面向领域内翻译的神经机器翻译最佳微调：词汇表与字节对编码设置的系统分析

Javad Pourmostafa Roshan Sharami、Dimitar Shterionov 和 Pieter Spronck蒂尔堡大学认知科学与人工智能系，荷兰蒂尔堡{j.pourmostafa,d.shterionov,p.spronck}@tilburguniversity.edu

arXiv:2303.00722v1 [cs.CL] 1 Mar 2023

摘要

神经机器翻译（NMT）模型的效果很大程度上取决于训练时使用的词汇表；词汇表过小可能导致词汇表外问题，过大则会造成内存问题。子词（SW）分词已成功用于缓解这些问题。词汇表和子词分词的选择会显著影响神经机器翻译模型的训练和微调。微调是根据新数据优化机器翻译模型的常见做法。然而，新数据可能会引入新词（或词元），若未将其纳入考量，可能导致性能不佳。此外，新数据中的词元分布可能与原始数据的分布不同。因此，原有的子词分词模型可能不太适用于新数据。

通过系统的实证评估，本文比较了不同的子词标记化和词汇表生成策略，最终旨在找出微调特定领域模型的最优设置。此外，我们开发了多个（领域内）模型，其中性能最佳的模型较基线提升了6个BLEU分数。

1 引言

微调是根据新数据优化机器翻译模型的常见做法。它可以是——在领域适应的情境下，为特定领域调整现有模型（该领域不同于模型最初接受训练的领域）（Dakwale 和 Monz，2017；Wang 等，2019；Mahdieh 等人，2020）；也可以只是为了提升模型性能（Wang 等，2017）。无论微调的目标是什么，新引入的数据都会带来新的信息。例如，新数据可能包含模型在先前训练周期中未见过的新词；针对新数据的词语切分也可能不够理想（Lim 等，2018；Yeung，2019；Sato 等人，2020）。若未妥善处理，这些新信息可能无法对机器翻译系统产生预期效果。例如，如果新词未纳入词汇表，就会造成词汇表不匹配，进而引发未登录词（OOV）问题。

本文对采用不同子词和词汇表生成设置组合训练的多个模型进行了实证评估，旨在确定用于微调机器翻译引擎的最佳设置。也就是说，我们旨在研究特定领域模型的哪些微调条件（或设置）能够带来最佳性能。我们的研究重点是通过微调来提升引擎的翻译质量，而不是进行领域自适应。通过这项研究，我们旨在回答以下主要研究问题：

RQ1 给定一个机器翻译模型和一个微调数据集，子词生成方法与词汇表的最佳组合是什么？

此外，我们还探讨两个次要研究问题：

© 2022 作者。本文根据知识共享3.0许可协议授权，不得进行演绎创作，需注明出处，CC-BY-ND。

RQ2 微调，即先在一个数据集上训练机器翻译系统，再在另一个数据集上训练，与一次性使用所有数据训练机器翻译系统相比，效果如何?

RQ3：与一次性使用全部数据训练机器翻译系统相比，微调能缩短多少时间（如果能够缩短的话）?

为回答这些问题，我们使用两个数据集（∼0.9M 和 ∼1.1M 个平行句）进行训练，并使用一个数据集（∼179k 个平行句）对多个机器翻译系统进行微调。这样做是为了研究：对于一个使用足量数据（约 $1M$ 个句子对）训练的模型，再用额外数据进行微调，能够在多大程度上提升其性能；而这些额外数据单独来看又不足以训练一个模型。

每种微调方案都采用不同的设置来创建子词和词汇表。对于微调过程，我们提出了一种基于可用数据寻找最佳微调设置的方法。例如，在我们的案例研究中，我们可以获得两个模型（初始模型和微调模型）的数据；然而，正如（Freitag 和 Al-Onaizan，2016年；Dakwale 和 Monz，2017）中所讨论的，初始模型通常已部署在应用程序中，因此在实际生产时可能无法获得这些数据。因此，制定一套基于可用数据的指南至关重要，以确定如何最佳地生成子词和词汇表。在本研究中，我们使用字节对编码（Sennrich 等，2016a）作为子词单元。

我们还使用所有可用数据一次性训练模型，以评估微调是否有任何益处。我们也评估了先在小型领域内数据集上训练、再在较大数据集上微调的机器翻译系统。这样做是为了检验这一假设：先在小型、聚焦的数据集上训练系统，再在较大数据集上微调，其效果不如反过来的做法。[1]

然而，值得注意的是，本研究旨在探讨最佳的微调设置，而不是寻找最佳模型。也就是说，如果我们从数据集 A 开始（无论它属于领域内、领域外、经过筛选还是其他类型），然后在数据集 B 上进行微调，那么在字节对编码和词汇表方面应考虑哪些因素?

本文结构如下。首先介绍我们在实验中使用的数据。第 3 节阐述我们的决策点。第 4 节介绍我们的实证实验，包括数据、子词、词汇表、系统规格、基线和结果的详细信息。第 5 节针对研究问题（ RQs ）进行分析。我们将在第 6 节讨论相关工作，并在第 7 节总结全文。

2 数据

在本研究中，我们使用了两个数据集：（i）两个选定的领域内数据语料库，其中的句子对是从域外语料库中选出的，即按照（Pourmostafa Roshan Sharami 等人，2022）提出的数据选择方法，从所收集的 WMT 语料库[2] 的∼31M 个句子中选出的 ∼0.9M 和 ∼1.1M；以及（ii）一个小型（∼179K 个平行句）原始领域内数据集。

