

# Looking beyond Unicode for Open-Vocabulary Text Representations

Elizabeth Salesky, Johns Hopkins University

# Looking beyond Unicode for Open-Vocabulary Text Representations

Elizabeth Salesky, Johns Hopkins University

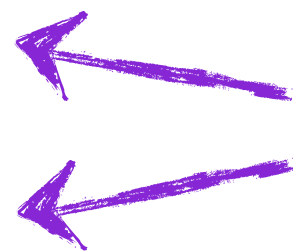


“Robust Open-Vocabulary Translation  
from Visual Text Representations”

Elizabeth Salesky, Dave Etter, Matt Post  
EMNLP 2021

# Introduction

# Motivation

- Challenge:
  - We want our models to be able to represent all words in a given language
    - “Open-vocabulary” modeling, able to represent unseen words at test time
- Common solutions:
  - Characters
  - \_Sub words

unobserved tokens can be broken into observed components
- Potential issues:
  - Robustness
  - Predetermined vocabulary



# Motivation

Phenomena	Word	BPE	
Vowelization	كتاب	كتاب	(1)
	الكِتاب	ا . ب . ت . ك	(5)
Misspelling	language	language	(1)
	langauge	la . ng . au . ge	(4)
Visually Similar Characters	really	really	(1)
	rea11y	re . a . 1 . 1 . y	(5)
Shared Character Components	확인한다	확인 . 한 . 다	(3)
	확인했다	확인 . 했다	(2)

- Challenge:

- We need a way to represent all words
- Common solutions (e.g., subwords) have potential issues:
  - Robustness
  - Predetermined vocabulary

Examples of common behavior which cause divergent representations for subword models

# Motivation

Phenomena	Word	BPE	
Vowelization	كتاب	كتاب	(1)
	الكتاب	أب . ت . الك	(5)
Misspelling	language	language	(1)
	langauge	la . ng . au . ge	(4)
Visually Similar Characters	really	really	(1)
	rea11y	re . a . 1 . 1 . y	(5)
Shared Character Components	확인한다	확인 . 한 . 다	(3)
	확인했다	확인 . 했다	(2)

Examples of common behavior which cause divergent representations for subword models

لحم

lhm

**l h m**

lh hm

لَحْم

laham

**l a h a m**

la ah ha am

lah ham

laha aham

Few possible subwords in common

# Motivation

Phenomena	Word	BPE	
Vowelization	کتاب	کتاب	(1)
	اَلکتاب	اِب . ت . اَلک	(5)
Misspelling	language	language	(1)
	langauge	la . ng . au . ge	(4)
Visually Similar Characters	really	really	(1)
	rea11y	re . a . 1 . 1 . y	(5)
Shared Character Components	확인한다	확인 . 한 . 다	(3)
	확인했다	확인 . 했다	(2)

ه U+06D5

ه U+06C0

ه U+0647

ه U+0647, U+0654, U+200C

ه U+06D5, U+0654

ه U+0647, U+0654

ي U+064A

ی U+06CC

ی U+0649

درې U+1583, U+1585, U+1744

دري U+1583, U+1585, U+064A

Different underlying unicode codepoints, visually similar

Examples of common behavior which cause divergent representations for subword models

# Motivation

Phenomena	Word	BPE	
Vowelization	كتاب	كتاب	(1)
	الكتاب	أب . ت . الك	(5)
Misspelling	language	language	(1)
	langauge	la . ng . au . ge	(4)
Visually Similar Characters	really	really	(1)
	rea11y	re . a . 1 . 1 . y	(5)
Shared Character Components	확인한다	확인 . 한 . 다	(3)
	확인했다	확인 . 했다	(2)

Examples of common behavior which cause divergent representations for subword models

- Motivation:
  - Common representations rely on consecutive (exact) character matches
  - Visually similar text may have a similar meaning
- Goal:
  - More robust input representations
  - Tokenization-free, open vocabulary



# Visual text representations

Given text in a font (ex: NotoSans),  
of a particular size...

Das ist ein Satz.

unicode string



1

render to  
an image

Das ist ein Satz.

# Visual text representations

Given text in a font (ex: NotoSans),  
of a particular size...

Das ist ein Satz.

unicode string



1

render to  
an image

Das ist ein Satz.



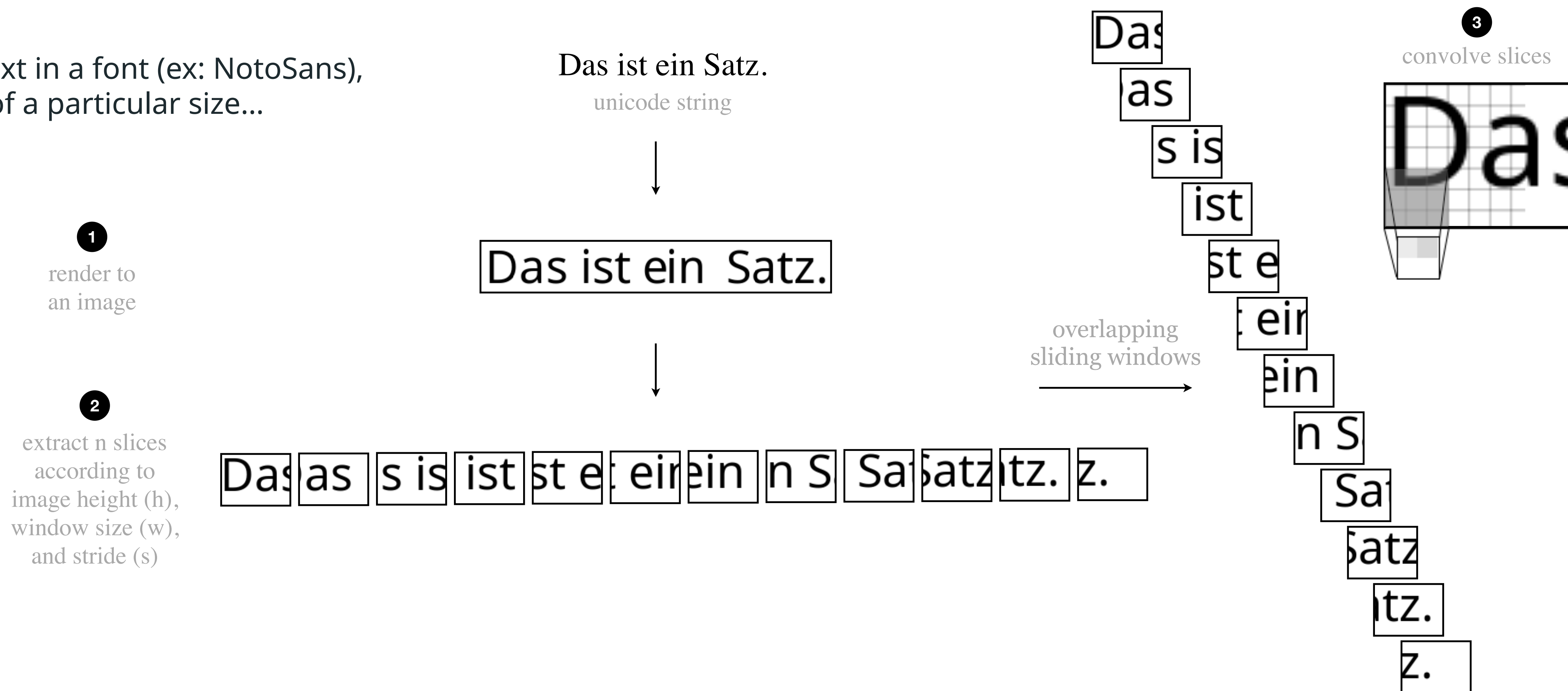
2

extract n slices  
according to  
image height (h),  
window size (w),  
and stride (s)

Das as s is ist st e: eir ein n S Sa Satz itz. z.

# Visual text representations

Given text in a font (ex: NotoSans),  
of a particular size...



# Visual text representations

- Why do we render the whole sentence?
  - As opposed to say, rendering each character or word
- Many scripts have *contextual* forms and require context to render correctly
  - In Arabic characters can appear differently based on whether they appear in isolation or in context, and based on what they precede or follow. Rendering diacritics individually places them incorrectly
  - To make sure we render correctly, we need the full sentence!

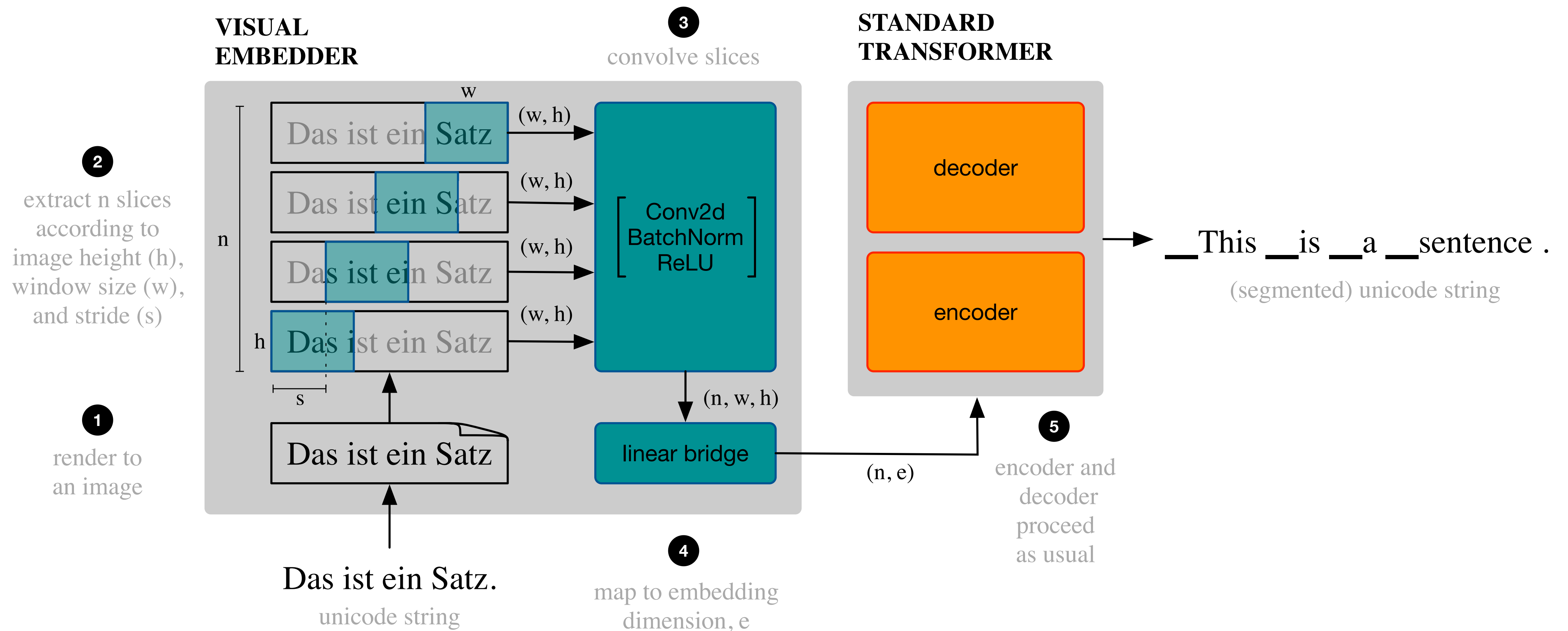
**Bad:** أنا لَكَ نَدِيَّةٌ ، وَأَنَا أَصْغَرُ إِخْوَانِي السَّبْعَةِ  
**Good:** أَنَا كَنَدِيَّةٌ ، وَأَنَا أَصْغَرُ إِخْوَانِي السَّبْعَةِ

- Many languages do not mark *whitespace*
  - No segmentation commitments during rendering!





