Language Model Prior for Low-Resource Neural Machine Translation

Christos Baziotis, Barry Haddow, Alexandra Birch

Institute for Language, Cognition and Computation
School of Informatics, University of Edinburgh
Edinburgh, UK

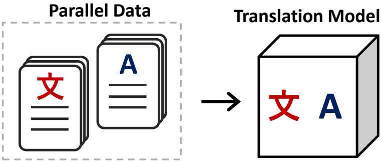
EMNLP 2020

Introduction

Neural Machine Translation

$$\hat{\boldsymbol{y}} = \arg\max_{\boldsymbol{y}} \log p(\boldsymbol{y}|\boldsymbol{x})$$

Modeling directly $p(\boldsymbol{y}|\boldsymbol{x})$ requires **large** amounts of parallel data.

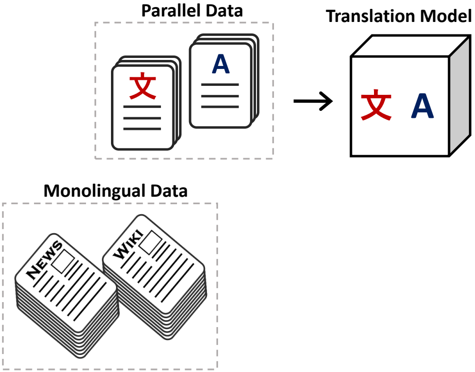


Introduction

Neural Machine Translation

$$\hat{\boldsymbol{y}} = \arg\max_{\boldsymbol{y}} \log p(\boldsymbol{y}|\boldsymbol{x})$$

Modeling directly $p(\boldsymbol{y}|\boldsymbol{x})$ requires **large** amounts of parallel data.

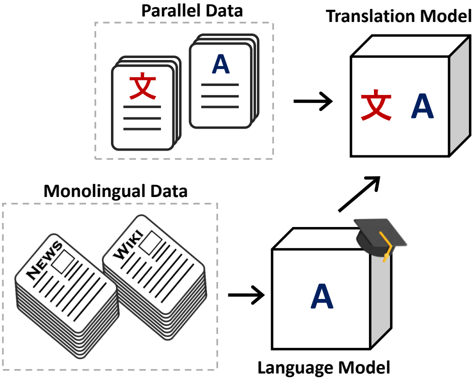


Introduction

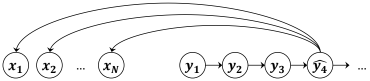
Neural Machine Translation

$$\hat{\boldsymbol{y}} = \arg\max_{\boldsymbol{y}} \log p(\boldsymbol{y}|\boldsymbol{x})$$

Modeling directly $p(\boldsymbol{y}|\boldsymbol{x})$ requires **large** amounts of parallel data.

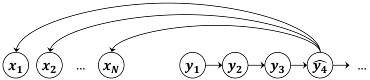


Prior Work



Noisy Channel

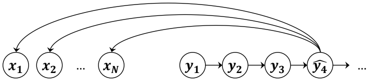
It models the "reverse translation probability" $p(\boldsymbol{y}|\boldsymbol{x}) \propto p_{\mathrm{TM}}(\boldsymbol{x}|\boldsymbol{y}) \times p_{\mathrm{LM}}(\boldsymbol{y})$



Noisy Channel

It models the "reverse translation probability" $p(\boldsymbol{y}|\boldsymbol{x}) \propto p_{\mathrm{TM}}(\boldsymbol{x}|\boldsymbol{y}) \times p_{\mathrm{LM}}(\boldsymbol{y})$

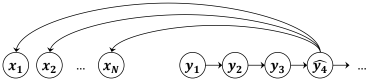
Prior Work



Noisy Channel

It models the "reverse translation probability" $p(\boldsymbol{y}|\boldsymbol{x}) \propto p_{\text{TM}}(\boldsymbol{x}|\boldsymbol{y}) \times p_{\text{LM}}(\boldsymbol{y})$

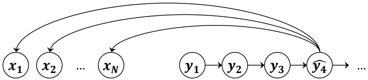
Prior Work



Noisy Channel

It models the "reverse translation probability" $p(\boldsymbol{y}|\boldsymbol{x}) \propto p_{\mathrm{TM}}(\boldsymbol{x}|\boldsymbol{y}) \times p_{\mathrm{LM}}(\boldsymbol{y})$

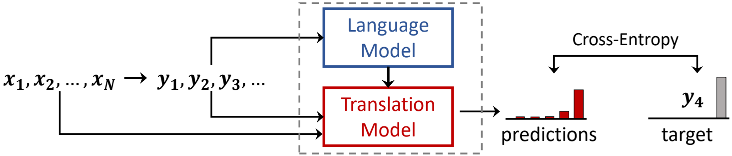
Prior Work

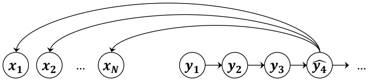


Noisy Channel

It models the "reverse translation probability" $p(\boldsymbol{y}|\boldsymbol{x}) \propto p_{\mathrm{TM}}(\boldsymbol{x}|\boldsymbol{y}) \times p_{\mathrm{LM}}(\boldsymbol{y})$

