# NiuTrans 开源统计机器翻译系统技术分析*

李强，孙坤杰，刘卓，肖桐，朱靖波
（东北大学自然语言处理实验室，辽宁 沈阳 110819）

**摘要：**本文主要介绍了最新版 *NiuTrans* 开源统计机器翻译系统的技术特点。在最新版的 NiuTrans 中，包含当前主流的统计机器翻译模型，即：基于短语的翻译模型，基于层次短语的翻译模型，基于句法（树到串/串到树/树到树）的翻译模型。与此同时，NiuTrans 包含中英文的数据预处理工具，以及自主开发的精简且高效的语言模型工具。为方便机器翻译研究人员使用，NiuTrans 开发接口对用户开放。经实验证明，NiuTrans 系统在 NIST 和 CWMT 的多个翻译任务中表现出优异的翻译性能。

**关键词：**NiuTrans；统计机器翻译；开源工具包
**中图分类号：**TP391　　**文献标识码：**A

# Technical Analysis of NiuTrans Open Source Statistical Machine Translation System

LI Qiang, SUN Kun-Jie, LIU Zhuo, XIAO Tong, ZHU Jing-Bo
(Northeastern University NLP Lab, Shenyang, Liaoning 110819, China)

**Abstract:** This paper introduces the technical features for the latest version of *NiuTrans* open source machine translation toolkit. The toolkit supports the state-of-the-art models in statistical machine translation, including the phrase-based model, the hierarchical phrase-based model, and various syntax-based models (tree-to-string/string-to-tree/tree-to-tree). Furthermore, NiuTrans supports a preprocessing module for Chinese/English, and a simple and fast language model. In order to make it easy to use for machine translation researchers, several interfaces are available for further development with NiuTrans. We evaluate the NiuTrans system on the NIST and CWMT translation datasets. Experiments show that NiuTrans exhibits the state-of-the-art translation performance.

**Key words:** NiuTrans；Statistical Machine Translation；Open Source Toolkit

## 1 简介

自 1949 年美国人 Weaver 发表 Translation 备忘录并正式提出机器翻译思想以来，机器翻译已经发展了六十四年。目前，性能优异的机器翻译方法不需要人工书写大量的翻译规则，而是自动从大规模双语平行语料中统计翻译信息，在翻译的过程中从巨大的搜索空间中搜索得到翻译结果，这种基于统计模型的翻译方法称之为统计机器翻译。在基于统计的机器翻译模型中，分为基于短语的翻译模型[1][2]，基于层次短语的翻译模型[3][4]，基于句法（树到串/串到树/树到树）的翻译模型[5][6][7][8][9]。基于短语、层次短语的翻译模型不需要对源语言及目标语言进行深入的语言学分析，而是直接利用表层串的对应关系来进行翻译。与之不同的

* **收稿日期：** 2013 年 9 月 27 日　　**定稿日期：**2013 年 10 月 13 日
**基金项目：**国家自然科学基金（61073140；61272376；61100089）；高等院校博士学科点专项科研基金（20100042110031）；中国博士后基金（2013M530131）；中央高校基本科研基金（N100204002）
**作者简介：**李强（1988—），男，博士研究生，主要研究方向为机器翻译；孙坤杰（1990—），男，硕士研究生，主要研究方向为机器翻译；刘卓（1989—），男，硕士研究生，主要研究方向为机器翻译；肖桐（1982—），男，博士，主要研究方向为机器翻译；朱靖波（1973—），男，教授，博士生导师，主要研究方向为机器翻译。

是，基于句法的翻译模型则主要使用句法分析的结果来指导翻译。由于句法树可以更加全面深入地表示句子的结构信息，因此它可以为翻译模型提供更多的依据来进行结构翻译和调序。

NiuTrans[10]是东北大学自然语言处理实验室开发的一套开源统计机器翻译系统，是一个完整构建高质量统计机器翻译系统的平台。目前 NiuTrans 在统一架构下支持上文提到的基于短语、层次短语、句法的翻译模型。NiuTrans 遵循 GNU 通用公共许可协议[1]。自 2011 年 7 月公开发布以来，经过 7 次系统升级，目前最新版本升级至 NiuTrans 1.3.0 Beta 版[2]。在最新版开源系统中，主要包含以下功能模块：中英文的数据数据预处理模块，基于短语的统计机器翻译模型的训练、解码模块，基于层次短语的统计机器翻译模型的训练、解码模块，基于句法（树到串/串到树/树到树）的统计机器翻译模型的训练、解码模块，数据后处理模块，精简且快速的语言模型模块，以及翻译系统参数优化的最小错误率训练（MERT）[11]模块。NiuTrans 允许用户修改配置文件，定制个性化的翻译系统。

在 NiuTrans 开源统计机器翻译系统中，核心模块使用 C++语言进行开发，功能模块的连接使用 Perl 脚本语言进行实现。使用 C++语言开发训练与解码模块，确保翻译系统高效率的运行。C++面向对象的特点，保证了 NiuTrans 设计、开发的模块化与结构化，方便 NiuTrans 的升级与维护。

目前，由于没有介绍 NiuTrans 系统整体架构、技术细节、翻译性能的相关中文文档和资料，应广大 NiuTrans 用户的需求，写作此篇文章对 NiuTrans 进行较为详尽的介绍。

