Modern, robust machine translation.

Nikolay Bogoychev Graeme Nail Jelmer van der Linde

University of Edinburgh
n.bogoych@ed.ac.uk
gnail@ed.ac.uk
jelmervanderlinde@ed.ac.uk

Building machine translations systems

It's easy as 1, 2, 3:

Building machine translations systems

It's easy as 1, 2, 3:

1. Download parallel data (easy-ish)

Building machine translations systems

It's easy as 1, 2, 3:

1. Download parallel data (easy-ish)
2. Clean data (not easy)

Building machine translations systems

It's easy as 1, 2, 3:

1. Download parallel data (easy-ish)
2. Clean data (not easy)
3. Train neural network

Building machine translations systems

It's easy as 1, 2, 3:

1. Download parallel data (easy-ish)
2. Clean data (not easy)
3. Train neural network

Then why is it so difficult to produce good systems?

Table of contents

Briefly, data acquisition, cleaning and OpusCleaner

Data sources

Where do we even start?

- MT Data
- Opus

Downloading importing all of that is kind'a of mess

OpusCleaner data fetch & clean

DEMO

Training for robustness and real world use cases

What are real world usecases

- SHOUTING
- TittleCase
- Wikipedia
- Twitter (All caps, no accents, 1337 SP34K, typos)
- Emoji
- urls..

Our users care about those, but we only care about newstestXX

Let's test on French-English

- Get all of the data (OpusCleaner)
- Clean it (OpusCleaner)
- concat and Train...

But data is imbalanced

Data size

- Clean:
- Almost clean
- Somewhat clean
- Dirty

Results

How do we evaluate? Newstest2015 is a good start, but it's not representative of real world usecases.

Results

How do we evaluate? Newstest2015 is a good start, but it's not representative of real world usecases.

Generate our own dirty datasets!

Results

How do we evaluate? Newstest2015 is a good start, but it's not representative of real world usecases.

Generate our own dirty datasets!

- Title Case Test Set
- ALL CAPS TEST SET
- Tset set wiht tipos
- Test set with /wɪθ/ noise /nɔɪz/ in the middle

Real world example

Demo

Basic model

Train vocabulary on clean data and train.

	newstest15 BLEU				
	Plain	TitleCase	TittleCase strict	CAPS	CAPS strict
baseline	40	34.2	8.6	21.5	20.5

Basic model

Train vocabulary on clean data and train.

	newstest15 BLEU					
	Plain	Typo aug	Noise aug	Noise chrf	emoji aug	emoji chrf
baseline	40	29.6	34.3	0	35.8	0.1

Basic model + spm sampling

Why do we fail on Title Case and Upper Case? We don't see the splits enough.

IDEA: Use SPM sampling during training.

Basic model + spm sampling

SPM sampling during training.

	newstest15 BLEU				
	Plain	TitleCase	TittleCase strict	CAPS	CAPS strict
baseline	40	34.2	8.6	21.5	20.5
+ spm sample	39	**36.9**	9.1	**29.2**	21.2

Basic model + spm sampling

SPM sampling during training.

	newstest15 BLEU					
	Plain	Typo aug	Noise aug	Noise chrf	emoji aug	emoji chrf
baseline	40	29.6	34.3	0	35.8	0.1
+ spm sample	39	30.5	33.4	0.1	34.7	0.2

Noise the training data

Augment the model with Tittle Case and ALLCAPS data:

- **10%** of data is turned to Tittle Case
- **10%** of data is turned to ALLCAPS

TittleCase and UPPERCASE noise

Training data with TittleCase and UPPERCASE noise.

	newstest15 BLEU				
	Plain	TitleCase	TittleCase strict	CAPS	CAPS strict
baseline	40	34.2	8.6	21.5	20.5
+ spm sample	39	36.9	9.1	29.2	21.2
+ UC/LC Noise	38.4	37.3	**36.3**	34.5	**34.5**

TittleCase and UPPERCASE noise

Training data with TittleCase and UPPERCASE noise.

	newstest15 BLEU					
	Plain	Typo aug	Noise aug	Noise chrf	emoji aug	emoji chrf
baseline	40	29.6	34.3	0	35.8	0.1
+ spm sample	39	30.5	33.4	0.1	34.7	0.2
+ UC/LC Noise	38.4	29.7	32.9	0.1	34.3	0.2

Add typos to the training data

Augment the model with typos

- Random typo applied to **10%** of the sentences

Typos during training

Training with typos in the training data.

	newstest15 BLEU				
	Plain	TitleCase	TittleCase strict	CAPS	CAPS strict
baseline	40	34.2	8.6	21.5	20.5
+ spm sample	39	36.9	9.1	29.2	21.2
+ UC/LC Noise	38.4	37.3	36.3	34.5	34.5
+ typos	38.9	38	36.8	35.1	35.1

Typos during training

Training with typos in the training data.

	newstest15 BLEU					
	Plain	Typo aug	Noise aug	Noise chrf	emoji aug	emoji chrf
baseline	40	29.6	34.3	0	35.8	0.1
+ spm sample	39	30.5	33.4	0.1	34.7	0.2
+ UC/LC Noise	38.4	29.7	32.9	0.1	34.3	0.2
+ typos	38.9	**36.7**	33.5	0.1	34.2	5.2

Unicode aware vocab

What about emoji? Chinese characters? Arabic text?

