自然语言处理

信息抽取（命名实体识别部分）

孙承杰 杨沐昀
sunchengjie@hit.edu.cn
哈尔滨工业大学计算学部
语言技术研究中心

主要内容

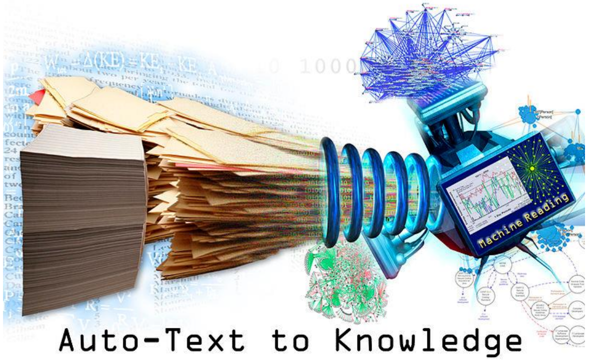
- 信息抽取的定义、任务及发展
- 命名实体识别
- 实体链接
- 关系抽取

主要内容

- □信息抽取的定义、任务及发展
- □命名实体识别
- □实体链接
- □关系抽取

信息抽取（IE)

- The goal of IE is to automatically extract structured information, i.e. categorized and contextually and semantically well-defined data from a certain domain, from unstructured machine-readable documents.
- 无结构数据结构化



信息抽取中的主要任务

- 命名实体识别
  - 识别和分类文本中出现的“实体提及”

昨天下午，市政协、市委统战部联合举办北京市全国政协委员视察考察活动，围绕历史文化街区改造和疏解整治促提升专项行动进展等视察并座谈。市政协主席吉林参加。

昨天下午，<ORG>市政协</ORG>、<ORG>市委统战部</ORG>联合举办<LOC>北京市</LOC><ORG>全国政协</ORG>委员视察考察活动，围绕历史文化街区改造和疏解整治促提升专项行动进展等视察并座谈。<ORG>市政协</ORG>主席<PER>吉林</PER>参加。

信息抽取中的主要任务

□实体链接

- 将“实体提及”链接到知识库中对应的实体

昨天下午，<ORG>市政协</ORG>、<ORG>市委统战部</ORG>联合举办<LOC>北京市</LOC> <ORG>全国政协</ORG>委员视察考察活动，围绕历史文化街区改造和疏解整治促提升专项行动进展等视察并座谈。 <ORG>市政协</ORG>主席<PER>吉林</PER>参加。

<PER>吉林</PER> → 吉林（北京市政协党组书记、主席）

信息抽取中的主要任务

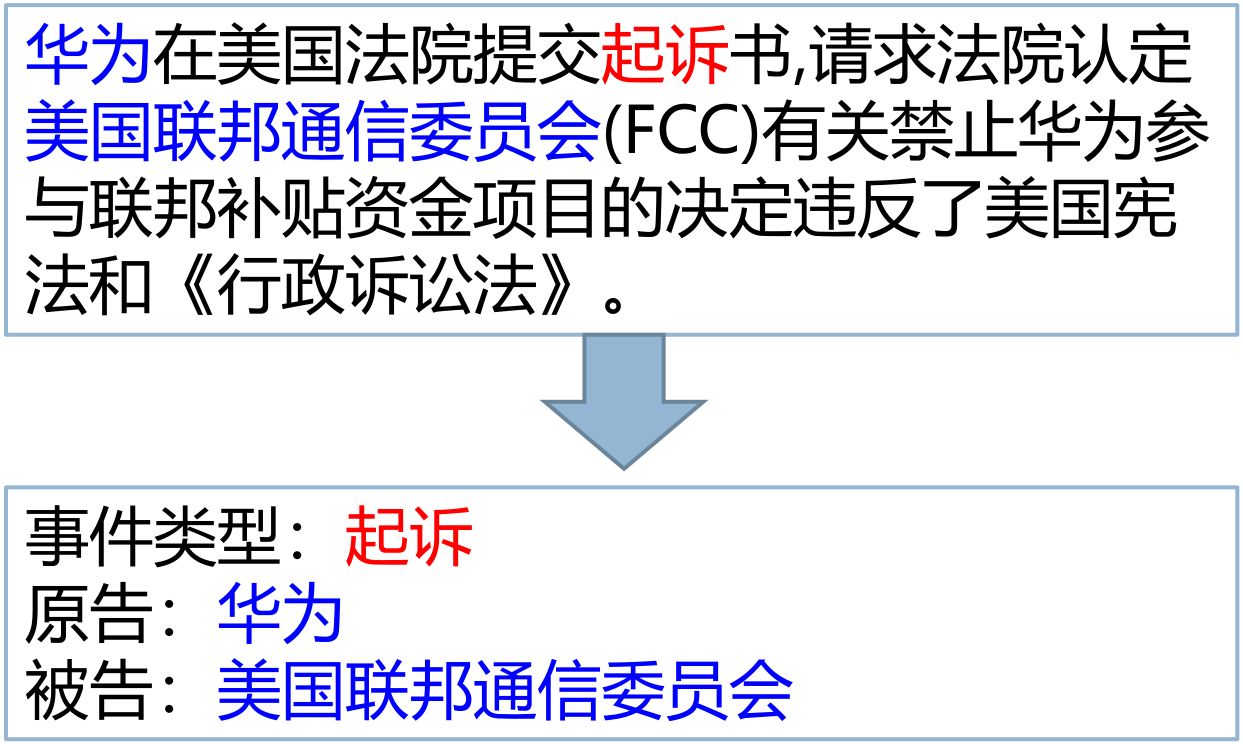
- 关系抽取
  - 找到句子中有关系的两个实体，并识别出他们之间的关系类型

昨天下午，<ORG>市政协</ORG>、<ORG>市委统战部</ORG>联合举办<LOC>北京市</LOC><ORG>全国政协</ORG>委员视察考察活动，围绕历史文化街区改造和疏解整治促提升专项行动进展等视察并座谈。<ORG>市政协</ORG>主席<PER>吉林</PER>参加。

R = （市政协，主席，吉林）

信息抽取中的主要任务

- 事件抽取
  - 事件的元素包括: 触发词、事件类型、论元及论元角色。
  - 事件抽取就要是找到一个事件对应的元素。



华为在美国法院提交起诉书,请求法院认定美国联邦通信委员会(FCC)有关禁止华为参与联邦补贴资金项目的决定违反了美国宪法和《行政诉讼法》。

事件类型：起诉
原告：华为
被告：美国联邦通信委员会

信息抽取的发展

- MUC (Message Understanding for Comprehension)
- MET (Multilingual Entity Task Evaluation)
- ACE (Automatic Content Extraction)
- DUC (Document Understanding Conferences)
- TAC (Text Analysis Conference)

MUC (1)

- 美国政府支持的一个专门致力于真实新闻文本理解的例会。
  - 1991--1997
  - DARPA (Defense Advanced Research Projects Agency)
  - 组织对来自世界各地不同单位的消息理解系统进行系列化的评测活动。
- 主要的评测项目是从新闻报道中提取特定的信息，填入某种数据库中。
  - 评测语料大都出自各大通讯社发布的新闻。
  - 对每一条消息，由专业人员人工给出标准答案，然后将参测系统的输出结果与标准答案比较，按一定的评价指标给出所有系统的评测结果，其中最主要的指标是准确率、查全率等。
- MUC定义的概念、模型和技术规范对整个信息抽取领域起到了引领作用。

MUC (2)

Conference	Year	Text Source	Topic (Domain)
MUC-1	1987	Mil. reports	Fleet Operations
MUC-2	1989	Mil. reports	Fleet Operations
MUC-3	1991	News reports	Terrorist activities in Latin America
MUC-4	1992	News reports	Terrorist activities in Latin America
MUC-5	1993	News reports	Corporate Joint Ventures, Microelectronic production
MUC-6	1995	News reports	Negotiation of Labor Disputes and Corporate Management Succession
MUC-7	1997	News reports	Airplane crashes, and Rocket/Missile Launches

MUC (3)

- 5个典型的提取阶段：(MUC-7 IE Task Definition Version 5.1)
  - - NE (Named Entities)
  - - ER (Entity Relations)
  - - Template Scenario (Event Structures)
  - - Coreference (Identity descriptions)
  - - Template Merger
- 具体提取哪些 NE, ER, Events 以及做哪些Coref, Merger是任务相关的(每次MUC独立定义)。

MET

- MET: Multilingual Entity Task Evaluation
- 也是DARPA发起的一个测评项目。
- MET主要是对日语、汉语以及西班牙语等多语种新闻文献进行命名实体抽取。
- MET-1和MET-2测试分别于1996年和1998年进行。

ACE(1)

- ACE (Automatic Content Extraction)
  - A research program for developing advanced information extraction technologies convened by the NIST from 1999 to 2008。
- 关注三种信息的自动化内容抽取：
  - 网络上的在线新闻
  - 通过ASR（自动语音识别的）得到的广播新闻
  - 以及通过OCR（光学字符识别）得到的报纸新闻
- 两个目的：
  - 希望在自动化内容抽取基础之上，为数据挖掘、链接分析、自动摘要等打下基础
  - 通过将相应的信息提供给相应的分析师，以提高信息分析的能力。

ACE(2)

- 项目为期10年
  - ACE Phase-1(1999.7-2000.12)优先发展的是实体探测及追踪(EDT, Entity Detection and Tracking) 。
  - ACE Phase2(2001-2008) 被称为 EDT + RDC。 其中 RDC 为 Relation Detection and Characterization。
  - ACE第二阶段希望在第一阶段实体探测的基础之上，引入对实体关系的评测，需要能够将标识出的实体之间的关系揭示出来。

DUC

- Sponsored by the Advanced Research and Development Activity (ARDA), the conference series is run by the National Institute of Standards and Technology (NIST) to further progress in summarization and enable researchers to participate in large-scale experiments.
- 2000—2007
- In 2008, DUC became a Summarization track in the Text Analysis Conference (TAC)

TAC

- From 2008
- Grew out of NIST's Document Understanding Conference (DUC) and the Question Answering Track of TREC.
- A series of workshops that provides the infrastructure for large-scale evaluation of Natural Language Processing technology
- TAC's primary purpose is *not* competitive benchmarking; the emphasis is on advancing the state of the art through evaluation results

Goals of TAC

- □to promote research in NLP based on large common test collections;
- □to improve evaluation methodologies and measures for NLP;
- □to build a series of test collections that evolve to anticipate the evaluation needs of modern NLP systems;
- □to increase communication among industry, academia, and government by creating an open forum for the exchange of research ideas;
- □to speed the transfer of technology from research labs into commercial products by demonstrating substantial improvements in NLP methodologies on real-world problems.