## 2 NiuTrans 整体架构

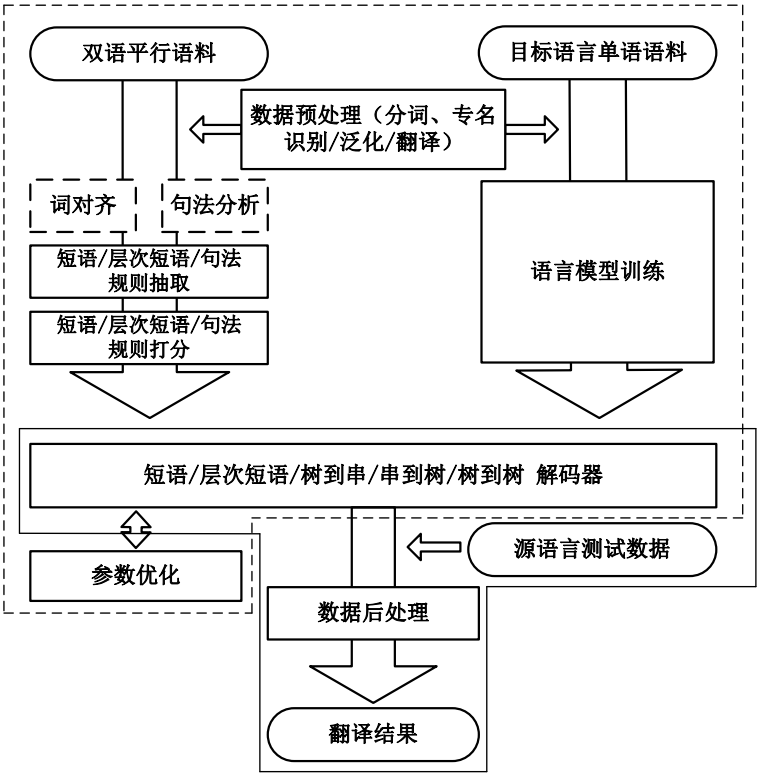


图 1. NiuTrans 开源统计机器翻译系统整体架构图

NiuTrans 在统一架构下实现对当前主流统计机器翻译模型的支持，包括：基于短语的

[1] http://www.gnu.org/licenses/gpl-2.0.html
[2] http://www.nlplab.com/NiuPlan/NiuTrans.html

翻译模型，基于层次短语的翻译模型，基于句法（树到串/串到树/树到树）的翻译模型。NiuTrans 的整体架构图如图 1 所示，每种翻译模型的翻译示例如图 2 所示，其中句法模型以树到树模型进行举例。

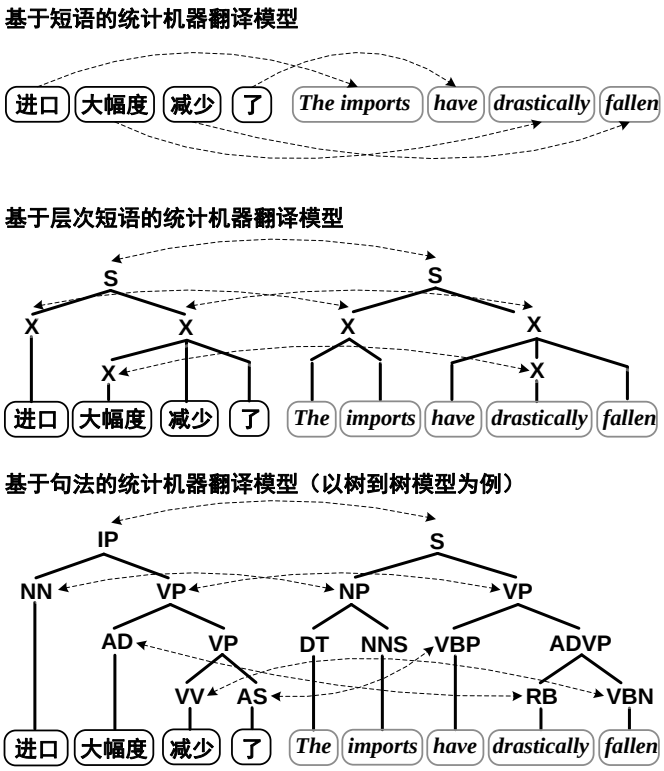


图 2. NiuTrans 系统中，基于短语、层次短语、句法的翻译模型

NiuTrans 开源统计机器翻译系统的搭建过程主要分成两个部分，即模型训练部分与解码部分。下面将分别对模型训练部分与解码部分的主要模块进行功能描述。

## 2.1 模型训练部分

模型训练部分如图 1 中左上方虚线框图所示，该部分主要功能如下：

- 数据预处理：完成对训练数据、开发集数据、测试集数据的预加工，包括：分词，词性标注，专名的识别、泛化、翻译等
- 词对齐：NiuTrans 未提供词对齐工具。用户可使用开源的词对齐工具，如 *GIZA++*[3] 对训练数据进行双向词对齐，使用 *grow-diag-final-and* 启发式算法对双向词对齐结果进行对称化处理
- 句法分析：NiuTrans 未提供句法分析工具。用户可使用开源句法分析工具，如 *Berkeley Parser*[4]，完成对源语言及目标语言句子的句法分析工作，得到源语言及目标语言对应的基于短语成分的句法分析树。句法分析树是 NiuTrans 基于句法的翻译系统中重要的组成部分
- 短语/句法规则抽取：自动从训练数据中抽取短语/句法翻译系统中使用的互译翻译片段
- 短语/句法规则打分：对互译翻译片段的合理性进行概率估计，包括：双向短语翻译概率、双向词汇化翻译概率等

[3] http://code.google.com/p/giza-pp
[4] http://code.google.com/p/berkeleyparser

- 语言模型训练：自动从目标语言单语语料中学习语言模型。语言模型是解码器使用的非常重要的特征，主要用于评价目标语言译文的流畅度
- 参数优化：使用最小错误率训练方法，在开发集数据上对翻译模型特征权重向量进行调优

### 2.2 解码部分

解码部分如图 1 中右下方实线框图所示，该部分主要功能如下：

- 解码器：NiuTrans 在统一框架下实现了基于短语、层次短语、句法（树到串/串到树/树到树）的统计机器翻译模型的解码器。解码器使用多种特征，完成从搜索空间中找出最佳目标语言译文，完成真正的翻译过程。解码器是 NiuTrans 系统中最重要的模块，也是对翻译性能影响最直接的一个模块。为了加速解码过程，解码器中使用多种剪枝技术，如束剪枝（beam pruning）和立方剪枝（cube pruning）[12] 技术
- 数据后处理：完成对机器翻译输出结果进一步的优化工作，如英文恢复大小写操作，使其更符合书写和阅读习惯

