中国科学技术信息研究所 CWMT'2015 技术报告

何彦青，孟令恩，石崇德
（中国科学技术信息研究所， 北京 100038）

摘要：本文介绍了中国科学技术信息研究所（ISTIC）参加第十一届全国机器翻译研讨会机器翻译评测的情况。本单位参加了维汉、藏汉、蒙汉三个机器翻译评测项目。本文阐述了本单位机器翻译系统的实现框架以及实施细节，并分析了它们在评测数据上的性能表现。

关键词：机器翻译；系统融合；假设对齐

ISTIC Evaluation Technical Report for CWMT'2015

Yanqing He, Ling'en Meng,Chongde Shi
Institute of Scientific and Technical Information of China, Beijing 100038，China

Abstract: This is an overview of ISTIC evaluation technical report for the 11th China workshop on machine translation. ISTIC participated in the Uyghur-Chinese, Mongolian-Chinese and Tibetan-Chinese machine translation task. This paper describes the implement framework of our machine translation system. We also give the key techniques and analyze the experimental results over the evaluation data.

Keywords: machine translation; system combination; hypothesis alignment

1 引言

中国科学技术信息研究所 (Institute of Scientific and Technical Information of China, ISTIC) 参加了 2015 年第十一届全国机器翻译研讨会（CWMT'2015）组织的机器翻译评测活动。在所有的评测项目中，ISTIC 参加了维汉、藏汉和蒙汉三个机器翻译项目。在本次的机器翻译评测中，ISTIC 采用了统计机器翻译多引擎相结合进行系统融合的策略，提交了多机器翻译融合的结果。

本文的结构安排如下：第二节给出 ISTIC 机器翻译系统的总体框架和系统融合策略；第三节介绍数据的使用和处理；实验结果及相关分析在第四节；最后在第五节给出结论和展望。

2 系统描述

ISTIC 提交的维汉、藏汉和蒙汉三个测试任务的翻译结果都采用了统一的翻译策略：利用多个机器翻译系统来生成多个翻译输出，之后通过系统融合来生成融合的翻译结果，根据开发集上的评价结果来进行选择，如果融合系统的翻译结果最优，则采用融合系统来翻译测试集；如果融合系统没有达到最优，则采用最优的单系统来翻译测试集。在机器翻译多系统中，我们使用了基于短语的统计机器翻译方法和基于层次短语的统计机器翻译方法，每一个方法使用不同的参数得到多机器翻译系统来生成 1-Best，汇集所有的 1-Best 组成 1-Best List，进而进行系统融合[1]。系统的总体框架见图 1。

2．1 多机器翻译系统

我们采用的统计机器翻译方法包括基于短语的统计机器翻译[2,3]和基于层次短语的统计机器翻译[4]，前者使用连续的短语对为翻译单元，后者采用了非连续的短语，都采用了对数线性模型来进行翻译结果的遴选。为了获取尽可能多的翻译结果来为后续的系统融合服务，我们采用了如下措施：1）不同的词对齐，除了常用的 GIZA++双向词对齐之后进行 grow-diag-final-and 扩展得到多对多的词对齐之外，还采用了 Berkeley1的词对齐。2）不同的语言模型：除了使用评测方提供的训练语料的目标语言端用来训练语言模型，还加入了搜狗语料。3）不同的翻译引擎：我们使用了 Moses[2]的基于短语的统计机器翻译引擎和基于层次

[1] http://code.google.com/p/berkeleyaligner/downloads/list
[2] http://www.statmt.org./moses/

