From Sparse Modeling to Sparse Communication in NLP

André Martins

TALN Récital 2021

Our Amazing Team (December 2019, pre-COVID)

DeepSPIN

- ERC starting grant (2018–23)
- Topics: deep learning, structured prediction, NLP
- More details: https://deep-spin.github.io

Natural Language Processing over Time

Old times: rule-based systems

Mid-90s (“empirical revolution”): statistical methods (HMMs, PCFGs, IBM models)

Natural Language Processing over Time

Old times: rule-based systems

Mid-90s ("empirical revolution"): statistical methods (HMMs, PCFGs, IBM models)

2000+: structured prediction, linear models with rich features (CRFs, structured SVMs), feature engineering/selection

Natural Language Processing over Time

Old times: rule-based systems

Mid-90s ("empirical revolution"): statistical methods (HMMs, PCFGs, IBM models)

2000+: structured prediction, linear models with rich features (CRFs, structured SVMs), feature engineering/selection

Today: neural models, attention, transformers, ...

Natural Language Processing over Time

Old times: rule-based systems

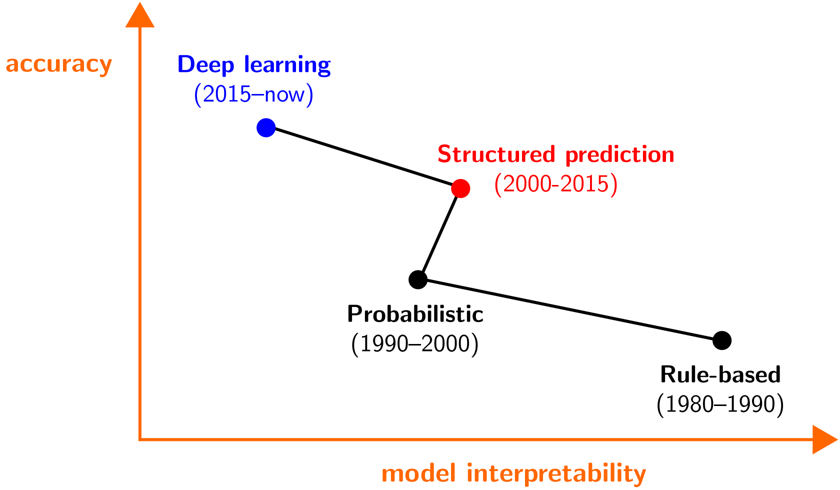
Mid-90s ("empirical revolution"): statistical methods (HMMs, PCFGs, IBM models)

2000+: structured prediction, linear models with rich features (CRFs, structured SVMs), feature engineering/selection

Today: neural models, attention, transformers, ...

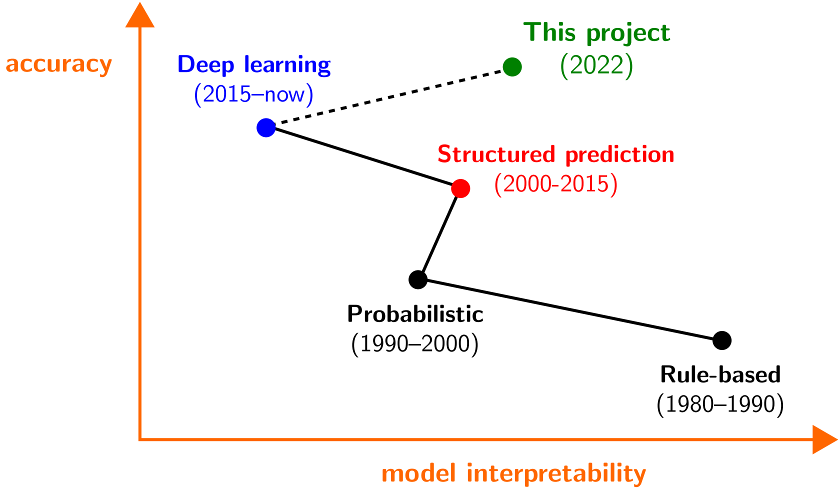
Structure, feature selection, and neural networks can go together!

Natural Language Processing over Time



(Slide from 2017)

Natural Language Processing over Time



(Slide from 2017)

Transformers Are Big Bulldozers

Very powerful, but highly overparametrized.

This Talk: Bet on Sparsity

What's inside a bulldozer? Can we redesign its components?

Sparsity can be useful:

- for interpretability
- for discovering linguistic structure
- for efficiency
- for generating.

From Sparse Modeling ...

- Mostly used with linear models, lots of work in the 2000s
- Main idea: embed a sparse regularizer (e.g. ℓ_1-norm) in the learning objective
- Irrelevant features get zero weight and can be discarded
- Extensions to structured sparsity (group-lasso, fused-lasso, etc.)

... to Sparse Communication:

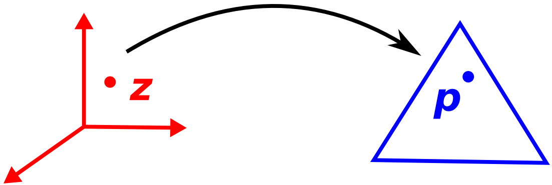
- Mostly used with neural networks, most work after 2015
- Main idea: sparse neuron activations (biological plausibility)
- Predictions are triggered by a few neurons only (input-dependent)
- Example: ReLUs, dropout, sparse attention mechanisms

This Talk

An inventory of transformations that capture **sparsity** and **structure**:

- All differentiable (efficient forward and backward propagation)
- Adaptively sparse
- Can be used at hidden or output layers
- Effective in many NLP tasks.

Building block:



Sparse transformations from the Euclidean space to the simplex $\triangle$.

Outline

Recap: Softmax and Argmax

Softmax exponentiates and normalizes:

$$\mathbf{softmax}(\boldsymbol{z}) = \frac{\exp(\boldsymbol{z})}{\sum_{k=1}^{K} \exp(z_k)}$$

- **Fully dense:** $\mathbf{softmax}(\boldsymbol{z}) > \mathbf{0}, \forall \boldsymbol{z}$
- Used both as a loss function (cross-entropy) and for attention.

Recap: Softmax and Argmax

Softmax exponentiates and normalizes:

$$\mathbf{softmax}(\boldsymbol{z}) = \frac{\exp(\boldsymbol{z})}{\sum_{k=1}^{K} \exp(z_k)}$$

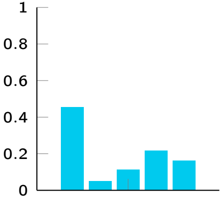
- **Fully dense:** $\mathbf{softmax}(\boldsymbol{z}) > \mathbf{0}, \forall \boldsymbol{z}$
- Used both as a loss function (cross-entropy) and for attention.

