论文发表流程

确定方向	统计机器翻译
确定问题	利用句法对长距离调序建模
确定思路	将树到串对泛化为树到串模板
确定方法	规则抽取，搜索算法
实验验证	数据集、基线系统、评价指标
撰写论文	投稿ACL

解决问题

思维独立性

先思考，再去查文献相互印证

语言学意义

具有语言学理论的支撑，符合语言学角度的直觉

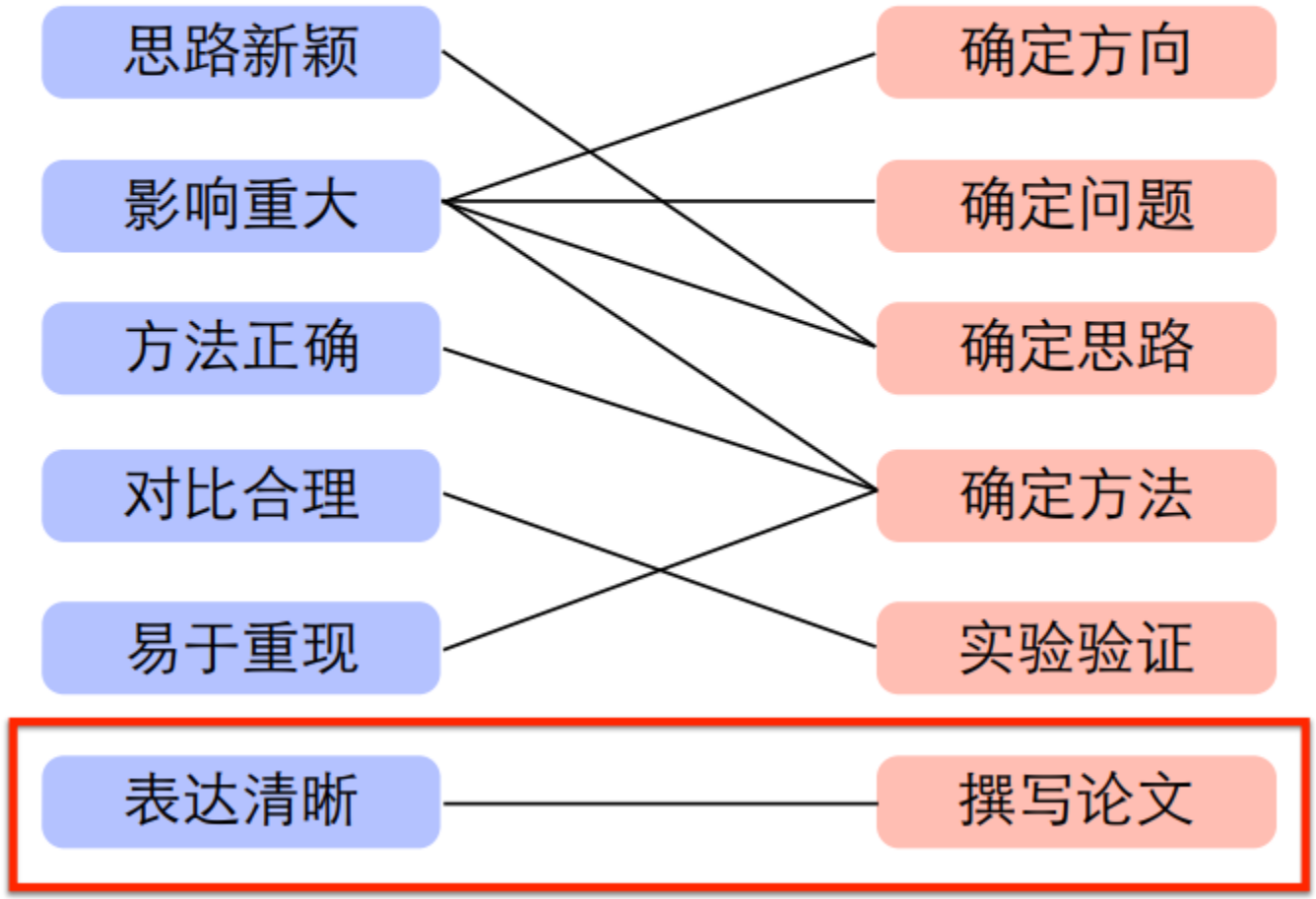
数学意义

使用数学工具做形式化，不臆造数学公式

简洁优美

简单、干净、优美

写论文时什么最重要?



全心全意为读者服务

信息的呈现符合读者的认知惯性

深入浅出，引人入胜，让读者快速找到想要的信息

尽量降低读者的理解难度

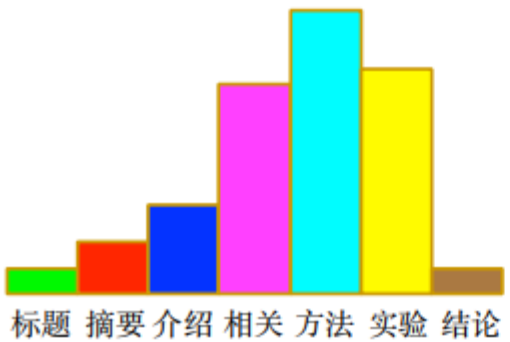
合理地综合使用信息元素：图>曲线>表>正文>公式

尽量提高读者阅读时的愉悦感

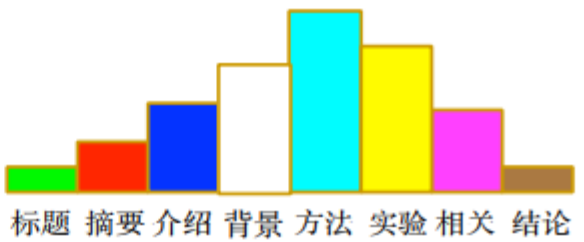
思想新颖、组织合理、逻辑严密
论证充分、文笔优美、排版美观

降低信息理解难度是关键

1 介绍
2 相关工作
3 方法
4 实验
5 结论



1 介绍
2 背景
3 方法
4 实验
5 相关工作
6 结论



摘要

- 几句话概括你的工作
- 误区
  - 力图把所有细节都说清楚
  - 用很专业的术语来描述
  - 出现数学符号

用语要简单，让外行能看懂

例子

问题是什么

我们大概怎么做的

Abstract

Conventional n-best reranking techniques often suffer from the limited scope of the n-best list, which rules out many potentially good alternatives. We instead propose *forest reranking*, a method that reranks a packed forest of exponentially many parses. Since exact inference is intractable with non-local features, we present an approximate algorithm inspired by forest rescoring that makes discriminative training practical over the whole Treebank. Our final result, an F-score of 91.7, outperforms both 50-best and 100-best reranking baselines, and is better than any previously reported systems trained on the Treebank.

我们做了什么

我们做得挺不错!

Liang Huang. **Forest Reranking: Discriminative Parsing with Non-Local Features**. In *ACL 2008*.

介绍的写法

- 比题目和摘要更进一步，用几段话说清你的工作
- 要点是充分论证你所做工作的必要性和重要性，要让审稿人认同并迫不及待想往下看。
- 行文逻辑严密，论证充分

逻辑

- 常见的逻辑
  - 说明问题是什么
  - 简单罗列前人工作
  - 描述我们的工作
- 更好的逻辑
  - 说明问题是什么
  - 目前最好的工作面临什么挑战
  - 我们的方法能缓解上述挑战

例子

Improving Tree-to-Tree Translation with Packed Forests

Yang Liu and **Yajuan Lü** and **Qun Liu**
Key Laboratory of Intelligent Information Processing
Institute of Computing Technology
Chinese Academy of Sciences
P.O. Box 2704, Beijing 100190, China
{yliu,lvyajuan,liuqun}@ict.ac.cn

Abstract

Current tree-to-tree models suffer from parsing errors as they usually use only 1-best parses for rule extraction and decoding. We instead propose a forest-based tree-to-tree model that uses packed forests. The model is based on a probabilistic synchronous tree substitution grammar (STSG), which can be learned from aligned forest pairs automatically. The decoder finds ways of decomposing trees in the source forest into elementary trees using the source projection of STSG while building target forest in parallel. Comparable to the state-of-the-art phrase-based system Moses, using packed forests in tree-to-tree translation results in a significant absolute improvement of 3.6 BLEU points over using 1-best trees.

1 Introduction

Approaches to syntax-based statistical machine translation make use of parallel data with syntactic annotations, either in the form of phrase structure trees or dependency trees. They can be roughly divided into three categories: *string-to-tree* models (e.g., (Galley et al., 2006; Marcu et al., 2006; Shen et al., 2008)), *tree-to-string* models (e.g., (Liu et al., 2006; Huang et al., 2006)), and *tree-to-tree* models (e.g., (Eisner, 2003; Ding and Palmer, 2005; Cowan et al., 2006; Zhang et al., 2008)). By modeling the syntax of both source and target languages, tree-to-tree approaches have the potential benefit of providing rules linguistically better motivated. However, while string-to-tree and tree-to-string models demonstrate promising results in empirical evaluations, tree-to-tree models have still been underachieving.