## 3 NiuTrans 技术细节

NiuTrans 工具是基于统计模型的开源机器翻译系统。统计机器翻译模型的核心思想是给每个潜在的翻译结果赋予一定的概率，选择概率最大的翻译结果作为最终的翻译结果。该方法将翻译过程归纳为一个形式化描述的数学模型：对于一个源语言句子 $s$，对应的目标语翻译 $t$ 有多种可能。通常，对于一个翻译对 $(s,t)$，将 $\Pr(t \mid s)$ 称之为 $t$ 作为 $s$ 的翻译结果的翻译概率。于是，翻译问题被转化为，对于已知的 $s$，找到翻译概率最大的目标语句子 $t^{'}$，其形式化定义如下：

$$t^{'} = \arg\max_{t} \Pr(t \mid s)$$

由于语言间的翻译现象非常复杂，在实际操作中不可能枚举所有可能的目标语翻译 $t$。此外，如何获得 $\Pr(t \mid s)$ 也是统计机器翻译必须解决的问题。因此，基于统计的机器翻译模型可以归纳为如下三个基本问题：

1) 建模问题：如何定义 $s$ 到 $t$ 的翻译概率，即定义 $\Pr(t \mid s)$
2) 训练问题：如何从数据中学习模型参数，即计算 $\Pr(t \mid s)$
3) 解码问题：如何在给定翻译模型条件下寻找最优目标语译文，即 $\arg\max$ 问题

针对建模问题，NiuTrans 采用了对数线性模型[13]。在对数模型的框架下，翻译概率 $\Pr(t \mid s)$ 被进一步分解，表示为一系列特征函数的加权值。针对训练问题，NiuTrans 的训练模块可以自动的从双语平行数据中学习翻译规则并优化模型参数。针对解码问题，NiuTrans 的解码模块可以快速、高效、高质量的完成对最优目标语译文的搜索，完成翻译过程。

### 3.1 数据预处理

NiuTrans 提供中英文双语文本的数据预处理工具。在进行数据预处理时，一些词汇作为专有名词需要进行特殊处理，以便提供高质量的翻译。这部分专有名词包括：数字、时间、日期、人名、地名、组织机构名、网址、数学公式等。其中数字、时间、日期这些可通过规律进行总结的专有名词，需要进行泛化并提供翻译；而人名、地名、组织机构名可提供双语

词典资源进行查询，从而提供精确的翻译；网址、数学公式这类专名本身是其翻译结果。NiuTrans 中英文数据预处理工具支持对上述专有名词的处理功能。

### 3.2 语言模型

NiuTrans 工具包中内置简单且高效的语言模型工具。在语言模型的训练过程中，生成的语言模型以 trie 数据结构存储为二进制文件[14]，该方法可降低语言模型所需硬盘和内存空间的大小。与此同时，为了限制语言模型的大小，减少噪音数据对模型质量的影响，语言模型训练模块引入了低频限制策略。在 NiuTrans 解码器中计算语言模型概率时，查询算法中集成了缓存（Cache）机制，大大提高了查询速度。

### 3.3 短语翻译系统

在 NiuTrans 开源工具中，包含性能优异的基于短语的统计机器翻译模型[1][2]。在基于短语的统计机器翻译系统中，输入的源语言待翻译句子被分解为一个短语序列，然后这些短语被一对一地映射为对应的翻译后的目标语言短语，而这些输出短语的顺序有可能与输入顺序不同，最终的输出结果为翻译结果。在这里，短语并没有实际的语言学意义，仅仅是一些连续的词串。短语翻译系统的翻译实例如图 2 中短语翻译模型所示。

在 NiuTrans 短语系统中，短语对抽取标准方法使用 Koehn 等提出的启发式方法[1]。与此同时，NiuTrans 提供基于组合的短语对抽取方法[5]，使用该方法可保证在不降低翻译性能的前提下，获得更加精简的短语翻译表。

NiuTrans 短语系统集成了两个调序模型，即基于最大熵的词汇化调序模型[15]与 MSD 调序模型[16][17][18]。

在 Wu 提出的 ITG 文法[19]作为约束的条件下，NiuTrans 短语翻译系统的解码器使用 CKY 算法[20]进行实现。CKY 算法是由 Cocke、Kasami 和 Younger 联合提出的用于句法分析的动态规划算法。在使用 CKY 算法的条件下，相邻两个短语对应的翻译候选可以进行顺序的组合，也可进行调序的组合。在机器翻译任务中，由于使用了语言模型这一不具有可加性的特征，在使用 CKY 算法进行解码时则失去了原有算法动态规划的性质。在使用 CKY 算法进行解码时，由于算法复杂度较高，所以为加速解码过程，解码器实现过程中使用束剪枝与立方剪枝技术。

NiuTrans 短语系统使用的特征如下：

- 短语翻译概率 $\Pr(\bar{t}|\bar{s})$，$\Pr(\bar{s}|\bar{t})$
- 词汇化翻译概率 $\Pr_{lex}(\bar{t}|\bar{s})$，$\Pr_{lex}(\bar{s}|\bar{t})$
- N 元语言模型 $\Pr_{lm}(t)$
- 目标词汇惩罚 $length(t)$
- 短语惩罚
- 删词惩罚
- 基于最大熵的调序模型 $f_{ME}(d)$
- MSD 调序模型

在 NiuTrans 开源系统提供的所有统计翻译模型中，基于短语的翻译模型表现出非常优秀的翻译性能。关于这方面的内容，将在性能分析中进行说明。

### 3.4 层次短语/句法翻译系统

NiuTrans 支持基于层次短语的翻译模型[3][4]。层次短语翻译模型是一种形式化的同步上下文无关文法（SCFG）。在不使用任何句法信息的前提下，层次短语翻译规则使用 Chiang

[5] http://www.nlplab.com/members/liqiang/ext.usage.html

提出的算法[4]从双语平行数据中抽取出来。层次短语翻译规则包含三个部分：包含终结符与非终结符的源语言层次短语，包含终结符与非终结符的目标语言层次短语，源语言与目标语言层次短语中非终结符的对应信息。在 NiuTrans 的实现中，层次短语翻译模型可看做是句法模型的一个特例。层次短语翻译系统的翻译实例如图 2 中层次短语翻译模型所示。