短语的统计机器翻译引擎，还使用了NiuTrans3[5]的基于短语的统计机器翻译引擎。通过不同措施的选择配置就可以得到多个翻译系统。

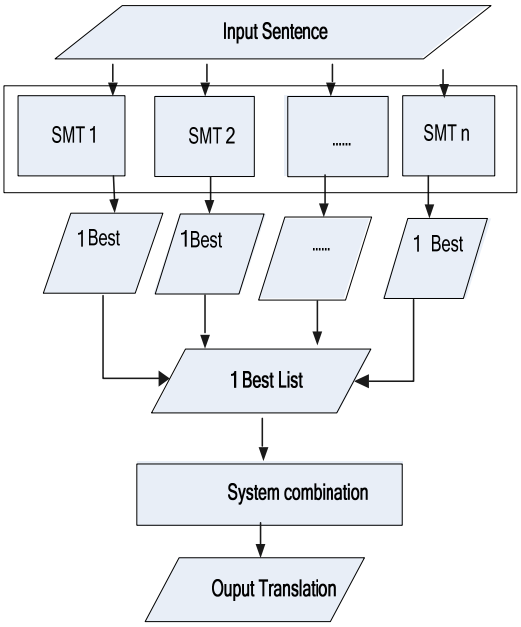

图 1　ISTIC 机器翻译系统框架

2.2 系统融合

常用的多机器翻译系统融合可以从句子、短语和词三个级别上独立进行[1,6,7,8,9]。ISTIC 采用词级的基于混淆网络(confusion-network)的系统融合算法[1]融合多个翻译系统的输出结果来形成一致性输出(Consensus output)作为最后的翻译结果。混淆网络包含一序列替换词集合，其中可能包括 null 词，并附带有相应的打分。通过从每一个替换词集合中选择一个词产生得分最好的序列。我们的整个系统融合框架如图 2 所示。对于每一个基于统计机器翻译的单引擎，采用了不同策略来生成 1-best，合并每一个单引擎的 1-Best 为 1-Best Lists 来进行系统融合。我们采用了最小贝叶斯风险解码技术[10]来动态地选择骨架（Backbone）。在将每个假设与骨架进行词对齐时选用了 IHMM（Indirect Hidden Markov）[1]假设对齐方法，利用相似度模型和扭曲模型，应用 Viterbi 算法获取最好的假设对齐，之后进行对齐归一化得到一对一对齐，生成混淆网络。

给定源语言句子 f，混淆网络解码过程就是搜索具有最大概率的目标翻译 e：

$$e^* = \arg\max_{e} \sum_{m=1}^{M} \lambda_m h_m(e, f)$$

其中，$h_m(e, f)$ 为特征函数，λ_m 为特征权重。混淆网络解码过程中，仍然采用对数线性模型来完成最终的翻译融合，使用的特征函数有：

1： 词后验概率；
2： 语言模型特征；
3： 基于距离的重排序特征；
4： 词惩罚；

每个特征函数的权重由最小错误训练算法训练得出。

3　数据的使用和处理

在蒙汉、藏汉、维汉三个翻译任务中，训练语料均采用了评测方发布的针对每个任务的所有训练语料，语言模型训练数据采用了相应任务的训练语料的中文部分和评测方发布的搜狗新闻语料库，所有的基于统计的机器翻译单系统的参数都是在该项目发布的开发集上进行训练。

对中文数据进行的处理有：中文的分词和全角变半角，采用了 Urhen2.2[4]的中文分词工具。对藏语数据进行了基于最大熵的分词处理，对蒙语和维语数据分别进行了 Tokenization 的处理。翻译工具采用了 Moses 和 NiuTrans，词对齐工具采用了 GIZA++[5]（全部使用默认的参数）并对该对齐结果进行扩展对齐（grow-diag-final）[2]和 Berkeley 对齐工具，语言模型工具采用了 Srilm[6] [11]工具包来获取 5 元文法概率信息。中文的后处理只是去掉了空格。Berkeley 对齐在藏汉任务上采用了 15 次迭代，在蒙汉任务上采用了 5 次迭代。

[3] http://www.nlplab.com/NiuPlan/NiuTrans.ch.html

[4] http://www.openpr.org.cn/index.php/NLP-Toolkit-For-Natural-Language-Processing/68-Urheen-A-Chinese/English-Lexical-Analysis-Toolkit/View-details.html