Language Model Fusion

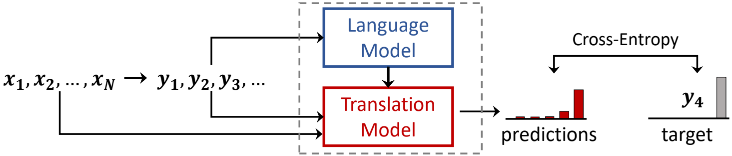




Noisy Channel

It models the "reverse translation probability" $p(\boldsymbol{y}|\boldsymbol{x}) \propto p_{\mathrm{TM}}(\boldsymbol{x}|\boldsymbol{y}) \times p_{\mathrm{LM}}(\boldsymbol{y})$

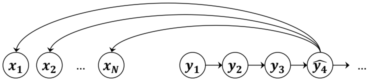
Language Model Fusion



Feature-based: incorporate the *representations* of the LM

- deep-fusion (Gulcehre et al., 2015), cold-fusion (Sriram et al., 2018)

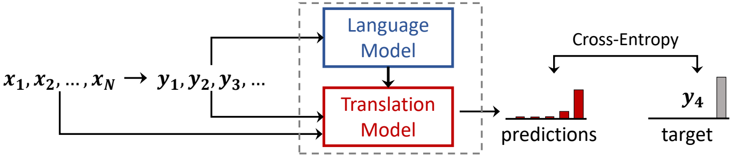
Prior Work



Noisy Channel

It models the "reverse translation probability" $p(\boldsymbol{y}|\boldsymbol{x}) \propto p_{\text{TM}}(\boldsymbol{x}|\boldsymbol{y}) \times p_{\text{LM}}(\boldsymbol{y})$

Language Model Fusion



Feature-based: incorporate the *representations* of the LM

- deep-fusion (Gulcehre et al., 2015), cold-fusion (Sriram et al., 2018)

Probability Interpolation: combine the *probabilities* of the TM & the LM

- shallow-fusion (Gulcehre et al., 2015), simple-fusion (Stahlberg et al., 2018)

Limitations of Prior Work

1. **Computational overhead**, as the LM is also required for decoding

Limitations of Prior Work

1. **Computational overhead**, as the LM is also required for decoding
2. **Unwanted LM-TM interference** in probability interpolation

Limitations of Prior Work

1. **Computational overhead**, as the LM is also required for decoding
2. **Unwanted LM-TM interference** in probability interpolation

$$p(\boldsymbol{y}_t) = \text{softmax}(\log p_{\text{TM}}(\boldsymbol{y}_t|\boldsymbol{y}_{<t}, \boldsymbol{x}) + \log p_{\text{LM}}(\boldsymbol{y}_t|\boldsymbol{y}_{<t})) \quad \text{POSTNORM}$$

(simple-fusion; Stahlberg et al., 2018)

DE: die Republikaner im Kongress drängen auf eine umfassendere Neufassung der Ozonregeln.

EN: Republicans → in → Congress → are → pushing → for → a → <u>broader</u> → rewrite → of → the → ozone → rules.

~~more~~

TM		×	LM		=		
broader	34.3%		new	8.9%		more	44.8%
wider	22.4%		repeal	7.1%		wider	31.8%
larger	7.2%		bill	3.7%		larger	11.8%
...			...			...	

Real example of POSTNORM from DE→EN newstest2018

More visualizations at http://data.statmt.org/cbaziotis/projects/lm-prior/analysis/

Limitations of Prior Work

1. **Computational overhead**, as the LM is also required for decoding
2. **Unwanted LM-TM interference** in probability interpolation

$$p(\boldsymbol{y}_t) = \text{softmax}(\log p_{\text{TM}}(\boldsymbol{y}_t|\boldsymbol{y}_{<t}, \boldsymbol{x}) + \log p_{\text{LM}}(\boldsymbol{y}_t|\boldsymbol{y}_{<t})) \quad \text{POSTNORM}$$

(simple-fusion; Stahlberg et al., 2018)

DE: die Republikaner im Kongress drängen auf eine umfassendere Neufassung der Ozonregeln.

EN: Republicans → in → Congress → are → pushing → for → a → broader → rewrite → of → the → ozone → rules.

~~more~~

TM			LM			
broader	34.3%	×	new	8.9%	=	more 44.8%
wider	22.4%		repeal	7.1%		wider 31.8%
larger	7.2%		bill	3.7%		larger 11.8%
...			...			...

Real example of POSTNORM from DE→EN newstest2018

More visualizations at http://data.statmt.org/cbaziotis/projects/lm-prior/analysis/

Limitations of Prior Work

1. **Computational overhead**, as the LM is also required for decoding
2. **Unwanted LM-TM interference** in probability interpolation

$$p(\boldsymbol{y}_t) = \text{softmax}(\log p_{\text{TM}}(\boldsymbol{y}_t|\boldsymbol{y}_{<t}, \boldsymbol{x}) + \log p_{\text{LM}}(\boldsymbol{y}_t|\boldsymbol{y}_{<t})) \quad \text{POSTNORM}$$

(simple-fusion; Stahlberg et al., 2018)

DE: die Republikaner im Kongress drängen auf eine umfassendere Neufassung der Ozonregeln.

EN: Republicans → in → Congress → are → pushing → for → a → broader → rewrite → of → the → ozone → rules.