TAC 2019

- Three Tracks
  - Entity Discovery and Linking (EDL)
    - The goal of the EDL track is to extract mentions of pre-defined entity types from any language, and link (disambiguate and ground) them to the entities in an English knowledge base (KB).
  - Streaming Multimedia Knowledge Base Population (SM-KBP)
    - The goal of the SM-KBP track is to develop and evaluate technologies that extract structured Knowledge Elements (KEs) from a variety of unstructured sources in order to generate explicit alternative interpretations of events, situations, and trends in noisy, conflicting, and potentially deceptive information environments.
  - Drug-Drug Interaction Extraction from Drug Labels (DDI)

https://tac.nist.gov/2019/index.html

TAC 2020

- Three Tracks
  - Epidemic Question Answering (EPIC-QA)
    - The goal of the EPIC-QA track is to evaluate systems on their ability to provide timely and well-supported answers to questions about the disease COVID-19, its causal virus SARS-CoV-2, related coronaviruses, and the recommended response to the pandemic. Because questions arise from both experts and non-experts in the field, EPIC-QA systems are challenged to return expert-level answers as expected by the scientific and medical communities as well as answers in consumer-friendly language for the general public.
  - Recognizing Ultra Fine-Grained Entities (RUFES)
    - The goal of the KBP RUFES track is to extract and corefer mentions of fine-grained entity types in text.
  - Streaming Multimedia Knowledge Base Population (SM-KBP)

https://tac.nist.gov/2020/index.html

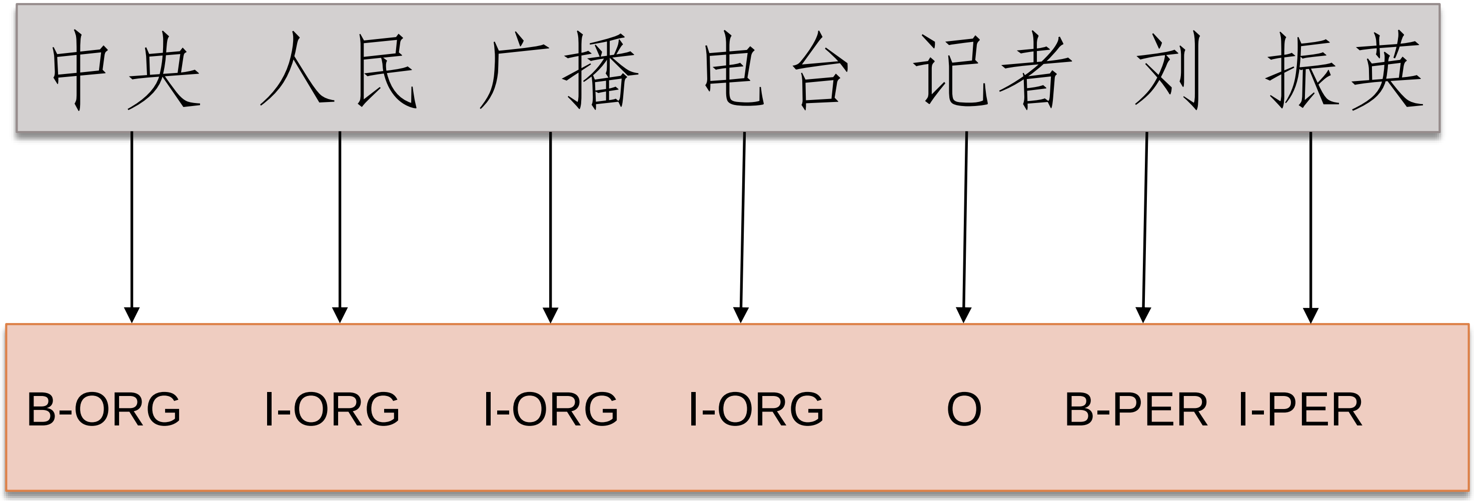
主要内容

- □信息抽取的定义、任务及发展
- □命名实体识别
- □实体链接
- □关系抽取

命名实体识别

- 定义
- 难点
- 应用
- 主要方法
- 评价

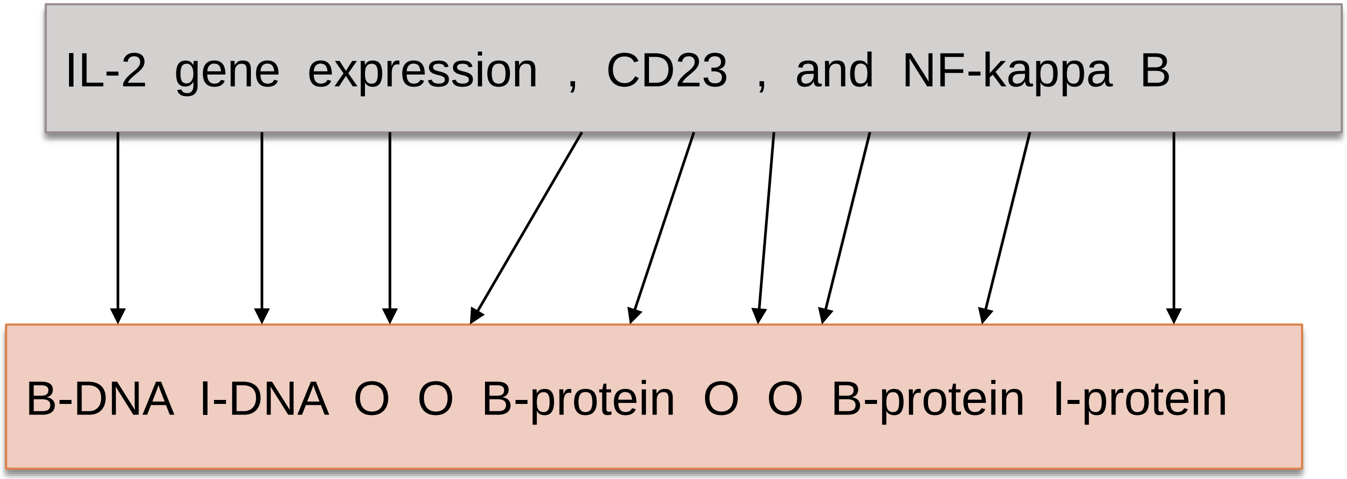
命名实体识别定义



中央 人民 广播 电台 记者 刘 振英

B-ORG I-ORG I-ORG I-ORG O B-PER I-PER

命名实体识别定义



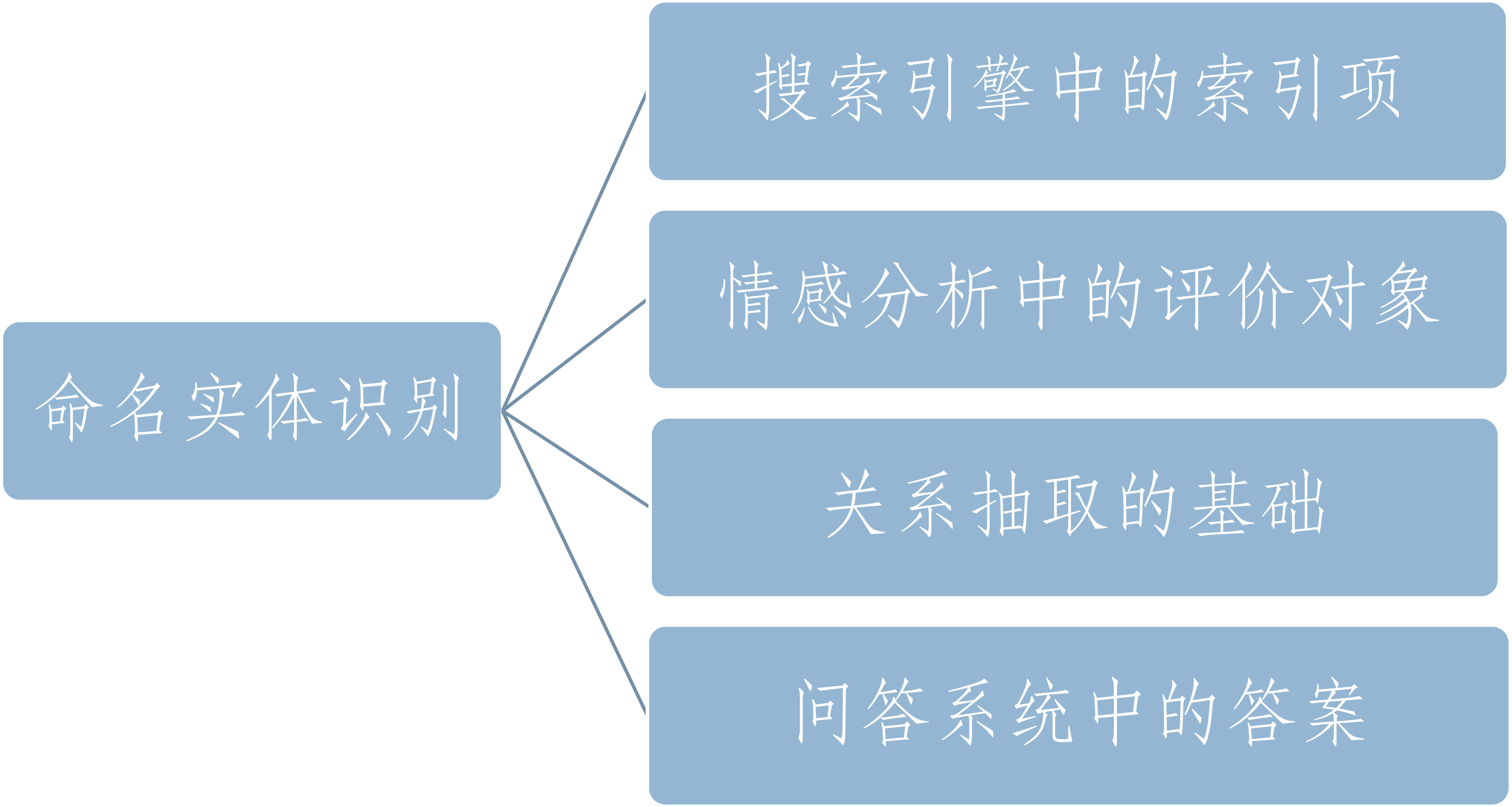
IL-2 gene expression , CD23 , and NF-kappa B

