CS772: Deep Learning for Natural Language Processing (DL-NLP)

NMT, Cross Attention

Pushpak Bhattacharyya
Computer Science and Engineering Department
IIT Bombay

Week 10 of 11mar24

Neural Machine Translation

Sourabh Deoghare, Pranav Gaikwad
CFILT, CSE@IIT Bombay

11th March 2024

Guide: Prof. Pushpak Bhattacharyya

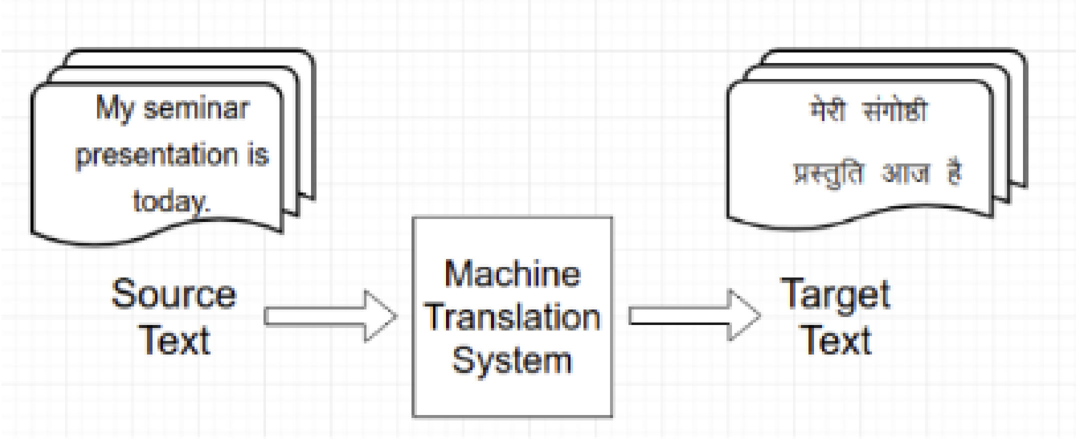
Topics To Be Discussed

- Introduction
- NMT: Encoder-Decoder Architecture
- Evaluation
- Data Filtering
- Data Augmentation
- Advanced NMT Approaches
- Demo

Introduction

What is Machine Translation (MT)?

- Automatic conversion of text from one language to another
  - Preserve the meaning
  - Fluent output text



My seminar presentation is today.

Source Text

Machine Translation System

मेरी संगोष्ठी प्रस्तुति आज है

Target Text

History of MT

- 1954: First public demo of MT by IBM
  - Georgetown IBM experiment
- 1956: First MT conference
- 1972: Logos MT system
  - Translating military manuals into Vietnamese
  - Rule based approach
- 1993: Statistical MT
  - IBM models
- 2013: Neural Machine Translation

Why MT is hard?

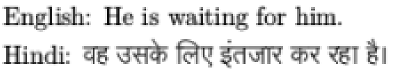
Language Divergence

Language divergence

- Languages express meaning in divergent ways
- **Syntactic divergence**
  - Arises because of the difference in structure
- **Lexical semantic divergence**
  - Arises because of semantic properties of languages

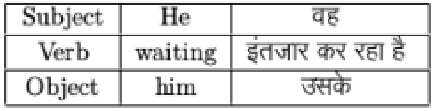
Different kinds of syntactic divergence

- Constituent order divergence (Word order)



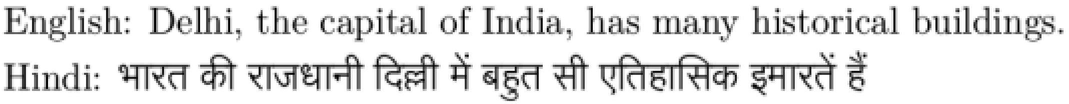
English: He is waiting for him.

Hindi: वह उसके लिए इंतजार कर रहा है।



Subject	He	वह
Verb	waiting	इंतजार कर रहा है
Object	him	उसके

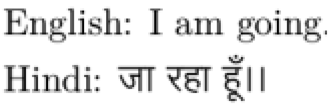
- Adjunction divergence



English: Delhi, the capital of India, has many historical buildings.

Hindi: भारत की राजधानी दिल्ली में बहुत सी एतिहासिक इमारतें हैं

- Null subject divergence



English: I am going.

Hindi: जा रहा हूँ।।

Different kinds of lexical semantic divergence

- Conflational divergence

English: He stabbed him.
Hindi: उसने उसे छुरे से मारा

- Categorial divergence (Lexical category change)

English: They are competing.
Hindi: वे प्रतिस्पर्धा कर रहे हैं

- Head-swapping divergence (Promotion or demotion of logical modifier)

English: The play is on.
Hindi: खेल चल रहा है

The Vauquois Triangle

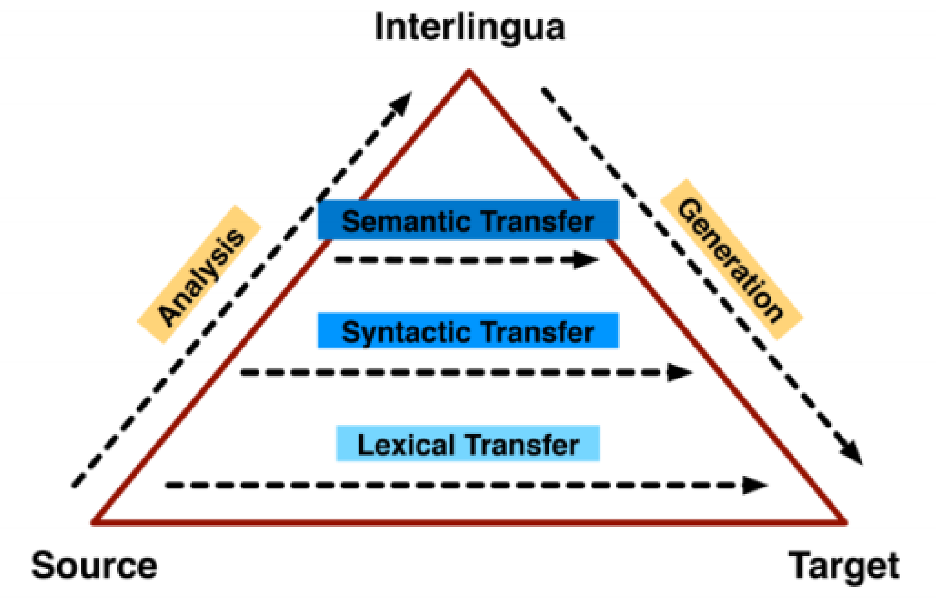
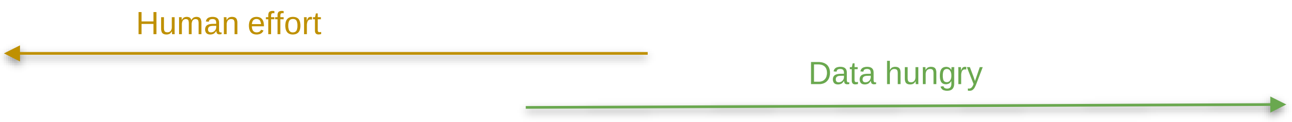


Image source: http://www.cs.umd.edu/class/fall2017/cmsc723/slides/slides15.pdf

Paradigms of Machine Translation


Human effort

Data hungry

RBMT → EBMT → SMT → NMT

(Rule based machine translation) (Example based machine translation) (Statistical machine translation) (Neural machine translation)

Rules handmade by human expert → Rules by learning from data

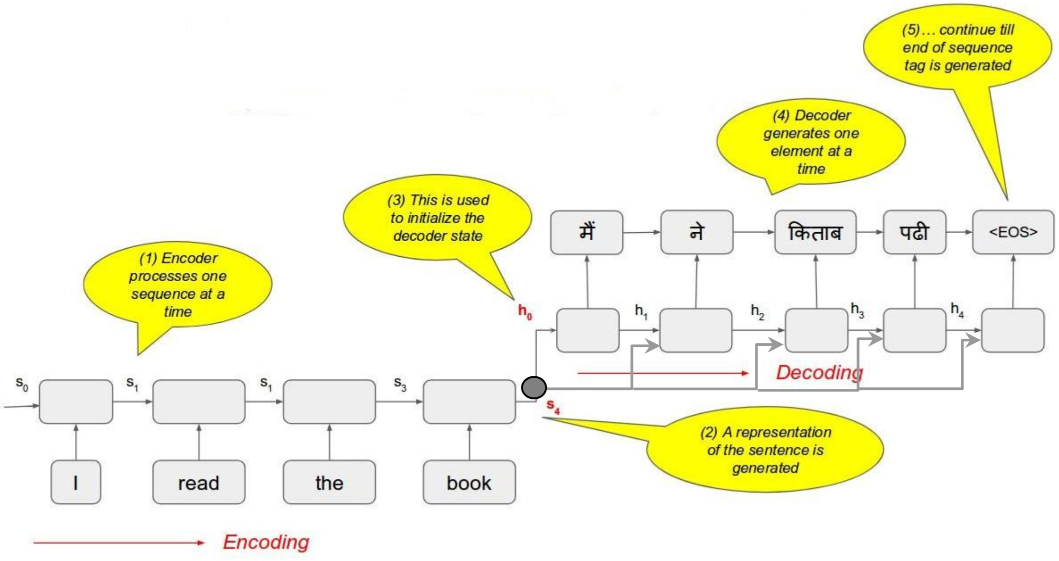
What is NMT?

- The task of MT is a sequence-to-sequence problem.
- It uses an encoder-decoder NN architecture with attention mechanism.
- NMT requires large parallel corpus.
- Here, we will discuss RNN-based and Transformer-based encoder-decoder architectures.

Why do we need NMT?

- In RBMT, EBMT, and SMT, there is no notion of similarity or relationship between symbolic representation of individual words.
- Ability to translate *I go to school* does not make these models capable of translating *I went to college*.
- However, Neural Network techniques work with distributed representations.
- NMT evaluates a single formula that explains all rules of the translation task. (Generalisation)

Simple RNN-based Encoder-Decoder Architecture


(5)… continue till end of sequence tag is generated
(4) Decoder generates one element at a time
(3) This is used to initialize the decoder state
(1) Encoder processes one sequence at a time
मैं
ने
किताब
पढी
<EOS>
h0
h1
h2
h3
h4
Decoding
s0
s1
s1
s3
s4
(2) A representation of the sentence is generated
I
read
the
book
Encoding

s_4 is the summary vector

Image source - [http://www.iitp.ac.in/~shad.pcs15/data/nmt-rudra.pdf]

Summary Vector Representation

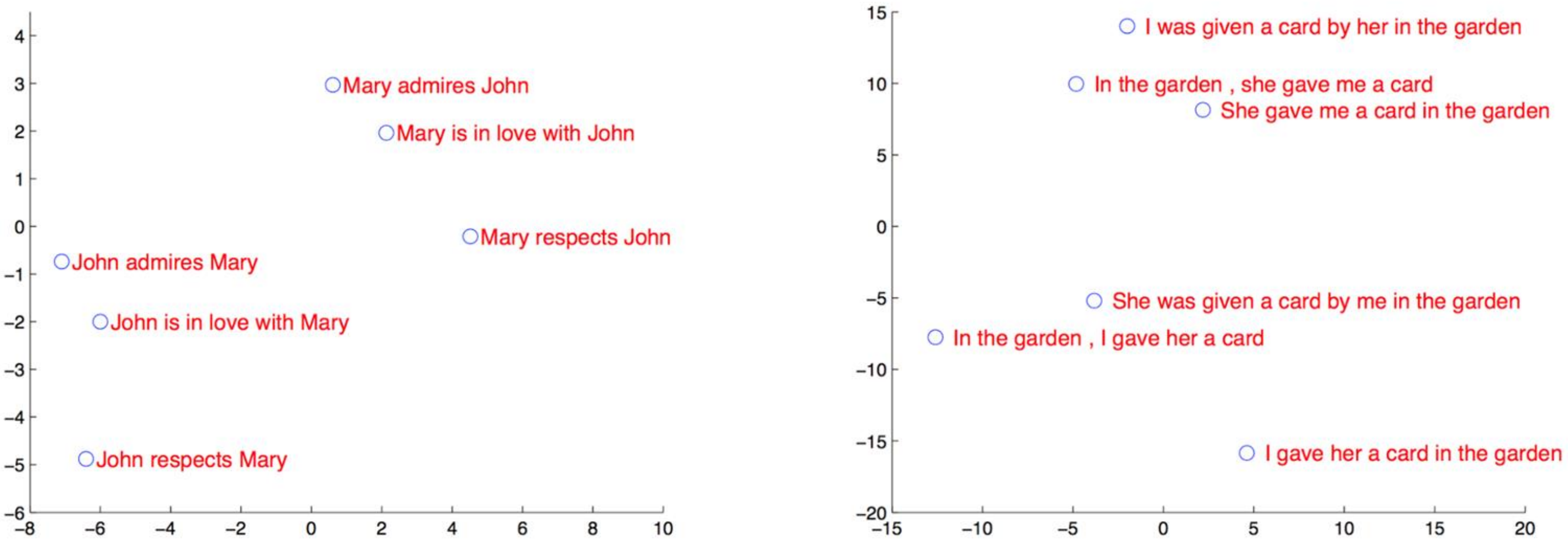


Image source- [9]

Problems with Simple Encode-Decode Paradigm (1/2)

What happens in enc-dec architecture?

1. Encoding transforms the entire sentence into a single vector.
2. Decoding process uses this sentence representation for predicting the output.

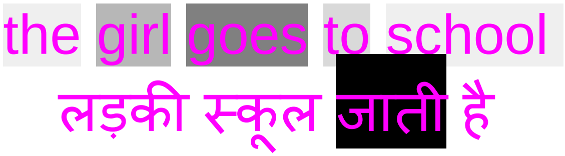
Problems:

- Quality of prediction depends upon the quality of sentence embeddings.
- After few time-step, summary vector may lose information of initial words of input sentence.

Problems with Simple Encode-Decode Paradigm (2/2)

Possible Solution:

- For prediction at each time step, present the representation of the relevant part of the source sentence only.

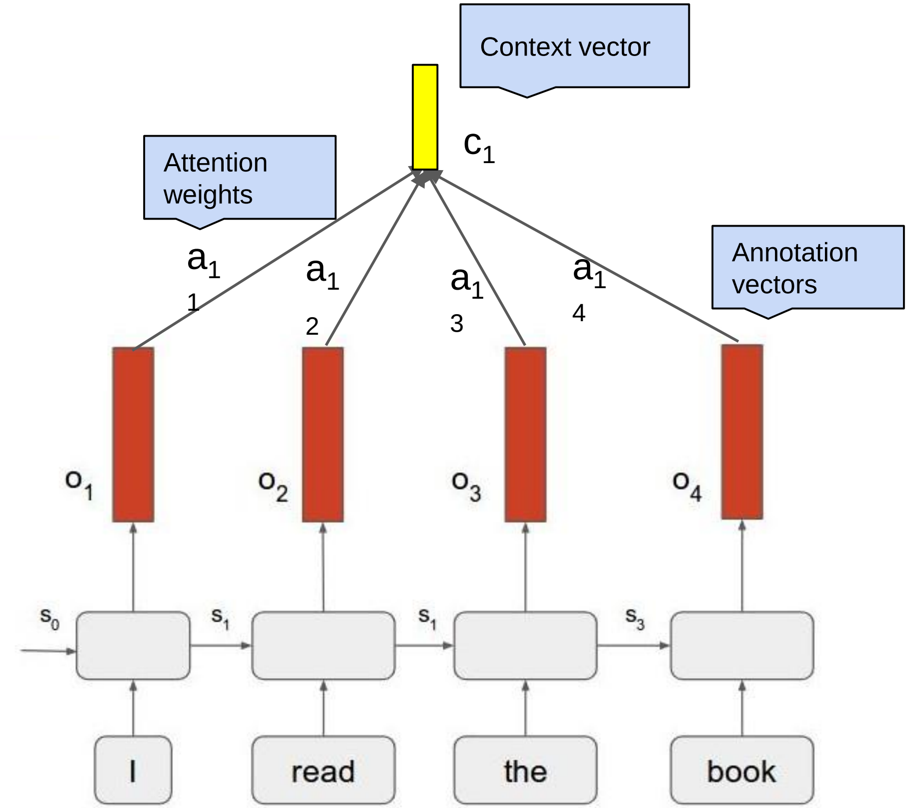


the girl goes to school

लड़की स्कूल जाती है

  - Attention-based encoder-decoder

Annotation Vectors and Context Vectors



Attention weights are calculated from alignment scores which are output of another feed-forward NN which is trained jointly.