# Visual text representations



# Evaluating visual text representations

# Experimental design (MT)

- Language pairs (7)

- Source, multiple scripts: Arabic Chinese French German Japanese Korean Russian  
عربي, 官话, Français, Deutsch, 日本語, 한국어, русский
- Target language: English

- Datasets (2)

- “Small” — MTTT (TED) ar zh fr de ja ko ru
- “Larger” — WMT (filtered) zh de

- Visual architecture

- Significant hyperparameters unknown at the offset — new approach!
- Convolutional blocks {0,1,7} 0≡Vision Transformer; 7≡OCR

# Text model baselines

- Transformer models in fairseq
- Carefully tune source representations
- Target vocabulary held constant
  - Direct comparison with visual text models
- Improvements of ~2 BLEU over previous work

fr	36.2	36.7	36.5	36.4	36.5	35.2	35.8	35.7	35.6	31.7
de	33.2	33.2	33.5	33.6	33.6	33.1	33.4	32.9	33.0	27.3
ar	32.1	31.7	31.8	32.1	31.0	31.0	30.6	30.7	30.3	17.6
ru	25.2	25.2	25.4	25.0	24.7	24.7	25.0	24.7	24.4	13.9
zh	17.9	*	18.3	17.7	17.2	17.4	17.2	17.5	17.2	0.5
ko	16.9	16.9	17.0	16.8	16.8	16.8	16.8	16.3	15.7	6.3
ja	13.7	14.4	14.3	13.5	13.9	13.6	12.7	12.7	12.2	5.8
	Chars	2.5k BPE	5k BPE	10k BPE	15k BPE	20k BPE	25k BPE	30k BPE	35k BPE	Words

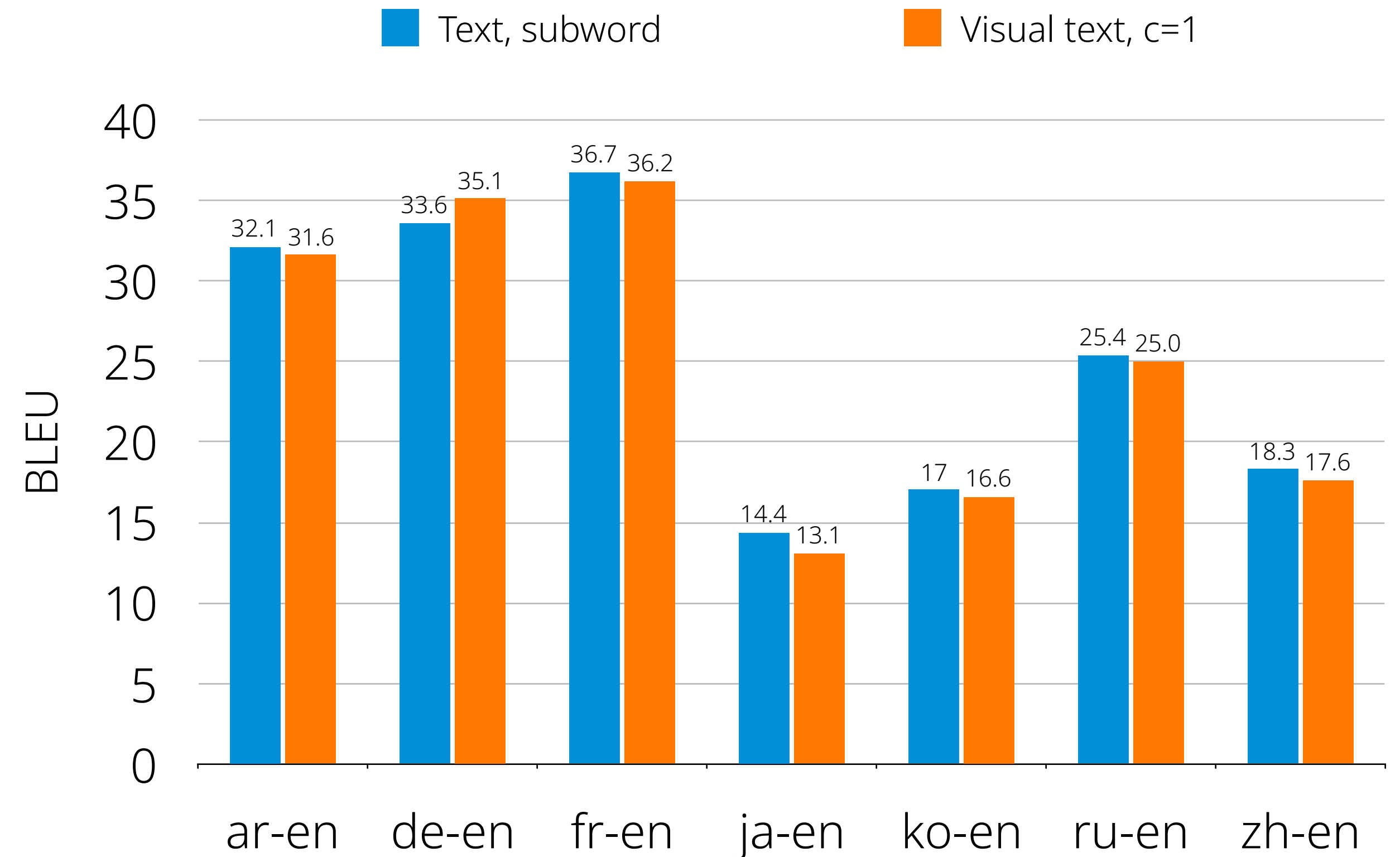
Standardized MTTT test set

\*The character vocabulary of zh is larger than 2.5k



# “Small” data – MTTT

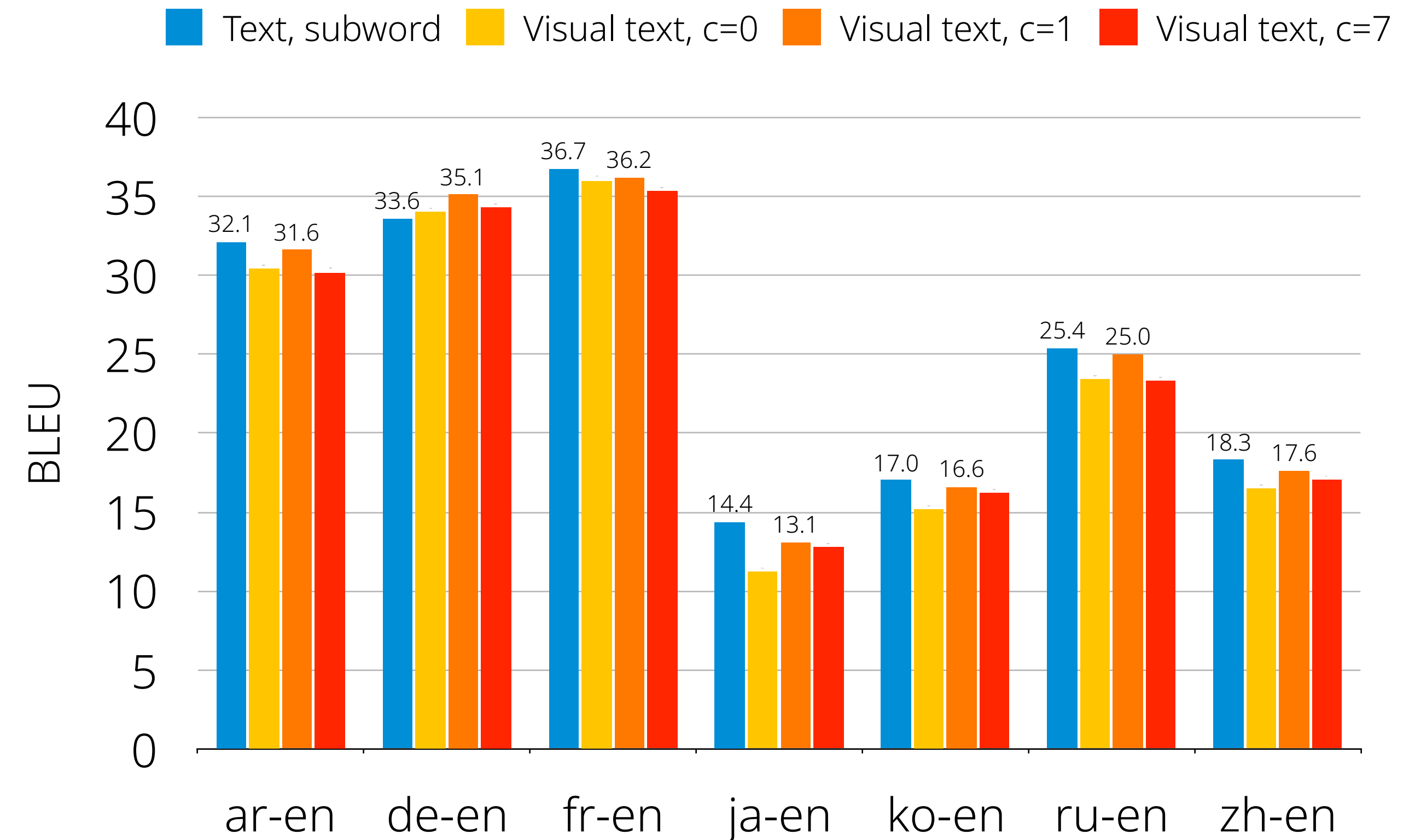
- Approach parity with best text models
  - Within  $[-1.3, +1.5]$  BLEU
- Best visual text results use  $c = 1$ 
  - Some structural biases from convolutions w/o excessive visual depth



Standardized MTTT test set  
 $c = \text{num. convolution blocks}$

# “Small” data – MTTT

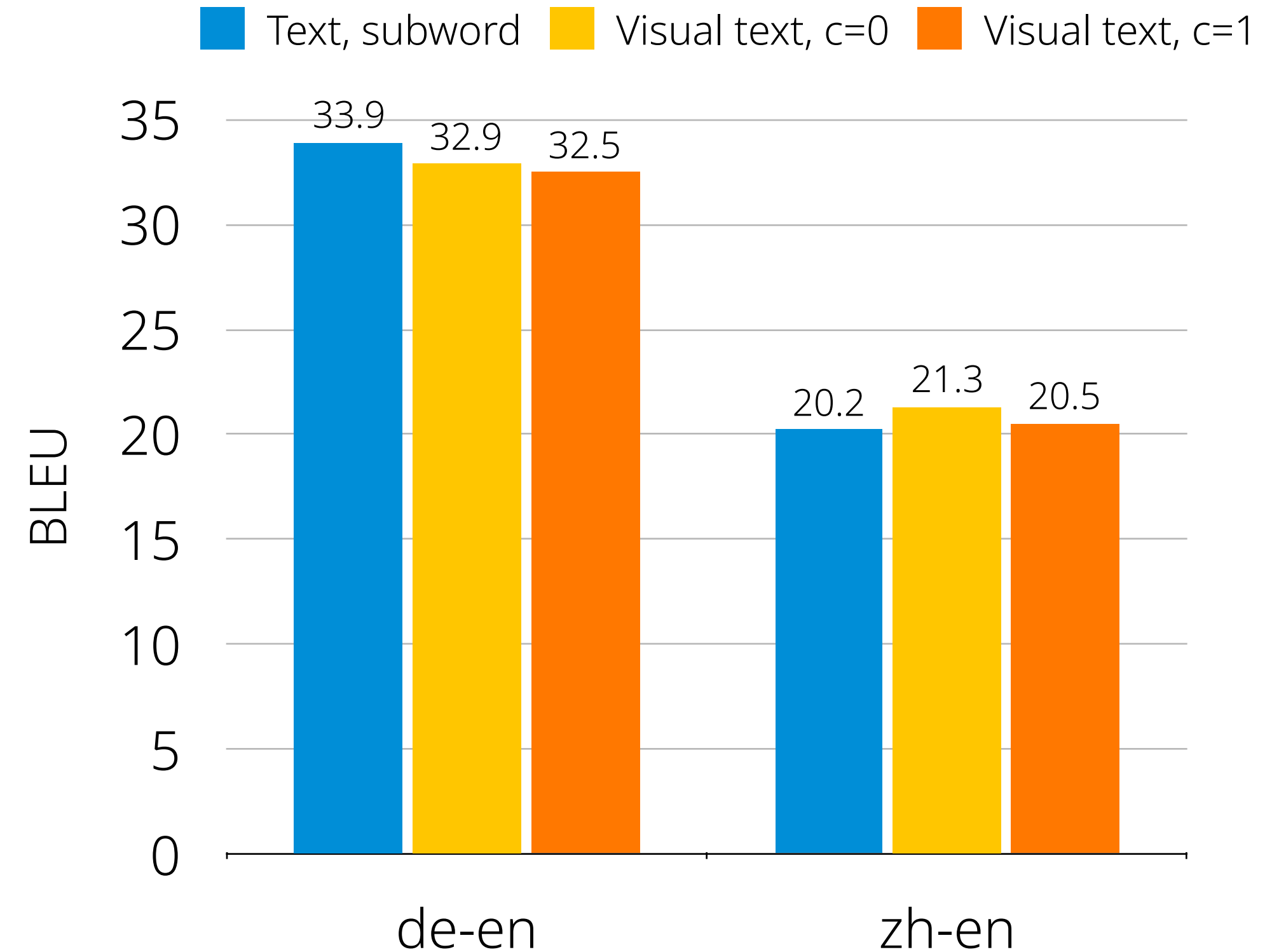
- Approach parity with best text models
  - Within  $[-1.3, +1.5]$  BLEU
- Best visual text results use  $c = 1$ 
  - Some structural biases from convolutions w/o excessive visual depth
- Greater visual capacity ( $c > 7$ ) does not improve results for our task



Standardized MTTT test set  
 $c = \text{num. convolution blocks}$

# “Larger” data — WMT

- Similar trends: on par with text models
  - This suggests our approach scales and its efficacy is not limited to lower-resource settings
- With more data,  $c = 0$  outperforms  $c = 1$ 
  - This ‘direct’ model may simply require more training data



WMT'20 newstest sets  
 $c = \text{num. convolution blocks}$



# Where are improvements from?

- Are our results due to visual representations (as opposed to say, sliding window segmentation)?
- Ablation: apply sliding windows to text, removing visual representations!
  - Essentially character n-grams: “this is a test”  $\mapsto$  “thi his si ...”

MODEL :	ar	de	fr	ja	ko	ru	zh
Visual text	31.6	35.1	36.2	13.1	16.6	25.0	17.6
<i>w/o visrep</i> (char n-grams)	31.5	34.6	36.4	1.4	1.3	24.6	5.5
Text, BPE	32.1	33.6	36.7	14.4	17.0	25.4	18.3

Table 10: **Ablation:** Sliding window segmentation (character n-grams) applied to text without visual rendering.





