南京大学 自然语言处理研究组 Natural Language Processing Group

南京大学 NANJING UNIVERSITY

k-Nearest-Neighbor Machine Translation

Shujian Huang and Wenhao Zhu

National Key Lab of Novel Software Technology

Department of Computer Science and Technology, Nanjing University

Outline

Part 1: Introduction

Development of Machine Translation

Proposals for Machine Translation (MT)
Weaver, 1949

Rule-based Machine Translation
since 1950s

Example-based Machine Translation
Nagao, 1980s

Statistical Machine Translation (SMT)
Brown et al., 1993
Koehn et al., 2003
Chiang et al., 2005

Neural Machine Translation (NMT)
Cho et al., 2014
Bahdanau et al., 2015

Deep Learning Era

Learning the Knowledge for Translation

- **In statistical machine translation, the knowledge are extracted as symbolic rules.**


Ankara is angry with the West for what it considers a weak response to the attempted takeover.
Add to that its
the EU and ste
chip away at T
The Russian le
support of the
Mind you, that
fear of regime
So the summit
to present wh
two countries
forces.
Still, despite th
differences.
The key one is
peacemaker b
It could be tel
presidents tol
the topic.
Turkey's presi
differences, w
There is no cle
on Syria.
But after mont
disaster when
it is surely bet
Royal Bank of

安卡拉对于西方世界对接管意图的微弱反应感到愤怒。
此外，安卡拉对于加入欧盟谈判的缓慢进展及普京的插手长期感到不满，普京热衷于利用政治寒意以及削弱土耳其与西方世界的关系。
由于在政变失败后拥护当选当局，俄罗斯领导人必将获得安卡拉的加分。
注意，这对于一直对政权更迭怀抱根深蒂固恐惧的莫斯科来说是一种馈赠。
因此，在这个金碧辉煌的海边宫殿所举行的会面使俄罗斯与土耳其两个被西方世界拒绝与虐待的国家结成盟友，一位分析师将其描述为“格格不入联盟”。
然而，尽管公开和解，但双方仍存在重大分歧。
叙利亚是关键因素之一。莫斯科近日在叙利亚扮演和事佬的角色，而俄罗斯与土耳其却支持相反派别。
可以预见到的是，在经过近三个小时的初步谈话后，两位总统在发布会上表示，尚未谈及那个话题。
土耳其总统刻意回避关于双方分歧的问题，而普京则予以强调。
双方就如何在叙利亚问题上求同存异未达成明确共识。
在北大西洋公约组织成员国土耳其击落俄战机所带来的数月公开敌对及引发大型灾难的可能下，两国领导人再次重启对话肯定是件好事。
苏格兰皇家银行将不再为苏格兰以外客户服务

Parallel Data (En-Chs)

30–>30	多年–>the, last, years
来–>over	友好–>friendly

(b) 单词翻译规则示例

30 多年–>the last 30 years	友好 合作–>friendly cooperation
30 多年 来–>over the last 30 years	的 友好–>friendly

(c) 短语翻译规则示例

30–>30	X 的 X–>X2 X1
X 多年–>the last X years	友好 合作–>friendly cooperation

(d) 层次翻译规则示例

QP(CD 30)(CD 多年)(LC 来)–>the last 30 years
友好 合作–>NP(JJ friendly)(NN cooperation)
QP(CD 30)(CD 多年)(LC 来)–>NP(DT the)(JJ last)(CD 30)(NNS years)

(e) 句法翻译规则示例

Translation Rules of Different Types (words, phrases, hierarchical phrases or syntactic phrases)

Learning the Knowledge for Translation

- In statistical machine translation, the knowledge are extracted as symbolic rules.
- These rules are later retrieved by an exact matching of symbols and assembled into sentences.
- Although general/syntactic placeholders are used to improve generalization, SMT suffers greatly from data sparseness.

30–>30	多年–>the, last, years
来–>over	友好–>friendly

(b) 单词翻译规则示例

30 多年–>the last 30 years	友好 合作–>friendly cooperation
30 多年 来–>over the last 30 years	的 友好–>friendly

(c) 短语翻译规则示例

30–>30	X 的 X–>X2 X1
X 多年–>the last X years	友好 合作–>friendly cooperation

(d) 层次翻译规则示例

QP(CD 30)(CD 多年)(LC 来)–>the last 30 years

友好 合作–>NP(JJ friendly)(NN cooperation)

QP(CD 30)(CD 多年)(LC 来)–>NP(DT the)(JJ last)(CD 30)(NNS years)

(e) 句法翻译规则示例

Translation Rules of Different Types (words, phrases, hierarchical phrases or syntactic phrases)

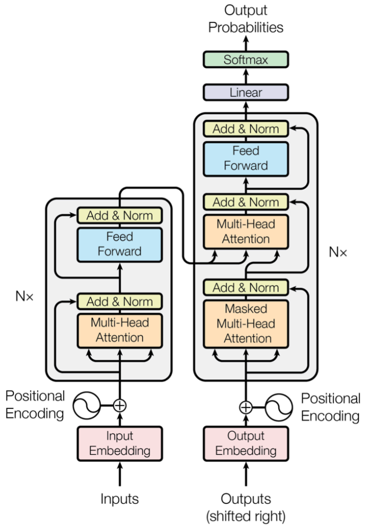
Learning the Knowledge for Translation

- **In neural machine translation, the knowledge is explicitly embedded in the parameters of the neural**


Ankara is angry with the West for what it considers a weak response to the attempted takeover.
Add to that its
the EU and ste
chip away at T
The Russian le
support of the
Mind you, that
fear of regime
So the summit
to present wh
two countries
forces.
Still, despite th
differences.
The key one is
peacemaker b
It could be tel
presidents tol
the topic.
Turkey's presi
differences, w
There is no cle
on Syria.
But after mont
disaster when
it is surely bett
Royal Bank of

安卡拉对于西方世界对接管意图的微弱反应感到愤怒。
此外，安卡拉对于加入欧盟谈判的缓慢进展及普京的插手长期感到不满，普京热衷于利用政治寒意以及削弱土耳其与西方世界的关系。
由于在政变失败后拥护当选当局，俄罗斯领导人必将获得安卡拉的加分。
注意，这对于一直对政权更迭怀抱根深蒂固恐惧的莫斯科来说是一种馈赠。
因此，在这个金碧辉煌的海边宫殿所举行的会面使俄罗斯与土耳其两个被西方世界拒绝与虐待的国家结成盟友，一位分析师将其描述为“格格不入联盟”。
然而，尽管公开和解，但双方仍存在重大分歧。
叙利亚是关键因素之一。莫斯科近日在叙利亚扮演和事佬的角色，而俄罗斯与土耳其却支持相反派别。
可以预见到的是，在经过近三个小时的初步谈话后，两位总统在发布会上表示，尚未谈及那个话题。
土耳其总统刻意回避关于双方分歧的问题，而普京则予以强调。
双方就如何在叙利亚问题上求同存异未达成明确共识。
在北大西洋公约组织成员国土耳其击落俄战机所带来的数月公开敌对及引发大型灾难的可能下，两国领导人再次重启对话肯定是件好事。
苏格兰皇家银行将不再为苏格兰以外客户服务

Parallel Data (En-Chs)

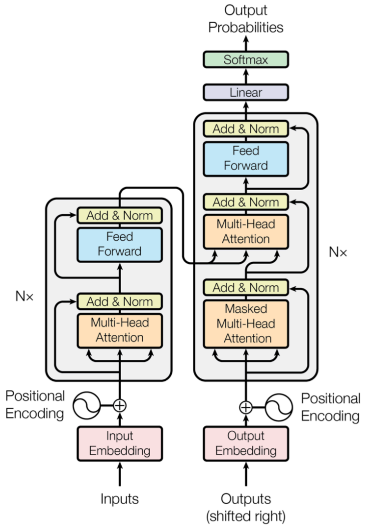


Transformer

Attention is all you need. Vaswani et al. NIPS2017.

Learning the Knowledge for Translation

- In neural machine translation, the knowledge is explicitly embedded in the parameters of the neural networks.
- Words are represented as continuous vectors, i.e. embeddings.
- Translation is performed by the computation with network parameter
- Better at capturing semantic relations than exact matching.



Transformer

Attention is all you need. Vaswani et al. NIPS2017.

Problems of the "Neural" Knowledge

- **The neural way of learning translation knowledge is better at generalization than the symbolic way**
  - employing computation instead of matching
  - learning big models from massive data
- **However, there are still several issues:**
  - Learnability: cannot memorize all translation knowledge in training data, especially for low-frequency events
  - Interpretability: cannot give evidence to support its translation decision
  - Extensibility: cannot incorporate new translation knowledge without updating neural parameters

Why not Combine the Two Philosophies?

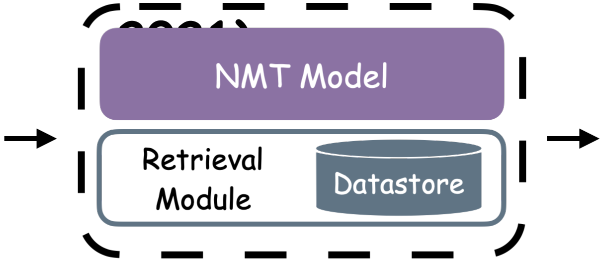
- Two systems are complementary.

Neural	learns general trends better generalization
Symbolic	memorizes specific events human interpretable easy to control or modify

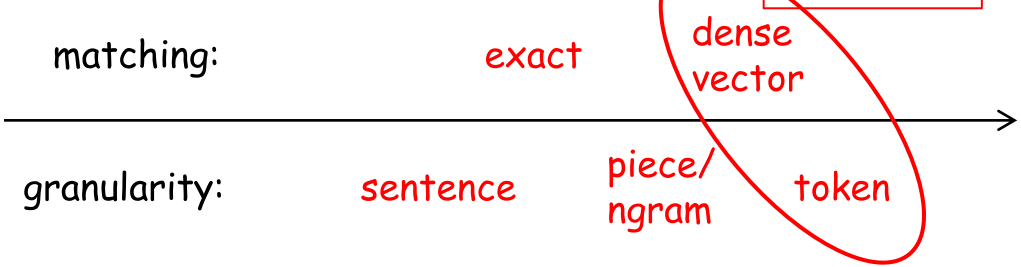
- Combining the two philosophies may bring further improvement to the whole learning system.

Retrieval-based Methods

- **Performing translation with the help of a symbolic datastore!**
  - Example based machine translation (Nagao 1984)
  - Search engine for sentences (Gu et al. 2018)
  - Search engine for translation pieces (Zhang et al. 2018)
  - n-gram retrieval using dense vectors (Bapna and Firat, 2019)
  - **token level retrieval using dense vectors (Khandelwal et al. 2021)** kNNMT



Retrieval-based MT system



Part 2: Basic Approach

The Idea of kNN-MT (previously kNN-LM)

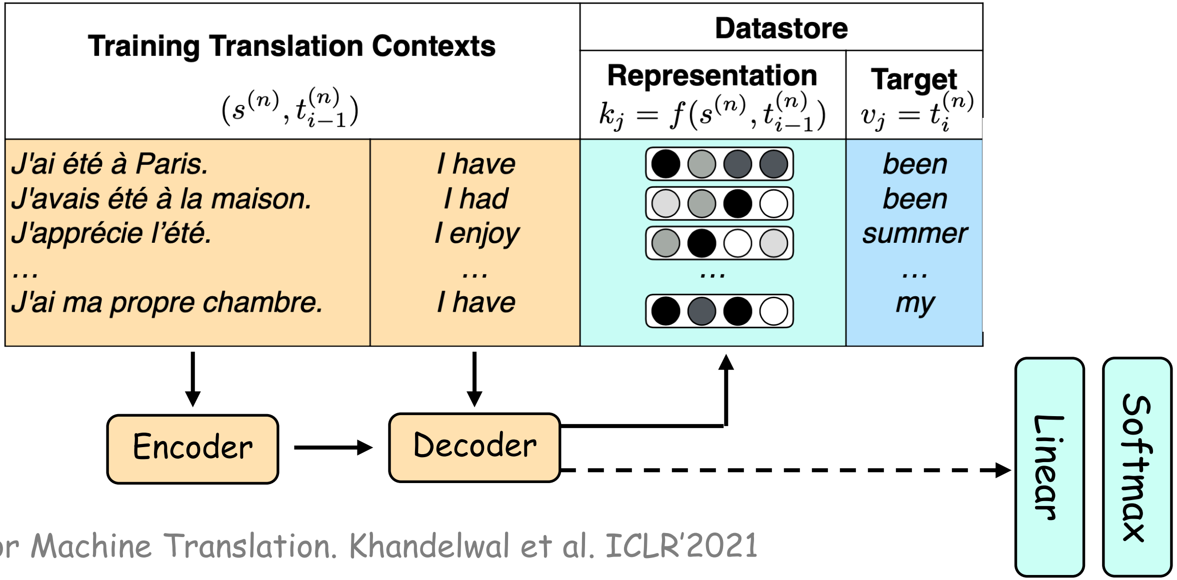
- **Build an extra symbolic datastore**
  - save linguistic knowledge as key-value pairs
  - (key: neural vector, value: symbolic token)

- **Leverage the extra datastore**
  - enable the neural model to retrieve knowledge from datastore
  - consider both systems and make final decision