[5] http://giza-pp.googlecode.com/

[6] http://www.speech.sri.com/projects/srilm/download.html

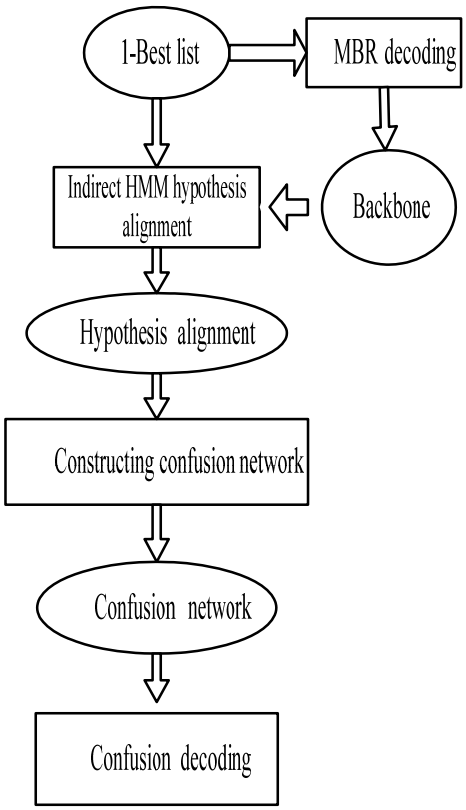

图 2 ISTIC 机器翻译系统融合框架

4 实验

我们分别在藏汉、维汉、蒙汉三个机器翻译项目发布的开发集和测试集上来验证翻译效果的优劣。在开发集上的打分使用了 CWMT2013 评测组织方发布的打分工具。在测试集上的打分使用了 CWMT2015 评测组织方开放的在线评测平台。

我们使用 100 个词来限制训练语料的最大长度，表 1 列出了使用的所有语料的详细统计量。

4.1 维汉翻译结果

在维汉翻译任务上，我们在开发集上测试了 7 个翻译系统。表 2 列出了在此任务上参与系统融合的单个翻译结果以及融合结果在开发集上的打分。表中的系统命名采用了“翻译工具-翻译模型-语言模型”的格式。翻译工具分别使用了 Moses 和 NiuTrans，Moses1 和 Moses2 表示采用了不同版本的两个 Moses，前者为分步运行 moses，后者采用 Moses 的“Experiment management system”。词对齐我们采用了 GIZA++的词对齐。翻译模型采用了基于短语的统计机器翻译模型（Phrase）基于层次短语的统计机器翻译模型（Hiero）。语言模型中，smalllm 表示只使用了当前任务的训练语料的中文部分来训练 5 元的语言模型，biglm 表示使用了当前任务的训练语料的中文部分加上 Sogou 语料来训练 5 元的语言模型。

表 1 实验语料的统计量

数据集		语言	句子个数	词汇表	平均句长
藏汉	训练集	藏语	126530	35843	13.40
		汉语	126530	45301	9.74
	开发集	藏语	650	2258	19.73
		汉语	2600	2877	13.65
	测试集	藏语	729	1520	12.41
维汉	训练集	维语	139801	137921	19.72
		汉语	139801	81656	19.71
	开发集	维语	700	5946	26.05
		汉语	2800	7006	24.70
	测试集	维语	1000	8110	21.45
蒙汉	训练集	蒙语	124240	81357	20.52
		汉语	124240	64277	14.71
	开发集	蒙语	1000	2246	7.40
		汉语	4000	2831	7.31
	测试集	蒙语	1001	3000	5.92

表 2：维汉任务在开发集上的比较

系统	NIST	BLEU	BLEU_SBP	GTM	mWER	mPER	ICT
Moses1-phrase-smalllm	10.5785	0.5395	0.5122	0.8123	0.5074	0.3329	0.4642
Moses1-hiero-biglm	11.0808	0.5705	**0.5482**	0.8303	0.4917	0.3148	0.4842
Moses2-phrase-smalllm	10.5343	0.5124	0.4953	0.8094	0.5337	0.3391	0.4330
Moses2-phrase-biglm	10.9255	0.5533	0.5314	0.8253	0.5077	0.3227	0.4671
Moses2-hiero-smalllm	10.5322	0.5124	0.4953	0.8091	0.5341	0.3396	0.4327
NiuTrans-phrase-smalllm	10.7617	0.5412	0.5204	0.8136	0.5179	0.3293	0.4608
NiuTrans-phrase-biglm	11.2141	0.5735	**0.5546**	0.8326	0.5012	0.3139	0.4810
Inhmm system combination	10.2833	0.5551	**0.5276**	0.8011	0.4840	0.3434	0.4886