选定的领域内数据集用于训练初始模型（即微调之前）的选定领域内数据，已在（Pourmostafa Roshan Sharami 等人，2022）中介绍。我们特别使用了 $Top5$ 和 $Top6$ ，因为它们在该研究中取得了最佳翻译性能。选定的领域内数据是根据领域外句子与某个领域内数据集的相似度对其排序后得到的。语言对为英法语。此外，正如该研究所指出的，数据量足以训练出翻译质量较高的机器翻译系统。

原始领域内数据集 我们所使用的原始领域内数据是国际口语语言翻译研讨会（IWSLT）2014 语料库（Cettolo 等人，2014）。该语料库由 TED 演讲组成。为了在训练期间评估模型并了解模型的表现，我们分别使用了一个开发集（dev2010）和两个测试集（test2010 和 test2011）。IWSLT 2014 和 WMT 常用于领域适应（DA）研究中的领域内数据集（Axelrod 等人，2011；Luong 和 Manning，2015；Chen 等人，2016；Wang 等人，2017；Pourmostafa Roshan Sharami 等人，2022），这有助于提高研究的可复现性。

[1]这一假设已在其他领域得到证实，例如机器人控制系统领域（Spronck 等人，2008）。

[2]http://statmt.org/wmt15/translation-task.html

表1展示了我们在实验中使用的数据统计信息。

数据类型	Name	句子
选定的领域内数据	Top5	895k
	Top6	∼1M
原始领域内数据（IWSLT 2014）	TED训练	179K
	TED dev2010	887
	TED test2010	1664
	TED test2011	818

表1： 领域内数据集（领域内和领域外）以及领域外数据集的摘要。

3 决策点

如第 1 节所述，本研究旨在确定在子词 和词汇方面对机器翻译系统进行微调的最有效方法。考虑一个在数据集 D 上训练的模型 M，该数据集能够代表某个领域 d，以及一个微调数据集 E。需要做出以下决策点：

可用数据： 选择应使用 E 中的哪个微调数据集，或使用其中哪些数据集的组合。正如以往研究所示，使用所有可用数据并不总是有益，因为它未必能提升整体性能（尤其是在特定领域），同时还会增加计算开销（Wang 等，2019年；Soto 等，2020；PourmostafaRoshanSharamiet al.,2022）。

子词： 选择一种模型来构建子词单元。可以选择 (i) 在集合 D 上训练的字节对编码模型，因此该模型也用于模型 M （D_{BPE}）的训练；(ii) 在选定的微调数据集（E_{BPE}）上训练新的字节对编码模型；或 (iii) 在 D 和 E 的拼接数据（$D+E_{BPE}$）上训练新的字节对编码模型。这样做主要是因为微调数据中可能包含未出现在集合 D 中的新词和独特词，因此原始字节对编码模型可能并非最优。如果我们将现有模型调整到不同于 d 的领域，就可能出现这种情况。不过，比较 (ii) 和 (iii) 可知，原始模型的训练数据可能无法获取，因此只有 (ii) 是可行的选择。

词汇表： 选择词汇表，即：（i）使用原始模型的词汇表 M ($|D|$)，（ii）使用微调集中的词元 E ($|D+E|$) 扩展该词汇表，或（iii）根据 E ($|E|$)创建新的词汇表。这一点非常重要，因为相关的词汇集是机器翻译性能的支柱。因此，找到这样的词汇集可以减轻未登录词和罕见词的影响。

这两个因素——子词和词汇表——需要结合起来考虑，因为它们都会显著影响机器翻译的性能。为此，我们面临一个涉及两个维度的优化问题。由于每个维度上的不同选项彼此独立，[3] 因此可以将这些选项组合起来，枚举所有可能的解决方案。由此得到 9 种组合：将原机器翻译系统的 BPE模型 D_{BPE} 应用于微调 E 数据，并使用不同的词汇表选项 $|D|$、$|D+E|$和 $|E|$训练三个系统；在微调数据上训练新的 BPE模型， E_{BPE} 再将其应用于数据，生成三个不同的词汇表（$|D|$、 $|D+E|$和 $|E|$）。

遵循这些决策点，对于给定的微调数据集，我们可以考虑三种 BPE模型。使用这些模型，我们可以（i）创建词汇表来源；以及（ii）创建用于微调的训练集。通常，这两个过程相互关联，也就是说，一旦学习并应用了 BPE模型，词汇表就是（经过处理的）数据中出现的子词单元集合。不过，这并非硬性约束。也就是说，我们可以使用一个词汇表，它源自采用与训练或微调数据不同的 BPE模型处理的数据。例如，数据集 E 可以用 BPEE_{BPE}处理，但训练所用的词汇表仍然可以基于通过应用 D_{BPE}得到的 D。

4 实验

为了找到微调领域内模型的最佳设置，我们遵循第 3 节中的决策点，使用第 2 节介绍的英法数据开展实验，并使用 BLEU（Papineni 等，2002）、TER（Snover 等，2006）和 chrF（Popovi´c，2015）对微调模型进行相互比较，并与不同的基线进行比较。

