小 型 微 型 计 算 机 系 统
Journal of Chinese Computer Systems
DOI:10.20009/j.cnki.21-1106/TP.2025-0077
2026 年 4 月 第 4 期
Vol.47 No.4 2026

置信度优化的 k 近邻机器翻译方法

周茂春,朱俊国

[1](昆明理工大学 信息工程与自动化学院,昆明 650500)
[2](昆明理工大学 云南省人工智能重点实验室,昆明 650500)
E-mail:20222204178@stu.kust.edu.cn

摘 要: k 近邻机器翻译(kNN-MT)通过检索外部数据存储中的翻译知识,显著地提升神经机器翻译(NMT)模型预测的准确性.然而,使用固定的融合比例聚合 NMT 模型预测和 kNN 检索的概率分布容易使模型受到检索结果中噪声的干扰,且 kNN 检索的高延迟特性限制了其实际应用.为此,本文提出了一种基于置信度优化的 k 近邻机器翻译方法.具体地,引入置信度估计模块动态评估 NMT 预测的概率分布与 kNN 检索分布的可靠性,以自适应的方式计算概率融合比例以提升翻译的准确性.同时,基于模型的置信度修剪数据存储中冗余的知识实例,提升模型的解码效率.在两组特定语言对翻译任务的实验结果表明,该方法在翻译质量和解码效率上均显著优于标准的 kNN-MT 模型.

关 键 词: k 近邻机器翻译;数据存储;检索;置信度
中图分类号: TP391 **文献标识码:** A **文 章 编 号:** 1000-1220(2026)04-0902-07

Research on k-nearest Neighbor Machine Translation Method with Confidence Optimization

ZHOU Maochun, ZHU Junguo
[1](Faculty of Information Engineering and Automation, Kunming University of Science and Technology, Kunming 650500, China)
[2](Yunnan Key Laboratory of Artificial Intelligence, Kunming University of Science and Technology, Kunming 650500, China)

Abstract: k-Nearest Neighbor Machine Translation (kNN-MT) significantly improves the accuracy of neural machine translation (NMT) model predictions by retrieving translation knowledge from external data stores. However, the use of a fixed fusion ratio to aggregate the probability distributions of NMT model predictions and kNN retrievals makes the model susceptible to noise in the retrieval results, and the high latency of kNN retrieval limits its practical application. To this end, this paper proposes a k-nearest neighbor machine translation method based on confidence optimization. Specifically, a confidence estimation module is introduced to dynamically evaluate the reliability of the probability distribution of NMT predictions and the kNN retrieval distribution, and the probability fusion ratio is calculated in an adaptive manner to improve the accuracy of translation. At the same time, redundant knowledge instances in the datastore are pruned based on the confidence of the model to improve the decoding efficiency of the model. Experimental results on two sets of specific language pair translation tasks show that this method significantly outperforms the standard kNN-MT model in both translation quality and decoding efficiency.

Keywords: k nearest neighbor machine translation; datastore; retrieval; confidence

0 引 言

神经机器翻译模型[1]采用神经网络参数来编码翻译知识,实现了从传统的符号匹配规则向基于连续空间表示学习的范式转变.这种参数化的知识表示方法不仅能够有效捕获深层语义特征,还在处理复杂语言现象和跨语言泛化方面展现出显著优势.然而,神经网络参数的知识学习能力存在一定的上限,特别是在处理低频语言现象[2]时往往效果欠佳.此外,当需要引入新的翻译知识时,神经网络模型往往需要对大量参数进行重新优化,这不仅带来了额外的计算开销,也制约了模型在实际应用中的可扩展性和适应能力[3,4].

检索增强型神经机器翻译(RE-NMT)[5,6]是一种结合信息检索和神经机器翻译的方法.传统 NMT 在处理低频词、专业术语等方面存在局限性,通过引入检索机制可以更好地利用已有的翻译知识,这种方法最早由 Collier 等人[7]提出,此后得到广泛研究和发展.其核心优势在于能够显式地从外部记忆模块检索翻译知识,而不仅局限于神经网络参数的隐式表达.由于支持推理过程中并行访问外部语料库,检索增强型方法相比传统参数化方法展现出更强大的表达能力.其中,Khandelwal 等人提出的 kNN-MT[8]作为检索增强型机器翻译的代表,因其简洁的设计理念和卓越的翻译效果而备受瞩目.该方法将训练语料中的所有 Token 级翻译实例以键值对(Key-Value)的形式存储,其中键为 NMT 模型的解码器状态,值为该状态对应的真实目标标记.在推理阶段,NMT 模型利用当前上下文表示作为查询向量,从数据存储中检索出 k 条与其最相似的实例来增强模型的翻译性能.该方法使 NMT 模型具

收稿日期:2025-02-25 收修改稿日期:2025-05-13 基金项目:国家自然科学基金地区基金项目(62166022)资助;云南省"兴滇英才支持计划"项目(KKXX202403023)资助. 作者简介:**周茂春**,男,1998 年生,硕士研究生,研究方向为自然语言处理、机器翻译;**朱俊国**(通信作者),男,1982 年生,博士,副教授,CCF 会员,研究方向为机器学习、自然语言处理、机器翻译.

备更强大的适应和泛化能力,并已成功扩展到文本生成[9,10]、语音识别[11,12]、情感分类[13]、语法纠错[14,15]等多个领域.