We believe that tree-to-tree models face two major challenges. First, tree-to-tree models are more vulnerable to parsing errors. Obtaining syntactic annotations in quantity usually entails running automatic parsers on a parallel corpus. As the amount and domain of the data used to train parsers are relatively limited, parsers will inevitably output ill-formed trees when handling real-world text. Guided by such noisy syntactic information, syntax-based models that rely on 1-best parses are prone to learn noisy translation rules in training phase and produce degenerate translations in decoding phase (Quirk and Corston-Oliver, 2006). This situation aggravates for tree-to-tree models that use syntax on both sides.

Second, tree-to-tree rules provide poorer rule coverage. As a tree-to-tree rule requires that there must be trees on both sides, tree-to-tree models lose a larger amount of linguistically unmotivated mappings. Studies reveal that the absence of such non-syntactic mappings will impair translation quality dramatically (Marcu et al., 2006; Liu et al., 2007; DeNeefe et al., 2007; Zhang et al., 2008).

Compactly encoding exponentially many parses, *packed forests* prove to be an excellent fit for alleviating the above two problems (Mi et al., 2008; Mi and Huang, 2008). In this paper, we propose a forest-based tree-to-tree model. To learn STSG rules from aligned forest pairs, we introduce a series of notions for identifying minimal tree-to-tree rules. Our decoder first converts the source forest to a translation forest and then finds the best derivation that has the source yield of one source tree in the forest. Comparable to Moses, our forest-based tree-to-tree model achieves an absolute improvement of 3.6 BLEU points over conventional tree-based model.

Proceedings of the 47th Annual Meeting of the ACL and the 4th IJCNLP of the AFNLP, pages 558–566,
Suntec, Singapore, 2-7 August 2009.

问题

Approaches to syntax-based statistical machine translation make use of parallel data with syntactic annotations, either in the form of phrase structure trees or dependency trees. They can be roughly divided into three categories: *string-to-tree* models (e.g., (Galley et al., 2006; Marcu et al., 2006; Shen et al., 2008)), *tree-to-string* models (e.g., (Liu et al., 2006; Huang et al., 2006)), and *tree-to-tree* models (e.g., (Eisner, 2003; Ding and Palmer, 2005; Cowan et al., 2006; Zhang et al., 2008)). By modeling the syntax of both source and target languages, tree-to-tree approaches have the potential benefit of providing rules linguistically better motivated. However, while string-to-tree and tree-to-string models demonstrate promising results in empirical evaluations, tree-to-tree models have still been underachieving.

Yang Liu, Yajuan Lv, and Qun Liu. **Improving Tree-to-Tree Translation with Packed Forests**. In *ACL 2009*.

例子

Improving Tree-to-Tree Translation with Packed Forests

Yang Liu and **Yajuan Lü** and **Qun Liu**
Key Laboratory of Intelligent Information Processing
Institute of Computing Technology
Chinese Academy of Sciences
P.O. Box 2704, Beijing 100190, China
{yliu,lvyajuan,liuqun}@ict.ac.cn

Abstract

Current tree-to-tree models suffer from parsing errors as they usually use only 1-best parses for rule extraction and decoding. We instead propose a forest-based tree-to-tree model that uses packed forests. The model is based on a probabilistic synchronous tree substitution grammar (STSG), which can be learned from aligned forest pairs automatically. The decoder finds ways of decomposing trees in the source forest into elementary trees using the source projection of STSG while building target forest in parallel. Comparable to the state-of-the-art phrase-based system Moses, using packed forests in tree-to-tree translation results in a significant absolute improvement of 3.6 BLEU points over using 1-best trees.

1 Introduction

Approaches to syntax-based statistical machine translation make use of parallel data with syntactic annotations, either in the form of phrase structure trees or dependency trees. They can be roughly divided into three categories: *string-to-tree* models (e.g., (Galley et al., 2006; Marcu et al., 2006; Shen et al., 2008)), *tree-to-string* models (e.g., (Liu et al., 2006; Huang et al., 2006)), and *tree-to-tree* models (e.g., (Eisner, 2003; Ding and Palmer, 2005; Cowan et al., 2006; Zhang et al., 2008)). By modeling the syntax of both source and target languages, tree-to-tree approaches have the potential benefit of providing rules linguistically better motivated. However, while string-to-tree and tree-to-string models demonstrate promising results in empirical evaluations, tree-to-tree models have still been underachieving.

We believe that tree-to-tree models face two major challenges. First, tree-to-tree models are more vulnerable to parsing errors. Obtaining syntactic annotations in quantity usually entails running automatic parsers on a parallel corpus. As the amount and domain of the data used to train parsers are relatively limited, parsers will inevitably output ill-formed trees when handling real-world text. Guided by such noisy syntactic information, syntax-based models that rely on 1-best parses are prone to learn noisy translation rules in training phase and produce degenerate translations in decoding phase (Quirk and Corston-Oliver, 2006). This situation aggravates for tree-to-tree models that use syntax on both sides.

Second, tree-to-tree rules provide poorer rule coverage. As a tree-to-tree rule requires that there must be trees on both sides, tree-to-tree models lose a larger amount of linguistically unmotivated mappings. Studies reveal that the absence of such non-syntactic mappings will impair translation quality dramatically (Marcu et al., 2006; Liu et al., 2007; DeNeefe et al., 2007; Zhang et al., 2008).

Compactly encoding exponentially many parses, *packed forests* prove to be an excellent fit for alleviating the above two problems (Mi et al., 2008; Mi and Huang, 2008). In this paper, we propose a forest-based tree-to-tree model. To learn STSG rules from aligned forest pairs, we introduce a series of notions for identifying minimal tree-to-tree rules. Our decoder first converts the source forest to a translation forest and then finds the best derivation that has the source yield of one source tree in the forest. Comparable to Moses, our forest-based tree-to-tree model achieves an absolute improvement of 3.6 BLEU points over conventional tree-based model.

Proceedings of the 47th Annual Meeting of the ACL and the 4th IJCNLP of the AFNLP, pages 558–566,
Suntec, Singapore, 2-7 August 2009.

挑战

We believe that tree-to-tree models face two major challenges. First, tree-to-tree models are more vulnerable to parsing errors. Obtaining syntactic annotations in quantity usually entails running automatic parsers on a parallel corpus. As the amount and domain of the data used to train parsers are relatively limited, parsers will inevitably output ill-formed trees when handling real-world text. Guided by such noisy syntactic information, syntax-based models that rely on 1-best parses are prone to learn noisy translation rules in training phase and produce degenerate translations in decoding phase (Quirk and Corston-Oliver, 2006). This situation aggravates for tree-to-tree models that use syntax on both sides.

Second, tree-to-tree rules provide poorer rule coverage. As a tree-to-tree rule requires that there must be trees on both sides, tree-to-tree models lose a larger amount of linguistically unmotivated mappings. Studies reveal that the absence of such non-syntactic mappings will impair translation quality dramatically (Marcu et al., 2006; Liu et al., 2007; DeNeefe et al., 2007; Zhang et al., 2008).

Yang Liu, Yajuan Lv, and Qun Liu. **Improving Tree-to-Tree Translation with Packed Forests**. In *ACL 2009*.

例子

Improving Tree-to-Tree Translation with Packed Forests

Yang Liu and **Yajuan Lü** and **Qun Liu**
Key Laboratory of Intelligent Information Processing
Institute of Computing Technology
Chinese Academy of Sciences
P.O. Box 2704, Beijing 100190, China
{yliu,lvyajuan,liuqun}@ict.ac.cn

Abstract

Current tree-to-tree models suffer from parsing errors as they usually use only 1-best parses for rule extraction and decoding. We instead propose a forest-based tree-to-tree model that uses packed forests. The model is based on a probabilistic synchronous tree substitution grammar (STSG), which can be learned from aligned forest pairs automatically. The decoder finds ways of decomposing trees in the source forest into elementary trees using the source projection of STSG while building target forest in parallel. Comparable to the state-of-the-art phrase-based system Moses, using packed forests in tree-to-tree translation results in a significant absolute improvement of 3.6 BLEU points over using 1-best trees.

1 Introduction

Approaches to syntax-based statistical machine translation make use of parallel data with syntactic annotations, either in the form of phrase structure trees or dependency trees. They can be roughly divided into three categories: *string-to-tree* models (e.g., (Galley et al., 2006; Marcu et al., 2006; Shen et al., 2008)), *tree-to-string* models (e.g., (Liu et al., 2006; Huang et al., 2006)), and *tree-to-tree* models (e.g., (Eisner, 2003; Ding and Palmer, 2005; Cowan et al., 2006; Zhang et al., 2008)). By modeling the syntax of both source and target languages, tree-to-tree approaches have the potential benefit of providing rules linguistically better motivated. However, while string-to-tree and tree-to-string models demonstrate promising results in empirical evaluations, tree-to-tree models have still been underachieving.

We believe that tree-to-tree models face two major challenges. First, tree-to-tree models are more vulnerable to parsing errors. Obtaining syntactic annotations in quantity usually entails running automatic parsers on a parallel corpus. As the amount and domain of the data used to train parsers are relatively limited, parsers will inevitably output ill-formed trees when handling real-world text. Guided by such noisy syntactic information, syntax-based models that rely on 1-best parses are prone to learn noisy translation rules in training phase and produce degenerate translations in decoding phase (Quirk and Corston-Oliver, 2006). This situation aggravates for tree-to-tree models that use syntax on both sides.