Argmax can be written as:

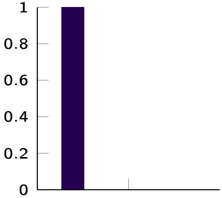
$$\begin{aligned} \mathbf{argmax}(\boldsymbol{z}) &:= \arg\max_{\boldsymbol{p} \in \triangle} \boldsymbol{z}^\top \boldsymbol{p} \\ &= \lim_{\tau \to 0^+} \mathbf{softmax}(\boldsymbol{z}/\tau) \qquad \text{(temperature trick)} \end{aligned}$$

- Retrieves a **one-hot vector** for the highest scored index.

softmax(z)

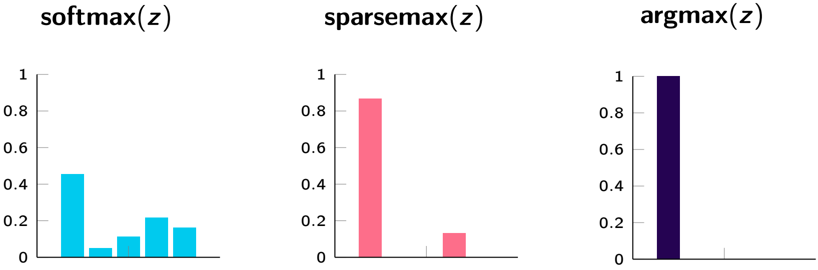


argmax(z)



(Same $\boldsymbol{z} = [1.0716, -1.1221, -0.3288, 0.3368, 0.0425]$)

- Argmax is an extreme case of sparsity, but it is **discontinuous.**
- Is there a **sparse** and **differentiable** alternative?



(Same $\boldsymbol{z} = [1.0716, -1.1221, -0.3288, 0.3368, 0.0425]$)

- Argmax is an extreme case of sparsity, but it is **discontinuous**.
- Is there a **sparse** and **differentiable** alternative?

Sparsemax (Martins & Astudillo, 2016, ICML)

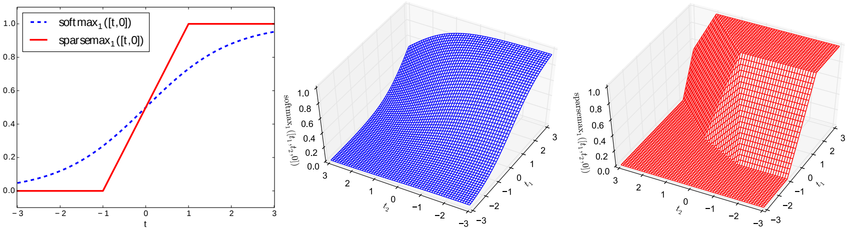
Euclidean projection of $\boldsymbol{z}$ onto the probability simplex $\triangle$:

$$\begin{aligned}\textbf{sparsemax}(\boldsymbol{z}) &:= \arg\min_{\boldsymbol{p}\in\triangle} \|\boldsymbol{p}-\boldsymbol{z}\|^2 \\ &= \arg\max_{\boldsymbol{p}\in\triangle} \boldsymbol{z}^\top\boldsymbol{p} - \frac{1}{2}\|\boldsymbol{p}\|^2.\end{aligned}$$

- Likely to hit the boundary of the simplex, in which case $\textbf{sparsemax}(\boldsymbol{z})$ becomes sparse (hence the name)
- End-to-end differentiable
- Forward pass: $O(K\log K)$ or $O(K)$, **(almost) as fast as softmax**
- Backprop: sublinear, **better than softmax!**

Sparsemax in 2D and 3D

(Martins & Astudillo, 2016, ICML)



- Sparsemax is piecewise linear, but asymptotically similar to softmax.

Ω-Regularized Argmax (Niculae & Blondel, 2017, NeurIPS)

For convex Ω, define the **Ω-regularized argmax transformation**:

$$\mathbf{argmax}_{\Omega}(\boldsymbol{z}) := \arg\max_{\boldsymbol{p}\in\triangle} \boldsymbol{z}^{\top}\boldsymbol{p} - \Omega(\boldsymbol{p})$$

- **Argmax** corresponds to **no regularization**, $\Omega \equiv 0$
- **Softmax** amounts to **entropic regularization**, $\Omega(\boldsymbol{p}) = \sum_{i=1}^{K} p_i \log p_i$
- **Sparsemax** amounts to **ℓ_2-regularization**, $\Omega(\boldsymbol{p}) = \frac{1}{2}\|\boldsymbol{p}\|^2$

Is there something in-between?

Entmax (Peters, Niculae & Martins, 2019, ACL)

Parametrized by $\alpha \geq 0$:

$$\Omega_\alpha(\boldsymbol{p}) := \begin{cases} \frac{1}{\alpha(\alpha-1)}\left(1 - \sum_{i=1}^{K} p_i^\alpha\right) & \text{if } \alpha \neq 1 \\ \sum_{i=1}^{K} p_i \log p_i & \text{if } \alpha = 1. \end{cases}$$

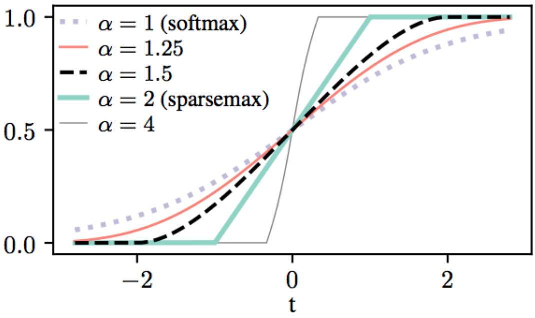
Related to **Tsallis generalized entropies** (Tsallis, 1988).

- **Argmax** corresponds to $\alpha \to \infty$
- **Softmax** amounts to $\alpha \to 1$
- **Sparsemax** amounts to $\alpha = 2$.

Key result: always sparse for $\alpha > 1$, sparsity increases with α

- Forward pass for general α can be done with a bissection algorithm
- Backward pass runs in sublinear time.

Entmax in 2D (Peters, Niculae & Martins, 2019, ACL)



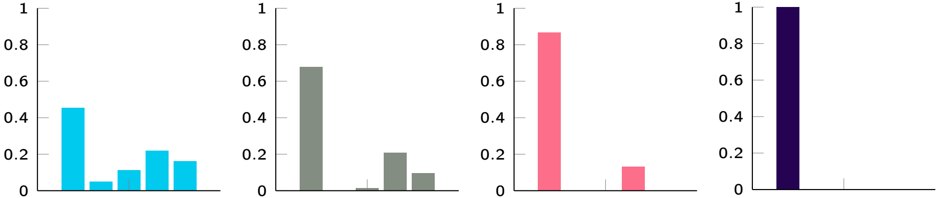
$\alpha = 1.5$ is a sweet spot!

- Efficient exact algorithm (nearly as fast as softmax), smooth, and good empirical performance.