B-DNA I-DNA O O B-protein O O B-protein I-protein

命名实体识别的挑战

- 种类繁多，命名方式灵活多样
- 同一实体对应很多变体
- 相同的词或者短语可以表示不同类别的实体
- 存在嵌套
- 细粒度
- 语言不断进化，新的挑战不断出现

命名实体识别的应用



命名实体识别

搜索引擎中的索引项

情感分析中的评价对象

关系抽取的基础

问答系统中的答案

命名实体识别

- 主要方法
  - 基于规则的方法
  - 基于词典的方法
  - 机器学习方法
    - 最大熵
    - 条件随机场
    - 深度学习

命名实体识别

- 主要方法
  - 基于规则的方法
  - 基于词典的方法
  - 机器学习方法
    - 最大熵
    - 条件随机场
    - 深度学习

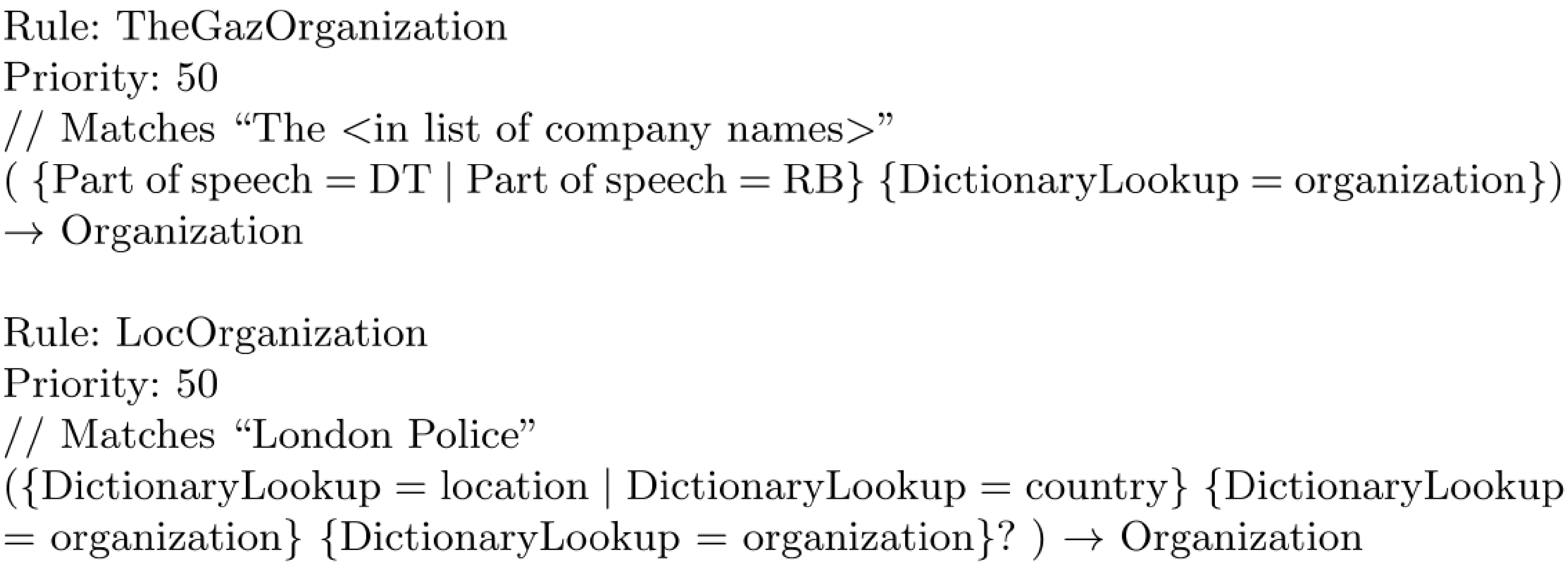
基于规则的方法

- Named entity recognition (NER) is particularly easy if it's possible to write a regular expression that captures the intended pattern of entities.
  - 例如：识别电子邮件地址的规则可以使用如下的正则表达式

```
EMAIL_REGEX
        = “[A-Za-z0-9](([_\\.\\-]?[a-zA-Z0-9]+)*)@([A-Za-z0-9]+)(([\\.\\-]?[a-zA-Z0-9]+)*)\\.([A-Za-z]{2,})”
```

- 有影响力的基于规则的NER系统
  - LingPipe （http://www.alias-i.com/lingpipe/)
  - GATE (http://gate.ac.uk/)

基于规则的方法

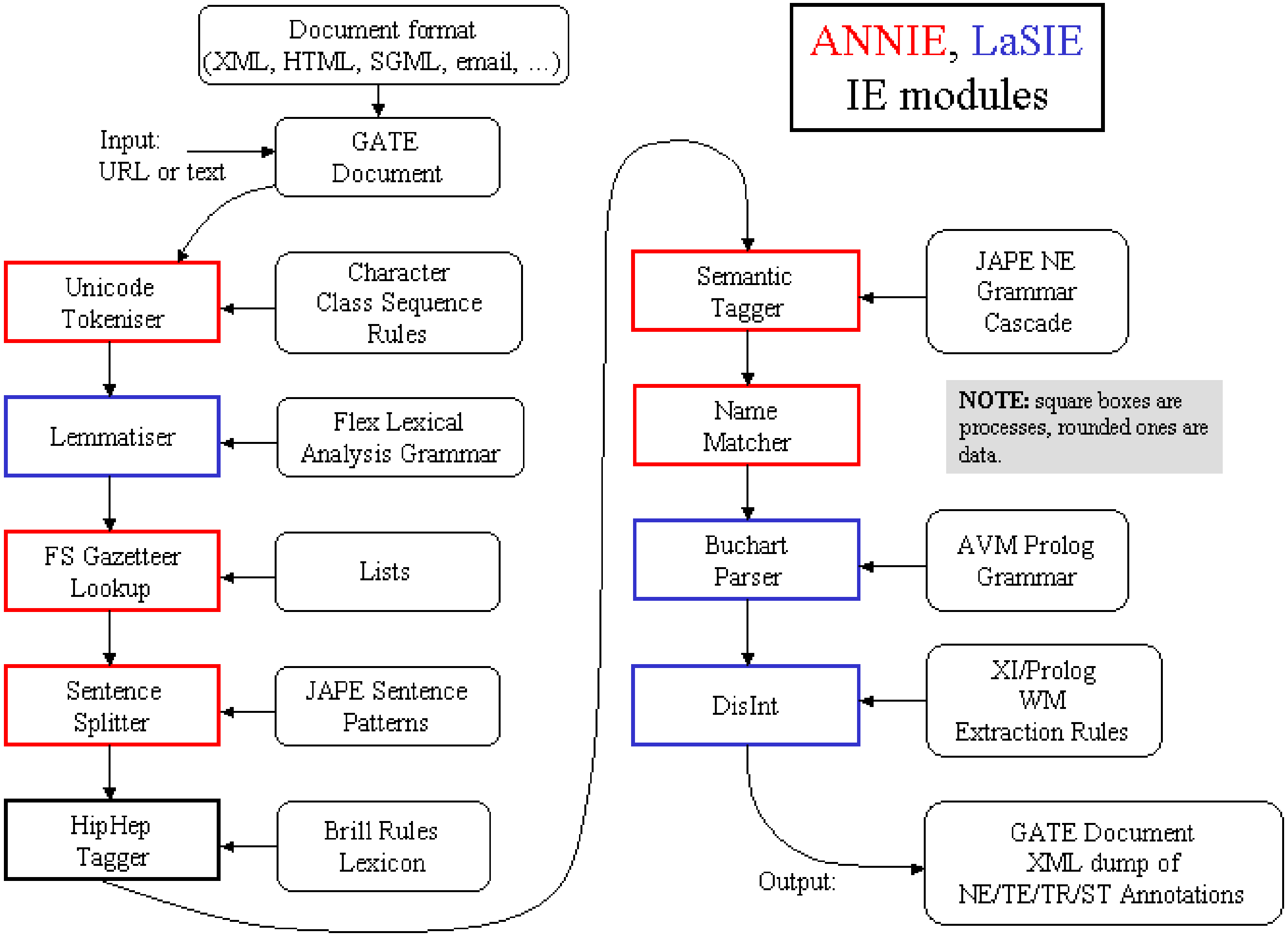


Rule: TheGazOrganization
Priority: 50
// Matches “The <in list of company names>”
({Part of speech = DT | Part of speech = RB} {DictionaryLookup = organization})
→ Organization

Rule: LocOrganization
Priority: 50
// Matches “London Police”
({DictionaryLookup = location | DictionaryLookup = country} {DictionaryLookup
= organization} {DictionaryLookup = organization}?) → Organization

- 规则来自GATE (General Arch. For Text Engineering), http://gate.ac.uk/
- 规则需要利用词性和词典

基于规则的方法



NOTE: square boxes are processes, rounded ones are data.

图片来源：https://gate.ac.uk/sale/tao/splitch6.html#chap:annie

基于词典的方法

- 对于某些应用，基于词典的NER方法直接有效
  - 词典易获取，且完备性好
    - 比如 westlaw.com网站有美国所有注册律师的名单；mlb.com有所有现役和退役棒球运动员的名单。
- 主要关注的问题
  - 词典的构建和更新
  - 词典查找的效率
    - LingPipe提供了一个Aho-Corasick算法的实现，它可以在线性时间内根据字典找到所有匹配项，时间复杂度不依赖于字典的大小。
- 经常与其他NER方法组合使用

命名实体识别

- 主要方法
  - 基于规则的方法
  - 基于词典的方法
  - 机器学习方法
    - 最大熵
    - 条件随机场
    - 深度学习

最大熵模型

- Maximum Entropy (Maxent) Classifier
  - Maximum Entropy was first introduced to NLP area by Berger, et al (1996) and Della Pietra, et al. 1997.

$$\mathrm{P}(y|x) = \frac{1}{Z(x)} exp \sum_{i=1}^{\mathrm{N}} \lambda_i f_i(x,y)$$

$$Z(x) = \sum_{y=1}^{C} exp \sum_{i=1}^{\mathrm{N}} \lambda_i f_i(x,y)$$

Classify result

$$y^* = \underset{y \in \{1,2,\ldots,C\}}{\operatorname{argmax}} P(y|x)$$

最大熵模型

- 我们日常生活中经常会运用最大熵模型
- 一个质地均匀的色子，每个面朝上的概率分别是多少?

