# CRF-based Chinese Word Segmentation for CIPS-SIGHAN-2014 Bakeoff

Guohua Wu, Dezhu He, Keli Zhong, Xue Zhou and Caixia Yuan
School of Computer,
Beijing University of Posts and Telecommunications,
China, 100876
trustwugh@gmail.com, hedezhu@bupt.edu.cn, zhk@126.com
bupt.zhouxue@gmail.com, yuancx@bupt.edu.cn

## Abstract

本文描述了我们在第三届 CIPS-SIGHAN 中文处理资源与评测国际会议中文分词评测任务中使用的系统。本文采用字符序列标注的方式进行中文切分，为了提高模型的跨领域分词能力，我们的系统集成了有监督特征，无监督特征和词典特征，对目标领域数据切分之后，从中发现新词，更新词典，然后对目标数据进行再分词，从而进一步提高对未登录词的处理能力。本文为所有的特征采用了一套统一的特征模板，使用 CRF 训练标注模型。在多个数据集上的测试结果证明我们的系统具有良好的切分性能，我们的系统在本次 Bakeoff 测试集上的 F 值是 0.9730。

## 1 简介

中文信息处理需要以词为基本单位，但是中文在书写时并不会显式地保留词与词之间的边界信息，所以中文分词（CWS）是中文信息处理中非常关键的第一步。一系列研究表明使用字符序列标注的方法能够简单有效地描述中文分词任务 (Xue et al., 2003; Peng et al., 2004; Low et al., 2005; Zhao et al., 2006a)，其中基于 CRF(Lafferty et al., 2001) 序列标注进行分词的方法得到广泛使用，并且具有良好的性能。然而，目前分词系统的泛化能力仍然不够强，并不存在一个无领域限制（domain-independent）的分词系统。因为一般来说我们只能在限定的领域获取到训练语料，当测试语料的用语、用词相差很大的时候，就会导致我们的系统分词性能大大降低。最直接的解决办法是标注多个领域的语料加入训练集，但是由于手工标注语料的成本很高，所以标注大量训练语料并不现实。

目前有两种思路改善跨领域分词性能，第一种方法由 (Zhao and Kit, 2007, 2008, 2011) 提出，他们提出了一个统一的框架把把无监督分词和有监督分词集成在一起，可以充分利用无监督分词方法从无标语料中发现新词的能力和有监督分词方法识别已知词的能力，从一定程度上增强分词系统的泛化能力；第二种方法是构建一个多层次的分词系统，第一层是分词子系统，这些子系统在不同领域上分词性能不尽相同，每个子系统可能都有一个表现特别好的领域，然后第二层结合所有子系统的输出，确定在测试数据集上的最佳分词边界，Gao and Vogel (2010) 通过这种方式在 Bakeoff-2010(Zhao and Liu, 2010) 的四个测试领域中的三个取得第一。本文主体按照第一种思路来改善跨领域分词性能，其中我们对无监督特征的使用方法进行了改进，并且加入了第二种方法中使用到的一些有效的特征，通过词典特征和新词发现提高未登录词切分能力。

本文组织如下，第 2节详细描述了我们使用的特征和我们对无监督特征使用熵的改进。第 3节阐述了本文如何从初步切分结果中发现新词，然后扩展词典更新词典特征，再次对测试数据进行分词来提高新词的切分性能。第 4节是我们的系统在 Bakeoff 数据集上的实验结果并且与官方公布的最佳分词性能对比。第 5节总结全文。

| Word length | Tag sequence for a word |
|---|---|
| 1 | S |
| 2 | BE |
| 3 | B$B_2$E |
| 4 | B$B_2B_3$E |
| 5 | B$B_2B_3$ME |
| $\geq 6$ | B$B_2B_3$M$\cdots$ME |

表 1: Illustration of character tagging

## 2 系统描述

本文中文分词看做序列标注问题，然后使用 CRF 来训练分词模型。本文使用 Taku Kudo 实现的 $CRF++$[1]工具包来训练模型。本文把‘，’，‘。’，‘？’，‘！’，‘；’当做句子边界，训练和测试语料都按照上述句子边界切成句子。