# Where are improvements from?

- What happens when we induce noise?
  - Character n-grams have worse performance than the text BPE models

MODEL :		ar	de	fr	ja	ko	ru	zh
Visual text		31.6	35.1	36.2	13.1	16.6	25.0	17.6
<i>w/o visrep</i> (char n-grams)		31.5	34.6	36.4	1.4	1.3	24.6	5.5
Text, BPE		32.1	33.6	36.7	14.4	17.0	25.4	18.3
NOISED :								
Visual text; swap p=0.5		21.7	29.4	28.4	—	11.5	18.3	—
<i>w/o visrep</i> ; swap p=0.5		11.2	10.8	11.9	—	1.1	9.5	—
Text, BPE; swap p=0.5		12.4	13.1	13.3	—	10.8	11.1	—

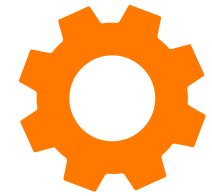
Table 10: **Ablation:** Sliding window segmentation (character n-grams) applied to text without visual rendering.



# Where are improvements from?

- These experiments suggest the visual text representations are the main source of improvement!
- Why?
  - Languages with (more) uniform char n-gram frequencies (ar de fr ru) did better; worse results for languages with many poorly trained embeddings (ja ko zh)
  - Char n-grams, like BPE, have no access to token composition — unlike visual text representations!

# Technical details



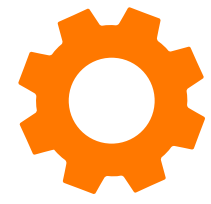
- Which hyperparameters matter, and how many require tuning?



- What is the relative training and inference speed?



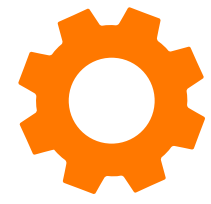
- Do visual text models change the number of model parameters?



# Hyperparameters

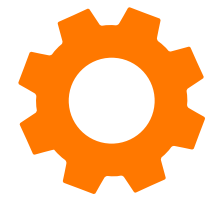
- Which hyperparameters matter, and how many require tuning?
  - Most tested do not require tuning — new approach, tested more than needed!
- What are they?
  - Rendering text:
    - *Font, font size*
  - Image “token” segmentation:
    - *Window size, window stride*
  - Visual architecture parameters:
    - *Number of convolutional layers, convolutional kernel size and stride*





# Hyperparameters

- Which hyperparameters matter, and how many require tuning?
  - Most tested do not require tuning — new approach, tested more than needed!
- What are they?
  - Rendering text:
    - *Font, ~~font size~~*
  - Image “token” segmentation:
    - *Window size, ~~window stride~~*
  - Visual architecture parameters:
    - *Number of convolutional layers, ~~convolutional kernel size and stride~~*



# Hyperparameters

- Which hyperparameters matter, and how many require tuning?
  - Most tested do not require tuning — new approach, tested more than needed!
    - Font needs to be large enough to not affect resolution: at least 10pt
    - $c=1$  is consistently better at low-resource settings; similar to  $c=0$  with more data
    - Convolutional kernel consistently best at a single setting (3x3)
  - Window size is ‘most’ language-specific, but differences small: large tolerance



# Hyperparameters

- Which hyperparameters matter, and how many require tuning?
  - Most tested do not require tuning — new approach, tested more than needed!
    - Font needs to be large enough to not affect resolution: at least 10pt
    - $c=1$  is consistently better at low-resource settings; similar to  $c=0$  with more data
    - Convolutional kernel consistently best at a single setting (3x3)
  - Window size is ‘most’ language-specific, but differences small: large tolerance

↴ small differences in BLEU across similar window sizes

DE-EN	$c = 1, font = 10pt$						
$s \downarrow / w \rightarrow$	10	15	20	25	30	35	40
5	0.7	32.6	35.1	0.5	33.1	33.9	32.5
10	0.6	34.6	34.8	32.8	32.9	34.4	33.5
15		32.8	33.9	32.0	31.4	33.7	33.9

FR-EN	$c = 1, font = 10pt$						
$s \downarrow / w \rightarrow$	10	15	20	25	30	35	40
5	35.4	35.7	35.7	35.5	0.7	0.6	0.8
10	35.6	36.2	36.1	36.1	34.7	34.7	35.0
15		35.7	35.8	35.6	34.4	34.3	34.6

↴ slight instability with small stride

Appendix A



# Changes in speed?

- What is the relative training and inference speed?
  - Changes in training time primarily depend on sequence length
    - Sequence length determined by **stride**
    - Best model settings result in training times are **between characters and BPE**
  - No observable difference with BPE models at inference time

Lang	Text		Visual text			
	BPE	char	$s = 5$	$s = 10$	$s = 15$	$s = 20$
ar	24.4	78.9	97.1	48.8	32.7	24.6
de	32.3	104.3	130.5	65.5	43.8	33.0
fr	28.8	107.6	130.2	65.4	43.7	32.9
ja	22.5	36.9	95.5	48.0	32.1	24.2
ko	24.7	50.8	97.0	48.7	32.6	24.6
ru	27.1	94.7	132.7	66.6	44.5	33.5
zh	23.0	29.8	75.6	38.1	25.5	19.3
Time	1.0×	2.3×	3.9×	2.0×	1.4×	1.2×

Table 2: Average sequence lengths of MTTT data for text models and visual models with varying stride  $s$ . The bottom row shows training time relative to the fastest model (BPE) with  $c = 1$ .





# Num. model parameters

- Do visual text models change the number of model parameters?
  - Not really!
- Any increase in model parameters is determined by window size and number of convolutional blocks
  - Essentially, trade the source embedding matrix parameters for convolutional layer
  - For our best models, # parameters are within 1% of original text models' :
    - MTTT TED:  $36.7\text{M} \pm 0.2\text{M}$

# Model robustness

# Recall: Motivation

Phenomena	Word	BPE	
Vowelization	كتاب	كتاب	(1)
	الكِتاب	ا ب . ت . ك . ال	(5)
Misspelling	language	language	(1)
	langauge	la . ng . au . ge	(4)
Visually Similar Characters	really	really	(1)
	rea1ly	re . a . 1 . 1 . y	(5)
Shared Character Components	확인한다	확인 . 한 . 다	(3)
	확인했다	확인 . 했다	(2)

Examples of common behavior which cause divergent representations for subword models

- Motivation:

- Common representations rely on consecutive (exact) character matches
- Visually similar word forms may have similar meanings

- Goal:

- More robust input representations
- Tokenization-free, open vocabulary



# Visual text (robust)

```
#####
```

```
## VISUAL TEXT (VISREP): German-English ##
```

```
#####
```

```
2021-09-27 16:53:40 | INFO | fairseq.tasks.visual_text | dictionary size (dict.en.txt): 10,072
```

```
2021-09-27 16:53:40 | INFO | fairseq_cli.interactive | loading model(s) from ./checkpoint_best.pt
```

```
2021-09-27 16:53:41 | INFO | fairseq.data.visual.image_generator | Image window size 20 stride 5
```

```
2021-09-27 16:53:45 | INFO | fairseq_cli.interactive | Type the input sentence and press return:
```



# Robustness

We induced five types of noise, as below:

- **diacritics**: diacritization of Arabic via [camel-tools](#) ar
- **unicode**: substitutes visually similar unicode characters ru
- **133tspeak**: substitutes numbers or other visually similar characters for Latin alphabet characters de fr
- **swap**: swaps two adjacent characters in a token  $|\text{token}| \geq 2$  ar de fr ko ru
- **cmabrigde**: permutes word-internal characters with first and last character unchanged  $|\text{token}| \geq 4$  ar de fr ko ru



# Diacritics & Unicode

ar-en

**src** أنا كنديّة، وأنا أصغر أخواني السبعة  
**noised** أنا كَنَدِيَّةٌ ، وَأَنَا أَصْغَرُ إِخْوَاني السَّبْعَةِ  
**ref** I'm Canadian, and I'm the youngest of seven kids.

**visrep** أنا كَنَدِيَّةٌ بَدِيَّةٌ ، وَأَنَا أَصْغَرُ إِخْوَاني السَّبْعَةِ  
I'm a Canadian, and I'm the youngest of my seven sisters.

**BPE** .أَنَا كَنَدِيَّةٌ ، وَأَنَا أَصْغَرُ إِخْوَاني السَّبْعَةِ  
We grew up as a teacher, and we gave me a hug.

ru-en

Я расскажу вам об этой технологии.  
**R** **рас**скажу **ва**М об **этой** **тех**но**ло**гии  
I'm going to tell you about that technology.

R рас рассказажу вам об этой технологии.  
I'm going to tell you about this technology.

\_R\_р\_а\_с\_с\_к\_а\_ж\_у\_в\_а\_М\_о\_б\_э\_т\_ой\_т\_е\_х\_н\_о\_л\_о\_г\_и\_и\_.  
I'm going to put my mouth in the dam of ecsta chhallogi.

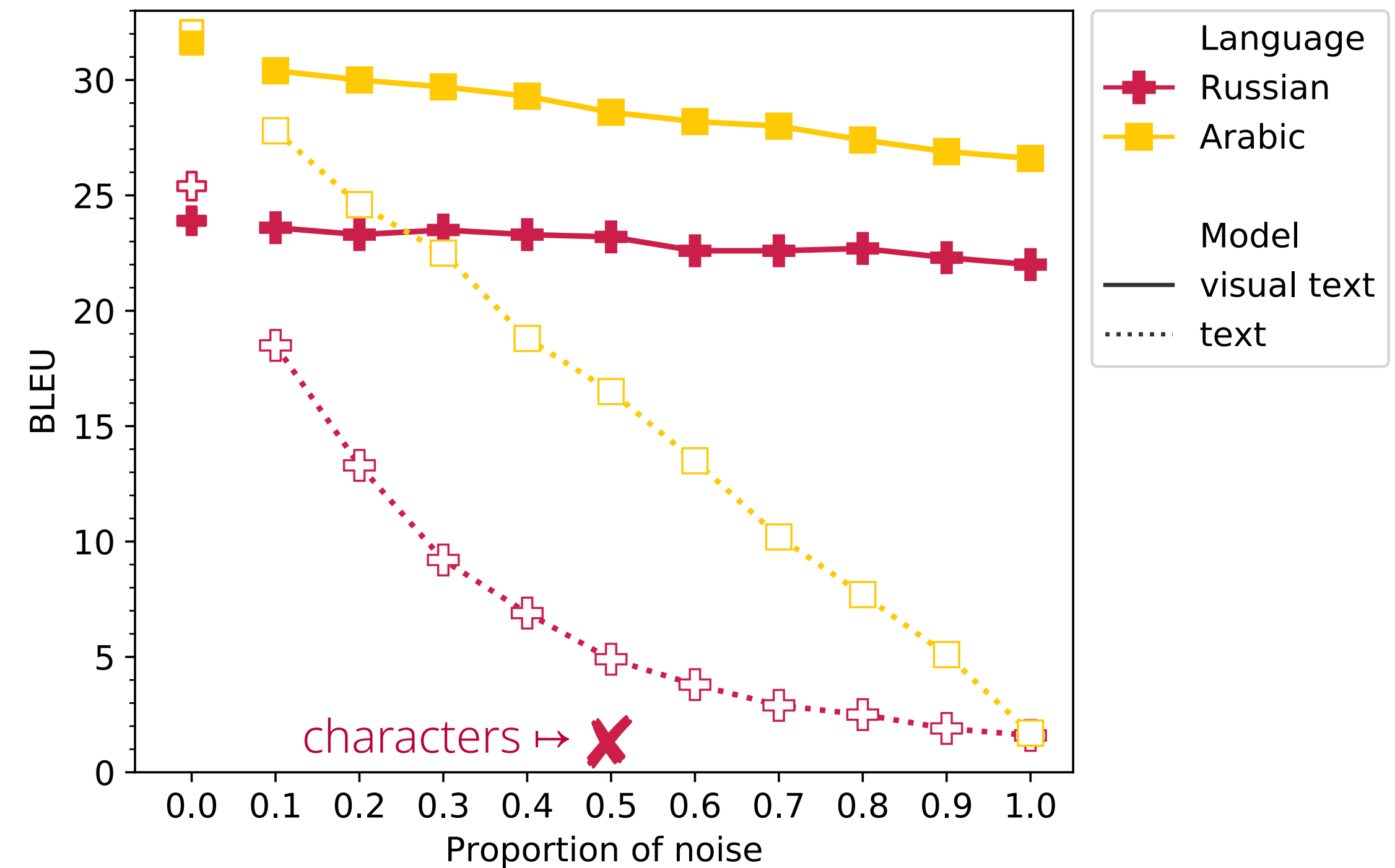
# Diacritics & Unicode

- Large changes to unicode sequences; visually, changes to only 0-5% pixels
  - Unsurprising our method does so well!

WIPO

The invention belongs to the field of biotechnology, pharmaceuticals and medicine, it could be applied for the production of drugs and for the realization of medicinal technologies, particularly for the immunotherapy of oncological diseases.