1/6

- 如果色子被特殊处理过，四点朝上的概率是三分之一，在这种情况下，每个面朝上的概率是多少?

2/15

最大熵模型

- □最大熵原理指出，当我们需要对一个随机事件的概率分布进行预测时，我们的预测应当满足全部已知的条件，而对未知的情况不要做任何主观假设。（不做主观假设这点很重要。）在这种情况下，概率分布最均匀，预测的风险最小。因为这时概率分布的信息熵最大，所以人们称这种模型叫"最大熵模型"。
- □匈牙利著名数学家、信息论最高奖香农奖得主希萨（Csiszar）证明，对任何一组不自相矛盾的信息，这个最大熵模型不仅存在，而且是唯一的。而且它们都有同一个非常简单的形式 -- 指数函数。

--吴军，“数学之美”，177-183页

最大熵模型—指数函数形式是如何得来的?

- 训练数据　(x_1, y_1)，$(x_2, y_2), \cdots$，(x_N, y_N)
- 预测任务 $p(y|x)$
- 特征 $f_i(x, y)$　$i \in (1,2, \cdots, N)$
- 特征$f(x, y)$的经验期望

$$\tilde{p}(f) = \sum_{x,y} \tilde{p}(x, y) f(x, y)$$

- 特征$f(x, y)$的关于$p(y|x)$的模型期望

$$p(f) = \sum_{x,y} \tilde{p}(x) p(y|x) f(x, y)$$

- 来自训练数据的约束：经验期望=模型期望

$$\tilde{p}(f) = p(f)$$

最大熵模型—指数函数形式是如何得来的?

- 熵的定义

$$\mathrm{H}(x) = -\sum_{x} p(x) \log[p(x)]$$

- 条件熵

$$H\big(p(y|x)\big) = -\sum_{x,y} \tilde{p}(x) p(y|x) \log p(y|x)$$

- 寻找最大熵（条件熵）

$$p^{*} = \underset{p(y|x)}{\operatorname{argmax}} H\big(p(y|x)\big)$$

- 同时满足下列约束条件

$$p(\mathrm{y}|\mathrm{x}) \geq 0, \text{对于任意的} x, y$$

$$\sum_{y} p(\mathrm{y}|\mathrm{x}) = 1, \text{对于任意的} x$$

$$\tilde{p}(f) = p(f), \text{对于每一个} f$$

最大熵模型—指数函数形式是如何得来的?

□根据优化目标和约束，使用拉格朗日方法

$$\varepsilon(p,\Lambda,\gamma)\mathrm{H}(x) = -\sum_{x,y} \tilde{p}(x)p(y|x)\log p(y|x)$$

$$+\sum_{i} \lambda_i(\sum_{x,y} \tilde{p}(x,y)f_i(x,y) - \tilde{p}(x)p(y|x)f_i(x,y))$$

$$+\gamma(\sum_{y} p(\mathrm{y}|\mathrm{x}) - 1)$$

$$\Lambda = \{\lambda_1, \lambda_2, \cdots, \lambda_N\}$$

$$\frac{\partial\varepsilon}{\partial p(y|x)} = -\tilde{p}(x)(\log p(y|x) + 1) - \sum_{i} \lambda_i\big(\tilde{p}(x)f_i(x,y)\big) + \gamma$$

令$\frac{\partial\varepsilon}{\partial p(y|x)} = 0$，可得$p^*(y|x) = \exp\big(\lambda_i f_i(x,y)\big)\exp(-\frac{\gamma}{\tilde{p}(x)} - 1)$

最大熵模型—指数函数形式是如何得来的?

- 根据归一化约束，可以把

$$p^*(y|x) = \exp\left(\lambda_i f_i(x,y)\right) \exp\left(-\frac{\gamma}{\tilde{p}(x)} - 1\right)$$ 改写成

$$\mathrm{P}(y|x) = \frac{1}{Z(x)} exp \sum_{i=1}^{K} \lambda_i f_i(x,y)$$

$$Z(x) = \sum_{y=1}^{C} exp \sum_{i=1}^{\mathrm{N}} \lambda_i f_i(x,y)$$

最大熵模型需要训练的参数

- Given this model form, we will choose parameters $\{\lambda_i\}$ that *maximize the conditional likelihood* of the training data according to this model.

- We construct not only classifications, but probability distributions over classifications.

基于指数函数的最大似然估计

- Maximum (Conditional) Likelihood Models :
  - Given a model form, choose values of parameters to maximize the (conditional) likelihood of the data.

$$\log P\,(C|D,\lambda) = \log \prod_{(c,d)\in(C,D)} P(c|d,\lambda)$$

$$= \sum_{(c,d)\in(C,D)} \log \frac{\exp \sum_i \lambda_i f_i(c,d)}{\sum_{c'} \exp \sum_i \lambda_i f_i(c',d)}$$

基于指数函数的最大似然估计

- The (log) conditional likelihood of iid data (*C*,*D*) according to maxent model is a function of the data and the parameters λ:

$$\log P(C|D,\lambda) = \log \prod_{(c,d)\in(C,D)} P(c|d,\lambda) = \sum_{(c,d)\in(C,D)} \log P(c|d,\lambda)$$

- If there aren’ t many values of *c*, it’ s easy to calculate:

$$\log P(C|D,\lambda) = \sum_{(c,d)\in(C,D)} \log \frac{\exp \sum_i \lambda_i f_i(c,d)}{\sum_{c'} \exp \sum_i \lambda_i f_i(c',d)}$$

基于指数函数的最大似然估计

- 似然度的计算
- We can separate this into two components:

$$\log P(C|D,\lambda) = \sum_{(c,d)\in(C,D)} \log\exp\sum_{i}\lambda_i f_i(c,d) - \sum_{(c,d)\in(C,D)} \log\sum_{c'}\exp\sum_{i}\lambda_i f_i(c',d)$$

$$\log P(C|D,\lambda) = N(\lambda) - M(\lambda)$$

- The derivative is the difference between the derivatives of each component

基于指数函数的最大似然估计

$$\frac{\partial N(\lambda)}{\partial \lambda_i} = \frac{\partial \sum_{(c,d)\in(C,D)} \log \exp \sum_i \lambda_i f_i(c,d)}{\partial \lambda_i}$$

$$= \frac{\partial \sum_{(c,d)\in(C,D)} \sum_i \lambda_i f_i(c,d)}{\partial \lambda_i}$$

$$= \sum_{(c,d)\in(C,D)} \frac{\partial \sum_i \lambda_i f_i(c,d)}{\partial \lambda_i} \qquad = \sum_{(c,d)\in(C,D)} f_i(c,d)$$

Derivative of the numerator is: the empirical count (f_i , c)

基于指数函数的最大似然估计

$$\frac{\partial M(\lambda)}{\partial \lambda_i} = \frac{\partial \sum_{(c,d)\in(C,D)} \log \sum_{c'} \exp \sum_i \lambda_i f_i(c',d)}{\partial \lambda_i}$$

$$= \sum_{(c,d)\in(C,D)} \frac{1}{\sum_{c''} \exp \sum_i \lambda_i f_i(c'',d)} \frac{\partial \sum_{c'} \exp \sum_i \lambda_i f_i(c',d)}{\partial \lambda_i}$$

$$= \sum_{(c,d)\in(C,D)} \frac{1}{\sum_{c''} \exp \sum_i \lambda_i f_i(c'',d)} \sum_{c'} \frac{\exp \sum_i \lambda_i f_i(c',d)}{1} \frac{\partial \sum_i \lambda_i f_i(c',d)}{\partial \lambda_i}$$

$$= \sum_{(c,d)\in(C,D)} \sum_{c'} \frac{\exp \sum_i \lambda_i f_i(c',d)}{\sum_{c''} \exp \sum_i \lambda_i f_i(c'',d)} \frac{\partial \sum_i \lambda_i f_i(c',d)}{\partial \lambda_i}$$

$$= \sum_{(c,d)\in(C,D)} \sum_{c'} P(c'|d,\lambda)\, f_i(c',d)$$

= predicted count (f_i,λ)