Zhao et al. (2006b) 证明了使用 6-tag set 训练的 CRF 模型分词性能优于使用其他 tag set 训练的模型。所以在本文中我们采用 6-tag（B，$B_2$，$B_3$，M，E，S）方法对词中的字符进行标注。表 1解释了对于不同长度的词如何对词中的字符进行打标。在本文中我们将沿用 (Zhao et al., 2006b; Zhao and Kit, 2008) 中使用的 6 个 n-gram 字符特征，分别是：$C_{-1}$，$C_0$，$C_1$，$C_{-1}C_0$，$C_0C_1$，$C_{-1}C_1$，其中 $C$ 表示字符，下标 -1，0 和 1 分别表示前一个字符，当前字符和下一个字符。本节接下来的部分将介绍我们的系统使用到的特征，对于其他特征本文将使用与字符特征类似的 6 个 n-gram 特征模板。

### 2.1 字符类型特征

我们根据字符的 Unicode 编码把字符简单地分成 5 类：中文字符（C），英文字符（E），数字[2]（N），标点符号（P）和其他字符（O）。我们把字符类型特征记作 $CTF$，对于该特征我们使用的特征模板与字符类型特征类似，特征模板为：$CTF_{-1}$，$CTF_0$，$CTF_1$，$CTF_{-1}CTF_0$，$CTF_0CTF_1$，$CTF_{-1}CTF_1$。

### 2.2 条件熵特征

Gao and Vogel (2010) 在 CIPS-SIGHAN CLP 2010 Bakeoff(Zhao and Liu, 2010) 数据集上使用条件熵特征提高了分词性能。对于特定字符 $C$ 前向条件熵指的是在整个语料中 $C$ 的下一个位置出现的所有字符的熵，记为 $H_f(C)$。后向条件熵计算的则是整个语料中 $C$ 的上一个位置出现的所有字符的熵，记为 $H_b(C)$。在实际系统中我们可以混合多个领域的无标语料来计算前后向条件熵使得该特征具有一定的领域适应性，我们通过统计基于字符的 Bigram 矩阵来辅助计算前后向条件熵。

条件熵是一个连续值，我们使用 (Gao and Vogel, 2010) 中提出的离散化方法，使用如下的熵区间到离散值的映射：$[0, 1.0) \mapsto 0$，$[1.0, 2.0) \mapsto 1$，$[2.0, 3.5) \mapsto 2$，$[3.5, 5.0) \mapsto 4$，$[5.0, 7.0) \mapsto 5$，$[7.0, +\infty) \mapsto 6$。我们使用与字符特征类似的模板，前后条件熵的模板一致，这里只列出前向条件熵的特征模板：$H_f(C_{-1})$，$H_f(C_0)$，$H_f(C_1)$，$H_f(C_{-1})H_f(C_0)$，$H_f(C_0)H_f(C_1)$，$H_f(C_{-1})H_f(C_1)$。

### 2.3 词典特征

通过适当的方法把词典量化成特征能够提高分词性能，我们按照 (Shi and Wang, 2007; Gao and Vogel, 2010) 的方式生成词典特征。我们使用 $L_{begin}(C)$ 和 $L_{end}(C)$ 特征，前者表示的是在当前句子中 $C$ 字符位置开始进行前向最大匹配，寻找在词典中以 $C$ 为开头的词的最大长度；后者表示的是从 $C$ 字符位置开始进行后向最大匹配，寻找在词典中以 $C$ 为结尾的词的最大长度。进行前后向最大长度匹配时只匹配长度大于等于 2 的词，没有匹配上则认为特征值为 0。特别地，当特征值大于等于 6 时，我们将特征值统一设定为 6。我们希望通过使用一个大规模的跨领域词典提高分词性能。对于 $L_{begin}(C)$ 我们定义六个特征模板：$L_{begin}(C_{-1})$，$L_{begin}(C_0)$，$L_{begin}(C_1)$，$L_{begin}(C_{-1})L_{begin}(C_0)$，$L_{begin}(C_0)L_{begin}(C_1)$，$L_{begin}(C_{-1})L_{begin}(C_1)$。$L_{end}(C)$ 的六个特征模板与之类似，不再赘述。

