

如何做一个精彩的学术报告

车万翔

哈尔滨工业大学

社会计算与信息检索研究中心

2019-10-18



思考题

- 为什么做学术报告？
 - 为了展示自己的工作
 - 论文像产品，报告则是广告
 - 为了更好地交流
 - 帮助理清思路
- 选择题：做怎样的学术报告？
 - 让听众明白我的论文中的技术细节 ✗
 - 引起听众的兴趣，愿意深入阅读我的论文 ✓
 - 将我所知的一切相关内容都告诉听众 ✗
 - 向听众显示我非常聪明 ✗
 - 让听众觉得来听报告很值得 ✓

听众模型

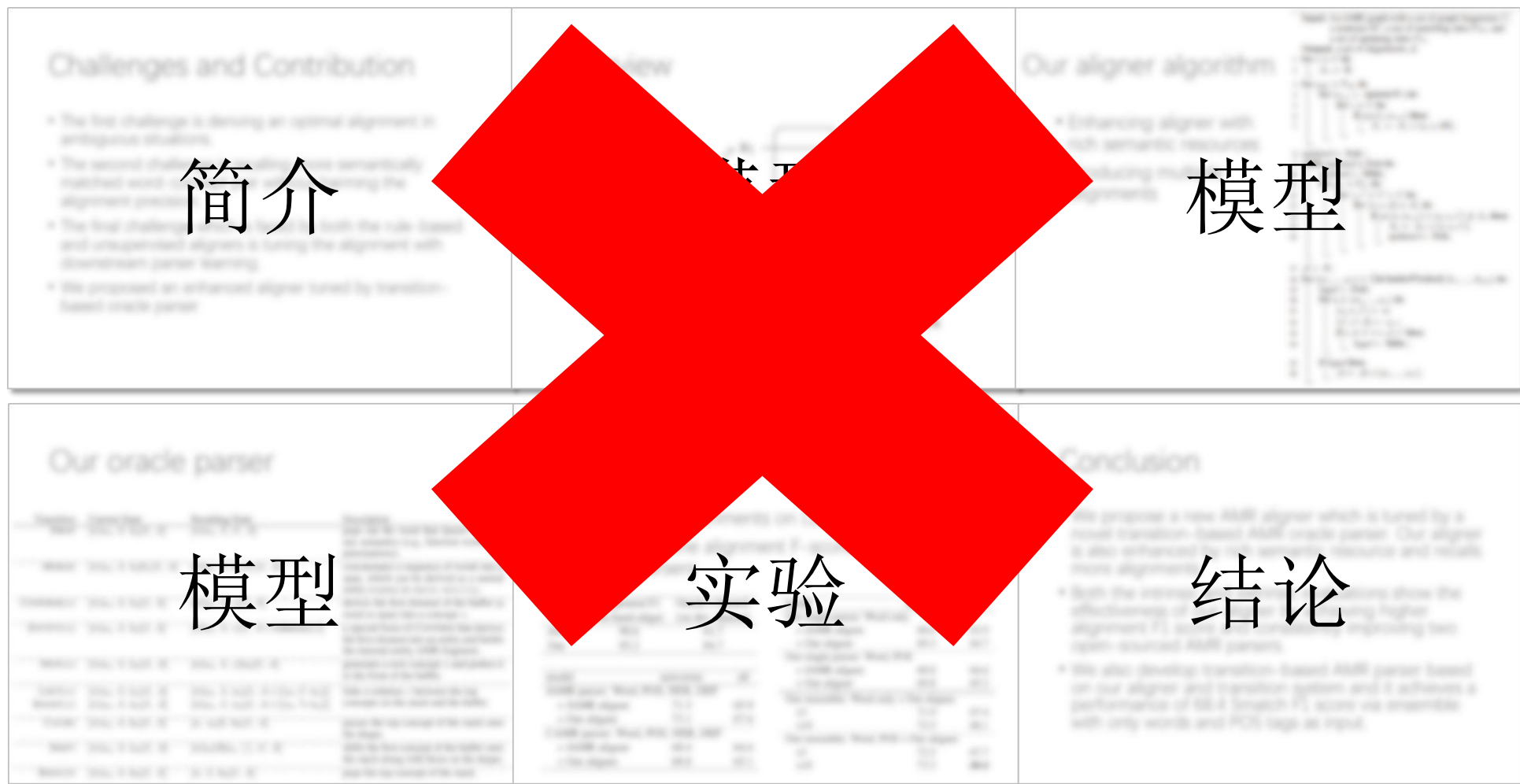
理想中的听众

- 领域专家
- 了解你的前期工作
- 已经读过你的论文
- 对于你的工作非常感兴趣

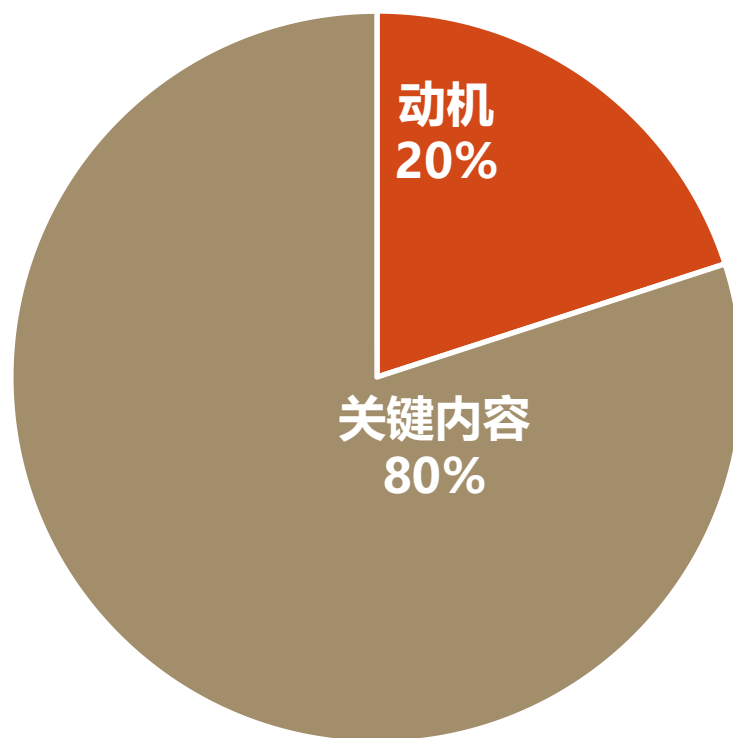
现实中的听众

- 来自其他领域
- 根本不了解到你的工作
- 这个时段没什么事情，恰巧发现这屋子有空座

幻灯片内容如何呈现？



幻灯片内容如何呈现？



动机

You have **two minutes** to engage your audience before they start to doze. -- Simon Peyton Jones in *How to give a great research talk*

- 问题是什么？它为什么有趣？
- 之前的方法有什么问题？
- 我们的方法如何解决了这个问题？
- 核心创新点用不超过**25**个字描述
 - 屠呦呦：提取出青蒿素，有效降低疟疾患者死亡率（**18**字）
 - 袁隆平：杂交水稻育种，把水稻亩产从**300**公斤提升到**900**公斤（**20**字）
 - 刘永坦：发明海地波超视距雷达，能在沿海**5000**公里内发现隐形战机（**25**字）