基于指数函数的最大似然估计

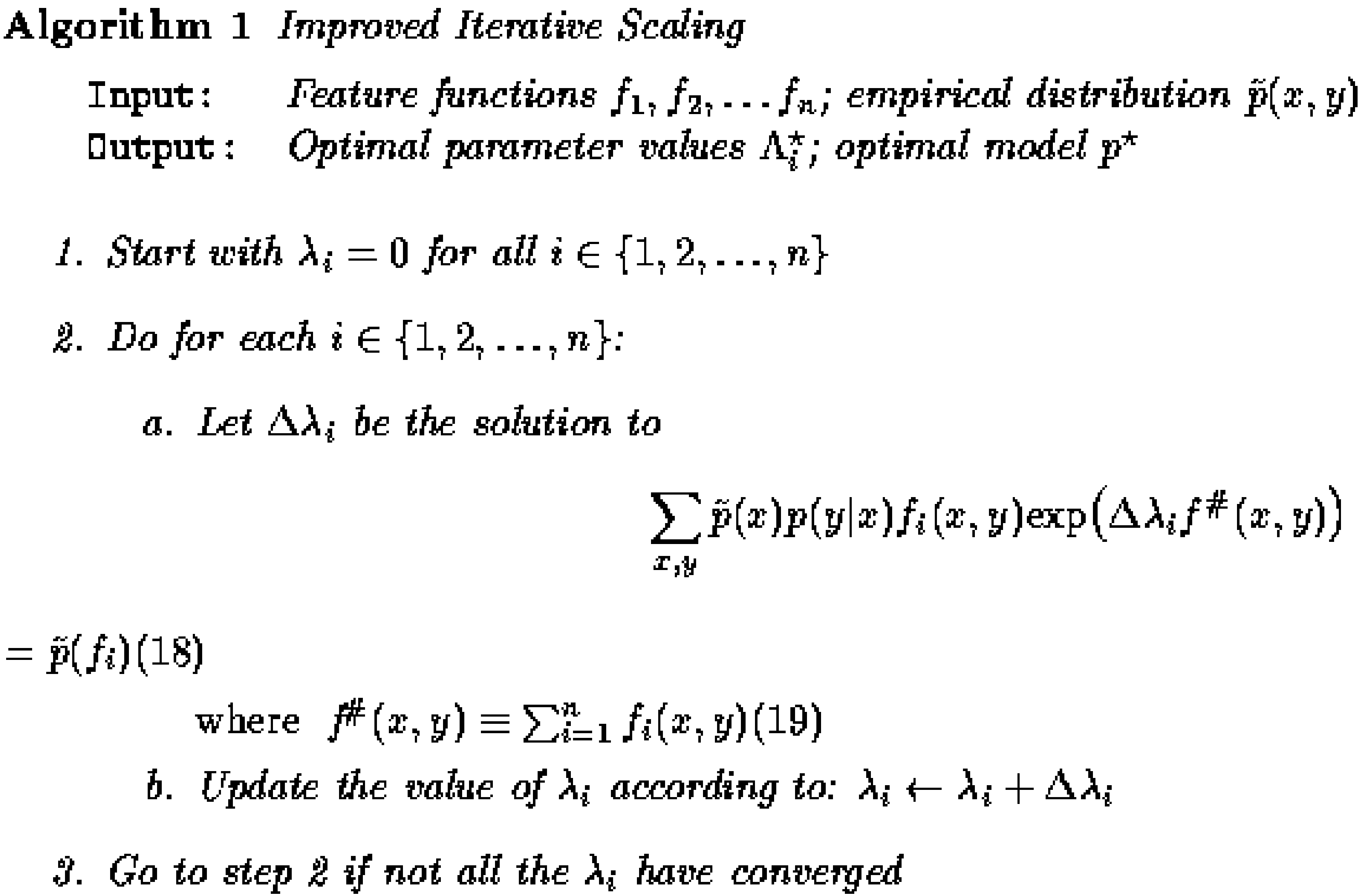
$$\frac{\partial \log P\left(C|D,\lambda\right)}{\partial \lambda_i} = \text{actual count}(f_i, C) - \text{actual count}(f_i, C)$$

$$E_p(f_j) = E_{\tilde{p}}(f_j), \forall j$$

- The optimum parameters are the ones for which each feature’ s predicted expectation equals its empirical expectation. The optimum distribution is:
  - Always unique (but parameters may not be unique)
  - Always exists (if feature counts are from actual data).

参数优化方法

- Limited-memory BFGS Method
  - Most effective [Mal 02]
  - Popular in late 1990s
- Iterative Scaling Methods
  - Improved Iterative Scaling
  - Generalized Iterative Scaling
  - Correction-Free GIS (SCGIS)
    - A faster GIS variant [Goodman 02] [Curran 03]
    - Needn’t the constant C in GIS



Algorithm 1 *Improved Iterative Scaling*

```
Input:   Feature functions $f_1, f_2, \ldots f_n$; empirical distribution $\tilde{p}(x, y)$
Output:  Optimal parameter values $\Lambda_i^\star$; optimal model $p^\star$

1. Start with $\lambda_i = 0$ for all $i \in \{1, 2, \ldots, n\}$
2. Do for each $i \in \{1, 2, \ldots, n\}$:
    a. Let $\Delta\lambda_i$ be the solution to
           $\sum_{x,y} \tilde{p}(x) p(y|x) f_i(x, y) \exp\big(\Delta\lambda_i f^{\#}(x, y)\big)$
       $= \tilde{p}(f_i) (18)$
           where $f^{\#}(x, y) \equiv \sum_{i=1}^{n} f_i(x, y) (19)$
    b. Update the value of $\lambda_i$ according to: $\lambda_i \leftarrow \lambda_i + \Delta\lambda_i$
3. Go to step 2 if not all the $\lambda_i$ have converged
```

逻辑回归与最大熵模型

- Maxent models in NLP are essentially the same as multiclass logistic regression models
  - The parameterization is slightly different in a way that is advantageous for NLP-style models with tons of sparse features
  - The key role of feature functions in NLP

最大熵模型的应用

- 最大熵模型是一种分类模型，适用于很多自然语言处理任务:
  - Sentence boundary detection (Mikheev 2000)
    - Is a period end of sentence or abbreviation?
  - Sentiment analysis (Pang and Lee 2002)
    - Word unigrams, bigrams, POS counts, …
  - PP attachment (Ratnaparkhi 1998)
    - Attach to verb or noun? Features of head noun, preposition, etc.
  - Parsing decisions in general (Ratnaparkhi 1997; Johnson et al. 1999, etc.)

最大熵模型中的特征

- *features f* are elementary pieces of evidence that link aspects of what we observe *d* with a category *c* that we want to predict
- A feature is a function with a bounded real value
- Models will assign to each feature a *weight:*
  - A positive weight votes that this configuration is likely correct
  - A negative weight votes that this configuration is likely incorrect

特征举例

LOCATION	LOCATION	DRUG	PERSON
in Arcadia	*in Québec*	*taking Zantac*	*saw Sue*

- $f_1(c, d) \equiv$

$[c = \text{LOCATION} \wedge w_{-1} = \text{“in”} \wedge \text{isCapitalized}(w)]$

- $f_2(c, d) \equiv$

$[c = \text{LOCATION} \wedge \text{hasAccentedLatinChar}(w)]$

- $f_3(c, d) \equiv$

$[c = \text{DRUG} \wedge \text{ends}(w, \text{“c”})]$

如何利用特征进行分类?

- Make a probabilistic model from the linear combination $\sum_i \lambda_i f_i(c,d)$

$$P(c|d,\lambda) = \frac{\exp \sum_i \lambda_i f_i(c,d)}{\sum_{c'} \exp \sum_i \lambda_i f_i(c',d)}$$

Makes votes positive

Normalizes votes

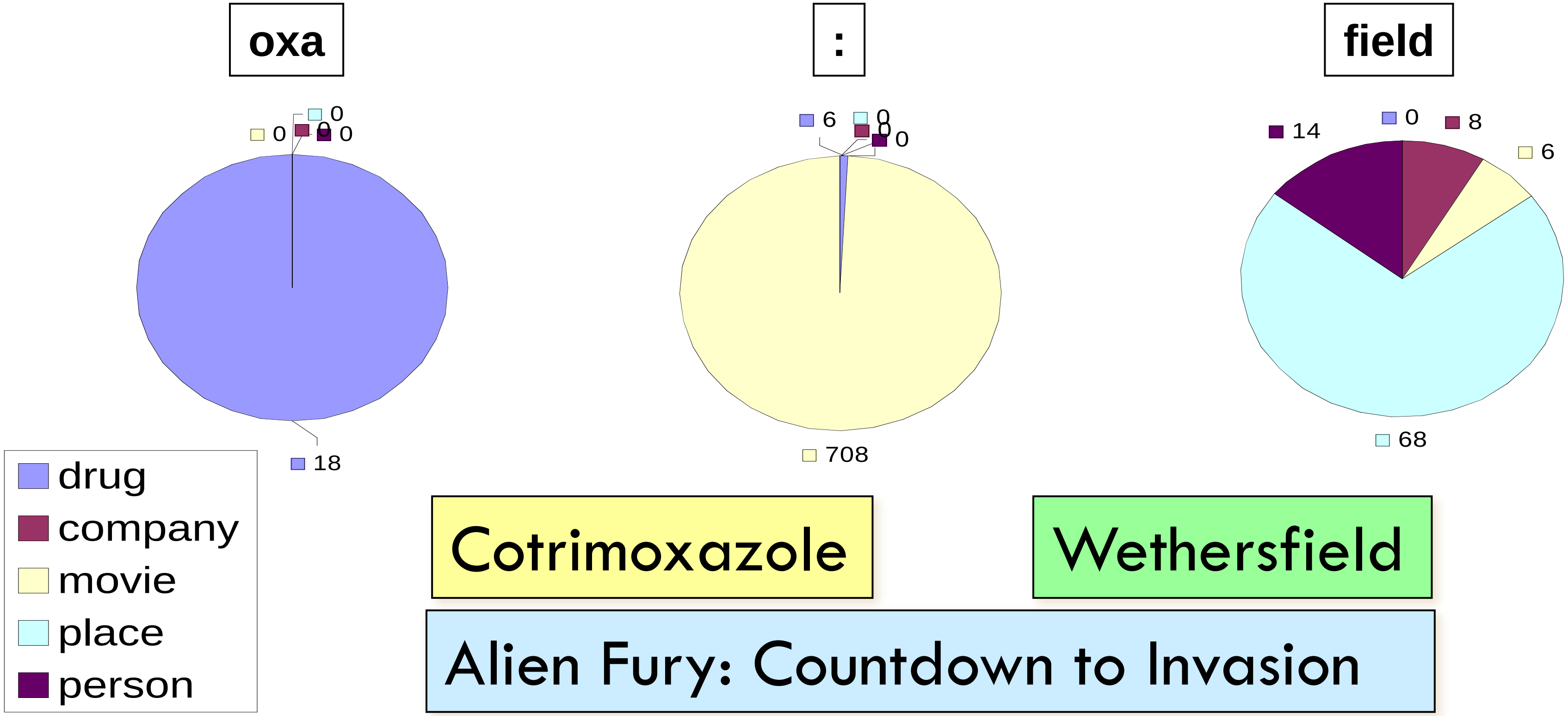
  - P(LOCATION|*in Québec*) = $e^{1.8}e^{-0.6}/(e^{1.8}e^{-0.6} + e^{0.3} + e^{0}) = 0.586$
  - P(DRUG|*in Québec*) = $e^{0.3}/(e^{1.8}e^{-0.6} + e^{0.3} + e^{0}) = 0.238$
  - P(PERSON|*in Québec*) = $e^{0}/(e^{1.8}e^{-0.6} + e^{0.3} + e^{0}) = 0.176$

- The weights are the parameters of the model, combined via a “soft max” function