### 2.4 Accessor variety feature

Feng et al. (2004) 提出的 Accessor variety (AV) 可以用来度量一个子串是一个中文词的可能性。Zhao and Kit (2008, 2011) 认为这种思想与 Harris 从 unfamiliar language 中发现词素的思路一致 (Harris, 1970)。Zhao and Kit (2008) 的实验证明 AV 特征在 Bakeoff-2003，Bakeoff-2005，Bakeoff-2006 数据集 (Sproat and Emerson, 2003;

[1] http://crfpp.googlecode.com/svn/trunk/doc/index.html

[2] 本文定义的数字包括阿拉伯数字 0 到 9 和中文数字零到九

| In | Accessor Variety Feature Selection | | | | | | | | | | T |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | 1 char | | 2 char | | 3 char | | 4 char | | 5 char | | |
| 而 | 9 | 9 | 5 | 5 | 2 | 2 | 0 | 0 | 0 | 0 | S |
| 在 | 10 | 10 | 5 | 5 | 2 | 1 | 1 | 1 | 1 | 1 | S |
| 我 | 9 | 9 | 5 | 3 | 2 | 2 | 1 | 0 | 1 | 0 | S |
| 心 | 8 | 8 | 5 | 5 | 2 | 2 | 1 | 0 | 1 | 0 | B |
| 中 | 9 | 9 | 8 | 8 | 2 | 0 | 1 | 0 | 1 | 0 | E |
| ， | 11 | 11 | 8 | 0 | 2 | 0 | 0 | 0 | 1 | 0 | S |

表 2: AV 特征使用上的对比

Emerson, 2005; Levow, 2006) 上都使得 CRF 模型的分词效果得到提高，并且在 Bakeoff-2008(Chen and Jin, 2008) 的封闭测试上获得了最佳性能，因此本文使用 AV 特征，并改进 AV 特征的使用方法，使得分词性能进一步提高。对于子串 s，AV 值的定义如下：

$$AV(s) = min\{L_{av}(s), R_{av}(s)\}$$

$L_{av}(s)$ 和 $R_{av}(s)$ 分别表示子串 s 之前的不同字符数目和之后的不同字符数目，但是句子开头和结尾标记会被重复计算。

本文同 (Zhao and Kit, 2008; Yang et al., 2011) 中使用 AV 的方式类似，考虑句子中长度小于等于 5 的子串的 AV 值，并且对应的设计多个特征模板，使用以下的方式对子串 $s$ 的 AV 值进行离散化：

$$f_n(s) = t, \text{ if } 2^t \leq AV(s) < 2^{t+1}$$

把离散化得到的 $t$ 当做特征值。本文在该特征的使用上有一点与他们不同，对于一个子串 $s$，我们把 $s$ 的特征值标记在 $s$ 的第一个字符上，而不是为每个字符都标上 $s$ 的特征值，这种标记方法借鉴了第2.3节词典特征的表示方法，我们认为这样标记更能突出词的边界信息。具体区别如表 2所示，例如当我们考虑长度为 4 个字符的所有子串时，其中有一个子串是 “在我心中” 的 AV 特征值是 $t = 1$，所以我们对 “4 char” 子串那一列的特征值进行更新，该列的左边是表示的是 (Zhao and Kit, 2008; Yang et al., 2011) 中使用的方法需要对子串的每个字符 “在”，“我”，“心” 和 “中” 都赋予特征值 1；该列的右边表示的是我们的方法得到的特征值，只对子串的第一个字符 “在” 赋予特征值 1。本文为表 2 每一列都构造了与字符特征模板类似的 6 个模板。

为了证明本文对 AV 特征的改进是有效的，我们按照 (Zhao and Kit, 2008; Yang et al., 2011) 的实验设置方式在 Bakeoff-2005(Emerson, 2005) 和 Bakeoff-2010(Zhao and Liu, 2010) 的简体中文数据集上进行实验，把他们的 AV 特征记为 OldAV，我们的记为 NewAV，每次实验我们都使用了 6 个 n-gram 字符特征和第 2.1节的字符类型特征，评估指标为 F 值 $F = 2RP/(R + P)$ 其中 $R$ 为召回率 $P$ 为准确率。在 Bakeoff-2005 数据集上把对应的训练和测试集合并起来，去分词标记之后用来统计 AV 特征；将 Bakeoff-2010 提供的训练语料，无标语料和测试语料合并起来，去分词标记之后用来统计 AV 特征。表 3的实验结果表明我们对 AV 特征的改进是有效的，分词性能要比原来的使用方法好。为了更好地对比模型的切分性能，这里的实验结果没有使用后处理。