~~more~~

TM			LM				
broader	34.3%		new	8.9%		more	44.8%
wider	22.4%	×	repeal	7.1%	=	wider	31.8%
larger	7.2%		bill	3.7%		larger	11.8%
…			…			…	

Real example of POSTNORM from DE→EN newstest2018

More visualizations at http://data.statmt.org/cbaziotis/projects/lm-prior/analysis/

Limitations of Prior Work

1. **Computational overhead**, as the LM is also required for decoding
2. **Unwanted LM-TM interference** in probability interpolation

$$p(\boldsymbol{y}_t) = \text{softmax}(\log p_{\text{TM}}(\boldsymbol{y}_t|\boldsymbol{y}_{<t}, \boldsymbol{x}) + \log p_{\text{LM}}(\boldsymbol{y}_t|\boldsymbol{y}_{<t})) \quad \text{POSTNORM}$$

(simple-fusion; Stahlberg et al., 2018)

DE: die Republikaner im Kongress drängen auf eine umfassendere Neufassung der Ozonregeln.

EN: Republicans → in → Congress → are → pushing → for → a → broader → rewrite → of → the → ozone → rules.

~~more~~

TM			LM				
broader	34.3%		new	8.9%		more	44.8%
wider	22.4%	x	repeal	7.1%	=	wider	31.8%
larger	7.2%		bill	3.7%		larger	11.8%
...			...			...	

Real example of POSTNORM from DE→EN newstest2018

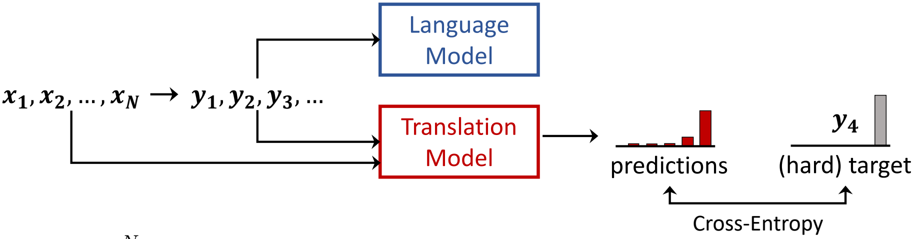
More visualizations at http://data.statmt.org/cbaziotis/projects/lm-prior/analysis/

Language Model Prior

We regularize the TM, by using the LM as **prior** over the decoder

Language Model Prior

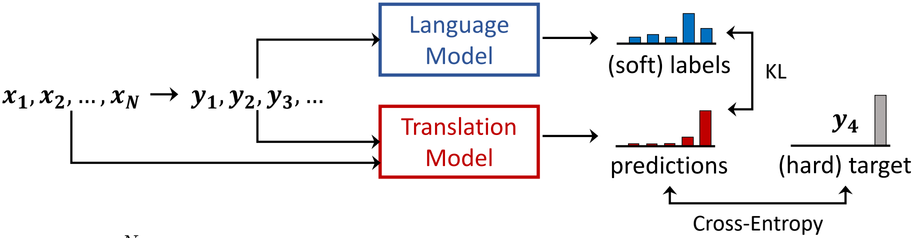
We regularize the TM, by using the LM as **prior** over the decoder



$$\mathcal{L} = \sum_{t=1}^{N} \underbrace{- \log p_{\text{TM}}(\boldsymbol{y}_t | \boldsymbol{y}_{<t}, \boldsymbol{x})}_{\text{translation}}$$

Language Model Prior

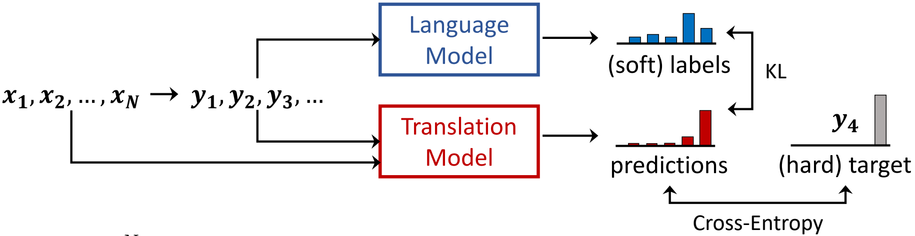
We regularize the TM, by using the LM as **prior** over the decoder



$$\mathcal{L} = \sum_{t=1}^{N} \underbrace{-\log p_{\text{TM}}(\boldsymbol{y}_t|\boldsymbol{y}_{<t}, \boldsymbol{x})}_{\text{translation}} + \lambda \underbrace{D_{\text{KL}}(p_{\text{LM}}(\boldsymbol{y}_t|\boldsymbol{y}_{<t}) \parallel p_{\text{TM}}(\boldsymbol{y}_t|\boldsymbol{y}_{<t}, \boldsymbol{x}))}_{\text{prior}}$$

Language Model Prior

We regularize the TM, by using the LM as **prior** over the decoder



$$\mathcal{L} = \sum_{t=1}^{N} \underbrace{-\log p_{\text{TM}}(\boldsymbol{y}_t | \boldsymbol{y}_{<t}, \boldsymbol{x})}_{\text{translation}} + \lambda \underbrace{D_{\text{KL}}(p_{\text{LM}}(\boldsymbol{y}_t | \boldsymbol{y}_{<t}) \parallel p_{\text{TM}}(\boldsymbol{y}_t | \boldsymbol{y}_{<t}, \boldsymbol{x}))}_{\text{prior}}$$

+ The TM can **overrule** the LM when needed
+ **Easier** than priors on neural network weights
+ The LM is **not required** during decoding

