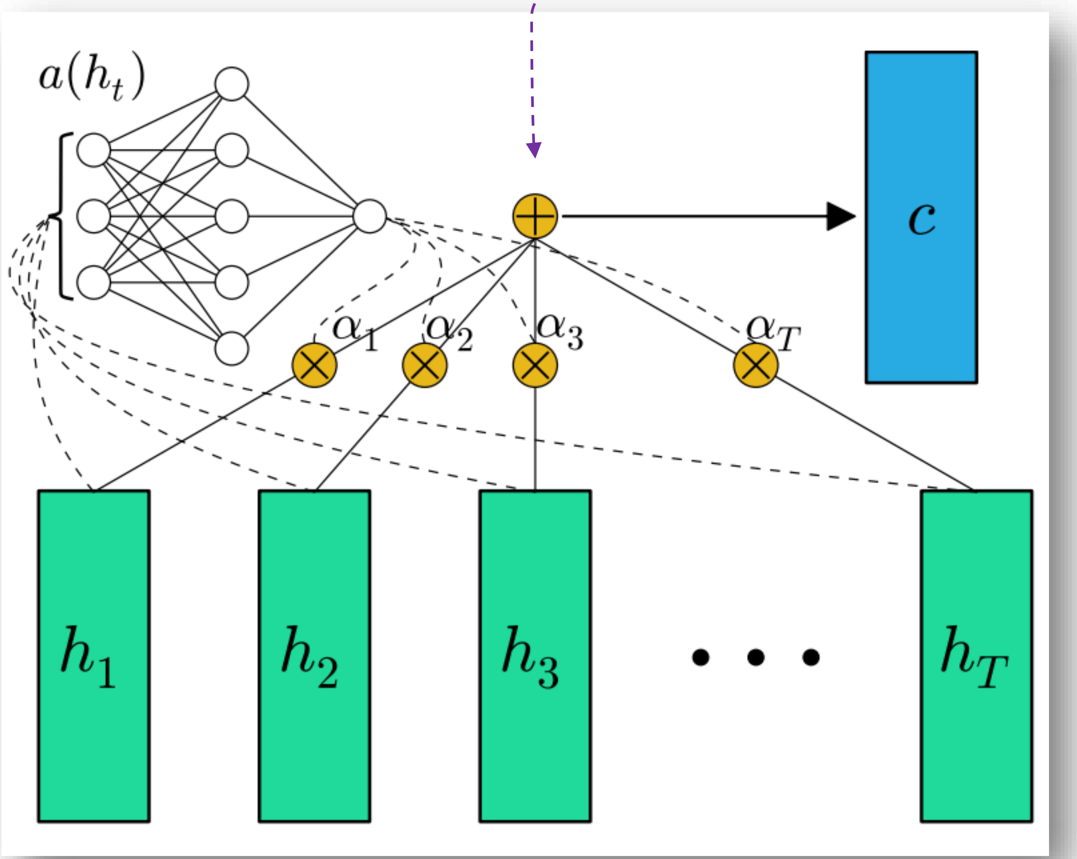
Transformer and BERT

W 4705 – Spring 2020

Yassine Benajiba

Attention refresher

Classification

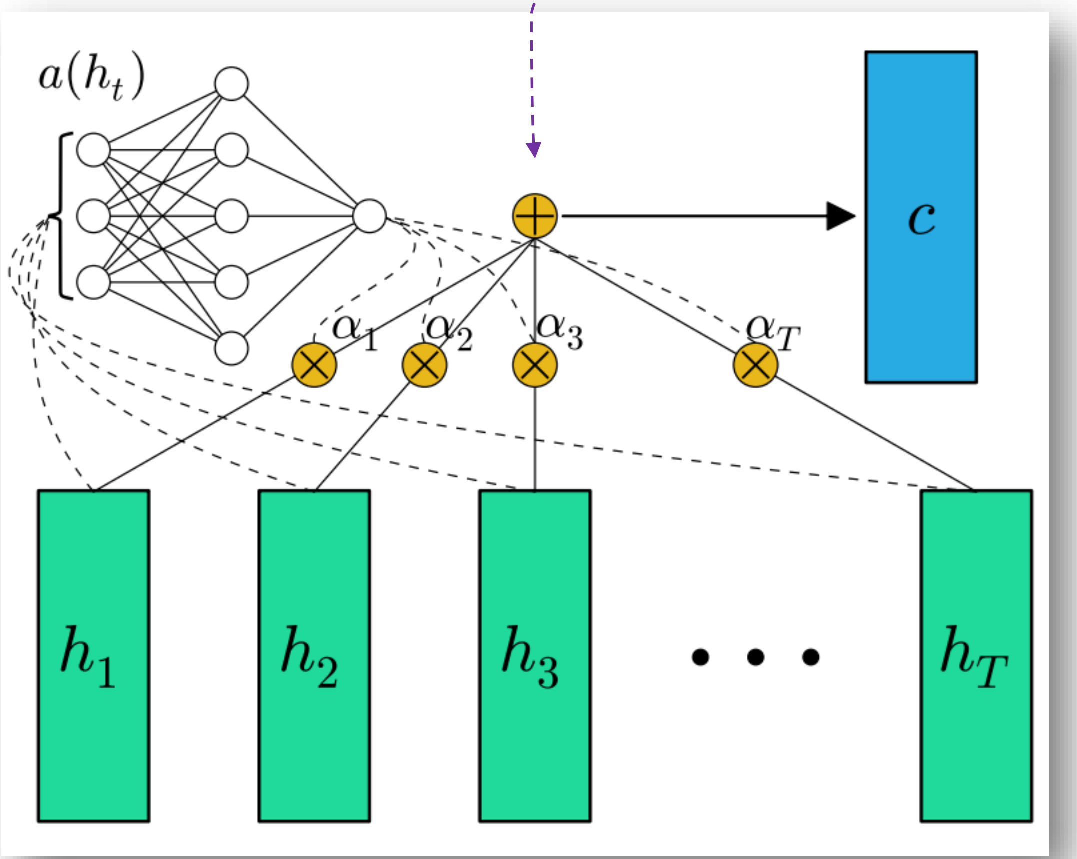


Output of the attention function:

$$y = \sum \alpha_i h_i$$

Attention refresher

Classification



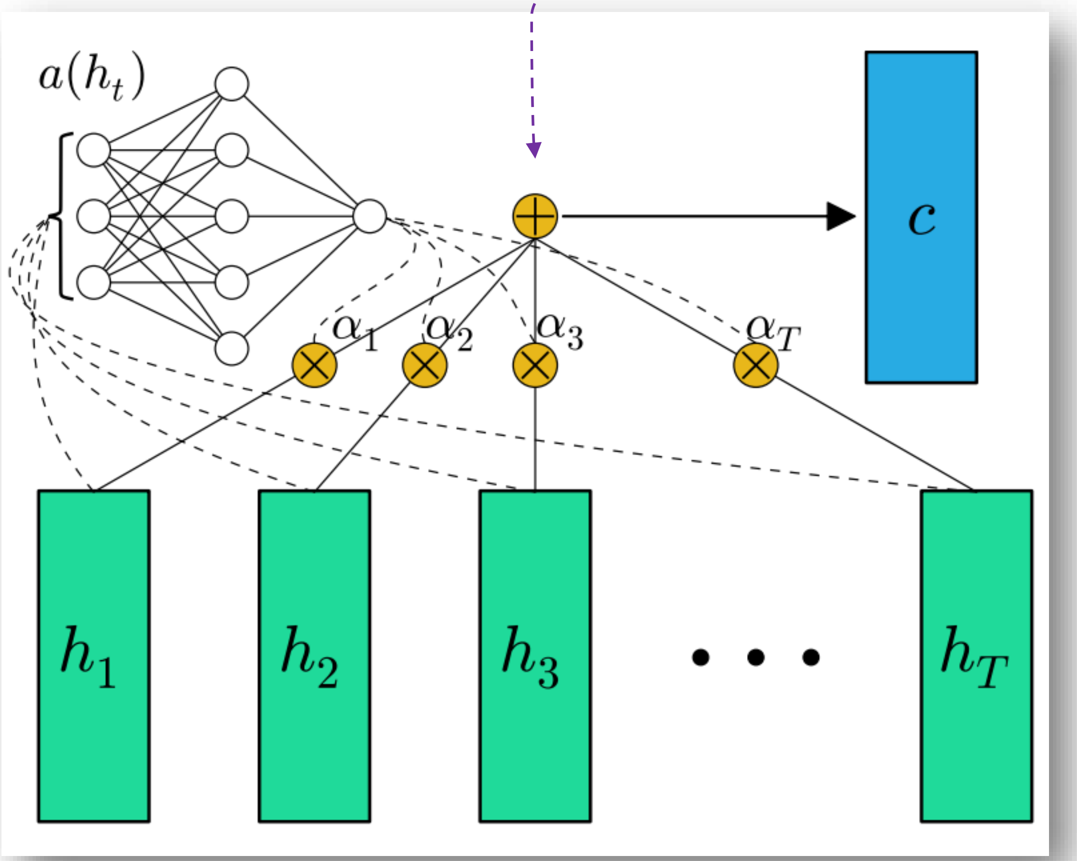
Output of the attention function:

$$y = \sum \alpha_i h_i$$

<u>**Consequently**</u>:
Given a sequence of contextual representations of the words in the sentence, attention can significantly help classification by pointing to highest information bearing words. However, …

Attention refresher

Classification



Output of the attention function:

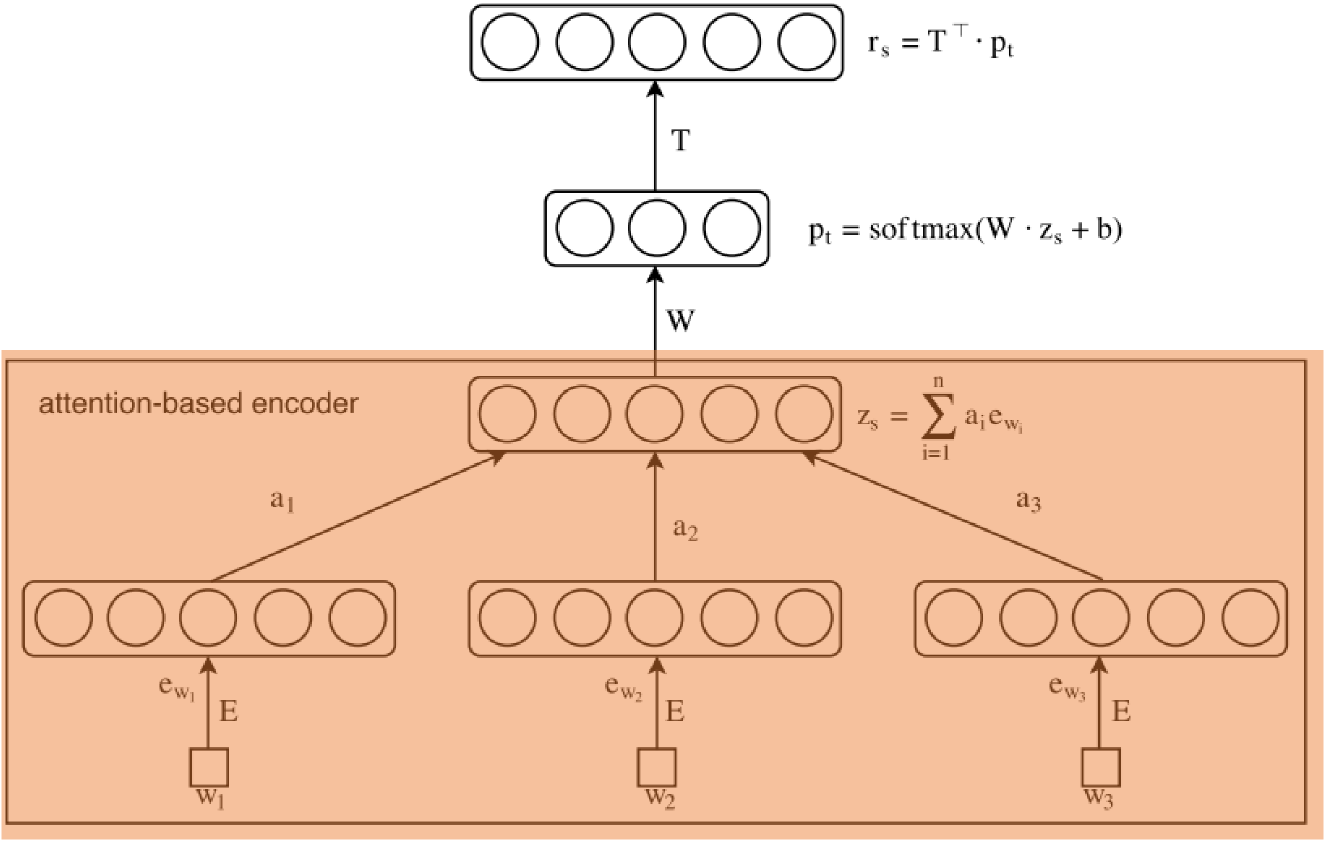
$$y = \sum \alpha_i h_i$$

Consequently:
Given a sequence of contextual representations of the words in the sentence, attention can significantly help classification by pointing to highest information bearing words. However, …

Our issue:
We can only build good contextual embeddings through a sequential model. Can we do something about that?

Attention refresher

Self-attention



<u>Our solution:</u>

We can build a contextual representation through some similarity function between the word embedding and the full context.

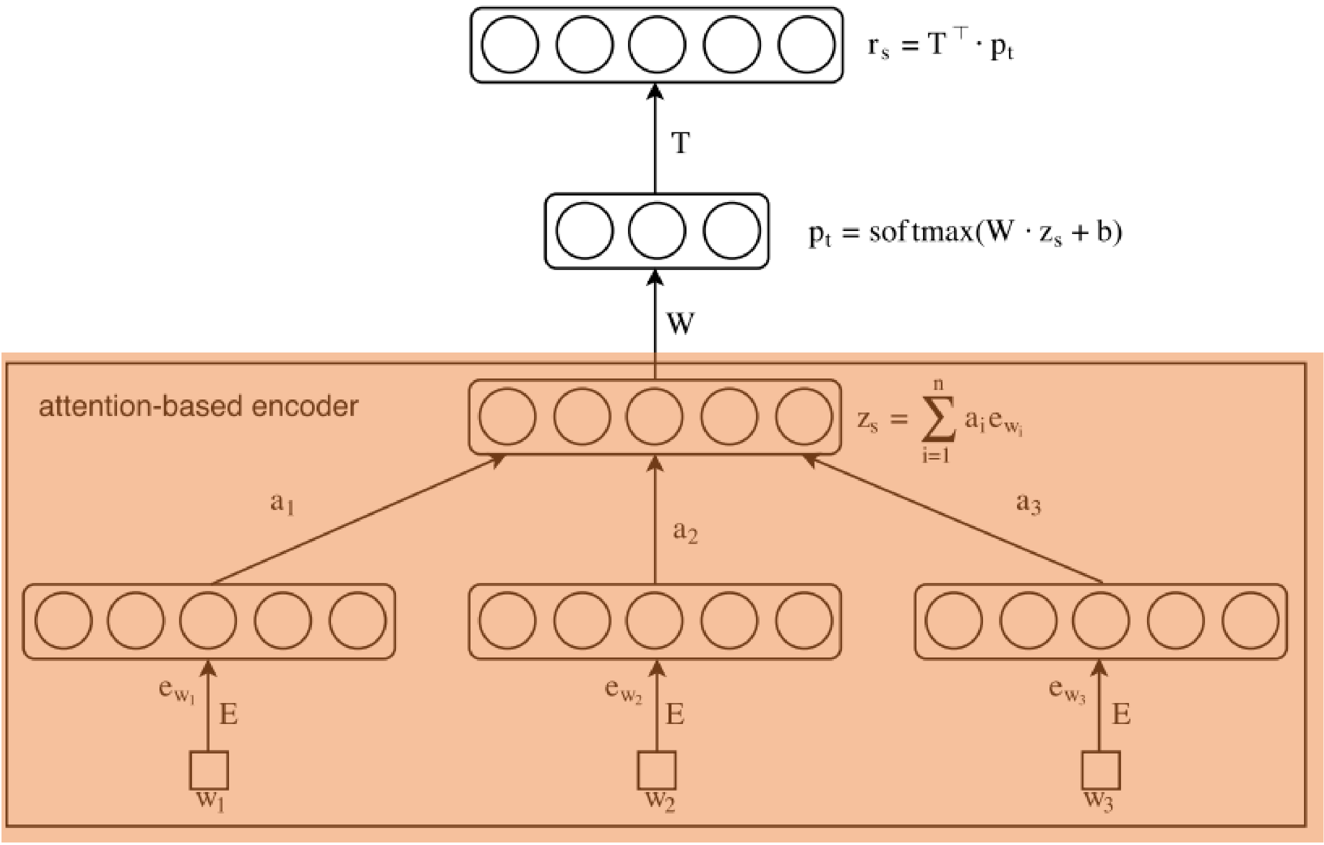
$$a_i = \frac{\exp(d_i)}{\sum_{j=1}^{n} \exp(d_j)}$$

$$d_i = \mathbf{e}_{w_i}^\top \cdot \mathbf{M} \cdot \mathbf{y}_s$$

$$\mathbf{y}_s = \frac{1}{n} \sum_{i=1}^{n} \mathbf{e}_{w_i}$$

Attention refresher

Self-attention



Our solution:

We can build a contextual representation through some similarity function between the word embedding and the full context.

Our 2nd issue:

This is too low resolution since we're averaging over words. If we were not worried about # of parameters, what can we do?

$$a_i = \frac{\exp(d_i)}{\sum_{j=1}^{n} \exp(d_j)}$$

$$d_i = \mathbf{e}_{w_i}^\top \cdot \mathbf{M} \cdot \mathbf{y}_s$$

$$\mathbf{y}_s = \frac{1}{n} \sum_{i=1}^{n} \mathbf{e}_{w_i}$$

Attention Is All You Need

Ashish Vaswani* Google Brain avaswani@google.com

Noam Shazeer* Google Brain noam@google.com

Niki Parmar* Google Research nikip@google.com

Jakob Uszkoreit* Google Research usz@google.com

Llion Jones* Google Research llion@google.com

Aidan N. Gomez* † University of Toronto aidan@cs.toronto.edu

Łukasz Kaiser* Google Brain lukaszkaiser@google.com

Illia Polosukhin* ‡ illia.polosukhin@gmail.com

We're going to use Peter Bloem's blog post:
http://www.peterbloem.nl/blog/transformers

Architecture

Let’s work this out …

N.B.: We’re going to address only the encoder part. But the same ideas apply to understand the decoder.