表 3：维汉任务在测试集上的比较

系统	BLEU5-SBP	BLEU5	BLEU6	NIST6	NIST7	GTM	mWER	mPER	ICT	METEOR	TER
NiuTrans-phrase-biglm	0.4190	0.4390	0.3956	10.07	10.09	0.7645	0.5009	0.3504	0.4473	0.5782	0.4415
Moses1-hiero-biglm	0.3839	0.4004	0.3565	9.47	9.49	0.7481	0.5016	0.3767	0.4286	0.5519	0.4537

从表 2 中可以看出，同样的设置参数下，大的语言模型要优于小的语言模型，Niutrans 的翻译效果要优于 Moses[7]。对这 7 个系统进行系统融合，这里使用了 1best 的翻译假设。表 2 显示出 1best 的融合结果没有达到预期的融合目的，虽然比最差的翻译结果要好，但是比最好的翻译结果要差。因此在测试集中我们选择了最优的单系统 NiuTrans-phrase-biglm 的翻译结果作为 primary，第二优的单系统 Moses1-hiero-biglm 的翻译结果作为 contrast 进行提交，表 3 给出测试集上的翻译效果对比结果，从中可以看出在开发集上的最优单系统仍然要优于第二优的单系统。需要说明的是：由于当时参加评测的时间比较紧张，在维汉上我们没有使用 Berkeley 对齐进行实验。

4.2 藏汉翻译结果

在藏汉翻译任务上，我们在开发集上测试了 6 个翻译系统。表 4 列出了在此任务上参与系统融合的单个翻译结果以及融合结果在开发集上的打分。表中的系统命名采用了“词对齐_语言模型_翻译模型_翻译工具”的格式。词对齐我们采用了 Berkeley 的词对齐和 GIZA++的词对齐两种，语言模型中，smalllm 表示只使用了当前任务的训练语料的中文部分来训练 5 元的语言模型，biglm 表示使用了当前任务的训练语料的中文部分加上 Sogou 语料来训练 5 元的语言模型。翻译模型采用了基于短语的统计机器翻译模型（Phrase）基于层次短语的统计机器翻译模型（Hiero）。翻译工具分别使用了 Moses 和 NiuTrans。

从表 4 中可以看出，同样的设置参数下，大的语言模型要优于小的语言模型，Berkeley 的对齐要优于 GIZA 的对齐，Hiero 的翻译效果要优于 Phrase 的翻译效果，Niutrans 的翻译效果要优于 Moses。从这 6 个系统中我们选择了前 4 个系统进行系统融合，这里分别使用了 1best 的翻译假设和 100best 的翻译假设，表 4 中最后两行显示，1best 的融合结果要优于 100best。表 4 也显示出 1best 的融合结果达到了预期的融合目的，优于每个单系统的翻译，因此在测试集中我们选择了 1best 的融合

[7] 这个主要的原因是 Moses 翻译中采用了二进制。

表 4：藏汉任务在开发集上的比较

系统	NIST	BLEU	BLEU_SBP	GTM	mWER	mPER	ICT
Berkeley_biglm_Hiero_moses	10.8134	0.6584	**0.6378**	0.8504	0.3522	0.2177	0.6388
Berkeley_biglm_Phrase_moses	10.7301	0.6398	**0.6145**	0.8442	0.3778	0.2337	0.6253
Berkeley-biglm-Phrase_NiuTrans	10.9445	0.6735	**0.6567**	0.8561	0.3336	0.2081	0.6428
Berkeley-smalllm-Phrase_NiuTrans	10.6057	0.6387	**0.6196**	0.8392	0.3650	0.2291	0.6114
Giza-biglm-Phrase-Niutrans	10.1647	0.6176	0.6031	0.8253	0.3794	0.2541	0.5982
Giza-smalllm-Phrase-Niutrans	9.9586	0.5994	0.5843	0.8132	0.3916	0.2713	0.5752
1best inhmm-system-combination	11.0447	0.6830	**0.6630**	0.8583	0.3260	0.2068	0.6580
100 best inhmm-system-combination	10.7737	0.6504	0.6382	0.8622	0.3430	0.2094	0.6181

