Nearest Neighbor Machine Translation

(Khandelwal et al., ICLR 2021)

Hiroyuki Deguchi

deguchi.hiroyuki.db0@is.naist.jp

2021/05/12 NAIST MT study group

Links

Paper

`https://openreview.net/pdf?id=7wCBOfJ8hJM`

Introduction

Combining NMT and k-nearest-neighbors based EBMT models

Summary

- The decoder **retrieves** translation examples from training data **at test time**.
- Learned NMT models can be used **w/o additional training**.

Contributions

- The proposed method:
  - improves a **SOTA De-En translation model** by **1.5 BLEU**.
  - can **adapt models to diverse domains** by using a in-domain datastore, improving results by an average of **9.2 BLEU**.
  - improves a **multilingual model** by **3 BLEU** on En-{De, Zh} translation.

k Nearest Neighbors (kNN) classification

Non-parametric classification method

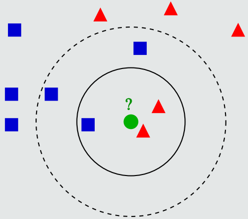
- The object is assigned to the class most common among its k nearest neighbors.

Example of k-NN classification:

- green dot →

  $k = 3$: red triangle class

  $k = 5$: blue square class

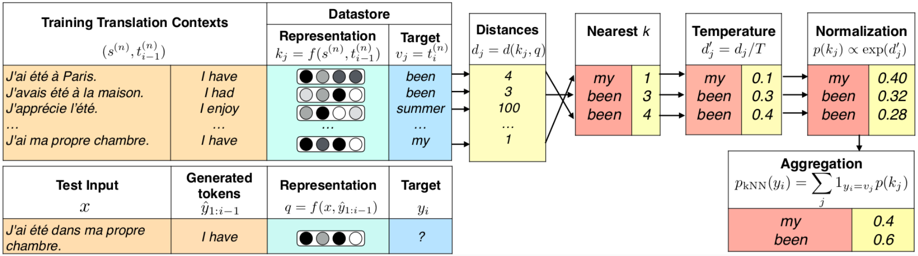


https://en.wikipedia.org/wiki/K-nearest_neighbors_algorithm

Proposed Method

Nearest Neighbor Machine Translation

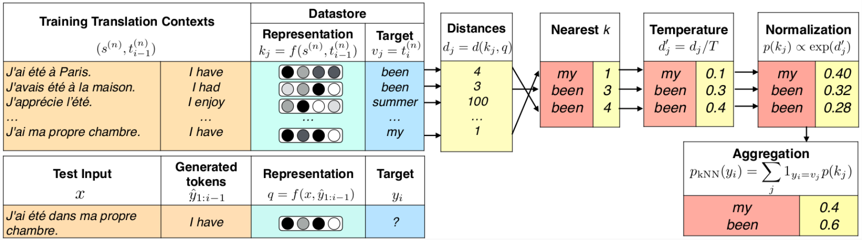
Augmenting the decoder of a pre-trained NMT model with a nearest neighbor retrieval at each time step



Datastore: Datastore is constructed from parallel corpus by a single forward pass over each example.

$q = f(x, \hat{y}_{1:i-1})$: an intermediate representation of the decoder

Nearest Neighbor Machine Translation



At test time

1. Search k nearest neighbors from the datastore based on distances between q and each intermediate representation.
2. Compute the distribution by applying a softmax with temperature to each k nearest neighbors.
3. Aggregate the 2. results and obtain probability $p_{kNN}(y_i)$.
4. Interpolate the NMT and kNN distribution.

Datastore creation

Store the entire translation context, preliminarily

$$(\mathcal{K}, \mathcal{V}) = \{(f(s, t_{1:i-1}), t_i), \forall t_i \in t \mid (s, t) \in (\mathcal{S}, \mathcal{T})\}$$

- f : NMT model (returns the decoder's intermediate representations)
- $(\mathcal{S}, \mathcal{T})$: parallel corpus
- $\mathcal{K}$: intermediate representations, $\mathcal{V}$: target tokens t_i
    - Conditioning on the source is implicit via the keys
    - The values are only target language tokens

Generation

Compute distance-based probability distribution by applying a softmax with temperature

$$p_{kNN}(y_i|x, \hat{y}_{1:i-1}) \propto \sum_{(k_j,v_j)\in\mathcal{N}} \mathbb{1}_{y_i=v_j} \exp\left(\frac{-d(k_j, f(x, \hat{y}_{1:i-1}))}{T}\right)$$

- $\hat{y}$: generated tokens
- $\mathcal{N}$: k nearest neighbors according to squared-L^2 distance