Relation to Other Methods with Soft Targets

Knowledge Distillation

Also uses **soft targets** from a teacher model, but needs **parallel data**

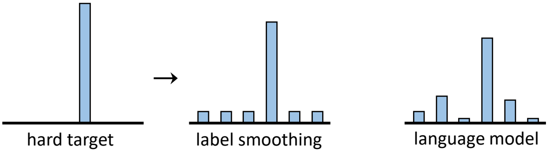
Relation to Other Methods with Soft Targets

Knowledge Distillation

Also uses **soft targets** from a teacher model, but needs **parallel data**

Label Smoothing

Converts the one-hot into **soft targets** to penalize **over-confidence**



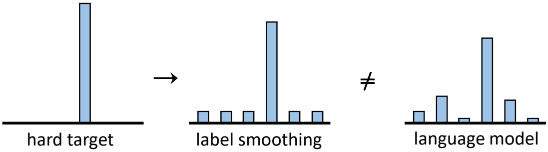
Relation to Other Methods with Soft Targets

Knowledge Distillation

Also uses **soft targets** from a teacher model, but needs **parallel data**

Label Smoothing

Converts the one-hot into **soft targets** to penalize **over-confidence**



- Assigns **equal** probability to all the incorrect classes, unlike the LM.

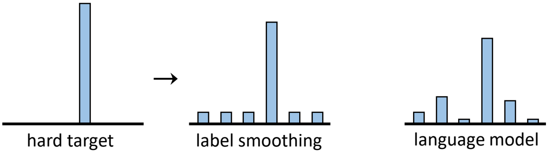
Relation to Other Methods with Soft Targets

Knowledge Distillation

Also uses **soft targets** from a teacher model, but needs **parallel data**

Label Smoothing

Converts the one-hot into **soft targets** to penalize **over-confidence**



$$\mathcal{L} = \sum_{t=1}^{N} \underbrace{-\log p_{\mathrm{TM}}(\boldsymbol{y}_t|\boldsymbol{y}_{<t},\boldsymbol{x})}_{\text{translation}} + \lambda \underbrace{D_{\mathrm{KL}}(p_{\mathrm{LM}}(\boldsymbol{y}_t|\boldsymbol{y}_{<t}) \parallel p_{\mathrm{TM}}(\boldsymbol{y}_t|\boldsymbol{y}_{<t},\boldsymbol{x}))}_{\text{prior}}$$

- Assigns **equal** probability to all the incorrect classes, unlike the LM.
- **Orthogonal** to posterior regularization (LM-prior)

Experimental Setup

Language Pair	Sentences	Corpus
English-German	275K	News Commentary v13
English-Turkish	190K	SETIMES

Table: Parallel data from WMT-2018

Language	Small (3M)	Large (30M)	Corpus
English	✓	✓	News Crawl 2016
German	✓	✓	News Crawl 2016
Turkish	✓	–	News Crawl 2010-2018

Table: Monolingual data from News Crawl articles

Models

- Architecture: Transformer (LMs and TMs)
- Vocabulary: 16K symbols (sentencepiece)

Translation Results

Method	DE→EN	EN→DE	TR→EN	EN→TR
Base	26.6	25.6	16.6	11.2
Shallow-fusion	27.8	26.0	17.3	11.5
POSTNORM+ LS	26.4	23.3	16.0	11.0
Base + LS	28.4	27.3	18.4	12.6
Base + Prior	**30.2**	**29.1**	**19.5**	**13.8**

Table: BLEU scores of each NMT method (mean 3 runs)

Translation Results

Method	**DE→EN**	**EN→DE**	**TR→EN**	**EN→TR**
Base	26.6	25.6	16.6	11.2
Shallow-fusion	27.8	26.0	17.3	11.5
POSTNORM+ LS	26.4	23.3	16.0	11.0
Base + LS	28.4	27.3	18.4	12.6
Base + Prior	**30.2**	**29.1**	**19.5**	**13.8**

Table: BLEU scores of each NMT method (mean 3 runs)

Translation Results

Method	DE→EN	EN→DE	TR→EN	EN→TR
Base	26.6	25.6	16.6	11.2
Shallow-fusion	27.8	26.0	17.3	11.5
POSTNORM+ LS	26.4	23.3	16.0	11.0
Base + LS	28.4	27.3	18.4	12.6
Base + Prior	**30.2**	**29.1**	**19.5**	**13.8**

Table: BLEU scores of each NMT method (mean 3 runs)

Translation Results

Method	DE→EN	EN→DE	TR→EN	EN→TR
Base	26.6	25.6	16.6	11.2
Shallow-fusion	27.8	26.0	17.3	11.5
POSTNORM+ LS	26.4	23.3	16.0	11.0
Base + LS	28.4	27.3	18.4	12.6
Base + Prior	**30.2**	**29.1**	**19.5**	**13.8**

Table: BLEU scores of each NMT method (mean 3 runs)

Translation Results

Method	DE→EN	EN→DE	TR→EN	EN→TR
Base	26.6	25.6	16.6	11.2
Shallow-fusion	27.8	26.0	17.3	11.5
POSTNORM+ LS	26.4	23.3	16.0	11.0
Base + LS	28.4	27.3	18.4	12.6
Base + Prior	**30.2** (+1.8)	**29.1** (+1.8)	**19.5** (+1.1)	**13.8** (+1.2)

Table: BLEU scores of each NMT method (mean 3 runs)