Attention-based Encoder-Decoder Architecture (1/3)

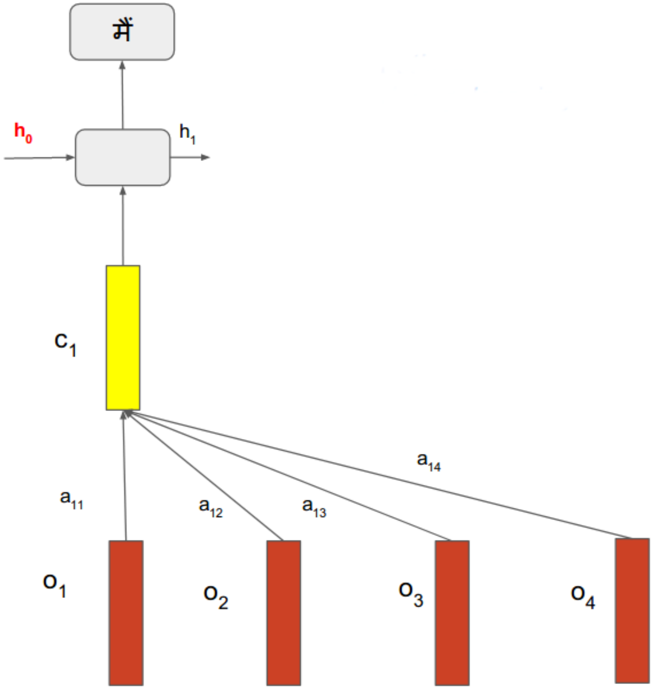


Image source- [http://www.iitp.ac.in/~shad.pcs15/data/nmt-rudra.pdf]

Attention-based Encoder-Decoder Architecture (2/3)

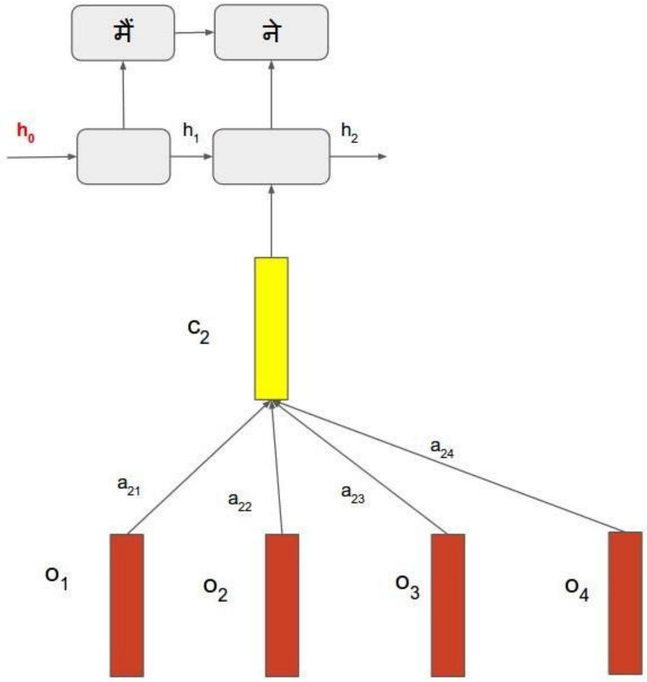


Image source- [http://www.iitp.ac.in/~shad.pcs15/data/nmt-rudra.pdf]

Attention-based Encoder-Decoder Architecture (3/3)

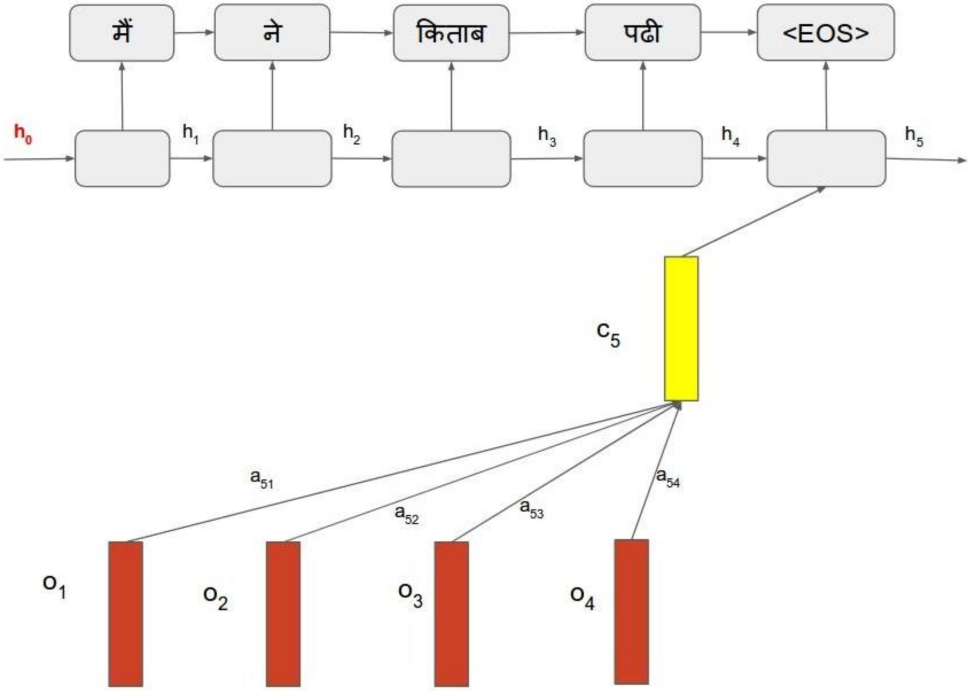


Image source- [http://www.iitp.ac.in/~shad.pcs15/data/nmt-rudra.pdf]

Transformer

- Motivations to choose Transformer over RNN:
  - Faster
  - More efficient.

- Architecture:
  - This is an encoder-decoder architecture with Transformers instead of RNNs.

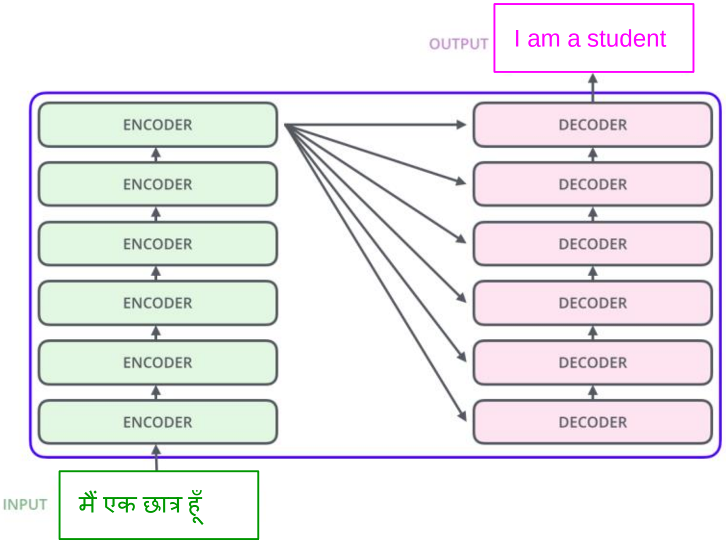


Image source: [jalammar.github.io/illustrated-transformer/]

Transformer: Embedding

- Embedding:
  - Input of the encoder = f(word_embedding, positional encoding)
  - To set a constant and small vector_size of positional encoding, researchers apply a strategy using sinusoidal function for which model can translate long sentences of the training set.

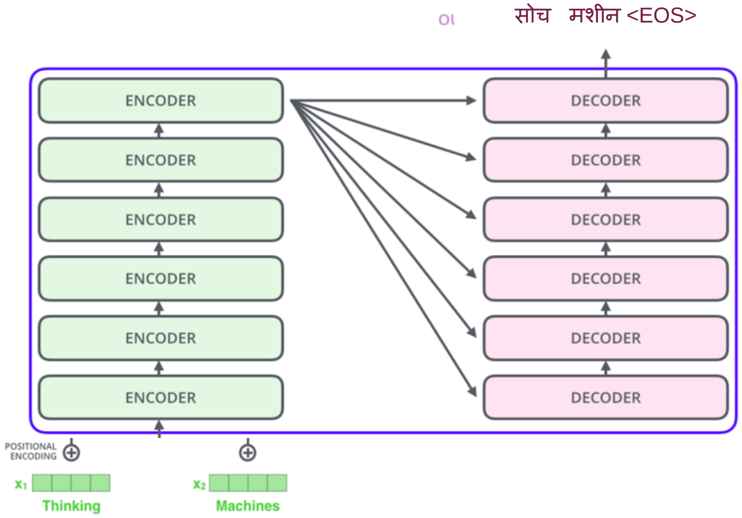


Image source: [jalammar.github.io/illustrated-transformer/]

Transformer: Encoder

1. Self attention:
   a. For each input token *X*, self attention mechanism generates an output vector *Z* of same size.
   b. Multi-head attention:
      i. Input vectors are processed for multiple sets (heads) to get an output for each set.
      ii. Outputs are combined and processed to get final encoder output *Z*.
2. Add and normalise: *LayerNorm(X+Z)*
3. Feedforward
4. Add and normalise

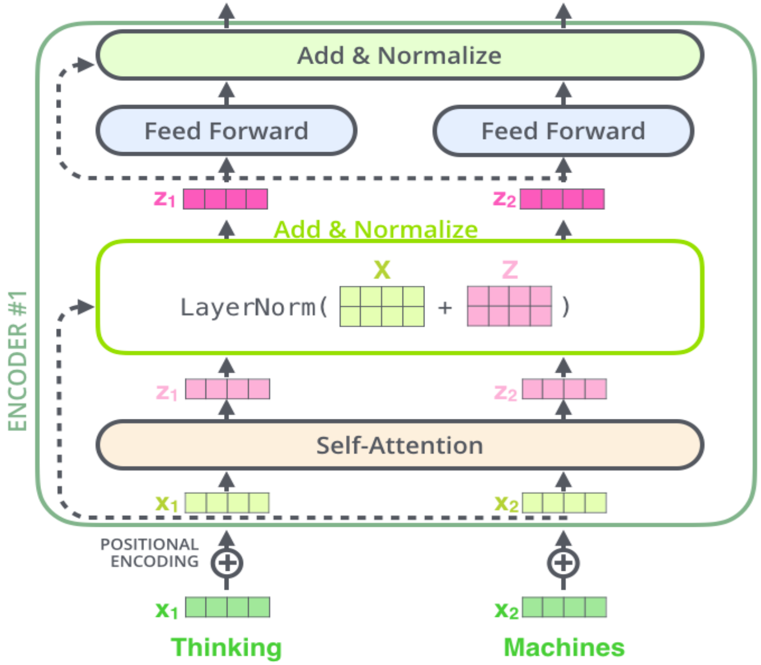
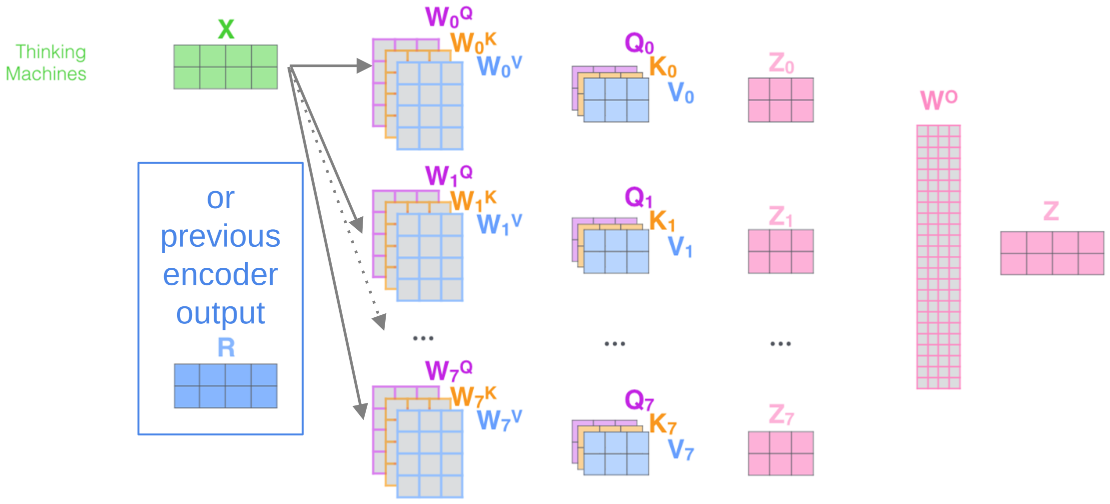


Image source: [jalammar.github.io/illustrated-transformer/]

Transformer: Multi-head attention



Q: Query; K: Key; V: Value.

1. Multiply the input *X* (or output *R* of last encoder) with trainable W_i^Q, W_i^K, W_i^V to get Q_i, K_i, V_i, for each head *i* . (8 number of heads used).
2. Prepare Z_i of *X* for *i*-th head as $\sum softmax((Q.K^T)/\sqrt{d})V^x$, where *d* is size of *Q*.
3. Z_i to *Z* conversion → concatenate then multiply with trainable W^o to transform into a vector *Z* matching size of *X*.

Image source: [jalammar.github.io/illustrated-transformer/]

Transformer: Enc-Dec Attention

Same as self-attention, except:

- It takes the *K* and *V* from the output of the encoder stack and creates its *Q* from the layer below it.

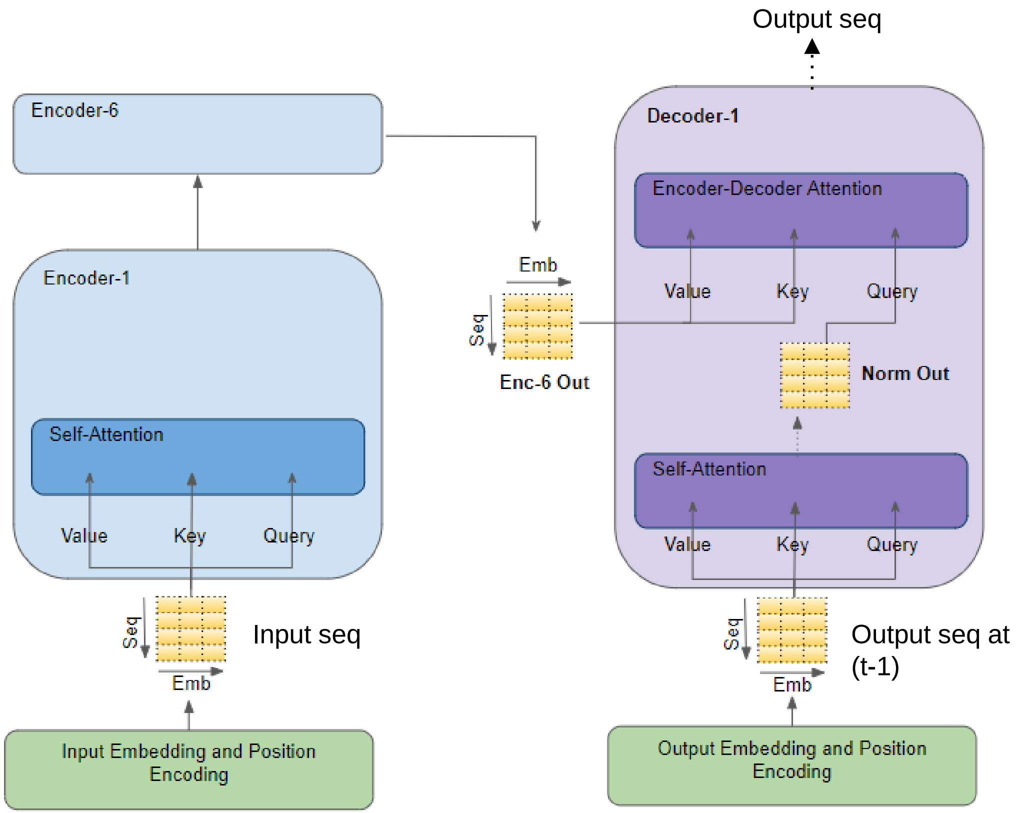


Image source: [https://towardsdatascience.com]

Transformer: Decoder

1. Self-attention: In decoder side the self-attention layer is **only allowed to attend to earlier positions in the output sequence**. This is done by masking future positions.
2. Add and normalize
3. Encoder-decoder attention
4. Add and normalize
5. Feedforward
6. Add and normalize

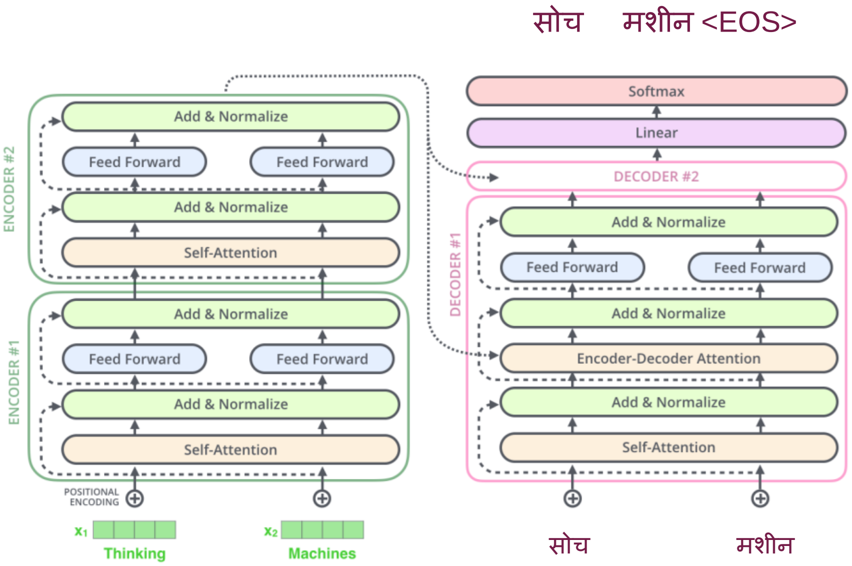


Image source: [jalammar.github.io/illustrated-transformer/]

Transformer: Entire scenario

- Training: Similar to the training of a language model
  - Source sentence ***x***: (x_1, x_2, …., x_n)
  - Target sentence ***y***: (y_1, y_2, …., y_m)

- Minimize the **cross-entropy loss**:

$$-\sum_{t=1}^{n} \log p_\theta(y_t | \boldsymbol{x}, \hat{\boldsymbol{y}}_{<t})$$

Where, $\hat{y}_t$ is a hypothesis, and θ denotes the model parameters

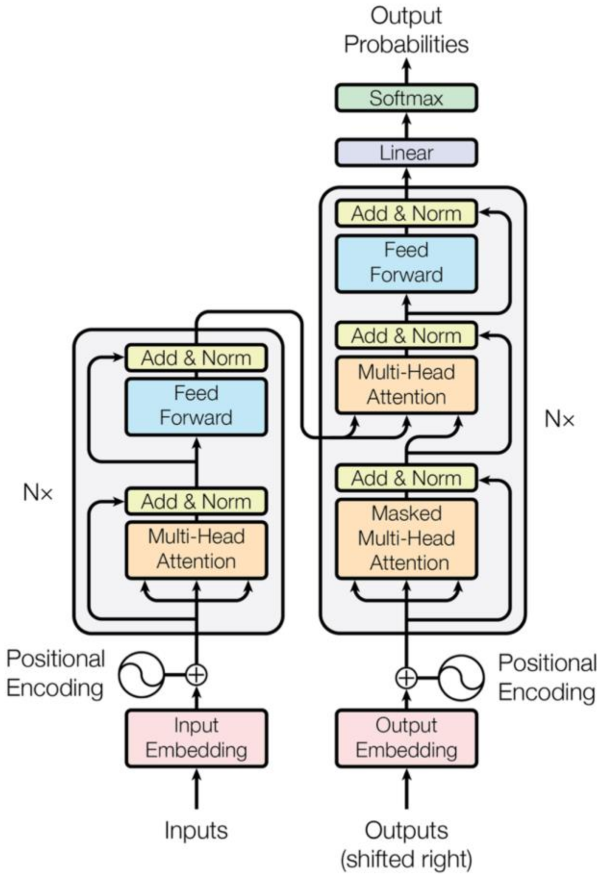


Image source: [11]

Beam Search

- In greedy search, at each time step, one best hypothesis is considered, in beam search at each step b best hypothesis is considered.