Generalization through Memorization: Nearest Neighbor Language Models. Khandelwal et al. ICLR'2020
Nearest Neighbor Machine Translation. Khandelwal et al. ICLR'2021

kNN-MT

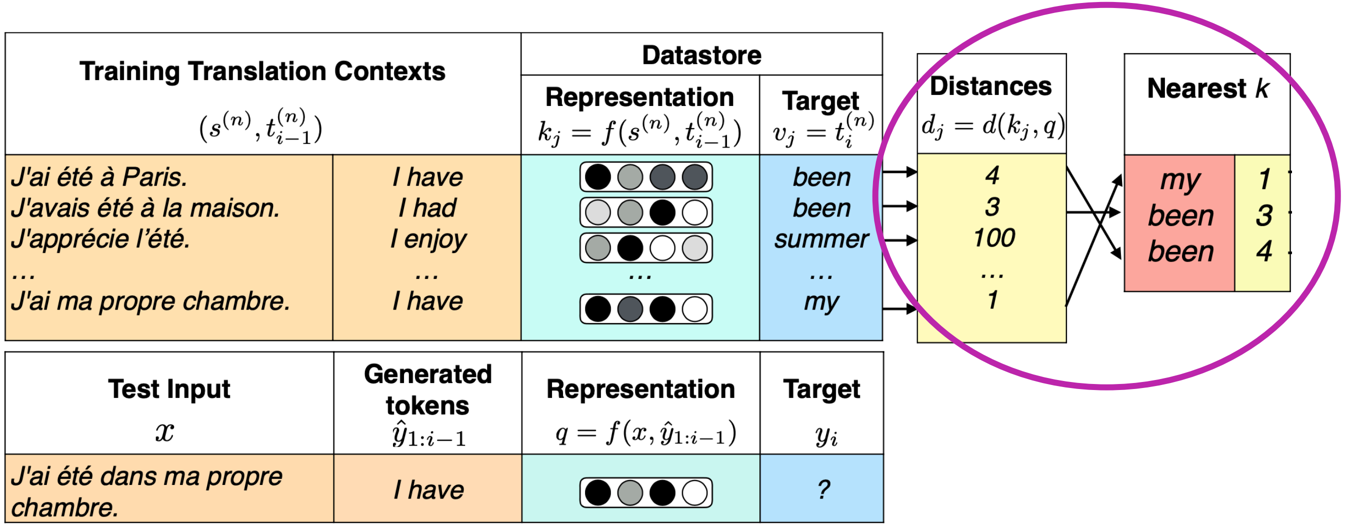
- **Step 1- Build datastore for NMT model**
  - A single forward pass over a bilingual corpus (e.g. training set)
  - (key: translation context representation, value: target token)



Nearest Neighbor Machine Translation. Khandelwal et al. ICLR'2021

kNN-MT

- **Step 2- Query datastore at each inference step**
  - Query the datastore with the representation of test translation context to retrieve k nearest neighbors

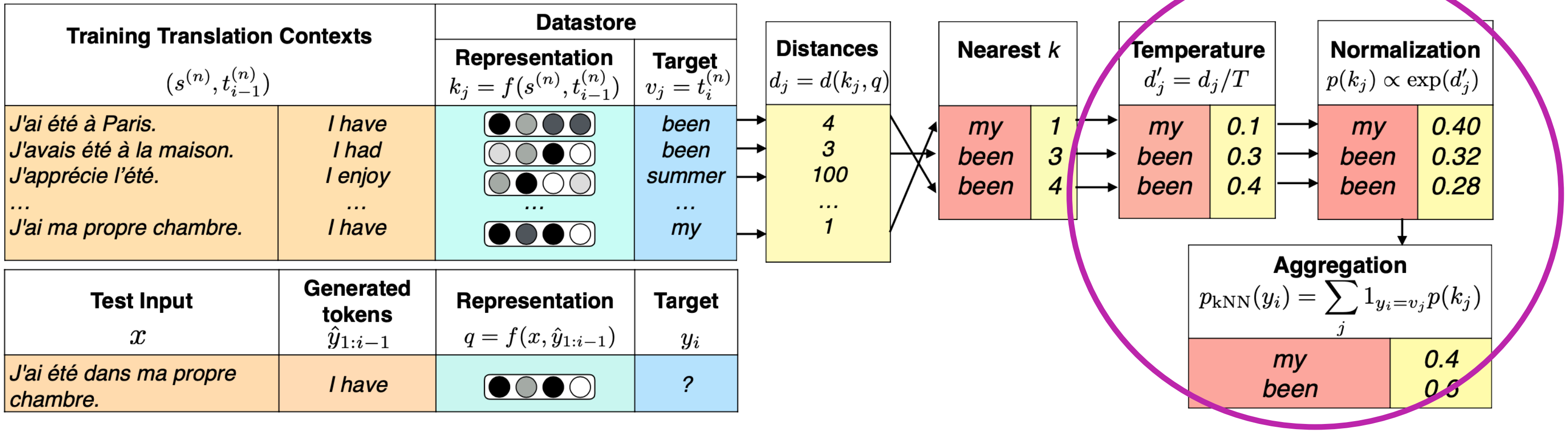


Nearest Neighbor Machine Translation. Khandelwal et al. ICLR' 2021

kNN-MT

- **Step 3 – Utilize query results**
  - Compute prediction distribution over vocabulary

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



Nearest Neighbor Machine Translation. Khandelwal et al. ICLR' 2021

kNN-MT

- **Step 4 – Get final prediction**
  - interpolate the model and kNN distribution with a weight λ

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

Nearest Neighbor Machine Translation. Khandelwal et al. ICLR' 2021

kNN-MT

- **Empirical results show that kNN-MT outperforms a simple NMT model in three settings:**
  - Single language pair MT
  - Multilingual MT
  - Domain adaptation

Nearest Neighbor Machine Translation. Khandelwal et al. ICLR' 2021

Single Language Pair MT

- NMT model: winner model of WMT'19 German-English News Translation task

- Datastore: 770M tokens of WMT'19 training data

- Main results
  - 37.59 BLEU -> 39.08 BLEU on newstest2019

- Even very strong translation models can be improved with a symbolic datastore of the training set.

Nearest Neighbor Machine Translation. Khandelwal et al. ICLR' 2021

Multilingual MT

- **kNN-MT achieves an average improvement of 1.4 BLEU across 17 language pairs.**

Test set sizes	**de-en** 2,000	**ru-en** 2,000	**zh-en** 2,000	**ja-en** 993	**fi-en** 1,996	**lt-en** 1,000	**de-fr** 1,701	**de-cs** 1,997	**en-cs** 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

Test set sizes	**en-de** 1,997	**en-ru** 1,997	**en-zh** 1,997	**en-ja** 1,000	**en-fi** 1,997	**en-lt** 998	**fr-de** 1,701	**cs-de** 1,997	**Avg.** -
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	-

Nearest Neighbor Machine Translation. Khandelwal et al. ICLR' 2021

Supervised Domain Adaptation in MT

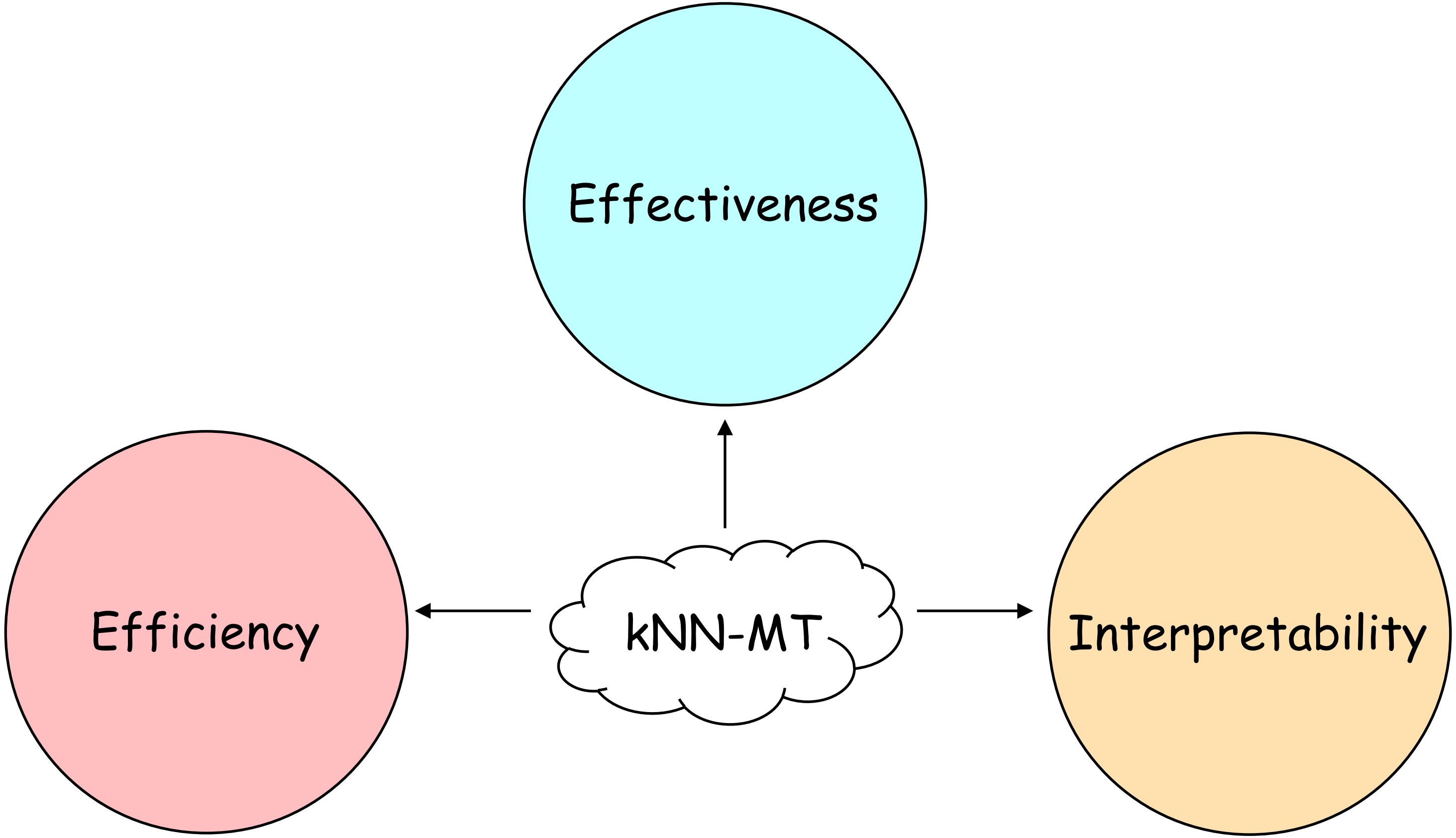
- kNN-MT presents a new paradigm for domain adaptation, with performance similar to fine-tuning.
- kNN-MT enables quick adaptation by switching datastores.

	Medical	Law	IT	Koran	Subtitles	Avg.
Test set sizes	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	39.91	45.71	37.98	16.30	29.21	33.82
+*k*NN-MT:						
in-domain datastore	54.35	**61.78**	45.82	19.45	**31.73**	**42.63**

Nearest Neighbor Machine Translation. Khandelwal et al. ICLR' 2021

Part 3: Dive into kNN-MT

Recent Advances in kNN-MT



Effectiveness

- **Although ability demonstrated in previous scenarios, there are still issues and issues affect the effectiveness.**
  - stability issues
  - resource issues

Adaptive Nearest Neighbor Machine Translation. Zheng et al. ACL'2021
Learning Kernel-Smoothed Machine Translation with Retrieved Examples. Jiang et al. EMNLP'2021
Non-Parametric Online Learning from Human Feedback for Neural Machine Translation. Wang et al. AAAI'2022
Non-Parametric Unsupervised Domain Adaptation for Neural Machine Translation. Zheng et al. EMNLP'2021

Setting 1: Domain Adaptation

- **Hyper-parameter matters for kNN-MT !**
- **The number of nearest neighbors need to be tuned on the dev set, to avoid the two cases:**
  - too small – may overfit to closest neighbors
  - too large – may include irrelevant neighbors
- **It would be better to dynamically determine k at each decoding step.**
  - If there are more relevant neighbors, use a larger k.
  - Otherwise, use a smaller k.

Setting 1: Domain Adaptation

- **Evaluating relevance of retrieved knowledge**
  - Distance between query and key
    (close neighbors are more relevant)
  - Consistency among retrieved knowledge
    (consistent query results are more relevant)

Distance
d_1
d_2
…
d_{K-1}
d_K

Value
v_1
v_2
…
v_{k-1}
v_k

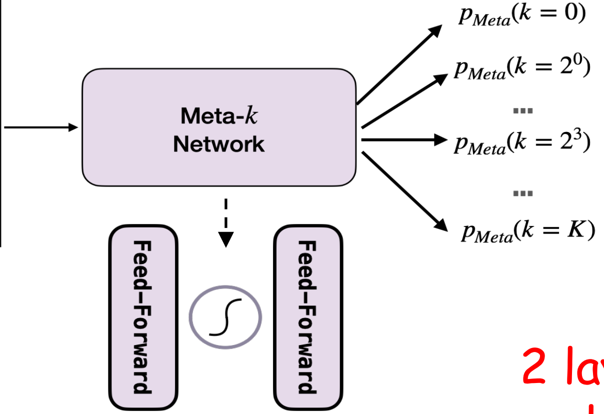
Distinct Values
1
$\|\{v_1, v_2\}\|$
…
$\|\{v_1, v_2, \ldots, v_{k-1}\}\|$
$\|\{v_1, v_2, \ldots, v_k\}\|$

Adaptive Nearest Neighbor Machine Translation. Zheng et al. ACL'2021.

Setting 1: Domain Adaptation

- Use a meta-k network to choose k from {0, 1, 2, 4, 8, ...} dynamically according to relevance of retrieved knowledge.
- The network could be very simple, because the input is simple.