Second, tree-to-tree rules provide poorer rule coverage. As a tree-to-tree rule requires that there must be trees on both sides, tree-to-tree models lose a larger amount of linguistically unmotivated mappings. Studies reveal that the absence of such non-syntactic mappings will impair translation quality dramatically (Marcu et al., 2006; Liu et al., 2007; DeNeefe et al., 2007; Zhang et al., 2008).

Compactly encoding exponentially many parses, *packed forests* prove to be an excellent fit for alleviating the above two problems (Mi et al., 2008; Mi and Huang, 2008). In this paper, we propose a forest-based tree-to-tree model. To learn STSG rules from aligned forest pairs, we introduce a series of notions for identifying minimal tree-to-tree rules. Our decoder first converts the source forest to a translation forest and then finds the best derivation that has the source yield of one source tree in the forest. Comparable to Moses, our forest-based tree-to-tree model achieves an absolute improvement of 3.6 BLEU points over conventional tree-based model.

Proceedings of the 47th Annual Meeting of the ACL and the 4th IJCNLP of the AFNLP, pages 558–566,
Suntec, Singapore, 2-7 August 2009.

我们的工作

Compactly encoding exponentially many parses, *packed forests* prove to be an excellent fit for alleviating the above two problems (Mi et al., 2008; Mi and Huang, 2008). In this paper, we propose a forest-based tree-to-tree model. To learn STSG rules from aligned forest pairs, we introduce a series of notions for identifying minimal tree-to-tree rules. Our decoder first converts the source forest to a translation forest and then finds the best derivation that has the source yield of one source tree in the forest. Comparable to Moses, our forest-based tree-to-tree model achieves an absolute improvement of 3.6 BLEU points over conventional tree-based model.

Yang Liu, Yajuan Lv, and Qun Liu. **Improving Tree-to-Tree Translation with Packed Forests**. In *ACL 2009*.

段落的写法

- 每个段落有个论断性的中心句
- 其余部分都是支撑句，围绕中心句展开论证
  - 前人工作
  - 具体数据
- 支撑句之间可分类组织
- 段尾可以加上衔接句

中心句与支撑句

The need to segment and label sequences arises in many different problems in several scientific fields. Hidden Markov models (HMMs) and stochastic grammars are well understood and widely used probabilistic models for such problems. In computational biology, HMMs and stochastic grammars have been successfully used to align biological sequences, find sequences homologous to a known evolutionary family, and analyze RNA secondary structure (Durbin et al., 1998). In computational linguistics and computer science, HMMs and stochastic grammars have been applied to a wide variety of problems in text and speech processing, including topic segmentation, part-of-speech (POS) tagging, information extraction, and syntactic disambiguation (Manning & Schütze, 1999).

John Lafferty, Andrew McCallum, and Fernando Pereira. **Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data**. In *ICML 2003*.

中心句与支撑句

The need to segment and label sequences arises in many different problems in several scientific fields. Hidden Markov models (HMMs) and stochastic grammars are well understood and widely used probabilistic models for such problems. In computational biology, HMMs and stochastic grammars have been successfully used to align biological sequences, find sequences homologous to a known evolutionary family, and analyze RNA secondary structure (Durbin et al., 1998). In computational linguistics and computer science, HMMs and stochastic grammars have been applied to a wide variety of problems in text and speech processing, including topic segmentation, part-of-speech (POS) tagging, information extraction, and syntactic disambiguation (Manning & Schütze, 1999).

John Lafferty, Andrew McCallum, and Fernando Pereira. **Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data**. In *ICML 2003*.

中心句与支撑句

The need to segment and label sequences arises in many different problems in several scientific fields. Hidden Markov models (HMMs) and stochastic grammars are well understood and widely used probabilistic models for such problems. In computational biology, HMMs and stochastic grammars have been successfully used to align biological sequences, find sequences homologous to a known evolutionary family, and analyze RNA secondary structure (Durbin et al., 1998). In computational linguistics and computer science, HMMs and stochastic grammars have been applied to a wide variety of problems in text and speech processing, including topic segmentation, part-of-speech (POS) tagging, information extraction, and syntactic disambiguation (Manning & Schütze, 1999).

John Lafferty, Andrew McCallum, and Fernando Pereira. **Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data**. In *ICML 2003*.

中心句与支撑句

The need to segment and label sequences arises in many different problems in several scientific fields. Hidden Markov models (HMMs) and stochastic grammars are well understood and widely used probabilistic models for such problems. In computational biology, HMMs and stochastic grammars have been successfully used to align biological sequences, find sequences homologous to a known evolutionary family, and analyze RNA secondary structure (Durbin et al., 1998). In computational linguistics and computer science, HMMs and stochastic grammars have been applied to a wide variety of problems in text and speech processing, including topic segmentation, part-of-speech (POS) tagging, information extraction, and syntactic disambiguation (Manning & Schütze, 1999).

John Lafferty, Andrew McCallum, and Fernando Pereira. **Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data**. In *ICML 2003*.

中心句与支撑句

We believe that it is important to make available to syntax-based models all the bilingual phrases that are typically available to phrase-based models. On one hand, phrases have been proven to be a simple and powerful mechanism for machine translation. They excel at capturing translations of short idioms, providing local re-ordering decisions, and incorporating context information straightforwardly. Chiang (2005) shows significant improvement by keeping the strengths of phrases while incorporating syntax into statistical translation. On the other hand, the performance of linguistically syntax-based models can be hindered by making use of only syntactic phrase pairs. Studies reveal that linguistically syntax-based models are sensitive to syntactic analysis (Quirk and Corston-Oliver, 2006), which is still not reliable enough to handle real-world texts due to limited size and domain of training data.

Yang Liu, Yajuan Lv, and Qun Liu. **Improving Tree-to-Tree Translation with Packed Forests**. In *ACL 2009*.

衔接句

Finding word alignments between parallel texts, however, is still far from a trivial work due to the diversity of natural languages. For example, the alignment of words within idiomatic expressions, free translations, and missing content or function words is problematic. When two languages widely differ in word order, finding word alignments is especially hard. Therefore, it is necessary to incorporate all useful linguistic information to alleviate these problems.

Tiedemann (2003) introduced a word alignment approach based on combination of association clues. Clues combination is done by disjunction of single clues, which are defined as probabilities of associations. The crucial assumption of clue combination that clues are independent of each other, however, is not always true. Och and Ney (2003) proposed

Yang Liu, Qun Liu, and Shogun Lin. **Log-Linear Models for Word Alignment**. In *ACL 2005*.

支撑句要论证严密

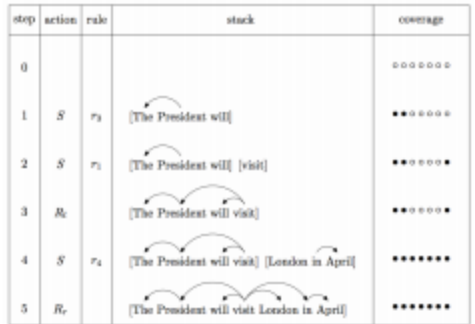
compute within the baseline system. But despite its apparent success, there remains a major drawback: this method suffers from the limited scope of the n-best list, which rules out many potentially good alternatives. For example 41% of the correct parses were not in the candidates of ~30-best parses in (Collins, 2000). This situation becomes worse with longer sentences because the number of possible interpretations usually grows exponentially with the sentence length. As a result, we often see very few variations among the n-best trees, for example, 50-best trees typically just represent a combination of 5 to 6 binary ambiguities (since $2^5 < 50 < 2^6$).

Liang Huang. **Forest Reranking: Discriminative Parsing with Non-Local Features**. In *ACL 2008*.

新技巧

- 在首页放置一个图或者表，让读者一目了然你所做的工作；
- 不要去写“This paper is organized as follows. Section 2 …”，而是直接列出自己的贡献。

信息元素的易理解度



图

★

System	Setting	English–French	Chinese–English
GIZA++	Model 4 s2t	7.7	20.9
	Model 4 t2s	9.2	30.3
	Intersection	6.8	21.8
	Union	9.6	28.1
	Refined method	5.9	18.4
Cross-EM	HMM, joint	5.1	18.9
Vigne	Model 4 s2t	7.8	20.5
	+Model 4 t2s	5.6	18.3
	+link count	5.5	17.7
	+cross count	5.4	17.6
	+neighbor count	5.2	17.4
	+exact match	5.3	-
	+linked word count	5.2	17.3
	+bilingual dictionary	-	17.1
	+link co-occurrence count (GIZA++)	5.1	16.3
	+link co-occurrence count (Cross-EM)	4.0	15.7

表格

★★

Shift-reduce parsing is efficient but suffers from parsing errors caused by syntactic ambiguity. Figure 3 shows two (partial) derivations for a dependency tree. Consider the item on the top, the algorithm can either apply a shift action to move a new item or apply a reduce left action to obtain a bigger structure. This is often referred to as **conflict** in the shift-reduce dependency parsing literature (Huang et al., 2009). In this work, the shift-reduce parser faces four types of conflicts:

正文

★★★

$$\frac{\partial L(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}_k} = \sum_{i=1}^{I} \sum_{\mathbf{y} \in \mathcal{Y}(\mathbf{x}^{(i)})} P(\mathbf{y}|\mathbf{x}^{(i)};\boldsymbol{\theta}) \boldsymbol{\phi}_k(\mathbf{x}^{(i)}, \mathbf{y}) - \sum_{\mathbf{x} \in \mathcal{X}} \sum_{\mathbf{y} \in \mathcal{Y}(\mathbf{x})} P(\mathbf{x}, \mathbf{y}; \boldsymbol{\theta}) \boldsymbol{\phi}_k(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^{I} \mathbb{E}_{\mathbf{y}|\mathbf{x}^{(i)};\boldsymbol{\theta}}[\boldsymbol{\phi}_k(\mathbf{x}^{(i)}, \mathbf{y})] - \mathbb{E}_{\mathbf{x},\mathbf{y};\boldsymbol{\theta}}[\boldsymbol{\phi}_k(\mathbf{x}, \mathbf{y})]$$

公式

★★★★

Algorithm 1 A beam search algorithm for word alignment

```
1: procedure ALIGN(f, e)
2:     open ← ∅                                   ▷ a list of active alignments
3:     N ← ∅                                      ▷ n-best list
4:     a ← ∅                                      ▷ begin with an empty alignment
5:     ADD(open, a, β, b)                         ▷ initialize the list
6:     while open ≠ ∅ do
7:         closed ← ∅                             ▷ a list of promising alignments
8:         for all a ∈ open do
9:             for all (i ∈ [1 × I − a do         ▷ enumerate all possible new links
10:                a' ← a ∪ {i}                   ▷ produce a new alignment
11:                g ← GAIN(f, e, a, i)           ▷ compute the link gain
12:                if g > 0 then                  ▷ ensure that the score will increase
13:                    ADD(closed, a', β, b)      ▷ update promising alignments
14:                end if
15:                ADD(N, a', 0, n)               ▷ update n-best list
16:            end for
17:        end for
18:        open ← closed                          ▷ update active alignments
19:    end while
20:    return N                                   ▷ return n-best list
21: end procedure
```

算法

★★★★★

Proof of Theorem 1: Let $\bar{\alpha}^k$ be the weights before the k'th mistake is made. It follows that $\bar{\alpha}^1 = 0$. Suppose the k'th mistake is made at the i'th example. Take z to the output proposed at this example, $z = \arg\max_{y \in \mathbf{GEN}(x_i)} \Phi(x_i, y) \cdot \bar{\alpha}^k$. It follows from the algorithm updates that $\bar{\alpha}^{k+1} = \bar{\alpha}^k + \Phi(x_i, y_i) - \Phi(x_i, z)$. We take inner products of both sides with the vector $\mathbf{U}$:

$$\mathbf{U} \cdot \bar{\alpha}^{k+1} = \mathbf{U} \cdot \bar{\alpha}^k + \mathbf{U} \cdot \Phi(x_i, y_i) - \mathbf{U} \cdot \Phi(x_i, z) \geq \mathbf{U} \cdot \bar{\alpha}^k + \delta$$

where the inequality follows because of the property of $\mathbf{U}$ assumed in Eq. 3. Because $\bar{\alpha}^1 = 0$, and therefore $\mathbf{U} \cdot \bar{\alpha}^1 = 0$, it follows by induction on k that for all k, $\mathbf{U} \cdot \bar{\alpha}^{k+1} \geq k\delta$. Because $\mathbf{U} \cdot \bar{\alpha}^{k+1} \leq ||\mathbf{U}|| \; ||\bar{\alpha}^{k+1}||$, it follows that $||\bar{\alpha}^{k+1}|| \geq k\delta$.

证明

★★★★★★

眼动仪的佐证

图片来自清华大学刘奕群

读者潜意识里优先选择易理解度高的信息元素

首页加图表

Forest Reranking: Discriminative Parsing with Non-Local Features*

Liang Huang
University of Pennsylvania
Philadelphia, PA 19104
lhuang3@cis.upenn.edu

Abstract

Conventional n-best reranking techniques often suffer from the limited scope of the n-best list, which rules out many potentially good alternatives. We instead propose *forest reranking*, a method that reranks a packed forest of exponentially many parses. Since exact inference is intractable with non-local features, we present an approximate algorithm inspired by forest rescoring that makes discriminative training practical over the whole Treebank. Our final result, an F-score of 91.7, outperforms both 50-best and 100-best reranking baselines, and is better than any previously reported systems trained on the Treebank.

	local	*non-local*
conventional reranking	only at the root	
DP-based discrim. parsing	exact	N/A
this work: forest-reranking	exact	*on-the-fly*

Table 1: Comparison of various approaches for incorporating local and non-local features.

1 Introduction

Discriminative reranking has become a popular technique for many NLP problems, in particular, parsing (Collins, 2000) and machine translation (Shen et al., 2005). Typically, this method first generates a list of top-n candidates from a baseline system, and then reranks this n-best list with arbitrary features that are not computable or intractable to

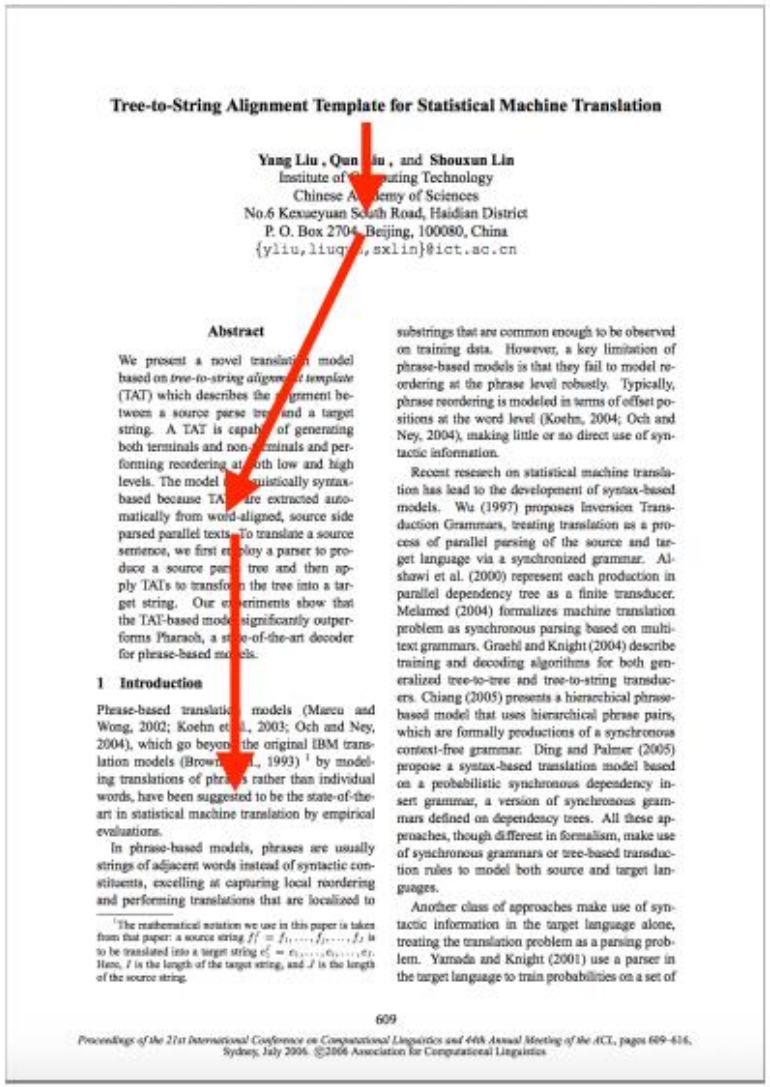
sentence length. As a result, we often see very few variations among the n-best trees, for example, 50-best trees typically just represent a combination of 5 to 6 binary ambiguities (since $2^5 < 50 < 2^6$).

Alternatively, discriminative parsing is tractable with exact and efficient search based on dynamic programming (DP) if all features are restricted to be *local*, that is, only looking at a local window within the factored search space (Taskar et al., 2004; McDonald et al., 2005). However, we miss the benefits of non-local features that are not representable here.

Ideally, we would wish to combine the merits of both approaches, where an efficient inference algorithm could integrate both local and non-local features. Unfortunately, exact search is intractable (at least in theory) for features with unbounded scope.

Liang Huang. **Forest Reranking: Discriminative Parsing with Non-Local Features**. In *ACL 2008*.

信息流的变化



Tree-to-String Alignment Template for Statistical Machine Translation

Yang Liu , Qun Liu , and Shouxun Lin
Institute of Computing Technology
Chinese Academy of Sciences
No.6 Kexueyuan South Road, Haidian District
P. O. Box 2704, Beijing, 100080, China
{yliu,liuqun,sxlin}@ict.ac.cn

Abstract

We present a novel translation model based on *tree-to-string alignment template* (TAT) which describes the alignment between a source parse tree and a target string. A TAT is capable of generating both terminals and non-terminals and performing reordering at both low and high levels. The model is linguistically syntax-based because TATs are extracted automatically from word-aligned, source side parsed parallel texts. To translate a source sentence, we first employ a parser to produce a source parse tree and then apply TATs to transform the tree into a target string. Our experiments show that the TAT-based model significantly outperforms Pharaoh, a state-of-the-art decoder for phrase-based models.

1 Introduction

Phrase-based translation models (Marcu and Wong, 2002; Koehn et al., 2003; Och and Ney, 2004), which go beyond the original IBM translation models (Brown et al., 1993) [1] by modeling translations of phrases rather than individual words, have been suggested to be the state-of-the-art in statistical machine translation by empirical evaluations.