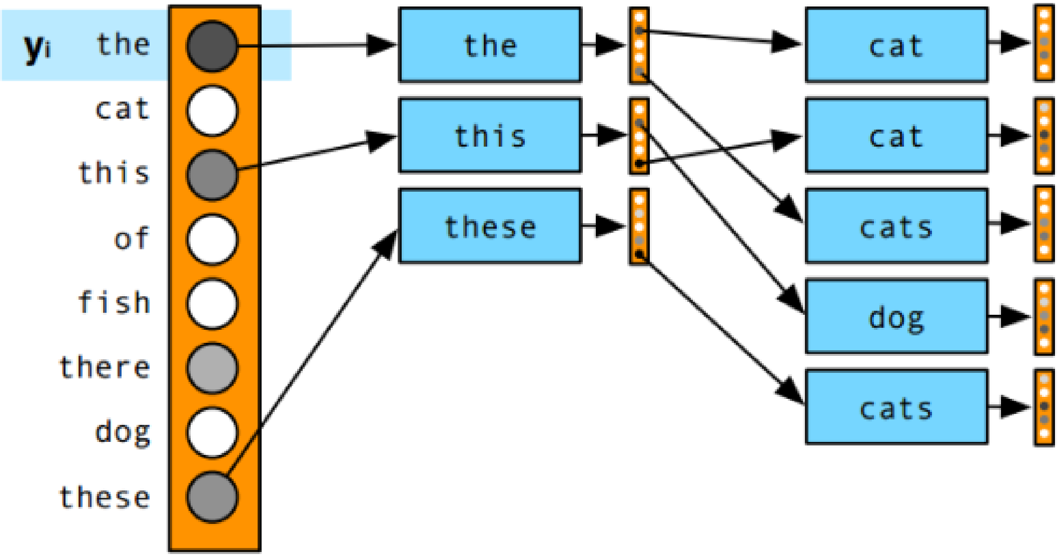


Image source: Philipp Koehn. Neural machine translation. CoRR, abs/1709.07809, 2017.

NMT Evaluation

NMT Evaluation

- How do we judge a good translation?

- Can a machine do this?

- Why should a machine do this?
  - Because human evaluation is time-consuming and expensive!
  - Not suitable for rapid iteration of feature improvements

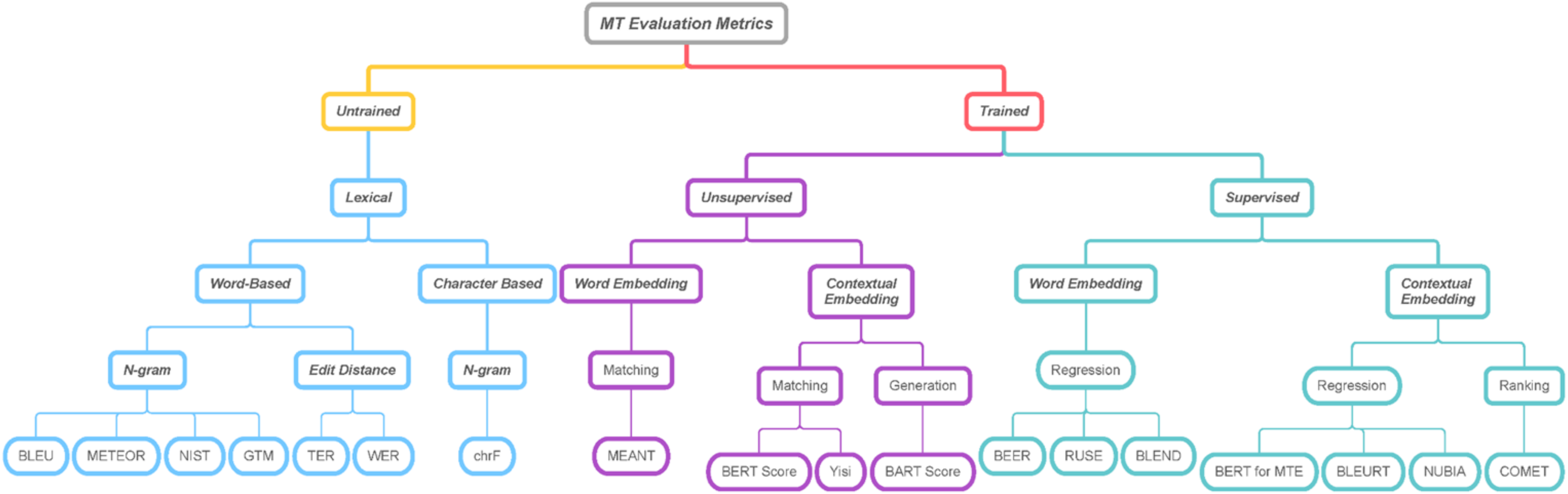
Human Evaluation

- Types: Blind/Open

- Evaluation with respect to:
  - **Adequacy:** How good the output is in terms of preserving content of the source text.
    - Source: I am attending a lecture; Target: मैं एक व्याख्यान बैठा हूँ
  - **Fluency:** How good the output is as a well-formed target language entity.
    - Source: I am attending a lecture; Target: मैं व्याख्यान हूँ

- **Direct Assessment**: Combined score in the range of 0-100.

Automatic Evaluation



Seungjun Lee & Jungseob Lee & Hyeonseok Moon & Chanjun Park & Jaehyung Seo & Sugyeong Eo & Seonmin Koo & Heuiseok Lim, 2023. "A Survey on Evaluation Metrics for Machine Translation," Mathematics, MDPI, vol. 11(4), pages 1-22, February.

Some BLEU scores for Indian Language NMT

Language pair (src\tgt)	Hi	Pa	Bn	Gu	Mr
Hi	-	60.77	28.75	52.17	31.66
Pa	64.67	-	25.32	44.74	27.78
Bn	31.79	26.96	-	24.82	16.61
Gu	55.02	46.48	25.33	-	25.62
Mr	42.97	37.08	21.82	33.29	-

Source: [Dewangan, Shubham, et al. "Experience of neural machine translation between Indian languages." Machine Translation 35.1 (2021): 71-99.]

Data Filtering

Introduction

Techniques to extract good quality parallel data from a comparable/pseudo-parallel corpus.

Motivation

- Neural Machine Translation (NMT) models are *“data hungry”.*

- The comparable corpora have increased tremendously on the World Wide Web, making it an important source for MT task.

- The mined sentence pairs are high in quantity but their quality varies a lot. This affects the quality of the MT systems.

- Hence, there is a need to come up with a preprocessing step to extract only the good quality sentence pairs from the comparable and parallel corpora before passing them to the MT model.

Filtering Techniques

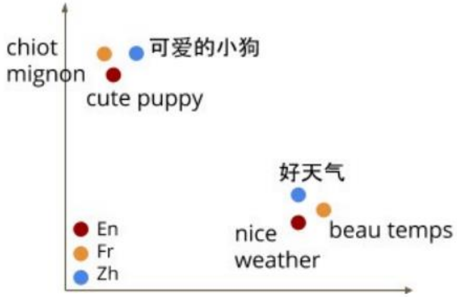
- Techniques :
  - LaBSE-based
  - Distilled PML-based

LaBSE (1/2)

- Language agnostic BERT sentence embedding model is based on a multilingual BERT model.
- Supports 109 languages including some Indic-languages.

LaBSE (2/2)

- What is Multilingual Embedding Model?
  - that maps text from multiple languages to a shared vector space.
  - Means similar words will be closer and unrelated words will be distant in the vector space as shown in fig:

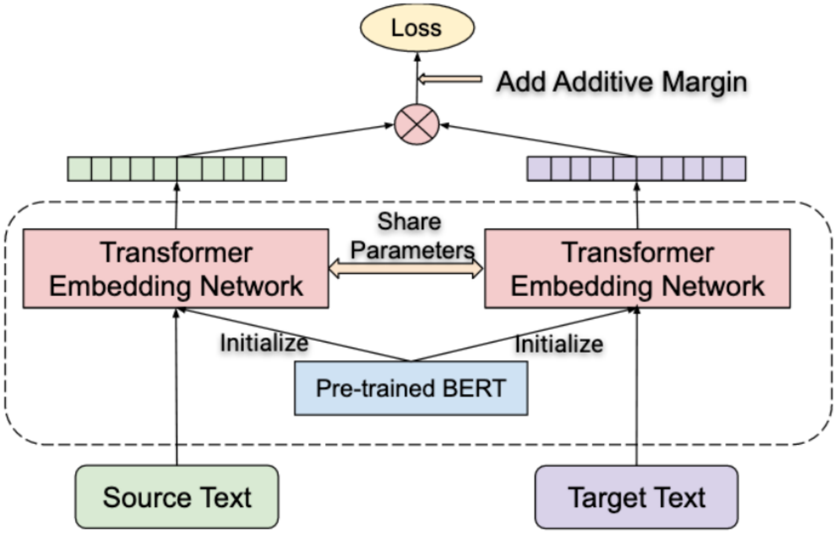


Multilingual Embedding Space via Google AI Blog

Image source- Language agnostic Bert Sentence Embedding [1]

Model Architecture

- The model architecture is based on Bi-Directional dual encoder with an additive margin loss.



Bidirectional Dual Encoder with Additive Margin Softmax and Shared Parameters via LaBSE Paper

Image source- Language agnostic Bert Sentence Embedding [1]

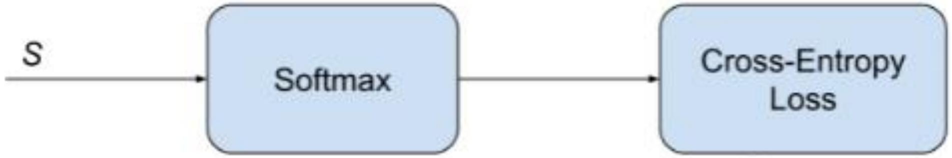
LaBSE Training Pipeline

- Firstly multilingual BERT model is trained on 109 languages for MLM (Masked Language Modelling) task.

- The obtained BERT encoders is used in parallel at source and target for fine-tuning the Translation Ranking Task.

What is Softmax Loss?

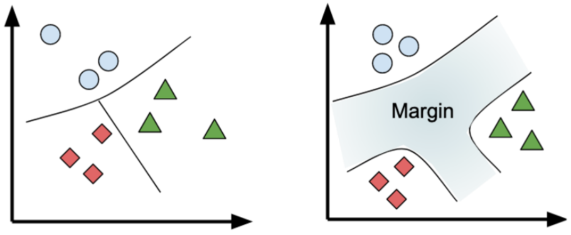
- Confusion? Softmax activation and Softmax loss are different?
- It is a softmax activation followed a Cross-Entropy loss
- It is used for multiclass classification.
- Also known as Categorical Cross-Entropy loss.



$$f(s)_i = \frac{e^{s_i}}{\sum_j^C e^{s_j}} \qquad CE = -\sum_i^C t_i log(f(s)_i)$$

Additive Margin Softmax loss

- Motivation :
  - In a classification task, we face a problem when output lies near the decision boundary in the vector space.
  - AM-Softmax aims to solve this by adding a margin to the decision boundary in order to increase the separability of the classes and also making the intra-class distance more compact.



Original Embedding Space

Embedding Space w/ AMS

Image source- LaBSE [1]

Filtering Techniques

- Techniques :
  - LaBSE
  - Distilled PML

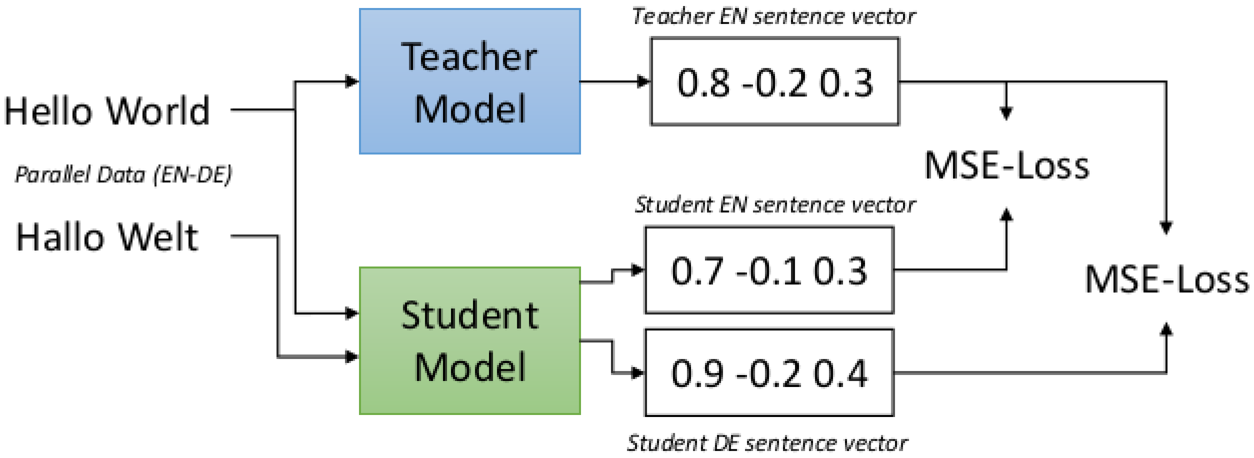
Distilled PML

- Distilled Paraphrase Multilingual Model is a Sentence BERT (SBERT) model extended to multiple languages using multilingual knowledge distillation.

- Knowledge Distillation : Compressing a model by teaching a smaller network exactly what to do at each step using an already bigger trained model.

- A Teacher-Student Model Architecture is use to train Distilled PML model.

Model Architecture



Given parallel data (e.g. English and German), train the student model such that the produced vectors for the English and German sentences are close to the teacher English sentence vector.

Image source- Making Monolingual Sentence Embedding Multilingual using Knowledge Distillation [2]

Model Training

- The English SBERT model is chosen as a teacher model.

- XLM-RoBERTa (XLM-R) model is chosen as a student model.

- So in short student model is trained using XLM-R and further fined tuned on English NLI (Natural language Inference) and STS (Semantic Text Similarity) task using English SBERT model.

Dataset Used (1/3)

- Samanantar Corpus
  - It is the biggest parallel corpus publically available for Indic languages. In our experiments we used Hindi-Marathi Samantar corpus

Dataset	# of Parallel Sentences
Samanantar Corpus	19L

Dataset Used (2/3)

- Combined Corpus:

Dataset	# of Parallel Sentences
PIB	1,08,063
PMI	29,973
Tatoeba	46,277
ILCI	4,62,777
Total Combined Corpus	6,07,832

Dataset Used (3/3)

- Test Datasets:

Corpus Name	# of Test Sentences
WAT21	2390
ILCI	2000

Approach

- LaBSE model is used to generate the sentence embeddings of the Hindi-Marathi Samanantar Corpus.

- These embeddings are used to compute the cosine similarity between the Hindi-Marathi sentence pairs.

- Based on these similarity scores we extract the good quality sentence pairs using a threshold similarity score.

- Then we use these good quality sentence pairs to train the Hindi-Marathi MT systems.

Implementation

- Experiments:
  - Baseline
  - Without LaBSE Filtering
  - LaBSE

Baseline

- We use only the combined corpus to train the Hindi-Marathi Baseline models.

- The combined corpus consists of 6L sentences. The train and tune split given below

Corpus	**#Train**	**#Tune**
Combined Corpus + Tatoeba (ILCI + PMI + PIB +Bible +Tatoeba)	6,07,832	14,390

Without LaBSE Filtering

- In this experiment we trained another Hindi-Marathi MT model using the Combined Corpus and whole Samanantar Corpus

- The train and tune split is shown below:

Corpus	# Train	# Tune
Combined Corpus + Tatoeba (ILCI + PMI + PIB +Bible +Tatoeba)	6,07,832	14,390
Samanantar	19,72,689	-
Total	25,80,677	14,390

LaBSE based Filtering

- Hindi-Marathi MT model is trained using the Combined Corpus and LaBSE filtered Samanantar Corpus.
- We use the LaBSE model provided by the huggingface to generate the LaBSE scores for the whole Samanantar Corpus.
- We also computed the LaBSE scores on the PMI corpus, which is a good quality Hindi-Marathi parallel corpus.
- We computed the average LaBSE score which turned out to be 0.89. So we chose 0.9 as the threshold LaBSE score.