图 1 展示了我们的实验方法。

根据给定的数据和我们的任务，即使用 $Top5$ 或 $Top6$ （$D=$

[3]值得注意的是，我们知道各类组合中的词汇表集合都依赖于 BPE 模型；在此，我们仅将“组合”定义为彼此独立。

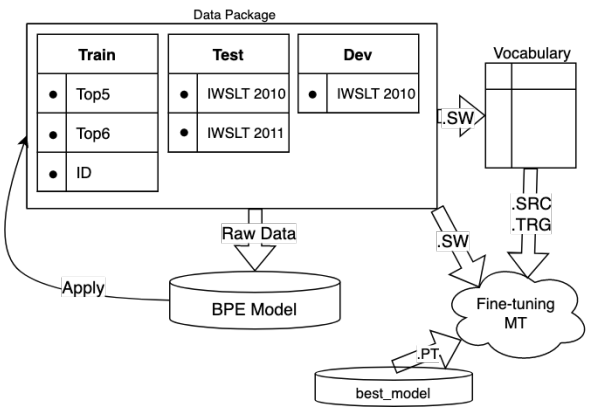

图 1： 机器翻译微调概览。数据包展示了我们用于实验的数据集。这些数据集可用于训练初始模型，或微调已训练的模型。

$Top5$ 或 $D = Top6$）训练的模型，并使用原始领域内数据（E =领域内数据）进行微调，我们共有54种方案，每个训练好的模型对应27种。词汇表的数据来源有3种选择（$Top5$ 或 $Top6$、ID 或它们的组合），构建BPE也有3种选择；这会影响用于构建词汇表的数据的切分，也会影响微调数据的切分。[4]也就是说，有3种方案可用于构建BPE模型，以切分用于创建词汇表的数据；另有3种方案可用于构建BPE模型，以切分微调数据。

值得注意的是，字节对编码的合并操作次数为 50K，并且针对每种源语言和目标语言分别创建了一个 BPE 模型。

图 2 概述了不同的选项。

基于这54种选项，我们定义了三类实验。第一类实验中，BPE模型基于原始数据 $D = Top5/6$ 或微调数据 $E = ID$构建（但不基于二者的组合构建），词汇表也从相同的数据中生成。通过这类实验，我们研究以下（假设的）场景：要么无法获取原始数据集（D）和词汇表（$|D|$），因此只能使用微调数据集（E）；要么可以获取这些资源，并且能够直接利用它们，而无需花费时间或资源处理微调数据以提取 $|E|$。第二类实验中，词汇表基于 D 和 E两者构建（$Top5/6 + ID$）。这将被视为一种非常有利的场景，在该场景中，D 和 E 均可用，并且可以联合利用。在这一假设下，我们还在 $D+E$上构建基线模型（见第4.2节）。第三类实验中，用于分割构建词汇表所依据数据的BPE，与用于分割微调数据的BPE不同。这些实验涵盖以下（假设的）场景：词汇表已给定，但它与微调数据的子词生成方式并不完全对应。通过这第三类实验，我们希望了解在受限条件下是否仍能达到足够的质量。

为了减少计算时间和资源消耗，我们实施了 27 组实验中的 11 组。对于第二类实验，我们排除了词汇表和微调数据使用不同 BPE 模型的情况。这是因为，鉴于 D 和 E 都采用混合 BPE 模型，这样做既没有必要，也不切实际。基于同样的理由，我们排除了第 1 类和第 3 类实验中使用两个 BPE 模型的情况——一个根据 $D+E$ 训练，另一个则根据 D 或 E 训练。这样，我们最终选出了表3中列出的 11 项实验。

4.1 神经机器翻译系统描述

我们使用 OpenNMT-py[5] 框架（Klein 等，2017）训练并微调神经机器翻译模型。我们对 Transformer 模型（Vaswani 等，2017）进行了微调，最多训练 200K 步；每隔 1000 步保存并验证一次中间模型，直至收敛。我们设置了提前停止条件：若连续 10 次验证均未见改进，则停止微调过程。由于我们微调的模型由（Pourmostafa Roshan Sharami 等，2022）提出，因此我们保持神经机器翻译设置一致，并使用相同的超参数。为高效运行所有神经机器翻译系统，并使其与研究问题相契合，我们还采用了 OpenNMT-py 社区建议的其他超参数，以模拟谷歌的默认设置（Vaswani 等，2017）。

微调时，我们将训练任务分配到三块 NVIDIA Tesla V100 GPU 上。我们使用字节对编码（BPE）算法（Sennrich 等，2016年a）将所有数据编码为子词单元序列。BPE 合并操作的数量

[4]正如第 3 节末尾所述，在微调测试案例中，我们并未严格禁止训练数据和词汇表采用不同的词语切分方式。

[5]https://opennmt.net/OpenNMT-py/

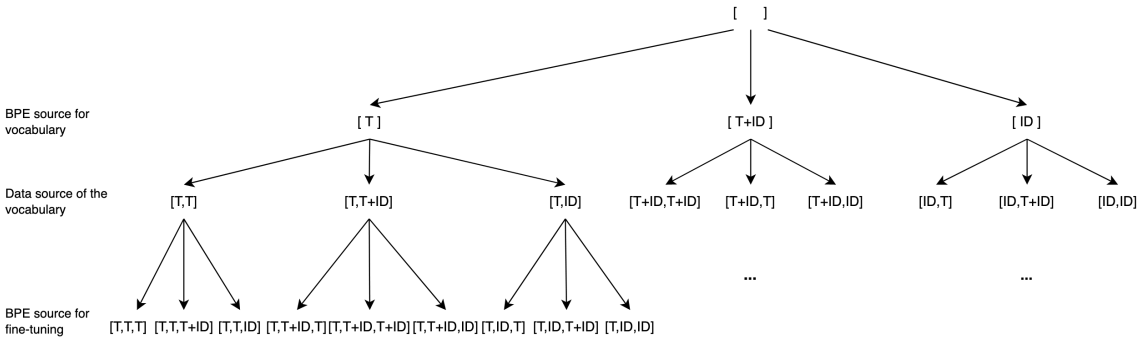

图 2： 树的叶节点展示了我们尝试过的不同组合。为简化报告，我们将 $Top5$ 和 $Top6$ 视为同一份训练数据。也就是说，我们分别尝试了 $Top5$ 和 $Top6$，但没有为它们各自展开树。“T”也是“Top”的缩写。