Translation Results

Method	DE→EN	EN→DE	TR→EN	EN→TR
Base	26.6	25.6	16.6	11.2
Shallow-fusion	27.8	26.0	17.3	11.5
POSTNORM+ LS	26.4	23.3	16.0	11.0
Base + LS	28.4	27.3	18.4	12.6
Base + Prior	**30.2**	**29.1**	**19.5**	**13.8**
Base + Prior + LS	30.3 (+0.1)	29.7 (+0.6)	19.5	14.1 (+0.3)

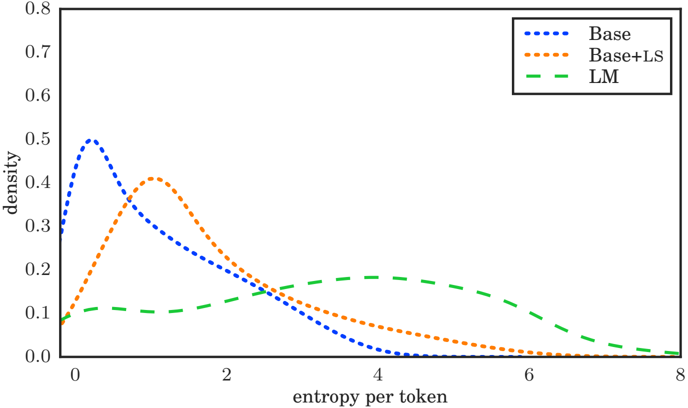
Table: BLEU scores of each NMT method (mean 3 runs)

Translation Results

Method	DE→EN	EN→DE	TR→EN	EN→TR
Base	26.6	25.6	16.6	11.2
Shallow-fusion	27.8	26.0	17.3	11.5
POSTNORM+ LS	26.4	23.3	16.0	11.0
Base + LS	28.4	27.3	18.4	12.6
Base + Prior	**30.2**	**29.1**	**19.5**	**13.8**
Base + Prior + LS	30.3	29.7	19.5	14.1
Base + Prior (30M)	30.0	29.8 (+0.7)	19.5	–

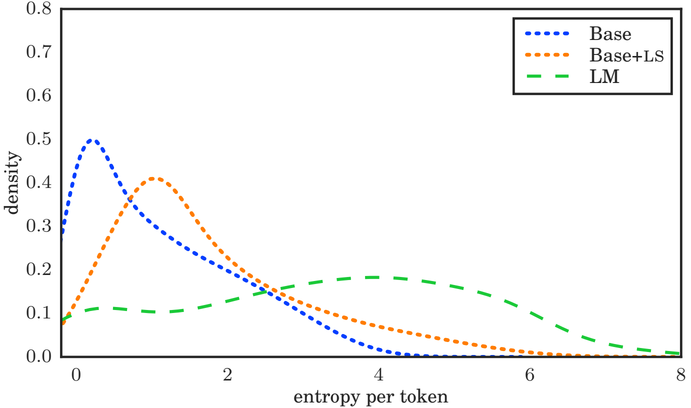
Table: BLEU scores of each NMT method (mean 3 runs)

Analysis



Estimated density of each model's entropy on the DE→EN test set

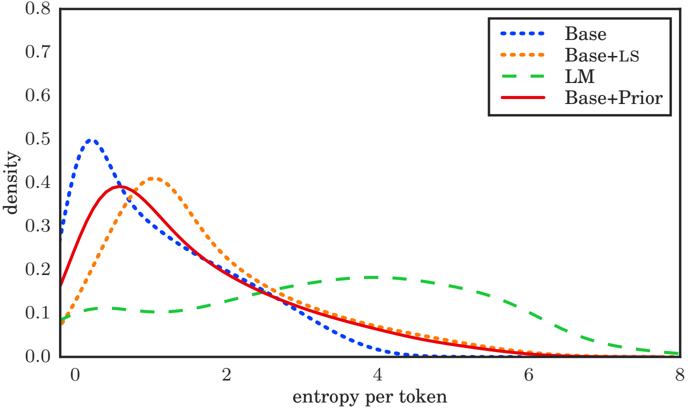
Analysis



Estimated density of each model's entropy on the DE→EN test set

- LS successfully mitigates overfitting, by **penalizing confidence**

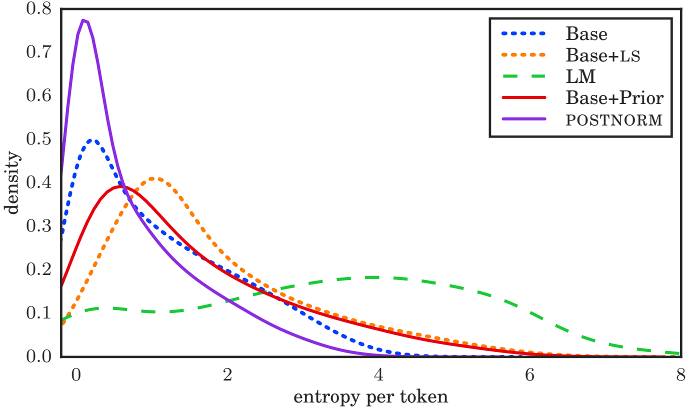
Analysis



Estimated density of each model's entropy on the DE→EN test set

- LS successfully mitigates overfitting, by **penalizing confidence**
- "Base+Prior" does **not just** penalize confidence, but **exploits** the LM