尽管 kNN-MT 取得了令人瞩目的进展,但仍然存在两个亟待解决的性能瓶颈. 首先,kNN-MT 模型的翻译性能对检索时超参数的设置表现出高度敏感性,尤其是用于融合翻译概率和检索概率的比例系数,不同的翻译场景需要探索不同的融合比例. 当检索结果与查询向量的匹配度较低或者 NMT 模型自身具有较高的置信度时,使用固定的融合比例往往会削弱模型的鲁棒性和泛化性. 其次,在解码过程的每一时刻,NMT 模型都需要对数据存储中所有的翻译实例进行一次完整的搜索,这种频繁的检索操作不仅导致计算开销巨大,而且随着数据规模的扩大,计算成本将以指数级别的速度增长[16]. 即便借助 Faiss[17]等高效的向量检索工具,kNN-MT 模型的解码效率仍然难以令人满意.

为了有效解决上述问题,本文提出了一种基于置信度优化的 k 近邻机器翻译方法. 具体而言,在 Transformer 模型的解码器端构建置信度估计模块以评估其对当前预测的置信度,以无监督的方式将置信度估计引入至 NMT 模型的训练过程中. 根据 NMT 模型的置信度与检索实例的可信度动态调整融合模型预测概率与 kNN 检索概率的比例,减少检索结果中噪声的干扰,从而提升模型的泛化性能. 为提高模型的翻译速度,基于模型置信度实现自适应检索,即确定 NMT 模型是否需要检索数据存储中的翻译知识,减少不必要的检索次数;其次,对数据存储中的翻译知识进行筛选,仅保留能弥补模型预测错误或者可能出错的知识实例. 深入分析发现,基于检索增强的方法不仅能有效缓解 NMT 模型因标准交叉熵损失训练导致的过度校正问题[18],还提升其对稀有词的翻译准确率. 在两组特定语言对的翻译实验表明,该方法相比于基线模型,翻译质量和解码效率上均实现显著提升.

1　相关工作

1.1　检索增强型神经机器翻译

近年来,检索增强型神经机器翻译方法已经成为机器翻译领域中的研究热点. 该方法通过在 NMT 模型中引入信息检索技术,能够从训练语料库中检索翻译记忆[19](Translation Memory,TM)来增强模型的性能,其通常是由一组特定语言对的句子或者短语组成的数据库,用于存储由人工翻译人员翻译的样例.

将翻译记忆集成到神经机器翻译模型的主要方法可分为两类:在输入中串联类似的翻译样例以训练更强大的 NMT 模型或者用于约束模型的解码过程. 前者旨在训练生成模型以学习如何处理检索到的 TM,后者的主要思想是在 TM 的基础上增加 NMT 模型对目标词的生成概率. 例如,为了促进不同翻译记忆之间的直接通信以获取局部信息,并通过消息传递机制收集全局上下文,大量的研究工作[20-22]通过门控单元、注意力模块、记忆编码器等在翻译模型中融入检索到的翻译示例. 然而,这类方法通常需要仔细修改 NMT 模型的架构或者引入额外的检索模型,以充分利用检索到的相似样例,这使得其在其在现实场景中部署的代价较高. 其次,基于句子级别的检索往往难以找到与测试句子足够相似且高质量的翻译示例,引入低重叠率的翻译示例会严重影响模型的翻译性能.

1.2　k 近邻机器翻译的优化

k 近邻神经机器翻译通过将检索粒度由句子或短语级别细化至 Token 级别,并采用连续向量空间的相似度检索机制替代符号化的精确匹配规则,通过在模型的输出层中融合相似翻译示例的 Logits 来提高机器翻译的质量. 然而,其有效性很大程度上取决于检索过程中超参数的设置以及所构建数据存储的质量. Adaptive kNN-MT[23]通过构建额外的轻量级网络以动态确定 kNN 检索过程中的最优的检索数量. Jiang 等人[24]基于高斯或者拉普拉斯核的平滑技术处理检索到的翻译示例,并通过神经网络自适应建模混合权重,提升模型的泛化能力. Cao 等人[25]通过修正数据存储中的键向量以缩小领域差距以实现更加高效的机器翻译领域自适应. Yang 等人[26]则认为简单的线性插值策略不能根据检索句子的匹配程度动态调整融合比例,提出基于登普斯特-谢弗理论(DST)的动态融合策略以适应不同场景. 此外,还有部分工作建议使用更强大的模型[27]以提升键向量与值标记的一致性,或者在神经网络模型中添加适配器[28]模块,在模型训练阶段向目标函数注入外部知识,迭代更新 NMT 模型并且刷新数据存储.

其次,k 近邻机器翻译方法所带来的翻译延迟和存储开销是巨大的. 为此,一系列研究致力于探索高效的数据存储压缩策略或者对检索方式进行优化以加快机器翻译的生成速度. 对于压缩数据存储的规模,Martins 等人[29]提出的 E-kNN-MT 贪婪地合并共享相同值标记的相邻对来减少数据存储中的实例数量,并运用主成分分析(PCA)等算法对数据存储中高维向量进行压缩处理. 虽然它能有效的缩减数据存储的规模,但缺点是在降低向量维度的过程中会丢失高维位置的信息,从而严重影响翻译性能. Meng 等人[30]为每个源标记构建额外的数据存储,并使用其数据存储和字对齐来减少 kNN 搜索空间. Dai 等人[31]使用 Elastic Search 工具,通过 BM25 分数搜索与测试句子相似的少量文本,构建不同的小规模数据存储减少 kNN 检索的空间. Wang 等人[32]提出的 PCK kNN-MT 则使用基于集群的紧凑网络来压缩存储密钥的维度,并结合基于集群的修剪策略来丢弃冗余对以提高模型的解码效率,但代价是该方法训练成本是巨大的. 而 Marinho 等人[33]建议从数据存储中检索多个连续标记,而不是单个词标记. Lv 等人[34]则将相邻的 N 元隐藏表示进行拼接作为键,而目标标记的元组(Tuple)作为值,这本质上是通过增大检索粒度来提升解码速度. 最近,Faster kNN-MT[35]引入一个基于多层感知机(MLP)的二分类器模块,用于在每个解码时间步判断是否可以跳过 kNN 检索以提升解码速度. 然而,由于正负样本的极度不平衡,以及解码器表示较为复杂,简单的 MLP 网络难以实现较高的识别性能,而更复杂的网络结构则会带来更大的训练成本.