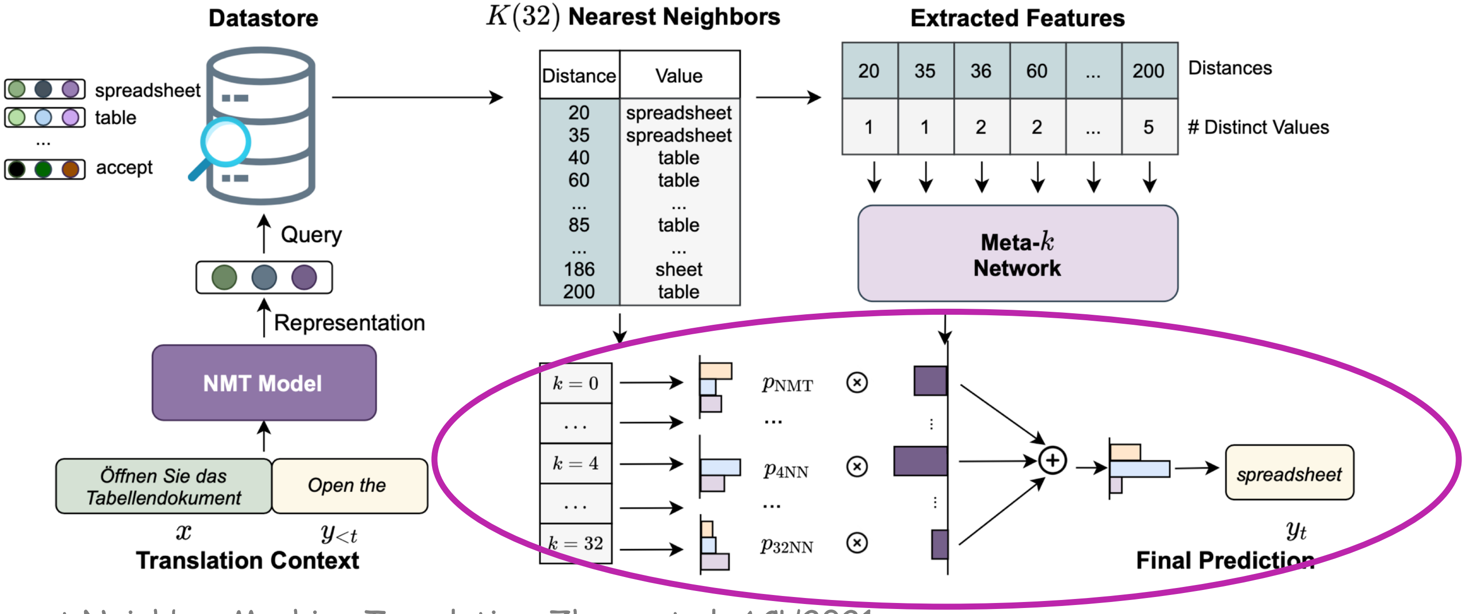
Distance	Distinct Values
d_1	1
d_2	$\lvert\{v_1, v_2\}\rvert$
...	...
d_{K-1}	$\lvert\{v_1, v_2, \ldots, v_{k-1}\}\rvert$
d_K	$\lvert\{v_1, v_2, \ldots, v_k\}\rvert$



2 layers, d=32, trained with only 2000 sentences

Setting 1: Supervised Domain Adaptation in MT

- **Plug meta-k network into kNN-MT**
  - Weighted sum different kNN distribution and model distribution (also eliminate the need to manually set λ)



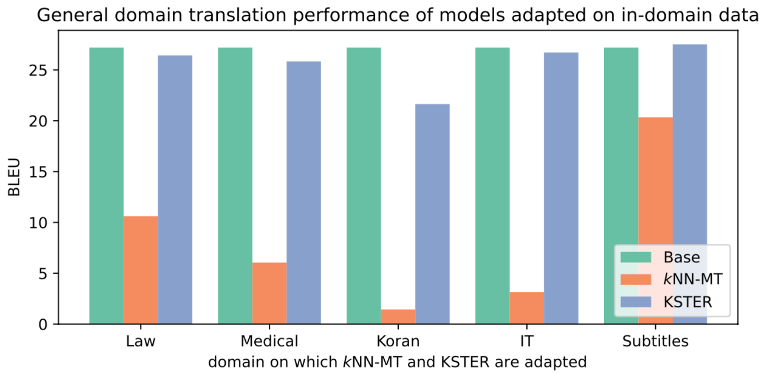
Setting 1: Domain Adaptation

- Outperform vanilla kNN-MT on different target domains
- Show better robustness (smaller variance) with different k

	Domain	IT (Base NMT: 38.35)			Med (Base NMT: 39.99)			Koran (Base NMT: 16.26)			Law (Base NMT: 45.48)			Avg (Base NMT: 35.02)		
	Model	V	U	A	V	U	A	V	U	A	V	U	A	V	U	A
K	1	42.19	41.21	42.52	51.41	50.32	51.82	18.12	17.15	18.10	58.76	58.05	58.81	42.62	41.68	42.81
	2	44.20	41.43	46.18	53.65	52.44	55.20	19.37	17.36	19.12	60.80	59.81	61.76	44.50	42.76	45.56
	4	44.89	42.31	47.23	54.16	53.01	55.84	19.50	17.88	19.69	61.31	60.75	62.89	44.97	43.49	46.41
	8	45.96	42.46	**48.04**	54.06	53.46	56.31	20.12	18.59	20.57	61.12	61.37	**63.21**	45.32	43.97	47.03
	16	45.36	43.05	47.71	53.54	54.08	**56.41**	20.30	19.45	**21.09**	60.21	61.52	63.07	44.85	44.53	**47.07**
	32	44.81	43.78	47.68	52.52	53.95	56.21	19.66	19.99	20.96	59.04	61.53	63.03	44.00	44.81	46.97
	$\sigma^2(K \geq 4)$	0.21	0.33	**0.08**	0.42	0.18	**0.05**	**0.10**	0.65	0.30	0.81	0.10	**0.01**	0.24	0.26	**0.07**

Setting 2: Multi-domain MT

- Adapted model often suffers from catastrophic forgetting problem and performs poorly on general domain.



- For general domain translation, it would be better to discard knowledge retrieved from specific-domain datastore.
- The decision should be made according to the domain!

Setting 2: Multi-domain MT

- **Use a learnable kernel to dynamically control the shape of kNN distribution.**

$$p_{kNN}(y_i|x, \hat{y}_{<i}) \propto \sum_{y_i=v_j} \exp(\frac{-d(\mathbf{q}_i, \mathbf{k}_j)}{T})$$

$$p_e(y_i|x, \hat{y}_{<i}) = \frac{\sum_{y_i=v_j} K(\mathbf{q}_i, \mathbf{k}_j; \sigma)}{\sum_j K(\mathbf{q}_i, \mathbf{k}_j; \sigma)}$$

- **Model the bandwidth σ of kernel function and mixing weight λ with learnable neural networks.**

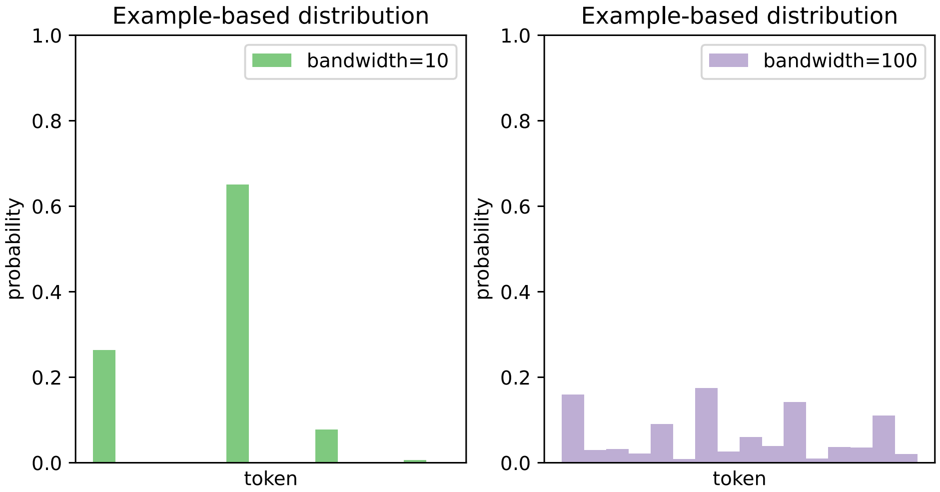
Learning Kernel-Smoothed Machine Translation with Retrieved Examples. Jiang et al. EMNLP'2021.

Setting 2: Multi-domain MT

- **Commonly used kernel function**
  - Gaussian Kernel $K_g(\mathbf{q}_i, \mathbf{k}_j; \sigma) = \exp(-\frac{\|\mathbf{q}_i - \mathbf{k}_j\|^2}{\sigma})$
  - Laplacian Kernel $K_l(\mathbf{q}_i, \mathbf{k}_j; \sigma) = \exp(-\frac{\|\mathbf{q}_i - \mathbf{k}_j\|}{\sigma})$

- **Estimate bandwidth σ at each decoding step with a learned affine network.**

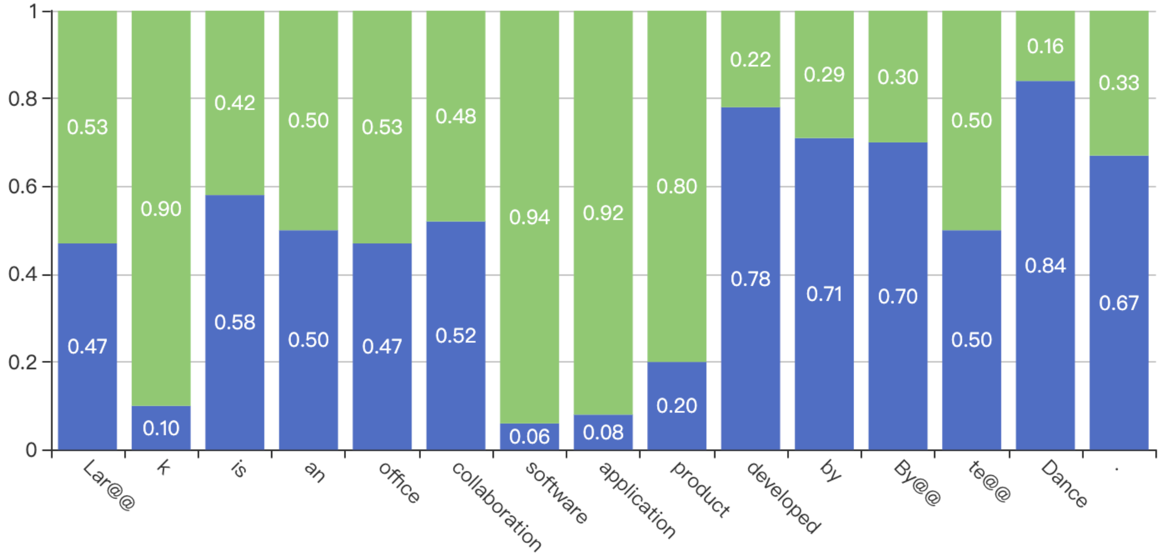
$$\sigma = \exp(\mathbf{W}_1[\mathbf{q}_i; \overline{\mathbf{k}}_i] + \mathbf{b}_1)$$



Setting 2: Multi-domain MT

- **Estimate weight λ at each decoding step with a learned multi-layer perceptron.**

$$\lambda = \mathrm{sigmoid}(\mathbf{W}_3 \mathrm{ReLU}(\mathbf{W}_2[\mathbf{q}_i; \widetilde{\mathbf{k}}_i] + \mathbf{b}_2) + \mathbf{b}_3)$$



Learning Kernel-Smoothed Machine Translation with Retrieved Examples. Jiang et al. EMNLP'2021.

Setting 2: Multi-domain MT

- **outperforms kNN-MT in domain-specific translation**
- **performs far better in general domain after adaptation**

Direction	Methods	Law	Medical	Koran	IT	Subtitles	Average-specific	Average-general (WMT14)
EN-DE	Base	33.36	30.54	10.16	22.99	20.65	23.54	**27.20**
	Finetuning	49.07	47.10	**25.98**	**36.28**	**26.00**	**36.89**	14.17
	kNN-MT	51.88	47.02	18.51	29.12	22.46	33.80	8.32
	KSTER	**53.63**	**49.18**	19.10	30.28	22.54	34.95	25.63
DE-EN	Base	36.80	33.36	11.24	29.21	23.13	26.75	**31.49**
	Finetuning	55.19	51.35	**22.87**	**41.88**	**28.33**	**39.92**	17.82
	kNN-MT	57.40	50.92	15.74	34.92	25.38	36.87	13.18
	KSTER	**59.41**	**53.40**	16.97	35.74	25.94	38.29	30.23

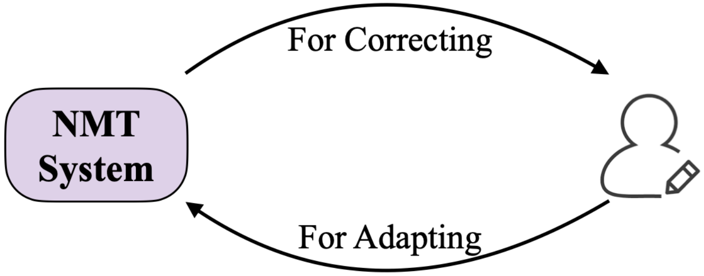
Setting 2: Multi-domain MT

- after a joint training on multiple domains, KSTER outperforms kNN-MT in most domains with a mixed datastore

Direction	Methods	General (WMT14)	Law	Medical	Koran	IT	Subtitles	Average-specific
EN-DE	Base	27.20	33.36	30.54	10.16	22.99	20.65	23.54
	Joint-training	27.25	45.02	44.52	15.43	**34.48**	**25.16**	32.92
	kNN-MT	24.72	51.24	46.54	**16.29**	29.55	21.80	33.08
	KSTER	**27.69**	**53.04**	**49.23**	15.94	31.82	22.63	**34.53**
DE-EN	Base	31.49	36.80	33.36	11.24	29.21	23.13	26.75
	Joint-training	31.62	50.95	47.48	**18.13**	**39.57**	**27.73**	36.77
	kNN-MT	25.87	57.38	50.83	14.57	37.56	22.86	36.64
	KSTER	**31.94**	**58.64**	**52.79**	15.24	36.90	25.15	**37.74**