方法部分：多用例子

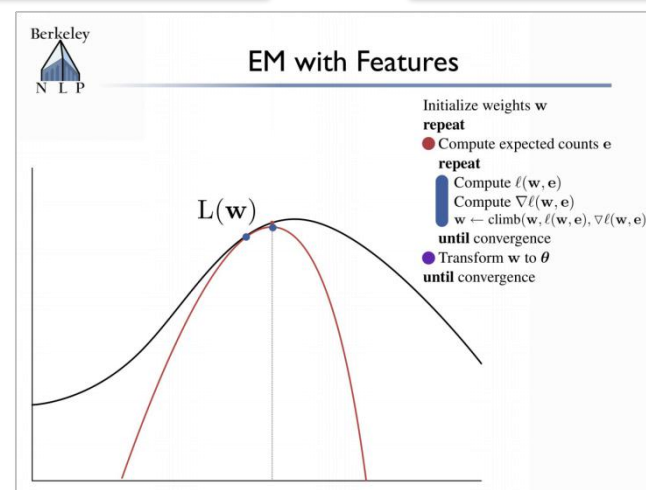
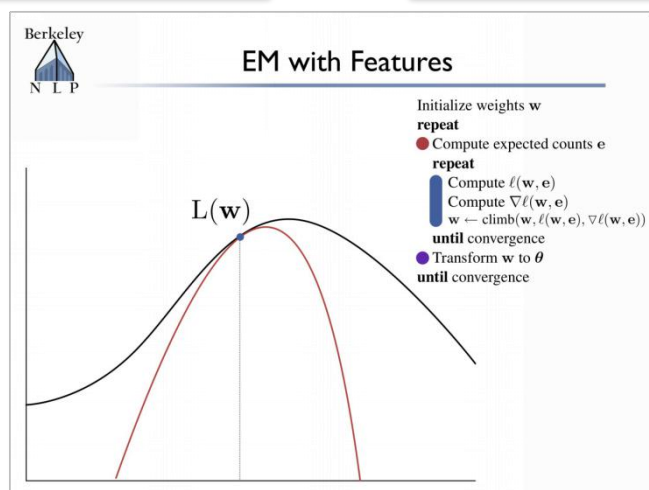
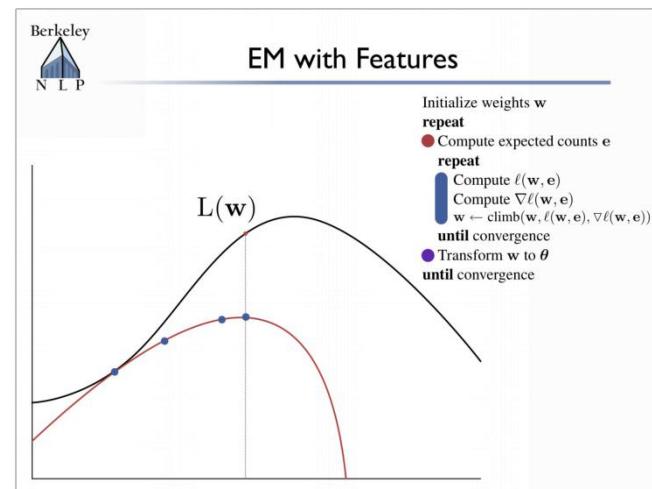
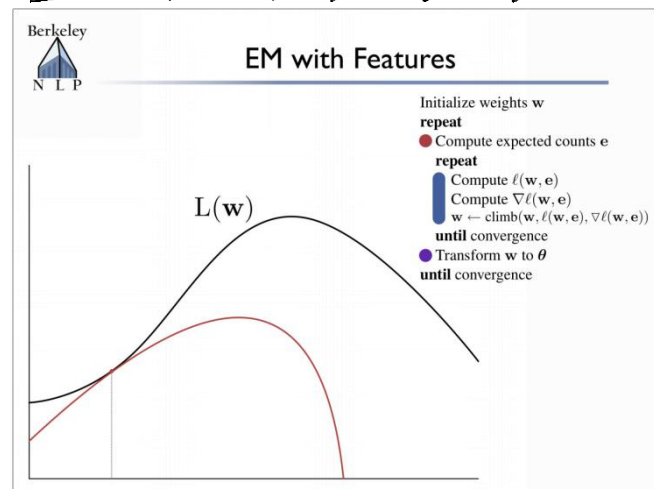
Berkeley
NLP

EM with Features

EM

Fit Params

Initialize weights w
repeat
● Compute expected counts e
repeat
■ Compute $\ell(w, e)$
■ Compute $\nabla \ell(w, e)$
 $w \leftarrow \text{climb}(w, \ell(w, e), \nabla \ell(w, e))$
until convergence
● Transform w to θ
until convergence