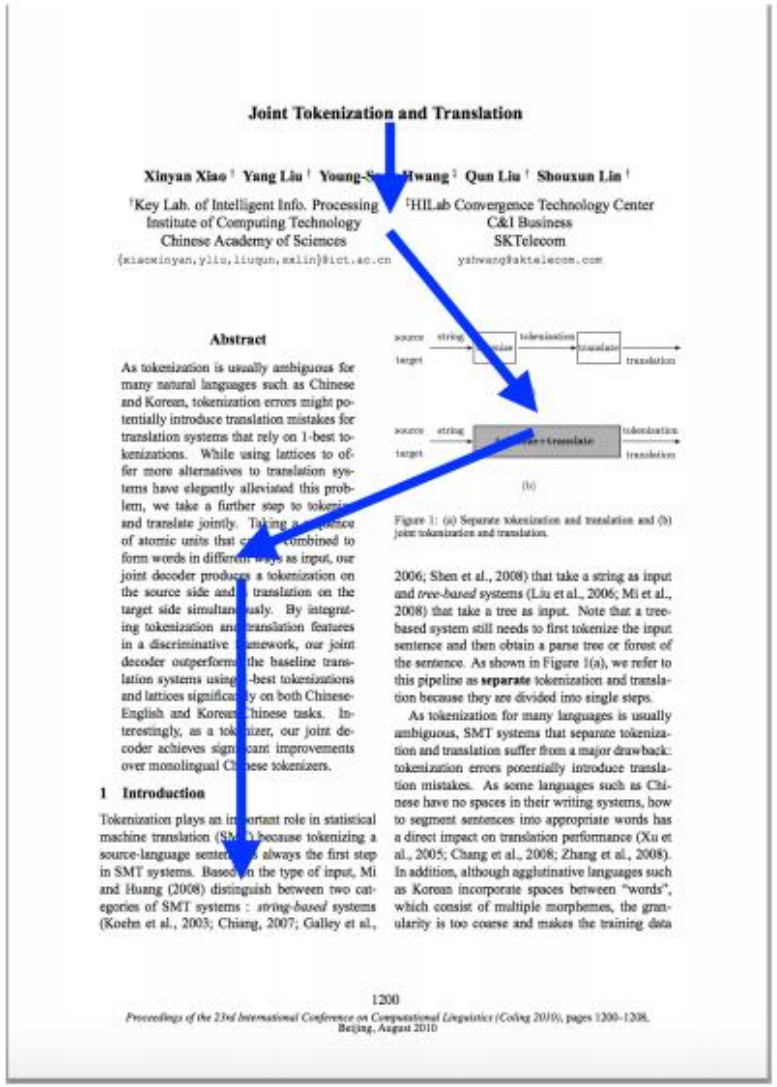
In phrase-based models, phrases are usually strings of adjacent words instead of syntactic constituents, excelling at capturing local reordering and performing translations that are localized to substrings that are common enough to be observed on training data. However, a key limitation of phrase-based models is that they fail to model reordering at the phrase level robustly. Typically, phrase reordering is modeled in terms of offset positions at the word level (Koehn, 2004; Och and Ney, 2004), making little or no direct use of syntactic information.

Recent research on statistical machine translation has lead to the development of syntax-based models. Wu (1997) proposes Inversion Transduction Grammars, treating translation as a process of parallel parsing of the source and target language via a synchronized grammar. Alshawi et al. (2000) represent each production in parallel dependency tree as a finite transducer. Melamed (2004) formalizes machine translation problem as synchronous parsing based on multi-text grammars. Graehl and Knight (2004) describe training and decoding algorithms for both generalized tree-to-tree and tree-to-string transducers. Chiang (2005) presents a hierarchical phrase-based model that uses hierarchical phrase pairs, which are formally productions of a synchronous context-free grammar. Ding and Palmer (2005) propose a syntax-based translation model based on a probabilistic synchronous dependency insert grammar, a version of synchronous grammars defined on dependency trees. All these approaches, though different in formalism, make use of synchronous grammars or tree-based transduction rules to model both source and target languages.

Another class of approaches make use of syntactic information in the target language alone, treating the translation problem as a parsing problem. Yamada and Knight (2001) use a parser in the target language to train probabilities on a set of

[1] The mathematical notation we use in this paper is taken from that paper: a source string $f_1^J = f_1, \ldots, f_j, \ldots, f_J$ is to be translated into a target string $e_1^I = e_1, \ldots, e_i, \ldots, e_I$. Here, I is the length of the target string, and J is the length of the source string.

Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the ACL, pages 609–616, Sydney, July 2006.



Joint Tokenization and Translation

Xinyan Xiao [†] Yang Liu [†] Young-Sook Hwang [‡] Qun Liu [†] Shouxun Lin [†]
[†]Key Lab. of Intelligent Info. Processing
Institute of Computing Technology
Chinese Academy of Sciences
{xiaoxinyan,yliu,liuqun,sxlin}@ict.ac.cn
[‡]HILab Convergence Technology Center
C&I Business
SKTelecom
yshwang@sktelecom.com

Abstract

As tokenization is usually ambiguous for many natural languages such as Chinese and Korean, tokenization errors might potentially introduce translation mistakes for translation systems that rely on 1-best tokenizations. While using lattices to offer more alternatives to translation systems have elegantly alleviated this problem, we take a further step to tokenize and translate jointly. Taking a sequence of atomic units that can be combined to form words in different ways as input, our joint decoder produces a tokenization on the source side and a translation on the target side simultaneously. By integrating tokenization and translation features in a discriminative framework, our joint decoder outperforms the baseline translation systems using 1-best tokenizations and lattices significantly on both Chinese-English and Korean-Chinese tasks. Interestingly, as a tokenizer, our joint decoder achieves significant improvements over monolingual Chinese tokenizers.

(b)

Figure 1: (a) Separate tokenization and translation and (b) joint tokenization and translation.

1 Introduction

Tokenization plays an important role in statistical machine translation (SMT) because tokenizing a source-language sentence is always the first step in SMT systems. Based on the type of input, Mi and Huang (2008) distinguish between two categories of SMT systems : *string-based* systems (Koehn et al., 2003; Chiang, 2007; Galley et al., 2006; Shen et al., 2008) that take a string as input and *tree-based* systems (Liu et al., 2006; Mi et al., 2008) that take a tree as input. Note that a tree-based system still needs to first tokenize the input sentence and then obtain a parse tree or forest of the sentence. As shown in Figure 1(a), we refer to this pipeline as **separate** tokenization and translation because they are divided into single steps.

As tokenization for many languages is usually ambiguous, SMT systems that separate tokenization and translation suffer from a major drawback: tokenization errors potentially introduce translation mistakes. As some languages such as Chinese have no spaces in their writing systems, how to segment sentences into appropriate words has a direct impact on translation performance (Xu et al., 2005; Chang et al., 2008; Zhang et al., 2008). In addition, although agglutinative languages such as Korean incorporate spaces between "words", which consist of multiple morphemes, the granularity is too coarse and makes the training data

Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010), pages 1200–1208, Beijing, August 2010

图和表的重要性

- 图和表是论文的骨架，争取让读者按照顺序看就能理解论文的主要思想，不用通过看正文才能懂
  - 一般第一遍看，都会看图、找例子
  - 然后翻到后面找主要结果
  - 再从头看正文
- 把论文的元素放在最应该被放在的地方，符合读者的认知惯性，降低理解难度

直接列出自己的贡献

coding phase. [1] Based on max-translation decoding and max-derivation decoding used in conventional *individual* decoders (Section 2), we go further to develop a *joint* decoder that integrates multiple models on a firm basis:

- Structuring the search space of each model as a *translation hypergraph* (Section 3.1), our joint decoder packs individual translation hypergraphs together by merging nodes that have identical partial translations (Section 3.2). Although such *translation-level combination* will not produce new translations, it does change the way of selecting promising candidates.
- Two models could even share derivations with each other if they produce the same structures on the target side (Section 3.3), which we refer to as *derivation-level combination*. This method enlarges the search space by allowing for mixing different types of translation rules within one derivation.
- As multiple derivations are used for finding optimal translations, we extend the minimum error rate training (MERT) algorithm (Och, 2003) to tune feature weights with respect to BLEU score for max-translation decoding (Section 4).

全局连贯性

如何描述自己的方法

- 不要一上来就描述你的工作，可以先介绍背景知识（往往就是baseline）
  - 有利于降低初学者或其他领域学者的理解难度
  - 有利于对introduction中的论文做更详细的解释
  - 有利于对比baseline和你的方法

例子

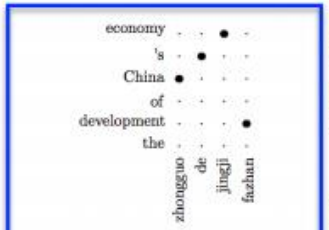


Figure 1: An example of word alignment between a pair of Chinese and English sentences.

ear space to encode the probabilities of exponentially many alignments. We develop a new algorithm for extracting phrase pairs from weighted matrices and show how to estimate their relative frequencies and lexical weights. Experimental results show that using weighted matrices achieves consistent improvements in translation quality and significant reduction in extraction time over using n-best lists.

2 Background

Figure 1 shows an example of word alignment between a pair of Chinese and English sentences. The Chinese and English words are listed horizontally and vertically, respectively. The dark points indicate the correspondence between the words in two languages. For example, the first Chinese word "*zhongguo*" is aligned to the fourth English word "*China*".