Cyrillic Latin



# 133tpeak

fr-en

**src** Un homme de 70 ans qui voudrait une nouvelle hanche, pour qu'il puisse retourner au golf ou s'occuper de son jardin.  
**noised** Un homme de 70 an<sup>5</sup> qu<sup>1</sup> voudrait un<sup>3</sup> nouvelle h<sup>4</sup>nche, pour qu'il pui<sup>5</sup>s<sup>3</sup> re<sup>7</sup>ourner au golf ou s'occuper de son jardin.  
**ref** Some 70-year-old who wanted his new hip so he could be back golfing, or gardening.

**visrep** Un Un h hc hor pm mm me ne e d de le 7 70 70 a 0 ar an 5 5q qu u1 1v vo ouc udr dra rait ait t u un n3 3n no ...  
A 70-year-old man who would like a new hip, so that he could turn to golf or take care of his garden.

**BPE** \_Un \_homme \_de \_70 \_an 5 \_qu 1 \_voudr ait \_un 3 \_nouvelle \_h 4 nch e , \_pour \_qu ' il \_pu i 5 s 3 \_re 7 our ner \_au ...  
A 75-year-old man wants a third new hip, so that he can punish himself for the golf or take care of his garden.

# l33tspeak

- Improvements of up to 7 BLEU, but, reduced as context contains increasingly more noise
  - Convention dictates l33t substitutions as much as visual similarity (which can be font-specific)

sample l33t mappings

e→3      a→4

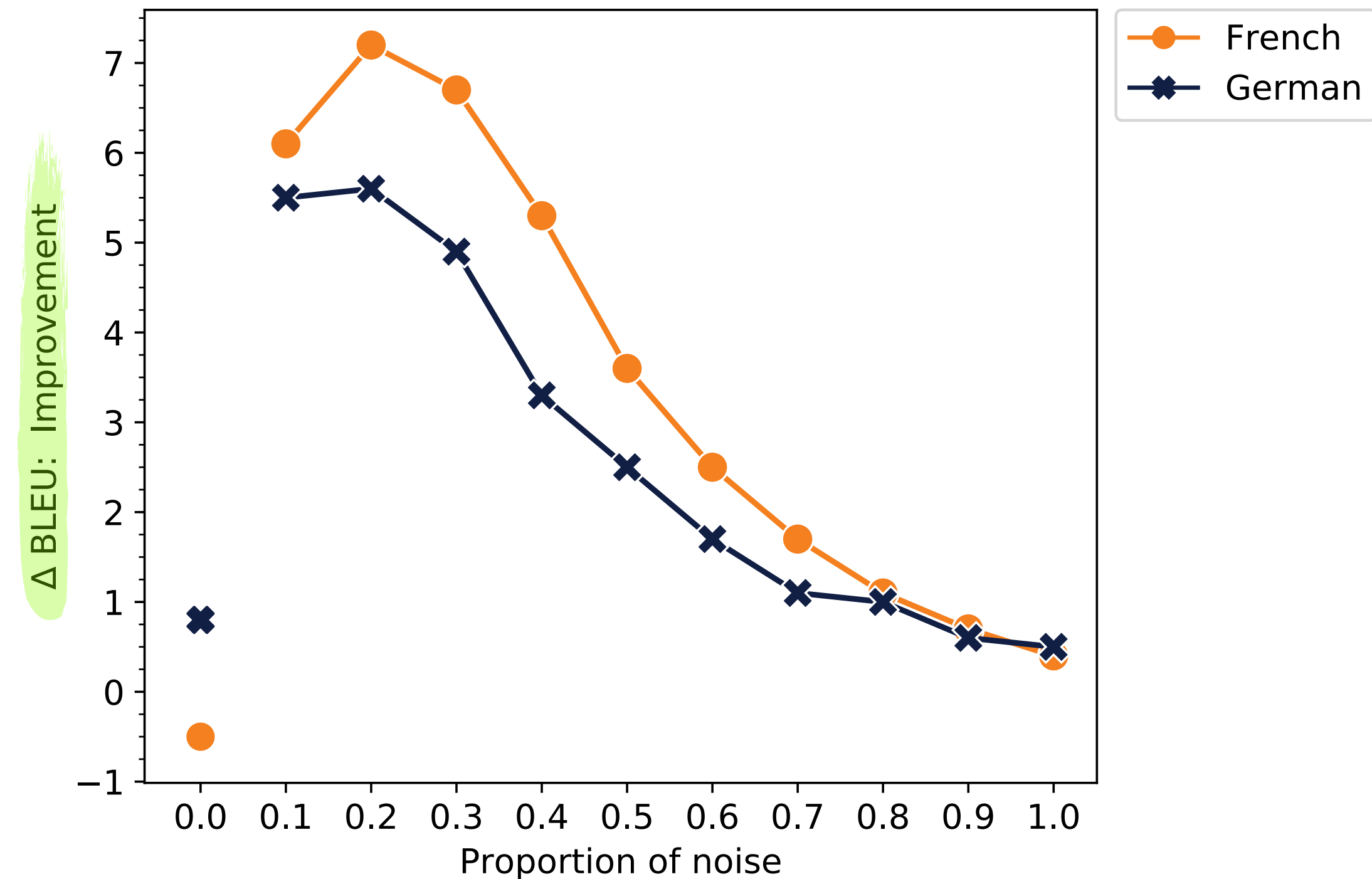
t→7

different fonts

a→4      e→3

confusable non-l33t pairs

t→1      z→7





# Character permutations

de-en

src

Aber Sie müssen zuerst zwei Dinge über mich wissen.

noised

Abre Sie müssen zuerts wz ei Dnige über mcih wisse.n

ref

But first you need to know two things about me.

visrep

Ab bre re e Si Sie sie re m mü hüs üss sse sen n z zu zue uer erts tsw swz wz ei D Dr Dni nig ige je ü ü ü be ber er r r ...

But you have to know two things about me first.

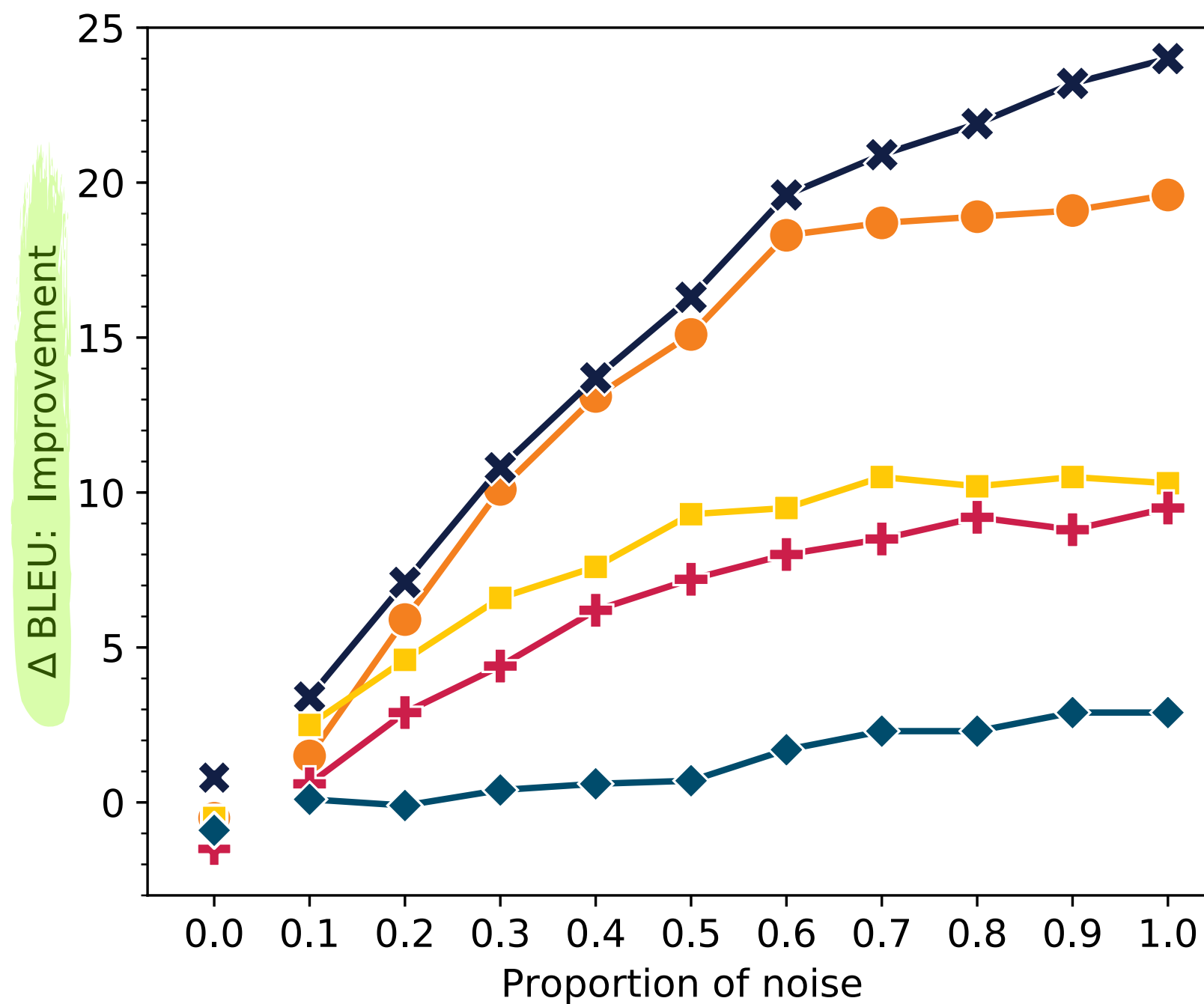
BPE

\_Ab re \_Sie \_müssen \_zu ert s \_w z ei \_D n ige \_über \_m ci h \_wiss e . n

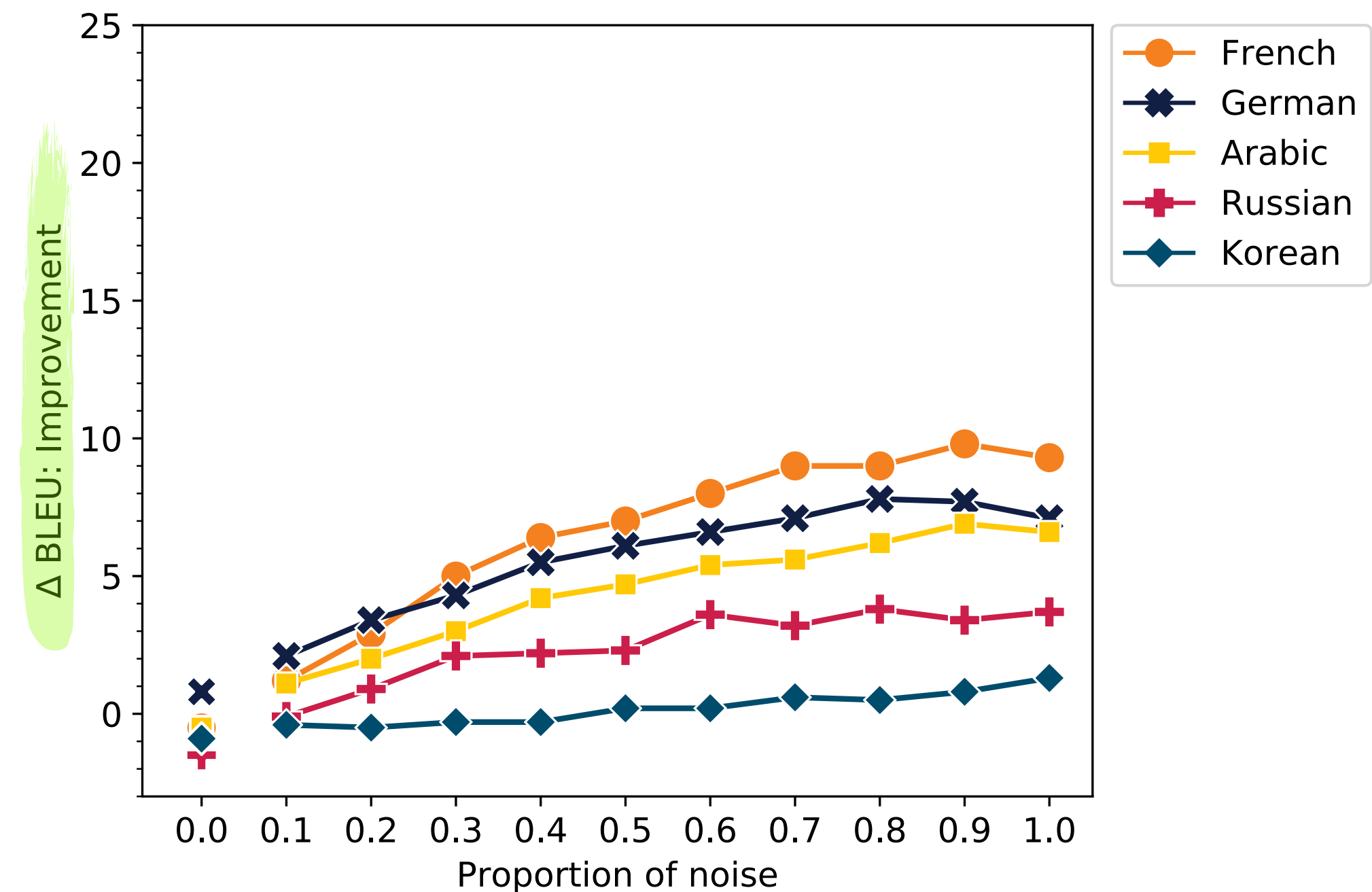
But you've got to get into a little about you.