LaBSE model provided by the huggingface :
https://huggingface.co/sentence-transformers/LaBSE

Samanantar LaBSE Data Analysis

LaBSE score Range	No. of Parallel Sentences
>=0.9	3,54,315
>=0.91	2,89,802
>=0.92	2,32,187
>=0.93	1,80,776
>=0.94	1,36,200
>=0.95	97,860
>=0.96	65,167
>=0.97	38,699
>=0.98	17,796
>=0.99	4,103

LaBSE based Filtering

- We extracted 3.5L sentences from Samanantar Corpus that had a LaBSE score of 0.9 and above.
- The train, tune split for this model is given below

Corpus	#Train	#Tune + Test
Combined Corpus + Tatoeba (ILCI + PMI + PIB +Bible +Tatoeba)	6,07,832	14,390
Samanantar_labse (labse>=0.9)	3,54,314	-
Total	9,62,146	14,390

Implementation (1/2)

- Training
  - We have used transformer architecture for all our models.
  - We trained the NMT model with the help of OpenNMT-py library

OpenNMT-py is an open source Toolkit
https://github.com/OpenNMT/OpenNMT-py

Implementation (2/2)

- Training
  - The parameters for the transformer model are shown below

Encoder Type	Transformer
Decoder Type	Transformer
Number of layers in encoder/decoder	6
Number of attention heads	8
Size of encoder embedding dimensions	512
Dropout	0.1

Results (1/5)

- Hindi-Marathi MT model

Models	BLEU Score	
	WAT21	ILCI
Baseline	13.8	33.2
Without LaBSE filtering	16.9	33.0
LaBSE filtering	17.8	33.2

We used sacrebleu python library to calculate the BLEU scores.
https://github.com/mjpost/sacrebleu

Results (2/5)

- Hindi-Marathi MT Model
  - We see an increment of 4 BLEU score points in LaBSE filtered model as compared to Baseline on WAT21 test data.
  - Increment of 1 BLEU score points as compared to the “without LaBSE filtered model” on WAT21 test data.
  - We also see that the BLEU score on ILCI dataset remains the same for Baseline and LaBSE filtered model, while it decreases by 0.2 points for “without LaBSE filtered model”.

Results (3/5)

- Marathi-Hindi MT model

Models	BLEU Score	
	WAT21	ILCI
Baseline	22.1	37.4
Without LaBSE filtering	21.6	33.6
LaBSE filtering	25.1	37.9

Results (4/5)

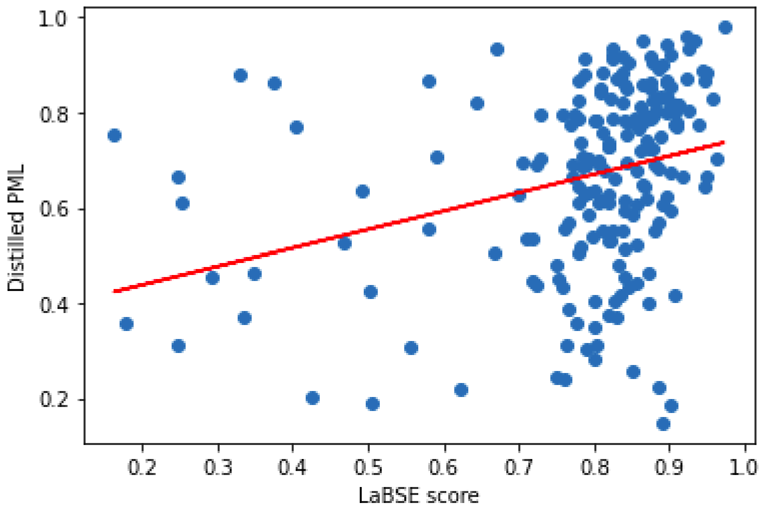
- Marathi-Hindi MT Model
  - Increment of 3 BLEU score points in LaBSE filtered model as compared to Baseline on WAT21 test data.
  - Increment of 4 BLEU score points as compared to “without LaBSE filtered model” on WAT21 test data.
  - We also see that the BLEU score on ILCI dataset, increments by 0.5 for LaBSE filtered model as compared to Baseline, while it decreases by 4 points for “without LaBSE filtered model”.
  - This is because the Samanantar corpus doesn’t consist of the in-domain data of ILCI dataset.

Results (5/5)

- We also computed the Spearman’s rank correlation coefficient between LaBSE and Distilled PML scores.

- These scores were computed on a set of 5000 Hindi-Marathi parallel sentences.

- The correlation coefficient turned out to be **0.38**



Scatter plot of LaBSE and Distilled PML scores

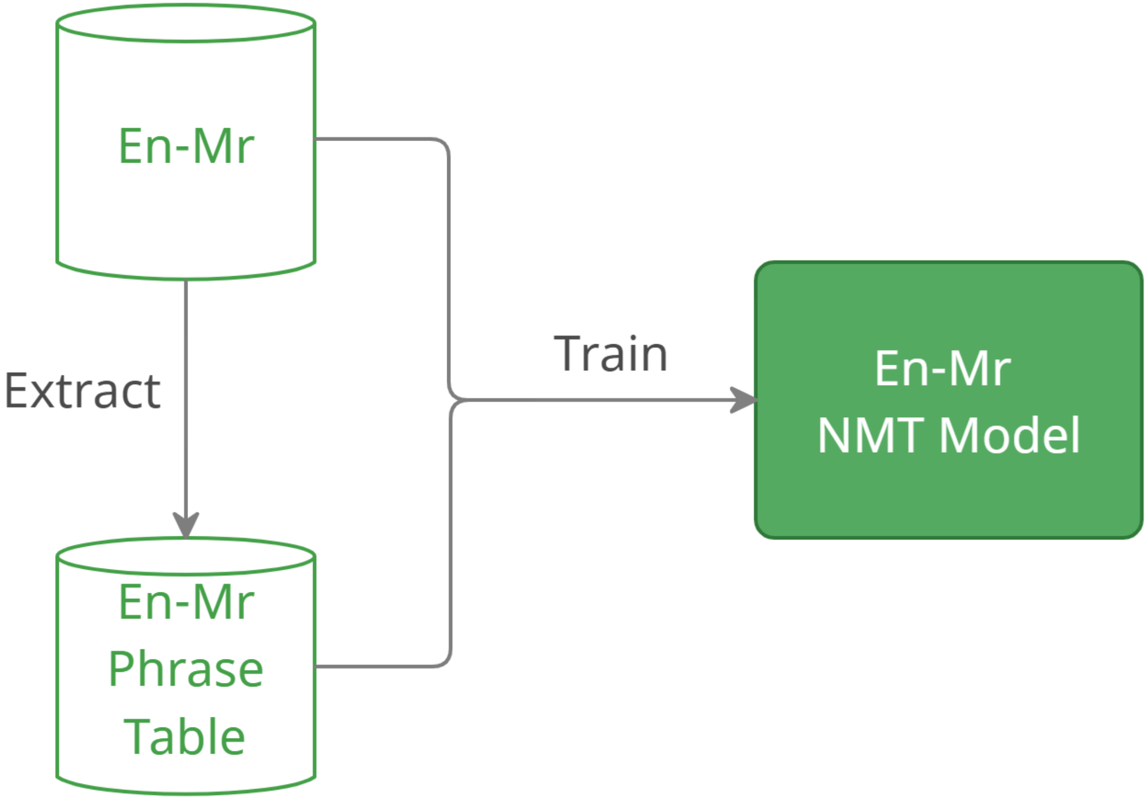
Data Augmentation

Phrase Table Injection (1/3)

- In this technique, phrase table is extracted from the Source-Target parallel corpus.

- Finally the Source-Target NMT model is trained using the Source-Target Parallel Corpus and Source-Target Phrases.

Phrase Table Injection (2/3)



Phrase Table Injection (3/3)

- Dataset

	Train	Test (WAT 2021)
Number of Sentences	250,347	2390

- Results

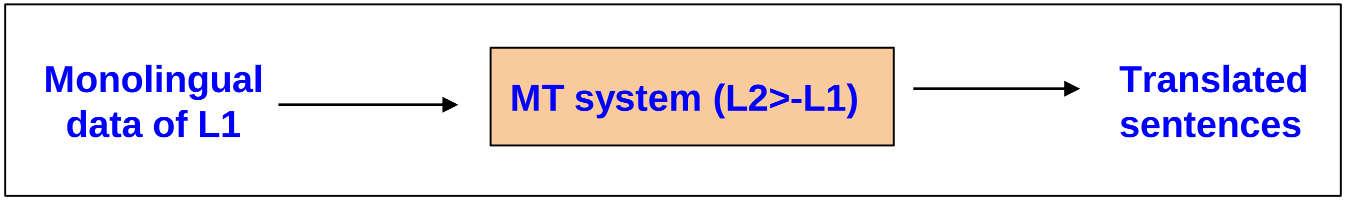
	BLEU Score
Baseline	16.26
Phrase Table Injection	17.15

[12]

Advanced NMT Approaches

Back-Translation

- Utilize monolingual data of target language
- Generate pseudo parallel data using MT system in opposite direction (target->source)



Monolingual data of L1

MT system (L2>-L1)

Translated sentences

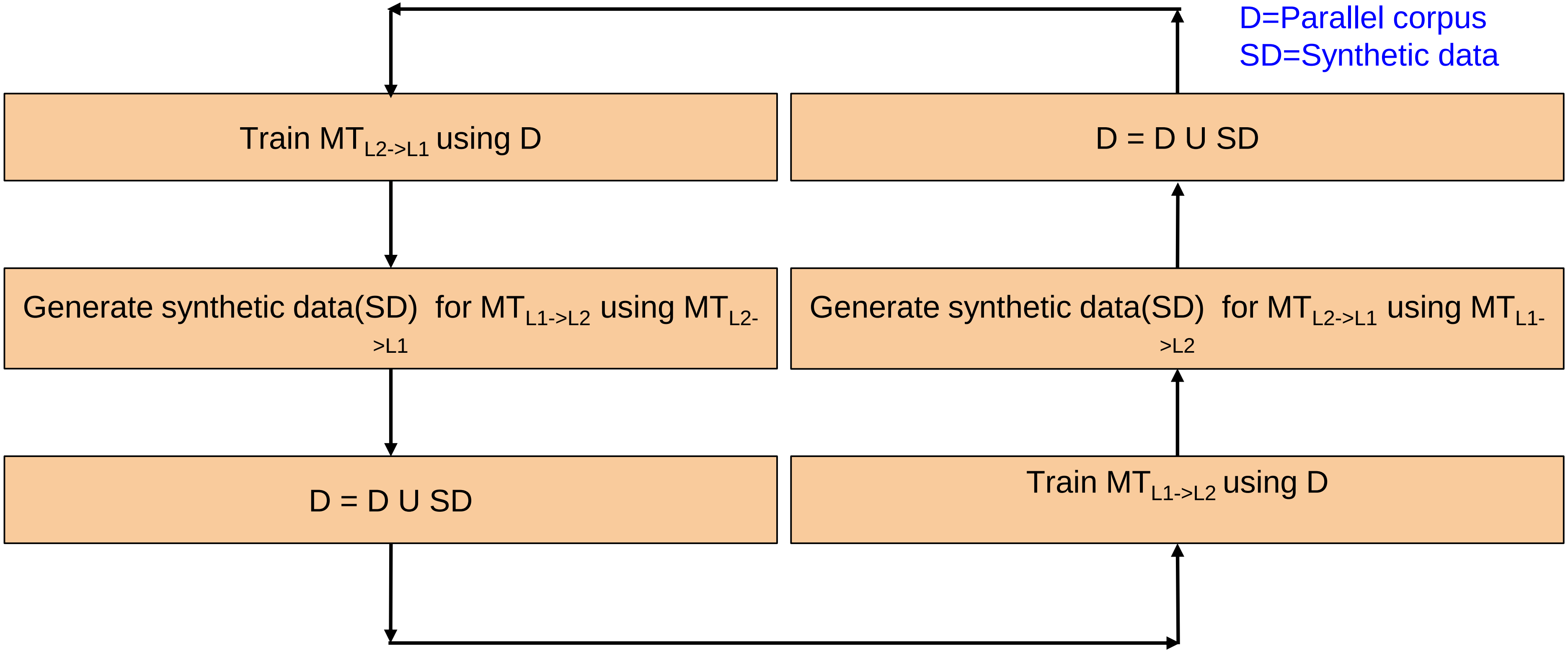
Generation of pseudo parallel sentences

- Train MT system (L1->L2) using a combination of parallel and generated synthetic data both

Sennrich, Rico, Barry Haddow, and Alexandra Birch. "Improving Neural Machine Translation Models with Monolingual Data." In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 86-96. 2016.

Iterative Back-Translation (1/2)

D=Parallel corpus
SD=Synthetic data



Train $MT_{L2\text{->}L1}$ using D

Generate synthetic data(SD) for $MT_{L1\text{->}L2}$ using $MT_{L2\text{->}L1}$

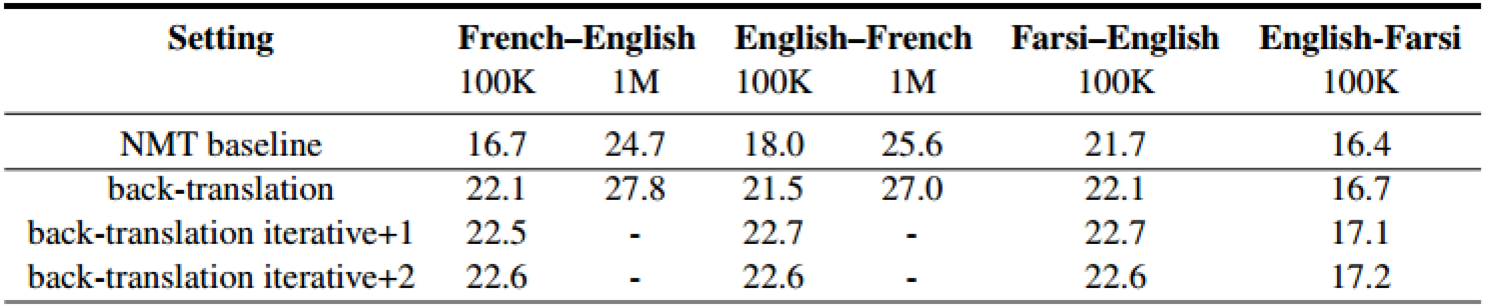
D = D U SD

Train $MT_{L1\text{->}L2}$ using D

Generate synthetic data(SD) for $MT_{L2\text{->}L1}$ using $MT_{L1\text{->}L2}$

D = D U SD

Iterative Back-Translation (2/2)



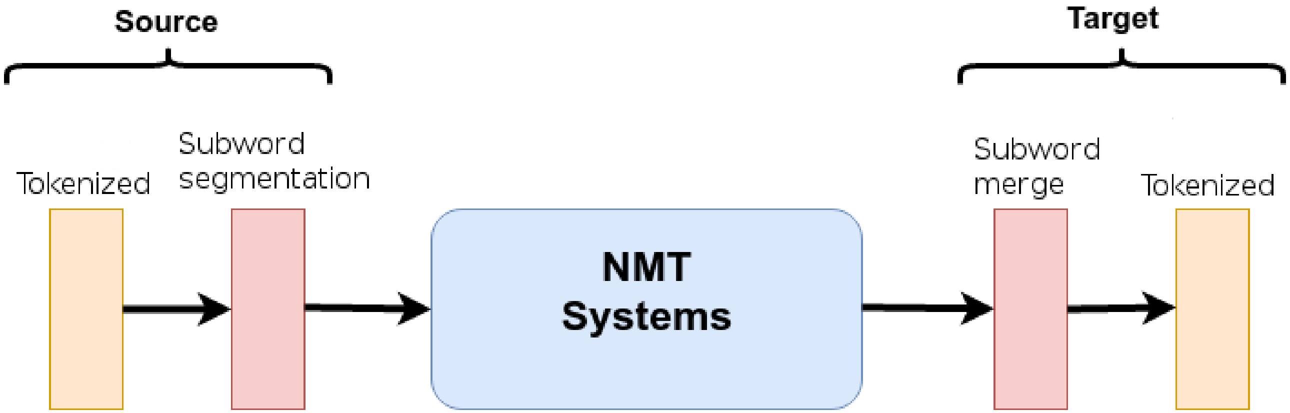
Setting	French–English		English–French		Farsi–English	English-Farsi
	100K	1M	100K	1M	100K	100K
NMT baseline	16.7	24.7	18.0	25.6	21.7	16.4
back-translation	22.1	27.8	21.5	27.0	22.1	16.7
back-translation iterative+1	22.5	-	22.7	-	22.7	17.1
back-translation iterative+2	22.6	-	22.6	-	22.6	17.2

- Beneficial for Low resource languages too

Image source: Hoang, Vu Cong Duy, Philipp Koehn, Gholamreza Haffari, and Trevor Cohn. "Iterative back-translation for neural machine translation." In *Proceedings of the 2nd Workshop on Neural Machine Translation and Generation*, pp. 18-24. 2018.

