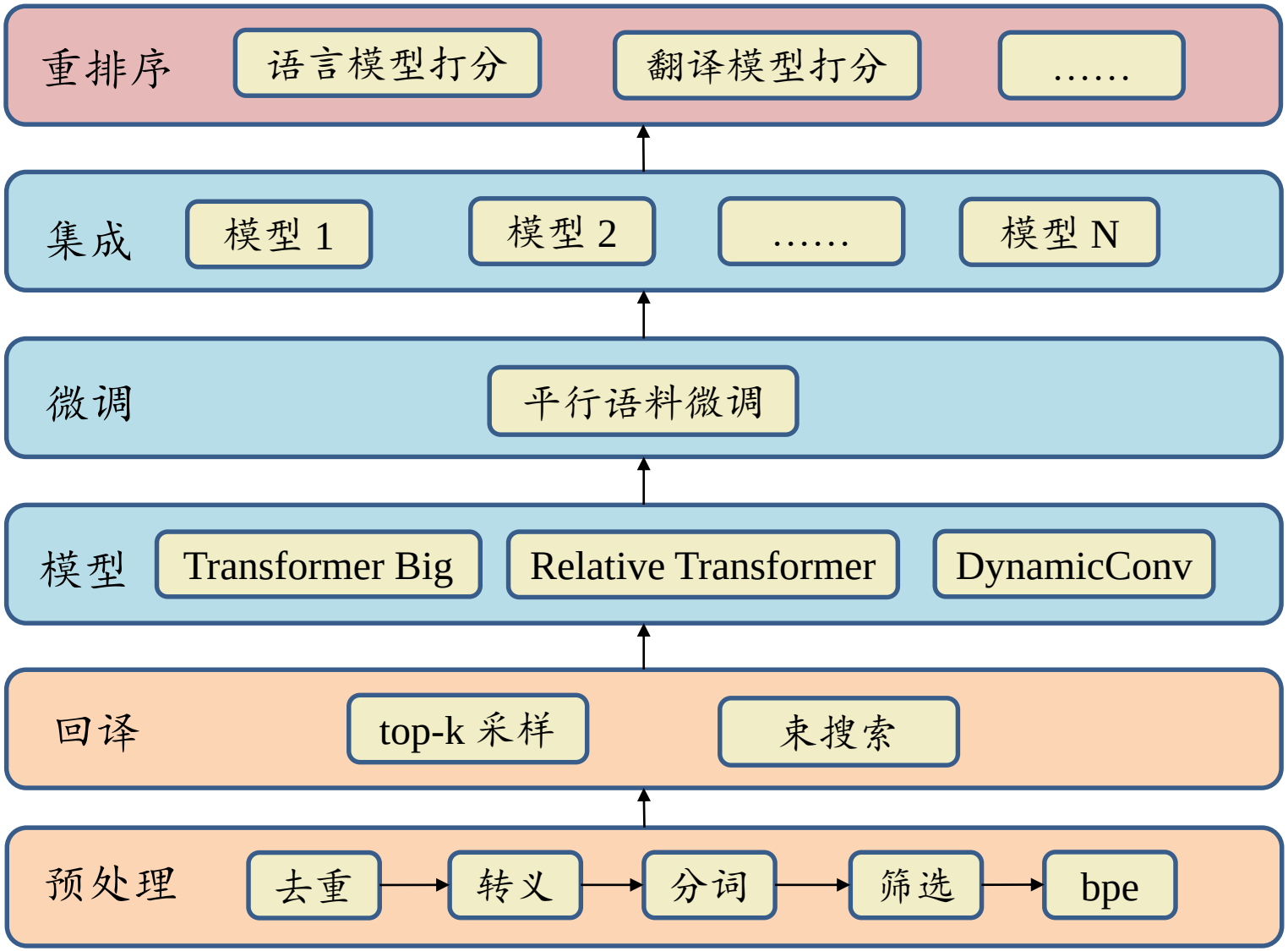
南京大学CCMT 维汉方向评测报告

王东琪 刘子涵 蒋庆男 孙泽维 黄书剑 陈家骏

南京大学 自然语言处理实验室

2020年10月

系统架构

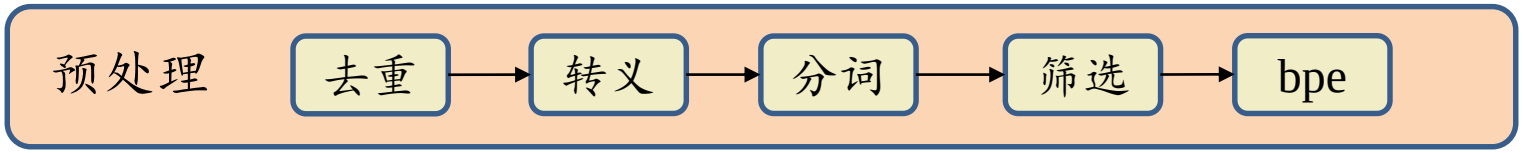

重排序
语言模型打分
翻译模型打分
……
集成
模型 1
模型 2
……
模型 N
微调
平行语料微调
模型
Transformer Big
Relative Transformer
DynamicConv
回译
top-k 采样
束搜索
预处理
去重
转义
分词
筛选
bpe

数据预处理

预处理步骤：

1. 去重、全半角转换
2. 转义、标点符号规范化 (moses系统)
3. 分词(pkuseg库)
4. 根据长度、 长度比例筛选
5. 对齐率筛选
6. BPE分词[1]

处理后：16w平行句对



回译

回译策略：

1. 对比了top-k采样[5]和束搜索两种不同的解码方式，基于top-k采样的结果(45.34)要优于基于束搜索(45.27)的结果。
2. 区分真实数据和基于回译构造的数据[6]也会有提升。
3. 两次top-k采样得到两个不同的数据集：sample1和sample2。

表2 不同的回译策略对翻译质量的影响

setting	BLEU
Transformer Base w\o BT	38.56
Transformer Big w\o BT	37.01
Transformer Big w\ BT(beam search)	45.27
Transformer Big w\ BT(top-10 sampling)	45.34
Transformer Big w\ BT(top-10 sampling) + tag	46.00

模型架构

模型架构：

- Transformer Big模型[2]
- 基于相对位置的Transformer Big模型[3]
- DynamicConv模型[4]

表1 模型参数

Hyper-parameter name	Hyper-parameter value
Embedding size	1014
Hidden size	1024
FFN inner size	4096
Attention heads	16
Dropout	0.2
Label smoothing	0.1

微调

微调的提升很显著：

基于回译构造的数据和真实数据之间存在差异，因此在使用混合数据训练模型后，使用真实数据来微调模型。

表3 在真实数据上微调

Index	Model	Before tuning	After tuning
1	Transformer(sample1)	46.00	46.96
2	Transformer(sample2)	45.32	46.76
3	Checkpoint Averaging(sample1)	45.53	47.26