Analysis



Estimated density of each model's entropy on the DE→EN test set

- LS successfully mitigates overfitting, by **penalizing confidence**
- "Base+Prior" does **not just** penalize confidence, but **exploits** the LM
- "POSTNORM" is **over-confident**, due to unwanted interference from LM

Conclusions

1. **Simple** approach to incorporate prior knowledge in NMT
   - The decoupling of the LM from the TM enables **fast decoding**
   - We change **only** the training objective
2. **Promising results** in two low-resource translation datasets
   - Improvements even with **modest** monolingual data
3. **Analysis** of TM output distributions using different methods
   - Evidence that the model **exploits** the **knowledge** of the LM-prior

Thank you

Appendix

Bonus Slides

Full Results

Method	DE→EN		EN→DE		TR→EN		EN→TR	
	dev	test	dev	test	dev	test	dev	test
Base	22.6$_{\pm 0.1}$	26.6$_{\pm 0.1}$	18.3$_{\pm 0.3}$	25.6$_{\pm 0.2}$	15.9$_{\pm 0.0}$	16.6$_{\pm 0.3}$	12.2$_{\pm 0.1}$	11.2$_{\pm 0.2}$
Shallow-fusion	23.4$_{\pm 0.1}$	27.8$_{\pm 0.1}$	18.5$_{\pm 0.2}$	26.0$_{\pm 0.1}$	16.5$_{\pm 0.1}$	17.3$_{\pm 0.3}$	12.7$_{\pm 0.0}$	11.5$_{\pm 0.1}$
POSTNORM	20.4$_{\pm 0.2}$	24.5$_{\pm 0.3}$	16.6$_{\pm 0.1}$	22.9$_{\pm 0.3}$	13.8$_{\pm 0.2}$	11.0$_{\pm 0.1}$	10.0$_{\pm 0.1}$	10.2$_{\pm 0.1}$
Base + LS	23.8$_{\pm 0.6}$	28.4$_{\pm 0.7}$	19.2$_{\pm 0.3}$	27.3$_{\pm 0.3}$	17.5$_{\pm 0.1}$	18.4$_{\pm 0.2}$	13.8$_{\pm 0.2}$	12.6$_{\pm 0.0}$
Base + Prior	**24.9**$_{\pm 0.0}$	**30.2**$_{\pm 0.1}$	**20.5**$_{\pm 0.3}$	**29.1**$_{\pm 0.7}$	**18.5**$_{\pm 0.2}$	**19.5**$_{\pm 0.2}$	**15.1**$_{\pm 0.1}$	**13.8**$_{\pm 0.1}$
Base + Prior + LS	<u>25.1</u>$_{\pm 0.3}$	<u>30.3</u>$_{\pm 0.3}$	<u>20.8</u>$_{\pm 0.4}$	<u>29.7</u>$_{\pm 0.7}$	18.5$_{\pm 0.3}$	19.5$_{\pm 0.2}$	<u>15.5</u>$_{\pm 0.1}$	<u>14.1</u>$_{\pm 0.2}$
Base + Prior (30M)	24.9$_{\pm 0.1}$	30.0$_{\pm 0.1}$	<u>21.0</u>$_{\pm 0.4}$	<u>29.8</u>$_{\pm 0.3}$	<u>18.6</u>$_{\pm 0.0}$	19.5$_{\pm 0.2}$	–	–

Table: BLEU scores of each NMT method (mean and std of 3 runs)

The hyperparameters of each method were tuned on the DE→EN dev-set: LS($\alpha = 0.1$), shallow-fusion(β=0.1), LM-prior (λ=0.5, τ=2)

Results: Extremely Low-Resource Conditions

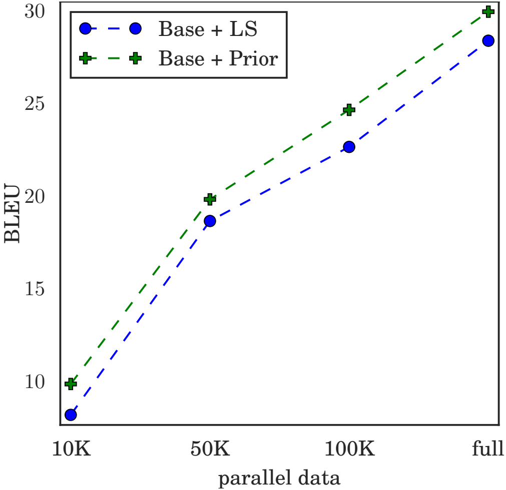


Figure: BLEU scores on the DE→EN test set with different scales of parallel data. Mean of 3 runs reported.

Perplexity Scores of LMs

language	3M (PPL↓)	30M (PPL↓)
English	29.70	**25.02**
German	22.71	**19.22**
Turkish	**22.78**	–

Table: Perplexity scores for LMs trained on each language's monolingual data, computed on a small held-out validation set per language.