Subword NMT

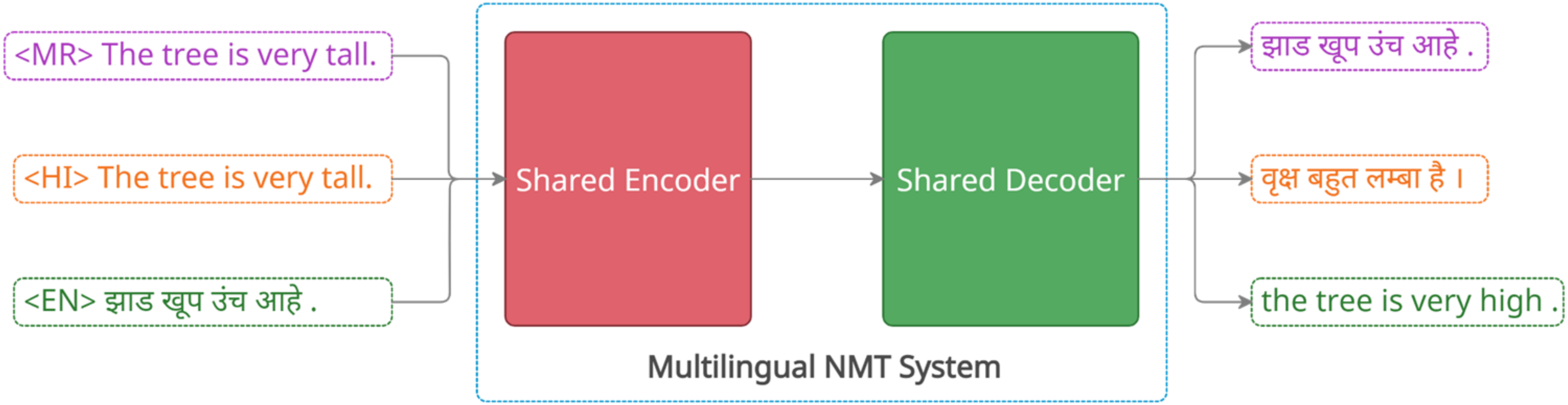
- Compound words, words with morphological variation (need for morphological segmentation), named entities are very common
- We can utilise this phenomena, if we look into subword level.



Multilingual NMT

- Motivation
  - Translation between ***N*** languages to ***N*** languages will require **O(N^2)** models.
  - A single **N-to-N** multilingual model can translate between all **O(N^2)** language directions.
  - Multilingual Models share knowledge between all languages improving performance for low resource language pairs.

Multilingual NMT



- Parameter sharing: Shared encoder and decoder
- Need to find the right amount of shared parameters

Johnson, Melvin, Mike Schuster, Quoc V. Le, Maxim Krikun, Yonghui Wu, Zhifeng Chen, Nikhil Thorat et al. "Google’s multilingual neural machine translation system: Enabling zero-shot translation." *Transactions of the Association for Computational Linguistics* 5 (2017): 339-351.

Google’s MNMT System (Johnson et al., 2017)

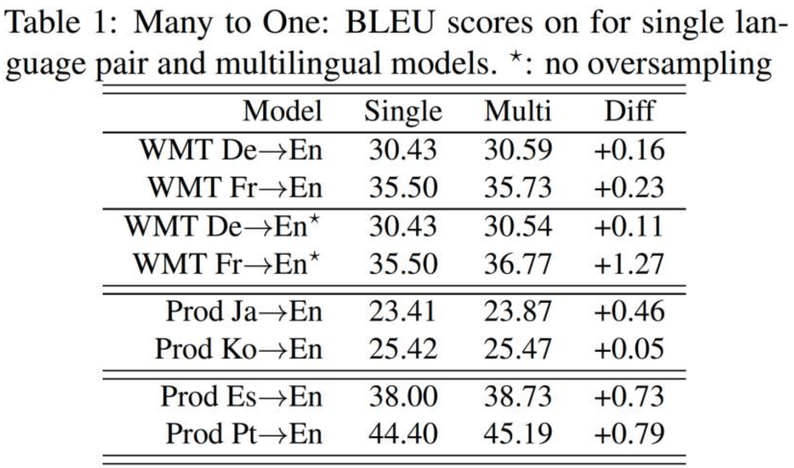


Table 1: Many to One: BLEU scores on for single language pair and multilingual models. ⋆: no oversampling

Model	Single	Multi	Diff
WMT De→En	30.43	30.59	+0.16
WMT Fr→En	35.50	35.73	+0.23
WMT De→En⋆	30.43	30.54	+0.11
WMT Fr→En⋆	35.50	36.77	+1.27
Prod Ja→En	23.41	23.87	+0.46
Prod Ko→En	25.42	25.47	+0.05
Prod Es→En	38.00	38.73	+0.73
Prod Pt→En	44.40	45.19	+0.79

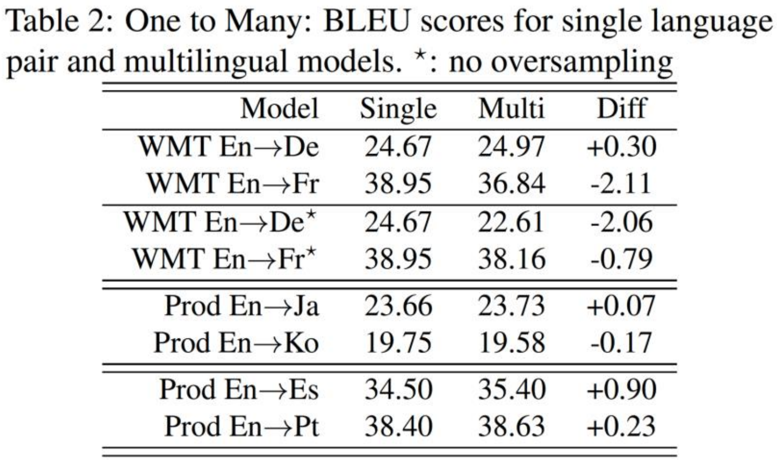


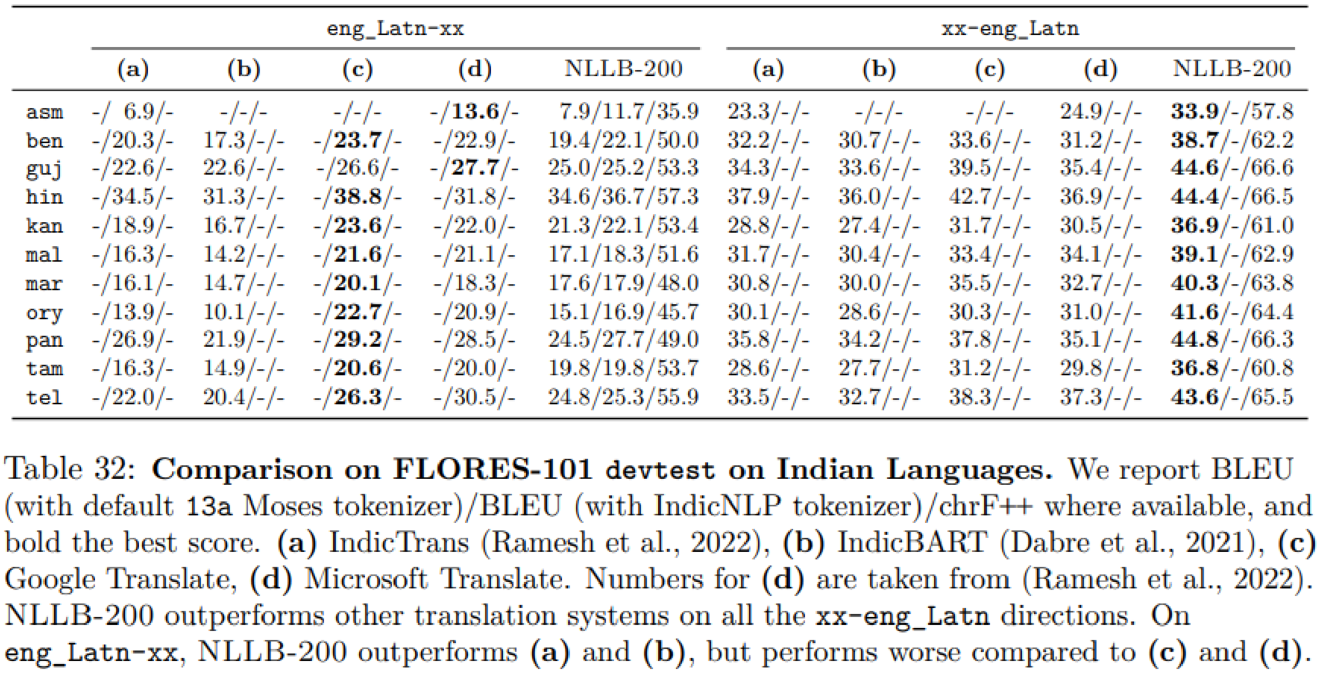
Table 2: One to Many: BLEU scores for single language pair and multilingual models. ⋆: no oversampling

Model	Single	Multi	Diff
WMT En→De	24.67	24.97	+0.30
WMT En→Fr	38.95	36.84	-2.11
WMT En→De⋆	24.67	22.61	-2.06
WMT En→Fr⋆	38.95	38.16	-0.79
Prod En→Ja	23.66	23.73	+0.07
Prod En→Ko	19.75	19.58	-0.17
Prod En→Es	34.50	35.40	+0.90
Prod En→Pt	38.40	38.63	+0.23

- A single multilingual model

No Language Left Behind (NLLB)

- Trained Multilingual model on 200 languages.



	eng_Latn-xx					xx-eng_Latn				
	(a)	(b)	(c)	(d)	NLLB-200	(a)	(b)	(c)	(d)	NLLB-200
asm	-/ 6.9/-	-/-/-	-/-/-	-/**13.6**/-	7.9/11.7/35.9	23.3/-/-	-/-/-	-/-/-	24.9/-/-	**33.9**/-/57.8
ben	-/20.3/-	17.3/-/-	-/**23.7**/-	-/22.9/-	19.4/22.1/50.0	32.2/-/-	30.7/-/-	33.6/-/-	31.2/-/-	**38.7**/-/62.2
guj	-/22.6/-	22.6/-/-	-/26.6/-	-/**27.7**/-	25.0/25.2/53.3	34.3/-/-	33.6/-/-	39.5/-/-	35.4/-/-	**44.6**/-/66.6
hin	-/34.5/-	31.3/-/-	-/**38.8**/-	-/31.8/-	34.6/36.7/57.3	37.9/-/-	36.0/-/-	42.7/-/-	36.9/-/-	**44.4**/-/66.5
kan	-/18.9/-	16.7/-/-	-/**23.6**/-	-/22.0/-	21.3/22.1/53.4	28.8/-/-	27.4/-/-	31.7/-/-	30.5/-/-	**36.9**/-/61.0
mal	-/16.3/-	14.2/-/-	-/**21.6**/-	-/21.1/-	17.1/18.3/51.6	31.7/-/-	30.4/-/-	33.4/-/-	34.1/-/-	**39.1**/-/62.9
mar	-/16.1/-	14.7/-/-	-/**20.1**/-	-/18.3/-	17.6/17.9/48.0	30.8/-/-	30.0/-/-	35.5/-/-	32.7/-/-	**40.3**/-/63.8
ory	-/13.9/-	10.1/-/-	-/**22.7**/-	-/20.9/-	15.1/16.9/45.7	30.1/-/-	28.6/-/-	30.3/-/-	31.0/-/-	**41.6**/-/64.4
pan	-/26.9/-	21.9/-/-	-/**29.2**/-	-/28.5/-	24.5/27.7/49.0	35.8/-/-	34.2/-/-	37.8/-/-	35.1/-/-	**44.8**/-/66.3
tam	-/16.3/-	14.9/-/-	-/**20.6**/-	-/20.0/-	19.8/19.8/53.7	28.6/-/-	27.7/-/-	31.2/-/-	29.8/-/-	**36.8**/-/60.8
tel	-/22.0/-	20.4/-/-	-/**26.3**/-	-/30.5/-	24.8/25.3/55.9	33.5/-/-	32.7/-/-	38.3/-/-	37.3/-/-	**43.6**/-/65.5

Table 32: **Comparison on FLORES-101 devtest on Indian Languages.** We report BLEU (with default 13a Moses tokenizer)/BLEU (with IndicNLP tokenizer)/chrF++ where available, and bold the best score. **(a)** IndicTrans (Ramesh et al., 2022), **(b)** IndicBART (Dabre et al., 2021), **(c)** Google Translate, **(d)** Microsoft Translate. Numbers for **(d)** are taken from (Ramesh et al., 2022). NLLB-200 outperforms other translation systems on all the xx-eng_Latn directions. On eng_Latn-xx, NLLB-200 outperforms **(a)** and **(b)**, but performs worse compared to **(c)** and **(d)**.

NLLB Team.2022. No Language Left Behind: Scaling Human-Centered Machine Translation. arXiv:2207.04672

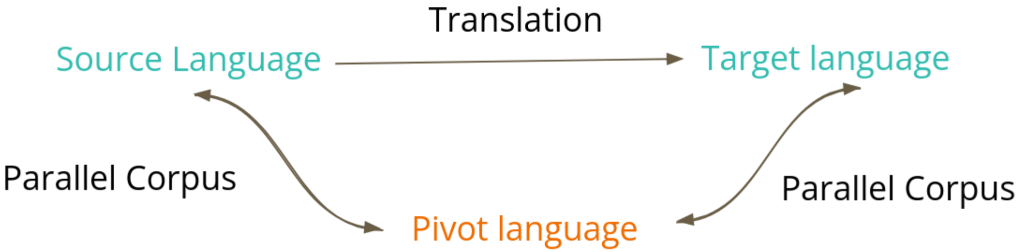
Pivot-based Neural Machine Translation

Pranav Gaikwad
Under the guidance of Prof. Pushpak Bhattacharyya & Prof. Preethi Jyothi
Computer Science and Engineering Department
IIT Bombay

11th March 2024

Introduction to pivot-based Machine Translation

- Pivot-based Machine Translation is a technique applied in low-resource scenarios that leverages a third language having a parallel corpus with both the source and target languages.
- This intermediary language, known as the **pivot language**, facilitates improved translations from the source to the target language.
- The quality of translation depends heavily on the pivot language. Ideally, the pivot language should be a high-resource language, linguistically similar to the source or target language.



A basic framework for pivot-based MT

Basic idea of pivot based translation

Parallel corpus for English-Marathi:

Source Language (English)	Target Language (Marathi)
I like English language.	मला इंग्रजी भाषा आवडते.
I like Hindi language.	मला हिंदी भाषा आवडते.
....	....

Parallel corpus for English-Hindi and Hindi-Marathi:

Source Language (English)	Pivot Language (Hindi)
I like English language.	मुझे अंग्रेजी भाषा पसंद है।
I like Hindi language.	मुझे हिंदी भाषा पसंद है।
....	....

Pivot Language (Hindi)	Target Language (Marathi)
मुझे अंग्रेजी भाषा पसंद है।	मला इंग्रजी भाषा आवडते.
मुझे हिंदी भाषा पसंद है।	मला हिंदी भाषा आवडते.
....	....

Intuition for using pivoting

- Training the model to grasp linguistic nuances in a low-resource language is difficult.
- The abundance of data enables the model to effectively learn linguistic phenomena in high-resource pivot languages.
- This can lead to better translation performance for low-resource language by exploiting linguistic similarity.

Piping hot tea -> खुप गरम चहा (More difficult to learn)
Piping hot tea -> गर्मागरम चाय (Easier to learn due to abundance of data)
गर्मागरम चाय -> गरमगरम चहा (Easier to learn due to linguistic similarity)

An example of how pivoting can help low-resource translation

Approaches for pivot-based translation

- **Cascading approaches**
- Phrase translation approach (Utiyama and Isahara, 2007)
- Sentence translation approach (Utiyama and Isahara, 2007)
- Multiple-pivot approach (Dabre et al., 2015)
- Cascading NMT models (Kim et al., 2016)

- **Transfer learning-based approaches**
- Transfer learning (Zoph et al., 2016)
- Transfer learning with pivoting (Kim et al., 2019)
- Step-wise pre-training (Kim et al., 2019)
- Cross-lingual encoder approach (Kim et al., 2019)
- ConsisTL: a modified transfer learning approach (Kim et al., 2019)

Pivot-based approaches with cascading

Phrase translation approach:

- A phrase table from source to target is created using phrase tables of source-pivot and pivot-target languages.
- For this, translation probabilities and lexical probabilities are generated using source-to-pivot and pivot-to-target corpora.

Sentence translation approach:

- A source language sentence was translated to ‘n’ pivot language sentences and these pivot language sentences were separately translated to target language sentences.
- The highest-scoring target sentence was chosen as the output translation. The scoring is done using various features of the output sentence.

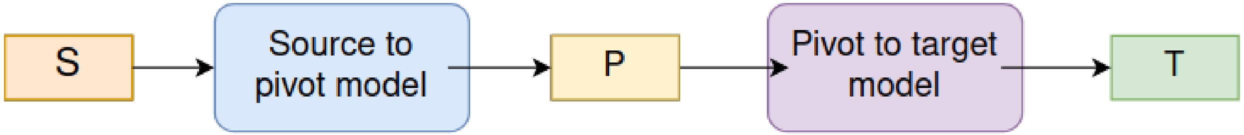
Pivot-based approaches with cascading

Multiple-pivot approach:

- A multilingual corpus was leveraged to generate translations using multiple pivots.
- The translation was generated by using a weighted average of translation probabilities with all pivots.

Cascading NMT models (Kim et al., 2016)

- A source-to-pivot model and a pivot-to-target model are trained using source-pivot and pivot-target parallel data.
- The output produced by the source-to-pivot model is fed into the pivot-to-target model to produce the translated target sentence.



The figure shows the framework of cascading NMT models. ‘S’ is the source sentence, ‘P’ is the pivot sentence and ‘T’ is the sentence in the target language.