| Bakeoff-2005 | | AS | CityU | MSRA | PKU |
|---|---|---|---|---|---|
| Baseline | F | 0.954 | 0.955 | 0.971 | 0.950 |
| | $R_{OOV}$[1] | 0.700 | 0.798 | 0.772 | 0.778 |
| OldAV | F | 0.957 | 0.961 | 0.973 | 0.952 |
| | $R_{OOV}$ | 0.688 | 0.807 | 0.747 | 0.770 |
| NewAV | F | 0.957 | 0.964 | 0.973 | 0.954 |
| | $R_{OOV}$ | 0.688 | 0.822 | 0.743 | 0.773 |
| Bakeoff-2010 | | A | B | C | D |
| Baseline | F | 0.921 | 0.93 | 0.918 | 0.953 |
| | $R_{OOV}$ | 0.629 | 0.773 | 0.72 | 0.853 |
| OldAV | F | 0.933 | 0.94 | 0.935 | 0.956 |
| | $R_{OOV}$ | 0.656 | 0.784 | 0.77 | 0.848 |
| NewAV | F | 0.935 | 0.945 | 0.936 | 0.956 |
| | $R_{OOV}$ | 0.659 | 0.807 | 0.763 | 0.843 |

[1] Recall of out-of-vocabulary (OOV) words.

表 3: AV 特征对比实验，每个实验结果均使用了字符 n-gram 特征和字符类型特征

## 2.5 后处理

后处理的目的是处理英文单词，阿拉伯数字串，网址的切分错误。这种情况一般要将这些字符整体当作一个切分单位，但是本文的系统可能会出现错误切分的情况，例如表 4里面的网址切分错误，表中 raw 表示原始句子，result 表示模型切分后的结果，final 表示后处理之后的结果。处理这类错误的思想是分词后的句子去空格后能和原始句子的字符一一对应上，下面简要描述修复网址切分错误的步骤，把原始句子存储到一个字符串里面，把

| | |
|---|---|
| raw | 点击网址 http://t.cn/aBPxzO |
| result | 点击　网址　http://t.cn/aB　PxzO |
| final | 点击　网址　http://t.cn/aBPxzO |

表 4: 特殊字串（网址）后处理

分词结果存储到一个列表，列表的每个元素是一个词，下标从 0 开始。

1. 正则匹配网址 “http://t.cn/aBPxzO” 在原始句子中的开始和结束位置，分别是 4 和 22。
2. 从左往右把词列表中的词的长度累加起来，可以求得网址在词列表的开始下标是 2，结束下标是 3。
3. 最后把词列表的 2，3 位置的词合并成一个词

英文单词和阿拉伯数字串按照同样的思路处理。

# 3　提高新词切分性能

本文第 2节描述的分词系统在新词切分上存在一定的不稳定性，新词在某些上下文能够正确切分，但是在另一些上下文却会被错误切分。例如，“涅维拉济莫夫” 这个词，在 “文官涅维拉济莫夫在起草一封贺信” 这个句子中能被正确切分，但是在 “于是涅维拉济莫夫开始绞尽脑汁” 这个句子中却被切分错了。我们处理这类问题的方法是通过规则找出这种新词，然后将这些新词加入到词典中，重新计算测试语料的词典特征，最后对测试语料进行再次切分。假设训练分词模型的时用于提取词典特征的词典记为 $Lexicon_{train}$，根据人民日报 1998 年和 2000 年的已切分语料统计 Bigram 记为 $PKU_{bigram}$，对于初步切分结果中的词 $w$ 满足下列条件的，我们认为 $w$ 是新词：

1. ($w$ 长度在 2 到 6 之间) or ($w$ 的长度大于 6 且 $w$ 是外国人名（$w$ 中有外国人名连字符 “•”）)
2. $w$ 没有在 $Lexicon_{train}$ 中出现
3. $w$ 不是中国人姓名
4. $w$ 的任意拆分都不能在 $PKU_{bigram}$ 中查找到