Pytorch code: `https://github.com/deep-spin/entmax`

Sparse Transformations (Peters, Niculae & Martins, 2019, ACL)

$\alpha = 1$	$\alpha = 1.5$	$\alpha = 2$	$\alpha = \infty$
softmax(z)	**1.5-entmax**(z)	**sparsemax**(z)	**argmax**(z)

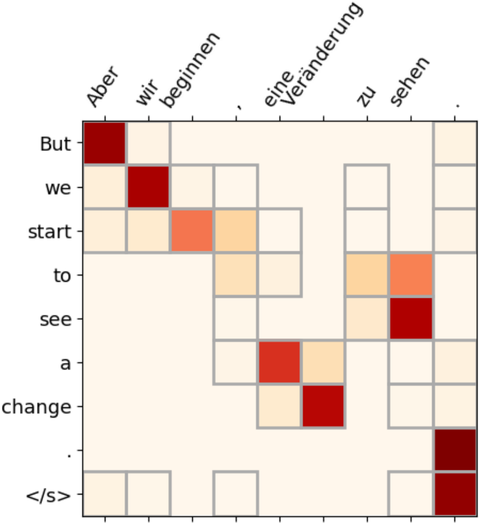


(Same $\boldsymbol{z} = [1.0716, -1.1221, -0.3288, 0.3368, 0.0425]$)

Example: Sparse Attention for Machine Translation

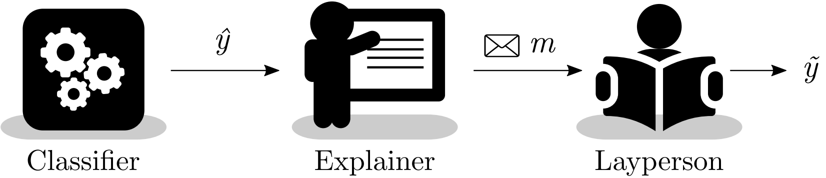
(Peters, Niculae & Martins, 2019, ACL)

- **Selects** source words when generating a target word (sparse alignments)
- Better interpretability
- Can also model fertility: **constrained** sparsemax (Malaviya, Ferreira & Martins, 2018, ACL)



Example: Sparse Attention for Explainability

(Treviso & Martins, 2020, BlackboxNLP)



- A classifier makes a prediction
- An "explainer" (embedded or not in the classifier) generates a sparse **message** that explains the classifier's decision
- The layperson receives the message and tries to guess the classifier's prediction (also called *simulatability*, *forward simulation/prediction*)
- Communication success rate: how often the two predictions match?

From Sparse Modeling to Sparse Communication

(Treviso & Martins, 2020, BlackboxNLP)

	Model interpretability	Prediction explainability
Wrappers	• Forward selection • Backward elimination (Kohavi & John, 1997)	• Input reduction (Feng et al., 2018) • Erasure (leave-one-out) (Li et al., 2016; Serrano & Smith, 2019) • LIME (Ribeiro et al., 2016)
Filters	• PMI (Church & Hanks, 1990) • recursive feature elimination (Guyon et al., 2002)	• Input gradient (Li et al., 2016) • LRP (Bach et al., 2015) • top-k softmax attention
Embedded	• ℓ_1-regularization (Tibshirani, 1996) • elastic net (Zou & Hastie, 2005)	• Stochastic attention (Xu et al., 2015; Lei et al., 2016; Bastings et al., 2019) • Sparse attention

Comparing Explainers

(Treviso & Martins, 2020, BlackboxNLP)

		SST		IMDB		AgNews		Yelp		SNLI	
Clf.	Explainer	CSR	ACC_L	CSR	ACC_L	CSR	ACC_L	CSR	ACC_L	CSR	ACC_L
C	Random	69.41	70.07	67.30	66.67	92.38	91.14	58.27	53.06	75.83	68.74
C	Erasure	80.12	81.22	92.17	88.72	97.31	95.41	78.72	68.90	77.88	70.04
C	Top-k gradient	79.35	79.24	86.30	83.93	96.49	94.86	70.54	62.86	76.74	69.40
C	Top-k softmax	84.18	82.43	93.06	89.46	**97.59**	95.61	81.00	70.18	78.66	71.00
C_{ent}	Top-k 1.5-entmax	**85.23**	**83.31**	93.32	89.60	97.29	**95.67**	82.20	70.78	80.23	73.39
C_{sp}	Top-k sparsemax	**85.23**	81.93	93.34	89.57	95.92	94.48	82.50	70.99	**82.89**	**74.76**
C_{ent}	Selec. 1.5-entmax	83.96	82.15	92.55	89.96	97.30	95.66	81.38	70.41	77.25	71.44
C_{sp}	Selec. sparsemax	**85.23**	81.93	93.24	89.66	95.92	94.48	83.55	71.60	82.04	73.46
C_{bern}	Bernoulli	82.37	78.42	91.66	86.13	96.91	94.43	84.93	66.89	76.81	69.65
C_{hk}	HardKuma	85.17	80.40	**94.72**	**90.16**	97.11	95.45	**87.39**	**71.64**	74.98	71.48

See paper for human experiments.

- In general, attention $>$ erasure $\gg$ gradient methods (in terms of CSR).

Questions

- Which α to choose?
- The bigger the α, the higher propensity to sparsity.
- What if we have many attention heads, and we don't know how sparse we want each one to be?
- Can we learn α from data?

Transformer (Vaswani et al., 2017)

Attention in three places:

- Self-attention in the encoder
- Self-attention in the decoder
- Contextual attention.

Multi-head attention: 6 layers, 8 attention heads (48 total).
Each head involves a query, a key, and a value matrix:

$$\bar{\mathbf{V}} = \mathbf{softmax}\left(\frac{\mathbf{QK}^\top}{\sqrt{d_k}}\right)\mathbf{V}.$$

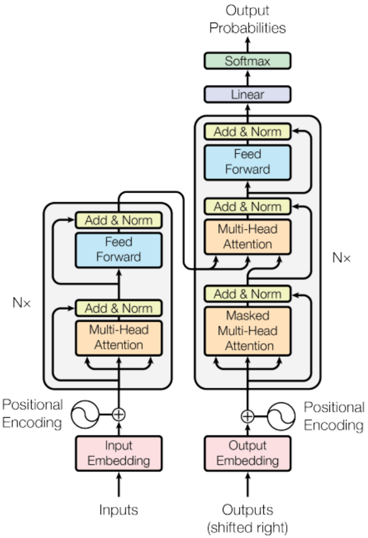


Figure 1: The Transformer - model architecture.

Adaptively Sparse Transformers

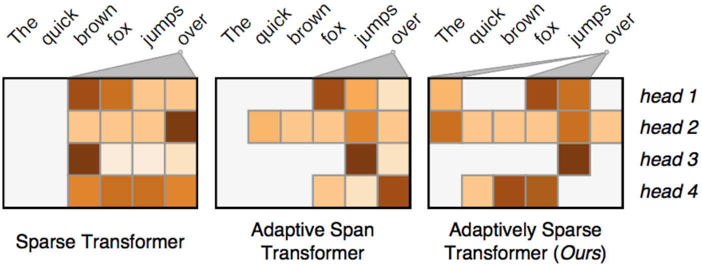
(Correia, Niculae & Martins, 2019, EMNLP)

Key idea: replace softmax in attention heads by α-entmax!

- Recall: α controls propensity to sparsity
- Learn each $\alpha \in [1, 2]$ **adaptively**!
- One α for each attention head and each layer.

Related Work: Other Sparse Transformers

(Child et al., 2019; Sukhbaatar et al., 2019)

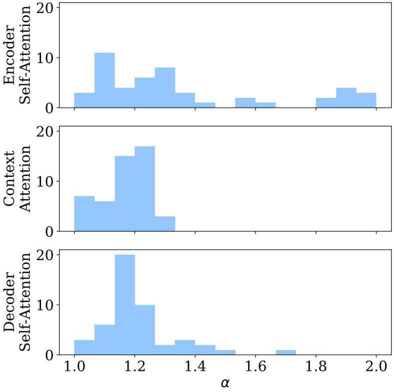


Our model allows **non-contiguous** attention for each head, learned **adaptively**.