Architecture

The self attention operation

- Input vectors: $x_1, x_2, \dots, x_t$ and output vectors: $y_1, y_2, \dots, y_t$

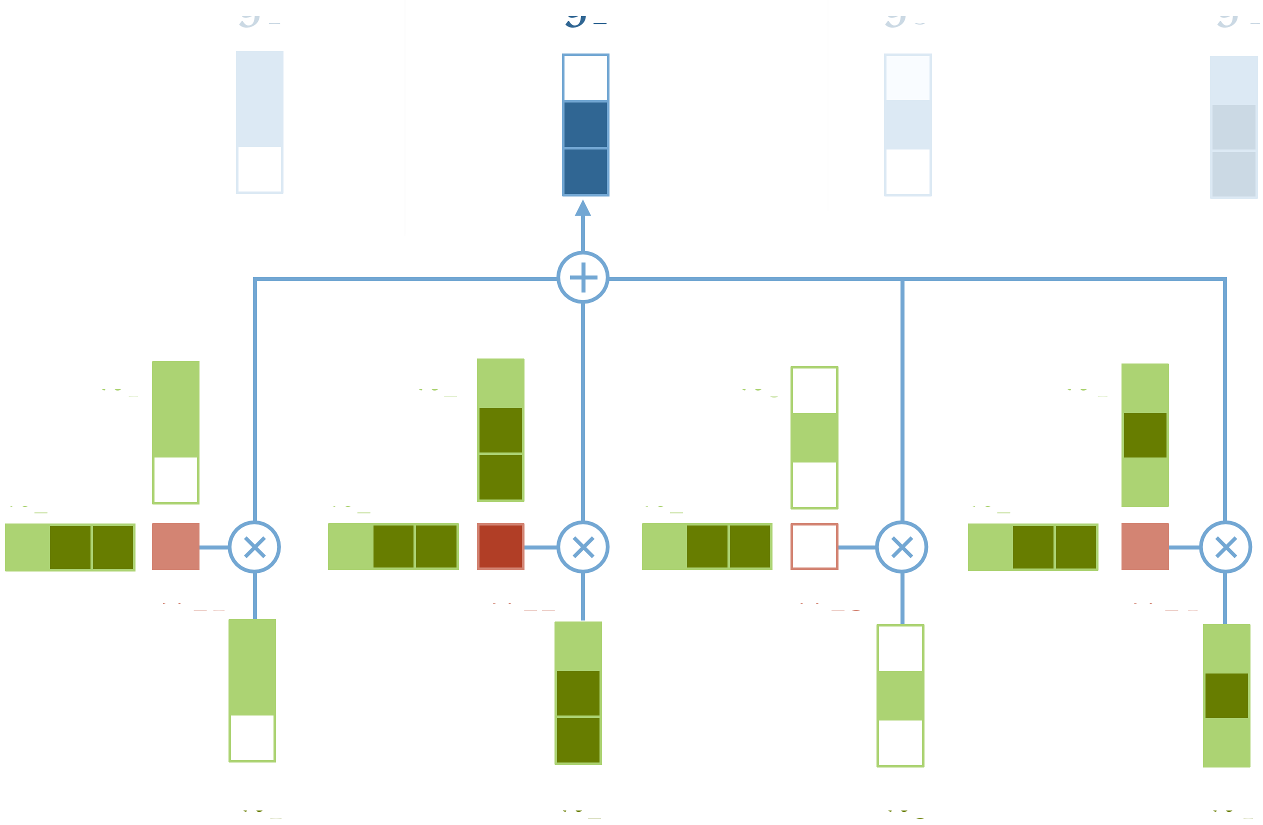
$$\mathbf{y}_i = \sum_j w_{ij} \mathbf{x}_j$$

- Simplest way to get the w_{ij}'s

$$w_{ij} = \frac{\exp w'_{ij}}{\sum_j \exp w'_{ij}} \qquad \textit{where} \qquad w'_{ij} = \mathbf{x}_i^{\mathsf{T}} \mathbf{x}_j$$

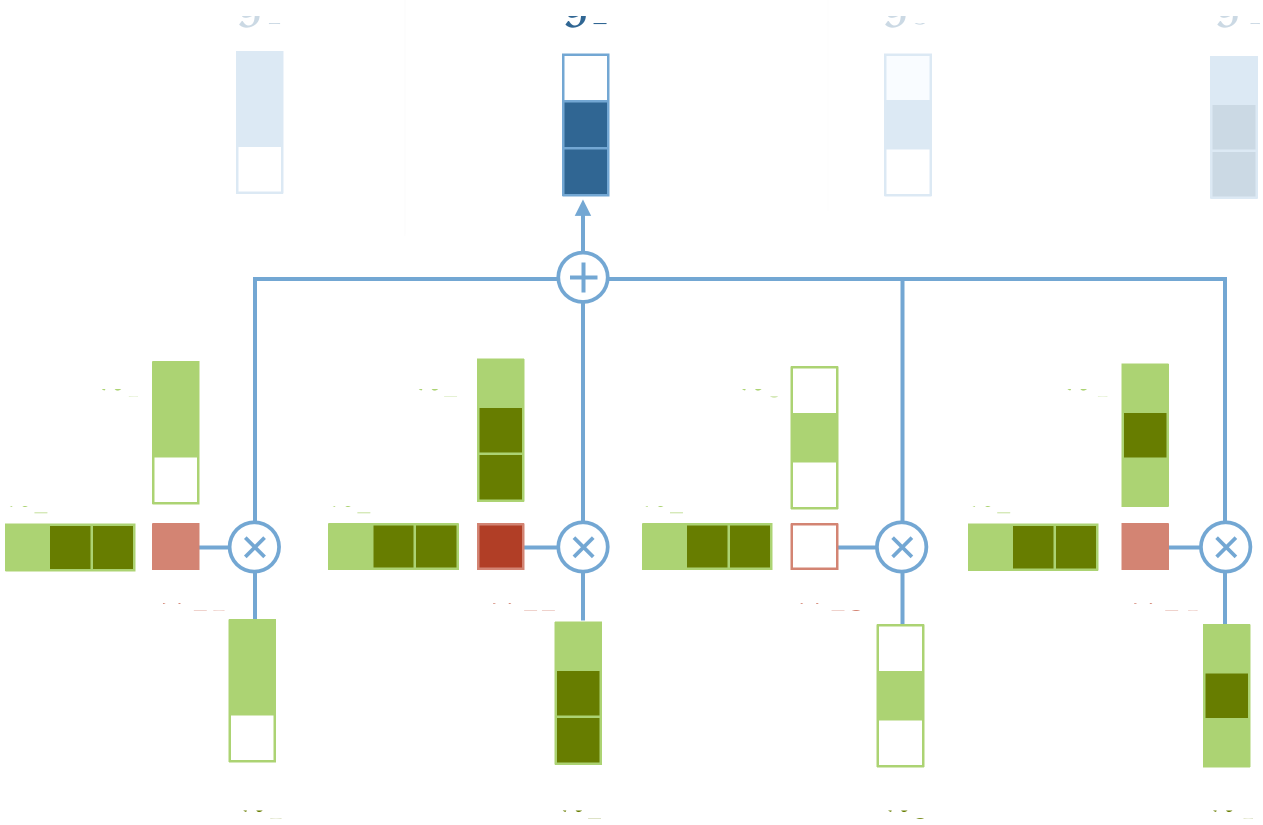
Architecture

The self attention operation



Architecture

The self attention operation



So far, no parameters .. Let’s throw in a bunch of them

Architecture
Queries, keys and values

$$w'_{ij} = \mathbf{x}_i^{\mathsf{T}} \mathbf{x}_j$$

$$w_{ij} = \frac{\exp w'_{ij}}{\sum_j \exp w'_{ij}}$$

$$\mathbf{y}_i = \sum_j w_{ij} \mathbf{x}_j$$

Every input vector $\mathbf{x}_i$ is used in three different ways in the self attention operation:

- It is compared to every other vector to establish the weights for its own output $\mathbf{y}_i$
- It is compared to every other vector to establish the weights for the output of the j-th vector $\mathbf{y}_j$
- It is used as part of the weighted sum to compute each output vector once the weights have been established.

Architecture
Queries, keys and values

Every input vector $\mathbf{x}_i$ is used in three different ways in the self attention operation:

- It is compared to every other vector to establish the weights for its own output $\mathbf{y}_i$
- It is compared to every other vector to establish the weights for the output of the j-th vector $\mathbf{y}_j$
- It is used as part of the weighted sum to compute each output vector once the weights have been established.

$$w'_{ij} = \mathbf{x}_i{}^{\mathsf{T}}\mathbf{x}_j$$

$$w_{ij} = \frac{\exp w'_{ij}}{\sum_j \exp w'_{ij}}$$

$$\mathbf{y}_i = \sum_j w_{ij}\mathbf{x}_j$$

$$\Downarrow$$

$$\mathbf{q}_i = \mathbf{W}_q\mathbf{x}_i \qquad \mathbf{k}_i = \mathbf{W}_k\mathbf{x}_i \qquad \mathbf{v}_i = \mathbf{W}_v\mathbf{x}_i$$

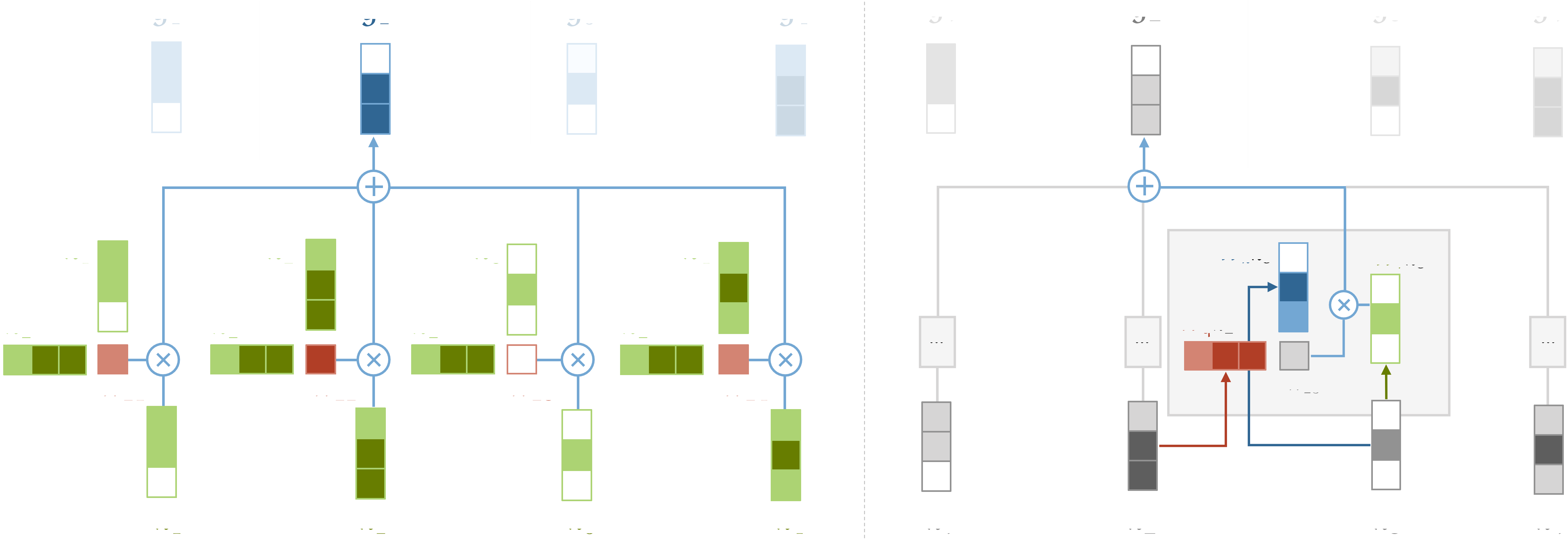
$$w'_{ij} = \mathbf{q}_i{}^{\mathsf{T}}\mathbf{k}_j$$