Formally, given a source sentence $\mathbf{f} = f_1^J = f_1, \ldots, f_j, \ldots, f_J$ and a target sentence $\mathbf{e} = e_1^I = e_1, \ldots, e_i, \ldots, e_I$, we define a link $l = (j, i)$ to exist if f_j and e_i are translation (or part of translation) of one another. Then, an alignment $\mathbf{a}$ is a subset of the Cartesian product of word positions:

$$\mathbf{a} \subseteq \{(j, i) : j = 1, \ldots, J; i = 1, \ldots, I\} \quad (1)$$

Usually, SMT systems only use the 1-best alignments for extracting translation rules. For example, given a source phrase $\bar{f}$ and a target phrase $\bar{e}$, the phrase pair $(\bar{f}, \bar{e})$ is said to be *consistent* (Och and Ney, 2004) with the alignment if and only if: (1) there must be at least one word inside one phrase aligned to a word inside the other phrase and (2) no words inside one phrase can be aligned to a word outside the other phrase.

After all phrase pairs are extracted from the training corpus, their translation probabilities can be estimated as *relative frequencies* (Och and Ney, 2004):

$$\phi(\bar{e}|\bar{f}) = \frac{count(\bar{f}, \bar{e})}{\sum_{\bar{e}'} count(\bar{f}, \bar{e}')} \quad (2)$$

where $count(\bar{f}, \bar{e})$ indicates how often the phrase pair $(\bar{f}, \bar{e})$ occurs in the training corpus.

Besides relative frequencies, *lexical weights* (Koehn et al., 2003) are widely used to estimate how well the words in $\bar{f}$ translate the words in $\bar{e}$. To do this, one needs first to estimate a lexical translation probability distribution $w(e|f)$ by relative frequency from the same word alignments in the training corpus:

$$w(e|f) = \frac{count(f, e)}{\sum_{e'} count(f, e')} \quad (3)$$

Note that a special source NULL token is added to each source sentence and aligned to each unaligned target word.

As the alignment $\bar{a}$ between a phrase pair $(\bar{f}, \bar{e})$ is retained during extraction, the lexical weight can be calculated as

$$p_w(\bar{e}|\bar{f}, \bar{a}) = \prod_{i=1}^{|\bar{e}|} \frac{1}{|\{j|(j, i) \in \bar{a}\}|} \sum w(e_i|f_j) \quad (4)$$

If there are multiple alignments $\bar{a}$ for a phrase pair $(\bar{f}, \bar{e})$, Koehn et al. (2003) choose the one with the highest lexical weight:

$$p_w(\bar{e}|\bar{f}) = \max_{\bar{a}} \left\{ p_w(\bar{e}|\bar{f}, \bar{a}) \right\} \quad (5)$$

Simple and effective, relative frequencies and lexical weights have become the standard features in modern discriminative SMT systems.

3 Weighted Alignment Matrix

We believe that offering more candidate alignments to extracting translation rules might help improve translation quality. Instead of using n-best lists (Venugopal et al., 2008), we propose a new structure called *weighted alignment matrix*.

We use an example to illustrate our idea. Figure 2(a) and Figure 2(b) show two alignments of a Chinese-English sentence pair. We observe that some links (e.g., (1,4) corresponding to the word

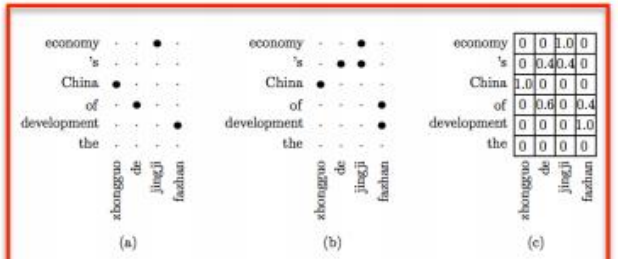


Figure 2: (a) One alignment of a sentence pair; (b) another alignment of the same sentence pair; (c) the resulting weighted alignment matrix that takes the two alignments as samples, of which the initial probabilities are 0.6 and 0.4, respectively.

pair ("*zhongguo*", "*China*")) occur in both alignments, some links (e.g., (2,3) corresponding to the word pair ("*de*","*of*")) occur only in one alignment, and some links (e.g., (1,1) corresponding to the word pair ("*zhongguo*", "*the*")) do not occur. Intuitively, we can estimate how well two words are aligned by calculating its relative frequency, which is the probability sum of alignments in which the link occurs divided by the probability sum of all possible alignments. Suppose that the probabilities of the two alignments in Figures 2(a) and 2(b) are 0.6 and 0.4, respectively. We can estimate the relative frequencies for every word pair and obtain a weighted matrix shown in Figure 2(c). Therefore, each word pair is associated with a probability to indicate how well they are aligned. For example, in Figure 2(c), we say that the word pair ("*zhongguo*", "*China*") is definitely aligned, ("*zhongguo*", "the") is definitely unaligned, and ("*de*", "*of*") has a 60% chance to get aligned.

Formally, a weighted alignment matrix m is a $J \times I$ matrix, in which each element stores a *link probability* $p_m(j, i)$ to indicate how well f_j and e_i are aligned. Currently, we estimate link probabilities from an n-best list by calculating relative frequencies:

$$p_m(j, i) = \frac{\sum_{a \in \mathcal{N}} p(a) \times \delta(a, j, i)}{\sum_{a \in \mathcal{N}} p(a)} \quad (6)$$

$$= \sum_{a \in \mathcal{N}} p(a) \times \delta(a, j, i) \quad (7)$$

where

$$\delta(a, j, i) = \begin{cases} 1 & (j, i) \in a \\ 0 & \text{otherwise} \end{cases} \quad (8)$$

Note that $\mathcal{N}$ is an n-best list, $p(a)$ is the probability of an alignment a in the n-best list, $\delta(a, j, i)$ indicates whether a link (j, i) occurs in the alignment a or not. We assign 0 to any unseen alignment. As $p(a)$ is usually normalized (i.e., $\sum_{a \in \mathcal{N}} p(a) = 1$), we remove the denominator in Eq. (6).

Accordingly, the probability that the two words f_j and e_i are not aligned is

$$\bar{p}_m(j, i) = 1.0 - p_m(j, i) \quad (9)$$

For example, as shown in Figure 2(c), the probability for the two words "*de*" and "*of*" being aligned is 0.6 and the probability that they are not aligned is 0.4.

Intuitively, the probability of an alignment a is the product of link probabilities. If a link (j, i) occurs in a, we use $p_m(j, i)$; otherwise we use $\bar{p}_m(j, i)$. Formally, given a weighted alignment matrix m, the probability of an alignment a can be calculated as

$$p_m(a) = \prod_{j=1}^{J} \prod_{i=1}^{I} (p_m(j, i) \times \delta(a, j, i) + \bar{p}_m(j, i) \times (1 - \delta(a, j, i))) \quad (10)$$

It proves that the sum of all alignment probabilities is always 1: $\sum_{a \in \mathcal{A}} p_m(a) = 1$, where $\mathcal{A}$

Yang Liu, Tian Xia, Xinyan Xiao, and Qun Liu. **Weighted Alignment Matrices for Statistical Machine Translation**. In EMNLP *2009*.

Running Example是利器

- 英语不好说不清楚？用例子！
- 全篇统一使用一个running example，用来阐释你的方法（甚至是baseline）
- 围绕着running example，展开描述你的工作
- 审稿人能从running example中更舒服地了解你的工作，读正文会花掉他/她更多时间
- 看完running example，审稿人便能知道核心思想

方法描述的逻辑顺序

- 错误的顺序
  - 形式化描述
  - 解释数学符号的意义

- 正确的顺序
  - 首先给出running example
  - 然后利用running example，用通俗语言描述你的想法
  - 最后才是形式化描述

每个公式都有语言学意义，都来自你的直觉和想法，
直接告诉审稿人，不要让他/她去揣摩

例子

We believe that offering more candidate alignments to extracting translation rules might help improve translation quality. Instead of using n-best lists (Venugopal et al., 2008), we propose a new structure called *weighted alignment matrix*.

We use an example to illustrate our idea. Figure 2(a) and Figure 2(b) show two alignments of a Chinese-English sentence pair. We observe that some links (e.g., (1,4) corresponding to the word pair ("*zhongguo*", "*China*")) occur in both alignments, some links (e.g., (2,3) corresponding to the word pair ("*de*","*of*")) occur only in one alignment, and some links (e.g., (1,1) corresponding to the word pair ("*zhongguo*", "*the*")) do not occur. Intuitively, we can estimate how well two words are aligned by calculating its relative frequency, which is the probability sum of alignments in which the link occurs divided by the probability sum of all possible alignments. Suppose that the probabilities of the two alignments in Figures 2(a) and 2(b) are 0.6 and 0.4, respectively. We can estimate the relative frequencies for every word pair and obtain a weighted matrix shown in Figure 2(c). Therefore, each word pair is associated with a probability to indicate how well they are aligned. For example, in Figure 2(c), we say that the word pair ("*zhongguo*", "*China*") is definitely aligned, ("*zhongguo*", "the") is definitely unaligned, and ("*de*", "*of*") has a 60% chance to get aligned.