NiuTrans 同时支持基于句法的翻译模型。句法翻译系统的翻译实例如图 2 中句法翻译模型所示。在 NiuTrans 的句法系统中，句法翻译规则的形式与层次短语翻译规则相同，但是句法翻译规则非终结符部分包含句法信息。句法翻译规则的抽取过程大致分为以下两个步骤：

- 使用 Galley 等提出的 GHKM 算法[21]从含有句法信息的双语平行句对中抽取最小翻译规则
- 使用两个或多个最小翻译规则组装含有更多上下文信息的翻译规则

关于边缘对空词汇的处理，与 Galley 等[21]的处理方式不同，NiuTrans 句法翻译规则的抽取向所有相邻的对空词汇进行扩展。

NiuTrans 在基于层次短语和基于句法的翻译模型中，支持基于字符串的解码（string-based decoding）、基于树的解码（tree-based decoding）。下面将对这两种解码方法进行简单介绍：

- 基于字符串的解码：该解码方法主要是指层次短语翻译模型中的基于同步上下文无关文法（SCFGs）的解码，以及句法系统中的基于同步树替换文法(STSGs)的解码。在上下文无关文法或者树替换文法的框架下，解码可以被看做是一个句法分析（parsing）问题。为在解码过程中有效引入语言模型特征，将对包含两个以上变量的规则进行二叉化操作。解码过程中除了使用从双语平行句对中学习的规则之外，同时引入粘合规则（glue rules）[4]对顺序的源语言片段对应的翻译结果进行粘合
- 基于树的解码：在句法翻译系统中，如果源语言句法树已知，则可以使用基于树的解码方法[22]。在基于树的解码方法中，翻译规则首先被映射到输入句法分析树的节点上。之后按照自底向上解码方法，对整棵句法分析树进行模型分数计算等分析操作。最终句法分析树根节点对应的翻译结果为整个句子的翻译结果。由于在解码过程中搜索空间受到源语言句法分析树的限制，所以基于树的解码方法效率高于基于字符串的解码。但由于更多的搜索错误，基于树的解码方法的翻译性能低于基于字符串的解码

为加速解码过程，基于层次短语/句法系统的解码器实现过程中同样使用束剪枝与立方剪枝技术。

NiuTrans 层次短语/句法翻译系统使用的特征如下：

- 双向短语翻译概率 $\Pr(\tau_t(r)|\tau_s(r))$， $\Pr(\tau_s(r)|\tau_t(r))$
- 双向词汇化翻译概率 $\Pr_{lex}(\tau_t(r)|\tau_s(r))$， $\Pr_{lex}(\tau_s(r)|\tau_t(r))$
- N 元语言模型 $\Pr_{lm}(t)$
- 目标词汇惩罚 $length(t)$
- 规则惩罚
- 删词惩罚

NiuTrans 句法系统专用特征如下：

- 使用树根（root）进行归一化的规则概率 $\Pr(r|root(r))$
- 是否为组合规则 $IsComposed(r)$
- 是否为词汇化规则 $IsLex(r)$
- 是否为低频规则 $IsLowFreq(r)$

在这里，$\tau(\alpha)$ 指代树片段 $\alpha$ 的边界序列。

## 4 性能分析

在 NiuTrans 最新版开源系统中，基于短语的翻译模型、基于层次短语的翻译模型、基于句法的翻译模型均表现出非常优秀的翻译性能。在性能分析这一节中，将在 NIST 机器翻译评测数据与 CWMT 机器翻译评测数据上，对 NiuTrans 支持的所有模型的翻译性能给予合理的分析。

### 4.1 NIST 数据翻译性能

首先，在 NIST 机器翻译评测数据上的实验中，所有翻译系统的实验配置均为 NiuTrans 开源系统的默认配置。需要注意的是，为了减少句法规则的数量，在 NiuTrans 句法系统的默认配置中，过滤掉出现频次为一次的翻译规则[6]。参数调优的过程中，使用 IBM BLEU[23] 作为优化特征权重的目标函数。

表 1. NIST 汉英新闻领域翻译中数据使用情况。其中“训练数据 1/训练数据 2”中“词汇数”一列中，左侧为中文词汇数，右侧为英文词汇数（M=百万）

| 数据 | 行数 | 词汇数 |
|---|---|---|
| 训练数据 1 | 0.35 M | 8.8 M/10.4 M |
| 训练数据 2 | 1.07 M | 25.6 M/29.5 M |
| GIGAWORD Xinhua | 8.44 M | 233.54 M |
| NIST 03 | 919 | 23164 |
| NIST 04 | 1788 | 47580 |
| NIST 05 | 1082 | 28863 |

为了更加系统、全面的验证 NiuTrans 开源系统的翻译性能，本文选择两组训练数据进行实验。实验中训练数据来源于汉英新闻翻译领域。

- 训练数据 1：使用 NIST 数据中 LDC2003E14，LDC2005T10，LDC2003E07，LDC2005T06 作为训练数据，经数据预处理后，训练数据包含 35 万行汉英双语平行句对。关于这部分数据更加详细的信息，见表 1 中“训练数据 1”行
- 训练数据 2：使用 NIST MT 2008 评测提供的大规模双语训练语料中 NIST 数据中的一部分[7]作为训练数据，经数据预处理后，训练数据包含 107 万行汉英双语平行句对。关于这部分数据更加详细的信息，见表 1 中“训练数据 2”行

训练数据使用 NiuTrans 最新版数据预处理工具进行数据预处理[8]。使用 GIZA++工具对训练数据进行双向词对齐，使用 *grow-diag-final-and* 启发式算法对双语词对齐结果进行对称化处理。此外，本实验中使用英语 GIGAWORD 的 Xinhua 部分和双语数据的目标语部分训练一个 5 元语言模型。关于开发集和测试集，本文使用 NIST MT 2003 的测试集作为权重调优的开发集，使用 NIST MT 2004 和 NIST MT 2005 的测试集作为评价 NiuTrans 系统翻译性能的测试集。关于开发集与测试集数据更多的信息见表 1。翻译性能通过使用上下文不敏感（case-insensitive）的 IBM 版本的 BLEU[23]评价指标进行评价。