$$w_{ij} = \text{softmax}(w'_{ij})$$

$$\mathbf{y}_i = \sum_j w_{ij}\mathbf{v}_j\,.$$

Architecture

Queries, keys and values



Architecture

Scaling the dot product

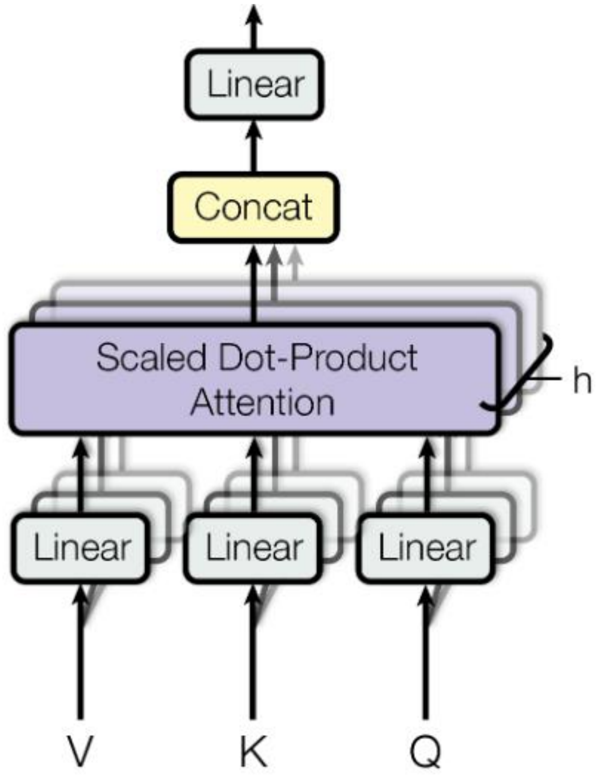
- To avoid having very large values for the attention scores (since they grow with k the dimension of the embeddings), we're going to add a normalization factor:

$$w'_{ij} = \mathbf{q}_i{}^{\mathsf{T}}\mathbf{k}_j \quad \Longrightarrow \quad w'_{ij} = \frac{\mathbf{q}_i{}^{\mathsf{T}}\mathbf{k}_j}{\sqrt{k}}$$

Architecture

Multi-head attention

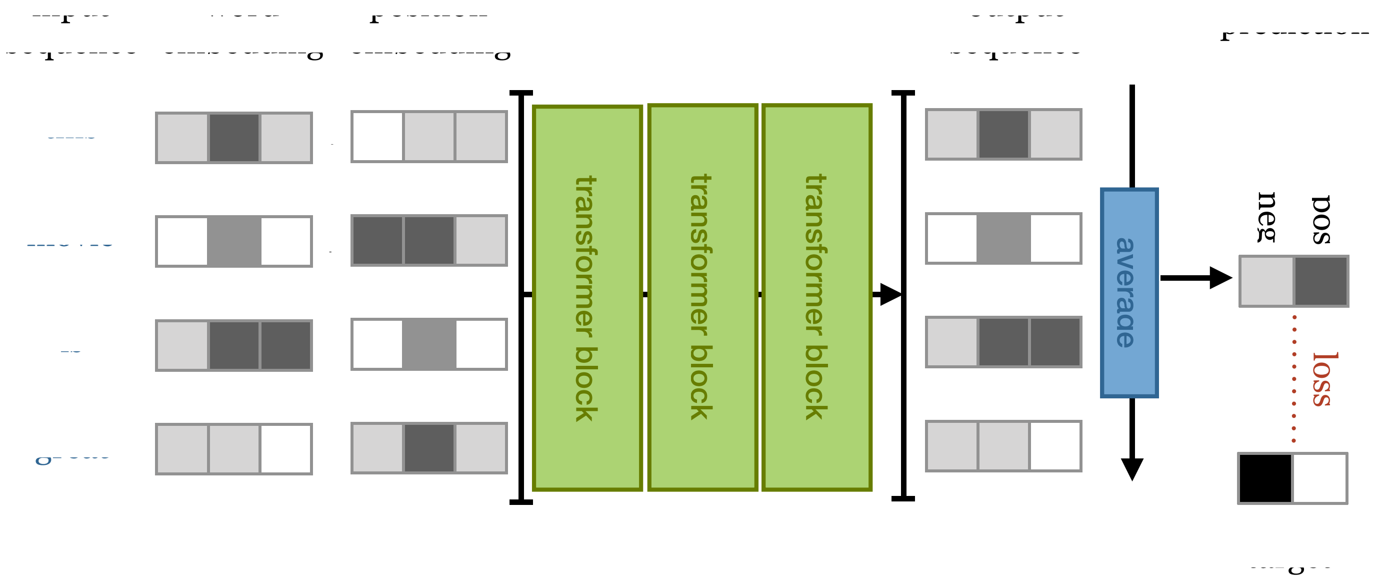
- Consider the idea that one self attention block can focus only one type of relations between words.
- For different types of tasks we might need to capture one or many types of relations
- We can borrow a nice trick from CNNs: replicate the same kernel size many times, init randomly and let the training process drive them towards finding different patterns.
    - In this case different relations since all the parameters operate between two word embeddings



Architecture
Positional encoding

- At this point, we have not included any mechanism to make sure the model takes into consideration the order of the words

- Positional encoding does exactly that. It is a vector that we build in function of the position of word in the sentence and the dimension. Afterwards, we ***add*** it to the original embedding.



Architecture
Positional encoding

- At this point, we have not included any mechanism to make sure the model takes into consideration the order of the words

- Positional encoding does exactly that. It is a vector that we build in function of the position of word in the sentence and the dimension. Afterwards, we ***add*** it to the original embedding.

- How do we compute that?

Architecture
Positional encoding

- At this point, we have not included any mechanism to make sure the model takes into consideration the order of the words

- Positional encoding does exactly that. It is a vector that we build in function of the position of word in the sentence and the dimension. Afterwards, we ***add*** it to the original embedding.

- How do we compute that?

In this work, we use sine and cosine functions of different frequencies:

$$PE_{(pos,2i)} = sin(pos/10000^{2i/d_{\text{model}}})$$
$$PE_{(pos,2i+1)} = cos(pos/10000^{2i/d_{\text{model}}})$$

where pos is the position and i is the dimension. That is, each dimension of the positional encoding corresponds to a sinusoid. The wavelengths form a geometric progression from 2π to $10000 \cdot 2\pi$. We chose this function because we hypothesized it would allow the model to easily learn to attend by relative positions, since for any fixed offset k, PE_{pos+k} can be represented as a linear function of PE_{pos}.

Architecture

Positional encoding

- At this point, we have not included any me
order of the words

- Positional encoding does exactly that. It is a
sentence and the dimension. Afterwards, w

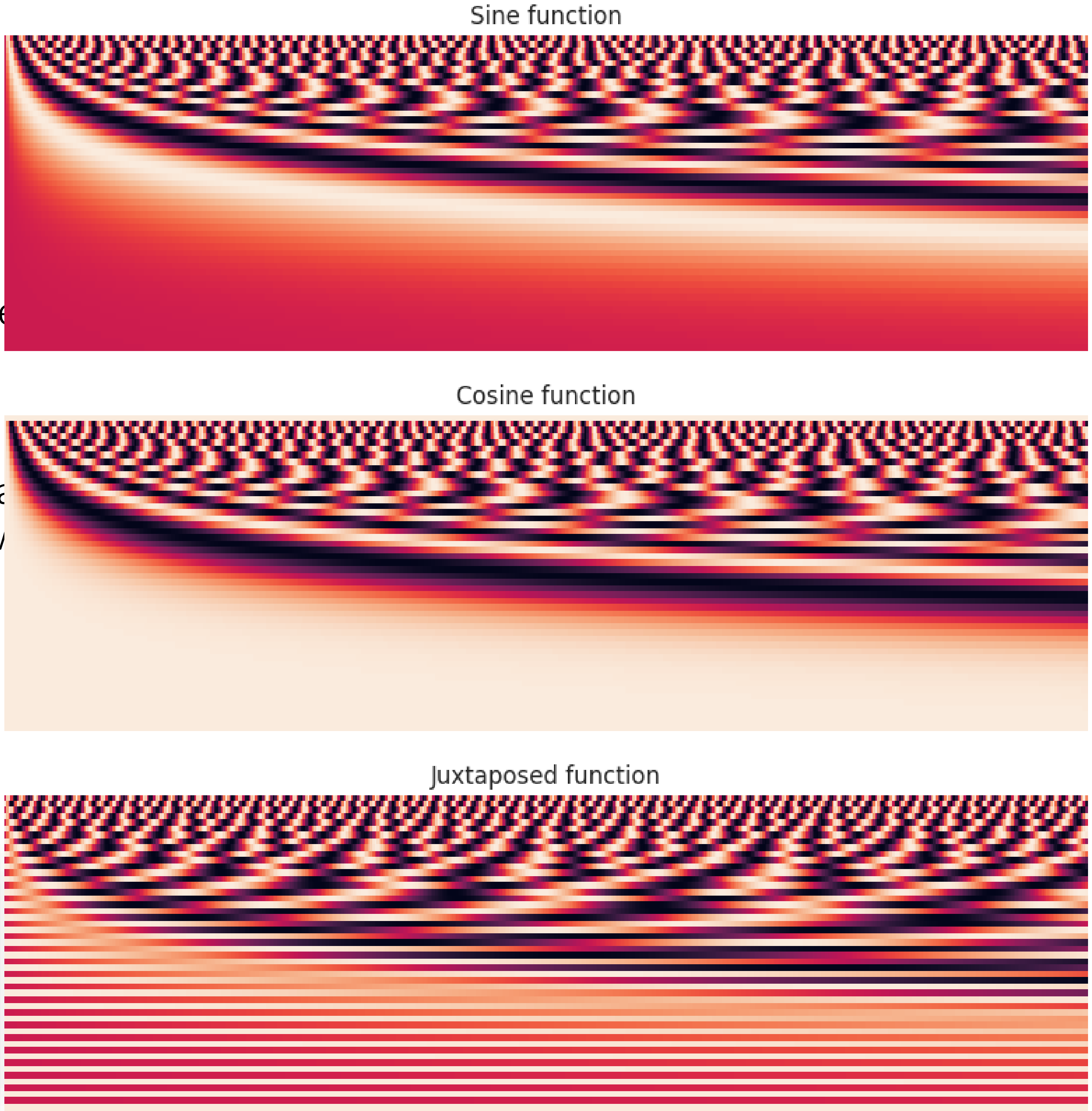
- How do we compute that?