4	Relative Transformer(sample1)	45.56	47.53
5	Relative Transformer(sample2)	45.40	47.43
6	Dynamic Convolution(sample1)	45.42	46.82

集成

集成不同的模型可以有效地提升翻译质量。

表4 模型集成的结果

Ensemble selection	**BLEU**
4 + 5 (beam size = 5)	47.97
3 + 4 + 5 (beam size = 5)	48.21
1 + 3 + 4 + 5 (beam size = 5)	48.51
1 + 3 + 4 + 5 + 6 (beam size = 5)	48.54
1 + 2 + 3 + 4 + 5 + 6 (beam size = 5)	48.63
1 + 2 + 3 + 4 + 5 + 6 (beam size = 24)	48.80

1：Transformer(sample1)　　2：Transformer (sample2)
3：Checkpoint Averaging(sample1)　　4: Relative Transformer (sample1)
5: Relative Transformer (sample2)　　6: Dynamic Convolution(sample1)

重排序

重排序指标

- 从右到左的得分
- 从目标端到源端的得分
- 语言模型得分
- 束索引
- 覆盖度

使用重排序，BLEU从48.80提到了49.17。

消融实验

消融实验分析

- 回译带来了很显著的提升。
- 微调和集成也带来了一定的提升。
- 重排序的提升没有那么显著。

表5 消融实验结果

System	Uyghur -> Chinese
Transformer Base	38.56
Transformer Big	37.01
+ Back Translation	46.00
+ Fine-tuning	46.96
+ Ensembling	48.80
+ Reranking	49.17

结论

- 回译策略
  - 低资源场景下，回译会有很大的提升。
  - 采样的回译策略要优于束搜索的回译策略。
  - 区分真实数据和基于回译构造的数据也会有提升。
- 微调提升很显著。
  - 在真实数据上微调提升很显著。
- 模型集成
  - 可选择不同的数据训练的模型。
  - 可选择不同架构的模型

参考文献

[1] Sennrich, Rico, et al. “Neural Machine Translation of Rare Words with Subword Units.” Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), vol. 1, 2016, pp. 1715–1725.