Setting 3: Human-in-the-Loop MT

- **Interactive Machine Translation requires Online learning**
  - The human translators revise the machine-generated translations
  - The corrected translations are used to improve the NMT system
- **kNN fits this scenario well, because it learns the modification without changing the original model.**
- **However, in this setting the datastore is gradually increasing. It is crucial to dynamically decide the usage of datastore items.**

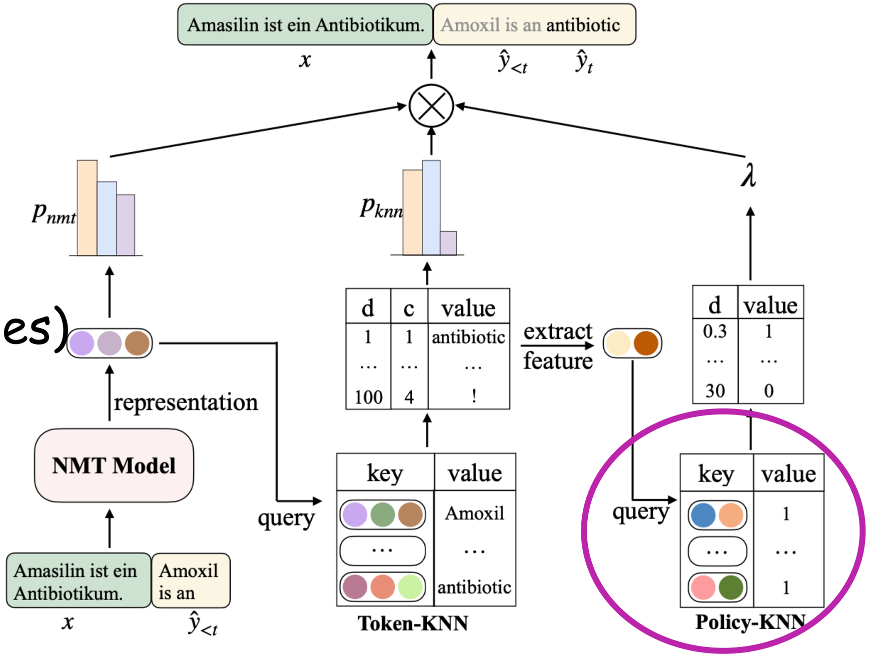


Setting 3: Human-in-the-Loop MT

- **Dynamically choose λ by querying a datastore that saves policy about whether retrieved knowledge can be trust (kNN over kNN).**

- **Policy Datastore**
  - Key: features of retrieved knowledge (distance + distinct values)
  - Value: gold value of λ

$$\lambda = \begin{cases} 1 & p_{\mathrm{KNN}}(y_t|\boldsymbol{x}, \boldsymbol{y}_{<t}) > p_{\mathrm{NMT}}(y_t|\boldsymbol{x}, \boldsymbol{y}_{<t}) \\ 0 & p_{\mathrm{KNN}}(y_t|\boldsymbol{x}, \boldsymbol{y}_{<t}) \le p_{\mathrm{NMT}}(y_t|\boldsymbol{x}, \boldsymbol{y}_{<t}) \end{cases}$$



Setting 3: Human-in-the-Loop MT

- achieve consistent improvements on different document lengths
- outperforms kNN-MT and online tuning

Bucket	0-50		50-100		100-200		200-500		500-1000		Average	
	BLEU	TER	BLEU	TER	BLEU	TER	BLEU	TER	BLEU	TER	BLEU	TER
Pre-Trained	43.8	**52.1**	43.1	52.8	38.3	54.0	41.9	53.8	40.8	53.4	41.6	53.2
Online Tuning	44.0	52.2	43.5	52.3	39.6	51.4	43.8	51.8	44.7	49.3	43.1	51.4
KNN-MT	43.8	52.6	43.6	52.5	40.0	53.1	43.8	52.3	44.2	50.8	43.1	52.3
Adaptive KNN-MT	29.7	70.2	28.9	70.3	35.9	58.4	37.2	61.2	48.2	50.3	36.0	62.1
KoK	**44.4**	**52.1**	**43.9**	**52.4**	**44.1**	**50.0**	**45.7**	**51.1**	**53.7**	**43.7**	**46.4**	**49.9**

Non-Parametric Online Learning from Human Feedback for Neural Machine Translation. Wang et al. AAAI'2022.

Setting 4: Unsupervised Domain Adaptation in MT

- Building datastore requires high-quality bilingual data, which is not available in unsupervised domain adaptation.

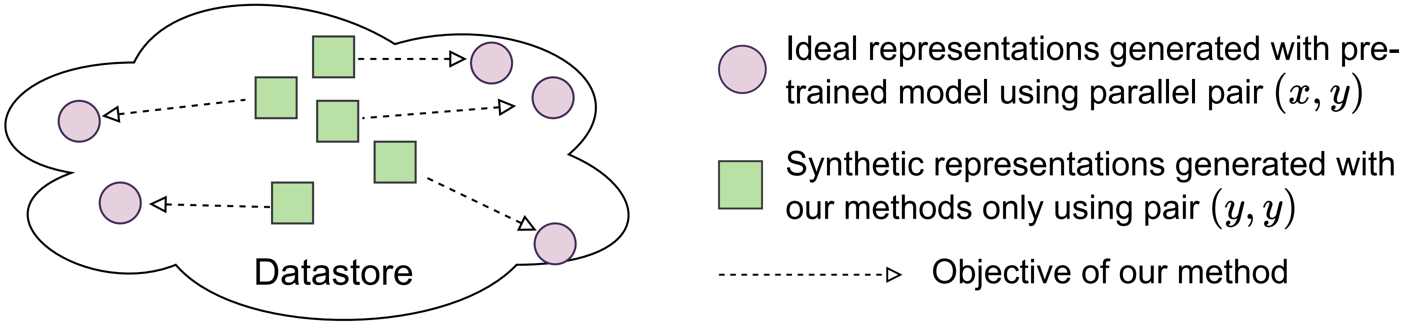
- Context representation of monolingual data and bilingual data are not in the same semantic space.

(Constructing pseudo bilingual data by back-translation is a trivial solution but requires an additional reverse translation model.)

Non-Parametric Unsupervised Domain Adaptation for Neural Machine Translation. Zheng et al. EMNLP'2021.

Setting 4: Unsupervised Domain Adaptation in MT

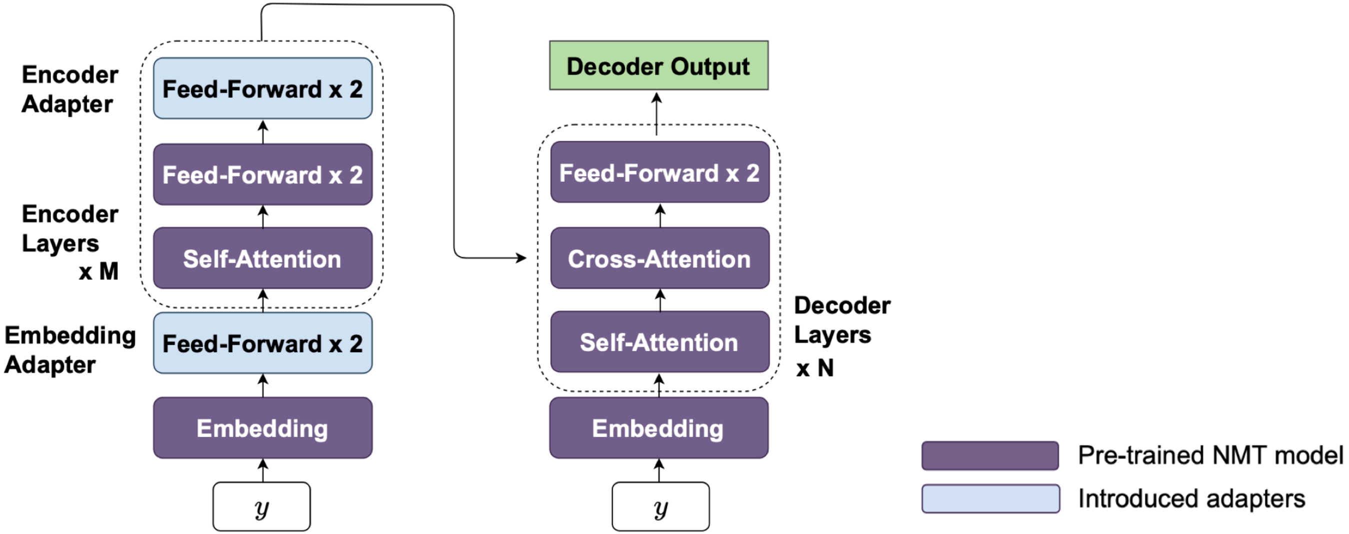
- **Obtain context representation of (y,y) with an auto-encoder and align target-side representation of (x,y) and (y,y)**



Setting 4: Unsupervised Domain Adaptation in MT

- **Train an adapter to learn and align representations**

$$\theta^* = \min_{\theta} \sum_{(x,\, y)\in(\mathcal{X},\mathcal{Y})} \sum_{t} ||h'_{(y;\; y_{<t})} - h_{(x;\; y_{<t})}||^2,$$



Non-Parametric Unsupervised Domain Adaptation for Neural Machine Translation. Zheng et al. EMNLP'2021.

Setting 4: Unsupervised Domain Adaptation in MT

- improved performance with only monolingual data
- achieve competitive results against BT-KNN, but without extra translation of monolingual data

Model	IT	Medical	Law	Koran	Avg
Basic NMT	38.35	39.99	45.48	16.26	35.02
Empty-kNN	38.06	40.01	45.62	16.44	35.03
Copy-kNN	38.96	40.86	46.00	17.06	35.72
BT-kNN	41.35	47.02	52.91	19.58	40.23
UDA-kNN	41.57	46.64	52.02	19.42	39.91
Parallel-kNN	45.96	54.16	61.31	20.30	45.43

Non-Parametric Unsupervised Domain Adaptation for Neural Machine Translation. Zheng et al. EMNLP'2021.

Effectiveness

- **kNN-MT is less stable because:**
  - different level of noises retrieved for different tokens
  - different domain requires different usage of the datastore
  - the datastore is changing (e.g. built gradually)
- **The datastore need to be built without parallel data.**

- **Different scenarios bring interesting challenges.**

Adaptive Nearest Neighbor Machine Translation. Zheng et al. ACL'2021.
Learning Kernel-Smoothed Machine Translation with Retrieved Examples. Jiang et al. EMNLP'2021.
Non-Parametric Online Learning from Human Feedback for Neural Machine Translation. Wang et al. AAAI'2022.
Non-Parametric Unsupervised Domain Adaptation for Neural Machine Translation. Zheng et al. EMNLP'2021.

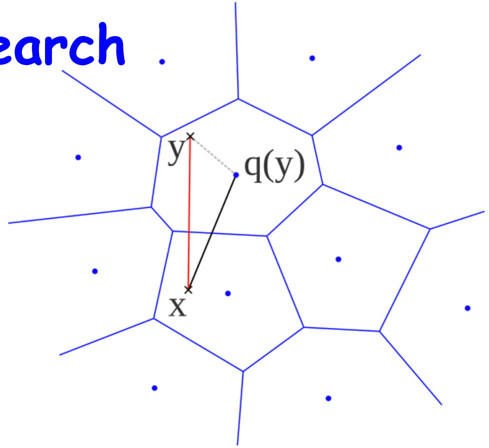
Part 3: Dive into kNN-MT: Efficiency

kNN-MT requires extra computations

- **extra computation cost in kNN-MT comes from:**
  - neural representations are high-dimensional vectors, so computing similarities are expensive
  - symbolic tokens are collected for all the occurrences of the training data, so the datastore is huge (billions of entries)
  - the query is performed at each decoding step

Can We Accelerate Inference Speed of kNN-MT?

- **FAISS: a Library for nearest neighbor search**
  - Product Quantizer (PQ)
  - Inverted File (IVF)
  - https://github.com/facebookresearch/faiss



- **However, kNN-MT's decoding speed is still much slower than the base MT system.**
  - x100, batch = 1

Nearest Neighbor Machine Translation. Khandelwal et al. ICLR' 2021
Product quantization for nearest neighbor search. Jégou et al., PAMI'2011
Searching in one billion vectors: re-rank with source coding. Tavenard et al., ICASSP'2011
Billion-scale similarity search with GPUs. Johnson et al., ArXiv'2017

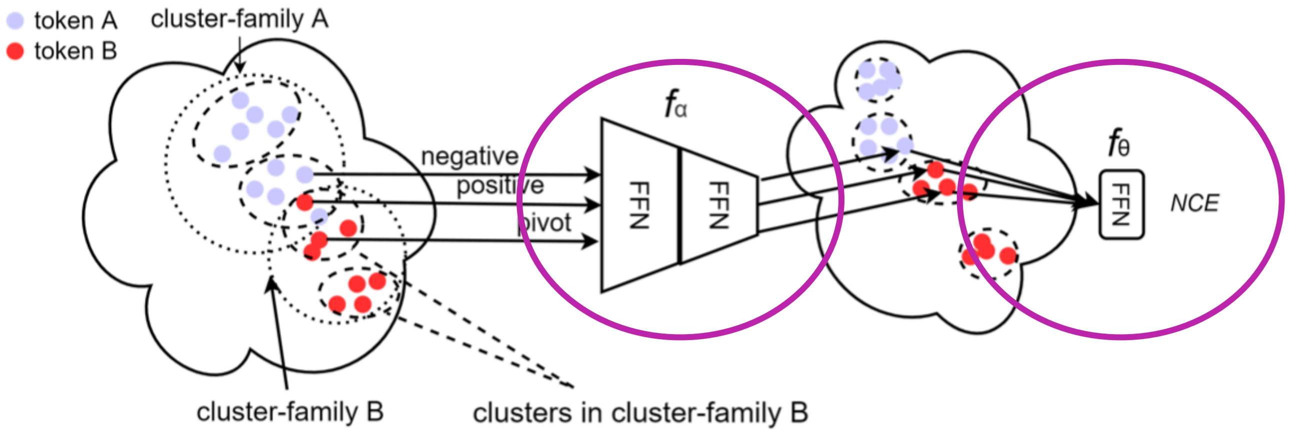
Solution 1: Compress Context Representation

- **Reduce the dimension of context representation**
  - Principal Component Analysis (PCA) (Martins et al.)
  - Singular Value Decomposition (SVD) (Wang et al.)
  - Trained compression with cluster-based (Wang et al.)