训练数据 1 翻译性能如表 2 所示，训练数据 2 翻译性能如表 3 所示。由于 NiuTrans 解码器在 MERT 过程中引入了随机扰动机制，即每次 MERT 的结果并不相同，所以在实验的过程中，采取进行多轮实验取平均值的策略。从表 2 和表 3 的实验结果可以看出，当使用的训练数据规模较小时（训练数据 1），短语翻译系统的翻译性能在所有模型中表现最优；层

[6] 可通过句法模型训练配置文件进行修改

[7] LDC2003E14,LDC2005T10,LDC2003E07,LDC2005T06,LDC2005E83,LDC2006E26,LDC2006E34, LDC2006E85,LDC2006E92,LDC2004T08

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

次短语翻译系统次之，但显著优于基于句法模型的翻译系统。在句法系统中，基于树到树的句法翻译模型的翻译性能表现最差。当扩充实验数据时（训练数据 2），短语翻译系统与层次短语系统相比并没有明显的性能优势；与此同时，基于串到树的句法系统翻译性能在 NIST MT 05 测试集上有了明显的提升，基于树到树的句法模型翻译性能仍在所有模型中表现最差。基于树到树的句法模型之所以性能表现很差，一方面是由于在 NiuTrans 句法翻译系统的默认配置中，过滤掉出现频次为一次的翻译规则；另一方面则是由于在训练和解码的过程中过多的受到了句法树的限制。目前，句法分析技术没有办法保证生成的双语句法分析树的正确性，同时句法分析技术也没有考虑不同语言之间结构的差异性，更没有同时对双语进行句法分析。在目前句法翻译系统的基础上，如果对输入的句法树进行二叉化操作[24]，句法翻译系统的翻译性能可以进一步提高。除此之外，使用基于森林的解码[25]、树结构模糊匹配方法[26]、模糊解码[27]的方法都可以有效的提高基于句法的翻译系统的翻译性能。

表 2. 使用训练数据 1，在汉英新闻领域 NiuTrans 系统的翻译性能，
其中每组实验结果通过 3 轮实验取平均值而来（M=百万）

| 系统 | 规则数（M） | NIST03 | NIST04 | NIST05 |
|---|---|---|---|---|
| 短语 | 32.94 | 0.3683 | 0.3550 | 0.3517 |
| 层次短语 | 63.84 | 0.3645 | 0.3511 | 0.3464 |
| 串到树 | 2.97 | 0.3571 | 0.3446 | 0.3393 |
| 树到串 | 2.83 | 0.3511 | 0.3445 | 0.3320 |
| 树到树 | 7.19 | 0.3271 | 0.3162 | 0.3102 |

表 3. 使用训练数据 2，在汉英新闻领域 NiuTrans 系统的翻译性能，
其中每组实验结果通过 3 轮实验取平均值而来（M=百万）

| 系统 | 规则数（M） | NIST03 | NIST04 | NIST05 |
|---|---|---|---|---|
| 短语 | 71.03 | 0.3785 | 0.3617 | 0.3634 |
| 层次短语 | 165.81 | 0.3769 | 0.3623 | 0.3633 |
| 串到树 | 8.44 | 0.3743 | 0.3559 | 0.3593 |
| 树到串 | 6.70 | 0.3563 | 0.3544 | 0.3487 |
| 树到树 | 14.60 | 0.3338 | 0.3217 | 0.3240 |

## 4.2 CWMT2013 基线系统翻译性能

表 4. CWMT 汉英新闻领域翻译中数据使用情况。其中“训练数据”中“词汇数”一列中，
左侧为中文词汇数，右侧为英文词汇数（M=百万）

| 汉英新闻数据 | 行数 | 词汇数 |
|---|---|---|
| 训练数据 | 4.55 M | 78.83 M/88.27 M |
| Routers-Corpora | 15.27 M | 316.91 M |
| 开发集 | 1006 | 26242 |
| 测试集 | 1003 | 24453 |

NiuTrans 为 CWMT2013 机器翻译评测提供汉英新闻领域短语、层次短语，英汉新闻领域短语、层次短语共四套基线统计机器翻译系统。在上述基线翻译系统中，训练数据使用 CWMT2013 官方提供的所有汉英双语平行句对，经数据预处理及语料过滤后，训练数据包

含 455 万行。训练数据使用 NiuTrans 最新版数据预处理工具进行数据预处理[9]。使用 GIZA++ 词对齐工具对训练数据进行双向词对齐，使用 *grow-diag-final-and* 启发式算法对双语词对齐结果进行对称化处理。在汉英新闻基线翻译系统中，使用 Routers-Corpora 大规模英语单语语料和双语数据中的英文部分训练 5 元语言模型；在英汉新闻基线翻译系统中，使用 Sogou 实验室提供的大规模中文单语语料和双语数据中的中文部分训练 5 元语言模型。使用 CWMT2013 评测组织方提供的开发集作为系统参数优化的开发集，测试集为 CWMT2013 官方测试集，开发集与测试集更加详细的信息见表 4、表 5。参数优化的过程中，使用 IBM BLEU 作为优化特征权重的目标函数。按照 CWMT2013 机器翻译评测要求，开发集与测试集性能评价采用多种自动评价标准，包括：BLEU-SBP、BLEU-NIST、NIST、GTM、mWER、mPER、ICT。CWMT2013 基线系统在开发集和测试集上的实验结果见表 6、表 7。从表 6 中可以看出，在汉英新闻领域的翻译中，基于短语的翻译系统在测试集上的翻译性能比基于层次短语的系统的翻译性能高 0.0060 BLEU-SBP。从表 7 中可以看出，在英汉新闻领域的翻译中，层次短语翻译系统的翻译性能略优于短语翻译系统。