Connection to Knowledge Distillation

Distillation also uses as **soft targets** the distributions of a teacher model

- Knowledge distillation (word-level) for NMT(Kim and Rush, 2016)

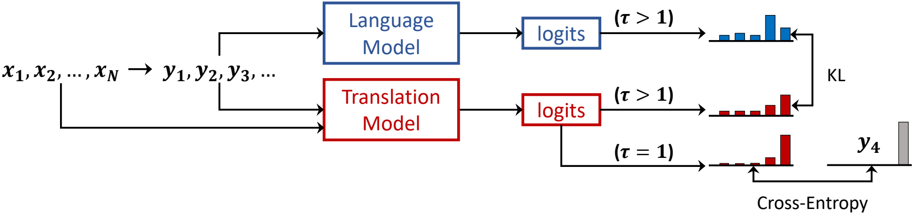
Connection to Knowledge Distillation

Distillation also uses as **soft targets** the distributions of a teacher model

- Knowledge distillation (word-level) for NMT(Kim and Rush, 2016)

Use softmax-temperature τ to reveal **extra information** from LM

- popularized as "*dark knowledge*" by (Hinton et al., 2015)



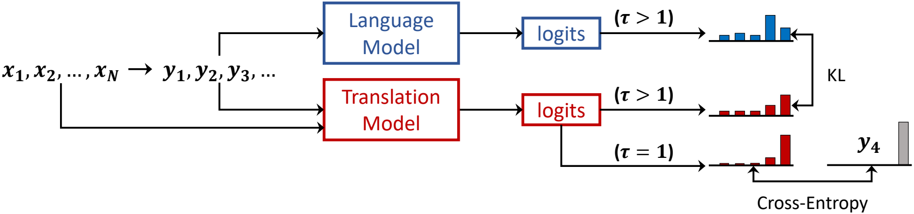
Connection to Knowledge Distillation

Distillation also uses as **soft targets** the distributions of a teacher model

- Knowledge distillation (word-level) for NMT(Kim and Rush, 2016)

Use softmax-temperature τ to reveal **extra information** from LM

- popularized as "*dark knowledge*" by (Hinton et al., 2015)



$$\mathcal{L} = \sum_{t=1}^{N} -\log p_{\text{TM}}(\boldsymbol{y}_t|\boldsymbol{y}_{<t}, \boldsymbol{x}) + \lambda\, \boldsymbol{\tau}^2 D_{\text{KL}}(p_{\text{LM}}(\boldsymbol{y}_t|\boldsymbol{y}_{<t}; \boldsymbol{\tau}) \parallel p_{\text{TM}}(\boldsymbol{y}_t|\boldsymbol{y}_{<t}, \boldsymbol{x}; \boldsymbol{\tau}))$$

$\mathcal{L}_{\mathrm{KL}}$ Sensitivity Analysis

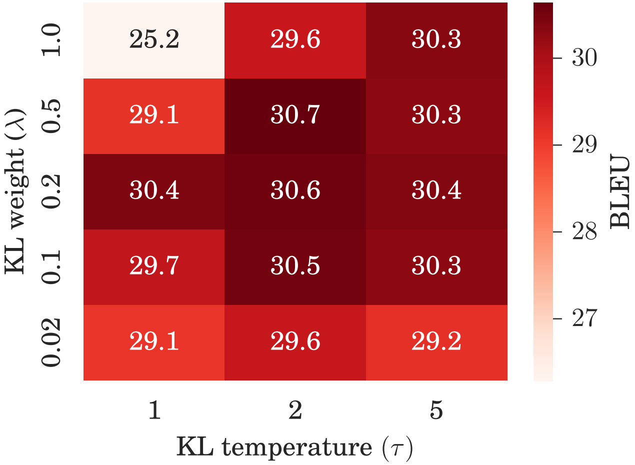


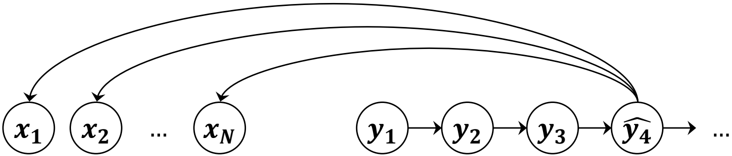
Figure: BLEU scores on the DE→EN test set of models trained with different λ and τ for the $\mathcal{L}_{\mathrm{KL}}$. Mean of 3 runs for each combination reported.

Prior Work: Noisy Channel

Statistical Machine Translation

It models the "reverse translation probability" $p(\boldsymbol{y}|\boldsymbol{x}) \propto p(\boldsymbol{x}|\boldsymbol{y}) \times p(\boldsymbol{y})$

$$\hat{\boldsymbol{y}} = \arg\max_{\boldsymbol{y}} \underbrace{p_{\mathrm{TM}}(\boldsymbol{x}|\boldsymbol{y})}_{\text{translation model}} \times \underbrace{p_{\mathrm{LM}}(\boldsymbol{y})}_{\text{language model}}$$

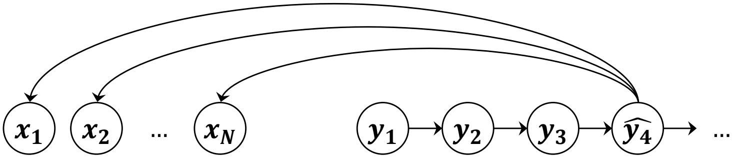


Prior Work: Noisy Channel

Statistical Machine Translation

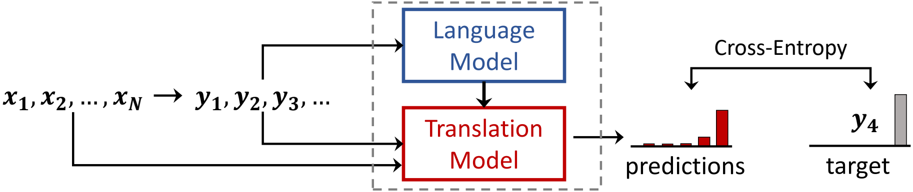
It models the "reverse translation probability" $p(\boldsymbol{y}|\boldsymbol{x}) \propto p(\boldsymbol{x}|\boldsymbol{y}) \times p(\boldsymbol{y})$

$$\hat{\boldsymbol{y}} = \arg\max_{\boldsymbol{y}} \underbrace{p_{\mathrm{TM}}(\boldsymbol{x}|\boldsymbol{y})}_{\text{translation model}} \times \underbrace{p_{\mathrm{LM}}(\boldsymbol{y})}_{\text{language model}}$$

