佛山市顺德区昊瑞电子科技有限公司

FOSHAN CITY SHUNDE HAORUI ELECTRON SCIENCE AND TECHNOLOGY CO,LTD

ITC SMT系统概述和单纯形算法

www.gdrohs.cn

昊瑞电子

- ITC SMT 系统概述
- Rescore中的单纯形算法

ITC系统框架

- 基于短语的log-linear模型
- Two pass search strategy(decoder)

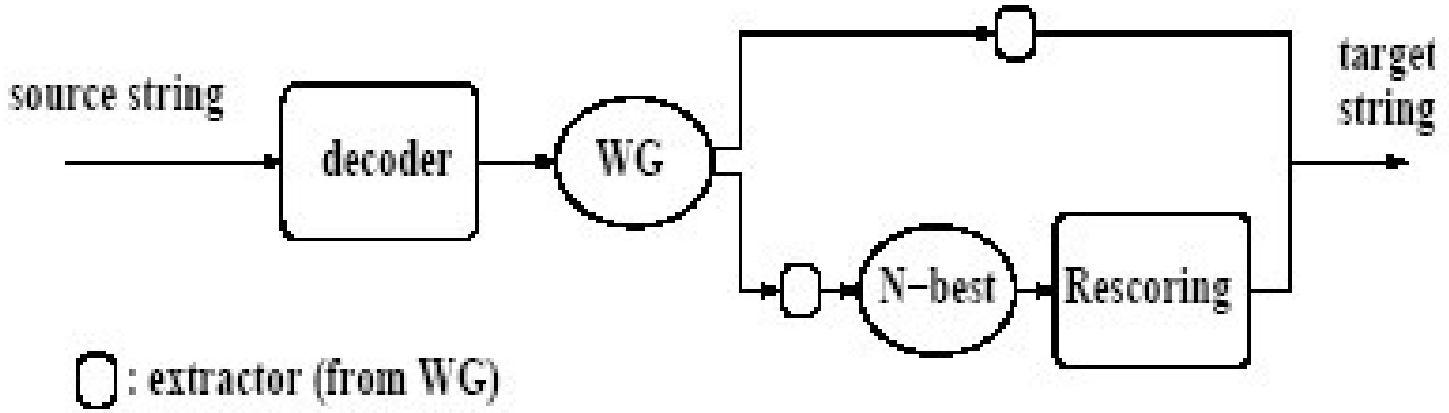


Figure 1: Decoding strategies: Given the word-graph (WG) produced by the decoder, either the 1-best translation is returned, or the N-best translations are extracted and re-scored with additional feature functions.

- First pass
  - Log-linear Model
  - Beam search decoder

    --threshold and histogram pruning
  - Non-monotone search constraints

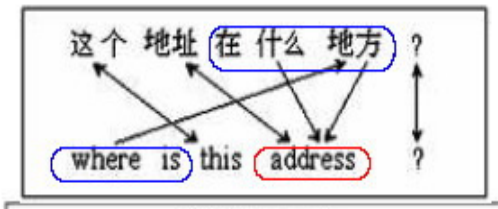
Tip:在最小错误率训练中，ITC系统优化的是100*BLEU+4*NIST

- Second pass
  - Extraction of 1000-best
  - Rescore algorithm（单纯形算法）

Phrase Extraction

- ITC系统抽取短语时除了利用GIZA++生成的词对齐外，还利用了CLA (Competitive Linking Algorithm)生成的词对齐，把利用CLA alignment抽取出来的短语简单的加到短语表里。

Phrase extraction from IBM and CLA alignments



IBM Alignment		CLA Alignment	
NULL_	is	这个	this
这个	this	地址	address
这个	is this	在	where
地址	address	什么	NULL_
在	NULL_	地方	is
什么	address	?	?
地方	address	这个 地址	this address
?	?	在 什么	where
这个 地址 在 什么 地方	where is this address	什么 地方	is
这个 地址 在 什么 地方 ?	where is this address ?	在 什么 地方	where is
		这个 地址 在 什么 地方	where is this address
		这个 地址 在 什么 地方 ?	where is this address ?

In this real example, the CLA alignment allows to extract the useful phrase “where is”.

优化技术

- Baseline:短语的最大长度设为8，单调搜索
- 优化方法
  - Translation lexicon
  - Additional word alignments
  - Re-segmented data
  - Non-monotone search

Table 5: Results of the optimization techniques on the IWSLT-04 development set (BLEU% score).