- It looks like this

In this work, we use sine and cosine functions of different frequencies:

$$PE_{(pos,2i)} = sin(pos/10000^{2i/d_{\text{model}}})$$
$$PE_{(pos,2i+1)} = cos(pos/10000^{2i/d_{\text{model}}})$$

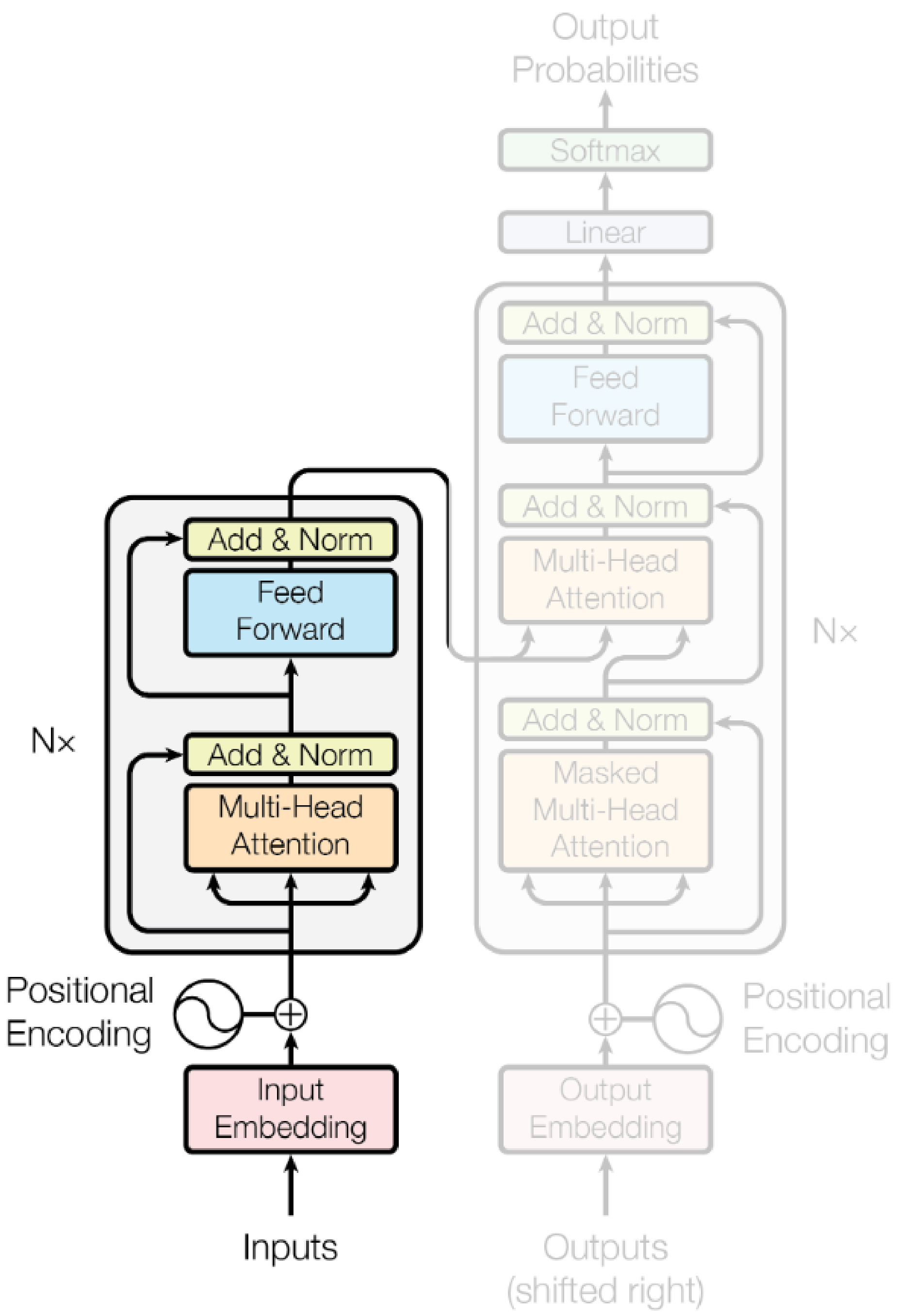
where pos is the position and i is the dimension. That is, each dimension of the positional encoding corresponds to a sinusoid. The wavelengths form a geometric progression from 2π to $10000 \cdot 2\pi$. We chose this function because we hypothesized it would allow the model to easily learn to attend by relative positions, since for any fixed offset k, PE_{pos+k} can be represented as a linear function of PE_{pos}.