按照上述的规则对切分结果中的所有词进行检查，得到一份新词列表记为 $Lexicon_{test}$，最后对 $Lexicon_{test}$ 做一个过滤，若 $Lexicon_{test}$ 中的两个词具有包含关系则只保留长度大的词。把 $Lexicon_{train}$ 和 $Lexicon_{test}$ 进行合并得到一份新的词表 $Lexicon_{new}$，用这个词表为测试语料计算词典特征，重新生成切分结果。

| Name | Features | Lexicon |
|---|---|---|
| Baseline | CF,CTF | None |
| Closed | CF,CTF,EF,AV | None |
| Open[1] | CF,CTF,EF,AV | Webdict |
| Refine[2] | CF,CTF,EF,AV | Webdict |

[1] 测试和训练的时候都使用 Webdict 计算词典特征。

[2] 训练的时候使用 Webdict 计算词典特征，然后按照第 3节的方式提高新词切分性能。

表 5: 特征组合，CF 表示字符的 6 个 n-gram 特征，CTF 表示字符类型特征，EF 表示条件熵特征，AV 表示 Accessor variety 特征

# 4　实验

为了证明我们的系统的分词性能，我们考虑表 5 中的四种特征组合，Closed 表示的是封闭测试，Open 表示的是开放测试因为我们使用到一个跨领域词典 Webdict[3]，Refine 表示我们在 Open 的基础上再使用第 3节的新词处理方法，由于我们的方法以需要一份语料统计 Bigram 和训练时使用的词表，这两种资源都使用的是简体中文语料，所以我们并没有对 Bakeoff-2005 的 AS 和 CityU 两份繁体中文语料做测试。本节所有的实验都使用了第 2.5节所述的后处理，我们在 Bakeoff-2005 和 Bakeoff-2010 数据集上进行了实验，主要度量指标是 F 值。

表 6是在 Bakeoff-2005 上的测试结果，计算条件熵特征和 AV 特征时，需要把对应的测试语料和训练合并起来去分词标记后，再统计得到。“Best closed” 表示的是在 Bakeoff-2005 中官方公布的最佳封闭测试结果，“Best open” 则表示官方公布的最佳开放测试结果。我们的封闭测试结果全面超过了 “Best closed”，开放测试上只有 PKU 测试集上结果比 “Best open” 差，这可能是训练语料不足的缘故，可以通过增大 PKU 的训练语料提高性能。

表 7是在 Bakeoff-2010 简体中文数据集的测

[3] https://github.com/ling0322/webdict

| Bakeoff-2005 | | AS | CityU | MSRA | PKU |
|---|---|---|---|---|---|
| Best closed | F | 0.952 | 0.943 | 0.964 | 0.95 |
| | $R_{OOV}$ | 0.696 | 0.698 | 0.717 | 0.636 |
| Baseline | F | 0.955 | 0.956 | 0.971 | 0.950 |
| | $R_{OOV}$ | 0.708 | 0.806 | 0.772 | 0.779 |
| Closed | F | 0.957 | 0.963 | 0.974 | 0.954 |
| | $R_{OOV}$ | 0.705 | 0.817 | 0.739 | 0.770 |
| Open | F | 0.958 | 0.965 | 0.977 | **0.962** |
| | $R_{OOV}$ | 0.700 | 0.811 | 0.751 | 0.765 |
| Refine | F | - | - | **0.976** | 0.962 |
| | $R_{OOV}$ | - | - | 0.751 | 0.766 |
| Best open | F | 0.956 | 0.962 | 0.972 | 0.969 |
| | $R_{OOV}$ | 0.684 | 0.806 | 0.59 | 0.838 |