为 50K，并为每种源语言和目标语言分别创建了一个 BPE 模型。

4.2 基线模型

我们不仅将微调模型相互比较，还将它们与三类不同的基线进行比较：

(1) 我们仅使用原始领域内数据或领域内数据训练神经机器翻译系统（不进行后续训练或微调），以建立以下基线。B1——使用原始领域内数据训练的神经机器翻译模型；B2 和 B3——分别使用选定的领域内数据 Top5 和 Top6 训练的神经机器翻译模型。如前所述，我们没有对任何模型进行微调来建立 B1、B2 和 B3，但将它们用作基线，以衡量在 IWSLT 数据集上微调所带来的提升。

（2）我们考察了几种最先进的微调模型，并将其与我们的模型进行比较，以展示我们的微调过程所产生的影响。这些模型包括 B4（Luong 和 Manning，2015）、B5（Axelrod 等，2011）、B6（Chen 等，2016）和 B7（Wang 等，2017）。值得一提的是，在比较所提出的组合方案之前，我们定义这些基线是为了评估微调过程本身的影响。

（3）我们混合了原始数据集（$Top5$ 或 $Top6$）和领域内数据集（ID），然后训练了一个模型。也就是说，我们没有对这一类别中的任何模型进行微调。这有助于展示并衡量合并操作与微调操作之间的差异。

基线结果如表2所示。

4.3 结果与分析

我们使用不区分大小写的BLEU（Papineni 等，2002）、TER（Snover 等，2006）和 chrF2（Popovi´c,

#	2010年测试集			2011年测试集		
	BLEU↑	TER↓	CHRF2↑	BLEU↑	TER↓	CHRF2↑
B1	31.9	56.6	57.0	38.3	49.7	61.0
B2	30.9	59.1	57.0	36.7	51.5	62.0
B3	31.3	58.3	58.0	36.5	50.9	62.0
B4	32.2	N/A	N/A	35.0	N/A	N/A
B5	32.2	58.3	N/A	35.5	N/A	N/A
B6	30.3	58.3	N/A	33.8	N/A	N/A
B7	32.8	58.3	N/A	36.5	N/A	N/A
B8	31.8	57.3	57.3	37.9	50.2	62.2
B9	32.2	56.8	57.6	38.8	48.7	62.7

表2： 基线模型的结果。B1、B2 和 B3 分别表示使用原始 ID 数据、$Top5$ 和 $Top6$ 训练的神经机器翻译模型。B4，...，B7 表示先前研究中经过微调的模型。B8 和 B9 分别表示使用 (ID, $Top5$) 和 (ID, $Top6$) 混合数据训练的模型。

2015）指标评估微调后的机器翻译系统在两个测试集上的性能，这些指标由 sacre-BLEU 工具包（Post, 2018）实现。我们的结果见表3。我们还分析了不同组合的结果在统计差异方面的表现（见第 5.1 节）。

根据表3，按照 BLEU、TER 和 chrF2 得出的 $Top5$ 和$Top6$ 两种情况下微调组合从最佳到最差的排名为$C3$, $C1$, $C9$, $C11$, $C2$, $C10$, $C4$, $C8$, $C6$, $C7$, $C5$。在我们的实验中， $C3$ 在所有组合中取得了最高的 BLEU 分数；只有两种情况下，其他组合取得了相同的分数：(1) 在 2010 年测试集上翻译 $Top5$ 时， $C1$ ；(2) 在 2010 年和 2011 年测试集上翻译 $Top5$ 时， $C9$ 。

$C3$、 $C1$和 $C9$ 也取得了最高的chrF2分数和最低的TER分数，这表明微调领域内模型的最佳设置可能是

将以下两者结合起来：(i) 根据初始模型的训练数据创建的BPE模型（即用于训练模型的数据——$top5$ 或{v2）；以及 (ii) 根据微调数据创建的词汇表（即 ID）；或者使用初始模型的训练数据，或将两者结合。值得注意的是，使用此设置微调模型并非总是可行，因为我们可能无法获取用于训练原始模型的数据。不过，如果可以获取，建议遵循 $C3$的设置。

下一个有前景的微调设置是 $C11$，它建议将初始模型的训练数据与微调数据结合起来。不过，如果两者都可用，我们更倾向于采用 $C3$，$C1$或 $C9$，因为这些方法可能对翻译质量有更好的提升。

根据评估指标，建议的第五、第六和第七种微调设置分别是 $C2$、$C10$和 $C4$。这些组合表明，可以采用以下方式实现有效的微调流程：使用（i）基于微调数据创建的BPE模型（即ID），以及（ii）基于微调数据（即 ID）、初始模型的训练数据或二者组合创建的词汇集。不过，只有在能够访问用于训练模型的数据时，才可行使用 $C10$ 和$C4$ 对模型进行微调。也就是说，如果唯一可用的数据是用于微调模型的数据，则建议采用 $C2$。

按评估分数从高到低排列的其他建议微调设置为：$C8$、$C6$、$C7$、$C5$。值得一提的是，这些设置同时使用了微调数据（即ID）和初始模型的训练数据，但翻译质量表现不佳。这说明了字节对编码模型和词汇表对微调的重要性。