命名实体识别常用的特征类型

- 词特征
  - 当前词(essentially like a learned dictionary)
  - 前一个词/后一个词（上下文特征）
  - 词缀（子串）特征
  - 词形特征
- 其他类型的语言分类信息
  - 词性（POS）标记
  - 组块（Chunk）标记
- 实体标记的上下文
  - 前一个词的实体标记

词缀（子串）特征详解



Cotrimoxazole

Wethersfield

Alien Fury: Countdown to Invasion

词缀（子串）特征详解

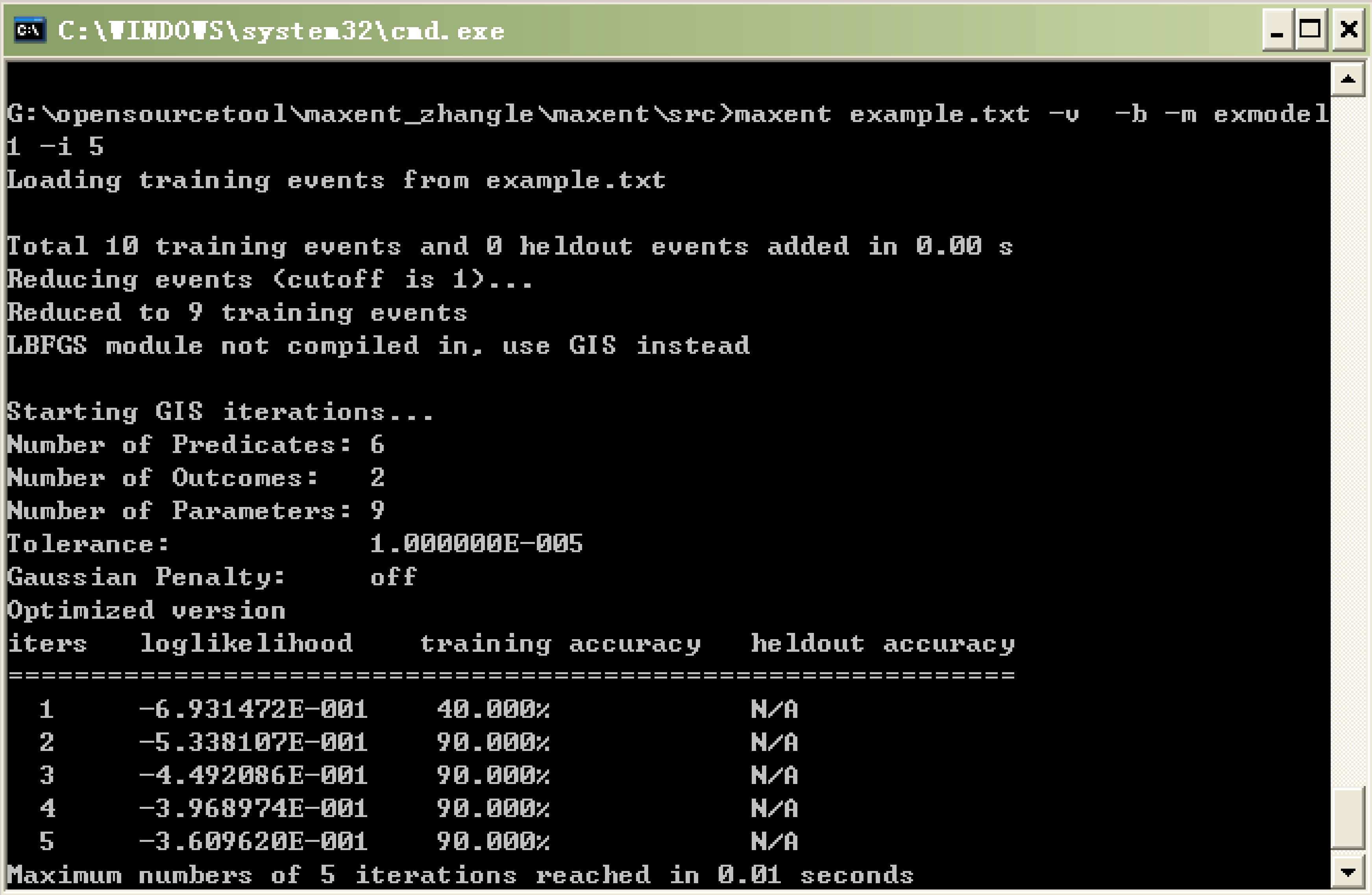
- 词形特征
  - Map words to simplified representation that encodes attributes such as length, capitalization, numerals, Greek letters, internal punctuation, etc.

Varicella-zoster	Xx-xxx
mRNA	xXXX
CPA1	XXXd

最大熵模型使用举例

- Example data
  1. No_Umbrella Warm Dry
  2. No_Umbrella Cold Dry
  3. Umbrella Cold Rainy
  4. Umbrella Cold Dry
  5. No_Umbrella Warm Dry
  6. Umbrella Cold Dry Early
  7. Umbrella Cold Rainy Early
  8. No_Umbrella Cold Dry Late
  9. No_Umbrella Warm Rainy Late
  10. No_Umbrella Warm Dry Late

使用Zhangle的工具包进行训练



```
G:\opensourcetool\maxent_zhangle\maxent\src>maxent example.txt -v  -b -m exmodel
1 -i 5
Loading training events from example.txt

Total 10 training events and 0 heldout events added in 0.00 s
Reducing events (cutoff is 1)...
Reduced to 9 training events
LBFGS module not compiled in, use GIS instead

Starting GIS iterations...
Number of Predicates: 6
Number of Outcomes:   2
Number of Parameters: 9
Tolerance:            1.000000E-005
Gaussian Penalty:     off
Optimized version
iters   loglikelihood    training accuracy   heldout accuracy
=============================================================
  1     -6.931472E-001    40.000%             N/A
  2     -5.338107E-001    90.000%             N/A
  3     -4.492086E-001    90.000%             N/A
  4     -3.968974E-001    90.000%             N/A
  5     -3.609620E-001    90.000%             N/A
Maximum numbers of 5 iterations reached in 0.01 seconds
```

输出的模型长什么样子?

- Predicate(6):
  - Warm, Dry, Cold, Rainy, Early， Late
- Outcomes(2):
  - No_Umbrella, Umbrella
- Parameters(9):
  - 1 0 #warm only for label 0(No_Umbrella)
  - 2 0 1 #Dry both for label 0 and label 1
  - 2 0 1 #Cold both for label 0 and label 1
  - 2 0 1 #Rainy both for label 0 and label 1
  - 1 1 #Early only for label 1
  - 1 0 #Late only for label 0

```
#txt,maxent
6
Warm
Dry
Cold
Rainy
Early
Late
2
No_Umbrella
Umbrella
1 0  #warm only for label 0(No_Umbrella)
2 0 1 #Dry both for label 0 and label 1
2 0 1 #Cold both for label 0 and label 1
2 0 1 #Rainy both for label 0 and label 1
1 1 #Early only for label 1
1 0 #Late only for label 0
9
0.67742682914650987
0.31066551494595002
-0.56536526381309366
-0.48696502571145761
0.31624489731139394
-0.31190677177012249
0.19747190915845117
0.74127573146151871
0.74141828027072216
```

命名实体识别

- 主要方法
  - 基于规则的方法
  - 基于词典的方法
  - 机器学习方法
    - 最大熵
    - 条件随机场
    - 深度学习

条件随机场模型（CRFs）

- Conditional Random Fields (CRFs) 是一种序列标注模型
  - A whole-sequence conditional model rather than a chaining of local models.

$$P(c \mid d, \lambda) = \frac{\exp \sum_i \lambda_i f_i(c,d)}{\sum_{c'} \exp \sum_i \lambda_i f_i(c',d)}$$

- The space of *c*′ s is now the space of sequences
  - But if the features f_i remain local, the conditional sequence likelihood can be calculated exactly using dynamic programming
- Training is slower, but CRFs avoid causal-competition biases

CRFs的参数形式

$$score(\mathbf{l}|s) = \sum_{j=1}^{m}\sum_{i=1}^{n} w_j f_j(s, i, l_i, l_{i-1})$$

$$p(\mathbf{l}|s) = \frac{exp[score(\mathbf{l}|s)]}{\sum_{\mathbf{l}'\in\mathbf{L}} exp[score(\mathbf{l}'|s)]}$$

$$= \frac{exp[\sum_{j=1}^{m}\sum_{i=1}^{n} w_j f_j(s, i, l_i, l_{i-1})]}{\sum_{\mathbf{l}'\in\mathbf{L}} exp[\sum_{j=1}^{m}\sum_{i=1}^{n} w_j f_j(s, i, l'_i, l'_{i-1})]}$$

Learning Weights for CRF

$$\frac{\partial \log p(\mathbf{l}|s)}{\partial w_j} = \sum_{i=1}^{n} f_j(s, i, l_i, l_{i-1}) - \sum_{\mathbf{l}' \in \mathbf{L}} p(\mathbf{l}'|s) \sum_{i=1}^{n} f_j(s, i, l'_i, l'_{i-1})$$

$$w_j = w_j + \alpha \times \frac{\partial \log p(\mathbf{l}|s)}{\partial w_j}$$

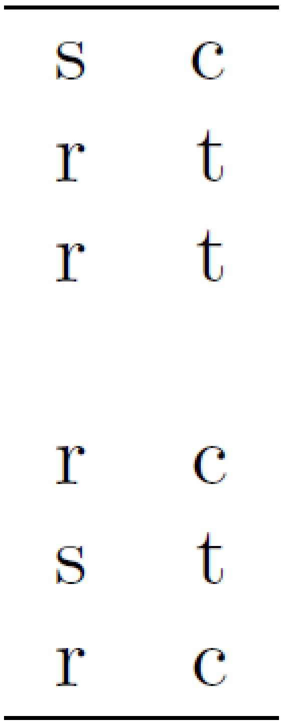
使用CRF的一个简单例子

- Lee loves tea (t) and coffee (c). However, he is only allowed to drink one cup of tea (t) or coffee (c) per day, on Monday, Tuesday and Wednesday.
- Assume Lee's drinking behavior on a particular day is affected by many factors, including the weather (sunny or rainy) and what he drinks on the previous day.