表 5. CWMT 英汉新闻领域翻译中数据使用情况。其中“训练数据”中“词汇数”一列中，左侧为英文词汇数，右侧为中文词汇数（M=百万）

| 英汉新闻数据 | 行数 | 词汇数 |
|---|---|---|
| 训练数据 | 4.55 M | 88.27 M/78.83 M |
| Sogou | 24.00 M | 647.48 M |
| 开发集 | 1000 | 25159 |
| 09 测试集 | 1002 | 25231 |
| 11 测试集 | 1001 | 26268 |

表 6. CWMT2013 汉英新闻基线系统翻译性能，开发集、测试集评价均采用大小写敏感（case-sensitive）方式，测试集结果由评测组织方提供

| 汉英新闻 | 数据集 | BLEU4-SBP | BLEU4 | NIST5 | GTM | mWER | mPER | ICT |
|---|---|---|---|---|---|---|---|---|
| 短语 | 开发集 | 0.2572 | 0.2625 | 8.4529 | 0.7573 | 0.6993 | 0.4875 | 0.3308 |
| | 测试集 | 0.2268 | 0.2408 | 7.7339 | 0.7022 | 0.7126 | 0.5055 | 0.3136 |
| 层次短语 | 开发集 | 0.2520 | 0.2591 | 8.4721 | 0.7525 | 0.7157 | 0.4858 | 0.3368 |
| | 测试集 | 0.2208 | 0.2363 | 7.6554 | 0.6992 | 0.7158 | 0.5062 | 0.3187 |

表 7. CWMT2013 英汉新闻基线系统翻译性能，测试集结果由评测组织方提供

| 英汉新闻 | 数据集 | BLEU5-SBP | BLEU5 | BLEU6 | NIST6 | NIST7 | GTM | mWER | mPER | ICT |
|---|---|---|---|---|---|---|---|---|---|---|
| 短语 | 开发集 | 0.3256 | 0.3451 | 0.2801 | 9.6589 | 9.6698 | 0.7833 | 0.6457 | 0.3874 | 0.4145 |
| | 09 测试集 | 0.3325 | 0.3467 | 0.2812 | 9.7548 | 9.7652 | 0.7858 | 0.6258 | 0.3735 | 0.4230 |
| | 11 测试集 | 0.3182 | 0.3292 | 0.2679 | 9.3422 | 9.3517 | 0.7666 | 0.6191 | 0.3810 | 0.3977 |
| 层次短语 | 开发集 | 0.3286 | 0.3460 | 0.2823 | 9.5904 | 9.6017 | 0.7773 | 0.6481 | 0.3890 | 0.4153 |
| | 09 测试集 | 0.3335 | 0.3459 | 0.2812 | 9.7019 | 9.7132 | 0.7863 | 0.6248 | 0.3711 | 0.4325 |
| | 11 测试集 | 0.3211 | 0.3306 | 0.2691 | 9.3050 | 9.3152 | 0.7681 | 0.6173 | 0.3794 | 0.4072 |

在 CWMT2013 机器翻译评测实验过程中，在汉英新闻领域同时验证了对训练数据、开发集、测试集进行数据预处理时，有无数字、时间、日期这类可自动识别的内容进行识别/

[9] 由于 CWMT2013 机器翻译评测只能使用评测组织方指定范围的数据进行训练，故数据预处理仅对数字、时间、日期这类可自动识别内容进行特殊处理

泛化/翻译对机器翻译系统翻译性能的影响。该组实验使用基于短语的翻译模型进行验证，具体实验结果见表 8。从实验结果可以看出，当对数字、时间、日期进行有效的处理时，翻译系统在测试集上的翻译性能与没有进行特殊处理的翻译系统相比，翻译性能上涨 0.0066 BLEU-SBP。

表 8. CWMT2013 汉英新闻领域翻译中，验证数据预处理中数字、时间、日期进行识别/泛化/翻译操作对机器翻译系统性能的影响，开发集、测试集评价均采用大小写敏感（case-sensitive）方式

| 汉英新闻 | 数据集 | BLEU4-SBP | BLEU4 | NIST5 | GTM | mWER | mPER | ICT |
|---|---|---|---|---|---|---|---|---|
| 短语 | 开发集 | 0.2572 | 0.2625 | 8.4529 | 0.7573 | 0.6993 | 0.4875 | 0.3308 |
| | 测试集 | 0.2268 | 0.2408 | 7.7339 | 0.7022 | 0.7126 | 0.5055 | 0.3136 |
| 短语+仅分词 | 开发集 | 0.2495 | 0.2542 | 8.2589 | 0.7582 | 0.7023 | 0.4871 | 0.3176 |
| | 测试集 | 0.2202 | 0.2323 | 7.6835 | 0.7046 | 0.7195 | 0.5141 | 0.3036 |

同时，在实验的过程中，将开发集对应的翻译结果进行数据预处理后混入到语言模型训练数据中，重新训练一个新的 5 元语言模型，同时与基线的短语翻译系统的性能进行比较。实验结果见表 9 中“短语+开发集”行。从实验结果可以看出，将开发集的参考答案混入到语言模型训练数据中会严重的影响参数优化过程，从而导致在测试集上翻译性能下降了 0.0269 BLEU-SBP。在表 9 中“短语+6 元”行，本文同样验证了使用 6 元语言模型的情况。从实验结果可以看出，在 CWMT2013 数据上，使用 6 元的语言模型时，系统翻译性能与基线翻译系统相比并没有明显上涨。从表 9“短语+bleu-sbp”行中可以看出，使用 BLEU-SBP 作为优化特征权重的目标函数，与以 IBM BLEU 作为目标函数优化特征权重的系统相比，翻译系统的翻译性能没有上涨。