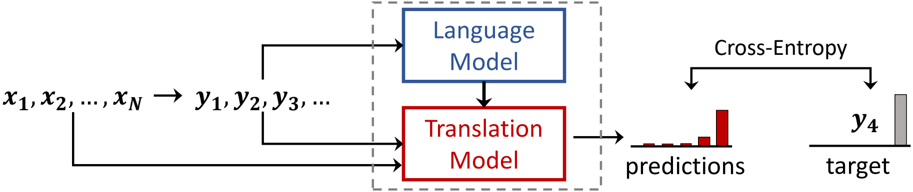


Decoding with seq2seq is slow!

Prior Work: Language Model Fusion



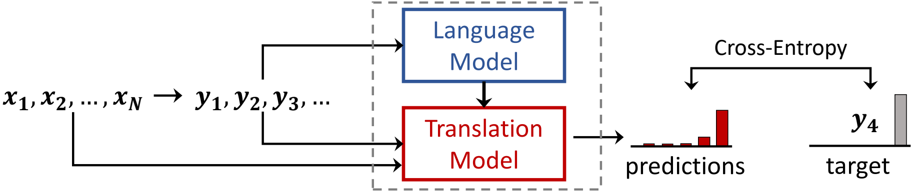
Prior Work: Language Model Fusion



Feature-based: incorporate the representations of the LM

- deep-fusion (Gulcehre et al., 2015)
- cold-fusion (Sriram et al., 2018)

Prior Work: Language Model Fusion



Feature-based: incorporate the representations of the LM

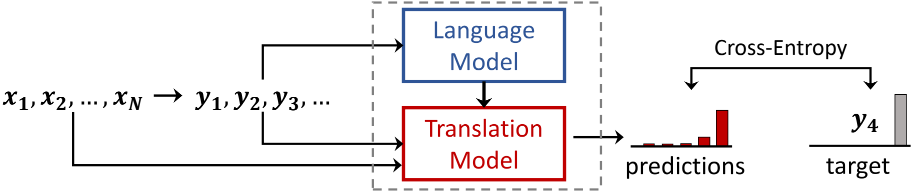
- deep-fusion (Gulcehre et al., 2015)
- cold-fusion (Sriram et al., 2018)

Probability Interpolation: incorporate the predictions of the LM

- shallow-fusion (Gulcehre et al., 2015)

$$\hat{\boldsymbol{y}} = \arg\max_{\boldsymbol{y}} \; \log p_{\text{TM}}(\boldsymbol{y}_t | \boldsymbol{y}_{<t}, \boldsymbol{x}) + \boldsymbol{\beta} \log p_{\text{LM}}(\boldsymbol{y}_t | \boldsymbol{y}_{<t})$$

Prior Work: Language Model Fusion



Feature-based: incorporate the representations of the LM

- deep-fusion (Gulcehre et al., 2015)
- cold-fusion (Sriram et al., 2018)

Probability Interpolation: incorporate the predictions of the LM

- shallow-fusion (Gulcehre et al., 2015)

$$\hat{\boldsymbol{y}} = \arg\max_{\boldsymbol{y}} \ \log p_{\mathrm{TM}}(\boldsymbol{y}_t | \boldsymbol{y}_{<t}, \boldsymbol{x}) + \boldsymbol{\beta} \log p_{\mathrm{LM}}(\boldsymbol{y}_t | \boldsymbol{y}_{<t})$$

- simple-fusion (Stahlberg et al., 2018) $\rightarrow$ POSTNORM:

$$p(\boldsymbol{y}_t) = \mathsf{softmax}(\log p_{\mathrm{TM}}(\boldsymbol{y}_t | \boldsymbol{y}_{<t}, \boldsymbol{x}) + \log p_{\mathrm{LM}}(\boldsymbol{y}_t | \boldsymbol{y}_{<t}))$$

References I

Caglar Gulcehre, Orhan Firat, Kelvin Xu, Kyunghyun Cho, Loic Barrault, Huei-Chi Lin, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2015. On using monolingual corpora in neural machine translation. *arXiv preprint arXiv:1503.03535*.

Geoffrey E. Hinton, Oriol Vinyals, and Jeffrey Dean. 2015. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, abs/1503.02531.

Yoon Kim and Alexander M. Rush. 2016. Sequence-level knowledge distillation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 1317–1327, Austin, Texas, USA.

Anuroop Sriram, Heewoo Jun, Sanjeev Satheesh, and Adam Coates. 2018. Cold fusion: Training seq2seq models together with language models. In *Proceedings of Interspeech*, pages 387–391.

Felix Stahlberg, James Cross, and Veselin Stoyanov. 2018. Simple fusion: Return of the language model. In *Proceedings of the Conference on Machine Translation*, pages 204–211, Belgium, Brussels.