Accuracies and Learned α

(Correia, Niculae & Martins, 2019, EMNLP)

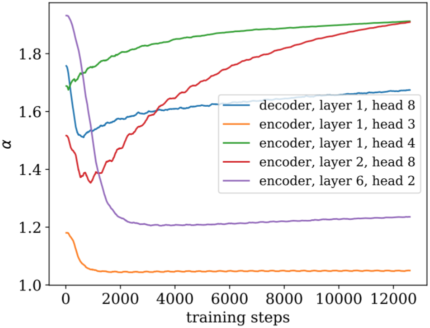
activation	DE→EN	JA→EN	RO→EN	EN→DE
softmax	29.79	21.57	32.70	26.02
1.5-entmax	29.83	**22.13**	**33.10**	25.89
α-entmax	**29.90**	21.74	32.89	**26.93**



Bimodal for the encoder, mostly unimodal for the decoder.

Trajectories of α During Training

(Correia, Niculae & Martins, 2019, EMNLP)

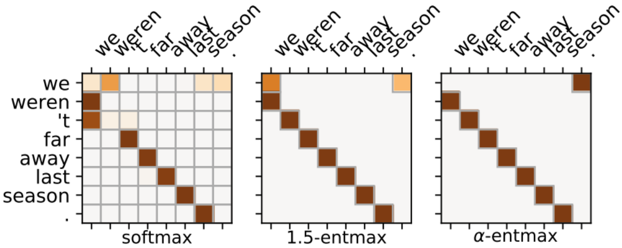


Most heads become denser in the beginning, before converging.

Dense attention more beneficial while the network is still uncertain, becomes sparser as the network learns.

Previous Position Head

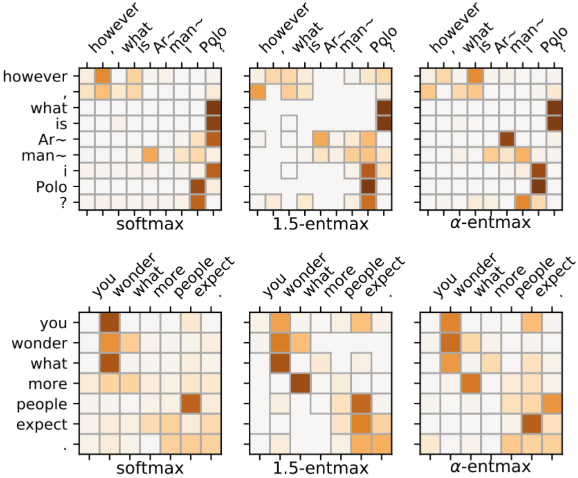
(Correia, Niculae & Martins, 2019, EMNLP)



(Learned $\alpha = 1.91$)

Interrogation-Detecting Head

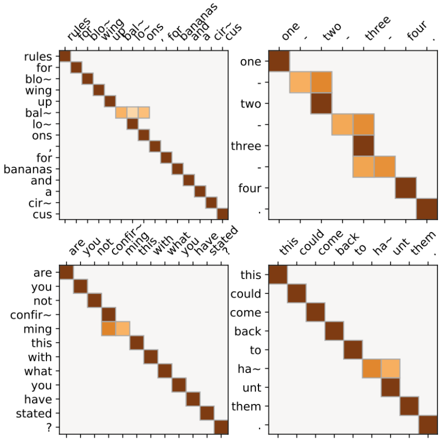
(Correia, Niculae & Martins, 2019, EMNLP)



(Learned $\alpha = 1.05$)

Subword-Merging Head

(Correia, Niculae & Martins, 2019, EMNLP)



(Learned $\alpha = 1.91$)

Other Related Transformations

Constrained softmax (Martins & Kreutzer, 2017, EMNLP),

Constrained sparsemax (Malaviya, Ferreira & Martins, 2018, ACL):

- Allows placing a **budget** on how much attention a word can receive
- Useful to model **fertility** in NMT

Fusedmax (Niculae & Blondel, 2017, NeurIPS):

- Can promote **structured sparsity** (e.g. contiguous words more likely to be selected together)

SparseMAP (Niculae, Martins, Blondel & Cardie, 2018, ICML):

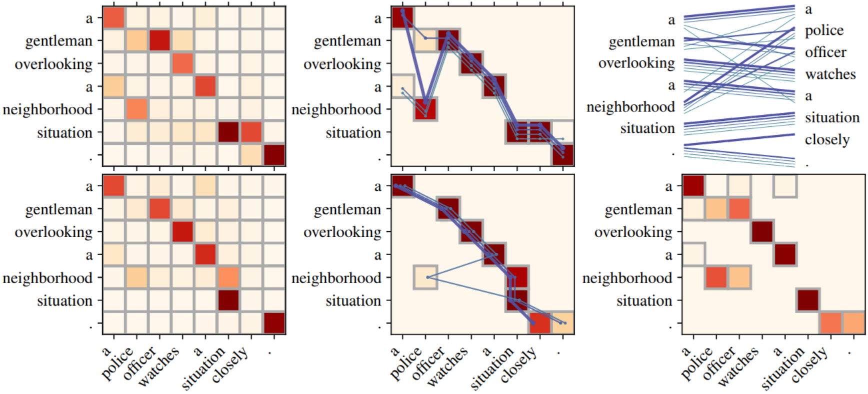
- Extends sparsemax to **sparse structured prediction**

SparseMAP

(Niculae et al., 2018, ICML)

- Generalizes sparsemax to **sparse structured prediction**
- Works both as output layer and hidden layer
- As hidden layer, similar to structured attention networks **(Kim et al., 2017)**, but sparse!
- Efficient forward/backprop requiring only an argmax (MAP) oracle!

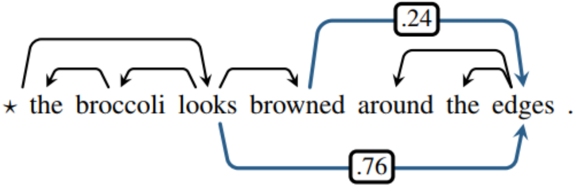
Example: Latent Structured Alignments in SNLI



(a) softmax (b) sequence (c) matching

Example: Dependency Parsing

- Suitable for capturing ambiguity in natural language!

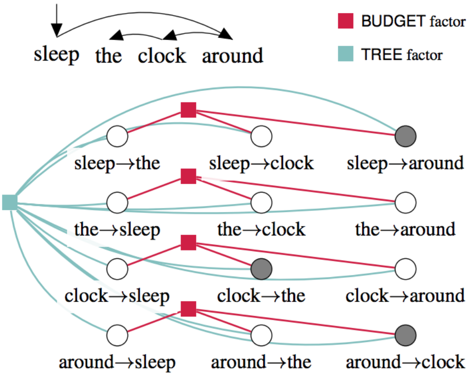


LP-SparseMAP (Niculae & Martins, 2020, ICML)

Extension of SparseMAP for latent factor graphs!