Efficient Machine Translation Domain Adaptation. Martins et al. WSMNLP'2022.
Efficient Cluster-Based k-Nearest-Neighbor Machine Translation. Wang et al. ACL'2022.

Solution 1: Compress Context Representation

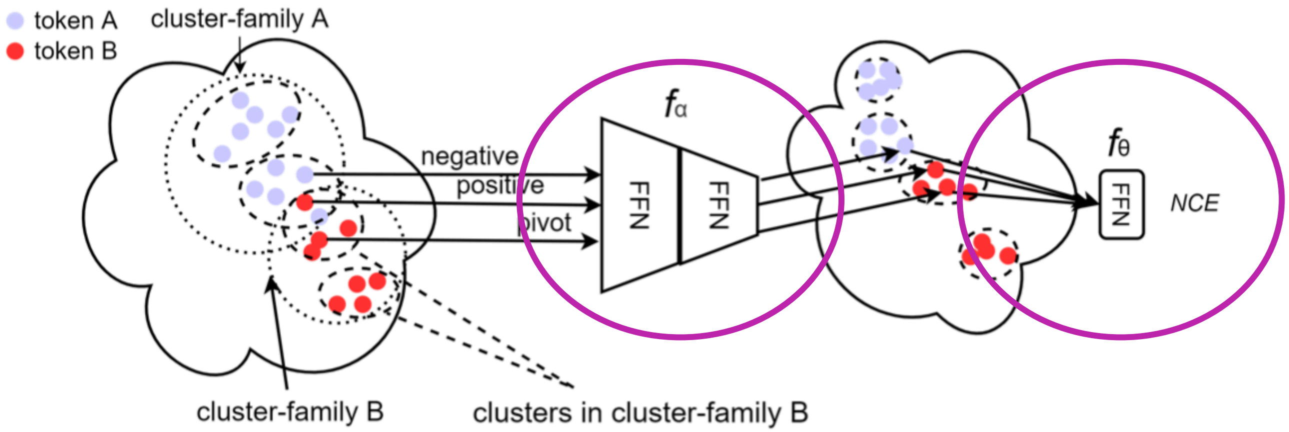
- **Cluster-based feature compression**
  - Conduct target-side clustering for the representations with the same target token
  - Compress feature with a learnable compact network (f_α+f_θ)



Efficient Cluster-Based k-Nearest-Neighbor Machine Translation. Wang et al. ACL'2022.

Solution 1: Compress Context Representation

- **Compact network**
  - f_α: project context representation into low-dimension space
  - f_θ: transfer the compressed representations into classification logit (discarded during inference)



Solution 1: Compress Context Representation

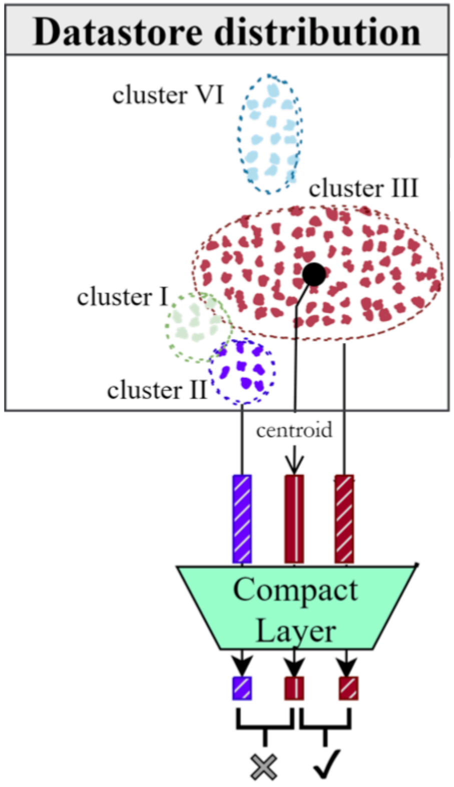
- **Learning object of compact network**
  - Triplet Noise-Contrastive Estimation (NCE)

$$\min -\log(\sigma(f_\theta([f_\alpha(v_+); f_\alpha(v_*)]))) \\ -\log(1-\sigma(f_\theta([f_\alpha(v_-); f_\alpha(v_*)])))$$

  - Triplet Distance Ranking (DR)

$$\min \|f_\alpha(v_+) - f_\alpha(v_*)\|_2 + 1/\|f_\alpha(v_-) - f_\alpha(v_*)\|_2$$

  - Word Prediction Loss (WP)



Solution 1: Compress Context Representation

- **Empirical results**
  - 1024-to-64 PCA/SVD is difficult to maintain translation performance
  - The best approach is to use compact network trained with triplet distance ranking loss
  - Compressing context representation can significantly improve inference speed (1.5x faster than adaptive KNN-MT)

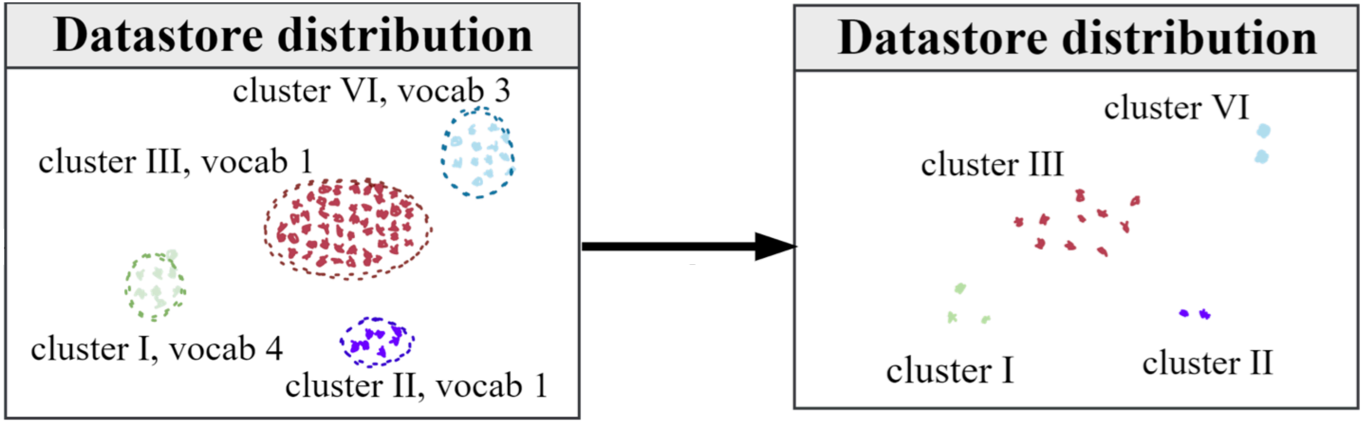
Model	BLEU
NMT	38.35
adaptive kNN-MT	47.20
+feature-wise PCA	46.84
+weight-wise SVD	45.96
[DY] CKMT+DR	37.10
[DY] CKMT+WP	46.41
[DY] CKMT+NCE	46.58
[DY] CKMT+NCE+DR	37.33
[DY] CKMT+NCE+WP	46.42
[DY] CKMT+NCE+CL	47.48
[ST] CKMT+NCE+CL	**47.94**
[ST] CKMT+NCE+CL+DR	47.64
[ST] CKMT+NCE+CL+WP	46.88

Model		BLEU	Sentences/s	Tokens/s
adaptive kNN-MT		31.36	58	660
k=16	CKMT*	31.64	74	849
	PCKMT*	31.58	85	963
k=8	CKMT*	31.43	78	890
	PCKMT*	**31.72**	**91**	**1024**
k=4	CKMT*	31.28	79	899
	PCKMT*	31.23	85	968

Efficient Cluster-Based k-Nearest-Neighbor Machine Translation. Wang et al. ACL'2022.

Solution 2: Prune Datastore Entries

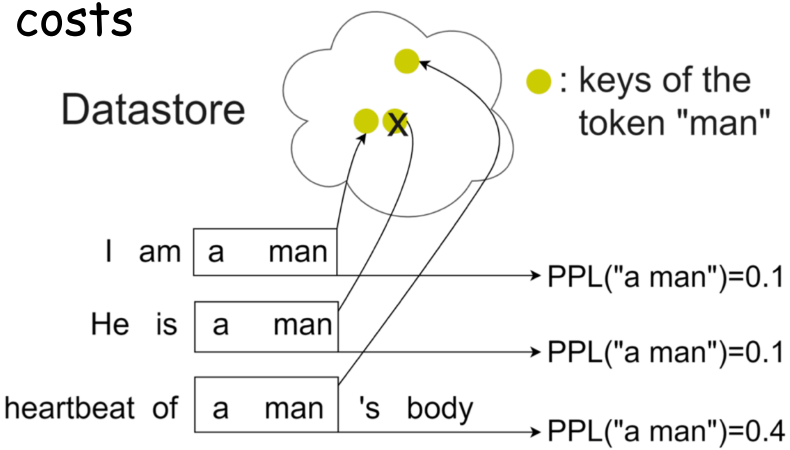
- **Reduce the number of datastore entries**
  - Merge datastore entries that share the same value while their keys are close to each other (Martins et al.)
  - Cluster-based datastore pruning (Wang et al.)



Efficient Machine Translation Domain Adaptation. Martins et al. WSMNLP'2022.
Efficient Cluster-Based k-Nearest-Neighbor Machine Translation. Wang et al. ACL'2022.

Solution 2: Prune Datastore Entries

- **Cluster-based datastore pruning**
  - Assumption: a key-value pair is redundant if there are other key-value pairs (with the same value) that have similar translation costs



  - Translation cost: the minimal PPL among all consecutive sub-sequences ending with that last token

Efficient Cluster-Based k-Nearest-Neighbor Machine Translation. Wang et al. ACL'2022.

Solution 2: Prune Datastore Entries

- Greedy merging datastore entries (Martins et al.) prunes 40% datastore entries with the cost of 1.4 BLEU in average
- Cluster-based method (Wang et al.) prunes 10% datastore entries with the cost of 0.9 BLEU in average

	Medical	Law	IT	Koran	Average
kNN-MT	54.47	61.23	45.96	21.02	45.67
$k = 1$	53.60	60.23	45.03	20.81	44.92
$k = 2$	52.95	59.40	44.76	20.12	44.31
$k = 5$	51.63	57.55	44.07	19.29	43.14

the number of neighbors used for greed merging

Model	Domain				Avg.
	IT	Koran	Law	Medical	
CKMT*	47.94	19.92	62.98	56.92	46.94
CKMT*+SP	43.01	19.50	59.40	52.16	43.52
CKMT*+LTP	46.78	19.28	**61.96**	55.21	45.81
CKMT*+HTP	45.95	**20.10**	59.51	55.14	45.18
CKMT*+RP	46.38	19.99	**61.96**	**55.45**	45.85
CKMT*+Ours	**47.06**	20.01	61.72	55.33	**46.03**

Efficient Cluster-Based k-Nearest-Neighbor Machine Translation. Wang et al. ACL'2022.
Efficient Machine Translation Domain Adaptation. Martins et al. WSMNLP'2022.

Solution 2: Prune Datastore Entries

- Pruning datastore brings speed improvement
- still x2 slower than NMT only

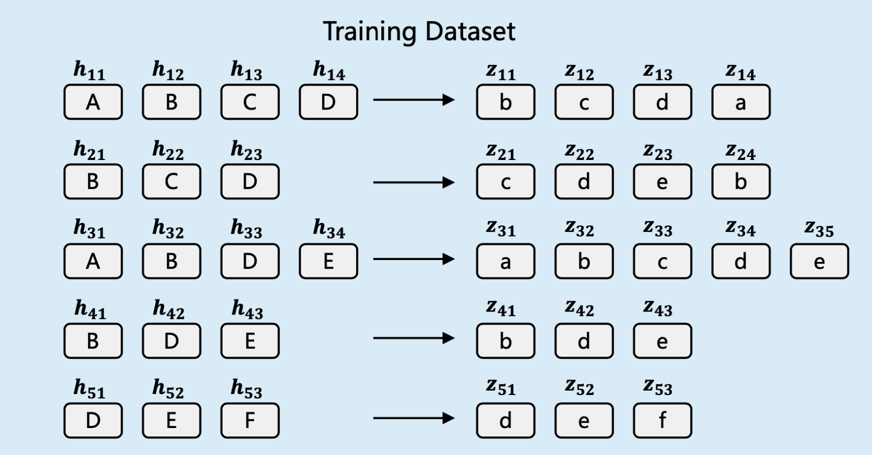
Model		BLEU	Sentences/s	Tokens/s	Datastore size	Pruning rate
adaptive *k*NN-MT		31.36	58	660	154M	0%
k=16	CKMT*	31.64	74	849	154M	0%
	PCKMT*	31.58	85	963	123M	20%
k=8	CKMT*	31.43	78	890	154M	0%
	PCKMT*	**31.72**	**91**	**1024**	108M	30%
k=4	CKMT*	31.28	79	899	154M	0%
	PCKMT*	31.23	85	968	138M	10%

Efficient Cluster-Based k-Nearest-Neighbor Machine Translation. Wang et al. ACL'2022.