2　方　法

2.1　置信度估计

随着深度神经网络在机器翻译领域的广泛应用,模型置信度估计的重要性日益凸显. 置信度估计能够量化模型对其预测结果的确信程度,而经过精确校准的置信度评估机制可

以有效识别神经机器翻译(NMT)模型潜在的失误,从而提供更可靠的翻译质量保证.

NMT 模型在推理过程中会输出对目标预测的概率分布,从理论上来看,概率值的大小应当反映模型对其预测结果的确信程度. 然而,Fomicheva 等人的实验表明[36],NMT 模型输出的概率分布与其实际预测的准确性之间存在显著偏差. 即便 NMT 模型做出明显错误的预测,其输出的概率分布依然可能呈现出高置信度,这种"过度自信"现象严重影响了模型可靠性的评估. 因此,在缺乏参考答案的推理阶段,仅凭模型的预测概率来评估其置信度是次优的.

受 DeVries 等人[37]在图像分类任务和 Lu 等人[38]在译文质量估计任务上的启发,本文将置信度定义为 NMT 模型需要多少真实标签分布的提示信息才能正确翻译当前单词. 请求提示的量越小,说明 NMT 模型对当前预测的置信度较高,翻译正确的可能性越大. 因此,置信度估计的目标是在 NMT 模型的训练过程中学习评估词级别预测的置信度.

为了实现这一思想,如图 1 所示,首先在 Transformer 模型的解码器端添加一个置信度估计模块(CEM),用于请求标准答案的提示,该 CEM 模块为轻量级的线性映射层.

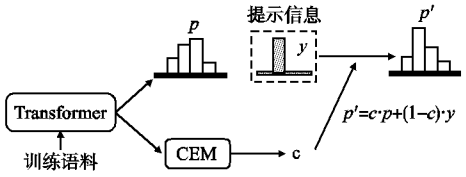

图 1 配备置信度估计的 NMT 模型

Fig. 1 NMT model with confidence estimation

CEM 模块以 t 时刻 Transformer 解码器的隐藏状态 h_t 作为输入,并输出一个标量 c_t,表示 NMT 模型在翻译当前单词时的置信程度. 置信度越高,表示模型对该单词的预测越有把握,所需的提示量越小;相反,若置信度较低,则会请求更多的提示以提高翻译的准确性.

$$\begin{aligned} h_t &= \mathrm{AVE}(h_1^t + h_2^t + h_3^t) \\ c_t &= sigmoid(\mathbf{W}_1 h_t + b_1) \end{aligned} \tag{1}$$

如公式(1)所示,为减轻 Transformer 高层解码器状态的负担,使用其前三层解码器状态的平均作为分支网络的输入. 其中,$\mathbf{W}_1$ 和 b_1 是该分支网络可训练的模型参数.

在模型训练过程中,CEM 模块可以请求一定量的提示 y_t 插值到 NMT 模型预测的概率分布 p_t 中,从而生成更加准确的翻译预测 p_t',请求提示的大小由 CEM 模块的预测 c_t 决定.

$$p_t' = c_t \cdot p_t + (1 - c_t) \cdot y_t \tag{2}$$

NMT 模型的训练目标定义为最小化其预测概率分布 p_t' 与真实标签分布 y_t 之间的交叉熵损失,其中 T 表示待翻译句子的总长度.

$$\mathcal{L}_{NMT} = \sum_{t=1}^{T} -y_t \log(p_t') \tag{3}$$

为了防止 NMT 模型始终将 c_t 设置为 0 来最小化翻译损失,在损失函数中引入了一个对数惩罚项:

$$\mathcal{L}_{Conf} = \sum_{t=1}^{T} -\log(c_t) \tag{4}$$

翻译损失与惩罚项通过参数 α 加权组合得到 NMT 模型训练的总损失:

$$\mathcal{L} = \mathcal{L}_{NMT} + \alpha \mathcal{L}_{Conf} \tag{5}$$

具体而言,当 c_t 越趋于 1 时,表明 NMT 模型对当前预测具有较高的置信度,此时请求提示的数量越少,其所受到的惩罚也相应减少. 相反,当模型对翻译结果不够自信时,请求提示的数量增多,而这一行为会受到较高的惩罚. 在此设置下,本文鼓励 NMT 模型在大多数情况下独立翻译以避免惩罚,但在不确定决策时请求提示以确保减少总损失.

在训练的初始阶段,模型往往比较脆弱,NMT 模型尚未能够充分学习到翻译任务的特性. 若在训练初期为模型给予提示,将导致模型会过度依赖于外部信息,抑制其主动学习的能力,最终影响 NMT 模型的翻译性能. 因此,本文根据训练过程中的更新步数来动态调整的值,如公式(6)所示:

$$\alpha(s) = \alpha_0 * e^{\frac{-s}{\beta_0}} \tag{6}$$

α_0 和 β_0 分别控制加权系数 α 的初始值与下降速度,s 则表示训练阶段的更新步数. 在 NMT 模型训练的早期阶段,若模型请求提示将会获得较大的惩罚. 但随着更新步数的增大,加权系数会逐渐减小,模型请求提示所获得的惩罚也就越少.