方法部分：反例

Transition	Current State	Resulting State	Description
DROP	$[\sigma s_0, \delta, b_0 \beta, A]$	$[\sigma s_0, \delta, \beta, A]$	pops out the word that doesn't convey any semantics (e.g., function words and punctuations).
MERGE	$[\sigma s_0, \delta, b_0 b_1 \beta, A]$	$[\sigma s_0, \delta, b_0 b_1 \beta, A]$	concatenates a sequence of words into a span, which can be derived as a named entity (name) or date-entity.
CONFIRM(c)	$[\sigma s_0, \delta, b_0 \beta, A]$	$[\sigma s_0, \delta, c \beta, A]$	derives the first element of the buffer (a word or span) into a concept c .
ENTITY(c)	$[\sigma s_0, \delta, b_0 \beta, A]$	$[\sigma s_0, \delta, c \beta, A \cup \text{relations}(c)]$	a special form of CONFIRM that derives the first element into an entity and builds the internal entity AMR fragment.
NEW(c)	$[\sigma s_0, \delta, b_0 \beta, A]$	$[\sigma s_0, \delta, c b_0 \beta, A]$	generates a new concept c and pushes it to the front of the buffer.
LEFT(r)	$[\sigma s_0, \delta, b_0 \beta, A]$	$[\sigma s_0, \delta, b_0 \beta, A \cup \{s_0 \xleftarrow{r} b_0\}]$	links a relation r between the top concepts on the stack and the buffer.
RIGHT(r)	$[\sigma s_0, \delta, b_0 \beta, A]$	$[\sigma s_0, \delta, b_0 \beta, A \cup \{s_0 \xrightarrow{r} b_0\}]$	
CACHE	$[\sigma s_0, \delta, b_0 \beta, A]$	$[\sigma, s_0 \delta, b_0 \beta, A]$	passes the top concept of the stack onto the deque.
SHIFT	$[\sigma s_0, \delta, b_0 \beta, A]$	$[\sigma s_0 \delta b_0, [], \beta, A]$	shifts the first concept of the buffer onto the stack along with those on the deque.
REDUCE	$[\sigma s_0, \delta, b_0 \beta, A]$	$[\sigma, \delta, b_0 \beta, A]$	pops the top concept of the stack.

方法部分：避免大量文字

13/25

四、泛在智慧环境下的系统软件智能化机理

- 传统基于规则的系统软件策略机制（例如调度机制、缓存机制、内存管理机制等）已经不能够匹配不断演进的硬件与应用
- 系统软件智能化：将感知、规划、决策、行动各模块有机合，突破规则策略机制的限制，学习获得新环境下的智能策略

–IBM: Milepost GCC是机器学习编译器，能智能的优化程序，因此能缩短开发时间，同时又能提升性能。对IBM System p服务器的初步试验显示，嵌入式软件性能平均提高了18%[7]

Google: 将机器学习运用在数据库系统中，对于真实数据的索引任务，神经网络建立的索引可以比传统的缓存优化 B 树索引方法提高 70% 的效率，同时存储空间还能节省一个数量级[8]

实验部分：图比表格好

LDC2014T12 Experiments

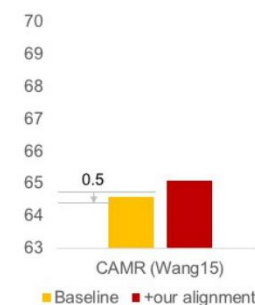
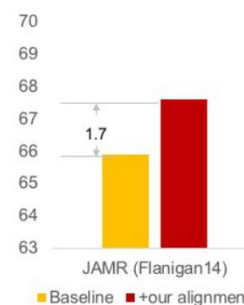
- alignment F-score

Aligner	Alignment F1 (on hand-align)	Oracle's Smatch (on dev. dataset)
JAMR	90.6	91.7
Our	95.2	94.7

- parser improvements

model	newswire	all
JAMR parser: Word, POS, NER, DEP		
+ JAMR aligner	71.3	65.9
+ Our aligner	73.1	67.6
CAMR parser: Word, POS, NER, DEP		
+ JAMR aligner	68.4	64.6
+ Our aligner	68.8	65.1