# Character permutations

swap



cmabrigde

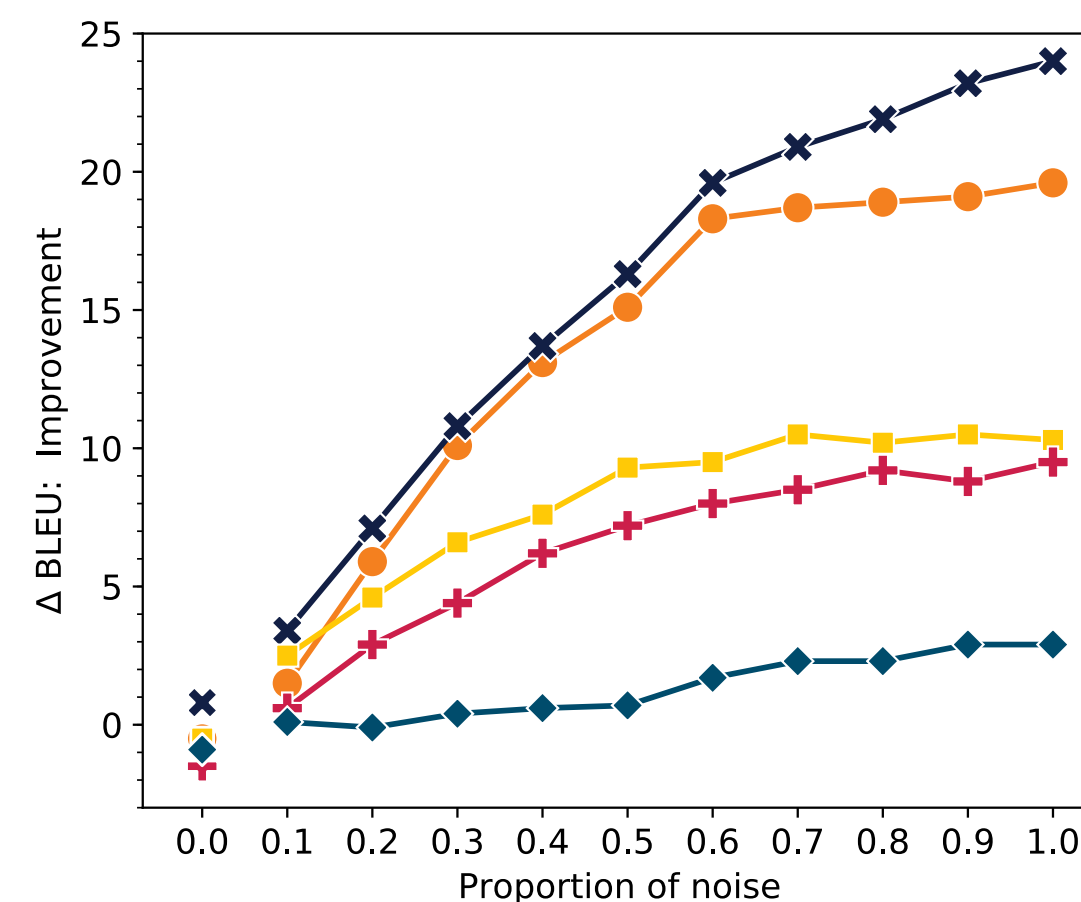


Korean: less noise applied as fewer tokens have length  $\geq 4$

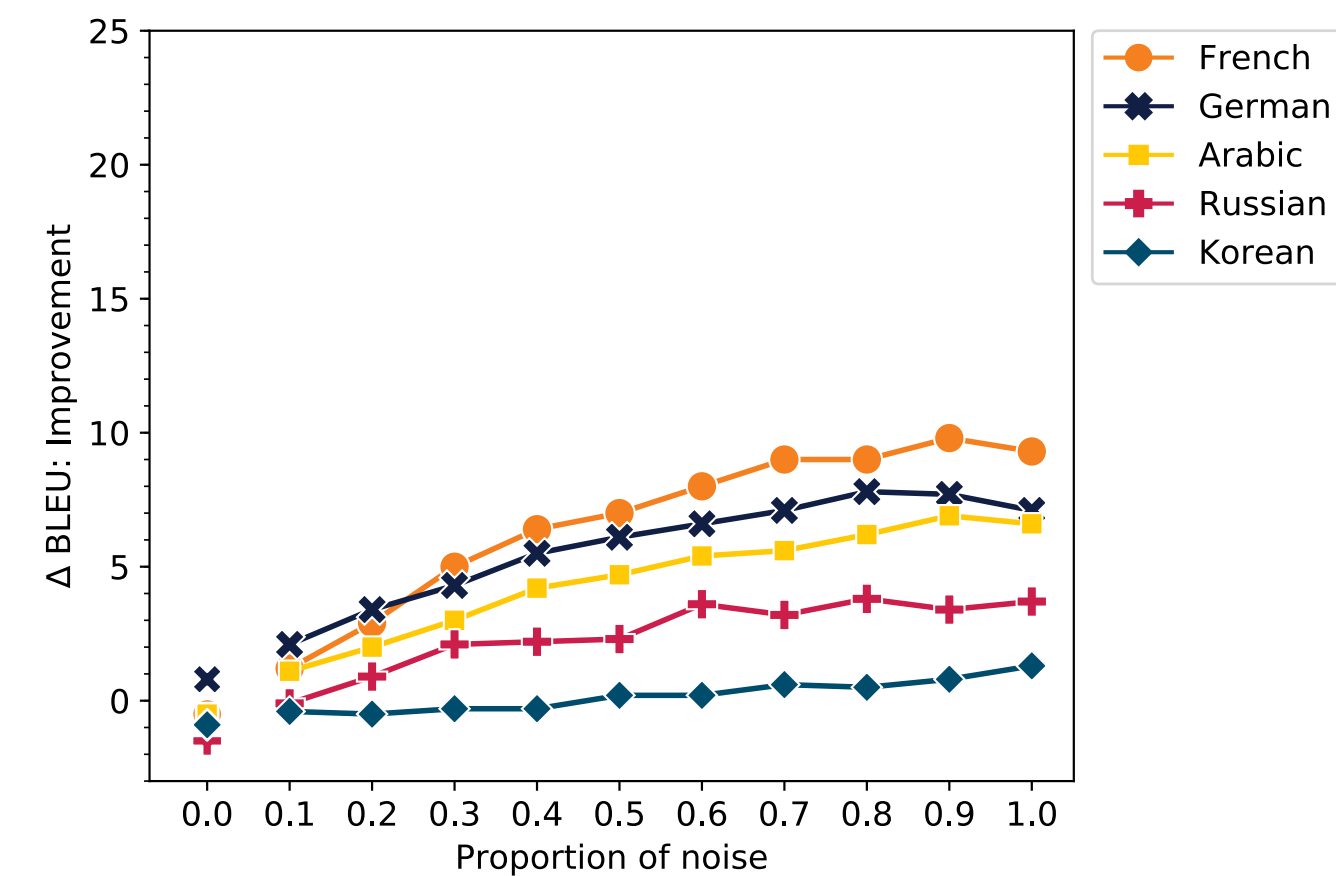
# Character permutations

- Significant improvements for all pairs, even if slight performance gap on clean text
  - Highlighting **German-English**:
    - At `swap p=1.0`, the visrep model is usable (25.9 BLEU) while the text model is not (1.9 BLEU)

swap



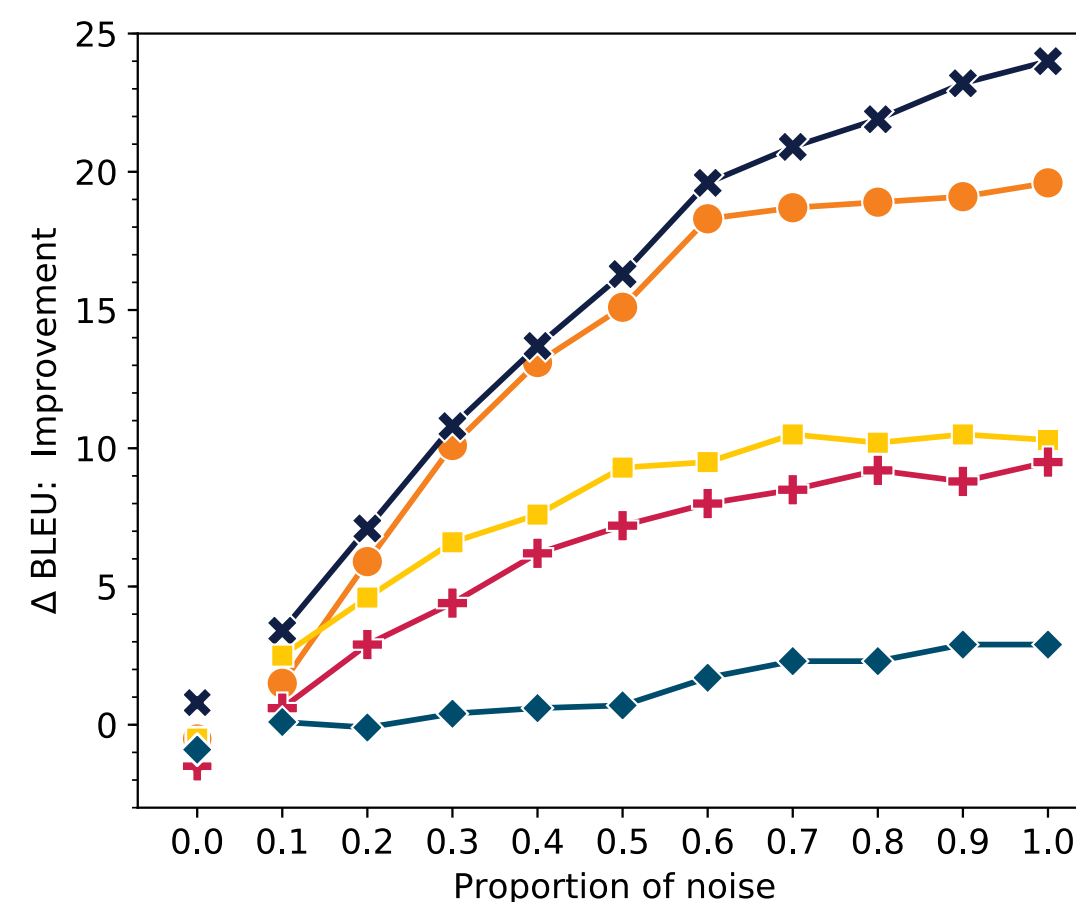
cmabrigde



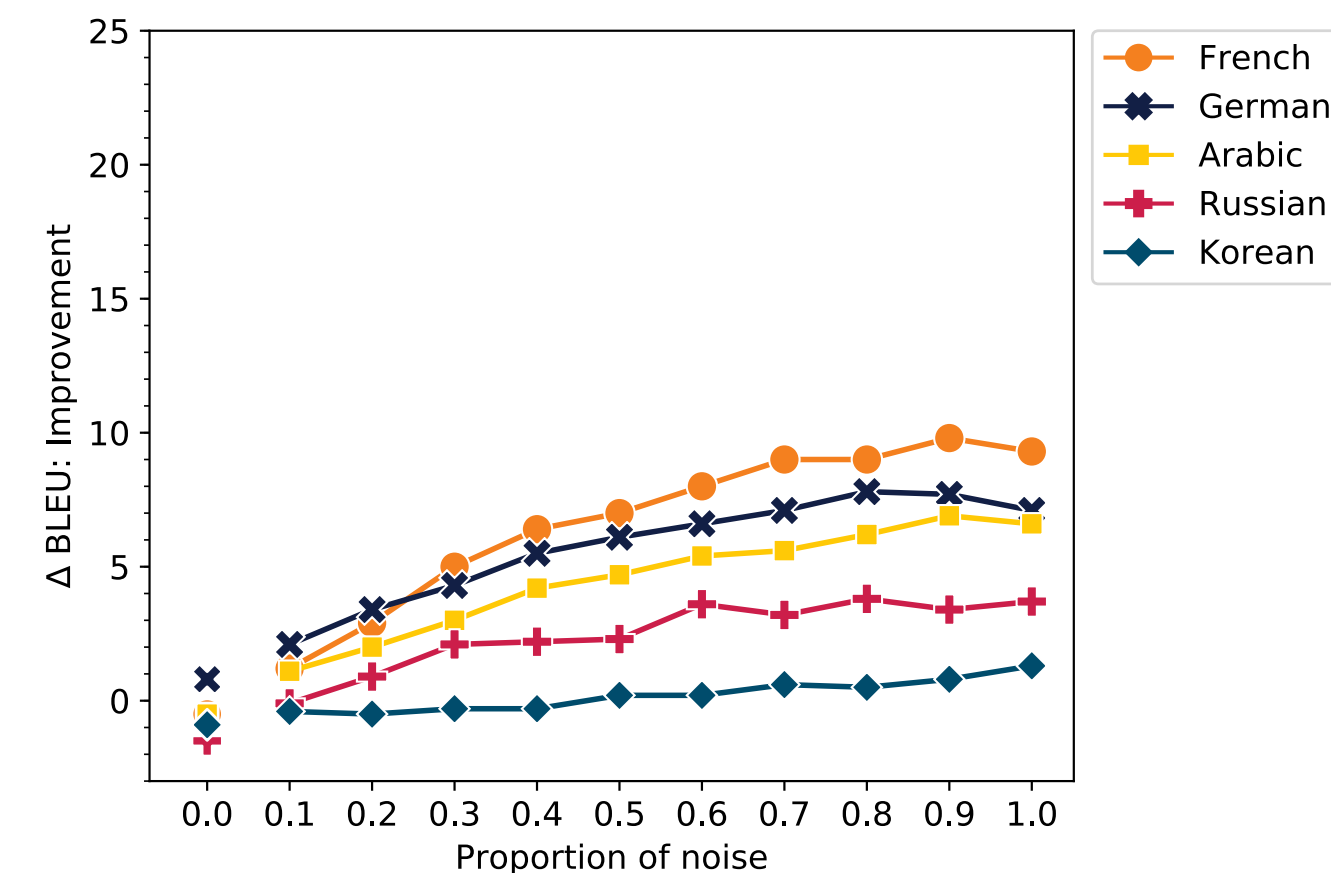
# Character permutations

- More improvement with more noise (opposite of 133t):
  - Previous types of noise shown are typically substitutions rather than permutations
  - Permutations affect more character sequence for a given token, shattering subword representations
    - At **swap**  $p=1.0$ , the German BPE model backs off to  $2.25\times$  more subwords than without noise