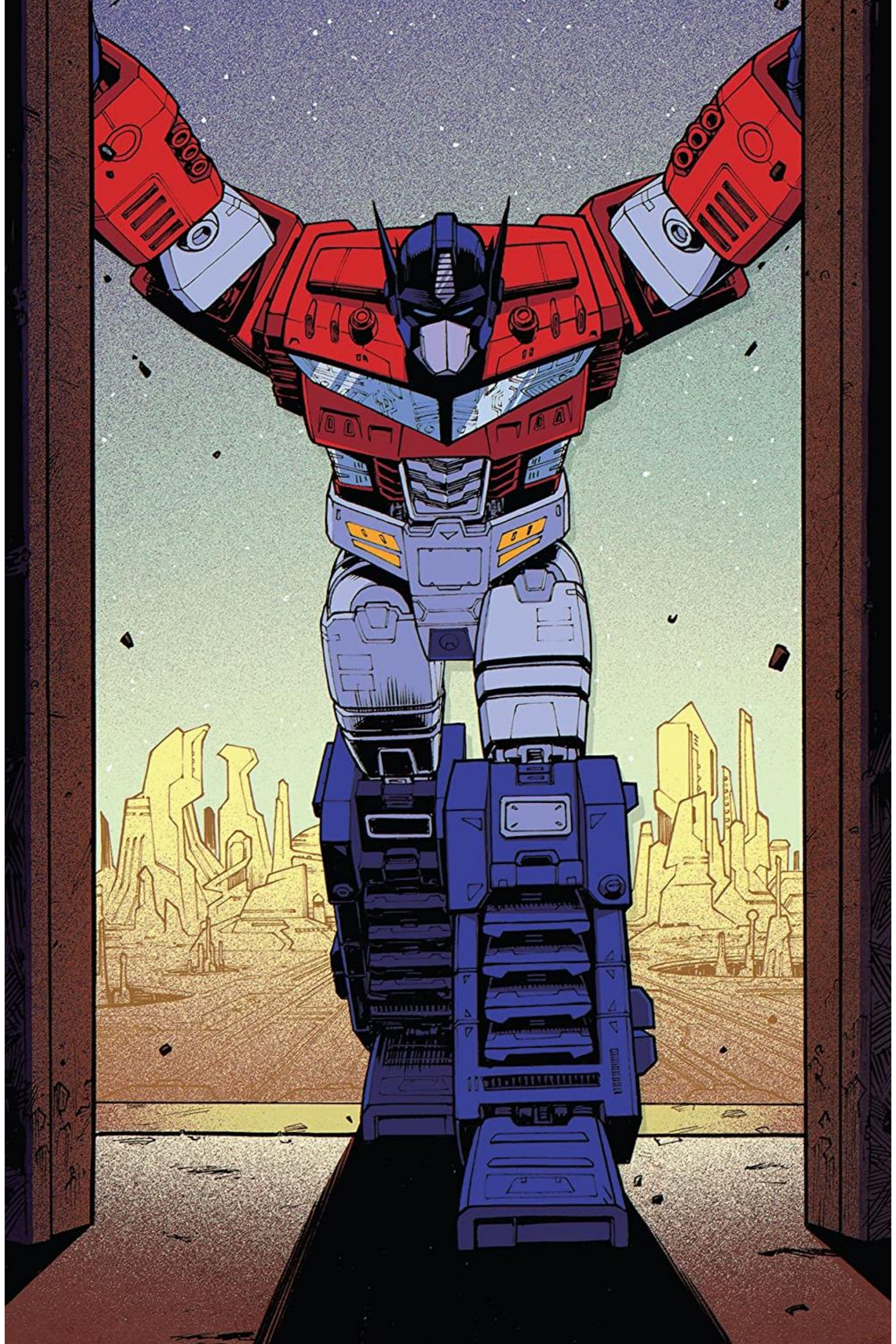


Architecture

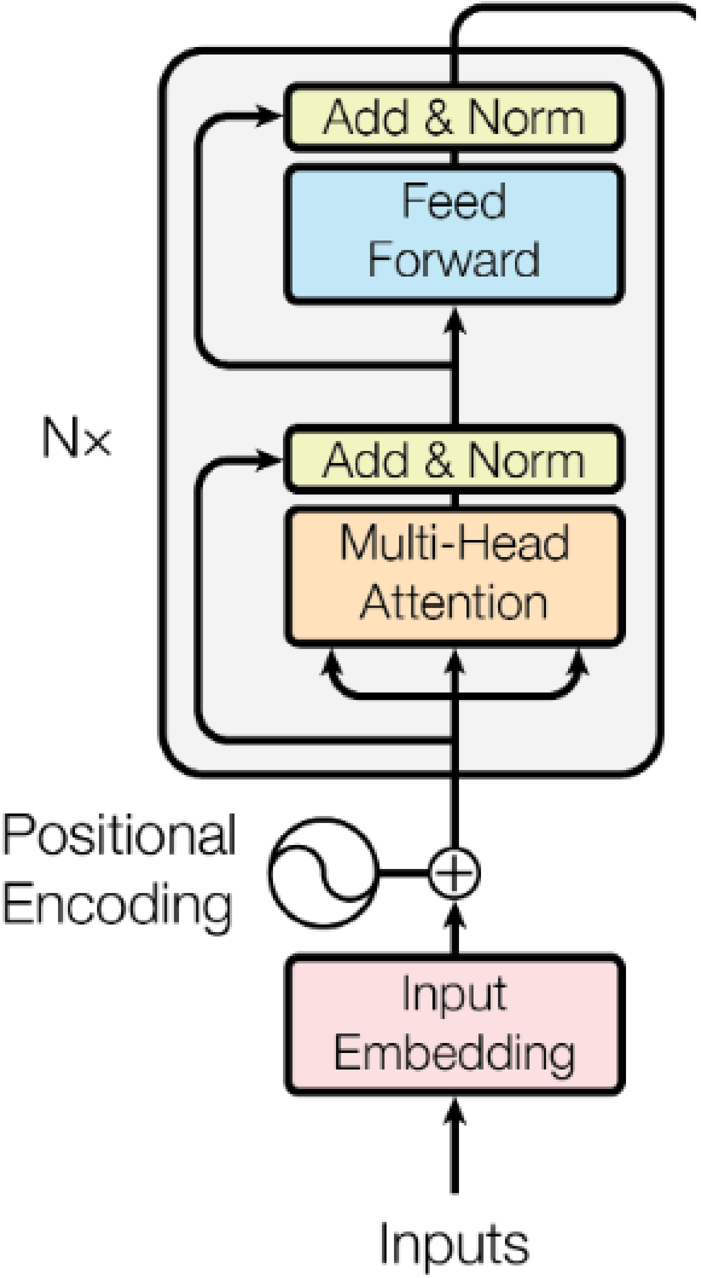
Other tricks

- The architecture uses norm layers and residuals that help to train better.

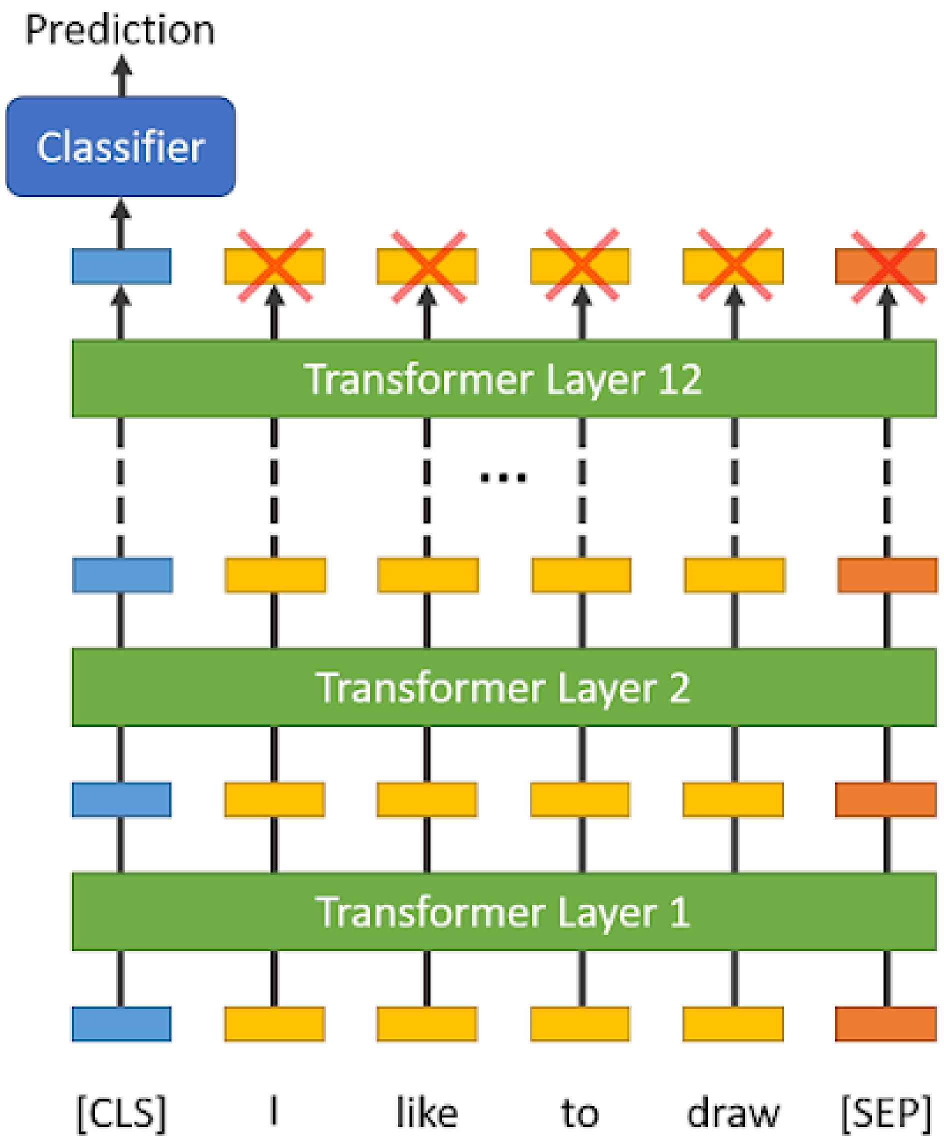




Transformer



BERT


Prediction
Classifier
Transformer Layer 12
...
Transformer Layer 2
Transformer Layer 1
[CLS]
I
like
to
draw
[SEP]

The Idea
In a nutshell

- Take 7000 books and all of Wikipedia as a corpus and organize the whole thing as pairs of sentences (we'll say why pairs in a sec)
- Stack N transformers on top of each other
- Task: for each word in the first sentence, mask it and try to predict it. Also, try to predict the next sentence (this is why we need pairs)
- Let's make it more exciting: we're also going to perturb the sentences in many ways: removing words, swapping words, etc.
- With 12 layers (as our architecture) we'll end up with 109,482,240 parameters, and that's BERT.
- Let it train for four days on 4 to 16 Cloud TPUs.
- Now for each NLP task, we fine tune BERT .. Will that work?

Results

System	MNLI-(m/mm)	QQP	QNLI	SST-2	CoLA	STS-B	MRPC	RTE	**Average**
	392k	363k	108k	67k	8.5k	5.7k	3.5k	2.5k	-
Pre-OpenAI SOTA	80.6/80.1	66.1	82.3	93.2	35.0	81.0	86.0	61.7	74.0
BiLSTM+ELMo+Attn	76.4/76.1	64.8	79.8	90.4	36.0	73.3	84.9	56.8	71.0
OpenAI GPT	82.1/81.4	70.3	87.4	91.3	45.4	80.0	82.3	56.0	75.1
$\text{BERT}_{\text{BASE}}$	84.6/83.4	71.2	90.5	93.5	52.1	85.8	88.9	66.4	79.6
$\text{BERT}_{\text{LARGE}}$	**86.7/85.9**	**72.1**	**92.7**	**94.9**	**60.5**	**86.5**	**89.3**	**70.1**	**82.1**

Results

System	MNLI-(m/mm)	QQP	QNLI	SST-2	CoLA	STS-B	MRPC	RTE	**Average**
	392k	363k	108k	67k	8.5k	5.7k	3.5k	2.5k	-
Pre-OpenAI SOTA	80.6/80.1	66.1	82.3	93.2	35.0	81.0	86.0	61.7	74.0
BiLSTM+ELMo+Attn	76.4/76.1	64.8	79.8	90.4	36.0	73.3	84.9	56.8	71.0
OpenAI GPT	82.1/81.4	70.3	87.4	91.3	45.4	80.0	82.3	56.0	75.1
$\text{BERT}_{\text{BASE}}$	84.6/83.4	71.2	90.5	93.5	52.1	85.8	88.9	66.4	79.6
$\text{BERT}_{\text{LARGE}}$	**86.7/85.9**	**72.1**	**92.7**	**94.9**	**60.5**	**86.5**	**89.3**	**70.1**	**82.1**

More importantly though:
(drum roll)