Aligner Experiments: Two Open-sourced AMR Parsers



信息元素的易理解程度

step	action	rule	stack	coverage
0				*****
1	S	r_3	[The President will]	*****
2	S	r_1	[The President will] [visit]	*****
3	R		[The President will visit]	*****
4	S	r_4	[The President will visit] [London in April]	*****
5	R		[The President will visit London in April]	*****

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

用例子与图来描述方法和实验

公式

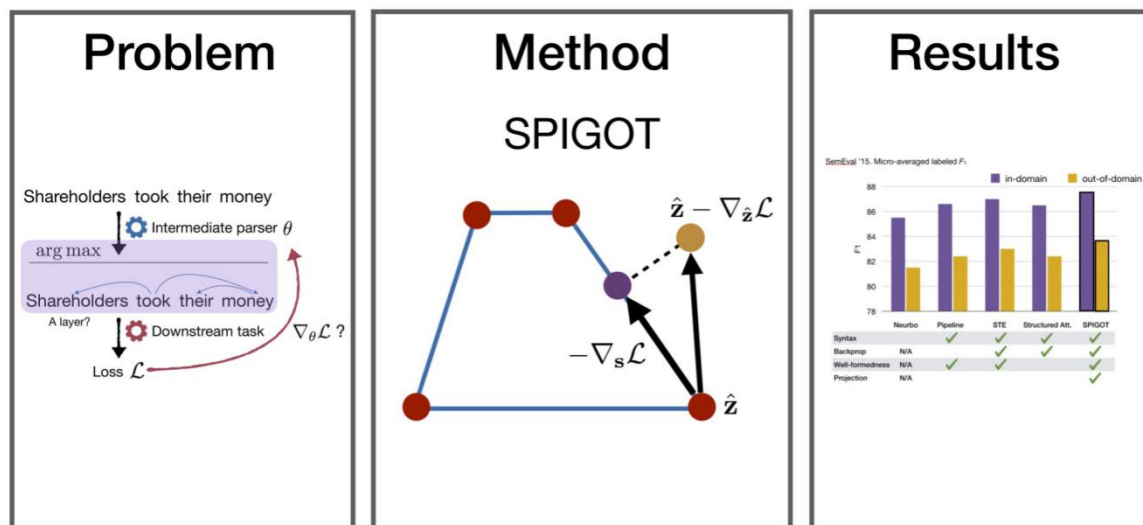
算法

证明



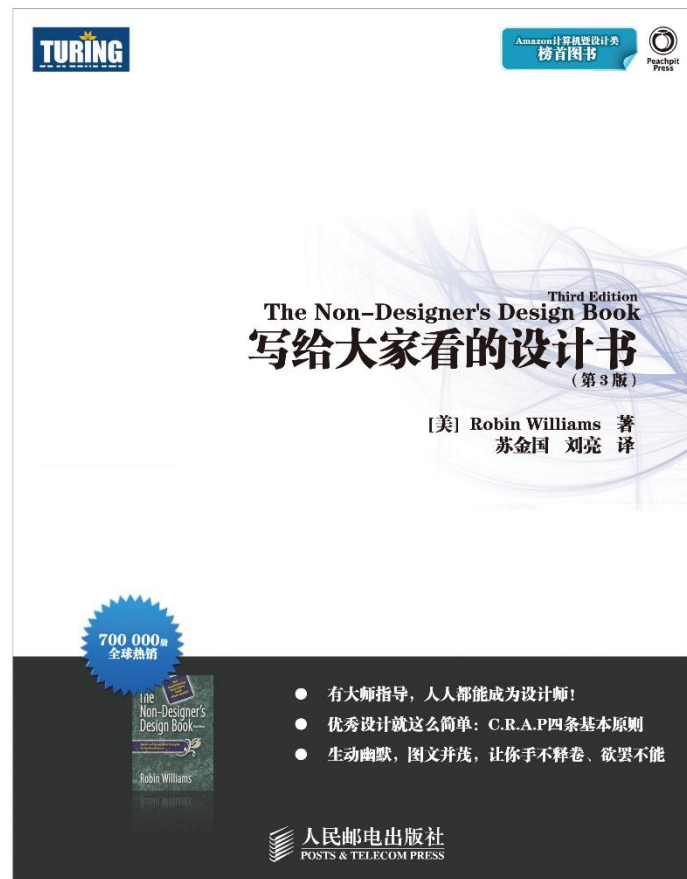
结论部分：新的展现形式

Conclusion



设计原则

- 亲密性：相关的元素应该组织到一起
- 重复：相同的内容达到形式的统一
- 对比：如果两项不完全相同，就应使之截然不同
- 对齐：使元素之间产生关联，有关联的都应对齐