在分析结果并总结出一套关于数据、字节对编码和词汇表的全面指南，以用于微调模型后，我们将不同组合与三类基线进行了比较，以展示微调本身的有效性，而不受不同微调组合的影响。根据表2和表3总结的结果，所有微调模型的表现都优于基线。例如，与基线 $B1$，$B2$，$B7$ 和相比，C3 的 BLEU 分数分别提高了约 13%（31.9 提高到 36.1）、16.8%（30.9 提高到 36.1）、10%（32.8提高到 36.1）和13.5%（31.8 提高到 36.1），而 $B8$ 分别如此。[7] 这些数据表明，与一次性训练模型相比，微调过程是更好的选择。这些结果引出了一个有趣的问题：训练时应如何向神经网络输入数据，才能实现最佳性能（无论是从翻译质量还是训练时间来看）。

5 讨论

本节首先讨论经过微调的神经机器翻译模型之间评估分数的成对统计显著性。其次，我们将微调的训练时间与基线的训练时间进行比较。第三，我们展示初始模型的选择在多大程度上会影响翻译性能。

5.1 统计显著性检验

我们使用自助重采样和95%置信区间，基于1000次迭代和300个句子样本，计算了表3所示结果在两个测试集（2010年和2011年）上的 BLEU 分数的成对统计显著性。根据检验结果，除表4所列的系统对外，大多数微调模型之间的差异均具有统计显著性。

除第 4.3 节所述的分析外，这还说明了两点：(1) 如果无法同时获取初始模型和微调模型的数据，仅使用初始数据也能达到相当接近的性能。例如，在 2010年测试集上，按照 $C1$ 和 $C3$ 对初始模型 –$Top5$– 进行微调后，其 BLEU 分数没有差异。(2) 使用一个数据集而非两个不同的数据集来创建字节对编码模型和词汇表，并不总是会对模型性能产生显著影响。例如，$C2$ 使用单一数据集（ID）进行训练，而 $C4$ 同时采用了 ID 和 Top5，但仍然取得了相同的性能（在 2010年测试集上）。

5.2 训练时间

在表5中，我们展示了训练基线（$B1$、$B2$、$B3$、$B8$ 和 $B9$）以及对最佳模型（$C3$）进行微调的运行时间（RT），适用于 $Top5$ 和 $Top6$。[8] 微调时间约为

[6]Top5←标识符，并使用2010年测试集进行评估

[7]我们选择 B7 作为第二类基线的代表，因为它在微调基线中取得了最高的 BLEU 分数。[8]所有模型以及 SW 和词汇表组合的微调大约耗时 1 小时 30 分钟。因此，我们将讨论范围限定为 $C3$。

#	X	Y	Z	模型											
				顶部5 ← 标识符						顶部6 ← 标识符					
				2010年测试集			2011年测试集			2010年测试集			2011年测试集		
				BLEU↑	TER↓	chrF2↑	BLEU↑	TER↓	chrF2↑	BLEU↑	TER↓	chrF2↑	BLEU↑	TER↓	chrF2
C1	Top5/6	Top5/6	Top5/6	36.1	52.4	60.3	43.7	44.1	66.1	36.4	52.3	60.4	44.1	43.7	66.4
C2	ID	ID	ID	35.9	52.5	60	42.8	44.9	65.2	36.1	52.5	60.0	44.0	43.7	65.8
C3	Top5/6	ID	Top5/6	36.1	52.4	60.4	44.0	43.6	66.6	36.4	52.2	60.4	44.4	43.5	66.4
C4	ID	Top5/6	ID	35.5	52.8	59.7	43.3	44.5	65.6	35.5	53.0	59.6	43.6	44.1	65.7
C5	Top5/6	Top5/6	ID	33.4	53.8	58.6	40.7	45.4	64.2	33.6	53.3	58.7	40.9	45.5	64.2
C6	ID	Top5/6	Top5/6	35.4	53.0	59.7	43.5	44.6	65.5	35.5	53.6	59.9	43.3	44.4	65.7
C7	Top5/6	ID	ID	33.4	53.3	58.6	40.9	45.1	64.2	33.2	53.4	58.2	40.2	45.4	64.0
C8	ID	ID	Top5/6	35.4	52.8	59.6	42.8	44.6	65.1	35.1	53.1	59.6	43.3	44.4	65.7
C9	Top5/6	Top5/6+标识符	Top5/6	36.1	52.4	60.1	44.0	43.6	66.4	35.9	52.6	60.4	44.3	43.5	66.6
C10	ID	Top5/6+标识符	ID	35.6	52.7	59.9	43.0	44.9	65.2	36.1	52.3	60.2	44.0	43.6	66.0
C11	Top5/6+标识符	Top5/6+标识符	Top5/6+标识符	36.0	52.5	60.1	44.1	43.6	65.9	36.4	52.3	60.3	44.0	43.8	66.4

表3： 微调神经机器翻译系统的评估分数。X 表示用于创建词汇表的 BPE 模型的来源；Y 表示词汇集的来源，Z 表示用于创建微调数据的 BPE 模型的来源。$TopN \leftarrow ID$ 表示在 $TopN$ 上训练的模型在 ID 数据上进行微调。

	2010年测试集	2011年测试集
Top5	(C1, C3) (C6, C8) (C2, C4) (C5, C7)	(C1, C6) (C4, C5)
Top6	(C1, C3) (C6, C8)	(C1, C6) (C4, C5)

表4： 不具统计显著性的系统对结果（对于 $p < 0.05$）。（CX, CY）表示采用组合 CX 和 CY 中建议的设置进行微调的模型；根据 300 个样本，这些模型之间不存在统计显著差异。