表 9. CWMT2013 汉英新闻领域翻译中，在短语翻译系统上，不同方法对翻译性能的影响，开发集、测试集评价均采用大小写敏感（case-sensitive）方式

| 汉英新闻 | 数据集 | BLEU4-SBP | BLEU4 | NIST5 | GTM | mWER | mPER | ICT |
|---|---|---|---|---|---|---|---|---|
| 短语 | 开发集 | 0.2572 | 0.2625 | 8.4529 | 0.7573 | 0.6993 | 0.4875 | 0.3308 |
| | 测试集 | 0.2268 | 0.2408 | 7.7339 | 0.7022 | 0.7126 | 0.5055 | 0.3136 |
| 短语+开发集 | 开发集 | 0.3717 | 0.3806 | 9.5548 | 0.7869 | 0.6454 | 0.4497 | 0.3886 |
| | 测试集 | 0.1999 | 0.2146 | 7.1419 | 0.6807 | 0.7327 | 0.5293 | 0.3013 |
| 短语+6 元 | 开发集 | 0.2541 | 0.2587 | 8.3776 | 0.7601 | 0.6973 | 0.4846 | 0.3228 |
| | 测试集 | 0.2283 | 0.2396 | 7.8197 | 0.7078 | 0.7145 | 0.5028 | 0.3090 |
| 短语+bleu-sbp | 开发集 | 0.2570 | 0.2627 | 8.4620 | 0.7558 | 0.7007 | 0.4868 | 0.3326 |
| | 测试集 | 0.2278 | 0.2427 | 7.7297 | 0.7004 | 0.7105 | 0.5073 | 0.3139 |

表 10. CWMT2013 英汉新闻领域翻译上，使用基于字的语言模型对翻译系统性能的影响

| 英汉新闻 | 数据集 | BLEU5-SBP | BLEU5 | BLEU6 | NIST6 | NIST7 | GTM | mWER | mPER | ICT |
|---|---|---|---|---|---|---|---|---|---|---|
| 短语 | 开发集 | 0.3256 | 0.3451 | 0.2801 | 9.6589 | 9.6698 | 0.7833 | 0.6457 | 0.3874 | 0.4145 |
| | 09 测试集 | 0.3325 | 0.3467 | 0.2812 | 9.7548 | 9.7652 | 0.7858 | 0.6258 | 0.3735 | 0.4230 |
| | 11 测试集 | 0.3182 | 0.3292 | 0.2679 | 9.3422 | 9.3517 | 0.7666 | 0.6191 | 0.3810 | 0.3977 |
| 短语+基于字 | 开发集 | 0.3252 | 0.3435 | 0.2780 | 9.7223 | 9.7329 | 0.7827 | 0.6634 | 0.3919 | 0.4022 |
| | 09 测试集 | 0.3219 | 0.3306 | 0.2659 | 9.7292 | 9.7394 | 0.8010 | 0.6626 | 0.3793 | 0.3776 |
| | 11 测试集 | 0.3088 | 0.3228 | 0.2608 | 9.4813 | 9.4916 | 0.7802 | 0.6590 | 0.3821 | 0.3534 |

在英汉新闻领域的翻译中，本文验证了仅使用基于字的语言模型对翻译系统翻译性能

的影响。具体实验结果见表 10。从表 10 中实验结果可以看出，当仅使用基于字的语言模型时，在开发集上翻译系统的性能与基线系统可比。但是，在 09、11 测试集上，翻译系统的翻译性能分别下降了 0.0106 BLEU-SBP 和 0.0094 BLEU-SBP。从本实验结果看出，在英汉翻译中，仅使用基于字的语言模型无法获得高质量的翻译结果。

## 5 NiuTrans 高级功能

NiuTrans 系统为了方便用户实验，为用户预留了自主配置的接口，通过该接口，用户可改变系统的一些参数，定制个性化的翻译系统。NiuTrans 提供的高级功能归纳如下：

- 如何生成 *n* 最优翻译结果
  修改解码器配置文件中 *nbest* 参数值，配置解码器输出 *n* 个最优翻译结果
- 如何配置 *beam* 大小
  修改解码器配置文件中 *beamsize* 参数值，配置解码器 *beam* 大小
- 如何修改剪枝方法
  修改解码器配置文件中 *usepuncpruning* 与 *usecubepruning* 参数值，配置解码器是否使用相应的剪枝技术
- 如何加速解码
  修改解码器配置文件中 *nthread* 参数值，配置解码器运行过程中启动的线程数
- 如何使用多参考译文
  修改解码器配置文件中参数 *nref* 值，配置解码器 MERT 过程中使用参考译文的个数
- 如何使用高阶的 *N* 元语言模型
  修改语言模型训练-*n* 参数值，同时确保解码器配置文件中 *ngram* 参数值与之相同，完成语言模型的元数配置
- 如何控制短语翻译表大小
  修改短语翻译模型训练配置文件中参数 *Max-Source-Phrase-Size* 与 *Max-Target-Phrase-Size* 值，或设置 *Phrase-Cut-Off* 参数，控制生成短语翻译表的大小
- 如何使用更多语料训练 ME 调序模型
  模型训练配置文件中 *ME-max-sample-num* 参数控制训练 ME 调序模型使用数据的大小，值越大则使用数据越多，模型训练则越充分，同时对内存要求越高
- 如何使用自定义特征
  用户可通过在短语翻译表的每行行尾添加新的特征，同时修改解码器配置文件参数 *freefeature* 与 *tablefeature* 值，达到添加自定义特征的目的

上述高级功能均可通过对解码器配置文件以及模型训练配置文件的修改得以实现[10]。

## 6 总结与展望

NiuTrans 在统一框架下支持基于短语的翻译模型，基于层次短语的翻译模型，基于句法（树到串/串到树/树到树）的翻译模型。在每种翻译模型中，均包含完整的模型训练和解码部分。除此之外，NiuTrans 提供中英文数据预处理工具，语言模型工具，以及翻译后处理工具等。经上文大量实验证实，NiuTrans 是一套翻译性能优异的开源统计机器翻译系统。

NiuTrans 自 2011 年 7 月开放源码至今，已经度过两年。在这两年多的时间里，NiuTrans 已经被 50 多个国家的高校、研究机构、企业、以及个人下载 1500+次。NiuTrans 用户给本

[10] http://www.nlplab.com/NiuPlan/NiuTransAdvancedUsage.html

团队提供很多有用的建议，通过与用户的不断交流，NiuTrans 系统不断的进行升级与 bug 修复，至今 NiuTrans 已经升级了 7 个版本。

未来，NiuTrans 团队还将继续完善本开源系统的各方面功能：

- 增加新的机器翻译技术
- 增加数据分析工具
- 允许用户进行二次开发
- 对企业级用户使用自己的数据构建翻译私有云平台提供技术支持，即 NiuTrans Server 系统开放使用
- 对第三方软件开发者开放 NiuTrans API

目前，NiuTrans 团队正在利用 NiuTrans 技术帮助企业级用户利用自有的数据，构建私有翻译云，提供专业化、个性化的翻译服务。未来的 NiuTrans，将成为一套不单是为学术界服务的系统，更将是为工业界服务的平台。

## 参考文献


[1] Philipp Koehn, Fran J. Och and Daniel Marcu. Statistical phrase-based translation[C]//Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1. Association for Computational Linguistics, 2003: 48-54.