根据设计原则做幻灯片



避免不对齐

Our aligner algorithm

- Enhancing aligner with rich semantic resources
- Producing multiple alignments

Input: An AMR graph with a set of graph fragments C ; a sentence W ; a set of matching rules P_M ; and a set of updating rules P_U .

Output: a set of alignments \mathcal{A} .

```
1 for  $c \in C$  do
2    $A_c \leftarrow \emptyset$ ;
3 for  $\rho_M \in P_M$  do
4   for  $w_{s,e} \leftarrow \text{spans}(W)$  do
5     for  $c \in C$  do
6       if  $\rho_M(c, w_{s,e})$  then
7          $A_c \leftarrow A_c \cup (s, e, \text{nil})$ ;
8 updated  $\leftarrow \text{true}$ ;
9 while updated is true do
10  updated  $\leftarrow \text{false}$ ;
11  for  $\rho_U \in P_U$  do
12    for  $c, c' \in C \times C$  do
13      for  $(s, e, d) \in A_c$  do
14        if  $\rho_U(c, w_{s,e}) \wedge (s, e, c') \notin A_{c'}$  then
15           $A_{c'} \leftarrow A_{c'} \cup (s, e, c')$ ;
16          updated  $\leftarrow \text{true}$ ;
17  $\mathcal{A} \leftarrow \emptyset$ ;
18 for  $(a_1, \dots, a_c) \in \text{CartesianProduct}(A_1, \dots, A_{|C|})$  do
19   legal  $\leftarrow \text{true}$ ;
20   for  $a \in (a_1, \dots, a_c)$  do
21      $(s, e, c') \leftarrow a$ ;
22      $(s', e', d) \leftarrow a_{c'}$ ;
23     if  $s \neq s' \wedge e \neq e'$  then
24       legal  $\leftarrow \text{false}$ ;
25   if legal then
26      $\mathcal{A} \leftarrow \mathcal{A} \cup (a_1, \dots, a_c)$ ;
```

“乱” 的原因： 视线跳动过多

Experiments

- We conduct experiments on LDC2014T12
- We evaluate the alignment F-score and Smatch of resulted parsers

Aligner	Alignment F1 (on hand-align)	Oracle's Smatch (on dev. dataset)
JAMR	90.6	91.7
Our	95.2	94.7

model	newswire	all
JAMR parser: Word, POS, NER, DEP		
+ JAMR aligner	71.3	65.9
+ Our aligner	73.1	67.6
CAMR parser: Word, POS, NER, DEP		
+ JAMR aligner	68.4	64.6
+ Our aligner	68.8	65.1

model	newswire	all
Our single parser: Word only		
+ JAMR aligner	68.6	63.9
+ Our aligner	69.3	64.7
Our single parser: Word, POS		
+ JAMR aligner	68.8	64.6
+ Our aligner	69.8	65.2
Our ensemble: Word only + Our aligner		
x3	71.9	67.4
x10	72.5	68.1
Our ensemble: Word, POS + Our aligner		
x3	72.5	67.7
x10	73.3	68.4

“乱” 的原因： 视线跳动过多

Experiments

- We conduct experiments on LDC2014T12
- We evaluate the alignment F-score and Smatch of resulted parsers