该策略能够有效避免 NMT 模型在训练的初期过度依赖提示信息,从而保持模型具备一定的自我学习能力,并使模型在训练后期充分利用提示信息来提升自身的翻译性能. 在两组语言对的翻译实验中,均将 α_0 设置为 30,β_0 设置为 45000.

2.2 k 近邻机器翻译

训练好具有置信度估计模块的 NMT 模型后,将训练语料 $(x,y) \in (X,Y)$ 以教师强制(Teacher Forcing)的方式再次输入到 NMT 模型中,进行一次完整的前向传播以构造数据存储 D.

将训练语料的双语句子对转换成一组键值对的形式进行保存,键指的是每一时刻 NMT 模型的解码器状态 $f(x,y_{<t})$,该状态所对应的真实目标标记 y_t 作为值,完整的数据存储可定义为:

$$D = \bigcup_{(x,y) \in (X,Y)} \{ (f(x,y_{<t}), y_t), \forall y_t \in y \} \tag{7}$$

在推理过程中,NMT 模型首先利用当前解码器的状态作为查询向量去检索数据存储中与之距离最近的 k 条候选实例,可将其表示为:

$$N^t = \{ (h_i, v_i), i \in \{1,2,\cdots,k\} \} \tag{8}$$

然后,将这 k 条候选实例映射至词汇表上的概率分布,可通过公式(8)计算. 其中 y_t 表示 t 时刻正确的翻译标记,v_i 和 h_i 分别表示检索到的第 i 条实例数据中的值标记和键向量.

距离函数使用欧式距离或余弦相似度,使用大于 1 的温度系数 T 可以使 kNN 分布变得平坦,防止过度拟合于最相似的近邻实例.

$$P_{\mathrm{knn}}(y_t \mid x, y_{<t}) \propto \sum_{i=1}^{k} \mathbb{I}_{y_t = v_i} \exp\left(\frac{-d(h_i, f(x,y_{<t}))}{\mathrm{T}} \right) \tag{9}$$

最后,将 kNN 检索得到的概率分布与 NMT 模型预测的概率分布进行简单的线性插值:

$$P_{final}(y_t \mid x, y_{<t}) = \lambda P_{\mathrm{knn}} + (1 - \lambda) P_{\mathrm{nmt}} \tag{10}$$

2.3 动态计算融合比例

标准的 kNN-MT 使用固定值的融合比例 λ 来聚合 kNN 检索概率和翻译概率,但当 kNN 检索结果不准确或者 NMT 模型对其自身预测具有足够自信时,这会阻碍模型的泛化性,

导致最终的预测受到检索结果中的噪声干扰.

为此,本文构建一个由两层前馈网络组成的轻量级模块(REM)用于评估 kNN 检索结果的可靠性. 然后,同时利用 kNN 检索结果的可靠性与 NMT 模型对其预测的置信度来动态估计融合比例. 如图 2 所示,本文从检索结果中提取两种特征来构造 REM 模块的输入.

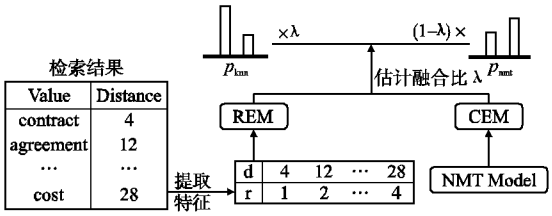

图 2　动态估计融合比例示意图

Fig. 2　Diagram of dynamic estimation fusion ratio

第 1 个特征是距离,该值为在翻译过程中,NMT 模型的翻译上下文 $f(x,y_{<t})$ 与检索结果中第 i 条数据的键向量 h_i 之间的欧几里德距离 d_i,距离是评估每个实例条目重要性的直接证据.

$$d_i = \| f(x,y_{<t}) - h_i \|_2 \tag{11}$$

第 2 个特征是检索结果的不同值的计数,该值衡量检索结果的一致性,记为 r_i,其中 UNIQUE 表示列表中的唯一元素. 如果每个检索结果差异性较大,那么 kNN 预测的可信度就会降低,应该更多地依赖 NMT 预测.

$$r_i = \text{UNIQUE}(v_1, v_2, \cdots, v_i) \tag{12}$$

REM 模块的输入由这两类特征拼接所构成,目标是预测检索结果的可靠性 z_t,这两个特征对于评估 kNN 检索结果的质量均至关重要.

$$\mathrm{z}_t = sigmoid(\mathbf{W}_2(\tanh(\mathbf{W}_3[d_i;r_i]) + b_2)) \tag{13}$$

融合比例的大小除了与检索结果的可靠性密切相关,还需要考虑 NMT 模型的置信度,它来自 CEM 模块的输出. 当 NMT 模型对其预测表现出高置信度时,对外部知识的依赖应相应降低,kNN 分布的权重应随之减小.

基于上述认识,将检索结果的可信度和 NMT 模型预测的置信度作为两个关键因素来自适应地计算融合比例.

$$\lambda_t = \frac{\mathrm{z}_t}{\mathrm{z}_t + c_t} \tag{14}$$

本文在训练过程中冻结了配备置信度估计模块的 NMT 模型,仅更新 REM 模块的参数,通过交叉熵损失训练模型直至收敛.

值得注意的是,使用训练数据集训练该模块是不合适的,因为初始的 NMT 模型已经拟合于该数据,并且数据存储也是由该数据所构建的. 为了推广到测试数据,本文按照 Zheng 等人的方法[23]保留 10% 的验证集进行测试,并使用剩余 90% 的验证集来训练 REM 模块.