[2] Franz J. Och and Heymann Ney. The alignment template approach to statistical machine translation[J]. Computational Linguistics, 2004, 30(4): 417-449.

[3] David Chiang. A hierarchical phrase-based model for statistical machine translation[C]//Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics. Association for Computational Linguistics, 2005: 263-270.

[4] David Chiang. Hierarchical phrase-based translation[J]. Computational Linguistics, 2007, 33(2): 201-228.

[5] Yang Liu, Qun Liu and Shouxun Lin. Tree-to-string alignment template for statistical machine translation[C]//Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics. Association for Computational Linguistics, 2006: 609-616.

[6] Michel Galley, Mark Hopkins, Kevin Knight and Daniel Marcu. What's in a Translation Rule?[C]// Proceedings of HLT-NAACL. Association for Computational Linguistics, 2004: 273-280.

[7] Daniel Marcu, Wei Wang, Abdessamad Echihabi and Kevin Knight. SPMT: Statistical machine translation with syntactified target language phrases[C]//Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, 2006: 44-52.

[8] Yuan Ding and Martha Palmer. Machine translation using probabilistic synchronous dependency insertion grammars[C]//Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics. Association for Computational Linguistics, 2005: 541-548.

[9] Brooke Cowan, Ivona Kučerová and Michael Collins. A discriminative model for tree-to-tree translation[C]//Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, 2006: 232-241.

[10] Tong Xiao, Jingbo Zhu, Hao Zhang and Qiang Li. NiuTrans: an open source toolkit for phrase-based and syntax-based machine translation[C]//Proceedings of the ACL 2012 System Demonstrations. Association for Computational Linguistics, 2012: 19-24.

[11] Franz J. Och. Minimum error rate training in statistical machine translation[C]//Proceedings of the 41st Annual Meeting on Association for Computational Linguistics-Volume 1. Association for Computational Linguistics, 2003: 160-167.

[12] Liang Huang and David Chiang. Forest rescoring: Faster decoding with integrated language models[C]// Proceedings of the 45th Annual Meeting on Association for Computational Linguistics. Association for Computational Linguistics, 2007: 144–151.

[13] Franz J. Och and Heymann Ney. Discriminative training and maximum entropy models for statistical machine translation[C]//Proceedings of the 40th Annual Meeting on Association for Computational Linguistics. Association for Computational Linguistics, 2002: 295-302.

[14] Adam Pauls and Dan Klein. Faster and Smaller N-Gram Language Models[C]//Proceedings of the 49th Annual Meeting on Association for Computational Linguistics. Association for Computational Linguistics, 2011: 258–267.

[15] Deyi Xiong, Qun Liu and Shouxun Lin. Maximum Entropy Based Phrase Reordering Model for Statistical Machine Translation[C]//Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the ACL. Association for Computational Linguistics, 2006: 521–528.

[16] Christoph Tillman. A unigram orientation model for statistical machine translation[C]//Proceedings of HLT-NAACL 2004: Short Papers. Association for Computational Linguistics, 2004: 101-104.

[17] Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondřej Bojar, Alexandra Constantin and Evan Herbst. Moses: Open source toolkit for statistical machine translation[C]//Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions. Association for Computational Linguistics, 2007: 177-180.

[18] Michel Galley and Christopher D. Manning. A simple and effective hierarchical phrase reordering model[C]//Proceedings of the Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, 2008: 848-856.

[19] Dekai Wu. Stochastic inversion transduction grammars and bilingual parsing of parallel corpora[J]. Computational Linguistics, 1997, 23(3): 377-403.

[20] Daniel H. Younger. Recognition and parsing of context-free languages in time $n^3$[J]. Information and Control, 1967, 10(2):189-208.

[21] Michel Galley, Jonathan Graehl, Kevin Knight, Daniel Marcu, Steve DeNeefe, Wei Wang and Ignacio Thayer. Scalable inference and training of context-rich syntactic translation models[C]//Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics. Association for Computational Linguistics, 2006: 961-968.

[22] Jason Eisner. Learning non-isomorphic tree mappings for machine translation[C]//Proceedings of the 41st Annual Meeting on Association for Computational Linguistics-Volume 2. Association for Computational Linguistics, 2003: 205-208.

[23] Kishore Papineni, Salim Roukos, Todd Ward and Weijing Zhu. BLEU: a method for automatic evaluation of machine translation[C]//Proceedings of the 40th annual meeting on association for computational linguistics. Association for Computational Linguistics, 2002: 311-318.

[24] Wei Wang, Kevin Knight and Daniel Marcu. Binarizing syntax trees to improve syntax-based machine translation accuracy[C]//Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning. Association for Computational Linguistics, 2007: 746–754.

[25] Haitao Mi, Liang Huang and Qun Liu. Forest-based translation[C]//Proceedings of ACL-08: HLT. Association for Computational Linguistics, 2008: 192-199.

[26] Jingbo Zhu and Tong Xiao. Improving decoding generalization for tree-to-string translation[C]//Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers-Volume 2. Association for Computational Linguistics, 2011: 418-423.

[27] David Chiang. Learning to translate with source and target syntax[C]//Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics. Association for Computational Linguistics, 2010: 1443-1452.