Results

https://cl.lingfil.uu.se/~nivre/docs/NivreEurNLP2019.pdf

UPPSALA
UNIVERSITET

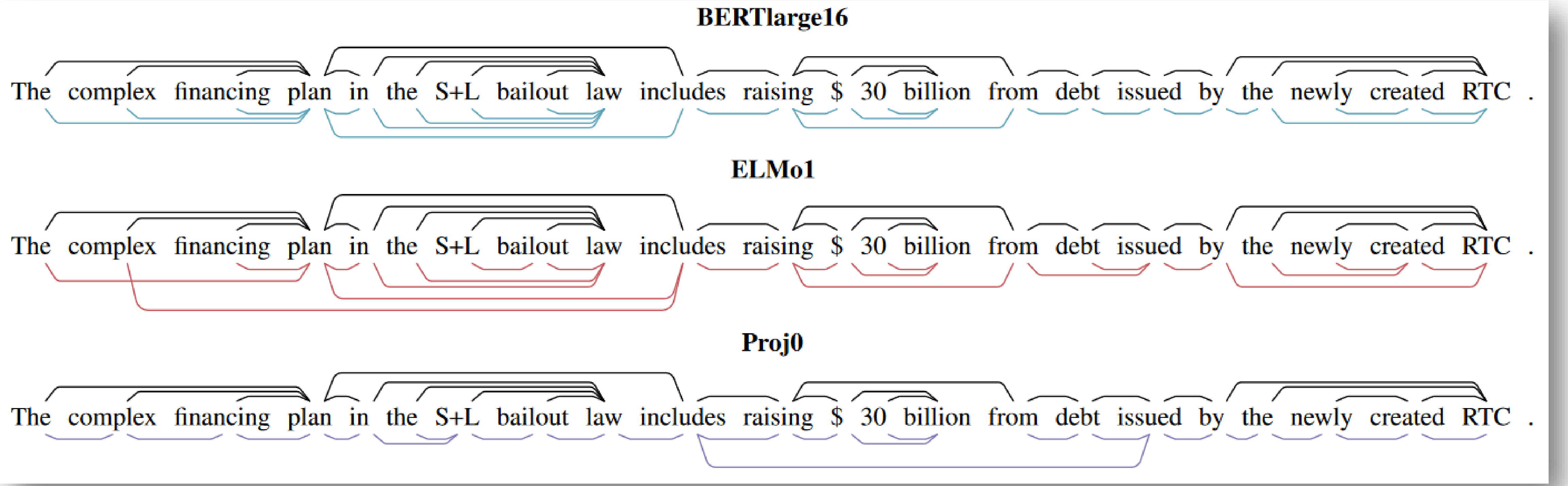
Is the End of Supervised Parsing in Sight?
Twelve Years Later

Joakim Nivre

Uppsala University
Department of Linguistics and Philology

Results

https://cl.lingfil.uu.se/~nivre/docs/NivreEurNLP2019.pdf



John Hewitt and Christopher D. Manning 2019. A Structural Probe for Finding Syntax in Word Representations. In *Proceedings of NAACL*, pages 4129–4138.

Results

Word segmentation, parsing, NER

- BERT Meets Chinese Word Segmentation
- Unified Multi-Criteria Chinese Word Segmentation with BERT
- Toward Fast and Accurate Neural Chinese Word Segmentation with Multi-Criteria Learning
- Establishing Strong Baselines for the New Decade: Sequence Tagging, Syntactic and Semantic Parsing with BERT (FLAIRS-33)
- Evaluating Contextualized Embeddings on 54 Languages in POS Tagging, Lemmatization and Dependency Parsing
- NEZHA: Neural Contextualized Representation for Chinese Language Understanding
- Deep Contextualized Word Embeddings in Transition-Based and Graph-Based Dependency Parsing -- A Tale of Two Parsers Revisited (EMNLP2019)
- Is POS Tagging Necessary or Even Helpful for Neural Dependency Parsing?
- Parsing as Pretraining (AAAI2020)
- Cross-Lingual BERT Transformation for Zero-Shot Dependency Parsing
- Recursive Non-Autoregressive Graph-to-Graph Transformer for Dependency Parsing with Iterative Refinement
- Named Entity Recognition -- Is there a glass ceiling? (CoNLL2019)
- A Unified MRC Framework for Named Entity Recognition
- Training Compact Models for Low Resource Entity Tagging using Pre-trained Language Models
- Robust Named Entity Recognition with Truecasing Pretraining (AAAI2020)
- LTP: A New Active Learning Strategy for Bert-CRF Based Named Entity Recognition
- Interpretability Analysis for Named Entity Recognition to Understand System Predictions and How They Can Improve
- Single-/Multi-Source Cross-Lingual NER via Teacher-Student Learning on Unlabeled Data in Target Language (ACL2020)
- MT-BioNER: Multi-task Learning for Biomedical Named Entity Recognition using Deep Bidirectional Transformers
- Portuguese Named Entity Recognition using BERT-CRF
- Towards Lingua Franca Named Entity Recognition with BERT

Results

Word segmentation, parsing, NER

- BERT Meets Chinese Word Segmentation
- Unified Multi-Criteria Chinese Word Segmentation with BERT
- Toward Fast and Accurate Neural Chinese Word Segmentation with Multi-Criteria Learning
- Establishing Strong Baselines for the New Decade: Sequence Tagging, Syntactic and Semantic Parsing with BERT (FLAIRS-33)
- Evaluating Contextualized Embeddings on 54 Languages in POS Tagging, Lemmatization and Dependency Parsing
- NEZHA: Neural Contextualized Representation for Chinese Language Understanding
- Deep Contextualized Word Embeddings in Transition-Based and Graph-Based Dependency Parsing -- A Tale of Two Parsers Revisited (EMNLP2019)
- Is POS Tagging Necessary or Even Helpful for Neural Dependency Parsing?
- Parsing as Pretraining (AAAI2020)
- Cross-Lingual BERT Transformation for Zero-Shot Dependency Parsing
- Recursive Non-Autoregressive Graph-to-Gra
- Named Entity Recognition -- Is there a glass
- A Unified MRC Framework for Named Entity
- Training Compact Models for Low Resource E
- Robust Named Entity Recognition with Truec
- LTP: A New Active Learning Strategy for Bert-
- Interpretability Analysis for Named Entity Rec
- Single-/Multi-Source Cross-Lingual NER via T
- MT-BioNER: Multi-task Learning for Biomedic
- Portuguese Named Entity Recognition using
- Towards Lingua Franca Named Entity Recogn

Sentiment analysis

- Utilizing BERT for Aspect-Based Sentiment Analysis via Constructing Auxiliary Sentence (NAACL2019)
- BERT Post-Training for Review Reading Comprehension and Aspect-based Sentiment Analysis (NAACL2019)
- Exploiting BERT for End-to-End Aspect-based Sentiment Analysis (EMNLP2019 WS)
- Adapt or Get Left Behind: Domain Adaptation through BERT Language Model Finetuning for Aspect-Target Sentiment Classification
- An Investigation of Transfer Learning-Based Sentiment Analysis in Japanese (ACL2019)
- "Mask and Infill" : Applying Masked Language Model to Sentiment Transfer
- Adversarial Training for Aspect-Based Sentiment Analysis with BERT
- Utilizing BERT Intermediate Layers for Aspect Based Sentiment Analysis and Natural Language Inference

Results

Word segmentation, parsing, NER

- BERT Meets Chinese Word Segmentation
- Unified Multi-Criteria Chinese Word Segmentation with BERT
- Toward Fast and Accurate Neural Chinese Word Segmentation with Multi-Criteria L
- Establishing Strong Baselines for the New Decade: Sequence Tagging, Syntactic and
- Evaluating Contextualized Embeddings on 54 Languages in POS Tagging, Lemmati
- NEZHA: Neural Contextualized Representation for Chinese Language Understandin
- Deep Contextualized Word Embeddings in Transition-Based and Graph-Based Dep Revisited (EMNLP2019)
- Is POS Tagging Necessary or Even Helpful for Neural Dependency Parsing?
- Parsing as Pretraining (AAAI2020)
- Cross-Lingual BERT Transformation for Zero-Shot Dependency Parsing
- Recursive Non-Autoregressive Graph-to-Gra
- Named Entity Recognition -- Is there a glass

Word sense disambiguation

- GlossBERT: BERT for Word Sense Disambiguation with Gloss Knowledge (EMNLP2019)
- Improved Word Sense Disambiguation Using Pre-Trained Contextualized Word Representations (EMNLP2019)
- Using BERT for Word Sense Disambiguation
- Language Modelling Makes Sense: Propagating Representations through WordNet for Full-Coverage Word Sense Disambiguation (ACL2019)
- Does BERT Make Any Sense? Interpretable Word Sense Disambiguation with Contextualized Embeddings (KONVENS2019)

Sentiment analysis

- nstructing Auxiliary Sentence (NAACL2019)
- l Aspect-based Sentiment Analysis (NAACL2019)
- nalysis (EMNLP2019 WS)
- T Language Model Finetuning for Aspect-Target Sentiment
- lysis in Japanese (ACL2019)
- ntiment Transfer
- ith BERT
- ent Analysis and Natural Language Inference

Relation extraction

- Matching the Blanks: Distributional Similarity for Relation Learning (ACL2019)
- BERT-Based Multi-Head Selection for Joint Entity-Relation Extraction (NLPCC2019)
- Enriching Pre-trained Language Model with Entity Information for Relation Classification
- Span-based Joint Entity and Relation Extraction with Transformer Pre-training
- Fine-tune Bert for DocRED with Two-step Process
- Entity, Relation, and Event Extraction with Contextualized Span Representations (EMNLP2019)
- Fine-tuning BERT for Joint Entity and Relation Extraction in Chinese Medical Text
- Downstream Model Design of Pre-trained Language Model for Relation Extraction Task
- Efficient long-distance relation extraction with DG-SpanBERT
- Robustly Pre-trained Neural Model for Direct Temporal Relation Extraction
- DARE: Data Augmented Relation Extraction with GPT-2
- Improving Scholarly Knowledge Representation: Evaluating BERT-based Models for Scientific Relation Classification

Results

Word segmentation, parsing, NER

- BERT Meets Chinese Word Segmentation
- Unified Multi-Criteria Chinese Word Segmentation with BERT
- Toward Fast and Accurate Neural Chinese Word Segmentation with Multi-C
- Establishing Strong Baselines for the New Decade: Sequence Tagging, Synt
- Evaluating Contextualized Embeddings on 54 Languages in POS Tagging, L
- NEZHA: Neural Contextualized Representation for Chinese Language Unde
- Deep Contextualized Word Embeddings in Transition-Based and Graph-Bas Revisited (EMNLP2019)
- Is POS Tagging Necessary or Even Helpful for Neural Dependency Parsing?
- Parsing as Pretraining (AAAI2020)
- Cross-Lingual BERT Transformation for Zero-Shot Dependency Parsing
- Recursive Non-Autoregressive Graph-to-Gra
- Named Entity Recognition -- Is there a glass