Can handle latent logic variables and constraints.

Example: latent syntax with valency constraints.




```
fg = TorchFactorGraph()
u = fg.variable_from(arc_scores)
fg.add(DepTree(u))
for k in range(n):
    fg.add(Budget(u[:, k], budget=5))
fg.solve()
```

Dynamic Computation Graphs and Exact Expectations

(Niculae, Martins & Cardie, 2018, EMNLP)

(Correia, Niculae, Aziz & Martins, 2020, NeurIPS)

When combinatorial structures are used as latent variables in a neural network, it may become intractable to compute expectations:

- Have to sum through exponentially many terms (one per structure)

SparseMAP offers a solution to this! Only a sparse subset of structure will have non-zero terms in the summation!

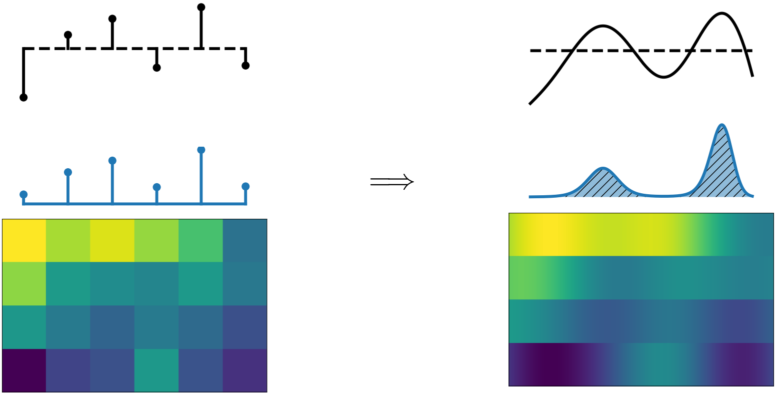
We do this for:

- Discriminative models with dynamic computation graphs
  (Niculae, Martins & Cardie, 2018, EMNLP)
- Generative models (combinatorial discrete VAEs)
  (Correia, Niculae, Aziz & Martins, 2020, NeurIPS)

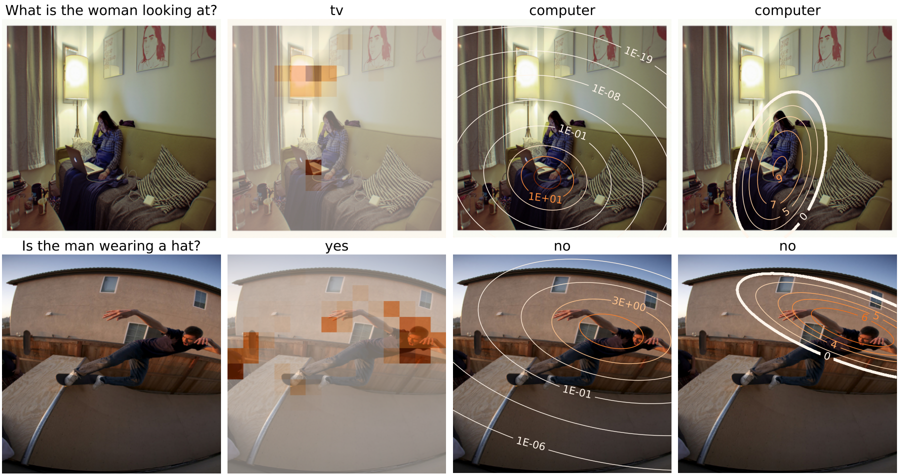
Sparse and Continuous Attention

(Martins, Farinhas, Treviso, Niculae, Aguiar & Figueiredo, 2020, NeurIPS)

- So far: attention over a **finite set** (words, pixel regions, etc.)
- We generalize attention to *arbitrary sets*, possibly continuous.



Example: Visual Question Answering



(original image) (discrete attention) (continuous softmax) (continuous sparsemax)

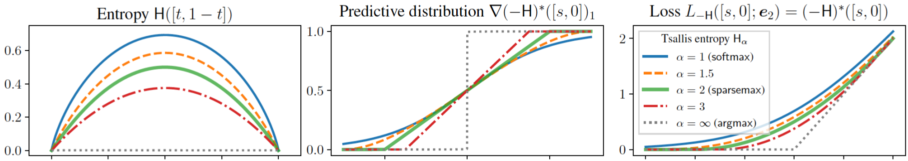
Outline

Entmax Losses

- Entmax can also be used as a loss in the **output layer** (to replace logistic/cross-entropy loss)
- However, not expressed as a log-likelihood (which could lead to $\log(0)$ problems due to sparsity)
- Instead, we build a entmax loss inspired by **Fenchel-Young losses**.

Entmax Transformations and Losses

(Blondel, Martins & Niculae, 2020, JMLR)



- For $\alpha > 1$, losses have **margins**
- Interesting case: 1.5-**entmax** (specialized forward pass algorithm).

Pytorch code: `https://github.com/deep-spin/entmax`

Sparse Sequence-to-Sequence (Peters, Niculae & Martins, 2019, ACL)

Key idea:

- Replace all instances of softmax by sparsemax or α**-entmax**.
- We consider both sparsity in the **attention mechanism** and sparsity in the **output layer**

Two tasks:

- Machine translation (word-based)
- Morphological inflection (character-based).

Training and Inference

Training

Minimize token-level loss:

$$\mathcal{L}(\boldsymbol{\theta}) = \sum_{(x,y)\in\mathcal{D}} \sum_{t=1}^{|y|} L(y_t, \boldsymbol{z}_t)$$

$$= \sum_{(x,y)\in\mathcal{D}} \sum_{t=1}^{|y|} -\log[\mathbf{softmax}(\boldsymbol{z}_t)]_{y_t}$$

Inference

Approximate MAP decoding:

$$\hat{y} = \mathop{\mathbf{argmax}}_{y\in V^*} p_{\boldsymbol{\theta}}(y \mid x)$$

$$= \mathop{\mathbf{argmax}}_{y\in V^*} \prod_{t=1}^{|y|} p_{\boldsymbol{\theta}}(y_t \mid x, y_{<t})$$

$$= \mathop{\mathbf{argmax}}_{y\in V^*} \prod_{t=1}^{|y|} \mathbf{softmax}(\boldsymbol{z}_t)_{y_t}$$

What's the Problem Here?

$$p_\theta(y \mid x) = \prod_{t=1}^{|y|} p_\theta(y_t \mid x, y_{<t}).$$

The chain rule favors short sequences!

What's the Problem Here?

$$p_{\boldsymbol{\theta}}(y \mid x) = \prod_{t=1}^{|y|} p_{\boldsymbol{\theta}}(y_t \mid x, y_{<t}).$$

The chain rule favors short sequences!

- Softmax $\rightarrow$ all strings have positive probability

What's the Problem Here?

$$p_{\theta}(y \mid x) = \prod_{t=1}^{|y|} p_{\theta}(y_t \mid x, y_{<t}).$$

The chain rule favors short sequences!