[2] Vaswani, Ashish, et al. “Attention Is All You Need.” Proceedings of the 31st International Conference on Neural Information Processing Systems, 2017, pp. 5998–6008.

[3] Shaw, Peter, et al. “SELF-ATTENTION WITH RELATIVE POSITION REPRESENTATIONS.” NAACL HLT 2018: 16th Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, vol. 2, 2018, pp. 464–468.

[4] Wu, Felix, et al. “Pay Less Attention with Lightweight and Dynamic Convolutions.” ICLR 2019 : 7th International Conference on Learning Representations, 2019.

[5] Edunov, Sergey, et al. “Understanding Back-Translation at Scale.” EMNLP 2018: 2018 Conference on Empirical Methods in Natural Language Processing, 2018, pp. 489–500.

[6] Caswell, Isaac, et al. “Tagged Back-Translation.” Proceedings of the Fourth Conference on Machine Translation (Volume 1: Research Papers), 2019, pp. 53–63.

结束

谢谢！

南京大学CCMT机器翻译质量评估评测报告

崔渠 耿祥 黄书剑 陈家骏

南京大学 自然语言处理实验室

2020年10月

目录

- 数据预处理
- 模型介绍
- 单模型结果
- 集成方法
- 集成结果

数据预处理

- 使用数据
  - 平行语料
    - WMT18 中英平行语料（7.5M）
    - neu2017（2M），datum2015（1M）, casia2015（1M）, casict2015（2M）中英平行语料
  - QE语料
    - 英中 / 中英 句子级别 / 词级别 语料
- 预处理方法
  - 中文：使用jieba分词对所有中文句子进行分词，使用词频最高的3W词作为中文词典
  - 英文：使用BPE[1]对所有英文句子进行分词，使用切分后的所有词作为英文词典

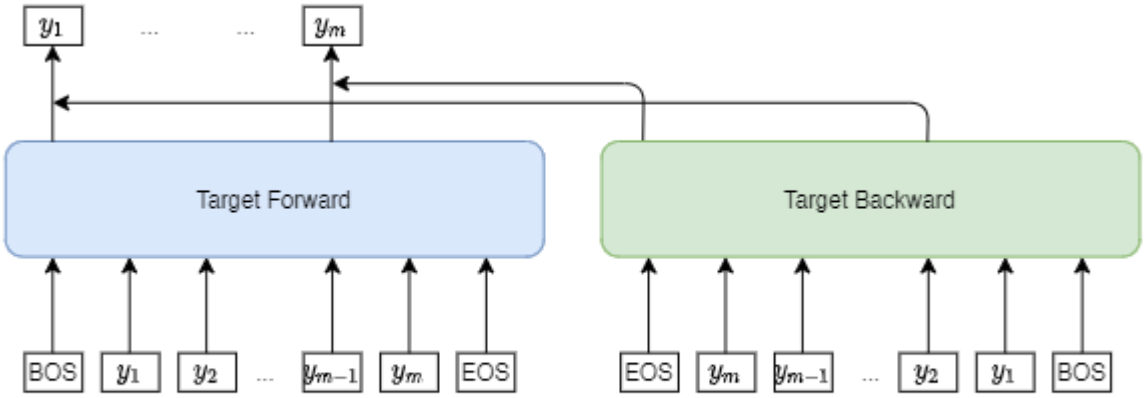
数据预处理

- 词级别QE任务处理
  - 由于一个单词可能会被切分成多个子词，我们会将这个词对应的标记进行相应的扩展。
  - 在预测时将合并子词的输出结果，若存在BAD，则整个词视为BAD。
  - 训练时：Apple →App@@ le OK → OK OK
  - 预测时：App@@ le OK BAD → Apple BAD

模型介绍-已有模型

- QUETCH[2]
  - 仅使用线性层与tanh激活函数直接在QE数据上进行训练。
- NuQE[3]
  - 使用GRU模型直接在QE数据上进行训练。
- 以上两个模型均使用OpenKiwi[4]开源代码。
- QE Brain[5]
  - 基于predictor-estimator框架；
  - predictor可以在平行语料上进行训练；
  - 在QE数据集上，predictor充当特征提取器，输送特征到estimator中预测QE标签。

模型介绍-QE Brain

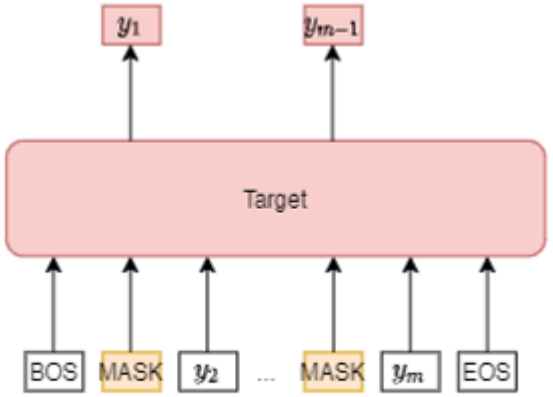


(a) Original QE Brain.