Aligner	Alignment F1 (on hand-align)	Oracle's Smatch (on dev. dataset)
JAMR	90.6	91.7
Our	95.2	94.7

model	newswire	all
JAMR parser: Word, POS, NER, DEP		
+ JAMR aligner	71.3	65.9
+ Our aligner	73.1	67.6
CAMR parser: Word, POS, NER, DEP		
+ JAMR aligner	68.4	64.6
+ Our aligner	68.8	65.1

model	newswire	all
Our single parser: Word only		
+ JAMR aligner	68.6	63.9
+ Our aligner	69.3	64.7
Our single parser: Word, POS		
+ JAMR aligner	68.8	64.6
+ Our aligner	69.8	65.2
Our ensemble: Word only + Our aligner		
x3	71.9	67.4
x10	72.5	68.1
Our ensemble: Word, POS + Our aligner		
x3	72.5	67.7
x10	73.3	68.4

“乱”的解法：重新组织内容

Experiments

- We conduct experiments on LDC2014T12
- We evaluate the alignment F-score and Smatch of resulted parsers

Aligner	Alignment F1 (on hand-align)	Oracle's Smatch (on dev. dataset)
JAMR	90.6	91.7
Our	95.2	94.7

model	newswire	all
JAMR parser: Word, POS, NER, DEP		
+ JAMR aligner	71.3	65.9
+ Our aligner	73.1	67.6
CAMR parser: Word, POS, NER, DEP		
+ JAMR aligner	68.4	64.6
+ Our aligner	68.8	65.1

model	newswire	all
Our single parser: Word only		
+ JAMR aligner	68.6	63.9
+ Our aligner	69.3	64.7
Our single parser: Word, POS		
+ JAMR aligner	68.8	64.6
+ Our aligner	69.8	65.2
Our ensemble: Word only + Our aligner		
x3	71.9	67.4
x10	72.5	68.1
Our ensemble: Word, POS + Our aligner		
x3	72.5	67.7
x10	73.3	68.4

LDC2014T12 Experiments

- alignment F-score
- parser improvements

Aligner	Alignment F1 (on hand-align)	Oracle's Smatch (on dev. dataset)
JAMR	90.6	91.7
Our	95.2	94.7

model	newswire	all
JAMR parser: Word, POS, NER, DEP		
+ JAMR aligner	71.3	65.9
+ Our aligner	73.1	67.6
CAMR parser: Word, POS, NER, DEP		
+ JAMR aligner	68.4	64.6
+ Our aligner	68.8	65.1

其它注意事项

- 不要在报告开始列大纲
 - 没有任何信息量
 - 无聊且浪费时间
- 不要罗列相关工作
 - 融入其它部分
- 不要罗列技术细节
 - 只会让听众昏昏欲睡
- 内容持续更新
 - 可以加入刚从其他报告中得到新的信息

演讲中最重要的武器

■ 热情！

- 如果你对自己的工作都不兴奋，听众更不会感兴趣
- 热情还能使你更放松
- 太紧张了怎么办？
 - 演讲之前深呼吸，或者去一下洗手间
 - 将开场白写下来（不建议写全部**notes**）
 - 来回走动或者使用一些肢体动作

其它注意事项

- 报告前
 - 提前测试好设备
 - 用U盘、网盘等备份slides
- 报告中
 - 面对听众，不能只盯着屏幕
 - 和听众有眼神的交流
 - 声音洪亮，有顿挫
 - 恰当运用动画
 - 控制好时间
- 报告后
 - 诚实回答观众的提问，不要避重就轻
 - 仔细倾听问题，不要打断对方
 - 分享幻灯片

总结

- 做学术报告的目的
- 幻灯片的内容和形式
 - 清晰阐述文章动机非常重要
 - 模型部分有取舍，用好图和例子
 - “结论”也有新思路
 - 四项设计的基本原则
- 演讲中的注意事项
 - 热情最重要
- 提高方法
 - 多学习其他人的报告
 - 多练习，任何topic



预祝大家的
研究更有影
响力！

参考资料

- Simon Peyton Jones. *How to give a great research talk*
 - <https://www.microsoft.com/en-us/research/academic-program/give-great-research-talk/>
- 刘洋. 机器翻译学术论文写作方法与技巧
 - http://nlp.csai.tsinghua.edu.cn/~ly/talks/cwmt14_tut.pdf