- Use unicode backoff for UNKs

Unicode aware vocab

Model with unicode aware vocab.

	newstest15 BLEU				
	Plain	TitleCase	TittleCase strict	CAPS	CAPS strict
baseline	40	34.2	8.6	21.5	20.5
+ spm sample	39	36.9	9.1	29.2	21.2
+ UC/LC Noise	38.4	37.3	36.3	34.5	34.5
+ typos	38.9	38	36.8	35.1	35.1
+ utf-8 backoff	38.5	38	36.8	34.7	34.7

Unicode aware vocab

Model with unicode aware vocab.

	newstest15 BLEU					
	Plain	Typo aug	Noise aug	Noise chrf	emoji aug	emoji chrf
baseline	40	29.6	34.3	0	35.8	0.1
+ spm sample	39	30.5	33.4	0.1	34.7	0.2
+ UC/LC Noise	38.4	29.7	32.9	0.1	34.3	0.2
+ typos	38.9	36.7	33.5	0.1	34.2	5.2
+ utf-8 backoff	38.5	36.8	35.2	**55.1**	37	**64.9**

Explicit model noising

We want the model to learn to copy OOV noise

- Add random noisy sentences to the data
- တုဓေါ်◌ះး えびそそだぃげご চৈর৯ ฦก တုဓေါ်◌ះး えびそそだぃげご চৈর৯ ฦก

Unicode noise

Explicit unicode noise

	newstest15 BLEU				
	Plain	TitleCase	TittleCase strict	CAPS	CAPS strict
baseline	40	34.2	8.6	21.5	20.5
+ spm sample	39	36.9	9.1	29.2	21.2
+ UC/LC Noise	38.4	37.3	36.3	34.5	34.5
+ typos	38.9	38	36.8	35.1	35.1
+ utf-8 backoff	38.5	38	36.8	34.7	34.7
+ noise	39.6	39.1	37.9	35.9	35.9

Unicode noise

Explicit unicode noise

	newstest15 BLEU					
	Plain	Typo aug	Noise aug	Noise chrf	emoji aug	emoji chrf
baseline	40	29.6	34.3	0	35.8	0.1
+ spm sample	39	30.5	33.4	0.1	34.7	0.2
+ UC/LC Noise	38.4	29.7	32.9	0.1	34.3	0.2
+ typos	38.9	36.7	33.5	0.1	34.2	5.2
+ utf-8 backoff	38.5	36.8	35.2	55.1	37	64.9
+ noise	39.6	37.6	38.9	**87**	38.7	**72.3**

In context model noising

But what about noise occurring in context?

- Use word alignment information to place noise in corresponding places.
- `C'est autre 😇😢 chose, bien plus profond.  This is something 😇😢 else, much deeper.`
- La loi ça うれざにゐべ sert à quoi? What うれざにゐべ use is the law
- C'est autre **蒙古自睽** chose, bien plus profond. This is something **蒙古自睽** else, much deeper.

Unicode noise

Explicit unicode noise inside genuine sentences during training.

	newstest15 BLEU				
	Plain	TitleCase	TittleCase strict	CAPS	CAPS strict
baseline	40	34.2	8.6	21.5	20.5
+ spm sample	39	36.9	9.1	29.2	21.2
+ UC/LC Noise	38.4	37.3	36.3	34.5	34.5
+ typos	38.9	38	36.8	35.1	35.1
+ utf-8 backoff	38.5	38	36.8	34.7	34.7
+ noise	39.6	39.1	37.9	35.9	35.9
+ context noise	39.2	38.3	37.2	35.3	35.3

Inline unicode noise

Explicit unicode noise inside genuine sentences during training.

	newstest15 BLEU					
	Plain	Typo aug	Noise aug	Noise chrf	emoji aug	emoji chrf
baseline	40	29.6	34.3	0	35.8	0.1
+ spm sample	39	30.5	33.4	0.1	34.7	0.2
+ UC/LC Noise	38.4	29.7	32.9	0.1	34.3	0.2
+ typos	38.9	36.7	33.5	0.1	34.2	5.2
+ utf-8 backoff	38.5	36.8	35.2	55.1	37	64.9
+ noise	39.6	37.6	38.9	87	38.7	72.3
+ context noise	39.2	37.5	**41.5**	**92**	**39.9**	**80.7**

It's not all fun and games

Our accuracy on URLs falls...

	URL testset			
	url bleu	url prec	url correct	url wrong
baseline	62.7	90%	1393	161
+ spm sample	61.4	87%	1354	200
+ UC/LC Noise	60.9	87%	1351	203
+ typos	61.2	86%	1331	223
+ utf-8 backoff	61	85%	1316	238
+ noise	61.3	86%	1331	223
+ context noise	61.2	86%	1336	218

What did we do it? How did we do it?

OpusTrainer

Demo: OpusTrainer and the Models

Background

Alternatives

Pixel based models

- Word Shape Matters: Robust Machine Translation with Visual Embedding (Wang et al. 2020)
- Robust Open-Vocabulary Translation from Visual Text Representations (Salesky et al. 2021)
- Language Modelling with Pixels (Rust et al. 2023)
- Pixel Representations for Multilingual Translation and Data-efficient Cross-lingual Transfer (Salesky et al. 2023)

Placeholding:

- An Exploration of Placeholding in Neural Machine Translation (Post et al. 2019)

Alternatives

Previous work on Noising:

- MTNT: A Testbed for Machine Translation of Noisy Text (Michel and Neubig, 2018)
- Training on Synthetic Noise Improves Robustness to Natural Noise in Machine Translation (Karpukhin et al. 2019)

Future work

- We still don't get everythig right
- URLs fail...
- Invalid UTF-8 Generation...

Conclusion

- We spend a lot of time and computation resources on gathering data.
- We throw away data that is *dirty*
- We throw away sentences with foreign language
- We run spellcheckers
- **We then expect to get translations right for all those cases**