swap



cmabridge





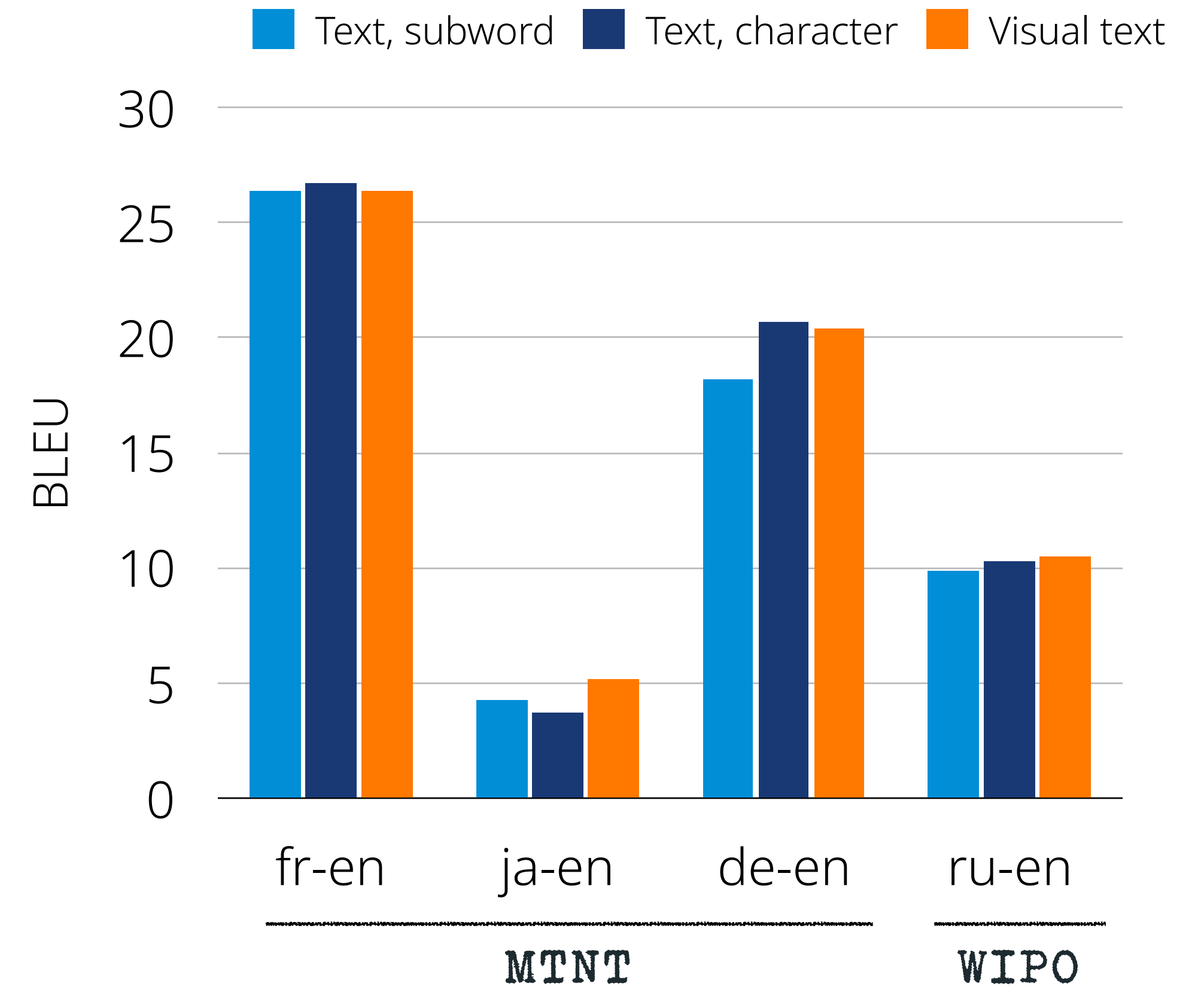
# Test sets with natural noise

- Natural noise contains many additional types of noise & in combination
  - Keyboard typos (where nearby keys are **substiyuted**)
  - Substitutions of phonetically-similar **characterz** or **worts**
  - Unconventional **s p a c e s** and **repetitionsss**
  - Natural **mispellings**
  - **...and more!**
- Parallel text created from ‘found’ data contains such noise in natural contexts
  - **MTNT**: Reddit (Michel et al. 2018); **WIPO**: patents (Junczys Dowmunt et al. 2016)



# Test sets with natural noise

- Evaluated in a zero-shot setting
  - 'Domain' is a confounding variable
- Character-level models are in some cases more robust than subwords
  - In others, unable to recover from variation (ja-en), where visual text does best
- Visual text improves over subwords and performs competitively with character-level text models



Improving text models

# Text models are brittle

Text models are not naturally robust, but can be improved!

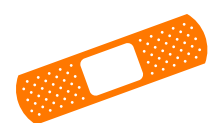


- Preprocessing techniques:
  - Use of normalization, spell-checkers



- Model regularization:
  - Subword regularization and BPE-dropout





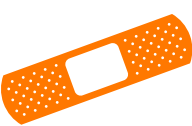
# Normalization

- What about normalization as preprocessing?
  - It helps text models, but selectively!

- While spell-checking helps, it:
  - is language-specific
  - is best suited to observed noise
  - relies on context to disambiguate:
    - noisy context hurts!

		Arabic		French		German		Korean		Russian	
		<i>BPE</i>	<i>visrep</i>	<i>BPE</i>	<i>visrep</i>	<i>BPE</i>	<i>visrep</i>	<i>BPE</i>	<i>visrep</i>	<i>BPE</i>	<i>visrep</i>
swap	no noise	<b>32.1</b>	31.6	<b>36.7</b>	36.2	33.6	<b>35.1</b>	<b>17.0</b>	16.6	<b>25.4</b>	25.0
	induced noise	2.3	<b>9.3</b>	2.4	<b>22.0</b>	1.9	<b>25.9</b>	5.4	<b>8.9</b>	5.4	<b>18.8</b>
	+ <i>spellcheck</i>	7.9	<b>11.9</b>	23.8	<b>29.1</b>	1.9	<b>14.1</b>	5.1	<b>6.9</b>	10.8	<b>18.2</b>
cambridge	induced noise	7.8	<b>13.2</b>	6.9	<b>18.3</b>	6.5	<b>16.9</b>	12.6	<b>14.1</b>	4.5	<b>11.1</b>
	+ <i>spellcheck</i>	10.9	<b>12.6</b>	16.4	<b>21.1</b>	10.0	<b>14.9</b>	10.3	<b>11.8</b>	5.9	<b>11.1</b>
l33tspeak	induced noise	—	—	0.3	<b>0.7</b>	0.7	<b>1.2</b>	—	—	—	—
	+ <i>spellcheck</i>	—	—	0.3	<b>0.7</b>	0.7	<b>1.2</b>	—	—	—	—
diacritics	induced noise	1.7	<b>25.2</b>	—	—	—	—	—	—	—	—
	+ <i>spellcheck</i>	2.1	<b>25.3</b>	—	—	—	—	—	—	—	—
unicode	induced noise	—	—	—	—	—	—	—	—	1.6	<b>22.0</b>
	+ <i>spellcheck</i>	—	—	—	—	—	—	—	—	2.1	<b>20.4</b>

Table 11: Translation performance on five types of induced noise with spellchecking as preprocessing; all test sets have noise induced with  $p = 1.0$ . Both traditional text models (*BPE*) and visual text models (*visrep*) are shown. We bold the best model for each condition.



# Normalization

Noise, with and without spellcheck

Not a perfect fix!

- What do we see?
  - Spellcheck generally helps BPE models...
    - but also visrep models!
- Spellcheck doesn't help all languages equally
  - See: German BPE vs French BPE, swap
- Spellcheck doesn't help all noise equally
  - See: l33tspeak
- Spellcheck can also *create* errors

		Arabic		French		German		Korean		Russian	
		BPE	visrep	BPE	visrep	BPE	visrep	BPE	visrep	BPE	visrep
swap	no noise	<b>32.1</b>	31.6	<b>36.7</b>	36.2	33.6	<b>35.1</b>	<b>17.0</b>	16.6	<b>25.4</b>	25.0
	induced noise	2.3	<b>9.3</b>	2.4	<b>22.0</b>	1.9	<b>25.9</b>	5.4	<b>8.9</b>	5.4	<b>18.8</b>
	+ spellcheck	7.9	<b>11.9</b>	23.8	<b>29.1</b>	1.9	<b>14.1</b>	5.1	<b>6.9</b>	10.8	<b>18.2</b>
cambridge	induced noise	7.8	<b>13.2</b>	6.9	<b>18.3</b>	6.5	<b>16.9</b>	12.6	<b>14.1</b>	4.5	<b>11.1</b>
	+ spellcheck	10.9	<b>12.6</b>	16.4	<b>21.1</b>	10.0	<b>14.9</b>	10.3	<b>11.8</b>	5.9	<b>11.1</b>
l33tspeak	induced noise	—	—	0.3	<b>0.7</b>	0.7	<b>1.2</b>	—	—	—	—
	+ spellcheck	—	—	0.3	<b>0.7</b>	0.7	<b>1.2</b>	—	—	—	—
diacritics	induced noise	1.7	<b>25.2</b>	—	—	—	—	—	—	—	—
	+ spellcheck	2.1	<b>25.3</b>	—	—	—	—	—	—	—	—
unicode	induced noise	—	—	—	—	—	—	—	—	1.6	<b>22.0</b>
	+ spellcheck	—	—	—	—	—	—	—	—	2.1	<b>20.4</b>

Table 11: Translation performance on five types of induced noise with spellchecking as preprocessing; all test sets have noise induced with  $p = 1.0$ . Both traditional text models (*BPE*) and visual text models (*visrep*) are shown. We bold the best model for each condition.