- Softmax $\rightarrow$ all strings have positive probability
- Often the **empty string** is the most likely sequence **(Stahlberg & Byrne, 2019)**

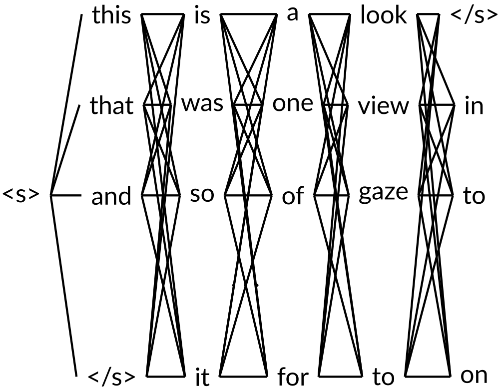
What's the Problem Here?

$$p_{\theta}(y \mid x) = \prod_{t=1}^{|y|} p_{\theta}(y_t \mid x, y_{<t}).$$

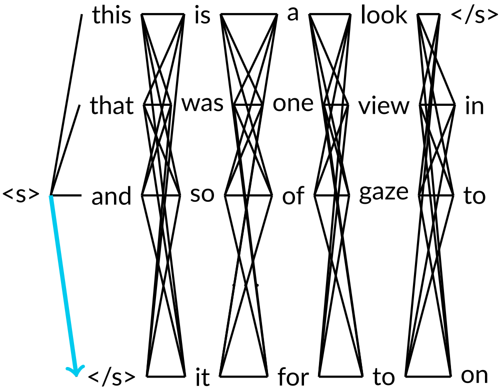
The chain rule favors short sequences!

- Softmax $\rightarrow$ all strings have positive probability
- Often the **empty string** is the most likely sequence **(Stahlberg & Byrne, 2019)**
- Beam search prunes it.

Cat Got Your Tongue?

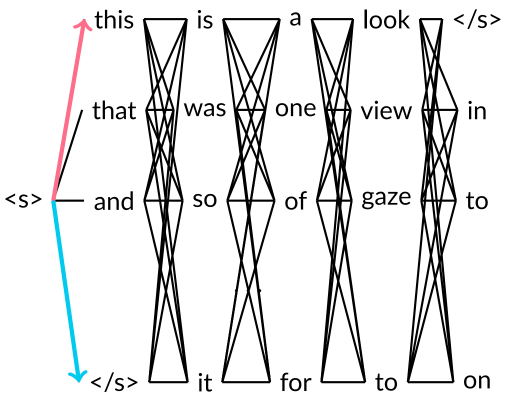


Cat Got Your Tongue?



$p(\cdot) = 10^{-4}$

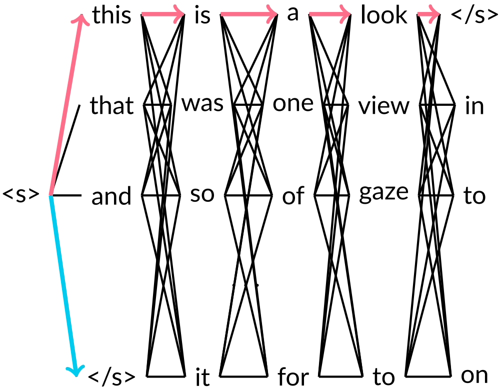
Cat Got Your Tongue?



$p(\cdot) = 10^{-1}$

$p(\cdot) = 10^{-4}$

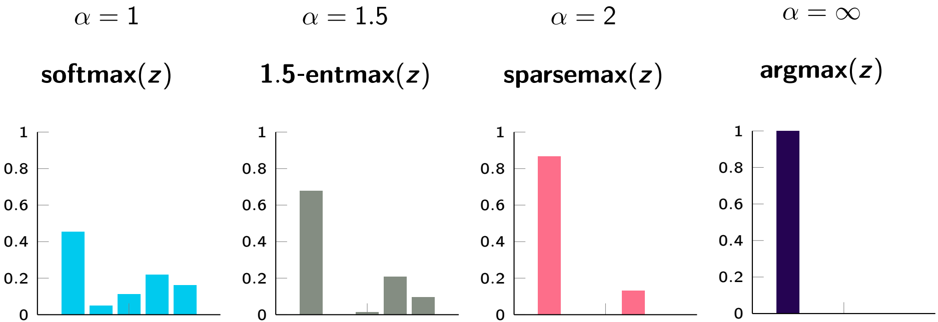
Cat Got Your Tongue?



$p(\cdot) = 10^{-5}$

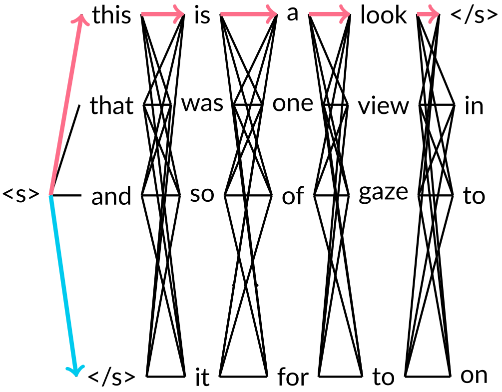
$p(\cdot) = 10^{-4}$

Can Entmax Save Us? (Peters, Niculae & Martins, 2019, ACL)



(Same $z = [1.0716, -1.1221, -0.3288, 0.3368, 0.0425]$)

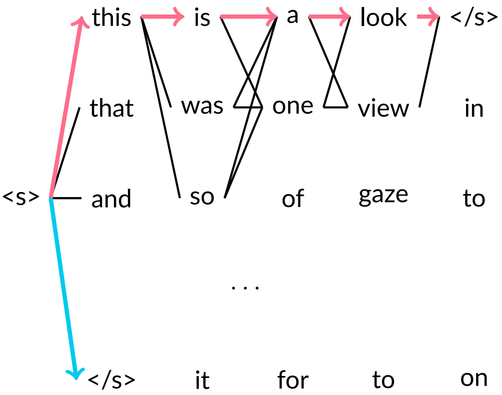
Cat Got Your Tongue? Maybe Not.



$p(\cdot) = 10^{-5}$

$p(\cdot) = 10^{-4}$

Cat Got Your Tongue? Maybe Not.



$p(\cdot) = 10^{-3}$

$p(\cdot) = 0$

Example: Machine Translation

(Peters et al., 2019, ACL)

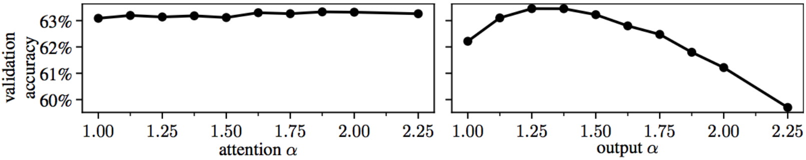
This	92.9%	**is another**	view	49.8%	**at**	95.7%	**the tree of life .**
So	5.9%		**look**	27.1%	on	5.9%	
And	1.3%		glimpse	19.9%	,	1.3%	
Here	<0.1%		kind	2.0%			
			looking	0.9%			
			way	0.2%			
			vision	<0.1%			
			gaze	<0.1%			

(Source: "Dies ist ein weiterer Blick auf den Baum des Lebens.")

