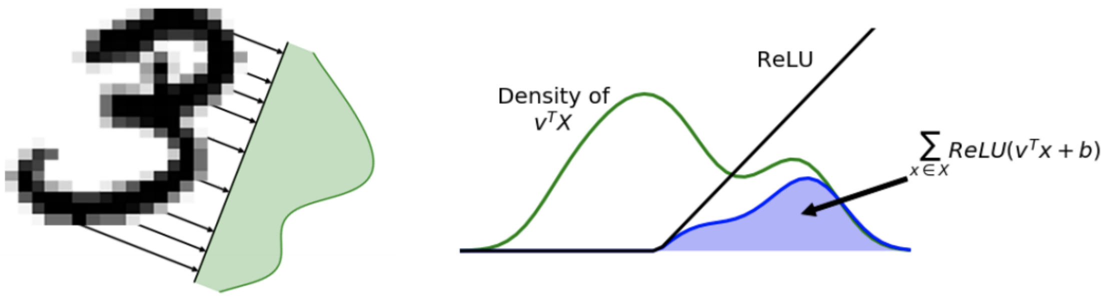


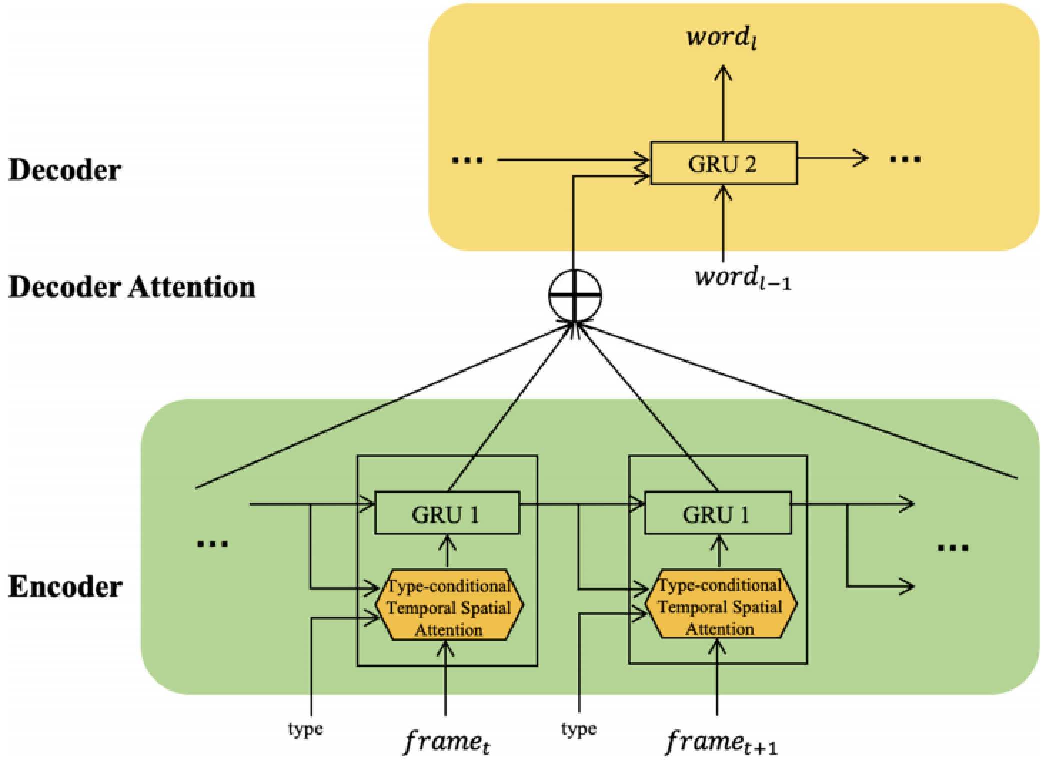
Fig. 2: The idea of our approach is to aggregate information from projections of a set onto several one-dimensional subspaces (left). Next non-linear activation function is applied to every set element and the results are aggregated (right).

Watch and ask – video question generation

Shenglei Huang, Shaohan, and Bencheng Yan

-Question generation (QG) has never been studied in video.
-We adopt the encoder-decoder based framework to deal with this task.
-We involve question type to guide the generation process.
-Specifically, a novel type conditional temporal-spatial attention is proposed, which could capture required information of different types from video content at different time steps.
-We are the first to apply the end-to-end model on video question generation.

Watch and ask – video question generation



Zero shot transfer learning based on visual and textual resemblance

-Existing image search engines, whose ranking functions are built based on labeled images or wrap texts, have poor results on queries in new, or low-frequency keywords.
-In this paper, we put forward the zero-shot transfer learning (ZSTL), which aims to transfer networks from given classifiers to new zero-shot classifiers with little cost, and helps image searching perform better on new or low-frequency words.

Zero shot transfer learning based on visual and textual resemblance

-The target of zero-shot transfer learning is to build a classifier containing few or no training data through applying another known similar classifier (e.g. building a classifier of tiger based on a classifier of cat).
-To meet this aim, we convert the known image classifier into a textual feature extractor, and transform its output space into somewhere near target labels' textual space.

We make an assumption that the structure of source and the target labels are similar in natural language, which means it is possible that their semantic feature space share the similar distribution with the textual feature space through non-linear transformation.

Zero shot transfer learning based on visual and textual resemblance