# Subword Regularization / BPE-Dropout

- Subword regularization techniques often improve performance and robustness
  - *Are the improvements similar to with visual text representations?*

• Recall BPE:

u-n-r-e-l-a-t-e-d  
u-n re-l-a-t-e-d  
u-n re-l-at-e-d  
u-n re-l-at-ed  
un re-l-at-ed  
un re-l-ated  
un rel-ated  
un-related  
unrelated

BPE-dropout:

u-n-r-e-l-a-t-e-d  
u-n re-l-a-t-e-d  
u-n re-l-at-e-d  
un re-l-at-e-d  
un re-l-at-ed  
un re-lat-ed  
un relat-ed

u-n-r-e-l-a-t-e-d  
u-n re-l-a-t-e-d  
u-n re-l-at-e-d  
u-n re-l-ate-d  
u-n rel-ate-d  
u-n relate-d

Different subword set with the same  
(overall) number of merges

Appendix G

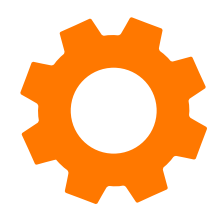
# Subword Regularization / BPE-Dropout

- BPE-Dropout ([Provilkov et al. 2020](#)):
  - *Subword segmentation using BPE algorithm*
  - *'Drop' candidate merges with some probability, and train with different segmentations each epoch*
    - *NOTE: small number of resulting subwords will not be in the MT model's vocabulary*
- Subword Regularization ([Kudo, 2018](#)):
  - *Subword segmentation using unigram LM probabilities*
  - *Can draw a stack of  $\ell$  candidates, and use different candidate segmentations each epoch*
    - *{\_hell o, \_h ello, \_he llo, \_h e l l o, \_h el l o}*

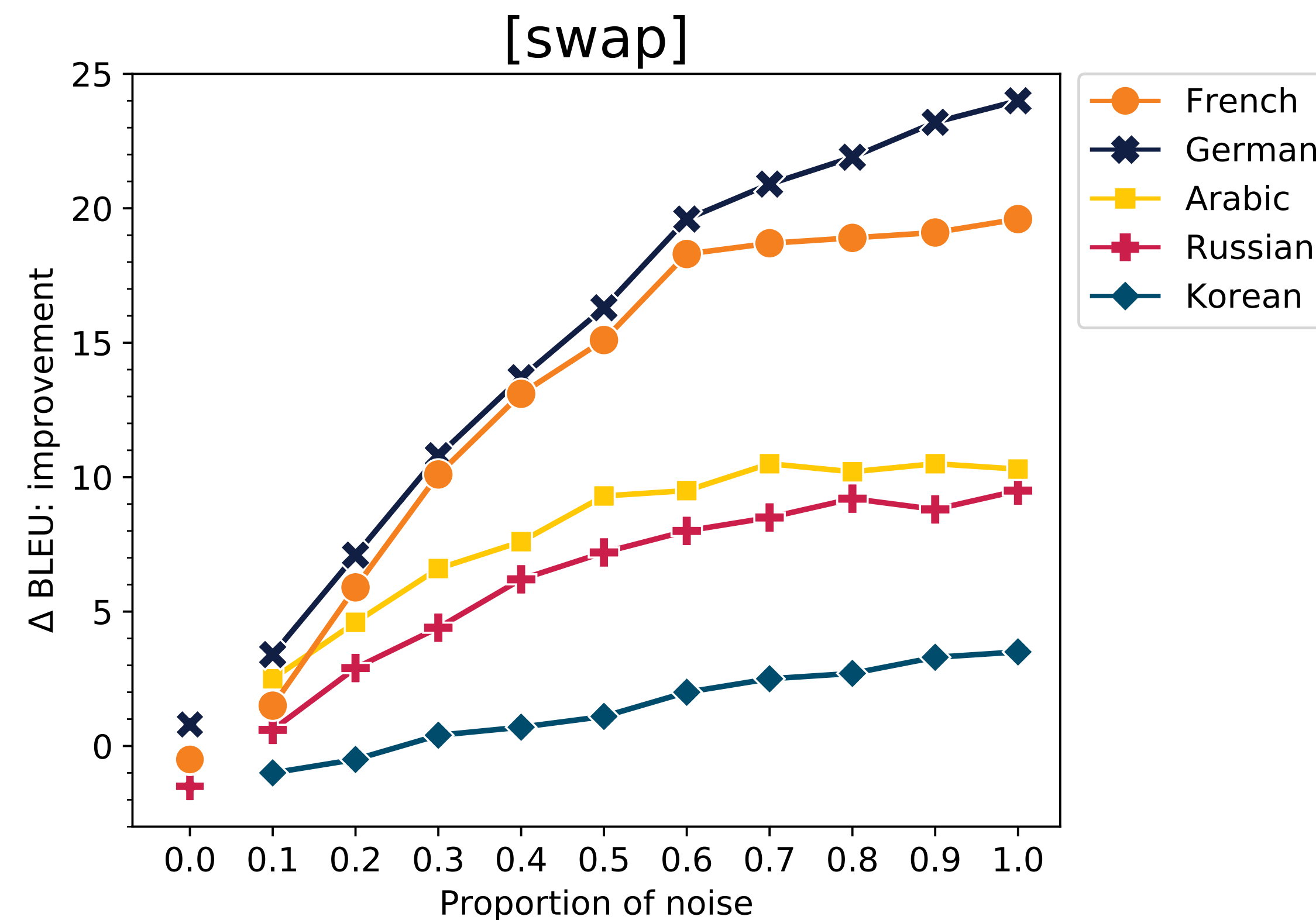
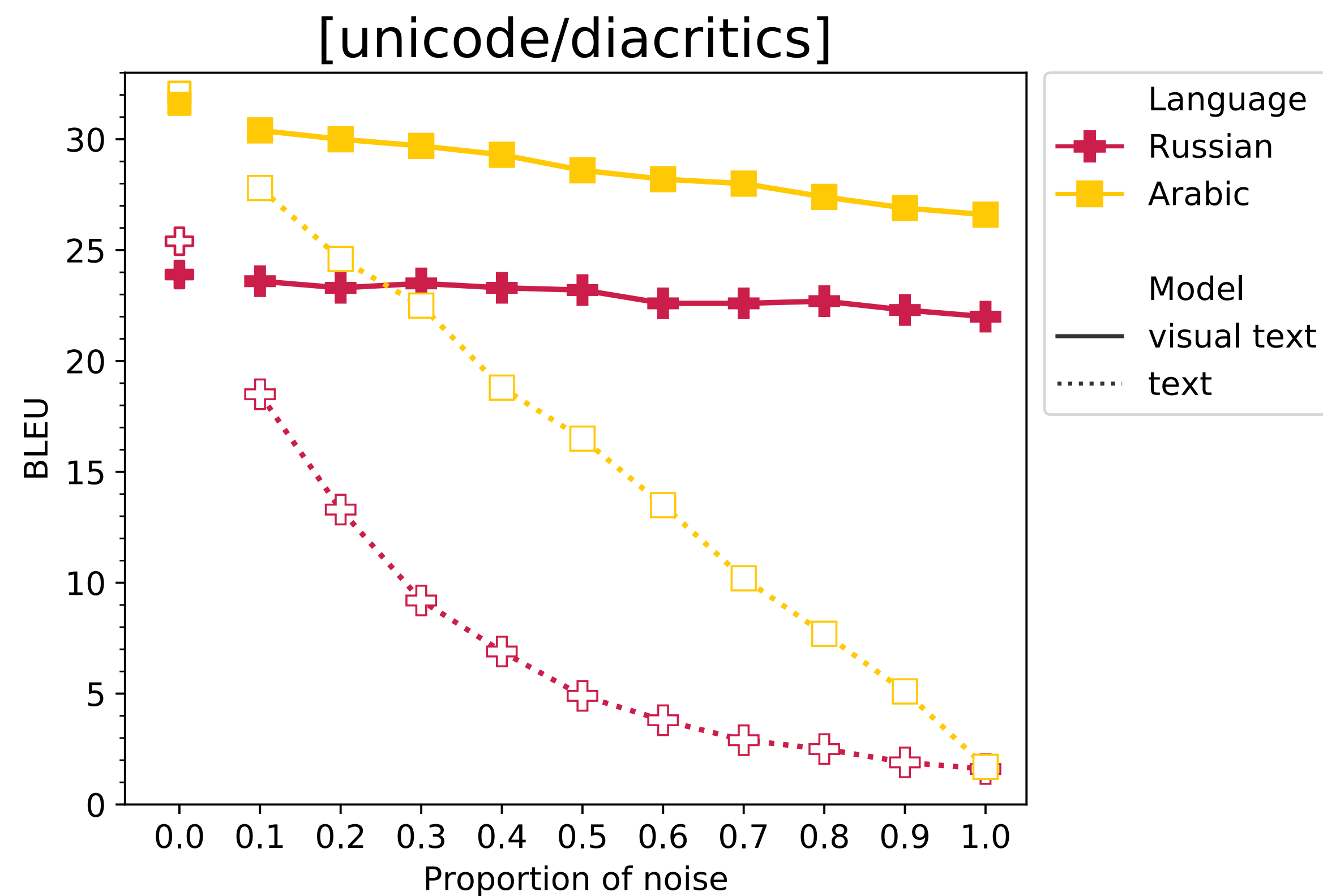


# Subword Regularization / BPE-Dropout

- Subword regularization techniques often improve performance and robustness
  - *Are the improvements similar to with visual text representations?*
    - > *In short, no.*
- Both techniques provide strong improvements over BPE (or character) models alone
  - Subword Regularization ([Kudo, 2018](#)) improved performance on both clean (0-2 BLEU) and noisy text (0-5 BLEU)
  - BPE-Dropout ([Provilkov et al. 2020](#)) further improved performance on clean (0.2-3 BLEU) and noisy text (0-9 BLEU)
- Visrep models remain more robust, though their base performance is lower than text models with regularization

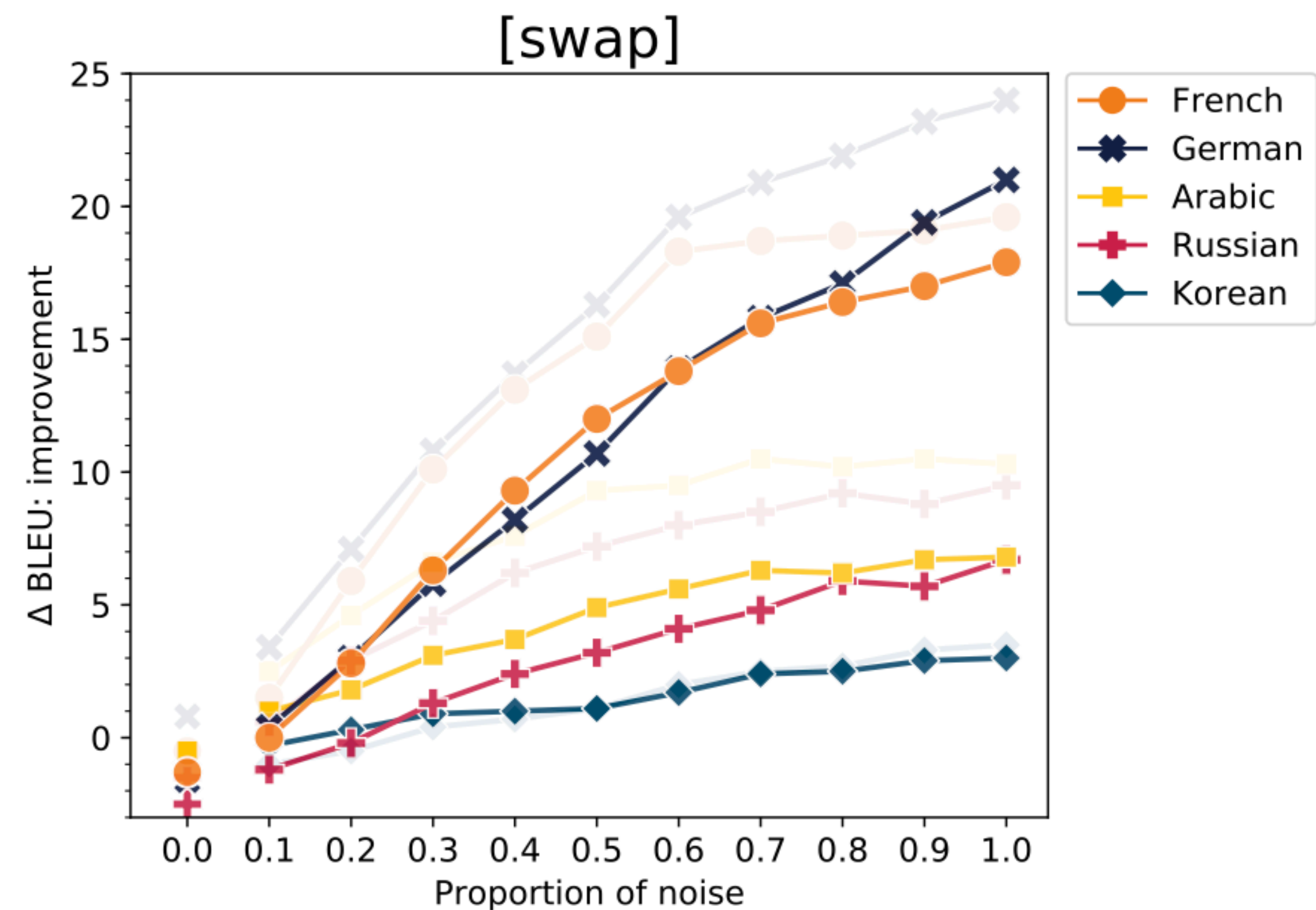
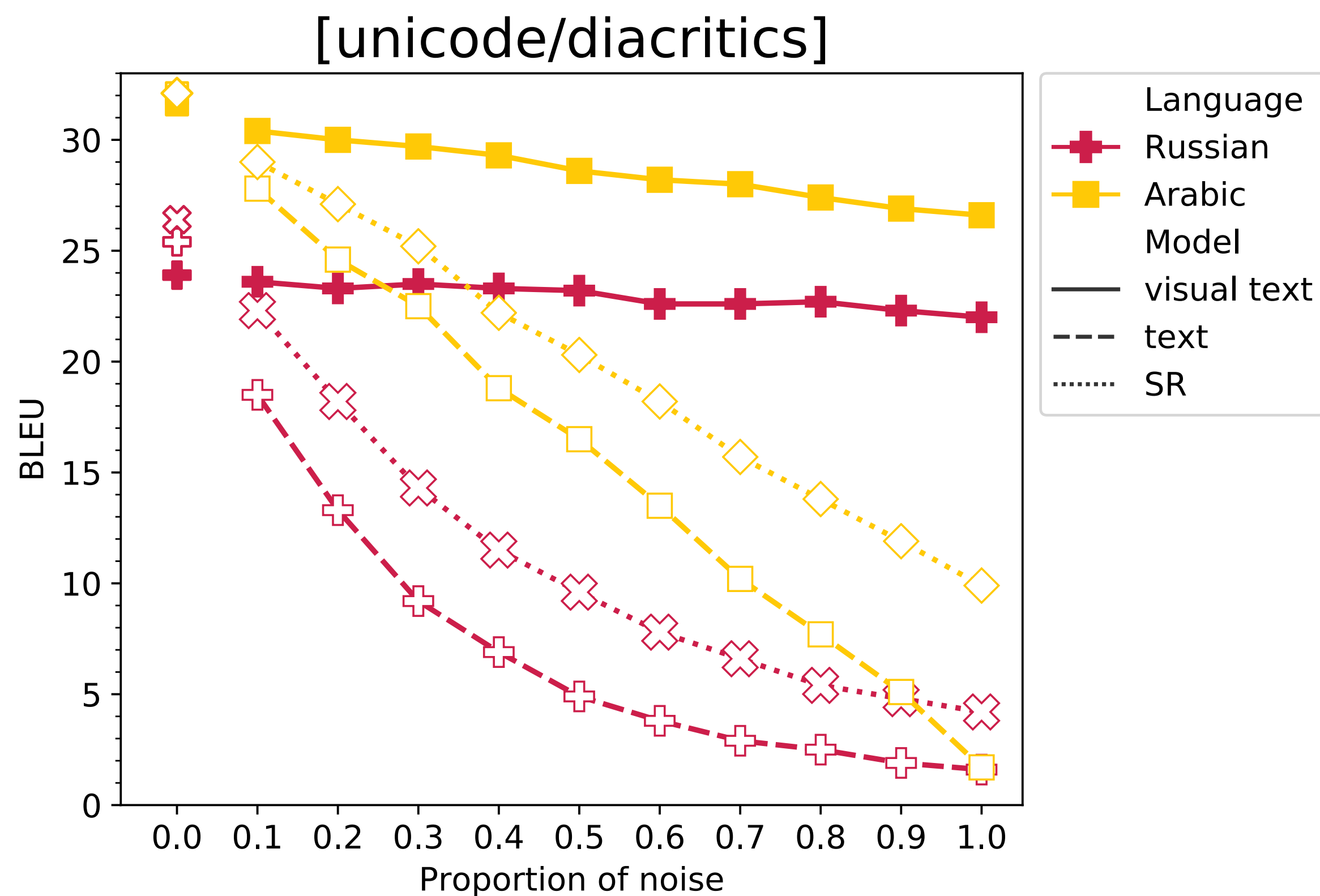