2.4　基于置信度的数据存储修剪策略

NMT 模型需要在每个解码时刻对数据存储中所有的翻译实例进行查询以计算 kNN 概率分布,这种全量检索策略虽然保证了模型的翻译质量,但却带来严重的翻译延迟. 在本节中,本文从修剪数据存储规模两个角度出发,基于模型的置信度来优化 kNN-MT 的解码效率.

直觉上,数据存储中只需保留 NMT 模型预测错误或可能出错的实例知识即可. 针对数据存储中的每个条目,本文检查 NMT 模型能否根据隐藏层的表示预测出目标词,对数据存储中所有知识条目的准确性进行系统性评估. 具体地,在构建数据存储的过程中,不仅记录每一时刻下 NMT 模型的解码器状态 $f(x,y_{<t})$ 与该状态对应的真实目标标记 y_t,还记录模型预测的标记 y'_t 以及 CEM 模块的输出 c_t.

修剪策略如下:首先,将 NMT 模型预测等于真实目标标记的知识条目的索引纳入删除候选列表. 然后,根据置信度将索引按从大到小的方式排序. 最后,从删除候选列表中按顺序选择一定比例的知识条目进行删除. 在本文的实验中,修剪比例设置为 0.40,这一过程可以用算法 1 中伪代码的形式描述.

算法 1. 基于置信度排序的数据存储修剪策略

输入: 数据存储 D、修剪比例

输出: 修剪后的数据存储 D'

```
candidates ← ∅  //创建可删除条目的候选索引列表
for each entry (h(x, y_<t), y_t) in D do
    if y'_t = y_t then:
    // 若模型预测标记 y'等于真实标记 y_t
        candidates ← candidates ∪ (h(x, y_<t), y_t)
    //将该条目的索引纳入可删除的候选列表
    end if
end for
candidates ranked by c_t//根据置信度从大到小排序索引
repeat  //重复执行
    select entry (h(x, y_<t), y_t) from candidates
     //按顺序挑选索引
    remove (h(x, y_<t), y_t) from D
    //从数据存储中移除该索引对应的知识条目
until pruning ratio is satisfied //直到达到预定的修剪比例
return pruned datastore D'
```

3　实　验

3.1　数据集与实验设置

为了验证所提方法的有效性,本文从 JRC-Acquis 翻译语料库[39]中选取西班牙语-英语(ES-EN)和德语-英语(DE-EN)翻译数据进行实验分析,该语料库由欧盟法律的平行立法文本组成,具有广泛的适用性. 为了进一步提高训练数据的质量,过滤源语言和目标语言长度比超过 1.5 的句子对. 然后,本文使用 Moses 工具包对翻译数据进行标准化和分词(https://github.com/moses-smt/mosesdecoder),并且利用 Subword-NMT(https://github.com/rsennrich/subword-nmt)学习 32k 操作数的 BPE 规则,生成相应的 BPE 编码数据和词汇表.

表 1　JRC-Acquis 语料库句子数量统计

Table 1　Statistics of sentence counts for JRC-Acquis corpus

	DE-EN	ES-EN
训练集	634891	626603
验证集	2454	2533
测试集	2483	2596
数据存储	22923208	21678595

表 1 展示了该语料库中句子对数量和数据存储的规模(即所包含的键值对数量).

本文使用 Fairseq[40] 和 kNN-BOX[41] 开源工具包对所有模型进行训练和评估. NMT 模型采用 Transformer 架构,其训练时设置最大更新步数为 100000 步,最大 Token 数为 8192,预热步数为 4000,初始学习率 0.0005,dropout 比例为 0.1. 为了提升模型训练的稳定性和效率,本文采用自适应调整学习率的 Adam 优化器,其动量参数 β_1 和 β_2 分别设置为 0.90 和 0.98. 其次,kNN 检索过程中的温度系数和检索个数分别设置为 10 和 8. 翻译质量均使用区分大小写的去标记化的 BLEU 得分进行评估,所有方法均是在配备了单张 GEROCER RTX 3090 GPU 的 Ubuntu 服务器上实现.

3.2 对比模型

1) Transformer[1]:神经机器翻译模型的常用架构,该模型也用于初始化其他 kNN-MT 模型.

2) TM-Augment[22]:从单语语料库中检索相关的翻译记忆片段,通过交叉熵损失函数联合优化记忆检索器和 NMT 模型.

3) V-kNN-MT[8]:基线模型,kNN 检索过程中所涉及的超参数(如检索数、融合比例等),由网格搜索(Grid Search)寻找最优组合,本文的优化工作均是在此模型的基础上进行开展.

4) A-kNN-MT[23]:训练轻量级网络动态估计检索个数,并且放弃简单的线性插值,而是采用所有检索数情况的聚合.

5) E-kNN-MT[29]:使用 PCA 算法降维数据存储中的高维向量,以及将数据存储中具有相同值标记的知识条目进行贪婪合并.

6) PCK kNN-MT[32]:Adaptive kNN-MT 模型的基础上,训练额外的神经网络以减少数据存储中高维向量的维度与数据存储的规模.

7) Faster kNN-MT[35]:在 Adaptive kNN-MT 的基础上,通过对比 NMT 模型的预测结果和正确标记是否一致制作训练样本,然后为 kNN-MT 训练一个二分类选择器以减少不必要的检索操作.

3.3 实验结果与分析

在两组特定语言对的实验结果如表 2 所示,其中 CP 代表基于置信度的数据存储修剪策略,且修剪数据存储的比例大小设置为 30%.