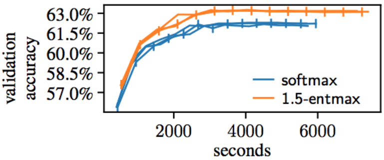
- Only a few words get non-zero probability at each time step
- Auto-completion when several words in a row have probability 1
- Useful for predictive translation.

Sparsity in Attention and in Output Layer

(Peters et al., 2019, ACL)

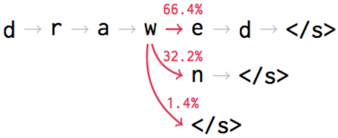


- Sparsity in the output leads to higher accuracy
- Sparse attention leads to more interpretable alignments.



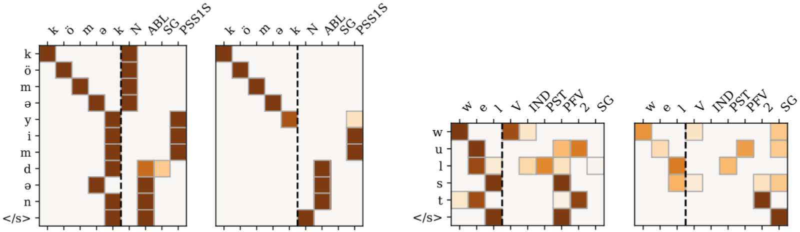
- 1.5-**entmax** attains better performance faster.

Example: Morphological Inflection



Only a few inflected words get nonzero probability.

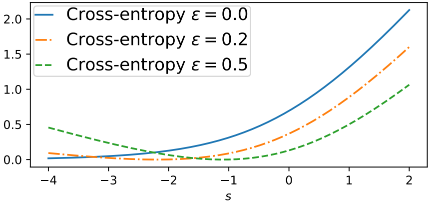
Variants with double/gated attention: **Peters & Martins (2019, SIGMORPHON)**.

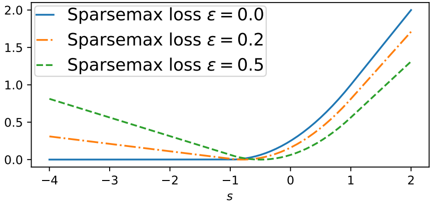


Entmax and Label Smoothing (Peters & Martins, 2021, NAACL)

- Sparse functions do mitigate the cat-got-your-tongue problem for MT
- **Fenchel-Young Label Smoothing**: FY loss with smoothed target instead of a one-hot label

$$L_{\Omega,\epsilon}(\boldsymbol{z}, \boldsymbol{e}_{y_t}) := L_\Omega(\boldsymbol{z}, (1-\epsilon)\boldsymbol{e}_{y_t} + \epsilon\boldsymbol{u}).$$





Entmax Loss and Label Smoothing (Peters & Martins, 2021, NAACL)

Grapheme-to-Phoneme (SIGMORPHON 2020 Task 1):

		Single		Ensemble	
α	ϵ	WER $\downarrow$	PER $\downarrow$	WER $\downarrow$	PER $\downarrow$
1	0	18.14	3.95	14.74	2.96
	0.15	15.55	3.09	13.87	2.77
1.5	0	15.25	3.05	13.79	2.77
	0.04	14.18	2.86	**13.47**	**2.69**

Machine Translation (BLEU scores; WMT14 is En-De):

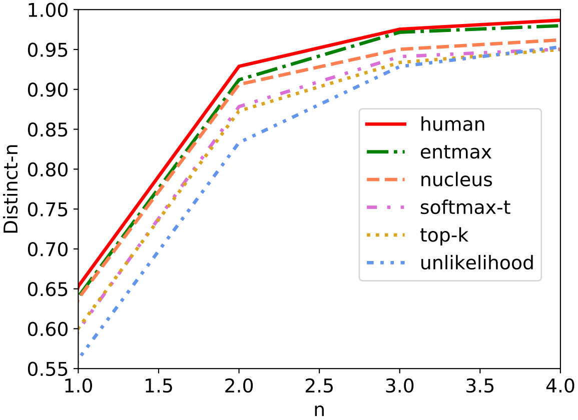
α	ϵ	De-En	En-De	Ja-En	En-Ja	Ro-En	En-Ro	WMT14
1	0	27.05	23.36	20.52	26.94	29.41	22.84	25.10
	> 0	27.72	24.24	20.99	27.28	30.03	23.15	25.21
1.5	0	**28.12**	24.03	21.23	**27.58**	30.27	**23.74**	**25.46**
	> 0	28.11	**24.36**	**21.34**	**27.58**	**30.37**	23.47	25.45

Entmax Sampling (Martins, Marinho & Martins, 2020, EMNLP)

Use the entmax loss for training language models.

At test time, sample from this sparse distribution.

Better quality with less repetitions than other methods:



Story Completion on WritingPrompts

Fine-tuned GPT-2; repetitions and off-topic text are *highlighted.*

Context:	*“Are they ready to go live?" Zuko asked. “Yup, in just a minute" Tomi replied as he leaned back in his reclining chair. This was a moment that the entire galaxy would see. After 10 years...*

Story Completion on WritingPrompts

Fine-tuned GPT-2; repetitions and off-topic text are *highlighted.*

Context:	*“Are they ready to go live?" Zuko asked. “Yup, in just a minute" Tomi replied as he leaned back in his reclining chair. This was a moment that the entire galaxy would see. After 10 years...*
Greedy:	... of study, the scientists had finally come to the conclusion that the universe was not a place for living. *The scientists* had been studying *the universe* for over a century. *They had been studying the universe for over a century.*

Story Completion on WritingPrompts

Fine-tuned GPT-2; repetitions and off-topic text are *highlighted.*

Context:	*“Are they ready to go live?" Zuko asked. “Yup, in just a minute" Tomi replied as he leaned back in his reclining chair. This was a moment that the entire galaxy would see. After 10 years...*
Top-k:	... of studying this phenomenon, the team had finally come up with a plan to *study* what the aliens were doing. *The team* would be looking for a way to create a space station. “And what would you call it?" *“What would you call the space station?"*

Story Completion on WritingPrompts

Fine-tuned GPT-2; repetitions and off-topic text are *highlighted.*

Context:	*“Are they ready to go live?" Zuko asked. “Yup, in just a minute" Tomi replied as he leaned back in his reclining chair. This was a moment that the entire galaxy would see. After 10 years...*
Nucleus:	... of diligent testing, *and some test run after that*, the results were well received. “they’re waiting for you, because *there’s no way they would.*” *Stowedly at a local wedding place, the two archaeologists spoke together.*

Story Completion on WritingPrompts

Fine-tuned GPT-2; repetitions and off-topic text are *highlighted.*

Context:	*“Are they ready to go live?" Zuko asked. “Yup, in just a minute" Tomi replied as he leaned back in his reclining chair. This was a moment that the entire galaxy would see. After 10 years...*
Entmax:	... of hibernation, Tomi was back. These last days had been a significant step forward in his mission. This time, Tomi was not alone. All the empires had aligned together and the world’s leadership began to openly support his mission.

Human Evaluation of Story Completion

	Fluency	Coherence	Engagement
Greedy	2.5	2.3	2.3
top-*k*	3.3	2.9	2.9
Nucleus	**3.5**	3.1	3.2
Unlikelihood	3.3	3.0	3.2
Entmax	**3.5**	**3.2**	**3.6**

Human Evaluation of Dialogue Generation

We followed the ConvAI2 challenge: 12 volunteers had 30 conversations each with models using the different sampling methods.