System	Chi2Eng	Jap2Eng	Ara2Eng
baseline	35.82	33.82	51.01
+translation lexicon	36.28	35.78	52.84
+additional alignments	37.59	38.77	54.14
+re-segmented data	38.29	38.97	–
+chunked data	–	39.59	–
+non-monotone search	42.51	44.66	56.40
+re-scoring	47.99	51.01	57.94

ITC文章中系统性能提升的讨论

- 一个是CLA的应用，对系统的性能有较大提升，相对于baseline有5%的提高。
- 另一个是non-monotone search的应用，对系统性能有较大提升11%relative.

Rescore

- Rescore的流程

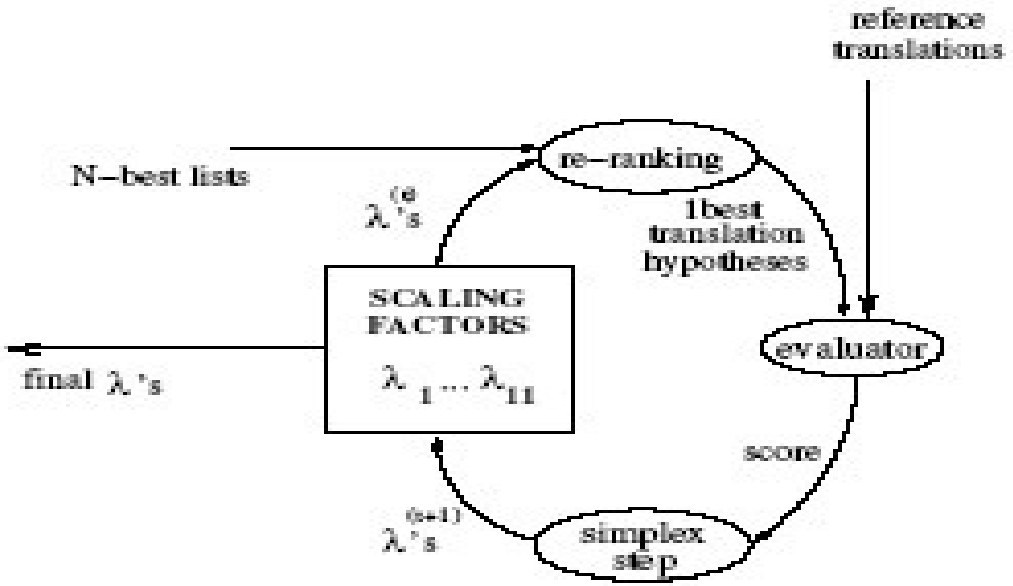


Figure 5: Estimation of weights for re-ranking

Rescore

- Rescoring features of ITC system

  1.IBM model1 lexicon score, over all possible alignments

$$\Pr(\mathbf{f}|\mathbf{e}) = \frac{\epsilon}{(l+1)^m} \prod_{j=1}^{m} \sum_{i=0}^{l} t(f_j|e_i).$$

  2.IBM model3 lexicon score

  3.CLA lexicon score

  4.question feature, a binary feature

  5.frequency of its n-grams (n=1,2,3,4) within the 1000-best translations

  6.ratio of the target length and source length

  7.2-grams target language model

  8.4-grams target language model

  9.5-grams target language model

Table 8: Contribution of each feature function in the re-scoring step on the IWSLT-04 development set (BLEU% score).

System	Chi2Eng	Jap2Eng	Ara2Eng
baseline	42.51	44.66	56.40
IBM model-1	42.31	44.48	56.00
IBM model-3	41.53	44.97	56.16
CLA score	42.42	45.20	56.31
question tag	42.81	45.83	56.66
n-grams	43.71	46.19	56.89
target length	41.11	41.00	50.87
2-grams LM	44.06	45.34	56.07
4-grams LM	45.88	45.51	56.72
5-grams LM	45.72	45.81	56.61
+all features	47.99	51.01	57.94

我在**rescore**中使用的特征

1.IBM model1 lexicon score
2.句首是否为标点（,.:;等），2值特征
3.frequency of its n-grams (n=1,2,3,4) within the 1000-best translations
4.ratio of the target length and source length
5.2-grams target language model
6.3-grams target language model
7.4-grams target language model

单纯形算法详解

- 全称应该叫求多维极小化问题的多维下降单纯形算法，不是线性规划中的单纯形算法。
- [C语言数值算法大全]