表 6: Bakeoff-2005 数据集上的测试结果

| Bakeoff-2010 | | A | B | C | D |
|---|---|---|---|---|---|
| Best closed | F | 0.946 | 0.951 | 0.939 | 0.959 |
| | $R_{OOV}$ | 0.816 | 0.827 | 0.75 | 0.827 |
| Baseline | F | 0.921 | 0.933 | 0.918 | 0.954 |
| | $R_{OOV}$ | 0.629 | 0.781 | 0.72 | 0.86 |
| Closed | F | **0.935** | 0.949 | 0.936 | 0.958 |
| | $R_{OOV}$ | 0.658 | 0.819 | 0.763 | 0.853 |
| Open | F | **0.95** | 0.949 | 0.943 | 0.963 |
| | $R_{OOV}$ | 0.509 | 0.766 | 0.571 | 0.879 |
| Refine | F | 0.95 | 0.949 | 0.943 | 0.963 |
| | $R_{OOV}$ | **0.519** | **0.768** | **0.572** | **0.883** |
| Best open | F | 0.955 | 0.95 | 0.938 | 0.96 |
| | $R_{OOV}$ | 0.655 | 0.82 | 0.768 | 0.847 |

表 7: Bakeoff-2010 数据集上的测试结果

试结果，计算条件熵特征和 AV 特征时，需要把 Bakeoff-2010 提供的所有简体中文语料合并起来去标注之后再统计特征值。“Best closed” 和 “Best open” 分别表示官方公布的封闭测试和开放测试最佳结果，我们的封闭测试结果在 A 测试集上与 “Best closed” 差距比较大，但是在其他测试集上效果都比较接近 “Best cloesed”，而且相比 Baseline，Close 在测试集上性能都提高了很多。在开放测试我们的方法在 C，D 测试集上效果超过 “Best open”，在 B 测试集上性能相当，在 A 还存在一些差距。

从表 6和表 7的 Open 实验结果和 Refine 实验结果的对比可以看出，发现新词之后加入词表再进行分词的这种方法能够在保持 $F$ 值的前提下略微提升对 OOV 的处理能力。

通过在两个 Bakeoff 数据集上的实验证明了我们的 Open 特征组合和 Refine 特征组合的有效性，由于本次 Bakeoff 没有提供训练语料，属于开放测试，所以我们使用表 5中的 Open 和 Refine 特征组合方法。为了让模型具有更好的跨领域效果我们使用了大量的无标语料提取条件熵特征和 AV 特征，用网络爬虫抓取财经，文学，新闻，微博，小说 5 个领域的语料，每个领域抓取约 300MB。定义如何的符号代表各种语料集合：

- PKU-Corpus：人民日报 1998 和 2000 年有标语料
- PKU-Raw：PKU-Corpus 去分词标记后的结果
- Web-Corpus: 爬虫抓取到的所有无标语料合并在一起
- Sample-Corpus: 从 Web-Corpus 中随机抽取 15%
- Entropy-Corpus：PKU-Raw 加上 Web-Corpus
- AV-Corpus：PKU-Raw 加上 Sample-Corpus

| | Precision | Recall | F Score |
|---|---|---|---|
| Open | 0.9673 | 0.9776 | 0.9724 |
| Refine | 0.9681 | 0.9779 | 0.9730 |

表 8: Our results in this bakeoff

最后我们使用以 PKU-Corpus 作为训练语料，用 Entropy-Corpus 提取条件熵特征，用 AV-Corpus 提取 AV 特征，再加上字符特征和字符类型特征训练 CRF 分词模型。本文的系统在本次 Bakeoff 测试集上的切分性能如表 8所示，我们的最佳 F 值是 0.9730，从表中可以看出 Refine 特征组合取得的结果要比 Open 好，再次验证了第 3节的新词发现方法可以提高切分性能。

## 5 总结

本文的主要目标是实现一个跨领域的分词系统，我们的系统把有监督分词和无监督全局特征集成在一起，并且通过使用跨领域词典特征提高了模型对未登录词的识别能力。再通过新词发现的方式从目标测试集中发现新词，然后重新计算词典特征

对目标测试集再次切分，进一步提高模型的领域适应性。

目前我们的系统仍然有很多的不足之处，可以从以下三个方面进行改进。首先我们只使用了一种无监督特征，可能还存在更好的无监督特征，或者通过组合多种无监督特征的方法可以得到更好的性能；其次本文对所有的特征都使用一套类似的模板，这样做有一个好处就是简单，但是不同的特征可能有更适用于自己的特征模板；最后我们的新词过程是基于规则的，可以改进成通过模型自动发现新词。

# 参考文献


Xiao Chen and Guangjin Jin. The fourth international chinese language processing bakeoff: Chinese word segmentation named entity recognition and chinese pos tagging. In *Proceedings of the Sixth SIGHAN Workshop on Chinese Language Processing*, 2008.