Solution 3: Narrow Down Search Space

- **Narrow down search space with prior hypothesis**
  - Source sentence may help narrow down search space (Meng et al. and Wang et al.)
- ***A toy dataset for illustration***
  - Training set

$$(x^{(1)}, y^{(1)}) = (\{A, B, C, D\},\ \{b, c, d, a\})$$
$$(x^{(2)}, y^{(2)}) = (\{B, C, D\},\ \{c, d, e, b\})$$
$$(x^{(3)}, y^{(3)}) = (\{A, B, D, E\},\ \{a, b, c, d, e\})$$
$$(x^{(4)}, y^{(4)}) = (\{B, D, E\},\ \{b, d, e\})$$
$$(x^{(5)}, y^{(5)}) = (\{D, E, F\},\ \{d, e, f\})$$



  - Test example: $\{B, C, E\}$

Fast Nearest Neighbor Machine Translation. Meng et al. ACL'2022.
Faster Nearest Neighbor Machine Translation. Wang et al. arXiv'2022.

Solution 3: Narrow Down Search Space

- Narrowing down search spaces causes translation performance decline on target domains (especially on Law)

Model	Medical	Law	IT	Koran	Subtitles	Avg.
Aharoni and Goldberg [1]	54.8	58.8	43.5	21.8	27.4	41.3
base MT	39.9	45.7	38.0	16.3	29.2	33.8
+*k*NN-MT	54.4(+14.5)	61.8(+16.1)	45.8(+7.8)	19.4(+3.1)	31.7(+2.5)	42.6(+8.8)
+fast *k*NN-MT	53.6(+13.7)	56.0(+10.3)	45.5(+7.5)	21.2(+4.9)	30.5(+1.3)	41.4(+7.6)

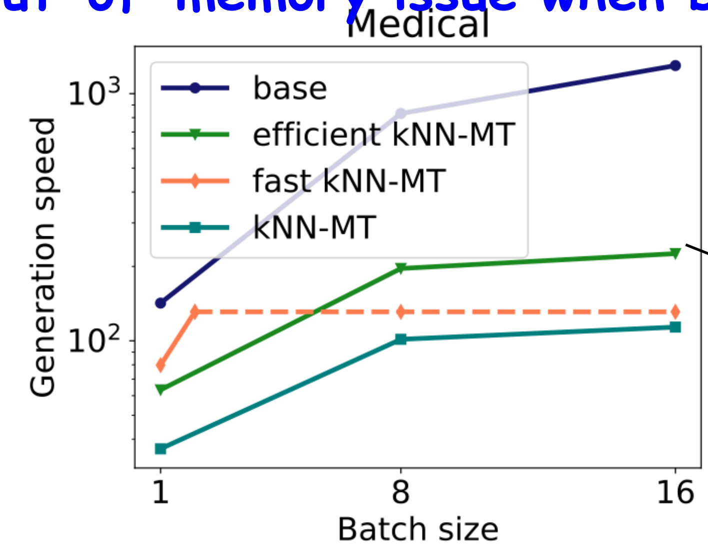
- Distance … her decline

Model	**Medical**	**IT**	**Koran**	**Subtitles**
Aharoni and Goldberg (2020)	54.8	43.5	21.8	27.4
Base MT	39.9	38.0	16.3	29.2
+ *k*NN-MT	54.4(+14.5)	45.8(+7.8)	19.4(+3.1)	31.7(+2.5)
+ Fast *k*NN-MT	53.6(+13.7)	45.5(+7.5)	21.2(+4.9)	30.5(+1.3)
+ Faster *k*NN-MT	52.7(+12.8)	44.9(+6.9)	20.4(+4.1)	30.2(+1.0)

Fast Nearest Neighbor Machine Translation. Meng et al. ACL'2022.
Faster Nearest Neighbor Machine Translation. Wang et al. arXiv'2022.

Solution 3: Narrow Down Search Space

- Narrowing down search space can improve inference speed
- But proposed approaches require large GPU memory and has out-of-memory issue when batch size is large than 2



reducing the dimension, merging the entries, (cached retrieval)

Fast Nearest Neighbor Machine Translation. Meng et al. ACL'2022.
Efficient Machine Translation Domain Adaptation. Martins et al. WSMNLP'2022.

Solution 4: Reduce Retrieval Frequency

- **Avoid querying datastore at each decoding step**
  - Adaptive retrieval with a learned neural network (Martins et al.)
  - Cache previous retrieval distributions as candidates (Martins et al.)
  - Use empirical schedule for retrieval (Martins et al.)

Efficient Machine Translation Domain Adaptation. Martins et al. WSMNLP'2022.
Chunk-based Nearest Neighbor Machine Translation. Martins et al. arXiv'2022.

Solution 4: Reduce Retrieval Frequency

- **Adaptive retrieval with a learned neural network**
  - Use a simple MLP to predict interpolation weight λ
  - Only performs retrieval when λ is greater than a threshold

- **Cache previous retrieval distributions as candidates**
  - If current decoder's representation is close to the keys on cache, the model retrieve the KNN distribution from the cache

$$\mathcal{C}=\{(\boldsymbol{f}(\boldsymbol{x},\boldsymbol{y}_{<t}),p_{k\mathrm{NN}}(y_t|\boldsymbol{y}_{<t},\boldsymbol{x}))\forall y_t\in\boldsymbol{y}\,|\,\boldsymbol{y}\in\mathcal{B}\}$$

  - Otherwise, the model search the datastore

Solution 4: Reduce Retrieval Frequency

- **Using a datastore with consecutive tokens (chunks) as values**
  - Retrieve chunks of tokens at retrieval steps
  - Reuse previously retrieved results at non-retrieval steps

- **Retrieval steps schedule**
  - Empirically, it is beneficial to perform retrieval steps more frequently at the beginning of the sentence
  - Interval between the current retrieval step and the next one

$$i(t) = \min\left(i_{\max},\ i_{\min} \times 2^{\frac{\frac{1}{2} i_{\max}\, t}{|\boldsymbol{x}|}}\right)$$

Chunk-based Nearest Neighbor Machine Translation. Martins et al. arXiv'2022.

Solution 4: Reduce Retrieval Frequency

- **Reducing retrieval frequency causes translation performance decline on target domains**

cache-based

	BLEU				
	Medical	Law	IT	Koran	Average
Baselines					
Base MT	40.01	45.64	37.91	16.35	34.98
kNN-MT	54.47	61.23	45.96	21.02	45.67
Fast kNN-MT	52.90	55.71	44.73	21.29	43.66
Efficient kNN-MT					
cache	53.30	59.12	45.39	20.67	44.62
PCA + cache	53.58	58.57	46.29	20.67	44.78
PCA + pruning	53.23	60.38	45.16	20.52	44.82
PCA + cache + pruning	51.90	57.82	44.44	20.11	43.57

MLP-based

	Medical	Law	IT	Koran	Average
kNN-MT	54.47	61.23	45.96	21.02	45.67
$\alpha = 0.25$	45.52	49.91	37.97	16.36	37.44
$\alpha = 0.5$	52.84	59.36	38.58	18.08	42.22
$\alpha = 0.75$	53.90	60.87	43.05	19.91	44.43

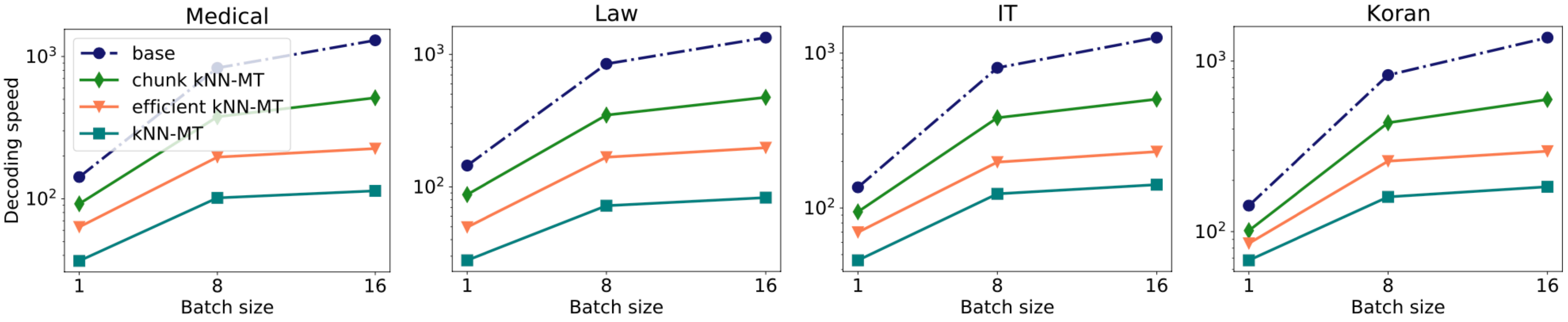
chunk-based

	BLEU				
	Medical	Law	IT	Koran	Average
Parametric models					
Base MT	40.01	45.64	37.91	16.35	34.98
Fine-tuned	50.47	56.56	43.82	21.54	43.10
Semi-parametric models					
kNN-MT	54.47	61.23	45.96	21.02	45.67
Efficient kNN-MT	51.90	57.82	44.44	20.11	43.57
Chunk-based kNN-MT	53.16	59.65	44.18	19.33	44.08

Efficient Machine Translation Domain Adaptation. Martins et al. WSMNLP'2022.

Solution 4: Reduce Retrieval Frequency

- Reducing retrieval frequency can improve inference speed
- The fastest approach is chunk-based KNN-MT (4X faster than vanilla KNN-MT), but is still slower than Base MT when batch size is large.



Chunk-based Nearest Neighbor Machine Translation. Martins et al. arXiv'2022.

Efficiency

- **Accelerating the inference speed of kNN-MT?**
  - improve the inference speed of kNN-MT in different ways, but trade off translation performance
  - still a large speed gap between optimized kNN-MT and base MT when the batch size is large (a more practical setting)

Efficient Cluster-Based k-Nearest-Neighbor Machine Translation. Wang et al. ACL'2022.
Efficient Machine Translation Domain Adaptation. Martins et al. WSMNLP'2022.
Fast Nearest Neighbor Machine Translation. Meng et al. ACL'2022.
Faster Nearest Neighbor Machine Translation. Wang et al. arXiv'2022.
Chunk-based Nearest Neighbor Machine Translation. Martins et al. arXiv'2022.

Part 3: Dive into kNN-MT: Interpretability

Interpretability

- **Why is retrieval useful for neural model?**
  - Khandelwal et al. ICLR'2020
  - Khandelwal et al. ICLR'2021
  - Jiang et al. EMNLP'2021

- **Which entries of symbolic datastore are important?**
  - Anonymous, Openreview'2022
  - Wang et al., ACL'2022

Generalization through Memorization: Nearest Neighbor Language Models. Khandelwal et al. ICLR'2020
Nearest Neighbor Machine Translation. Khandelwal et al. ICLR'2021
Learning Kernel-Smoothed Machine Translation with Retrieved Examples. Jiang et al. EMNLP'2021
When Is Retrieved Knowledge Helpful? Towards Explainable Memory for kNN-MT Domain Adaptation. Anonymous, Openreview'2022
Efficient Cluster-Based k-Nearest-Neighbor Machine Translation. Wang et al. ACL'2022

Why Is Retrieval Useful for Neural Model?

- Similar context has similar distribution over the next word

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

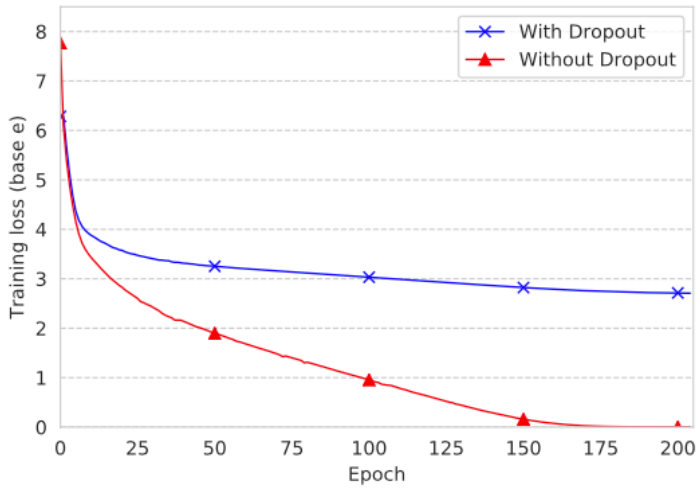
Training Set Translation Context (source and target)		Training Set Target	Context Probability
Dem charismatischen Ministerpräsidenten Recep Tayyip Erdoğan, der drei aufeinanderfolgende Wahlen für sich entscheiden konnte, ist es gelungen seine Autorität gegenüber dem Militär geltend zu machen.	*The charismatic 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

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.

Retrieval can predict target token correctly

Why Is Retrieval Useful for Neural Model?

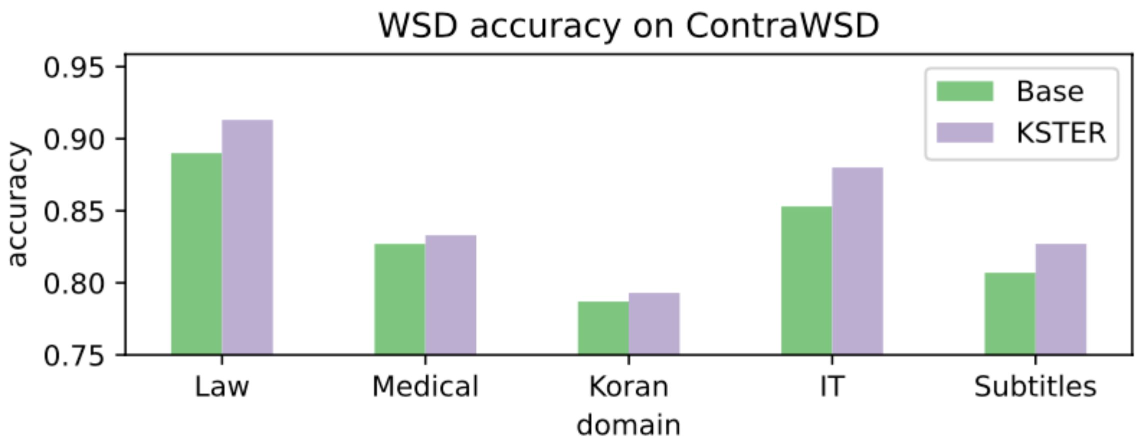
- **Implicit vs Explicit Memory**
  - Transformer is expressive enough to memorize all training examples (training loss drops to 0)
  - But retrieval-based KNN-LM memorized training data while improving generalization



Model	Perplexity on WIKITEXT-103
Base LM	17.96
Base LM + Implicit Memory	17.86
Base LM + Explicit Memory	16.06

Why Is Retrieval Useful for Neural Model?