Transfer learning based pivoting approaches

- Transfer learning is a method that focuses on storing knowledge gained while solving one problem and applying it to a different but related problem.
- The transfer learning based pivoting approaches are as follows:
  - Transfer learning (Zoph et al., 2016)
  - Transfer learning with pivoting (Kim et al., 2019)
  - Step-wise pre-training (Kim et al., 2019)
  - Pivot adapter approach (Kim et al., 2019)
  - Cross-lingual encoder approach (Kim et al., 2019)
  - ConsisTL: a modified transfer learning approach (Li et al., 2022)

1. Transfer learning (Zoph et al., 2016)

- A source-to-pivot model is pre-trained and used to initialize the source-to-target model if the pivot is closer to the low-resource target language (Fig. a).
- A pivot-to-target model is pre-trained and used to initialize the source-to-target model if the pivot is closer to the low-resource source language (Fig. b).

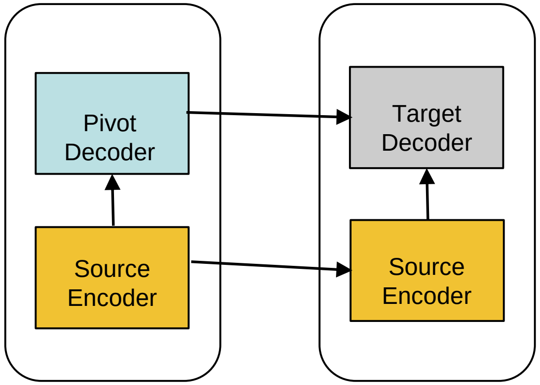


Fig. a

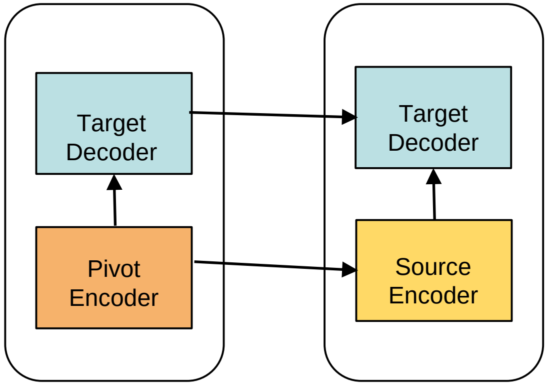
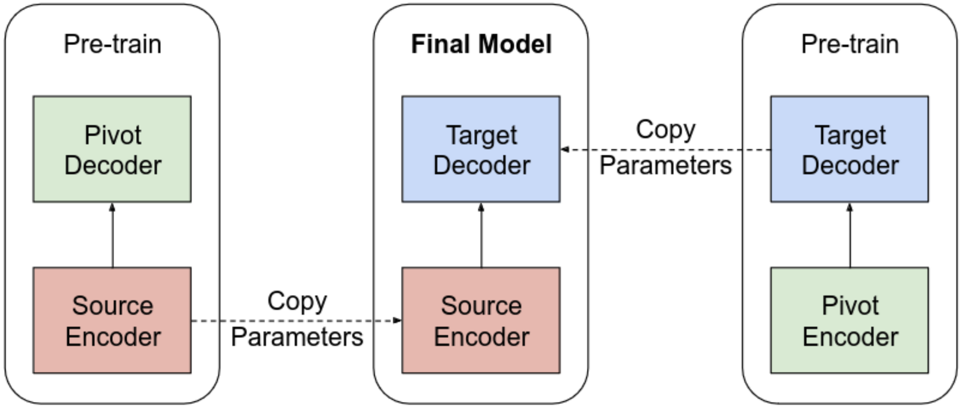


Fig. b

2. Transfer learning with pivoting (Kim et al., 2019)

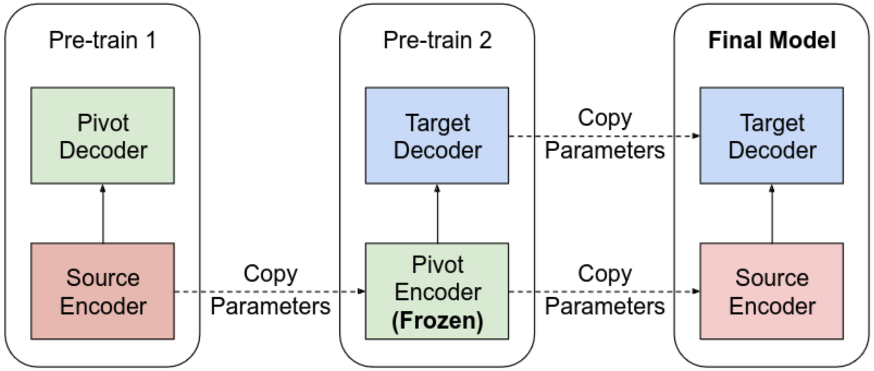
- A source-to-pivot and pivot-to-target model is pre-trained.
- The encoder of the source-to-pivot model and decoder of the pivot-to-target model is used to initialize the source-to-target model.



Transfer learning with pivoting

3. Step-wise pretraining approach (Kim et al., 2019)

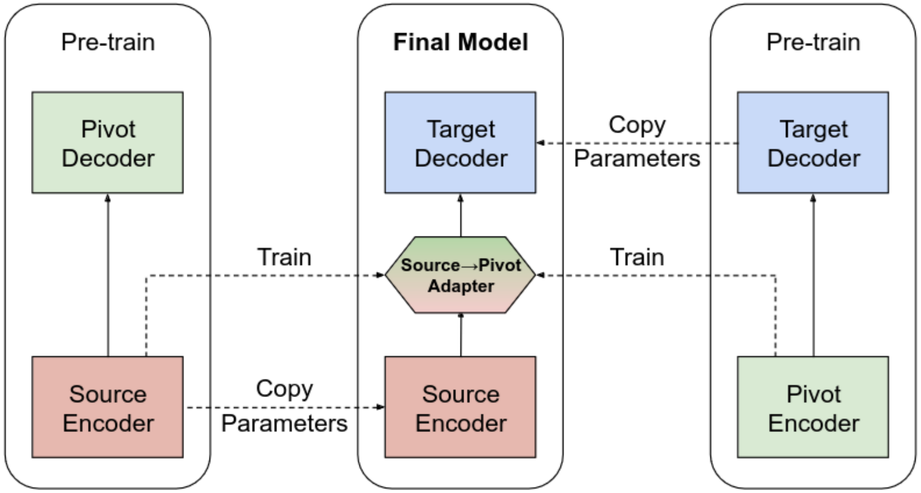
- A source-to-pivot model is pre-trained and used to initialize the pivot-to-target model.
- The encoder parameters are frozen while training the pivot-to-target model.
- This pre-trained model is fine-tuned on source-target parallel data to develop a source-to-target model.



Step-wise pretraining approach

4. Pivot adapter approach

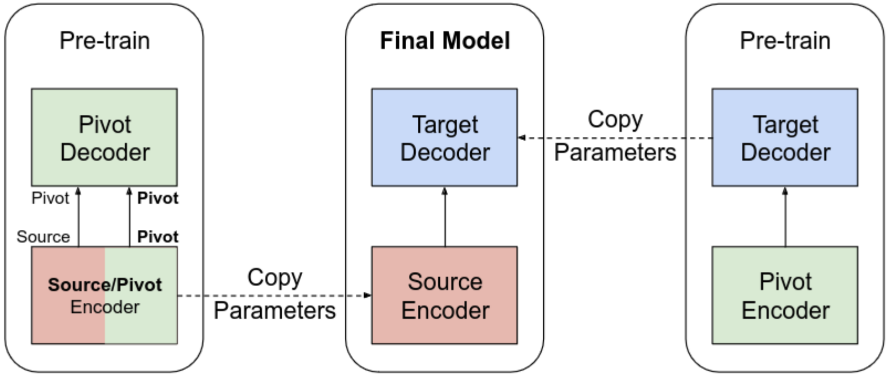
- Involves pre-training two models on source-pivot and pivot-target data.
- A source-pivot adapter is used to familiarize source encoder of source-pivot with target decoder of pivot-target.



Pivot adapter approach

5. Cross-lingual encoder approach

- Involves pre-training the encoder-decoder for source-pivot and pivot-target data while using multilingual encoder on source-pivot side.



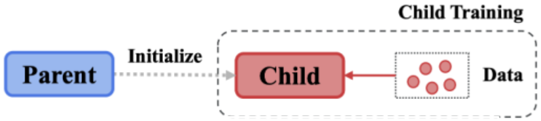
Cross-lingual encoder approach

6. ConsisTL: a modified approach for transfer learning (1/2)

(Li et al., 2022)

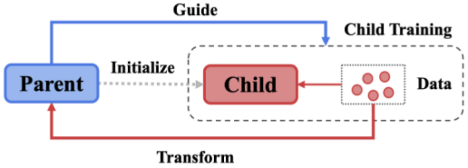
- A pivot-to-target model pre-trained using pivot-target parallel data.
- This model is used to initialize the source-to-target model. The pivot-to-target model is also used while fine-tuning the source-to-target model.

- Traditional transfer learning:



Transfer learning

- The proposed method of transfer learning:

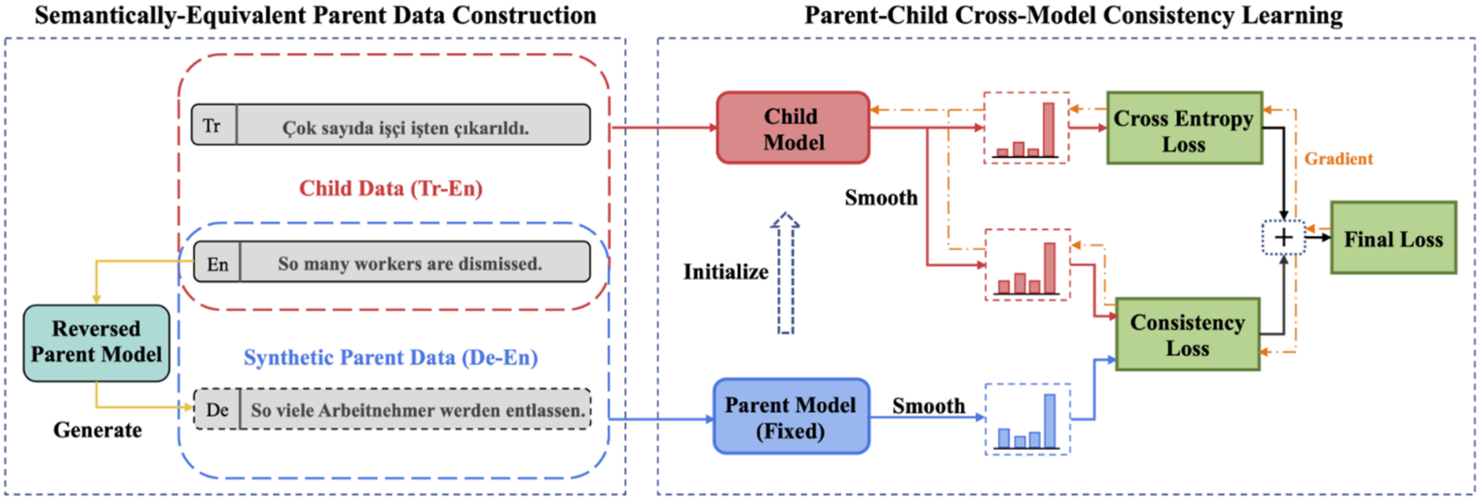


ConsisTL

4. ConsisTL: a modified approach for transfer learning (2/2)

(Li et al., 2022)

The architecture for the ConsisTL approach is as follows:



Semantically-Equivalent Parent Data Construction
Parent-Child Cross-Model Consistency Learning
Tr
Çok sayıda işçi işten çıkarıldı.
Child Data (Tr-En)
En
So many workers are dismissed.
Reversed Parent Model
Synthetic Parent Data (De-En)
De
So viele Arbeitnehmer werden entlassen.
Generate
Child Model
Initialize
Smooth
Parent Model (Fixed)
Smooth
Cross Entropy Loss
Gradient
+
Final Loss
Consistency Loss

Tr-Turkish is low resource source language
De- German is pivot language
En-English is target language

Limitations of pivot-based approaches

- In the cascade-based pivoting techniques, the source sentence remains underutilized as it passes through a source-to-pivot model to produce a pivot sentence, which is then translated into the target language using a separate pivot-to-target model.

- The cascade-based pivoting techniques are also prone to error propagation.

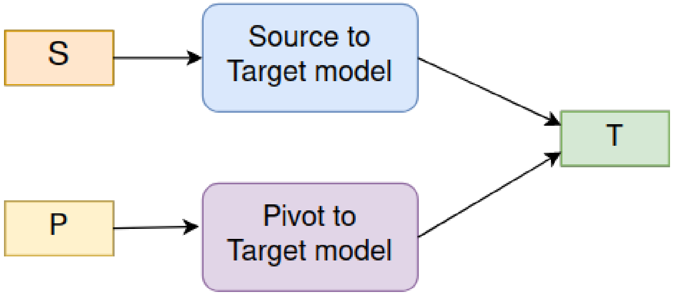
- The transfer learning based techniques only partially leverage the pivot, as it's only used for pre-training, with subsequent fine-tuning on source-target data.

- The transfer learning based techniques struggle with the issue of catastrophic forgetting.

- We can say that the previous pivoting techniques fail to completely utilize the knowledge embedded in source and pivot.

Multi-source pivoting with ensembling

- In this approach, we leverage the source-pivot corpus to train a robust source-to-pivot translation model.
- We utilize this model to generate the pivot sentence at inference time.
- We train the source-to-target model and pivot-to-target model and ensemble them.
- The following figure shows how this approach works:



Multi-source pivoting with ensembling

S: Source sentence
P: Pivot sentence
T: Target sentence

Thank you!

References

[1] Ondřej Bojar, Rajen Chatterjee, Christian Federmann, Yvette Graham, Barry Haddow, Matthias Huck, Antonio Jimeno Yepes et al. "Findings of the 2016 conference on machine translation." In Proceedings of the First Conference on Machine Translation: Volume 2, Shared Task Papers, pp. 131-198. 2016.

[2] Ondřej Bojar, Rajen Chatterjee, Christian Federmann, Yvette Graham, Barry Haddow, Shujian Huang, Matthias Huck et al. "Findings of the 2017 conference on machine translation (wmt17)." In Proceedings of the Second Conference on Machine Translation, pp. 169-214. 2017.

[3] Rajen Chatterjee, Matteo Negri, Raphael Rubino, Marco Turchi. “Findings of the WMT 2018 Shared Task on Automatic Post-Editing.” In Proceedings of the Third Conference on Machine Translation: Shared Task Papers, pp. 710-725. 2018

[4] Rajen Chatterjee, Christian Federmann, Matteo Negri, and Marco Turchi. "Findings of the WMT 2019 shared task on automatic post-editing." In Proceedings of the Fourth Conference on Machine Translation (Volume 3: Shared Task Papers, Day 2), pp. 11-28. 2019.

[5] Rajen Chatterjee, Markus Freitag, Matteo Negri, Marco Turchi. Findings of the WMT 2020 Shared Task on Automatic Post-Editing. Proceedings of the Fifth Conference on Machine Translation, EMNLP, Nov. 2020, 646-659, 2020.

References

[6] Junczys Dowmunt, Marcin, and Roman Grundkiewicz. "Log-linear combinations of monolingual and bilingual neural machine translation models for automatic post-editing." In Proceedings of the First Conference on Machine Translation: Volume 2, Shared Task Papers, ACL. 2016.

[7] Matteo Negri, Marco Turchi, Rajen Chatterjee, Nicola Bertoldi. “ESCAPE: a Large-scale Synthetic Corpus for Automatic Post-Editing.” In Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC). 2018.

[8] Yang, Hao, Minghan Wang, Daimeng Wei, Hengchao Shang, Jiaxin Guo, Zongyao Li, Lizhi Lei, Ying Qin, Shimin Tao, Shiliang Sun and Yimeng Chen. “HW-TSC’s Participation at WMT 2020 Automatic Post Editing Shared Task.” WMT 2020.

[9] Sutskever, Ilya, Oriol Vinyals, and Quoc V. Le. "Sequence to sequence learning with neural networks." Advances in neural information processing systems. 2014.

[10] Bahdanau, Dzmitry, Kyung Hyun Cho, and Yoshua Bengio. "Neural machine translation by jointly learning to align and translate." 3rd International Conference on Learning Representations, ICLR 2015. 2015.

[11] Vaswani, Ashish, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. "Attention is all you need." Advances in neural information processing systems. 2017.

References

[12] Aakash Banerjee, Aditya Jain, Shivam Mhaskar, Sourabh Dattatray Deoghare, Aman Sehgal, and Pushpak Bhattacharya. 2021. Neural machine translation in low-resource setting: a case study in English-Marathi pair. In Proceedings of Machine Translation Summit XVIII: Research Track, pages 35–47, Virtual. Association for Machine Translation in the Americas.

[13] Yunsu Kim, Petre Petrov, Pavel Petrushkov, Shahram Khadivi, and Hermann Ney. 2019a. Pivot-based transfer learning for neural machine translation between non-English languages. 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), pages 866–876, Hong Kong, China. Association for Computational Linguistics.

[14] Melvin Johnson, Mike Schuster, Quoc V. Le, Maxim Krikun, Yonghui Wu, Zhifeng Chen, Nikhil Thorat, Fernanda Viégas, Martin Wattenberg, Greg Corrado, Macduff Hughes, and Jeffrey Dean. 2017. Google’s multilingual neural machine translation system: Enabling zero-shot translation. Transactions of the Association for Computational Linguistics, 5:339–351.

References

[15] Utiyama Masao, and Hitoshi Isahara. 2007. A comparison of pivot methods for phrase-based statistical machine translation. *North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 484-491.