表 2 翻译性能对比

Table 2 Comparison of translation performance

Model	DE-EN	ES-EN	EN-DE	EN-ES
Transformer	58.59	62.70	54.27	60.20
TM-Augment	63.52	66.45	57.45	62.80
V-kNN-MT	62.91	65.64	57.60	62.37
A-kNN-MT	63.65	67.15	58.53	63.48
E-kNN-MT	62.20	65.06	56.50	62.20
PCK	63.28	66.84	57.96	63.04
Faster kNN-MT	62.96	66.27	57.32	62.62
Ours	64.15	67.20	58.98	63.86
Ours + CP	63.81	66.78	58.62	63.50

观察实验结果,本文可以得出以下结论:首先,基于 kNN 检索增强的神经机器翻译模型通过引入实例知识显著提升了翻译性能. 其次,本文提出的基于模型置信度的自适应概率融合方法通过动态计算融合比例系数,有效抑制了 kNN 检索中的噪声干扰,模型的翻译质量明显优于其他 kNN-MT 方法. 虽然 E-kNN-MT 等方法能够显著提升模型的解码速度,但却导致模型的翻译质量大幅下降. 相比之下,本文提出的存储修剪策略能够减少数据存储中大部分冗余的知识实例,从而优化模型的解码效率.

3.4 不同修剪比例对模型翻译性能的影响

表 3 展示了不同修剪比例下的 4 个翻译方向的翻译性能. 数据显示,随着修剪比例的增加,翻译性能呈现逐步下降的趋势. 当修剪比例为超过 30% 时,翻译性能的降幅逐渐增大. 考虑到翻译性能和模型解码效率的平衡,本文最终选择 30% 作为最优修剪比例大小.

表 3 不同修剪比例下翻译性能测试

Table 3 Translation performance test under different pruning ratios

修剪比例	DE-EN	ES-EN	EN-DE	EN-ES
10%	63.98	67.15	58.93	63.84
20%	63.85	66.92	58.75	63.78
30%	63.81	66.78	58.67	63.62
40%	63.14	66.37	58.12	63.06
50%	62.82	66.02	57.50	62.86

3.5 消融研究

为了验证不同组件对神经机器翻译模型性能的影响,本文以 Transformer 模型为基础进行消融研究. 其中,CEM 表示置信度估计模块,RE 代表 kNN 检索增强,括号里的数值表示相比于 NMT 模型翻译性能的增幅.

表 4 消融研究结果

Table 4 Results of ablation study

编号	方法	DE-EN	ES-EN
1	Transformer	58.59	62.70
2	1 + CEM	58.90(+0.31)	62.92(+0.22)
3	1 + RE	62.91(+4.32)	65.64(+2.94)
4	2 + RE	63.28(+4.69)	65.96(+3.26)
5	4 + 动态融合	64.15(+5.56)	67.72(+5.02)

从表 4 的实验结果可知,引入 CEM 模块不仅能够量化分析 NMT 模型预测的置信度,还能够少量提升其翻译性能. 原因在于,它可以在训练过程中校正模型本身的置信度估计,从而减轻由于暴露偏差而导致测试阶段的置信度偏差. 其次,kNN 检索外部数据存储为模型引入额外的翻译知识,能够显著提升 NMT 模型的翻译性能. 此外,自适应地调整聚合检索概率和翻译概率的融合比例,能够有效抑制检索结果中噪声的干扰,相比于使用固定系数的 kNN-MT 模型,平均取得了 1.3 BLEU 的性能增益.

3.6 检索增强的有效性分析

尽管 kNN 检索引入外部知识能够显著提升 NMT 模型的翻译性能,但其有效性的原因尚未得到充分探究. 本文推测主要源于两个方面:一方面,外部知识的引入有助于缓解 NMT 模型的过度校正问题;另一方面,它提高了模型对低频词的翻译准确性.

1) 过度校正问题分析

标准 NMT 模型使用交叉熵损失函数进行训练,这种训练范式要求模型预测与参考答案实现严格的字符匹配. 然而,

一个句子的表达具有多样性,即使模型生成的词汇在语义上合理但与参考答案存在偏差,交叉熵损失函数仍会对其施加惩罚. 这种现象被称为过度校正,它显著制约了 NMT 模型的泛化能力.

通过一个典型案例进行分析:给定一个德语的源语言句子“Der Vertrag enthält viele missverständliche Rechtsbegriffe.(该合同包含大量含糊不清的法律条款.)”和英语的目标子序列“The contract contains”作为翻译上下文,“many…”、“lots of…”和“a lot of…”都是正确的翻译. 将此翻译上下文输入至 NMT 模型与本文模型中,借助 kNN-BOX 的可视化工具观察它们在词汇表上的预测概率.

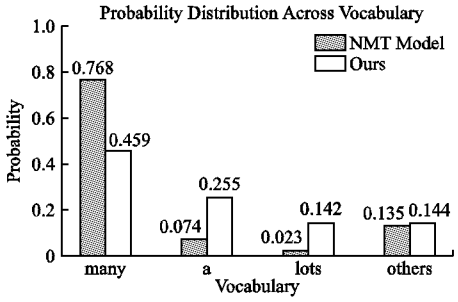

图 3　过度校正问题分析

Fig. 3　Overcorrection problem analysis

如图 3 所示,虽然两个模型都将最高概率赋予了与参考答案一致的“many”,但其概率分布存在显著差异. 标准 NMT 模型由于受交叉熵损失训练的影响,表现出明显的过度校正特征,它将极高的概率数值集中分配给标注答案,而严重抑制了其他语义合理的候选词(如“a”和“lots”). 相比之下,本文方法通过利用检索到的实例知识,实现了对其他合理候选词的概率增强. 这一案例分析有力地证明了检索增强机制在改善模型预测分布方面的积极作用,不仅提高了模型对语言表达多样性的适应能力,也为缓解神经机器翻译中的过度校正问题提供了一种有效解决方案.