使用CRF的一个简单例子

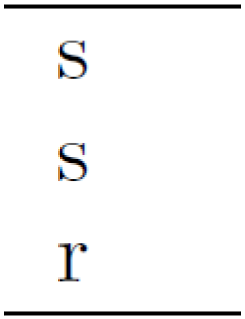
- In week one, the weather for Monday, Tuesday and Wednesday is [sunny (s), rainy (r), rainy (r)], and Lee's drinking behavior is [coffee (c), tea (t), tea (t)].
- In week two, the weather is [r, s, r] and Lee drinks [c, t, c].
- In week three, the weather forecast is [s, s, r], what's Lee's drinking behavior?

使用CRF的一个简单例子



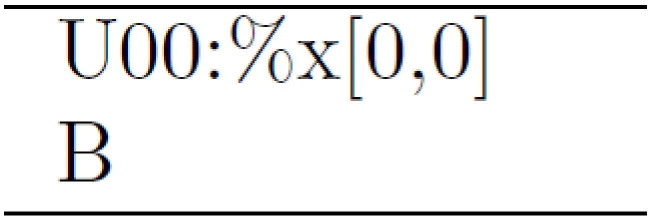
```
s   c
r   t
r   t

r   c
s   t
r   c
```

Training file



```
s
s
r
```

Test file

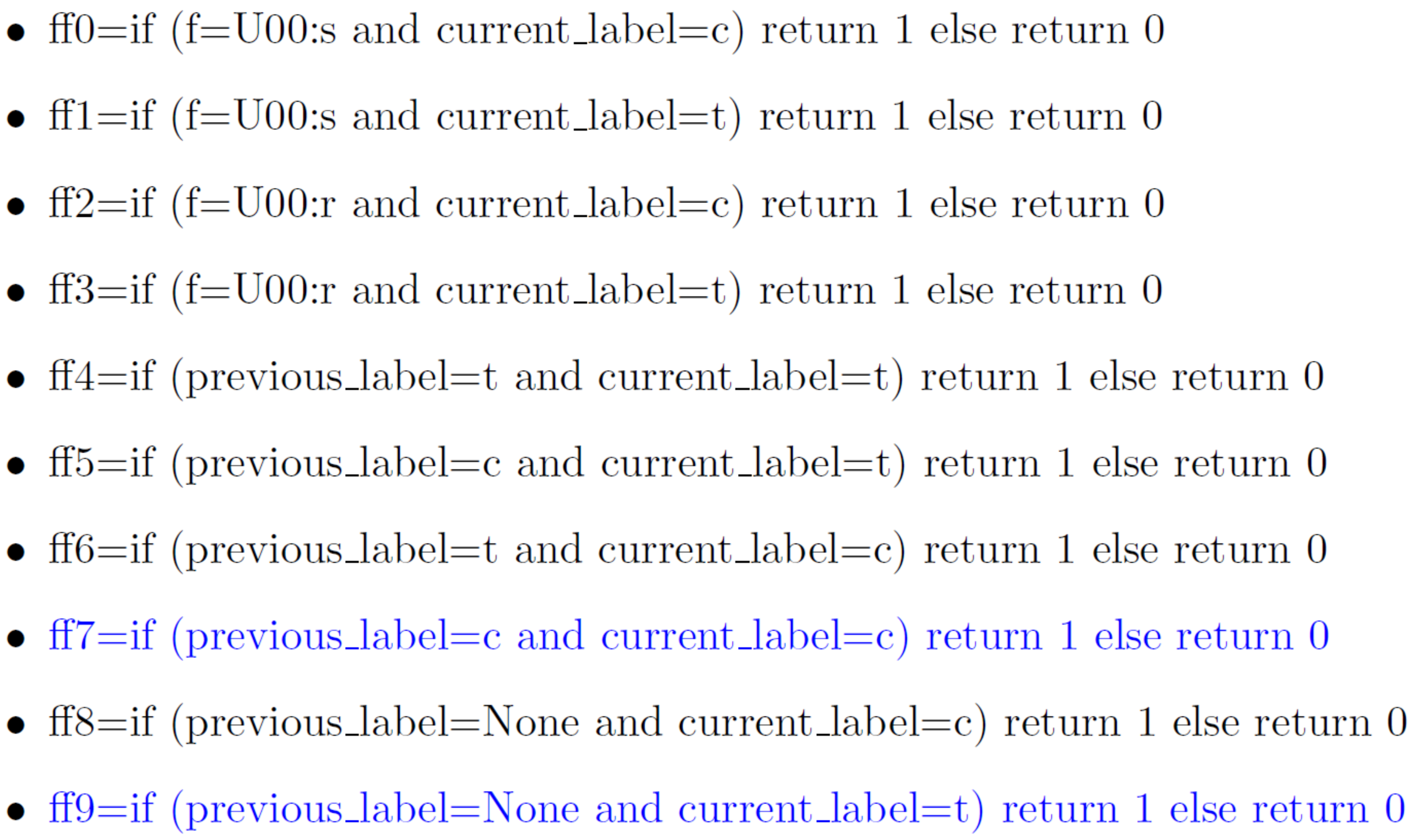


```
U00:%x[0,0]
B
```

Template file

使用CRF的一个简单例子

Features generated with training file and templates



- ff0=if (f=U00:s and current_label=c) return 1 else return 0
- ff1=if (f=U00:s and current_label=t) return 1 else return 0
- ff2=if (f=U00:r and current_label=c) return 1 else return 0
- ff3=if (f=U00:r and current_label=t) return 1 else return 0
- ff4=if (previous_label=t and current_label=t) return 1 else return 0
- ff5=if (previous_label=c and current_label=t) return 1 else return 0
- ff6=if (previous_label=t and current_label=c) return 1 else return 0
- ff7=if (previous_label=c and current_label=c) return 1 else return 0
- ff8=if (previous_label=None and current_label=c) return 1 else return 0
- ff9=if (previous_label=None and current_label=t) return 1 else return 0

使用CRF的一个简单例子

Learning weights

s	c	c	c	c	t	t	t	t
r	c	c	t	t	c	c	t	t
r	c	t	t	c	c	t	t	c

$$score(\mathbf{l}|s) = \sum_{j=1}^{m} \sum_{i=1}^{n} w_j f_j(s, i, l_i, l_{i-1})$$

$$score(\mathbf{l_0}|s) = (w_0 * f_0 + w_8 * f_8) + (w_2 * f_2 + w_7 * f_7) + (w_2 * f_2 + w_7 * f_7)$$
$$= (1 * 1 + 1 * 1) + (1 * 1 + 1 * 1) + (1 * 1 + 1 * 1) = 6.0$$

f0: s, c

f1: s, t

f2: r, c

f3: r, t

f4: t, t

f5: c, t

f6: t, c

f7: c, c

f8: None, c

f9: None, t

使用CRF的一个简单例子

Learning weights

$$score(\mathbf{l}|s) = \sum_{j=1}^{m} \sum_{i=1}^{n} w_j f_j(s, i, l_i, l_{i-1})$$

$$p(\mathbf{l}|s) = \frac{exp[score(\mathbf{l}|s)]}{\sum_{\mathbf{l'} \in \mathbf{L}} exp[score(\mathbf{l'}|s)]} = \frac{exp[\sum_{j=1}^{m} \sum_{i=1}^{n} w_j f_j(s, i, l_i, l_{i-1})]}{\sum_{\mathbf{l'} \in \mathbf{L}} exp[\sum_{j=1}^{m} \sum_{i=1}^{n} w_j f_j(s, i, l'_i, l'_{i-1})]}$$

$$w_j = w_j + \alpha \times \frac{\partial \log p(\mathbf{l}|s)}{\partial w_j}$$

$$w_0 = w_0 + 0.01 \times 0.5 = 1 + 0.01 \times 0.5 = 1.005$$

$$w_1 = w_1 + 0.01 \times -0.5 = 1 + 0.01 \times -0.5 = 0.995$$

$$w_2 = w_2 + 0.01 \times -1.0 = 1 + 0.01 \times -1.0 = 0.99$$

$$w_3 = w_3 + 0.01 \times 1.0 = 1 + 0.01 \times 1.0 = 1.01$$

f0: s, c

f1: s, t

f2: r, c

f3: r, t

f4: t, t

f5: c, t

f6: t, c

f7: c, c

f8: None, c

f9: None, t

使用CRF的一个简单例子

Learning weights

$$w_0 = w_0 + 0.01 \times 0.5 = 1 + 0.01 \times 0.5 = 1.005$$

$$w_1 = w_1 + 0.01 \times -0.5 = 1 + 0.01 \times -0.5 = 0.995$$

$$w_2 = w_2 + 0.01 \times -1.0 = 1 + 0.01 \times -1.0 = 0.99$$

$$w_3 = w_3 + 0.01 \times 1.0 = 1 + 0.01 \times 1.0 = 1.01$$

Before changing the weights:

$$\log p(\mathbf{l}_{\mathbf{trn_0}}|trn_0) + \log p(\mathbf{l}_{\mathbf{trn_1}}|trn_1) = \log 0.125 + \log 0.125 = -4.159$$