- Retrieval improve the predictions of **morphologically complex** word types, e.g. verbs, adverbs and nouns
- Retrieved examples contains useful context information which helps **word sense disambiguation** (WSD)



Which Entries of Datastore are Important?

- The relationship between NMT model and symbolic datastore is unclear

- The datastore saves all target language token occurrences in the parallel corpus, which is usually large and possibly redundant

Local Correctness

- **Intuitively, retrieved knowledge is only needed when the pre-trained NMT model fails. (Anonymous et al.)**

- **A novel notion called "local correctness" (LAC), which consists of entry correctness and neighborhood correctness.**

When Is Retrieved Knowledge Helpful? Towards Explainable Memory for kNN-MT Domain Adaptation. Anonymous, Openreview'2022

Local Correctness

- **Entry Correctness**
  - Entry correctness describes whether the NMT model knows a specific datastore entry
  - It can be evaluated by comparing target token and prediction token

$$(h(\mathbf{x}, \mathbf{y}_{<t}), y_t) \text{ is } \begin{cases} \text{known}, & \text{if } \hat{y}_t = y_t \\ \text{unknown}, & \text{o.w.} \end{cases}$$

When Is Retrieved Knowledge Helpful? Towards Explainable Memory for kNN-MT Domain Adaptation. Anonymous, Openreview'2022

Local Correctness

- **Entry Correctness**
  - The datastore entries are thus divided into two categories: known entries and unknown entries.
  - 56%~73% of datastore entries are known to the NMT.

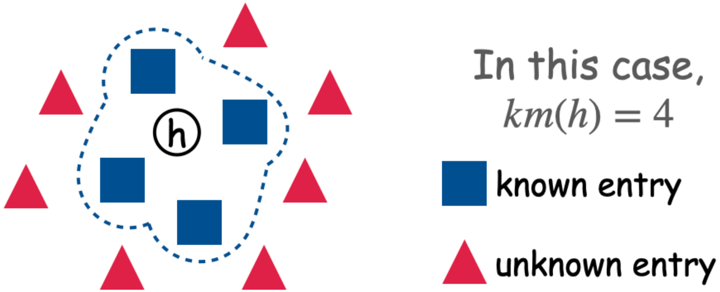
	OPUS-Medical	OPUS-Law	OPUS-IT	OPUS-Koran
known	5,070,607	14,803,149	2,514,757	294,094
unknown	1,844,966	4,287,906	1,093,974	230,677
$\|\mathcal{D}\|$	6,915,573	19,091,055	3,608,731	524,771
known ratio	73.32%	66.74%	69.69%	56.04%

When Is Retrieved Knowledge Helpful? Towards Explainable Memory for kNN-MT Domain Adaptation. Anonymous, Openreview'2022

Local Correctness

- **Neighborhood Correctness**
  - Neighborhood correctness evaluates the NMT model's prediction on a neighborhood in the representation space.
  - Knowledge margin is proposed as the metric.

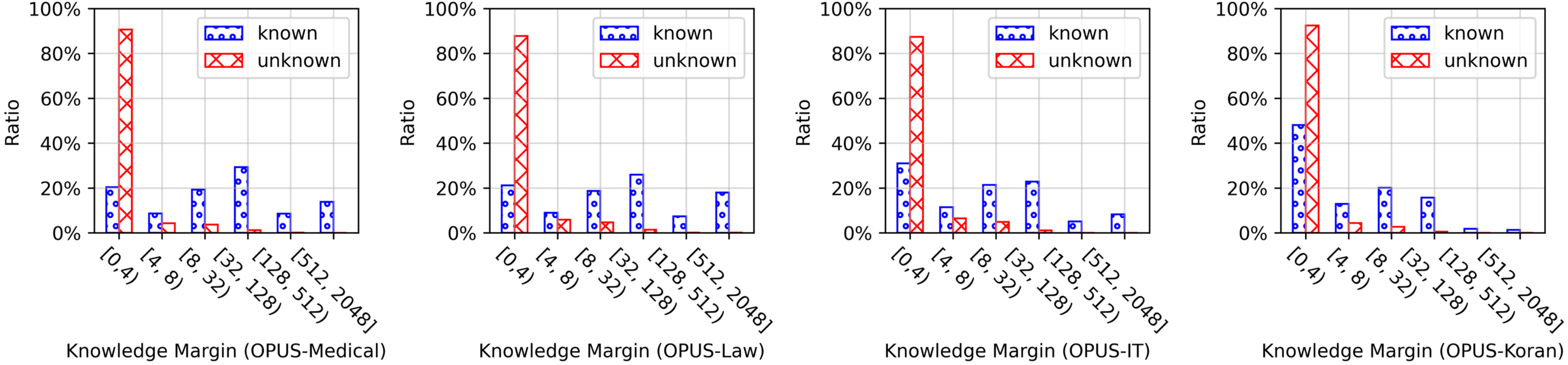
$$km(h) = \arg\max_{t} \forall (h^j, y^j) \in \mathcal{N}_t(h) \text{ is known}$$



When Is Retrieved Knowledge Helpful? Towards Explainable Memory for kNN-MT Domain Adaptation. Anonymous, Openreview'2022

Local Correctness

- **Neighborhood Correctness**
  - Most unknown entries has a very low knowledge margin.
  - The distribution for known entries is more diverse.



When Is Retrieved Knowledge Helpful? Towards Explainable Memory for kNN-MT Domain Adaptation. Anonymous, Openreview'2022

Local Correctness

- **Understand the role of different datastore entries.**

Helpful

- Unknown entries: contain knowledge that NMT model does not know

Helpful

- Known entries with small km: NMT model tends to fail when context are similar but different

Less Helpful

- Known entries with large km: NMT model generalizes well on these entries

Algorithm 1 Datastore Pruning by LAC

Input: datastore $\mathcal{D}$, the *knowledge margin* threshold k_p, the pruning ratio r
Output: pruned datastore $\mathcal{D}$

1: $candidates \leftarrow \emptyset$ ▷ step 1: collect
2: **for** each entry (h, y) in $\mathcal{D}$ **do**
3:     **if** (h, y) is known **and** $km(h) \geq k_p$ **then**:
4:         $candidates \leftarrow candidates \cup (h, y)$
5:     **end if**
6: **end for**
7: **repeat** ▷ step 2: drop
8:     randomly select entry (h, y) from $candidates$
9:     remove (h, y) from $\mathcal{D}$
10: **until** pruning ratio r is satisfied
11: **return** $\mathcal{D}$

When Is Retrieved Knowledge Helpful? Towards Explainable Memory for kNN-MT Domain Adaptation.
Anonymous, Openreview'2022

Empirical Results

- **Pruning with local correctness (PLAC) cuts off 25%-45% datastore entries while achieving comparable performance**
  - Previous pruning method (40% -1.4, 10% -0.9 BLEU)

	OPUS-Medical			OPUS-Law			OPUS-IT			OPUS-Koran		
	Ratio	BLEU↑	COMET↑	Ratio	BLEU↑	COMET↑	Ratio	BLEU↑	COMET↑	Ratio	BLEU↑	COMET↑
Base	-	39.73	0.4665	-	45.68	0.5761	-	37.94	0.3862	-	16.37	-0.0097
Finetune	-	58.09	0.5725	-	62.67	0.6849	-	49.08	0.6343	-	22.40	0.0551
Adaptive *k*NN	0%	57.98	0.5801	0%	63.53	0.7033	0%	48.39	0.5694	0%	20.67	0.0364
Random	45%	54.08*	0.5677*	45%	58.69*	0.6690*	40%	45.54*	0.5314*	25%	20.36	0.0434
Cluster	45%	53.31*	0.5689*	45%	58.68*	0.6779*	40%	45.80*	0.5788	25%	20.04*	0.0410*
Known	45%	56.44*	0.5691*	45%	61.61*	0.6885*	40%	45.93*	0.5563*	25%	20.35	0.0338
All Known	73%	42.73*	0.4926*	66%	51.90*	0.6200*	69%	40.93*	0.4604*	56%	17.76*	0.0008*
PLAC (ours)	45%	57.66	0.5773	45%	63.22	0.6953*	40%	48.22	0.5560	25%	20.96	0.0442

When Is Retrieved Knowledge Helpful? Towards Explainable Memory for kNN-MT Domain Adaptation. Anonymous, Openreview'2022

Interpretability

- **Retrieval is useful for neural model**
  - Memorize various patterns explicitly
  - Improve generalization ability of the MT system

- **Which part of symbolic datastore is redundant in the position of NMT model?**
  - Local correctness is good angle to interpret this issue
  - Known entries with large knowledge margin are less helpful

When Is Retrieved Knowledge Helpful? Towards Explainable Memory for kNN-MT Domain Adaptation. Anonymous, Openreview'2022

Part 4: Applications

kNN-box Toolkit

- **kNN-box is an open-source toolkit to build kNN-MT models**
  - easy-to-use: a few lines of code to deploy a kNN-MT model
  - research-oriented: provide implementations of various papers
  - extensible: easy to develop new kNN-MT models with our toolkit

https://github.com/NJUNLP/knn-box

kNN-box Toolkit

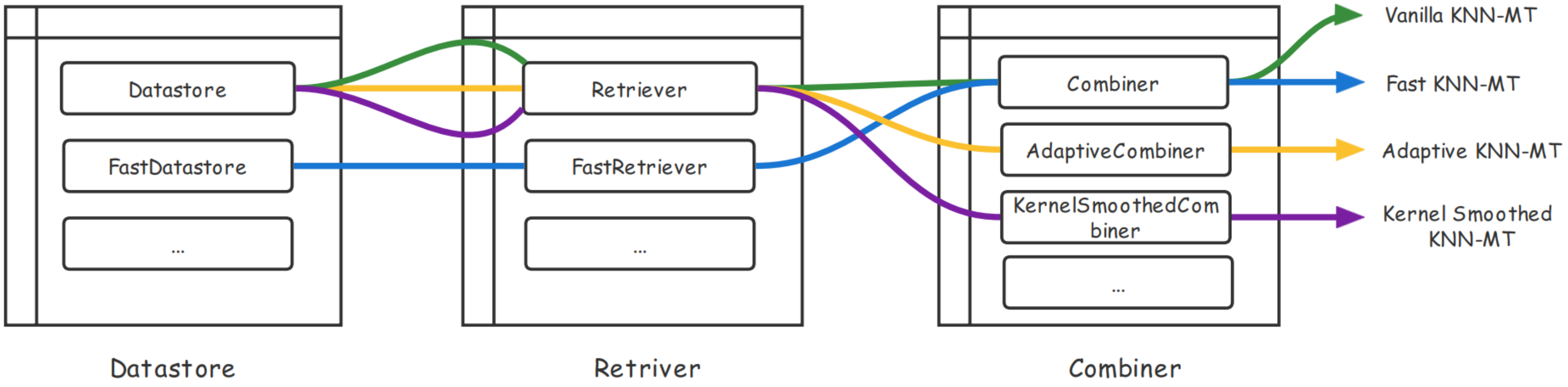
- **We unify different kNN-MT variants into a single framework, albeit they manipulate datastore in different ways.**

Datastore	save translation knowledge in key-values pairs
Retriever	retrieve translation knowledge from the datastore
Combiner	make final prediction based on retrieval results and NMT model

https://github.com/NJUNLP/knn-box

Build kNN models like Playing LEGO

- Users can easily develop different kNN-MT models by customizing three modules
- We also provide example implementations of various popular kNN-MT models and push-button scripts to run them



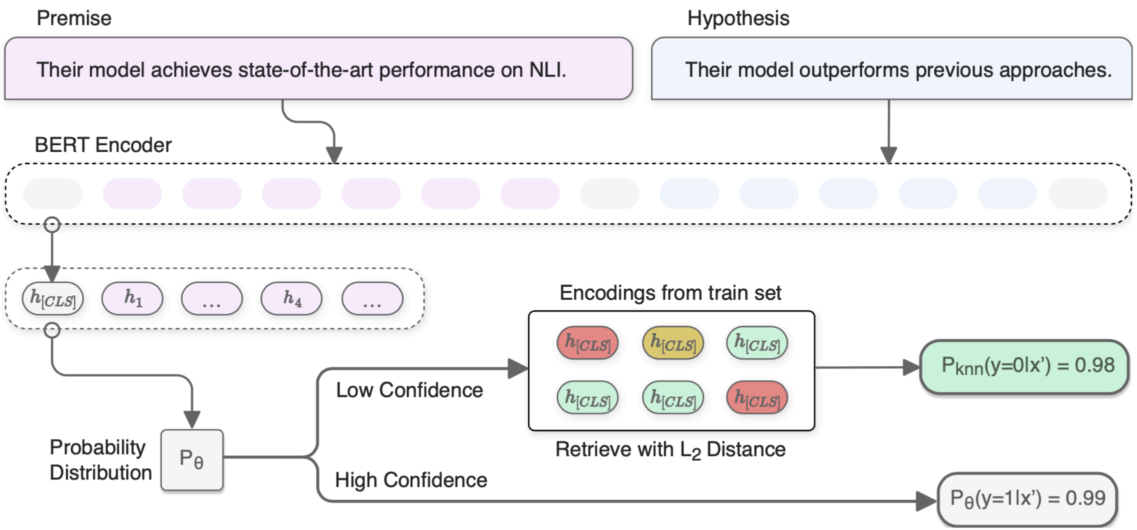
https://github.com/NJUNLP/knn-box

kNN for Other Tasks

- It is easy to fill other task-specific knowledge into the datastore
- The idea of kNN-LM/MT is applicable to other tasks
  - Natural Language Inference (NLI)
  - Question Answering (QA)
  - Visual Classification
  - Image Caption
  - Multi-Label Text Classification
  - Named Entity Recognition (NER)
  - ...