2)词频准确性分析

神经机器翻译模型通过参数化学习来捕获翻译模式,然而这种方法在处理低频词和未见词时往往表现出有限的泛化能力. 相比之下,检索增强的方法能够从数据存储中直接获取相似实例,为低频词的翻译提供具体的参考样例,从而有效提升其翻译准确率.

为了验证这一假设,本文使用 COMPARE-MT 工具(https://github.com/neulab/compare-mt),将模型输出的译文与参考译文进行对比,重点评估不同词频区间的翻译准确性. 正如表 5 所示,基于检索增强的方法在各词频区间的表现均优于传统 NMT 模型,特别是词频小于 100 的稀有词.

3.7　数据存储修剪策略对比

数据存储修剪的目标是减小数据存储的大小,以便模型能够更快地搜索相识实例,而不会严重影响模型的翻译质量. 本着控制变量的原则,将修剪比例统一设置为 30%,然后对比本文模型运用以下几种修剪策略后翻译性能的变化.

1)随机修剪(RP):随机选择数据存储中一定比例的知识实例进行删除.

2)k 均值聚类(k-Means):聚类是一种通过仅保留质心来修剪冗余向量的常用技术.

3)贪婪合并(GM):将数据存储中值标记相同的知识实例进行合并,即:数据存储中相同值标记所对应的键向量取平均.

4)基于置信度的修剪策略(CP):本文所提出的方法. 首先筛选出模型能够准确预测的实例,然后根据置信度对该部分实例进行排序,按顺序删除直至符合预设的修剪比例.

表 5　不同词频区间下各模型的翻译准确性比较

Table 5　Comparison of translation accuracy across different word frequency ranges

词频区间	NMT	Ours
[1,5)	0.5151	0.5511
[5,10)	0.5909	0.6294
[10,100)	0.6390	0.6771
[100,1000)	0.7387	0.7504
[1000,+)	0.7736	0.7800

表 6　各修剪策略对模型性能的影响

Table 6　Impact of each pruning strategy on model performance

Model	DE-EN	ES-EN
Ours	64.15	67.20
Ours + RP	62.27(−1.88)	65.36(−1.84)
Ours + k − Means	63.04(−1.11)	65.98(−1.22)
Ours + GM	63.25(−0.90)	66.15(−1.05)
Ours + CP	63.81(−0.34)	66.78(−0.42)

如表 6 所示,随机删除数据存储中的知识条目会导致翻译性能严重下降. 对于聚类与贪婪合并策略,翻译性能也大幅降低. 这表明同一聚类中的向量可能对应于各种目标标记,而简单地合并具有相同标记值的实例也是不合理的,因为这些实例虽然目标标记相同,但通常包含不同的语义信息,具有不同的重要性. 相比之下,本文提出的基于置信度的修剪策略能够更高效地修剪数据存储中的知识条目,从而在提升模型解码效率的同时取得更好的翻译效果.

4　结束语

本文提出一种基于置信度优化的 k 近邻机器翻译方法,该方法通过评估模型预测与检索结果的置信度,自适应调整概率融合比例,从而增强模型的泛化能力. 为提升解码效率,利用模型的置信度对数据存储中冗余的知识实例进行修剪. 未来,将进一步探索优化 k 近邻机器翻译的新方法,并尝试将该方法扩展至更广泛的序列生成任务领域.

References:

[1] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need [C]//Proceedings of the 31st International Conference on Neural Information Processing Systems, 2017: 6000-6010.

[2] Ranathunga S, Lee E S, Prifti S M, et al. Neural machine translation for low-resource languages: a survey [C]//ACM Computing Surveys, 2023: 1-37.

[3] Saunders D. Domain adaptation and multi-domain adaptation for neural machine translation: a survey [J]. Journal of Artificial Intel-