# Initial comparison



Improvement over standard BPE model

# Subword Regularization

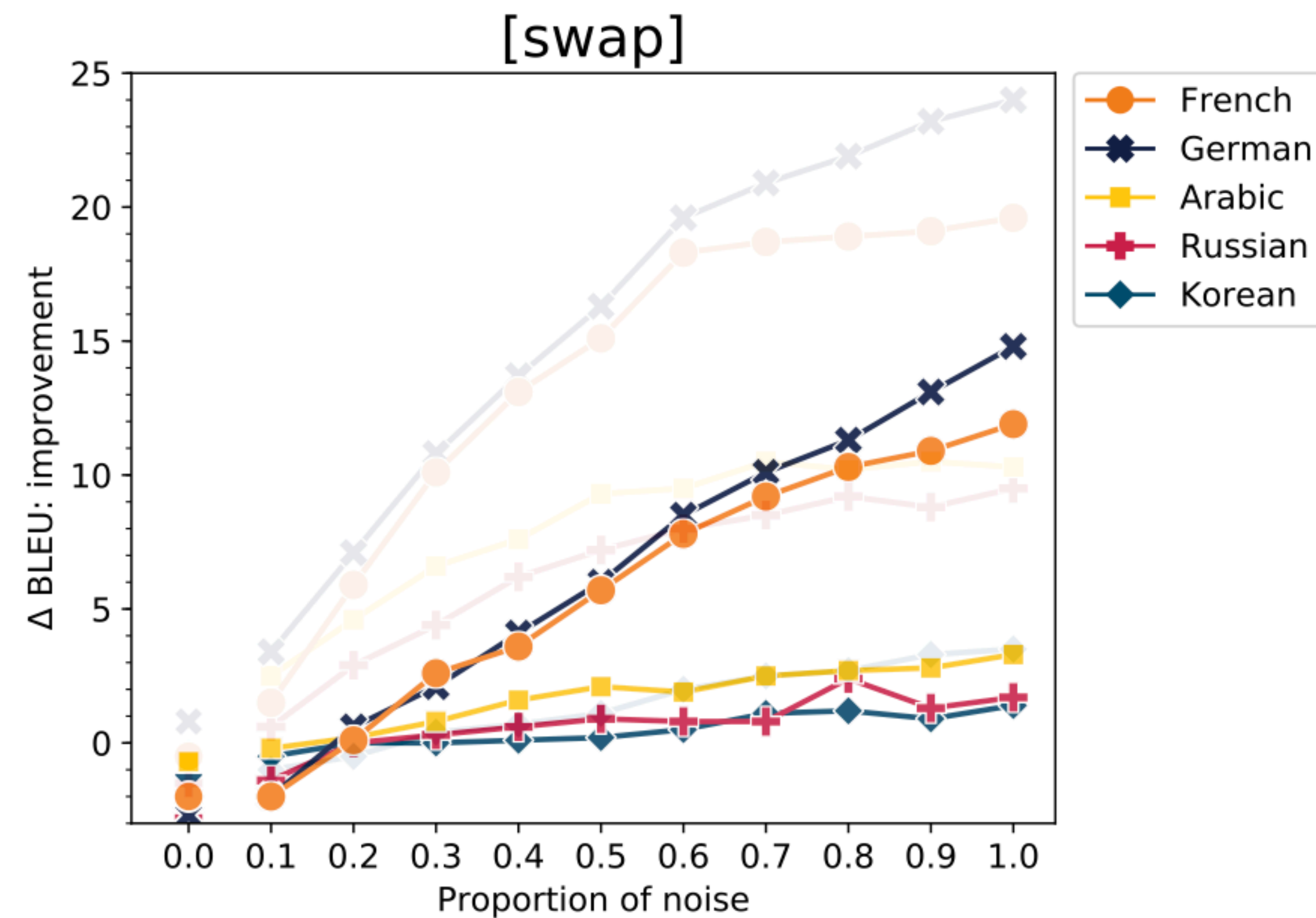
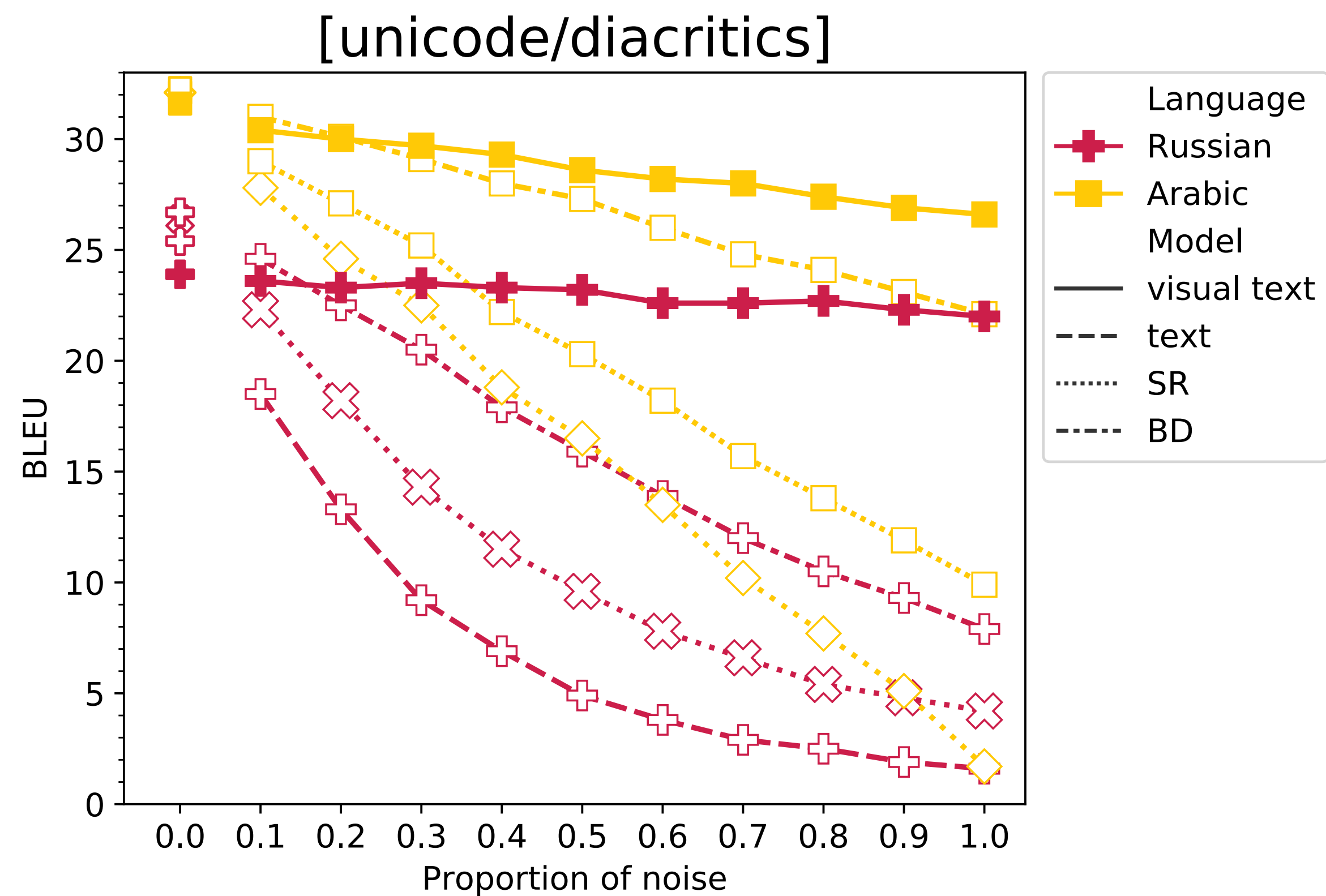


Improvement over stronger subword regularization baseline,  
compared to over standard BPE model (background)





# BPE Dropout



Improvement over stronger BPE dropout baseline,  
compared to over standard BPE model (background)



# Open Questions & Future Work

# Next steps

$x \rightarrow x$

- Pretraining or data augmentation
  - All robustness results are zero-shot, without training on noise

$xy \ z$

- Segmentation
  - Sliding windows inspired by speech recognition: work, but, may not be optimal!

$xyz$

- Target-side visual representations
  - Challenging! ([Mansimov et al. 2020](#))
  - Introduces evaluation complications; robustness typically a source-side problem

# Future directions

- Transfer learning

- [illegible]



y  
z

- Other NLP tasks

- Language ID ([Caswell et al. 2020: Table 2](#))

Pred. Language	Mined “Sentence” purporting to be in this language	Noise class
Manipuri	👤 🍌 🙋 🧏 🗿	General noise
Twi (Akan)	me: why you lyyɪn , why you always lyyɪn	General noise
Varhadi	Òyǎèè èè, áóðà- éýòëÿdö ýàèè ìeáí Éääóá löyyəèì öyi̯-. yǎ́á-yðêáiú èëý áó íŷñē [...]	Misrendered PDF
Aymara	Orilyzewuhubys ukagupixog axiqyh asozasuh uxilutidobyq osoqalelohan [...]	Non-Unicode font
Balinese	As of now 𑜁𑜨𑜃𑜫𑜀𑜂𑜆𑜄𑜐𑜫𑜇𑜨𑜊𑜦𑜰𑜫 is verified profile on Instagram.	Boilerplate
Cherokee	“ALL mY IhΘRΛS GREW bACK As fLOWERs ” . . . SWEET BʒBIES n DUGS	Creative use of Unicode
Oromo	My geology essay introduction essay on men authoring crosswords	Unlucky frequent n-gram
Pular	MEEEOW	Repeated n-grams
Chechen	Жи рновский ... Жи рновский рай онный Фе стив аль То Со в	A N T S P E A K
Kashmiri	ਸ਼ਾ.	Short/ambiguous
Nigerian Pidgin	This new model features a stronger strap for a secure fit and increased comfort.	High-resource cousin
Uyghur	نۇرسۇلتان نازاربايەۆ قىتاي دىڭ قازاقستانداغى دلشېسىمەن	Out-of-model cousin
Dimli	The S</><b class='b2'>urina</b><b class='b1'>m toa</b><b class='b3'>d is [...]	Deliberately Obfuscated

# Conclusions

- Discussed potential issues with unicode-driven text representations
- Explored an alternate approach, using visual text representations
  - Representations learned jointly with the target task (here, machine translation)
- Visual text representations are truly open-vocabulary
  - No fixed, predetermined model vocabulary
  - More robust than common unicode-based models to many types of induced noise
- Next steps
  - Potential benefits for more languages, settings, and tasks
  - Both ways to improve these models, and also limitations yet to be discovered



# Questions?



Code



Paper



Feel free to message  
the RocketChat channel,  
or email me at  
[esalesky@jhu.edu](mailto:esalesky@jhu.edu)