Interpolate with the NMT output distribution

$$p(y_i|x, \hat{y}_{1:i-1}) = \lambda \underbrace{p_{kNN}(y_i|x, \hat{y}_{1:i-1})}_{\text{kNN distribution}} + (1-\lambda) \underbrace{p_{MT}(y_i|x, \hat{y}_{1:i-1})}_{\text{NMT distribution}}$$

Experimental Setup

NMT Model

- Transformer big (`Fairseq`)

Tasks

- WMT19 De-En news translation
- Multilingual MT
  - train: CCMatrix
  - test: newstest2018, newstest2019, TED Talks
- Domain adaptation:
  - Medical, Law, IT, Koran, Subtitles

Experimental Setup

Implementation of kNN-MT

- kNN: `Faiss` (a library for fast k nearest neighbors search)
- Key: 1024-dimensional input to the final decoder layer FFN (quantized to 64-bytes)
  - Multilingual MT: 131K clusters
  - Domain adaptation: 4K clusters
- Inference: Query the datastore for 64 neighbors while searching 32 clusters

Computational Cost

kNN-MT adds some computational overhead

Datastore creation

- A single forward pass over all examples
  - Same as one epoch

Inference

- Retrieving 64 keys from a datastore containing billions of items
- A generation speed that is two orders of magnitude slower than the base MT system

Experiments

WMT'19 De-En

Model	BLEU (%)
Baseline	37.59
+kNN-MT	**39.08 (+1.5)**

- Improving by 1.5 BLEU % w/o additional training

Multilingual Machine Translation

Retrieving neighbors from same source language data

	de-en	**ru-en**	**zh-en**	**ja-en**	**fi-en**	**lt-en**	**de-fr**	**de-cs**	**en-cs**
Test set sizes	2,000	2,000	2,000	993	1,996	1,000	1,701	1,997	2,000
Base MT	34.45	36.42	24.23	12.79	25.92	29.59	32.75	21.15	22.78
+kNN-MT	**35.74**	**37.83**	**27.51**	13.14	26.55	29.98	**33.68**	21.62	**23.76**
Datastore Size	5.56B	3.80B	1.19B	360M	318M	168M	4.21B	696M	533M

	en-de	**en-ru**	**en-zh**	**en-ja**	**en-fi**	**en-lt**	**fr-de**	**cs-de**	**Avg.**
Test set sizes	1,997	1,997	1,997	1,000	1,997	998	1,701	1,997	-
Base MT	36.47	26.28	30.22	21.35	21.37	17.41	26.04	22.78	26.00
+kNN-MT	**39.49**	**27.91**	**33.63**	**23.23**	22.20	18.25	**27.81**	23.55	**27.40**
Datastore Size	6.50B	4.23B	1.13B	433M	375M	204M	3.98B	689M	-

Multilingual Machine Translation

Retrieving neighbors using English as the source language

	Ted Talks					Newstest2019			Avg.
	de-ja	**ru-ja**	**uk-ja**	**de-ru**	**de-zh**	**fr-de**	**cs-de**	**de-cs**	
Test set sizes	4,442	5,090	3,560	4,288	4,349	1,701	1,997	1,997	-
Base MT	10.11	9.69	8.36	17.24	20.48	26.04	22.78	21.15	16.98
+kNN-MT (en-$*$)	11.08	10.42	9.64	18.02	21.22	27.85	23.71	21.74	17.96
Datastore Size	433M	433M	433M	4.23B	1.13B	6.50B	6.50B	533M	-

Domain Adaptation

Domain-specific, out-of-domain, and multi-domain datastores

	Newstest 2019	Medical	Law	IT	Koran	Subtitles	Avg.
Test set sizes	2,000	2,000	2,000	2,000	2,000	2,000	-
Aharoni & Goldberg (2020):							
one model per domain	-	**56.5**	59.0	43.0	15.9	27.3	40.34
one model for all domains	-	53.3	57.2	42.1	20.9	27.6	40.22
best data selection method	-	54.8	58.8	43.5	**21.8**	27.4	41.26
Base MT	37.59	39.91	45.71	37.98	16.30	29.21	33.82
+kNN-MT:							
in-domain datastore	39.08	54.35	**61.78**	45.82	19.45	**31.73**	**42.63**
WMT'19 datastore	39.08	40.22	46.74	40.27	17.99	29.23	34.89
all-domains datastore	38.88	54.54	**61.11**	**48.63**	19.22	**31.70**	**43.04**
Datastore Size (in-domain)	770M	5.70M	18.3M	3.10M	450K	159M	-