Natural Language Inference (NLI)

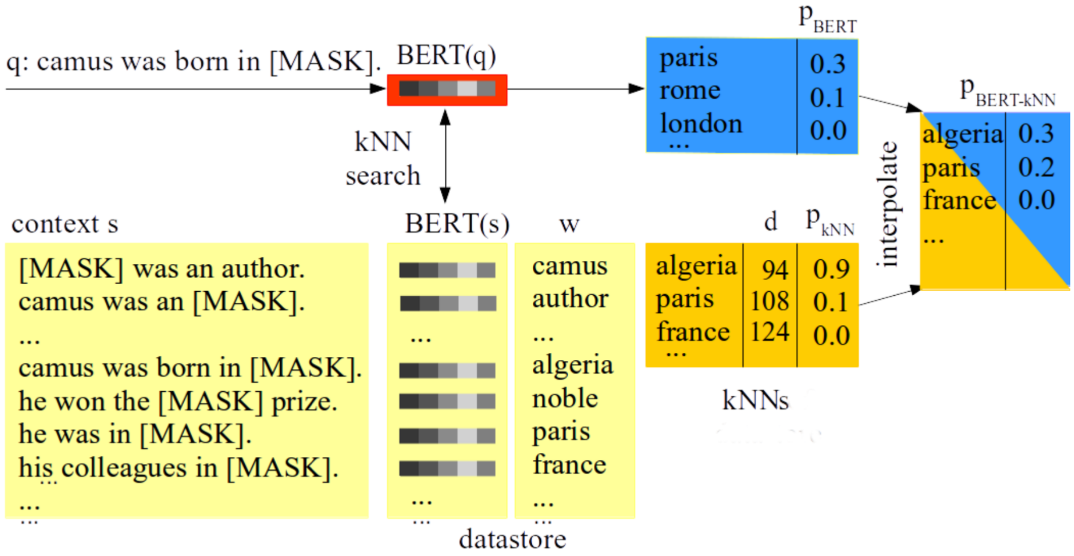
- **Task-specific Knowledge in Datastore**
  - Key: the representation of the input sentence pair
  - Value: the relationship between the premise and the hypothesis



Explaining and Improving Model Behavior with k Nearest Neighbor Representations. Rajani et al. ArXiv'2020

Question Answering (QA)

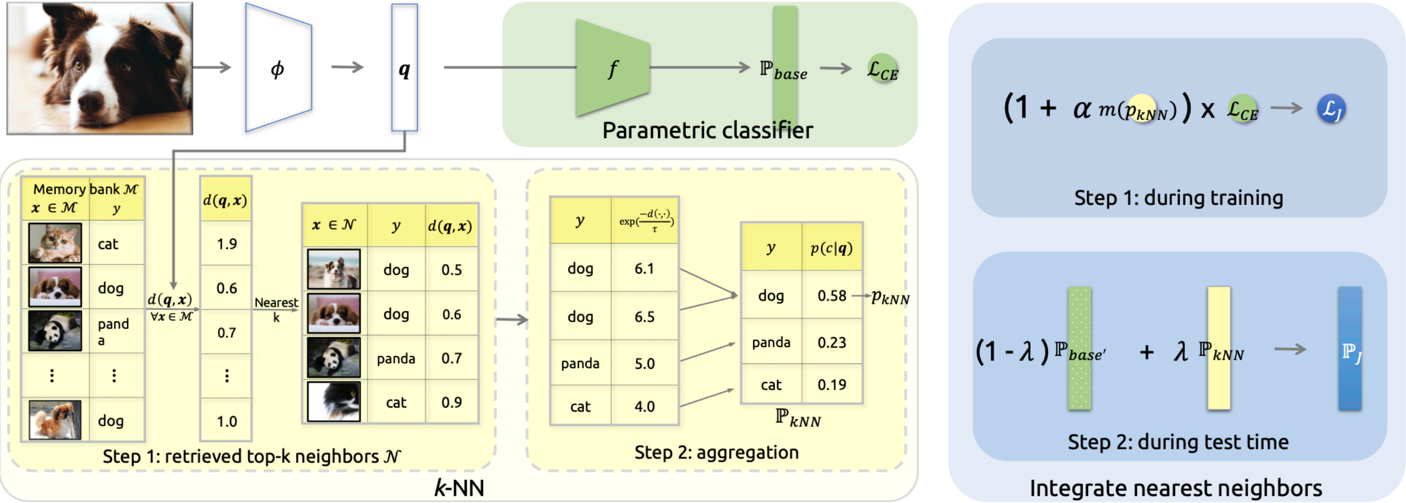
- **Task-specific Knowledge in Datastore**
  - key: the representation of the cloze question
  - value: the answer for the cloze question



BERT-kNN: Adding a kNN Search Component to Pretrained Language Models for Better QA. Kassner and Schuetze. EMNLP'2020

Visual Classification

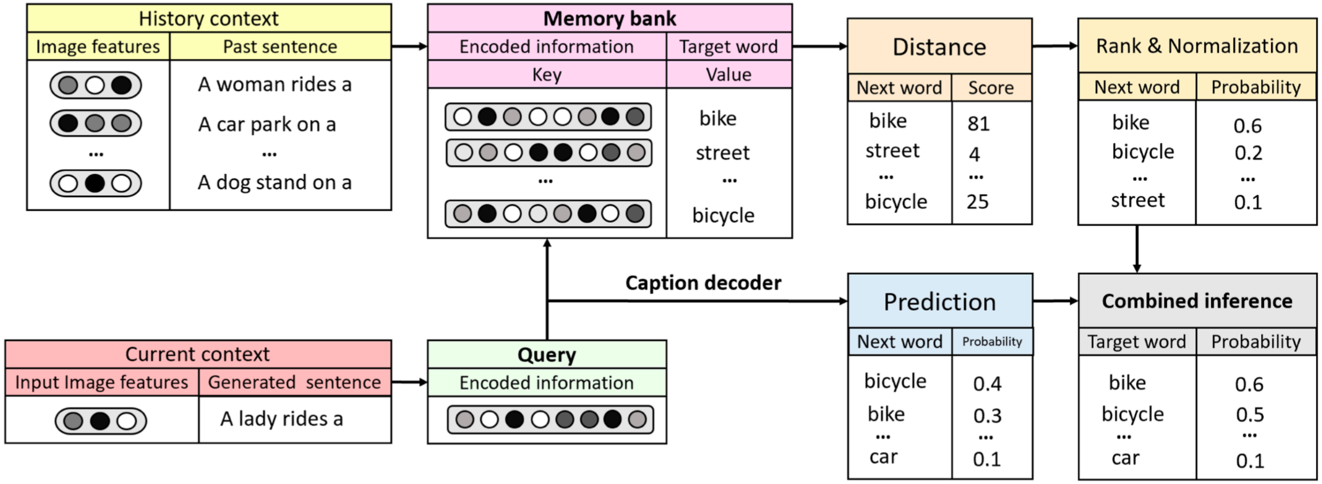
- **Task-specific Knowledge in Datastore**
  - key: the representation of the input image
  - value: target label of the input image



Rethinking Nearest Neighbors for Visual Classification. Jia et al. ArXiv;2021

Image Caption

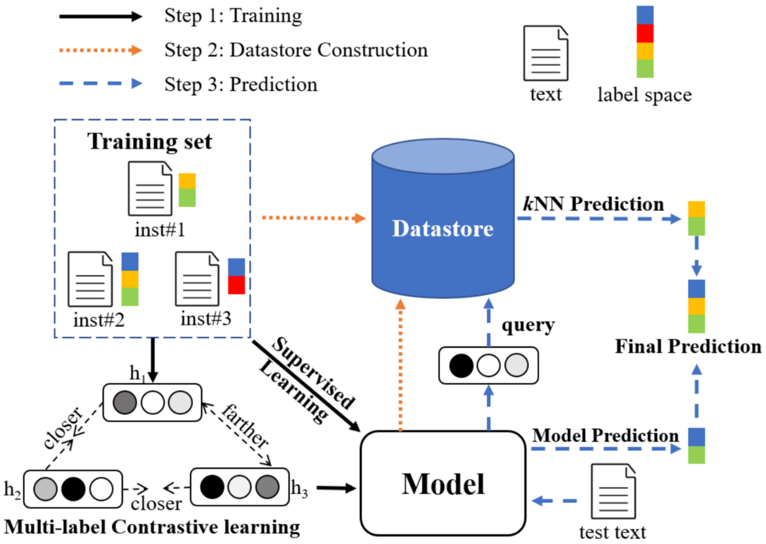
- **Task-specific Knowledge in Datastore**
  - key: the representation of the cross-modal context
  - value: ground truth word under the given context



Memory-Augmented Image Captioning. Fei. AAAI'2021

Multi-Label Text Classification

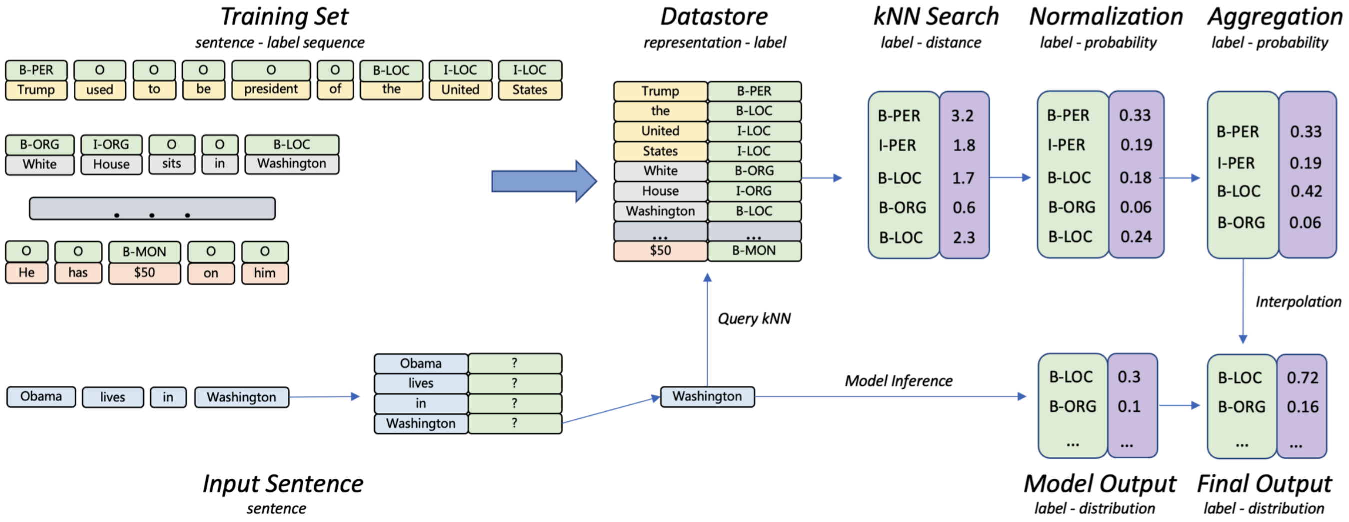
- **Task-specific Knowledge in Datastore**
  - key: the representation of the input text
  - value: target multi-label for the input text



Contrastive Learning-Enhanced Nearest Neighbor Mechanism for Multi-Label Text Classification. Su et al. ACL'2022

Named Entity Recognition (NER)

- **Task-specific Knowledge in Datastore**
  - key: the representation of a word from the given sentence
  - value: the name entity of the word



kNN-NER: Named Entity Recognition with Nearest Neighbor Search. Wang et al. ArXiv'2022

Conclusion and Future work

- **Symbolic system is a good compensation for neural system**

- **kNN-MT: a novel neuro-symbolic MT framework, which can also be transferred to other NLP tasks**

- **Recent advances has made kNN-MT**
  - Effective in more settings
  - Has faster inference speed
  - More explainable than a black box

Conclusion and Future work

- **Interesting problems to be explored**
  - Can we build a symbolic system that is tiny but effective?
  - Can we use neural vectors as values to construct the datastore?
  - Can we explain the inner-working of the neural system with the help of the symbolic system?

	Symbolic Value	Neural Value
Symbolic Key	exact matching	?
Neural Key	neural retrieval	?

Reference

- **(only those not shown in page)**
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. NIPS2017
- Makoto Nagao. A framework of a mechanical translation between Japanese and English by analogy principle. In A. Elithorn and R. Banerji. Artificial and Human Intelligence. Elsevier Science Publishers. 1984
- Jiatao Gu, Yong Wang, Kyunghyun Cho, and Victor O. K. Li. 2018. Search engine guided neural machine translation. AAAI2018
- Jingyi Zhang, Masao Utiyama, Eiichro Sumita, Graham Neubig, and Satoshi Nakamura. 2018. Guiding neural machine translation with retrieved translation pieces. NAACL2018

- Ankur Bapna and Orhan Firat. 2019. Non-parametric adaptation for neural machine translation. NAACL2019
- Urvashi Khandelwal, Angela Fan, Dan Jurafsky, Luke Zettlemoyer, and Mike Lewis. 2020. Nearest neighbor machine translation. ICLR2020
- Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. ACL2016.

Thanks for Watching !