Thomas Emerson. The second international chinese word segmentation bakeoff. In *Proceedings of the Fourth SIGHAN Workshop on Chinese Language Processing*, volume 133, 2005.

Haodi Feng, Kang Chen, Xiaotie Deng, and Weimin Zheng. Accessor variety criteria for chinese word extraction. *Computational Linguistics*, 30(1):75–93, 2004.

Qin Gao and Stephan Vogel. A multi-layer chinese word segmentation system optimized for out-of-domain tasks. In *Proceedings of CIPS-SIGHAN Joint Conference on Chinese Language Processing (CLP2010)*, pages 210–215, 2010.

Zellig S Harris. *Morpheme boundaries within words: Report on a computer test*. Springer, 1970.

John Lafferty, Andrew McCallum, and Fernando CN Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. 2001.

Gina-Anne Levow. The third international chinese language processing bakeoff: Word segmentation and named entity recognition. In *Proceedings of the Fifth SIGHAN Workshop on Chinese Language Processing*, pages 108–117, Sydney, Australia, July 2006. Association for Computational Linguistics. URL `http://www.aclweb.org/anthology/W/W06/W06-0115`.

Jin Kiat Low, Hwee Tou Ng, and Wenyuan Guo. A maximum entropy approach to chinese word segmentation. In *Proceedings of the Fourth SIGHAN Workshop on Chinese Language Processing*, volume 1612164, 2005.

Fuchun Peng, Fangfang Feng, and Andrew McCallum. Chinese segmentation and new word detection using conditional random fields. In *Proceedings of the 20th international conference on Computational Linguistics*, page 562. Association for Computational Linguistics, 2004.

Yanxin Shi and Mengqiu Wang. A dual-layer crfs based joint decoding method for cascaded segmentation and labeling tasks. In *IJCAI*, pages 1707–1712, 2007.

Richard Sproat and Thomas Emerson. The first international chinese word segmentation bakeoff. In *Proceedings of the Second SIGHAN Workshop on Chinese Language Processing*, pages 133–143, Sapporo, Japan, July 2003. Association for Computational Linguistics. doi: 10.3115/1119250.1119269. URL `http://www.aclweb.org/anthology/W03-1719`.

Nianwen Xue et al. Chinese word segmentation as character tagging. *Computational Linguistics and Chinese Language Processing*, 8(1):29–48, 2003.

Ting-hao Yang, Tian-Jian Jiang, Chan-hung Kuo, Richard Tzong-han Tsai, and Wen-lian Hsu. Unsupervised overlapping feature selection for conditional random fields learning in chinese word segmentation. In *Proceedings of the 23rd Conference on Computational Linguistics and Speech Processing*, pages 109–122. Association for Computational Linguistics, 2011.

Hai Zhao and Chunyu Kit. Incorporating global information into supervised learning for chinese word segmentation. In *10th Conference of the Pacific Association for Computational Linguistics*, pages 66–74, 2007.

Hai Zhao and Chunyu Kit. Unsupervised segmentation helps supervised learning of character tagging for word segmentation and named entity recognition. In *The Sixth SIGHAN Workshop on Chinese Language Processing*, pages 106–111, 2008.

Hai Zhao and Chunyu Kit. Integrating unsupervised and supervised word segmentation: The role of goodness measures. *Information Sciences*, 181(1):163–183, 2011.

Hai Zhao, Chang-Ning Huang, and Mu Li. An improved chinese word segmentation system with conditional random field. In *Proceedings of the Fifth SIGHAN Workshop on Chinese Language Processing*, volume 1082117. Sydney: July, 2006a.

Hai Zhao, Chang-Ning Huang, Mu Li, and Bao-Liang Lu. Effective tag set selection in chinese word segmentation via conditional random field modeling. In *Proceedings of PACLIC*, volume 20, pages 87–94, 2006b.

Hongmei Zhao and Qun Liu. The cips-sighan clp 2010 chinese word segmentation bakeoff. In *Proceedings of the First CPS-SIGHAN Joint Conference on Chinese Language Processing*, pages 199–209, 2010.