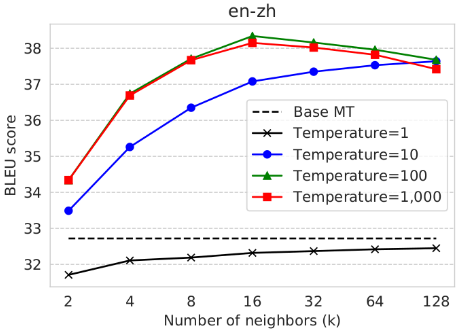
Tuning kNN-MT (on validation set)

of neighbors per query k

- $k = 64$ (the # of neighbors retrieved per query)
- *"we find that performance does not improve when retrieving a larger number of neighbors, and in some cases, performance deteriorates."* (noise?)

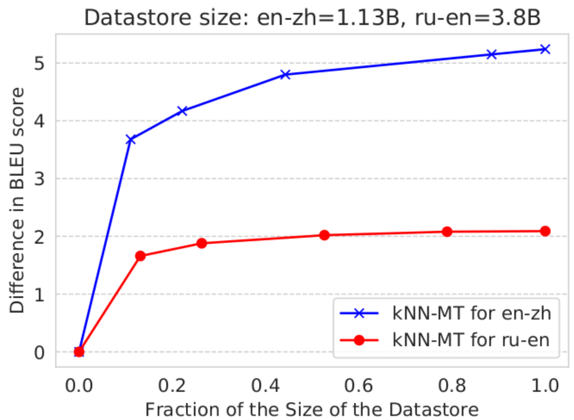
Softmax temperature T

- T greater than 1 will
  - flatten the distribution
  - increase diversity



Tuning kNN-MT (on validation set)

Datastore size



Qualitative Analysis

Generate w/ only the kNN distribution ($\lambda = 1$)

Test Input: *Dabei schien es, als habe Erdogan das Militär gezähmt.*
Generated tokens: *In doing so, it seems as if Erdogan has tamed the*

Training Set Translation Context (source and target)		Training Set Target	Context Probability
Dem charismatischen Ministerpräsidenten Recep Tayyip Erdoğan, der drei aufeinanderfolgende Wahlen für sich entscheiden konnte, ist es gelungen seine Autorität gegenüber dem Militär geltend zu machen.	*The charismatic prime minister, Recep Tayyip Erdoğan, having won three consecutive elections, has been able to exert his authority over the*	military	0.132
Ein bemerkenswerter Fall war die Ermordung des gemäßigten Premierministers Inukai Tsuyoshi im Jahre 1932, die das Ende jeder wirklichen zivilen Kontrolle des Militärs markiert.	*One notable case was the assassination of moderate Prime Minister Inukai Tsuyoshi in 1932, which marked the end of any real civilian control of the*	military	0.130
Sie sind Teil eines Normalisierungsprozesses und der Herstellung der absoluten zivilen Kontrolle über das Militär und bestätigen das Prinzip, dass niemand über dem Gesetz steht.	*They are part of a process of normalization, of the establishment of absolute civilian control of the*	military	0.129
Diese hart formulierte Erklärung wurde als verschleierte, jedoch unmissverständliche Warnung angesehen, dass das Militär bereit wäre einzuschreiten...	*That toughly worded statement was seen as a veiled but unmistakable warning that the*	military	0.123
...	...	...	...

Final kNN distribution: military = 1.0
Final Translation: In doing so, Erdogan seemed to have tamed the military.
Reference: In doing so, it seems as if Erdogan has tamed the military.

Related Work

Example-Based Machine Translation (EBMT)

A Framework of a mechanical translation between Japanese and English by analogy principle (Nagao, 1984)

- e.g. English-to-Japanese bilingual corpus

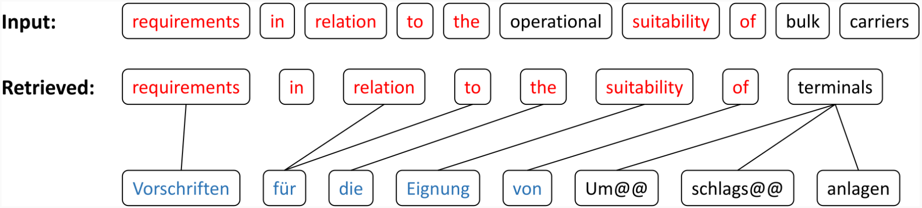
English	Japanese
Chick Corea is a fantastic **jazz pianist**.	チックコリアは素晴らしい**ジャズピアニスト**です。
Chick Corea is a fantastic **composer**.	チックコリアは素晴らしい**作曲家**です。

EBMT system learns three units from the above example:

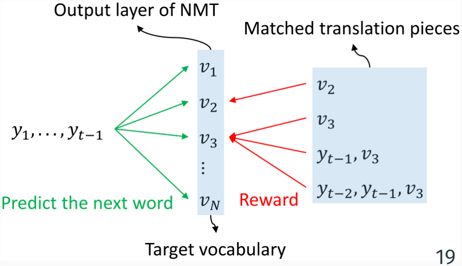
1. *"Chick Corea is a fantastic $\mathcal{X}$."* → *"チックコリアは素晴らしい $\mathcal{X}$ です。"*
2. *"jazz pianist"* → *"ジャズピアニスト"*
3. *"composer"* → *"作曲家"*

Incorporating retrieval mechanisms into NMT

Guiding Neural Machine Translation with Retrieved Translation Pieces (Zhang et al., 2018)



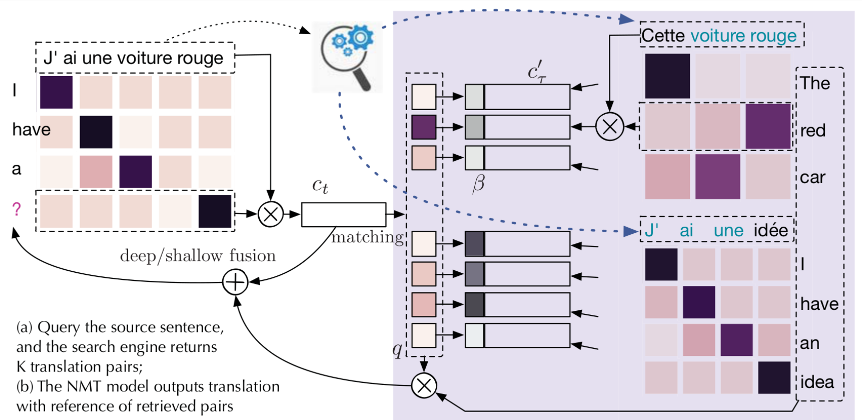
- Retrieve translation pieces (n-gram) of word-aligned parallel corpus
- Add rewards for n-grams that occur in the collected translation pieces



Retrieving translation examples

Search Engine Guided Neural Machine Translation (Gu et al., 2018)

- Retrieve examples similar to the test source sentence
- Incorporate retrieved information w/ *deep fusion* / *shallow fusion*



Augumenting source sequences with retrieved translations

Neural Fuzzy Repair: Integrating Fuzzy Matches into Neural Machine Translation **(Bulte et al., 2019)**

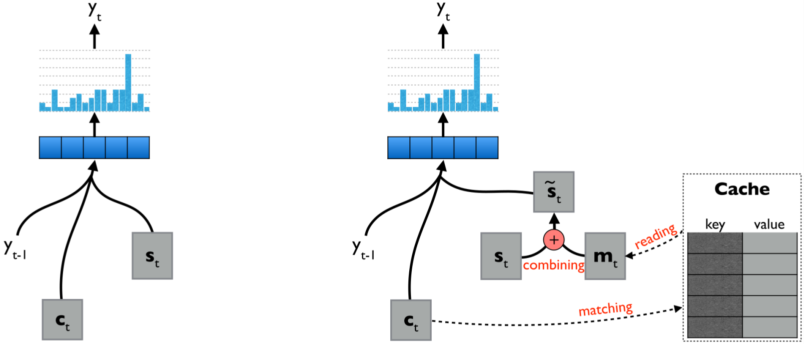
- Retrieve from translation memories by using edit distance based fuzzy-matching
- Augment source sequences with retrieved translations
  - e.g. “*こんにちは*” → “*こんにちは || hi || good evening || have a nice day*”
    - || : break token

Boosting Neural Machine Translation with Similar Translations **(Xu et al., 2020)**

- Improvement of “Neural Fuzzy Repair” (Bulte et al., 2019)
  - New score functions
    - N-gram matching score
    - Embedding-based score
  - Additional information
    - source tag, related target tag, un-related target tag, etc.

Saving and retrieving translation histories

- Proposed model awares cross-sentence context in documents to prevent translation inconsistency.



(a) Standard NMT (b) NMT augmented with a continuous cache

Conclusion

Summary

- kNN-MT can apply to any NMT model w/o further training.
- Similar contexts in a model's embedding space are more likely to be followed by similar next words, allowing the model to be improved by interpolation w/ kNN classifier.
- kNN-MT improves a SOTA model in-domain, leads to large gains out-of-domain, and can specialize a multilingual model for specific language-pairs.

Future work

- Improving efficiency
  - e.g. Down-sampling frequent target words in the datastore