[16] Dabre Raj, Fabien Cromieres, Sadao Kurohashi, and Pushpak Bhattacharyya. 2015. Leveraging small multilingual corpora for smt using many pivot languages. North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 1192-1202.

[17] Yunsu Kim, Petre Petrov, Pavel Petrushkov, Shahram Khadivi, and Hermann Ney. 2019. Pivot-based transfer learning for neural machine translation between non-English languages. Empirical Methods in Natural Language Processing (EMNLP), pp. 866-876.

[18] Wei-Jen Ko, Ahmed El-Kishky, Adithya Renduchintala, Vishrav Chaudhary, Naman Goyal, Francisco Guzmán, Pascale Fung, Philipp Koehn, and Mona Diab. 2021. Adapting high-resource NMT models to translate low-resource related languages without parallel data. The 59th Annual Meeting of the Association for Computational Linguistics , pp. 802-812.

References

[19] Firat, O., Sankaran, B., Al-onaizan, Y., Yarman Vural, F. T., and Cho, K. (2016). Zero-resource translation with multilingual neural machine translation. In Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, pages 268–277, Austin, Texas. Association for Computational Linguistics.

[20] Garmash, E. and Monz, C. (2016). Ensemble learning for multi-source neural machine translation. In Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers, pages 1409–1418, Osaka, Japan. The COLING 2016 Organizing Committee

[21] Mach´aˇcek, D., Pol´ak, P., Bojar, O., and Dabre, R. (2023). Robustness of multi-source MT to transcription errors. In Findings of the Association for Computational Linguistics: ACL 2023, pages 3707–3723, Toronto, Canada. Association for Computational Linguistics.

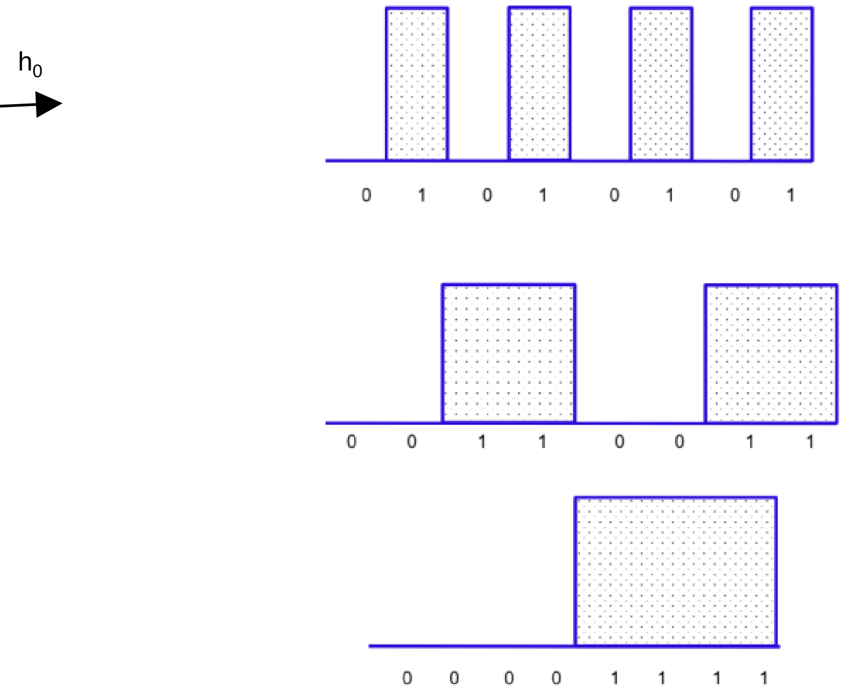
[22] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. (2017). Attention is all you need. Advances in neural information processing systems, 30.

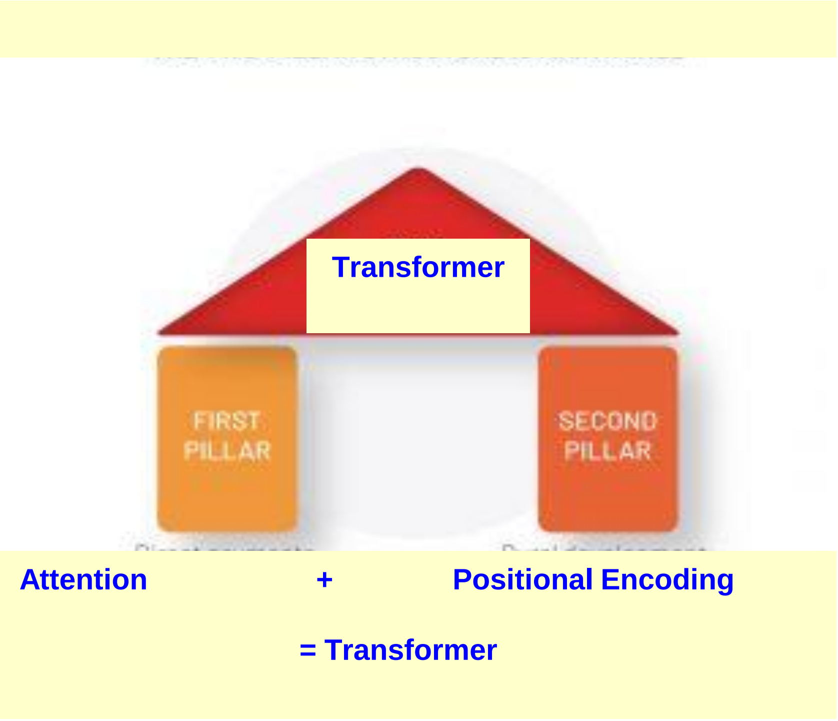
References

[23] Barret Zoph, Deniz Yuret, Jonathan May, and Kevin Knight. 2016. Transfer Learning for Low-Resource Neural Machine Translation. In Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, pages 1568–1575, Austin, Texas. Association for Computational Linguistics.

1-slide recap

[nesu]	“I carry”
[ponese]	“He will carry”
[nese]	“He carries”
[nesou]	“They carry”
[yedu]	“I drive”
[plavou]	“They swim”





Transformer

Attention + Positional Encoding

= Transformer

Bahadanu, 2015

“In order to address this issue (long distance dependency), we introduce an extension to the encoder–decoder model which learns to align and translate jointly. Each time the proposed model generates a word in a translation, it (soft-) searches for a set of positions in a source sentence where the most relevant information is concentrated. The model then predicts a target word based on the context vectors associated with these source positions and all the previous generated target words.”

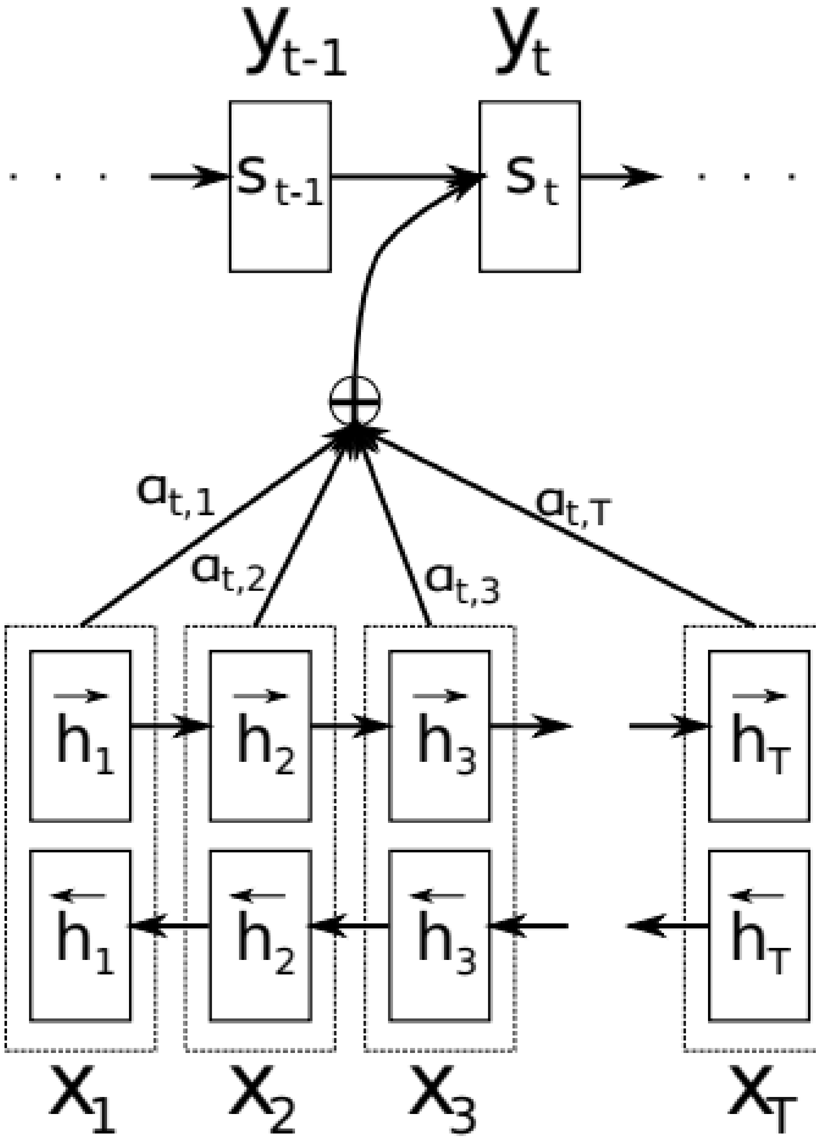
Bahadanu 2015- contribution

“proposed approach of jointly learning to align and translate achieves significantly improved translation performance over the basic encoder–decoder approach. The improvement is more apparent with longer sentences. On the task of English-to-French translation, the proposed approach achieves performance comparable to phrase-based system. Furthermore, qualitative analysis reveals that the proposed model finds a linguistically plausible (soft-)alignment between a source sentence and the corresponding target sentence”

Bahadanu 2015 again- motivation

“It encodes the input sentence into a sequence of vectors and chooses a subset of these vectors adaptively while decoding the translation. This frees a neural translation model from having to squash all the information of a source sentence, regardless of its length, into a fixed-length vector. We show this allows a model to cope better with long sentences.”

Attention Schematic (1/2)



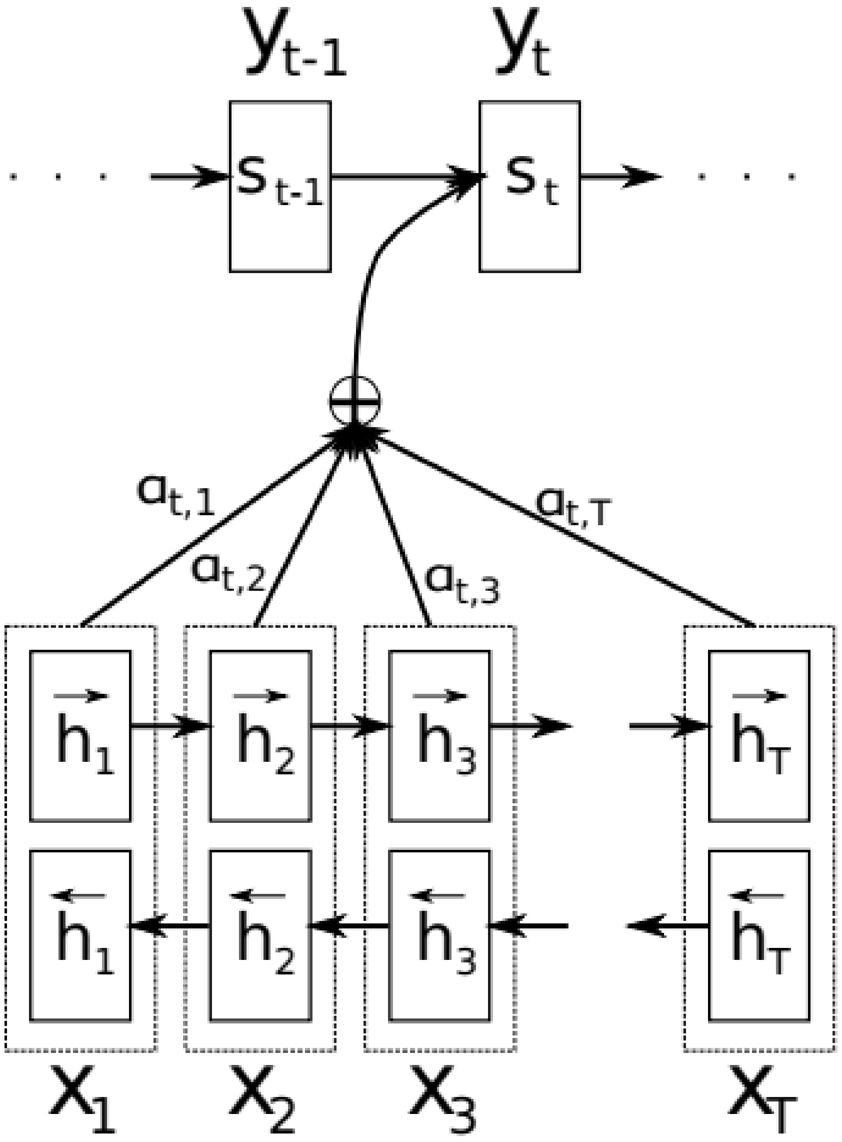
$$h_t = f(x_t, h_{t-1})$$

$$c = g(\{h_1, h_2, h_3, \ldots h_{T_x}\})$$

$$P(\bar{y}) = \prod_{t=1}^{T} P(y_t \mid \{y_1, y_2, y_3, \ldots y_{t-1}\}, c)$$

$$P(y_t \mid \{y_1, y_2, y_3, \ldots y_{t-1}\}, c) = g(y_{t-1}, s_t, c)$$

Attention Schematic (2/2)



$$p(y_i \mid y_1, y_2, \ldots y_{i-1}) = g(y_{i-1}, s_i, c_i)$$

$$s_i = f(s_{i-1}, y_{i-1}, c_i)$$

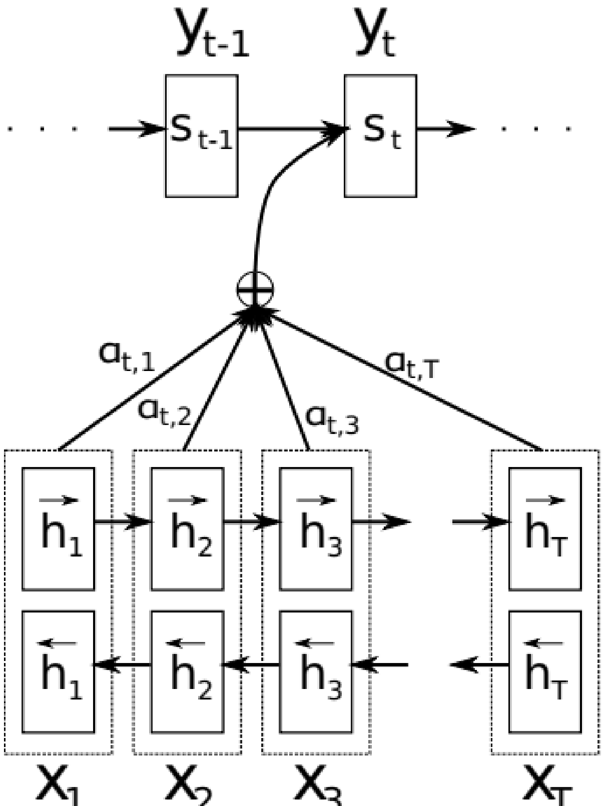
$$c_i = \sum_{j=1}^{T_x} \alpha_{ij} h_j$$

$$\alpha_{ij} = \frac{\exp(e_{ij})}{\sum_{k=1}^{T_x} \exp(e_{ik})}$$

$$e_{ij} = a(s_{i-1}, h_j)$$

BiRNN

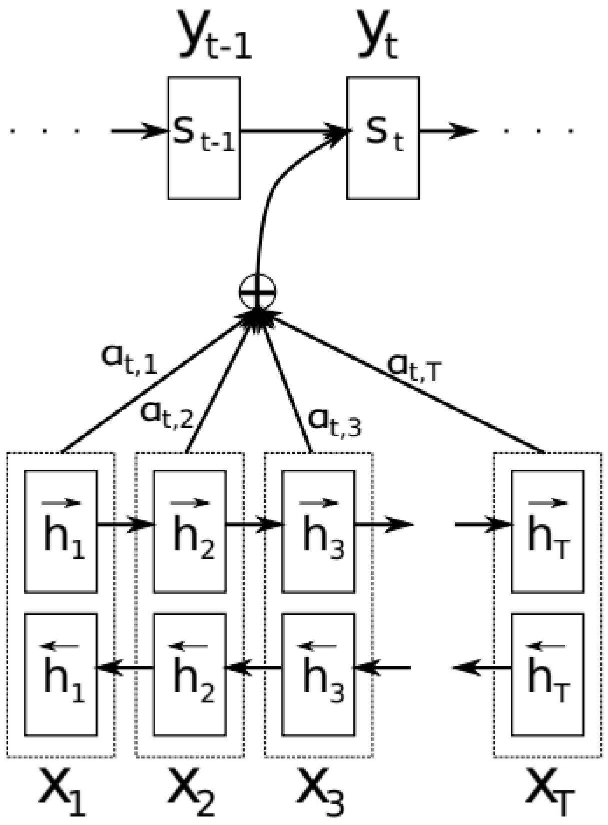
- Consists of forward and backward RNN’s; The forward RNN reads the input sequence from x_1 to x_{Tx}
- The backward RNN: *backward hidden states*