#	完整运行时间 天:时:分	Step	最佳模型运行时间 天:时:分	Step
B1	00:03:53	18,000	00:00:50	5,000
B2	00:10:33	35,000	00:05:50	20,000
B3	00:08:20	35,000	00:04:26	19,000
B8	00:05:47	23,000	00:03:16	13,000
B9	00:06:17	25,000	00:04:45	15,000
C3-Top5	00:01:33	12,000	00:00:17	2,000
C3-Top6	00:01:32	12,000	00:00:16	2,000

表5： 训练和微调的运行时间（RT）。前几个基线 B1、B2 和 B3 分别在原始领域内数据、 $Top5$ 和 Top6 上训练；基线 B8 和 B9 分别是 ID 与 Top5、ID 与 Top6 的组合。$C3 - Top5$ 和 $C3 - Top6$ 表示使用在 $Top5$ 和 $Top6$ 上训练的模型的最佳组合（$C3$）进行微调。

1小时30分钟，而使用全部数据一次性训练机器翻译系统（$B8$ 或$B9$）则约需5至6小时。这表明，微调所需时间仅占一次性使用全部数据训练模型所需时间的一小部分（27%）。一方面，与训练和微调时间之和相比——即先训练 B2 或 B3，再进行 C3，分别耗时12小时6分钟和9小时52分钟——从头开始训练模型更可取。另一方面，从头开始训练无法达到微调所能取得的相同性能（如第4.3节所述）。

5.3 反向微调

我们还评估了在本案例研究（$Top5/6$）中用于微调的初始模型是否有效，或者是否需要调整训练数据的呈现顺序。也就是说，先使用原始领域内数据训练机器翻译系统，再对其进行微调——使其适应选定的领域内数据。值得注意的是，我们开展这项实验，旨在监测性能以及微调过程对初始模型的敏感性。

根据表6，所有微调模型的表现都优于反向微调模型。这意味着，从选定的领域内模型（$Top5/6$）开始，比从原始领域内模型开始更加高效。这也表明，在大规模数据（如 $Top5/6$）上训练可以避免在小规模领域内数据（如 ID）上过拟合。向新参数的这种转换可能会导致模型在初始数据中的测试样本上的性能下降（Li 和 Hoiem，2016年；Dakwale 和 Monz，2017）。然而，

这在我们的案例研究中并不完全成立，因为整个研究都针对同一个领域。也就是说，初始数据和微调数据都来自同一个特定领域，只是前者的规模相对较大。未来，我们计划拓展到更多领域，并评估不同数据量和领域特异性所产生的影响。

#	2010年测试集			2011年测试集		
	BLEU↑	TER↓	CHRF2↑	BLEU↑	TER↓	CHRF2↑
$ID \leftarrow Top5$	30.8	59.4	56.8	36.3	52.3	61.1
$ID \leftarrow Top6$	31.5	58.2	57.3	37.8	50.3	61.9

表6： 反向微调的结果。 $ID \leftarrow TopN$ 表示在 I 上训练的模型在$TopN$ 数据上进行了微调。

6 相关工作

关于机器翻译微调这一主题已有大量研究。大多数先前研究考察了如何使模型适应不同的领域。即，它们首先采用大规模领域外数据，然后在小规模领域内数据上进行微调。其中，Luong 等人，2015 首次取得了成功：他们先在英语-德语通用领域数据上训练模型，然后在相同语言的新领域数据（会话领域）上进行微调。他们声称，与未经进一步训练的原始模型相比，BLEU分数提高了3.8分（从25.6提高至29.4）。

Freitag 和 Al-Onaizan（2016）提出了另一种方法，旨在提升新领域中的翻译性能，同时不降低通用领域测试集上的性能。为此，他们将微调后的模型与已训练好的基线模型进行集成，并在 IWSLT 2015 评测活动中评估了该方法（Cettolo 等人，2015）。作者报告称，该方法在领域内测试集上提升了 7.2 个 BLEU 分数，同时仍保持了领域外测试集上的性能。

随后，Freitag 和 Al-Onaizan，2016 证明，虽然用于微调的集成方法看起来是个不错的选择，但微调模型在通用领域任务上的性能仍会下降，尤其是在涉及特定领域语境（例如医学和法律领域）时。这主要是因为主题或体裁层面的领域内数据集（van der Wees 等人，2015）包含了不同于通用数据的新词汇和语言特征（Koehn 和 Knowles，2017）。为了解决这一问题，他们提出了一种基于知识蒸馏（Hinton 等人，2015）的微调方法。

此外，还有其他研究致力于解决微调过程中的词汇不匹配问题。例如，Sato 等人（2020）提出了一种将初始模型的嵌入层适配到目标领域的方法。他们将从目标领域单语数据中获得的通用词嵌入投射到源领域嵌入上，从而实现这一适配。作者报告称，在英语→日语和德语→英语翻译中，BLEU 分数分别提升了 3.86 和 3.28 分。

由于词语切分和词汇生成会极大地影响机器翻译性能（Ataman 和 Federico，2018），他们对词语子词级切分这一问题进行了广泛评估，并比较了两种用于神经机器翻译的词语切分方法：字节对编码（BPE）（Sennrich 等，2016b）和语言学动机词汇缩减（LMVR）（Ataman 等，2017）。他们在五种形态丰富的语言中比较了这些方法，并报告称，LMVR 在测试语言中取得了更好的性能。

7 结论与未来工作

本文基于一个常用的领域专用数据集（IWSLT 2014）进行了系统分析，旨在为神经机器翻译模型微调找到子词生成方法与词汇表的最佳组合。除了比较 22 个模型（两个训练集各有 11 种方案）的性能外，我们还研究了与使用全部数据一次性训练机器翻译模型相比，微调对训练时间和机器翻译性能的影响。通过实证评估，我们创建了一个用于领域内翻译的最先进模型，可应用于多种情境，包括多语言领域自适应（Cooper Stickland 等，2021）。