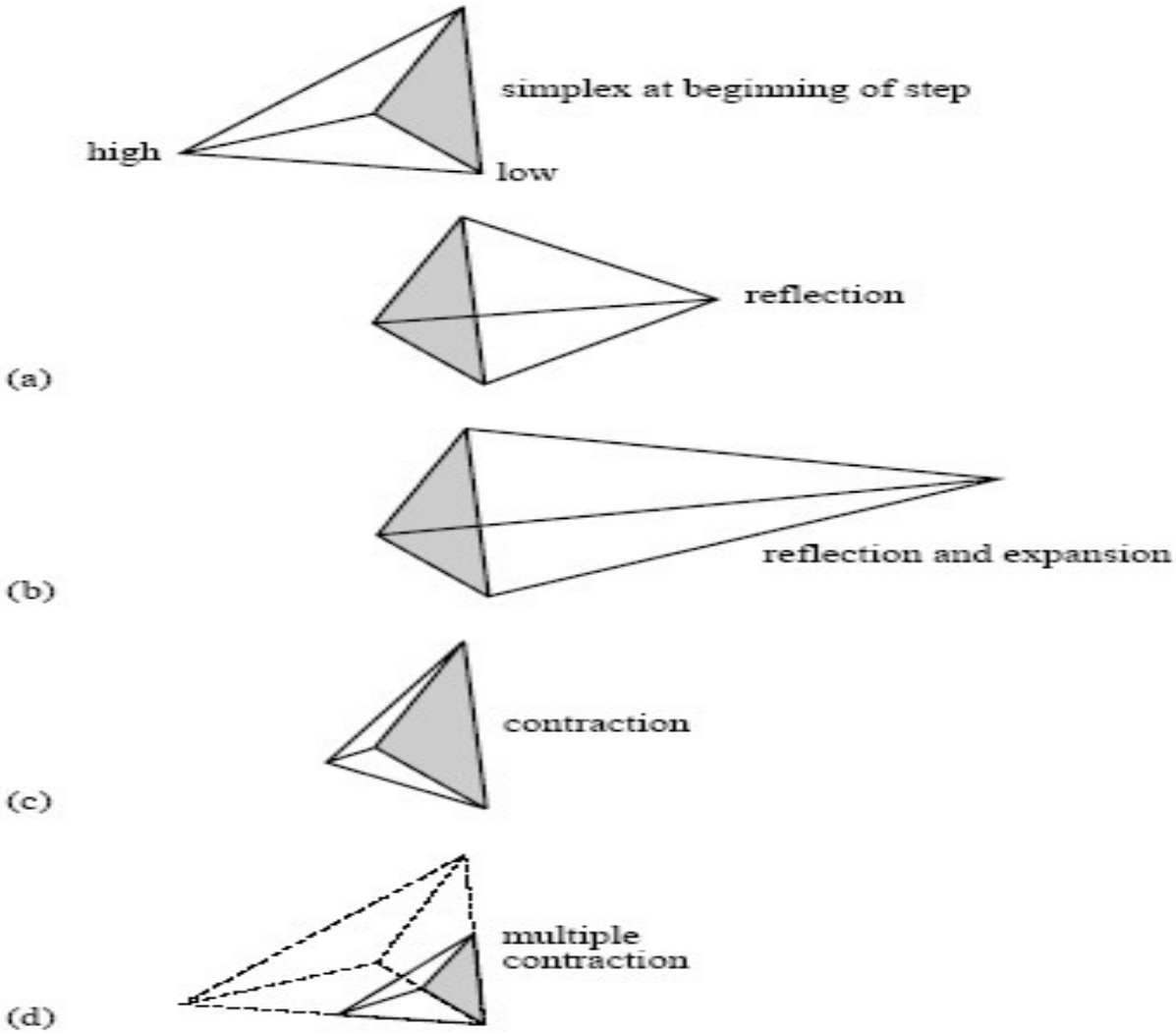
  一个单纯形是一个几何形体，它在N维情况下是由N+1个顶点所相互连接的线段以及多边形面所组成的几何形体。二维情况单纯形即为三角形，三维情况则为一个四面体，但不一定必须是规则的四面体。

昊瑞电子

- 下降的单纯形法必须从N+1个点儿不仅仅是单个点开始迭代，这N+1个点定义了一个初始的单纯形，如果把其中一个点（哪一个无关紧要）作为初始点P_0，则其他N个点可取为：