Sentiment analysis

- T Language Model Finetuning for Aspect-Target Sentiment
- lysis in Japanese (ACL2019)
- ntiment Transfer
- ith BERT
- ent Analysis and Natural Language Inference

Knowledge base

- KG-BERT: BERT for Knowledge Graph Completion
- Language Models as Knowledge Bases? (EMNLP2019) [github]
- BERT is Not a Knowledge Base (Yet): Factual Knowledge vs. Name-Based Reasoning in Unsupervised QA
- Inducing Relational Knowledge from BERT (AAAI2020)
- Latent Relation Language Models (AAAI2020)
- Pretrained Encyclopedia: Weakly Supervised Knowledge-Pretrained Language Model (ICLR2020)
- Zero-shot Entity Linking with Dense Entity Retrieval [github]
- Investigating Entity Knowledge in BERT with Simple Neural End-To-End Entity Linking (CoNLL2019)
- Improving Entity Linking by Modeling Latent Entity Type Information (AAAI2020)
- Global Entity Disambiguation with Pretrained Contextualized Embeddings of Words and Entities
- YELM: End-to-End Contextualized Entity Linking
- PEL-BERT: A Joint Model for Protocol Entity Linking
- How Can We Know What Language Models Know?
- REALM: Retrieval-Augmented Language Model Pre-Training
- Deep Entity Matching with Pre-Trained Language Models

2019)

ord Sense

s (KONVENS2019)

Relation extraction

- Matching the Blanks: Distributional Similarity for Relation Learning (ACL2019)
- BERT-Based Multi-Head Selection for Joint Entity-Relation Extraction (NLPCC2019)
- Enriching Pre-trained Language Model with Entity Information for Relation Classification
- Span-based Joint Entity and Relation Extraction with Transformer Pre-training
- Fine-tune Bert for DocRED with Two-step Process
- Entity, Relation, and Event Extraction with Contextualized Span Representations (EMNLP2019)
- Fine-tuning BERT for Joint Entity and Relation Extraction in Chinese Medical Text
- Downstream Model Design of Pre-trained Language Model for Relation Extraction Task
- Efficient long-distance relation extraction with DG-SpanBERT
- Robustly Pre-trained Neural Model for Direct Temporal Relation Extraction
- DARE: Data Augmented Relation Extraction with GPT-2
- Improving Scholarly Knowledge Representation: Evaluating BERT-based Models for Scientific Relation Classification

Results

Word segmentation, parsing, NER

- BERT Meets Chinese Word Segmentation
- Unified Multi-Criteria Chinese Word Segmentation with BERT
- Toward Fast and Accurate Neural Chinese Word Segmentation with Multi-C
- Establishing Strong Baselines for the New Decade: Sequence Tagging, Synt
- Evaluating Contextualized Embeddings on 54 Languages in POS Tagging, L
- NEZHA: Neural Contextualized Representation for Chinese Language Unde
- Deep Contextualized Word Embeddings in Transition-Based and Graph-Bas Revisited (EMNLP2019)
- Is POS Tagging Necessary or Even Helpful for Neural Dependency Parsing?
- Parsing as Pretraining (AAAI2020)
- Cross-Lingual BERT Transformation for Zero-Shot Dependency Parsing
- Recursive Non-Autoregressive Graph-to-Gra
- Named Entity Recognition -- Is there a glass

Knowledge base

- KG-BERT: BERT for Knowledge Graph Completion
- Language Models as Knowledge Bases? (EMNLP2019) [github]
- BERT is Not a Knowledge Base (Yet): Factual Knowledge vs. Name-Based Reasoning in Unsupervised QA
- Inducing Relational Knowledge from BERT (AAAI2020)
- Latent Relation Language Models (AAAI2020)
- Pretrained Encyclopedia: Weakly Supervised Knowledge-Pretrained Language Model (ICLR2020)
- Zero-shot Entity Linking with Dense Entity Retrieval [github]
- Investigating Entity Knowledge in BERT with Simple Neural End-To-End Entity Linking (CoNLL2019)
- Improving Entity Linking by Modeling Latent Entity Type Information (AAAI2020)
- Global Entity Disambiguation with Pretrained Contextualized Embeddings of Words and Entities

2019)

ord Sense

s (KONVENS2019)

Sent

Extractive summarization

- HIBERT: Document Level Pre-training of Hierarchical Bidirectional Transformers for Document Summarization (ACL2019)
- Deleter: Leveraging BERT to Perform Unsupervised Successive Text Compression
- Discourse-Aware Neural Extractive Model for Text Summarization
- AREDSUM: Adaptive Redundancy-Aware Iterative Sentence Ranking for Extractive Document Summarization
- Multi-Document Summarization with Determinantal Point Processes and Contextualized Representations (EMNLP2019 WS)

Relation extraction

- Matching the Blanks: Distributional Similarity for Relation
- BERT-Based Multi-Head Selection for Joint Entity-Relation
- Enriching Pre-trained Language Model with Entity Inform
- Span-based Joint Entity and Relation Extraction with Trans
- Fine-tune Bert for DocRED with Two-step Process
- Entity, Relation, and Event Extraction with Contextualized
- Fine-tuning BERT for Joint Entity and Relation Extraction in Chinese Medical Text
- Downstream Model Design of Pre-trained Language Model for Relation Extraction Task
- Efficient long-distance relation extraction with DG-SpanBERT
- Robustly Pre-trained Neural Model for Direct Temporal Relation Extraction
- DARE: Data Augmented Relation Extraction with GPT-2
- Improving Scholarly Knowledge Representation: Evaluating BERT-based Models for Scientific Relation Classification

ith BERT

ent Analysis and Natural Language Inference

Results

Generation

- BERT has a Mouth, and It Must Speak: BERT as a Markov Random Field Language Model (NAACL2019 WS)
- Pretraining-Based Natural Language Generation for Text Summarization
- Text Summarization with Pretrained Encoders (EMNLP2019) [github (original)] [github (huggingface)]
- Multi-stage Pretraining for Abstractive Summarization
- PEGASUS: Pre-training with Extracted Gap-sentences for Abstractive Summarization
- Abstractive Summarization with Combination of Pre-trained Sequence-to-Sequence and Saliency Models
- STEP: Sequence-to-Sequence Transformer Pre-training for Document Summarization
- MASS: Masked Sequence to Sequence Pre-training for Language Generation (ICML2019) [github], [github]
- Unified Language Model Pre-training for Natural Language Understanding and Generation [github] (NeurIPS2019)
- UniLMv2: Pseudo-Masked Language Models for Unified Language Model Pre-Training [github]
- ProphetNet: Predicting Future N-gram for Sequence-to-Sequence Pre-training
- Towards Making the Most of BERT in Neural Machine Translation
- Improving Neural Machine Translation with Pre-trained Representation
- On the use of BERT for Neural Machine Translation (EMNLP2019 WS)
- Incorporating BERT into Neural Machine Translation (ICLR2020)
- Recycling a Pre-trained BERT Encoder for Neural Machine Translation
- Leveraging Pre-trained Checkpoints for Sequence Generation Tasks
- Mask-Predict: Parallel Decoding of Conditional Masked Language Models (EMNLP2019)
- BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension
- PALM: Pre-training an Autoencoding&Autoregressive Language Model for Context-conditioned Generation
- ERNIE-GEN: An Enhanced Multi-Flow Pre-training and Fine-tuning Framework for Natural Language Generation
- Cross-Lingual Natural Language Generation via Pre-Training (AAAI2020) [github]
- Multilingual Denoising Pre-training for Neural Machine Translation
- PLATO: Pre-trained Dialogue Generation Model with Discrete Latent Variable
- CG-BERT: Conditional Text Generation with BERT for Generalized Few-shot Intent Detection
- QURIOUS: Question Generation Pretraining for Text Generation
- Unsupervised Pre-training for Natural Language Generation: A Literature Review

Word segmentation, parsing, NER

- BERT Meets Chinese Word Segmentation
- Unified Multi-Criteria Chinese Word Segment
- Toward Fast and Accurate Neural Chinese Wo
- Establishing Strong Baselines for the New De
- Evaluating Contextualized Embeddings on 54
- NEZHA: Neural Contextualized Representatio
- Deep Contextualized Word Embeddings in Tr Revisited (EMNLP2019)
- Is POS Tagging Necessary or Even Helpful for
- Parsing as Pretraining (AAAI2020)
- Cross-Lingual BERT Transformation for Zero-
- Recursive Non-Autoregressive Graph-to-Grap
- Named Entity Recognition -- Is there a glass

Relation extraction

- Matching the Blanks: Distributional Similarity for Re
- BERT-Based Multi-Head Selection for Joint Entity-R
- Enriching Pre-trained Language Model with Entity I
- Span-based Joint Entity and Relation Extraction wit
- Fine-tune Bert for DocRED with Two-step Process
- Entity, Relation, and Event Extraction with Contextu
- Fine-tuning BERT for Joint Entity and Relation Extra
- Downstream Model Design of Pre-trained Languag
- Efficient long-distance relation extraction with DG-
- Robustly Pre-trained Neural Model for Direct Temp
- DARE: Data Augmented Relation Extraction with GP
- Improving Scholarly Knowledge Representation: Evaluating BERT-based Models for Scientific Relation Classification

rvised QA
0)
019)
es
2019)
ord Sense
s (KONVENS2019)
ummarization (ACL2019)
mmarization
entations (EMNLP2019 WS)

I bet it got my haters hella sick (hella sick)
Come and follow me, follow me with your signs up (uh)
I'm so firin', firin', boy, your time's up (uh)
Keep on and runnin' and runnin' until I catch up (uh)
How you dare? How you dare? How you dare? (Dare, ah)
[Chorus: Jungkook, RM & Suga] Another trophy, my hands carry 'em (hey)
Too many that I can't even count 'em (turn it up now)
Mic drop, mic drop

[Desiigner]

?

Thanks for an amazing semester!

[Too bad we can’t take a selfie]

Keep in touch!

benajibayassine@gmail.com

yb2235@columbia.edu

yassine.benajiba@dailight.ai

https://www.linkedin.com/in/yassine-benajiba-0180516/