考虑到可用数据，我们提出了微调前需要确定的两个决策点（即子词和词汇表），以及第三个决策点：训练/微调数据和词汇表是否采用相同的 BPE 源。我们的实验勾勒出一条路线图，列出微调领域内模型的三种可能方案：(1) 如果初始模型的数据（D）和微调数据（E）都可用，

则分别使用 D 和 E 训练 BPE 模型并创建词汇表可能会很有效。(ii) 否则，可以基于 D 创建 BPE 模型和词汇表；(iii) 如果最后两种方案都不可行，则建议所有决策点均基于 E。值得注意的是，所有微调策略都提升了初始机器翻译模型的基线，最高提升了 6 个 BLEU分数（从 30.9 提高到 36.1）。

在未来的工作中，我们计划通过在领域外语料库上进一步训练，提升预训练（领域内）模型的泛化能力，从而使这些模型既能翻译通用输入，也能处理其专业领域内容，同时不遗忘已经学到的知识。另一项研究方向是考察所提出的决策点在其他语言对和领域中的表现。

数据和微调模型可在以下地址获取：https://github.com/JoyeBright/FT-IWSLT2014-BPEVocab。

参考文献

[Ataman 和 Federico2018] Ataman, Duygu 和 Marcello Federico。2018。对两种神经机器翻译词汇表缩减方法的评估。载于第*13*届美洲机器翻译协会会议论文集（第*1*卷：研究专题），第97–110页，波士顿，马萨诸塞州，3月。美洲机器翻译协会。

[Ataman 等人2017] Ataman, Duygu、Matteo Negri、Marco Turchi 和 Marcello Federico。2017。土耳其语到英语神经机器翻译中基于语言学的词汇表缩减。《布拉格数学语言学通报》，108，06。

[Axelrod 等人2011] Axelrod, Amittai, Xiaodong He 和 Jianfeng Gao。2011年。通过伪域内数据选择进行领域适应。载于*2011*年自然语言处理实证方法会议论文集，第355–362页，英国苏格兰爱丁堡，7月。计算语言学协会。

[Cettolo 等人2014] Cettolo, Mauro, Jan Niehues, Sebastian St¨uker, Luisa Bentivogli 和 Marcello Federico。2014年。第11届IWSLT评测活动报告。载于第*11*届国际口语翻译研讨会：评测活动论文集，第2–17页，加利福尼亚州太浩湖，十二月4日至5日。

[Cettolo等人2015] Cettolo, Mauro, Jan Niehues, Se- bastian St¨uker, Luisa Bentivogli, Roldano Cattoni 和 Marcello Federico。2015。《IWSLT 2015 评测活动》。载于第*12*届国际口语翻译研讨会论文集：评测活动，第2–14页，越南岘港，12月3日至4日。

[Chen 等人2016] Chen, Boxing, Roland Kuhn, George Foster, Colin Cherry 和 Fei Huang。2016年。用于机器翻译自适应训练数据选择的双语方法。载于美洲机器翻译协会会议：机器翻译研究者专题，第93–106页，美国得克萨斯州奥斯汀，10月28日至11月1日。美洲机器翻译协会。

[Cooper Stickland 等人2021] Cooper Stickland、Asa、Alexandre Berard 和 Vassilina Nikoulina。2021年。神经机器翻译的多语言领域自适应：通过适配器解耦语言信息与领域信息。载于第六届机器翻译大会论文集，第578–598页，线上举行，十一月。计算语言学协会。

[Dakwale 和 Monz2017] Praveen Dakwale 和 Christof Monz。2017。针对神经机器翻译的微调：在领域内和领域外数据上将性能退化降至最低。载于第*16*届机器翻译峰会（*MT-Summit 2017*）论文集，第156–169页。

[Freitag 和 Al-Onaizan2016] Markus Freitag 和 Yaser Al-Onaizan。2016年。神经机器翻译的快速领域适应。

[Hinton 等人2015] Geoffrey Hinton、Oriol Vinyals 和 Jeff Dean。2015。神经网络中的知识蒸馏。

[Kleinetal.2017] Klein, Guillaume, Yoon Kim, Yun- tian Deng, Jean Senellart 和 Alexander Rush。2017。OpenNMT：用于神经机器翻译的开源工具包。载于*ACL 2017* 系统演示论文集，第 67–72 页，温哥华，加拿大，7月。计算语言学协会。

[Koehn and Knowles2017] Koehn, Philipp 和 Rebecca Knowles。2017。神经机器翻译的六大挑战。载于第一届神经机器翻译研讨会论文集，第 28–39 页，温哥华，8月。计算语言学协会。

[Li 和 Hoiem2016] Li, Zhizhong 和 Derek Hoiem。2016年。《无遗忘学习》。收录于 Leibe、Bas- tian、Jiri Matas、Nicu Sebe 和 Max Welling 编，计算机视觉——第十四届欧洲计算机视觉会议，*ECCV 2016*，论文集，《计算机科学讲义》（包括《人工智能讲义》和《生物信息学讲义》子系列），第 614–629 页，德国。施普林格。资助信息：本研究部分由美国国家科学基金会奖项 14-46765、10-53768 以及美国海军研究办公室多学科大学研究计划 N000014-16-1-2007 资助。出版商版权：© 施普林格国际出版股份公司 2016年；第14届