ligence Research,2022:351-424.
[4] Xu Y, Wang S, Li P, et al. Pluggable neural machine translation models via memory-augmented adapters[C]//Proceedings of the Joint International Conference on Computational Linguistics, Language Resources and Evaluation,2024:12794-12808.
[5] Zamani H, Diaz F, Dehghani M, et al. Retrieval-enhanced machine learning[C]//Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval,2022:2875-2886.
[6] Zhong Z, Lei T, Chen D. Training language models with memory augmentation[C]//Proceedings of the Conference on Empirical Methods in Natural Language Processing,2022:5657-5673.
[7] Collier M, Beel J. Memory-augmented neural networks for machine translation[C]//Proceedings of Machine Translation Summit XVII: Research Track,2019:172-181.
[8] Khandelwal U, Fan A, Jurafsky D, et al. Nearest neighbor machine translation[C]//Proceedings of the 9th International Conference on Learning Representations,2021:1-14.
[9] Shi W, Michael J, Gururangan S, et al. kNN-prompt: nearest neighbor zero-shot inference[J]. arxiv preprint arxiv: 2205.13792,2022.
[10] Wang S, Song Y, Drozdov A, et al. Knn-lm does not improve open-ended text generation[C]//Proceedings of the Conference on Empirical Methods in Natural Language Processing, 2023: 15023-15037.
[11] Zhou J, Zhao S, Liu Y, et al. kNN-CTC: enhancing asr via retrieval of ctc pseudo labels[C]//ICASSP IEEE International Conference on Acoustics, Speech and Signal Processing,2024:11006-11010.
[12] Li S, Wei D, Shang H, et al. Speaker-smoothed kNN speaker adaptation for end-to-end asr[J]. arXiv preprint arXiv:2406.04791,2024.
[13] Wang S, Li X, Meng Y, et al. kNN-NER: named entity recognition with nearest neighbor search[J]. arxiv preprint arxiv: 2203.17103,2022.
[14] Kaneko M, Takase S, Niwa A, et al. Interpretability for language learners using example-based grammatical error correction[C]//Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics,2023:7176-7187.
[15] Vasselli J, Watanabe T. A closer look at k-nearest neighbors grammatical error correction[C]//Proceedings of the 18th Workshop on Innovative Use of NLP for Building Educational Applications (BEA),2023:220-231.
[16] Hao H, Huang G, Liu L, et al. Rethinking translation memory augmented neural machine translation[C]//Findings of the Association for Computational Linguistics,2023:2589-2605.
[17] Johnson J, Douze M, Jégou H. Billion-scale similarity search with GPUs[C]//IEEE Transactions on Big Data,2019:535-547.
[18] Zhang W, Feng Y, Meng F, et al. Bridging the gap between training and inference for neural machine translation[C]//Proceedings of the 29th International Conference on International Joint Conferences on Artificial Intelligence,2021:4790-4794.
[19] Reinke U. State of the art in translation memory technology[J]. Language Technologies for a Multilingual Europe, 2018: 55-84, doi:10.5281/zenodo.1291930.
[20] Cheng X, Gao S, Liu L, et al. Neural machine translation with contrastive translation memories[C]//Proceedings of the Conference on Empirical Methods in Natural Language Processing, 2022: 3591-3601.
[21] He Q, Huang G, Cui Q, et al. Fast and accurate neural machine translation with translation memory[C]//Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing,2021:3170-3180.
[22] Cai D, Wang Y, Li H, et al. Neural machine translation with monolingual translation memory[C]//Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing,2021:7307-7318.
[23] Zheng X, Zhang Z, Guo J, et al. Adaptive nearest neighbor machine translation[C]//Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics,2021:368-374.
[24] Jiang Q, Wang M, Cao J, et al. Learning kernel-smoothed machine translation with retrieved examples[C]//Proceedings of the Conference on Empirical Methods in Natural Language Processing, 2021:7280-7290.
[25] Cao Z, Yang B, Lin H, et al. Bridging the domain gaps in context representations for k-nearest neighbor neural machine translation[C]//Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics,2023:5841-5853.
[26] Yang Z, Hou H, Sun S, et al. Dynamic fusion nearest neighbor machine translation via dempster-shafer theory[C]//China Conference on Machine Translation, Singapore: Springer Nature Singapore,2022:82-92.
[27] Li J, Cheng S, Sun Z, et al. Better datastore, better translation: generating datastores from pre-trained models for nearest neural machine translation[J]. arxiv preprint arxiv:2212.08822,2022.
[28] Zhu W, Xu J, Huang S, et al. INK: injecting KNN knowledge in nearest neighbor machine translation[C]//Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics, 2023:15948-15959.
[29] Martins P H, Marinho Z, Martins A F. Efficient machine translation domain adaptation[C]//Proceedings of the 1st Workshop on Semiparametric Methods in NLP,2022:23-29.
[30] Meng Y, Li X, Zheng X, et al. Fast nearest neighbor machine translation[C]//Findings of the 60st Annual Meeting of the Association for Computational Linguistics,2022:555-565.
[31] Dai Y, Zhang Z, Liu Q, et al. Simple and scalable nearest neighbor machine translation[C]//Proceedings of the 11th International Conference on Learning Representations,2023:1-17.
[32] Wang D, Fan K, Chen B, et al. Efficient cluster-based k-nearest-neighbor machine translation[C]//Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics, 2022: 2175-2187.
[33] Martins P H, Marinho Z, Martins A F T. Chunk-based nearest neighbor machine translation[C]//Proceedings of the Conference on Empirical Methods in Natural Language Processing, 2022: 4228-4245.
[34] Lv R, Guo J, Wang R, et al. N-gram nearest neighbor machine translation[C]//ACM Transactions on Audio, Speech, and Language Processing,2024:17-29.
[35] Shi X, Liang Y, Xu J, et al. Towards faster k-nearest-neighbor machine translation[J]. Advances in Artificial Intelligence and Machine Learning, 2024, 4(1): 111, doi: 10.54364/AAIML.2024.41111.
[36] Fomicheva M, Sun S, Yankovskaya L, et al. Unsupervised quality estimation for neural machine translation[J]. Translations of the Association for Computational Linguistics,2020,8(1):539-555.
[37] DeVries T, Taylor G W. Learning confidence for out-of-distribution detection in neural networks[J]. arxiv preprint arxiv:1802.04865, 2018.
[38] Lu Y, Zeng J, Zhang J, et al. Learning confidence for transformer-based neural machine translation[C]//Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics, 2022:2353-2364.
[39] Steinberger R, Pouliquen B, Widiger A, et al. The JRC-acquis: a multilingual aligned parallel corpus with 20 + languages[J]. arxiv preprint cs/0609058,2006.
[40] Ott M, Edunov S, Baevski A, et al. Fairseq: a fast, extensible toolkit for sequence modeling[C]//Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics(Demonstrations),2019:48-53.
[41] Zhu W, Zhao Q, Lv Y, et al. knn-box: a unified framework for nearest neighbor generation[C]//Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics,2024:10-17.