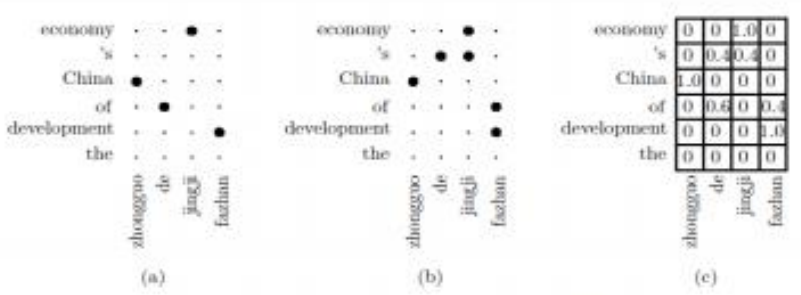


Figure 2: (a) One alignment of a sentence pair; (b) another alignment of the same sentence pair; (c) the resulting weighted alignment matrix that takes the two alignments as samples, of which the initial probabilities are 0.6 and 0.4, respectively.

Formally, a weighted alignment matrix m is a $J \times I$ matrix, in which each element stores a *link probability* $p_m(j, i)$ to indicate how well f_j and e_i are aligned. Currently, we estimate link probabilities from an n-best list by calculating relative frequencies:

Yang Liu, Tian Xia, Xinyan Xiao, and Qun Liu. **Weighted Alignment Matrices for Statistical Machine Translation**. In EMNLP *2009*.

实验设计

- 公认的标准数据和state-of-the-art系统
- 实验先辅后主
  - 辅助实验（开发集）： 参数的影响
  - 主实验（测试集）： 证明显著超过baseline
- 必须有显著性检验
- 不辞辛劳，做到极致

minimum ⇨ solid ⇨ maximum

先辅后主

We first used the validation sets to find the optimal setting of our approach: noisy generation, the value of n, feature group, and training corpus size.

Table 2 shows the results of different noise generation strategies: randomly shuffling, inserting, replacing, and deleting words. We find shuffling source and target words randomly consistently yields the best results. One possible reason is that the translation probability product feature (Liu, Liu, and Lin, 2010) derived from GIZA++ suffices to evaluate lexical choices accurately. It is more important to guide the aligner to model the structural divergence by changing word orders randomly.

Table 3 gives the results of different values of sample size n on the validation sets. We find that increasing n does not lead to significant improvements. This might result from the high concentration property of log-linear models. Therefore, we simply set $n = 1$ in the following experiments.

Table 4 shows the effect of adding non-local features. As most structural divergence between natural languages are non-local, including non-local features leads to significant improvements for both French-English and Chinese-English. As a result, we used all 16 features in the following experiments.

Table 5 gives our final result on the test sets. Our approach outperforms all unsupervised aligners significantly statistically ($p < 0.01$) except for the Berkeley aligner on the French-English data. The margins on Chinese-English are generally much larger than French-English because Chinese and English are distantly related and exhibit more non-local structural divergence. Vigne used the same features as our system but was trained in a supervised way. Its results can be treated as the upper bounds that our method can potentially approach.

Yang Liu and Massing Sun. **Contrastive Unsupervised Word Alignment with Non-Local Features**. *arXiv:1410.2082[cs.CL]*.

Caption包含充分的信息

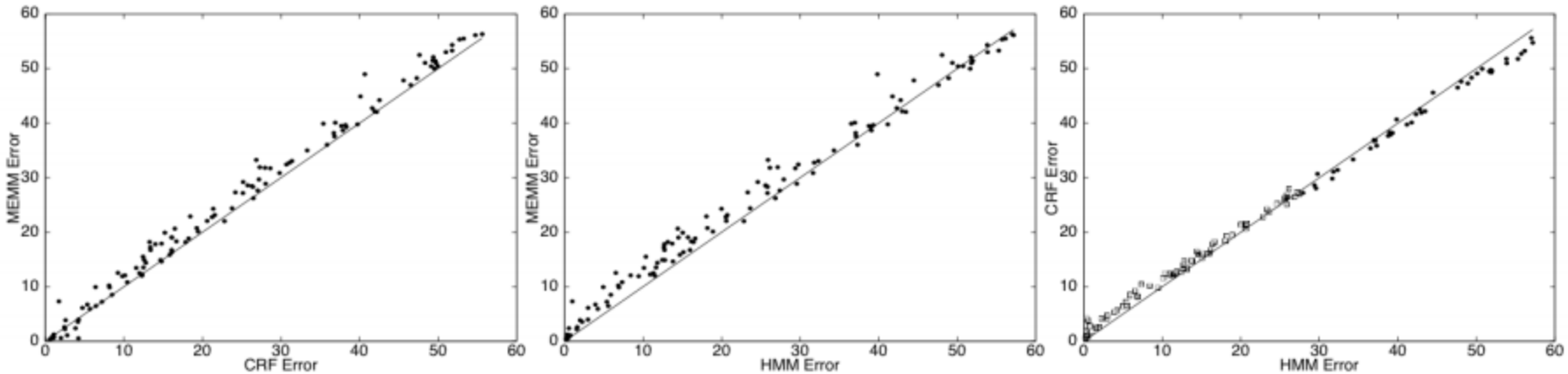


Figure 3. Plots of 2×2 error rates for HMMs, CRFs, and MEMMs on randomly generated synthetic data sets, as described in Section 5.2. As the data becomes "more second order," the error rates of the test models increase. As shown in the left plot, the CRF typically significantly outperforms the MEMM. The center plot shows that the HMM outperforms the MEMM. In the right plot, each open square represents a data set with $\alpha < \frac{1}{2}$, and a solid circle indicates a data set with $\alpha \geq \frac{1}{2}$. The plot shows that when the data is mostly second order ($\alpha \geq \frac{1}{2}$), the discriminatively trained CRF typically outperforms the HMM. These experiments are not designed to demonstrate the advantages of the additional representational power of CRFs and MEMMs relative to HMMs.

最好能直接看懂图，不用再去看正文

John Lafferty, Andrew McCallum, and Fernando Pereira. **Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data**. In *ICML 2003*.

如何写相关工作

错误

没有引用重要论文（可以直接作为rejection的理由）

简单的罗列和堆砌，缺乏深刻到位的评论

通过批评乃至攻击前人工作证明你的工作的创新性

正确

向审稿人显示你对本领域具有全面深刻的把握

通过与前人工作的对比凸显你的工作的创新性

为读者梳理领域的发展脉络，获得全局的认识

例子

2 Related Work

The CVG is inspired by two lines of research: Enriching PCFG parsers through more diverse sets of discrete states and recursive deep learning models that jointly learn classifiers and continuous feature representations for variable-sized inputs.

Improving Discrete Syntactic Representations
As mentioned in the introduction, there are several approaches to improving discrete representations for parsing. Klein and Manning (2003a) use manual feature engineering, while Petrov et al. (2006) use a learning algorithm that splits and merges the syntactic categories in order to maximize likelihood on the treebank. Their approach splits categories into several dozen subcategories. Another approach is lexicalized parsers (Collins, 2003; Charniak, 2000) that describe each category with a lexical item, usually the head word. More recently, Hall and Klein

Deep Learning and Recursive Deep Learning
Early attempts at using neural networks to describe phrases include Elman (1991), who used recurrent neural networks to create representations of sentences from a simple toy grammar and to analyze the linguistic expressiveness of the resulting representations. Words were represented as one-on vectors, which was feasible since the grammar only included a handful of words. Collobert and Weston (2008) showed that neural networks can perform well on sequence labeling lan-

Richard Socher, John Bauer, Christopher Manning, and Andrew Ng. **Parsing with Compositional Vector Grammars**. In *ACL 2013*.

传承与创新

in a factored parser. We extend the above ideas from discrete representations to richer continuous ones. The CVG can be seen as factoring discrete and continuous parsing in one model. Another different approach to the above generative models is to learn discriminative parsers using many well designed features (Taskar et al., 2004; Finkel et al., 2008). We also borrow ideas from this line of research in that our parser combines the generative PCFG model with discriminatively learned RNNs.

This paper uses several ideas of (Socher et al., 2011b). The main differences are (i) the dual representation of nodes as discrete categories and vectors, (ii) the combination with a PCFG, and (iii) the syntactic untying of weights based on child categories. We directly compare models with fully tied and untied weights. Another work that represents phrases with a dual discrete-continuous representation is (Kartsaklis et al., 2012).

Richard Socher, John Bauer, Christopher Manning, and Andrew Ng. **Parsing with Compositional Vector Grammars**. In *ACL 2013*.

附录

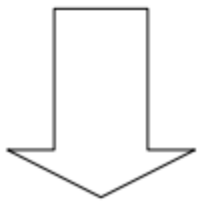
- 并非必需，但是对于读者深入理解你的工作有帮助，往往非常形式化
  - 证明
  - “鸡肋”
- 恰当地使用附录能显著提升论文的可读性

写作常见问题

- 句子过长
- 经常使用被动句式
- 结构松散、口语化
- 不定冠词和定冠词的使用
- 公式后面文字的缩进
- 引用的写法

被动句式+弱动词

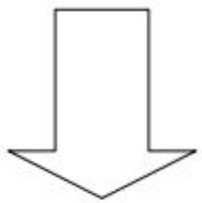
The whole process of finding fuzzy-matched word pairs and computing their similarity is demonstrated in detail. More importantly, the performance of BLEU is significantly improved by integrating fuzzy matching.