欧洲计算机视觉会议，ECCV 2016；会议日期：2016年11月10日至2016年10月14日。

[Lim 等人2018] Lim、Robert、Kenneth Heafield、Hieu Hoang、Mark Briers 和 Allen Malony。2018。探索 GPU 架构上神经机器翻译的超参数优化。

[Luong 和 Manning2015] Luong、Minh-Thang 和 Christopher Manning。2015。斯坦福大学面向口语领域的神经机器翻译系统。载于第十二届国际口语翻译研讨会论文集：评测活动，第 76–79 页，越南岘港，12月3日至4日。

[Luong 等人2015] Luong、Thang、Hieu Pham 和 Christopher D. Manning。2015。基于注意力的神经机器翻译的有效方法。载于*2015*年自然语言处理经验方法会议论文集，第 1412–1421 页，葡萄牙里斯本，九月。计算语言学协会。

[Mahdieh 等人2020] Mahdieh, Mahdis, Mia Xu Chen, Yuan Cao 和 Orhan Firat。2020 年。利用单语数据快速实现机器翻译的领域适应。

[Papineni 等人2002] Papineni, Kishore, Salim Roukos, Todd Ward 和 Wei-Jing Zhu。2002 年。BLEU：一种自动机器翻译评估方法。载于计算语言学协会第 *40* 届年会论文集（*ACL 2002*），第 311–318 页，美国宾夕法尼亚州费城，7月。计算语言学协会。

[Popovi´c2015] Popovi´c, Maja。2015 年。chrF：用于自动机器翻译评估的字符 n 元语法 F 分数。载于第十届统计机器翻译研讨会论文集，第 392–395 页，葡萄牙里斯本，9月。计算语言学协会。

[Post2018] Post，Matt。2018年。呼吁明确报告 BLEU 分数。载于第三届机器翻译会议论文集：研究论文，第186–191页，比利时布鲁塞尔，10月。计算语言学协会。

[Pourmostafa Roshan Sharami 等人2022] Pourmostafa Roshan Sharami，Javad、Dimitar Sterionov 和 Pieter Spronck。2022年。利用单语文本为神经机器翻译选择领域内平行句。《荷兰计算语言学杂志》，11：213–230，2月。

[Sato 等人2020] Sato、Shoetsu、Jin Sakuma、Naoki Yoshinaga、Masashi Toyoda 和 Masaru Kitsure- gawa。2020 年。神经机器翻译领域适应中的词汇适应。载于计算语言学协会论文集：*EMNLP 2020*，第 4269–4279 页，线上，十一月。计算语言学协会。

[Sennrich 等人2016a] Sennrich，Rico、Barry Haddow 和 Alexandra Birch。2016a。通过侧约束控制神经机器翻译中的礼貌。载于*2016*年北美计算语言学协会会议论文集：人类语言技术，第35–40页，加利福尼亚州圣迭戈，6月。计算语言学协会。

[Sennrich 等，2016b] Sennrich、Rico、Barry Haddow 和 Alexandra Birch。2016b。使用子词单元对低频词进行神经机器翻译。载于第*54*届计算语言学协会年会论文集（第*1*卷：长篇论文），第1715–1725页，德国柏林，8月。计算语言学协会。

[Snover 等2006] Snover、Matthew、Bonnie Dorr、Rich Schwartz、Linnea Micciulla 和 John Makhoul。2006。针对性人工标注下的翻译编辑率研究。载于美洲机器翻译协会第*7*届会议论文集：技术论文，第223–231页，美国马萨诸塞州剑桥，8月8日至12日。美洲机器翻译协会。

[Soto 等2020] Soto、Xabier、Dimitar Shterionov、Alberto Poncelas 和 Andy Way。2020。从多个来源选择回译数据，以改进神经机器翻译。载于第*58*届计算语言学协会年会论文集，第3898–3908页，线上，7月。计算语言学协会。

[Spronck 等人2008] Spronck、Pieter、Ida Sprinkhuizen-Kuyper 和 Eric Postma。2008年。Deca：一种由多巴驱动的进化控制算法。应用人工智能，22(3):169–197，3月。

[van der Wees 等人2015] van der Wees、Marlies、Arianna Bisazza、Wouter Weerkamp 和 Christof Monz。2015年。领域中有什么？分析统计机器翻译中的体裁和主题差异。载于计算语言学协会第*53*届年会暨第*7*届国际自然语言处理联合会议论文集（第*2*卷：短论文），第560–566页，中国北京，7月。计算语言学协会。

[Vaswani 等人2017] Vaswani、Ashish、Noam Shazeer、Niki Parmar、Jakob Uszkoreit、Llion Jones、Aidan N Gomez、Ł ukasz Kaiser 和 Illia Polosukhin。2017。Attention is all you need。载于 Guyon、I.、U. V. Luxburg、S. Bengio、H. Wallach、R. Fergus、S. Vishwanathan 和 R. Garnett（编），神经信息处理系统进展，第 30 卷。Curran Associates, Inc.

[Wang 等人2017] Wang、Rui、Andrew Finch、Masao Utiyama 和 Eiichiro Sumita。2017。用于神经机器翻译领域自适应的句子嵌入。载于计算语言学协会第 *55* 届年会论文集（第 *2* 卷：短篇论文），第 560–566 页，温哥华，加拿大，7月。计算语言学协会。

[Wang等人2019] Wang、Xu、Chunyang Chen 和 Zhenchang Xing。2019年。利用循环神经网络进行软件本地化的领域特定机器翻译。《实证软件工程》，24(6):3514–3545，十二月。

[Yeung2019] Yeung，Chin Man。2019年。插入领域词汇并微调 BERT 对德国法律语言的影响，十一月。