The model's personas were randomly selected from the PersonaChat validation set.

	Fluency	Consistency	Engagement
Greedy	**4.1**	3.0	2.5
top-k	**4.0**	3.2	3.3
Nucleus	**4.1**	3.4	3.3
Entmax	**4.1**	**3.6**	**3.9**

Outline

Conclusions

- Transformations from real numbers to distributions are ubiquitous
- We introduced new transformations that handle **sparsity**, **constraints**, and **structure**
- All are differentiable and their gradients are efficient to compute
- Can be used as hidden layers or as output layers
- The sparsity can be adaptive
- Encouraging results in NMT and other tasks
- Sparse communication potentially useful as a path for explainability.

Thank You!

DeepSPIN ("Deep Structured Prediction in NLP")

- ERC starting grant, started in 2018
- Topics: deep learning, structured prediction, NLP
- More details: https://deep-spin.github.io

Unbabel
instituto de telecomunicações
TÉCNICO LISBOA

References I

Bach, S., Binder, A., Montavon, G., Klauschen, F., Müller, K.-R., & Samek, W. (2015). On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PloS one*, *10*(7), 130–140.

Bastings, J., Aziz, W., & Titov, I. (2019). Interpretable neural predictions with differentiable binary variables. In *Proc. ACL*.

Blondel, M., Martins, A. F. T., & Niculae, V. (2020). Learning with fenchel-young losses. *Journal of Machine Learning Research*, *21*(35), 1–69.

Child, R., Gray, S., Radford, A., & Sutskever, I. (2019). Generating long sequences with sparse transformers. *arXiv preprint arXiv:1904.10509*.

Church, K. W. & Hanks, P. (1990). Word association norms, mutual information, and lexicography. *Computational linguistics*, *16*(1), 22–29.

Correia, G., Niculae, V., Aziz, W., & Martins, A. (2020). Efficient marginalization of discrete and structured latent variables via sparsity. *Advances in Neural Information Processing Systems*, *33*.

Correia, G., Niculae, V., & Martins, A. F. T. (2019). Adaptively sparse transformers. In *Proceedings of the Empirical Methods for Natural Language Processing*.

Feng, S., Wallace, E., Grissom II, A., Iyyer, M., Rodriguez, P., & Boyd-Graber, J. (2018). Pathologies of neural models make interpretations difficult. In *Proc. EMNLP*, (pp. 3719–3728).

Guyon, I., Weston, J., Barnhill, S., & Vapnik, V. (2002). Gene selection for cancer classification using support vector machines. *Machine learning*, *46*(1-3), 389–422.

References II

Kim, Y., Denton, C., Hoang, L., & Rush, A. M. (2017). Structured attention networks. *arXiv preprint arXiv:1702.00887*.

Kohavi, R. & John, G. H. (1997). Wrappers for feature subset selection. *Artificial intelligence*, *97*(1-2), 273–324.

Lei, T., Barzilay, R., & Jaakkola, T. (2016). Rationalizing neural predictions. In *Proc. EMNLP*, (pp. 107–117).

Li, J., Chen, X., Hovy, E., & Jurafsky, D. (2016). Visualizing and understanding neural models in nlp. In *Proc. NAACL-HLT*, (pp. 681–691).

Li, J., Monroe, W., & Jurafsky, D. (2016). Understanding neural networks through representation erasure. *arXiv preprint arXiv:1612.08220*.

Malaviya, C., Ferreira, P., & Martins, A. F. T. (2018). Sparse and Constrained Attention for Neural Machine Translation. In *Proc. of the Annual Meeting of the Association for Computation Linguistics*.

Martins, A., Farinhas, A., Treviso, M., Niculae, V., Aguiar, P., & Figueiredo, M. (2020). Sparse and continuous attention mechanisms. *Advances in Neural Information Processing Systems*, *33*.

Martins, A. F. T. & Astudillo, R. (2016). From Softmax to Sparsemax: A Sparse Model of Attention and Multi-Label Classification. In *Proc. of the International Conference on Machine Learning*.

Martins, A. F. T. & Kreutzer, J. (2017). Fully differentiable neural easy-first taggers. In *Proc. of Empirical Methods for Natural Language Processing*.

Martins, P. H., Marinho, Z., & Martins, A. F. T. (2020). Sparse text generation. In *Proc. of EMNLP*.

Niculae, V. & Blondel, M. (2017). A regularized framework for sparse and structured neural attention. *arXiv preprint arXiv:1705.07704*.

References III

Niculae, V., Martins, A. F., & Cardie, C. (2018). Towards dynamic computation graphs via sparse latent structure. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, (pp. 905–911).

Niculae, V. & Martins, A. F. T. (2020). Lp-sparsemap: Differentiable relaxed optimization for sparse structured prediction. In *Proc. of ICML*.

Niculae, V., Martins, A. F. T., Blondel, M., & Cardie, C. (2018). SparseMAP: Differentiable Sparse Structured Inference. In *Proc. of the International Conference on Machine Learning*.

Peters, B. & Martins, A. F. (2019). It–ist at the sigmorphon 2019 shared task: Sparse two-headed models for inflection. In *Proceedings of the 16th Workshop on Computational Research in Phonetics, Phonology, and Morphology*, (pp. 50–56).

Peters, B. & Martins, A. F. (2021). Smoothing and shrinking the sparse seq2seq search space. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, (pp. 2642–2654).

Peters, B., Niculae, V., & Martins, A. F. T. (2019). Sparse sequence-to-sequence models. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*.

Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). Why should i trust you?: Explaining the predictions of any classifier. In *Proc. ACM SIGKDD*, (pp. 1135–1144). ACM.

Serrano, S. & Smith, N. A. (2019). Is attention interpretable? In *Proc. ACL*.

Stahlberg, F. & Byrne, B. (2019). On nmt search errors and model errors: Cat got your tongue? In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, (pp. 3347–3353).

References IV

Sukhbaatar, S., Grave, E., Bojanowski, P., & Joulin, A. (2019). Adaptive attention span in transformers. *arXiv preprint arXiv:1905.07799*.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society*, *58*(1), 267–288.

Treviso, M. V. & Martins, A. F. T. (2020). The explanation gamae: Towards prediction explainability through sparse communication. In *Proc. of EMNLP Workshop on Analyzing and Interpreting Neural Networks for NLP (BlackboxNLP)*.

Tsallis, C. (1988). Possible generalization of boltzmann-gibbs statistics. *Journal of Statistical Physics*.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L. u., & Polosukhin, I. (2017). Attention is all you need. In *Advances in Neural Information Processing Systems* (pp. 5998–6008).

Xu, K., Ba, J., Kiros, R., Cho, K., Courville, A., Salakhudinov, R., Zemel, R., & Bengio, Y. (2015). Show, attend and tell: Neural image caption generation with visual attention. In *Proc. ICLR*, (pp. 2048–2057).

Zou, H. & Hastie, T. (2005). Regularization and variable selection via the elastic net. *Journal of the royal statistical society*, *67*(2), 301–320.