表 5：藏汉任务在测试集上的比较

系统	BLEU5-SBP	BLEU5	BLEU6	NIST6	NIST7	GTM	mWER	mPER	ICT	METEOR	TER
1best inhmm-system-combination	0.8809	0.8973	0.8773	12.90	13.03	0.9442	0.1384	0.0839	0.9218	0.9052	0.1014
Berkeley-biglm-Phrase_NiuTrans	0.8751	0.8895	0.8684	12.86	12.99	0.9438	0.1441	0.0885	0.9137	0.9001	0.1069

结果作为 primary 进行提交，同时采用开发集上最优单系统 Berkeley-biglm-Phrase_NiuTrans 翻译测试集作为对比结果。表 5 给出测试集翻译效果对比结果，可以看出，在测试集上系统融合也达到了比单系统更好的结果。

4.3 蒙汉翻译结果

在蒙汉翻译任务上，我们在开发集上测试了 5 个翻译系统。表 6 列出了在此任务上参与系统融合的单个翻译结果在开发集上的打分。表中的系统命名采用了“词对齐_语言模型_翻译模型_翻译工具”的格式。词对齐我们采用了 Berkeley 的词对齐和 GIZA++的词对齐两种，语言模型中，smalllm 表示只使用了当前任务的训练语料的中文部分来训练 5 元的语言模型，biglm 表示使用了当前任务的训练语料的中文部分加上 Sogou 语料来训练 5 元的语言模型。翻译模型采用了基于短语的统计机器翻译模型（Phrase）基于层次短语的统计机器翻译模型（Hiero）。翻译工具分别使用了 Moses 和 NiuTrans。从表 6 中可以看出，同样的设置参数下，大的语言模型要差于小的语言模型，Berkeley 的对齐要优于 GIZA 的对齐，Niutrans 的翻译效果要优于 Moses。

表 6：蒙汉任务在开发集上的比较

系统	NIST	BLEU	BLEU_SBP	GTM	mWER	mPER	ICT
Giza-smalllm-phrase-NiuTrans	4.8869	0.2404	0.2258	0.5361	0.6069	0.5403	0.3953
Berkeley-smalllm-phrase-NiuTrans	9.1365	0.5645	**0.5234**	0.7617	0.3827	0.3173	0.6179
Giza-biglm-phrase-NiuTrans	4.9575	0.2406	0.2242	0.5297	0.6232	0.5471	0.3847
Berkeley-biglm-phrase-NiuTrans	8.9369	0.5451	**0.5031**	0.7537	0.3963	0.3273	0.6118
Berkeley-biglm-phrase-Moses	8.8186	0.5383	0.4907	0.7526	0.3992	0.3329	0.6153

表 7：蒙汉任务在测试集上的比较

系统	BLEU5-SBP	BLEU5	BLEU6	NIST6	NIST7	GTM	mWER	mPER	ICT	METEOR	TER
Berkeley-smalllm-phrase-NiuTrans	0.3642	0.4061	0.3749	7.1364	7.1589	0.6060	0.5429	0.4831	0.4895	0.5038	0.5066
Berkeley-biglm-phrase-NiuTrans	0.3618	0.3933	0.3642	6.6981	6.7233	0.6007	0.5431	0.4830	0.5034	0.4916	0.5044

由于时间紧张以及各个单系统的翻译效果差别特别大，蒙汉任务上我们没有进行系统融合。在测试集中我们选择了最优单系统 Berkeley-smalllm-phrase-NiuTrans 在测试集上的翻译结果做为 primary，第二优单系统 Berkeley-biglm-phrase-NiuTrans 的翻译结果作为 contrast 进行提交。表 7 给出测试集翻译效果对比结果，可以看出，在开发集上的最优单系统仍然要优于第二优的单系统。