Annotation

- Annotation h_j consists of concatenation of forward and backward hidden states

- Contains the summaries of both the preceding words and the following words



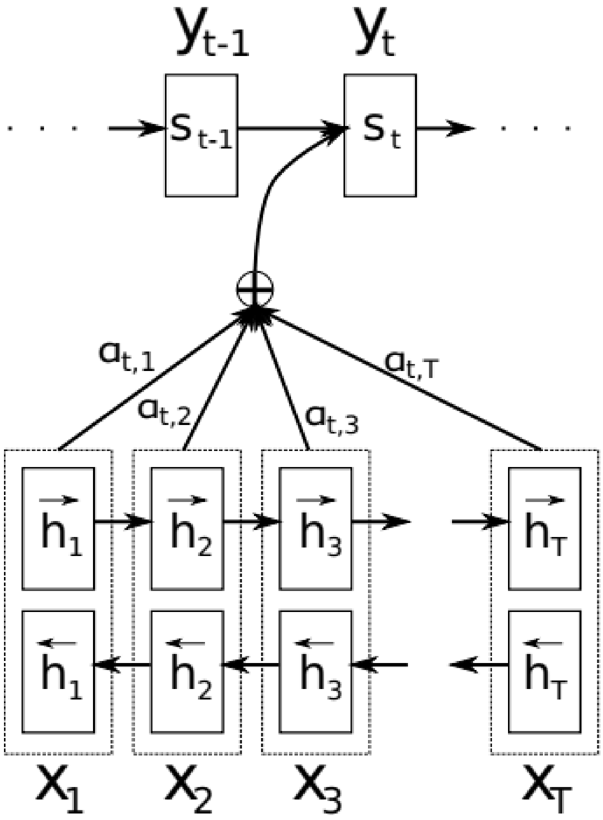
Context vector

Sequence of annotations used by the decoder and the alignment model to compute the context vector

$$c_i = \sum_{j=1}^{T_x} \alpha_{ij} h_j$$

$$\alpha_{ij} = \frac{\exp(e_{ij})}{\sum_{k=1}^{T_x} \exp(e_{ik})}$$

$$e_{ij} = a(s_{i-1}, h_j)$$



Experiment

- English-to-French translation

- Use the bilingual, parallel corpora provided by ACL WMT ’14

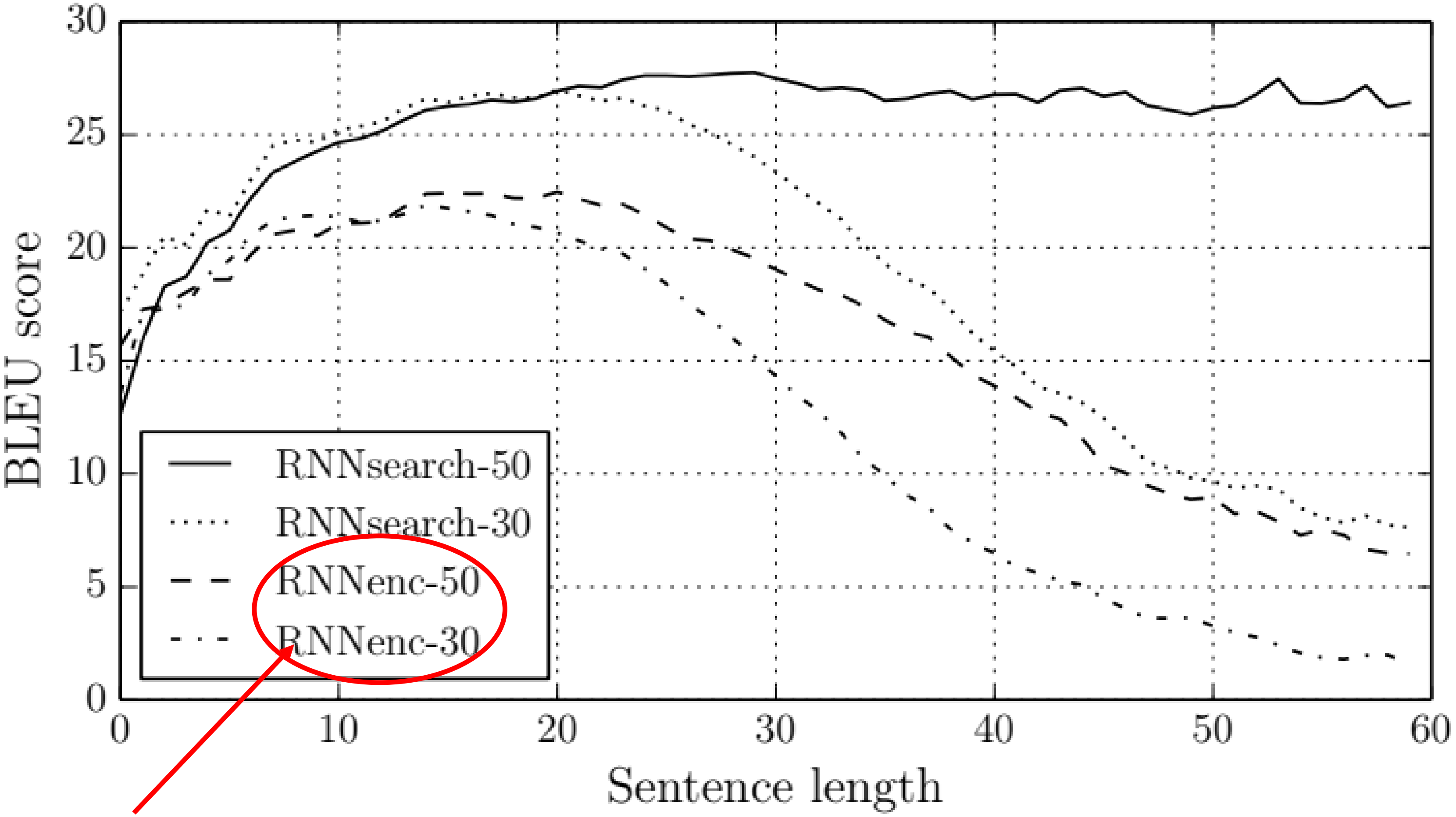
- Baseline: RNN Encoder–Decoder (Cho *et al.* 2014).

Dataset

- WMT ’14
  - Europarl (61M words),
  - News commentary (5.5M)
  - UN (421M) and
  - Two crawled corpora of 90M and 272.5M

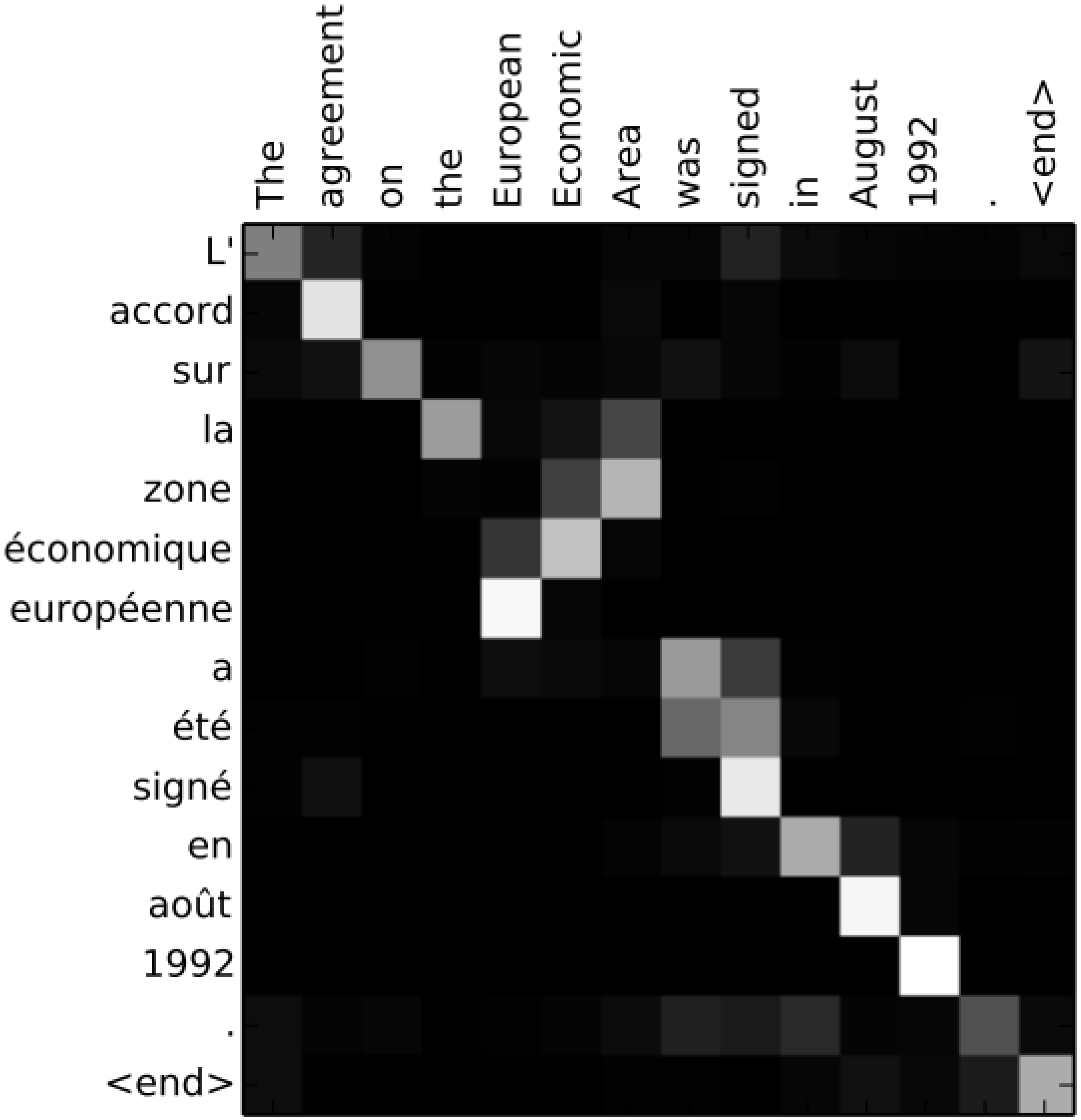
- Total 850M words.

Performance

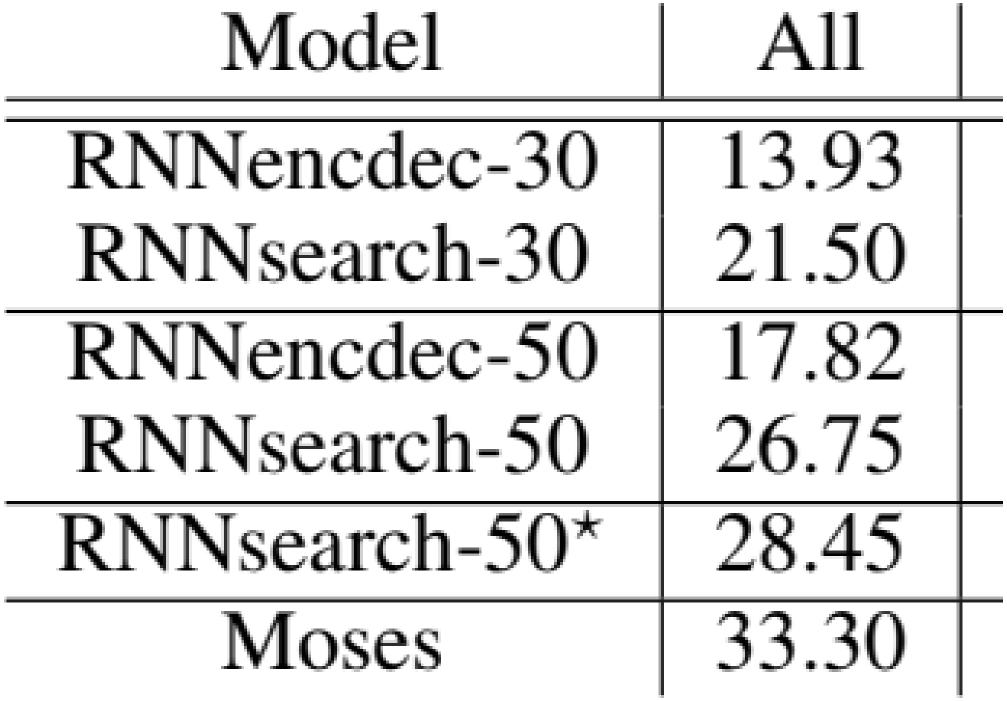


Baseline

Discovered Alignment example

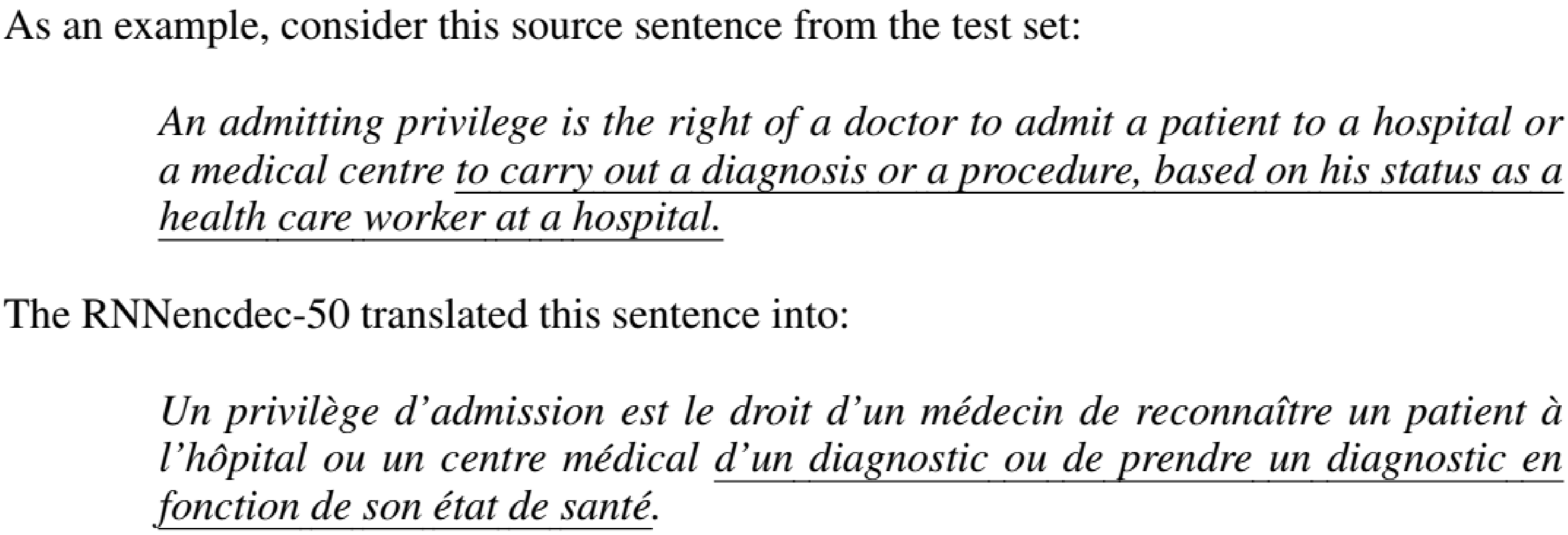


BLEU scores



Model	All
RNNencdec-30	13.93
RNNsearch-30	21.50
RNNencdec-50	17.82
RNNsearch-50	26.75
RNNsearch-50*	28.45
Moses	33.30

Qualitative (1/2)



As an example, consider this source sentence from the test set:

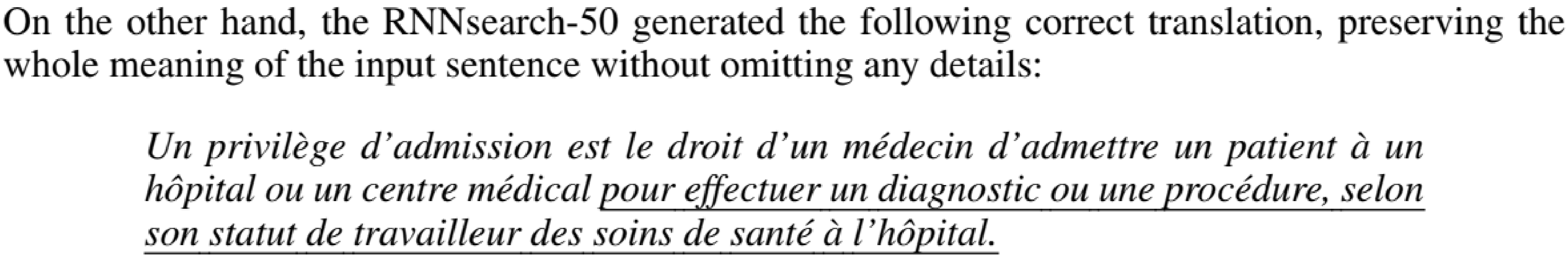
> *An admitting privilege is the right of a doctor to admit a patient to a hospital or a medical centre to carry out a diagnosis or a procedure, based on his status as a health care worker at a hospital.*

The RNNencdec-50 translated this sentence into:

> *Un privilège d'admission est le droit d'un médecin de reconnaître un patient à l'hôpital ou un centre médical d'un diagnostic ou de prendre un diagnostic en fonction de son état de santé.*

An admitting privilege is the right of a physician to recognize a patient ashospital or medical center for a diagnosis or to take a diagnosis independing on his state of health.

Qualitative (2/2)



On the other hand, the RNNsearch-50 generated the following correct translation, preserving the whole meaning of the input sentence without omitting any details:

> *Un privilège d'admission est le droit d'un médecin d'admettre un patient à un hôpital ou un centre médical pour effectuer un diagnostic ou une procédure, selon son statut de travailleur des soins de santé à l'hôpital.*

An admitting privilege is the right of a physician to admit a patient to anhospital or medical center to perform a diagnosis or procedure, according tohis status as a health care worker at the hospital.