  $P_i = P_0 + r*E_i$，其中E_i为N维单位向量，r为一常数，它是对问题特征大小长度的估计值（可以取相同的也可以取不同的值）

- 下降的单纯形法迭代过程中绝大多数的步骤就是：将函数值达到最大的单纯形的点（即最高点）通过单纯形的背向面移到一个较低点，这个步骤称为反射，之后，将对单纯形在某个方向上进行扩展以加大步长，当到达“谷底”时，单纯形将自行作横向收缩，且自行拉向最低点（即最佳点）附近。算法结束条件就是函数值下降的幅度小于某个阈值或达到最大迭代次数。
- 看下图


simplex at beginning of step
high
low
reflection
(a)
reflection and expansion
(b)
contraction
(c)
multiple
contraction
(d)

Rescore中单纯形算法的应用

- 首先假定BLEU=f(特征的和)，对n+1个顶点（n维向量）分别计算BLEU值（取相反数），然后从中选出BLEU(相反数)最大，次大和最小的三个点，算法每次都是把其中的最大点对应的各权重进行调整，使其变小向最小点靠拢，调整完毕后，计算其对应的BLEU，再从这些BLEU中选出BLEU(相反数)最大，次大和最小的三个点，一直迭代下去，直到最高点到最低点的比率范围合适或达到最大迭代次数为止。
- 具体过程如下：

- 单纯形算法的流程：

  假设有N个特征，则构建一个有N+1个顶点的单纯形，每个顶点为N维，即N个特征权重，算法流程如下：

  首先初始化N+1个顶点，假设N0为原点，其他顶点根据N0推得：

- 1.

  接着利用这N+1个顶点算出N+1个函数值Yi（i=1,2….N+1），从这N+1个函数之中选出最大值Ymax，最小值Ymin，和次大值Ysecond并记录其对应的顶点，然后就正式开始单纯形算法的迭代求极小值的过程。

- 2.
- 对当前最大值的顶点进行反射，求得一个新的点，该点对应的函数值记为Y反射，
- if(Y反射<Ymin)
- {//如果反射点比当前最好的点还要好，则用gamma再对反射点做一下外推扩展，得到一个新的点，外推点，该点对应的函数值记为Y外推
- if(Y外推< Ymin)
- {//如果外推点好于当前最小点，则以外推点作为新的单纯形中的一个顶点，此时把函数值最大的点从单纯形中替掉，返回1，重新选择三个值及三个顶点}
- else
- {//如果外推点不如当前最小点，则直接以反射点作为新的单纯形中的一个顶点，此时也把函数值最大的点从单纯形中替掉，返回1，重新选择三个值及三个顶点}
- }
- else
- {//如果反射点比当前最好的点要差，则拿反射点和当前次好的点Ysecond比较
- if((Y反射>Ysecond)
- {//如果反射点也不如当前次好点,，则比较反射点和当前最差的点
- if(Y反射<Ymax)
- {//如果反射点比当前最差的点要好， Ymax = Y反射，同时对反射点用beta做一次一维的收缩，得到的函数值记为Y收缩 }
- else
- {//对最差点用beta做一次一维的收缩，得到的函数值记为Y收缩 }
- if(Y收缩< Ymax)//（此处的Ymax已经在上面被替换过了），如果收缩后的点比较好，则正式将原单纯形中的最差点用该收缩点替掉，返回1，重新选择三个值及三个顶点。
- else//如果收缩后的点不好，算法认为原来的单纯形不好，需要对各顶点同时收缩，收缩后再计算出个顶点所对应的函数值，然后返回1，重新选择三个值及三个顶点
- }
- }
- else
- {//如果反射点好于当前的次好点，则以反射点替掉单纯形中的最差点，返回1，重新选择三个值及三个顶点。}
- }
- 3．直到函数值的下降幅度小于某一个阈值或达到最大的迭代次数为止。

昊瑞电子

- 我觉得单纯形算法使用中的一个问题就是初始顶点值的选择和r的选择(Ni= N0+r*Ei)，它没有一个固定的选择标准，在rescore中不同的选择得到的最终结果中最好和最差之间一般会差1%左右

- 参考文献

1. The ITC-irst SMT System for IWSLT-2005, B.Chen,R.Cattoni,N.Bertoldi,M.Cettololo,M.Federico

2. A Look inside the ITC-irst SMT System, M. Cettolo, M. Federico, N. Bertoldi, R. Cattoni and B. Chen

3. A Word-to-Word Model of Translational Equivalence, I .Dan.Melamed

4. C语言数值算法大全。P346-P350

昊瑞电子

谢谢！