5 结论

ISTIC 参加了 CWMT'2015 维汉、藏汉和蒙汉三个少数民族语言的翻译任务的机器翻译项目，在维汉和藏汉翻译任务上取得了较好的成绩，但是在蒙汉翻译任务上与最好的翻译系统仍存在一定的差距。经过上述实验分析以及整个参评过程，我们积累了下述经验：

1）数据的预处理问题。从实验中可以得知不同的预处理会影响到统计机器翻译系统的翻译效果。由于缺少精通少数民族语言的人才，对于源语言端语料出现的问题不能准确洞察，只能给以简单的分词和标点隔离。尤其是蒙汉翻译任务的训练语料质量比较差，蒙语中夹杂汉语，汉语中夹杂蒙语，很难对语料进行高质量的清洗，导致翻译的效果也大打折扣。因此需要加强对源语言端的语料的清洗和预处理研究。

2）机器翻译研究如果想获得大规模应用性发展，系统融合研究是必不可少的。但是我们的系统融合在藏汉翻译效果较好，但是在维汉上并没有同样的体现，这主要是因为对多个单系统各自的优势和缺陷分析不够细致，仍需要进一步进行筛选以期得到互补性融合，今后，我们致力于采取进一步改善单系统遴选方案来增强系统融合的翻译效果。

致谢

本文系国家自然科学基金项目“面向科技文献的统计机器翻译语境分析”（项目编号：61303152）和中日国际合作项目“面向科技文献的日汉双向实用型机器翻译合作研究”（项目编号：2014DFA11350）研究成果之一。

参考文献

[1] Xiaodong He, Mei Yang, Jianfeng Gao, Patrick Nguyen, and Robert Moore, Indirect-HMM-based Hypothesis Alignment for Combining Outputs from Machine Translation Systems. In Proceedings of EMNLP 2008.

[2] Philipp Koehn, Franz Josef Och, and Daniel Marcu. Statistical Phrase-Based

Translation[C]. In Proceedings of the Human Language Technology conference / North American Chapter of the Association for Computational Linguistics (HLT/NAACL-2003), 2003, pages 127-133.

[3] Franz Josef Och, Hermann Ney. A Systematic Comparison of Various Statistical Alignment Models[J], Computational Linguistics, 2003, volume 29, number 1, pp. 19-51.

[4] David Chiang. 2005. A Hierarchical Phrase-based model for Statistical Machine Translation. In Proceedings of ACL 2005, pages 263–270.

[5] Tong Xiao, Jingbo Zhu, Hao Zhang and Qiang Li. 2012. NiuTrans: An Open Source Toolkit for Phrase-based and Syntax-based Machine Translation. In Proc. of ACL, demonstration session.

[6] Srinivas Bangalore, German Bordel, and Giuseppe Riccardi. 2001. Computing consensus translation from multiple machine translation systems. In Proc. ASRU, pages 351–354.

[7] J.G. Fiscus. A Post-Processing System to Yield Reduced Word Error Rates: Recognizer Output Voting Error Reduction (ROVER). IEEE Workshop on Automatic Speech Recognition and Understanding, 1997.

[8] Antti-Veikko I.Rosti, Necip Fazil Ayan, Bing Xiang, Spyros Matsoukas, and Richard Schwartz, Bonnie J.Dorr. 2007a. Combining Outputs from Multiple Machine Translation Systems. In Proceedings of NAACL HLT, pages 228-235, Rochester,NY, April 2007.

[9] Antti-Veikko I.Rosti, Bing Zhang, Spyros Matsoukas, and Richard Schwartz. 2007b. Improved Word-level System Combination for Machine Translation. In Proceedings of ACL 2007.

[10] K.C. Sim, W. Byrne, M. Gales, H. Sahbi and P. Woodland. Consensus Network Decoding For Statistical Machine Translation System [A]. In: ICASSP, 2007.

[11] Andreas Stolcke, 2002. SRILM-An extensible language modeling toolkit. In Proceedings of International Conference on spoken language processing, volumn 2, pages 901-90.