We demonstrate how to find fuzzy-matched word pairs and compute their similarities in detail. More importantly, integrating fuzzy matching significantly improved the translation performance in terms of BLEU.

结构松散+口语化+缺乏力度

In this step, we want to induce an alignment between words and predicates. The alignment can give a roughly mapping between words and the predicates that express their meanings, so it would be a useful constraint for rule extraction and reduce the searching space.



This step induces an alignment between words and predicates. Reflecting a rough mapping between natural languages and logic, such alignments impose linguistically motivated constraints on the search space and improve the efficiency of rule extraction.

公式的缩进

$$\hat{\lambda}_1^M = \underset{\lambda_1^M}{\operatorname{argmin}} \left\{ \sum_{s=1}^{S} E(\mathbf{r}_s, \hat{\mathbf{a}}(\mathbf{f}_s, \mathbf{e}_s; \lambda_1^M)) \right\} \tag{7}$$

$$= \underset{\lambda_1^M}{\operatorname{argmin}} \left\{ \sum_{s=1}^{S} \sum_{k=1}^{K} E(\mathbf{r}_s, \mathbf{a}_{s,k}) \delta(\hat{\mathbf{a}}(\mathbf{f}_s, \mathbf{e}_s; \lambda_1^M), \mathbf{a}_{s,k}) \right\} \tag{8}$$

where $\hat{\mathbf{a}}(\mathbf{f}_s, \mathbf{e}_s; \lambda_1^M)$ is the best candidate alignment produced by the linear model:

$$\hat{\mathbf{a}}(\mathbf{f}_s, \mathbf{e}_s; \lambda_1^M) = \underset{\mathbf{a}}{\operatorname{argmax}} \left\{ \sum_{m=1}^{M} \lambda_m h_m(\mathbf{f}_s, \mathbf{e}_s, \mathbf{a}) \right\} \tag{9}$$

The basic idea of MERT is to optimize only one parameter (i.e., feature weight) each time and keep all other parameters fixed. This process runs iteratively over M parameters until it cannot further reduce the loss on the training corpus.

当公式后的文本与公式有关，则不缩进，否则缩进

引用的写法

Jack (2010) argues that it is important to use syntax.

This algorithm proves to runs in approximately linear time (Jack, 2010).

前者表示人，后者去掉应该不影响整句话的意思。

其它

- 论文中每个数学符号都应当找得到定义，除非众所周知。永远不要不加说明就是用数学符号。
- 要避免数学符号冲突，使用符号列表
- 不要生造术语，尤其是中式译法，尽量符合惯例
- 集成所有信息元素，排版美观和专业

提高英语写作的窍门

- 找著名学者（尤其是native speaker）的论文钻研，学习句式和词汇用法，做笔记
- 写作时手边放一部纸质词典，经常翻看
- 拿不住的地方找Google：双引号查询

学习句式和用法

The need to segment and label sequences arises in many different problems in several scientific fields. Hidden

句式　*the need to … arises in … problems (fields)*

造句

The need to learn latent-variable models from unlabeled data arises in many NLP problems.

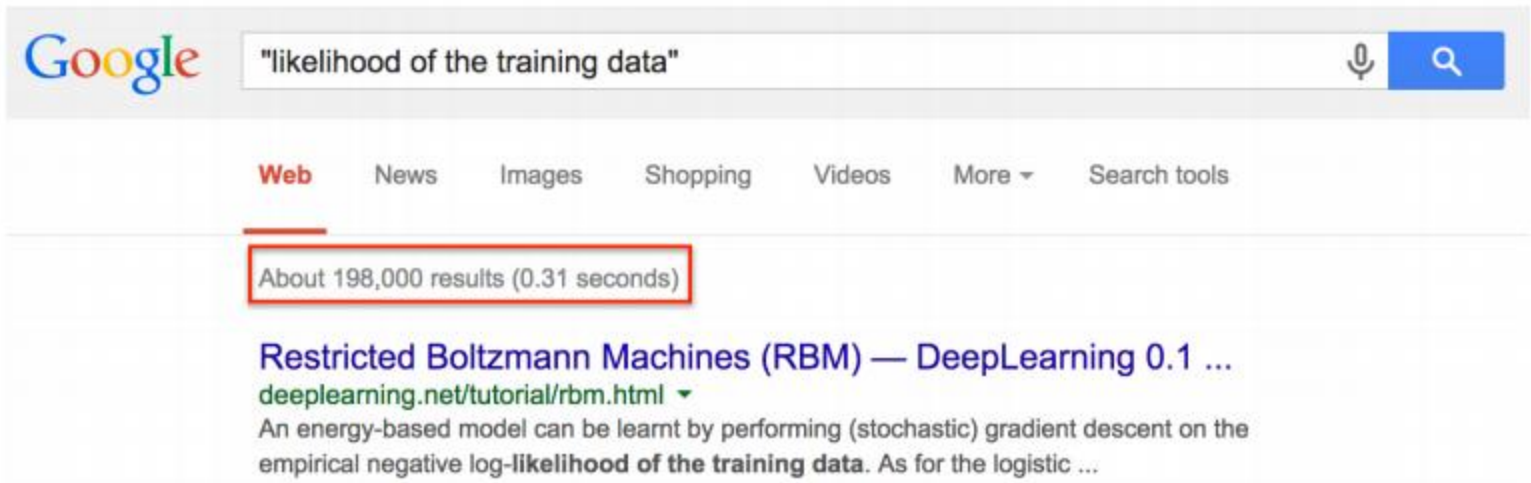
John Lafferty, Andrew McCallum, and Fernando Pereira. **Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data**. In *ICML 2003*.

利用搜索引擎

Maximizing the likelihood _____ the training data.

(A) in (B) on (c) of

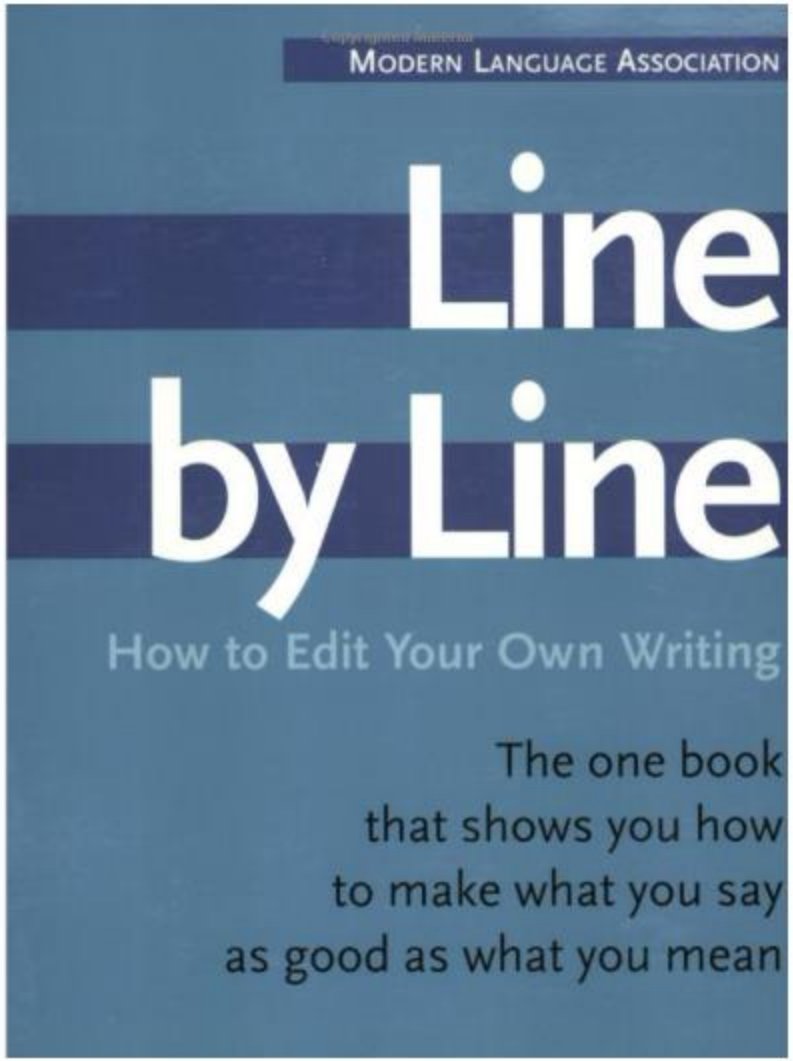
4 5,680 198,000



必须掌握的工具

- LaTex
  - 强烈建议用LaTex代替Word
  - http://www.ctex.org/HomePage
- Bibtex
  - 自动生成参考文献列表
- MetaPost
  - 编程画矢量图

英文写作进阶


MODERN LANGUAGE ASSOCIATION
Line by Line
How to Edit Your Own Writing
The one book
that shows you how
to make what you say
as good as what you mean

时间管理和获得反馈

- coarse-to-fine
  - 截稿前一个月开始写
  - 每隔两天改一次
- 听取不同背景读者的反馈意见
  - 专家：专业意见
  - 非专家：发现信息壁垒
- 写到极致，完成完美精致的艺术品

总结

- 写论文本质是分享思想，呈现信息
- 信息的呈现符合读者的认知惯性
- 全心全意为读者服务，降低阅读难度，提高愉悦感
- 细节决定成败
- 不要本末倒置：创新至上，技法为辅。