- 仅画出目标端模型；
- 叠加前向和后向的模型，预测译文中的每一个词。

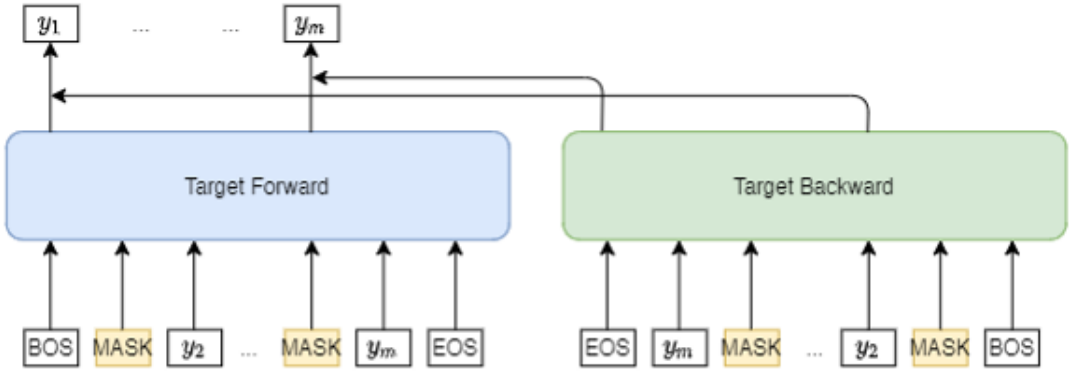
模型介绍-改进模型

- MTLM
  - 使用MLM[6]模型代替QE Brain中目标端两个叠加的单向模型。
  - 模型更小，能直接获得深度双向信息。



(c) MTLM.

- Masked QE Brain
  - 在QE Brain的基础上，在译文端进行与MLM一致的随机屏蔽策略。



(b) Masked QE Brain.

单模型结果

Parallel Dataset	Method	EN-ZH		ZH-EN	
		Sent	Word	Sent	Word
-	NuQE	19.39	**29.22**	23.75	35.56
-	QUETCH	29.08	11.72	25.04	30.24
WMT18	QE Brain	47.26	15.90	52.04	37.68
	Masked QE Brain	47.26	23.38	52.77	35.60
	MTLM	**52.16**	24.67	**55.94**	**43.75**
neu2017	MTLM	45.23	20.07	53.43	40.62
datum2015		44.58	14.49	50.49	41.30
casia2015		44.27	23.43	51.45	42.34
casict2015		39.19	14.74	52.34	42.53

- 利用平行语料的方法，效果更好。
- 平行语料的数量，对结果影响较大。

- 引入真实环境中不存在的MASK标记，并不会影响模型性能。
- 通过对比MTLM与Masked QE Brain，说明深度双向信息是重要的。

集成方法

- 句子级别
  - 按结果集成
    - 收集不同系统上，训练集，验证集，测试集上的HTER输出；使用输出HTER值预测真实的HTER值。
  - 按表示集成
    - 拼接QE Brain， masked QE Brain以及MTLM的高维输出表示，输入到一个线性层中预测真实的HTER值。
- 词级别
  - 投票法
    - 使用多个模型的预测结果进行投票。

Parallel Dataset	Method	EN-ZH		ZH-EN	
		Sent	Word	Sent	Word
ensemble	sent-neural	56.55	-	57.23	-
ensemble	sent-result	54.18	-	55.18	-
ensemble	word-voting	-	30.25	-	48.28

结论

- 在QE任务中，迁移平行语料的知识是必要的。
- 在目标端使用MLM模型优于使用两个叠加的LM模型。
- 在目标端引入mask标记，不会降低模型的性能。

参考文献

- [1] Neural machine translation of rare words with subword units.
- [2] Improving machine translation quality estimation with neural network features.
- [3] Deep learning for word-level translation quality estimation.
- [4] Openkiwi: An open source framework for quality estimation.
- [5] “bilingual expert” can find translation errors.
- [6] Pre-training of deep bidirectional transformers for language understanding.

结束

谢谢！