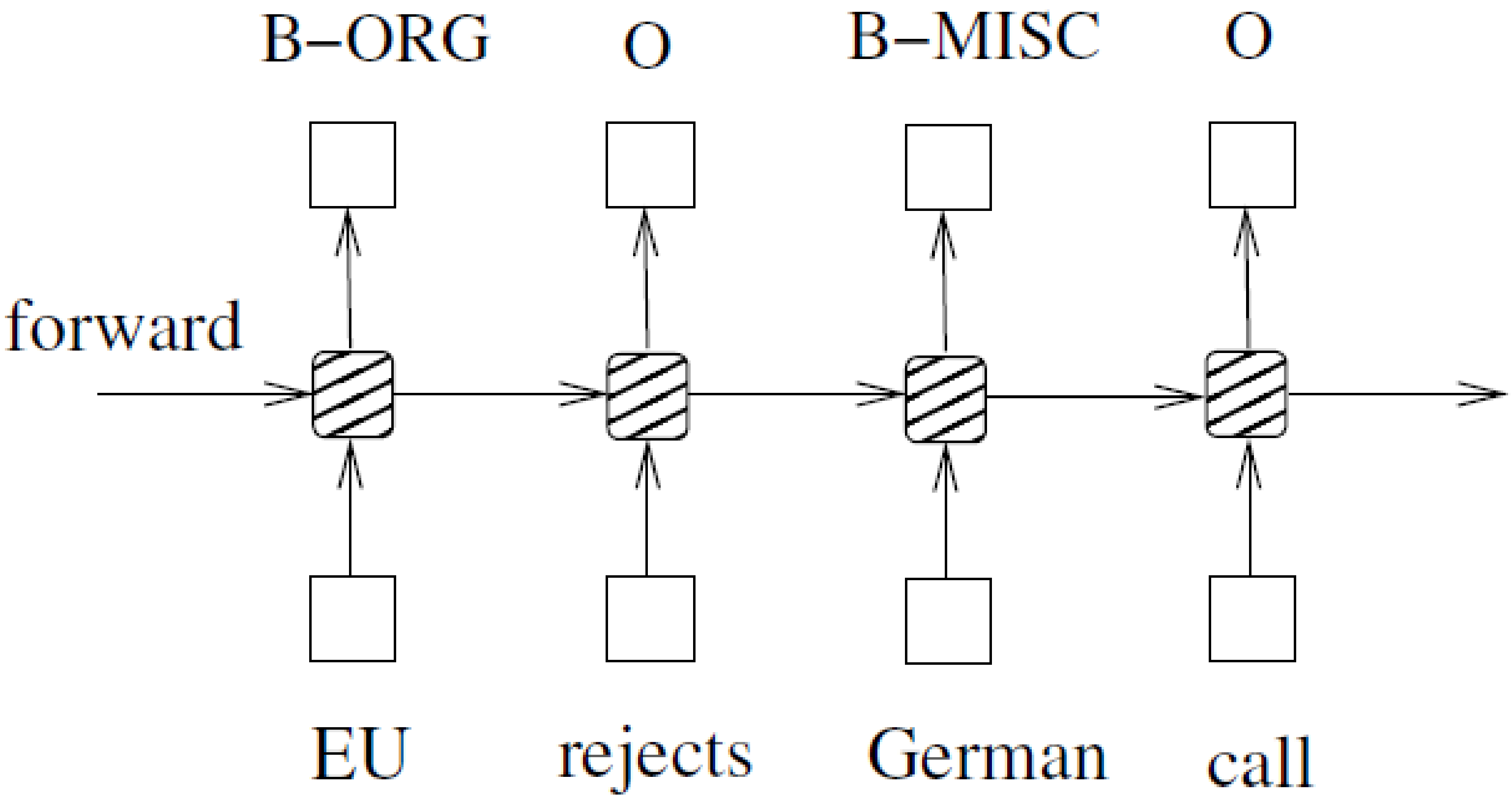
After changing the weights:

$$\log p(\mathbf{l}_{\mathbf{trn_0}}|trn_0) + \log p(\mathbf{l}_{\mathbf{trn_1}}|trn_1) = \log 0.130 + \log 0.123 = -4.139$$

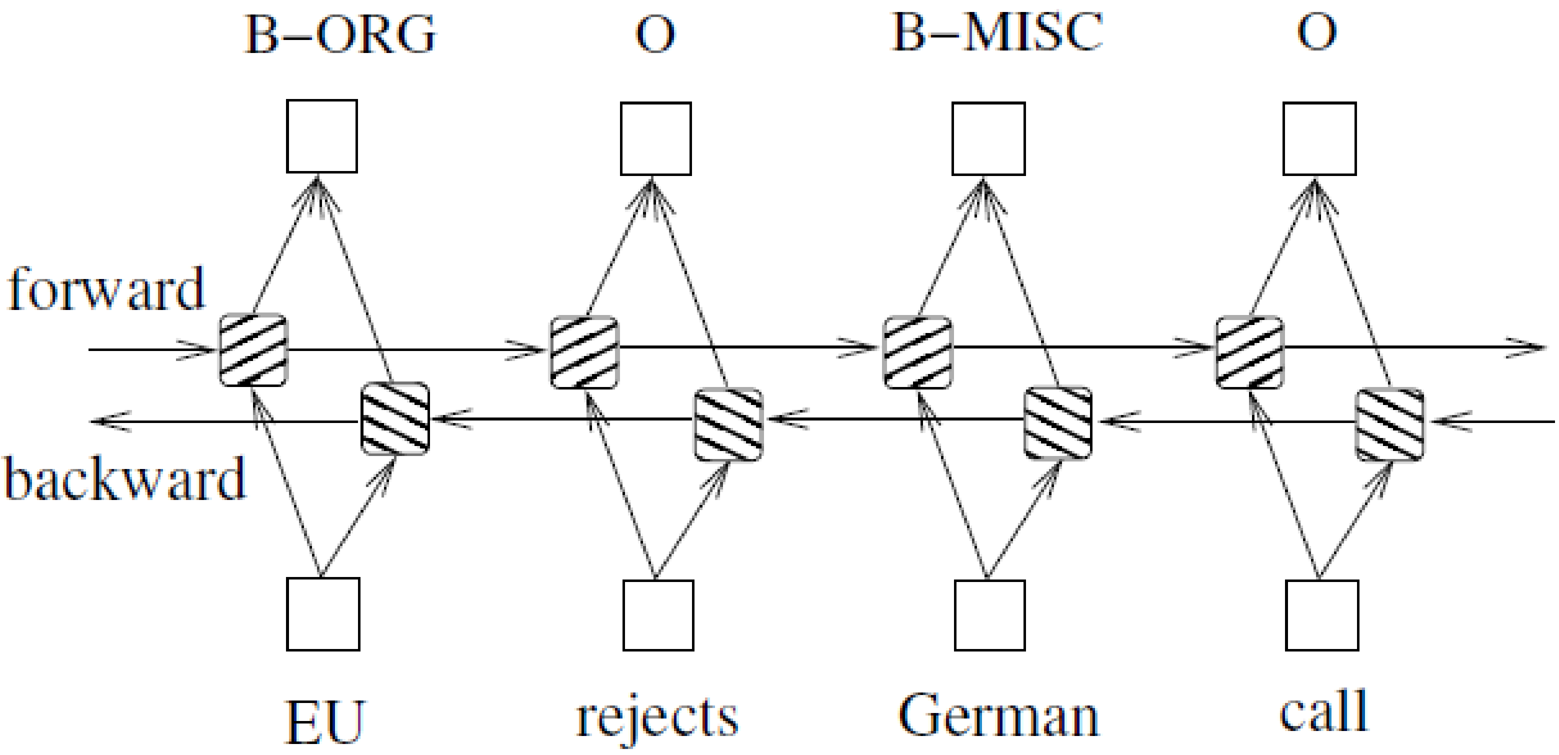
命名实体识别

- 主要方法
  - 基于规则的方法
  - 基于词典的方法
  - 机器学习方法
    - 最大熵
    - 条件随机场
    - 深度学习（选学）

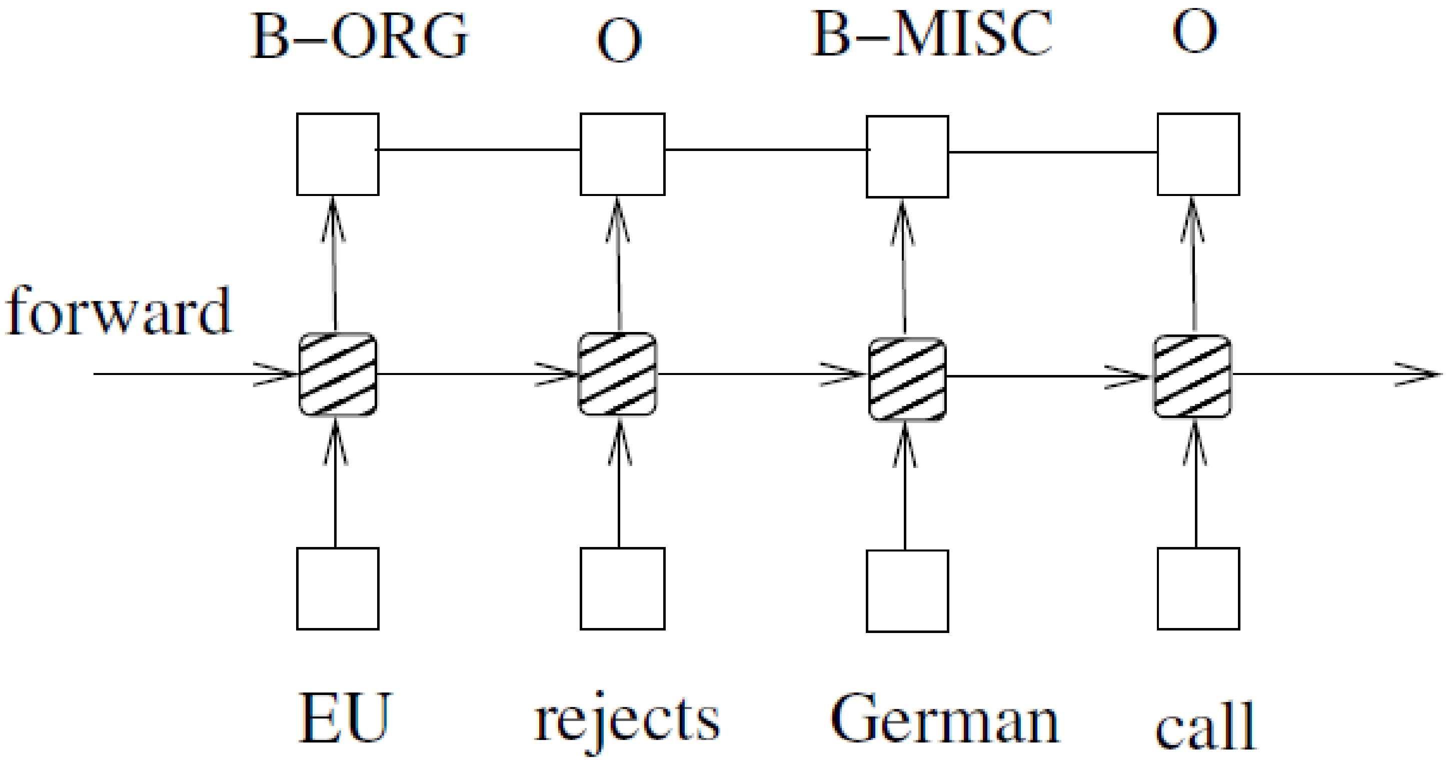
A LSTM sequence tagging model



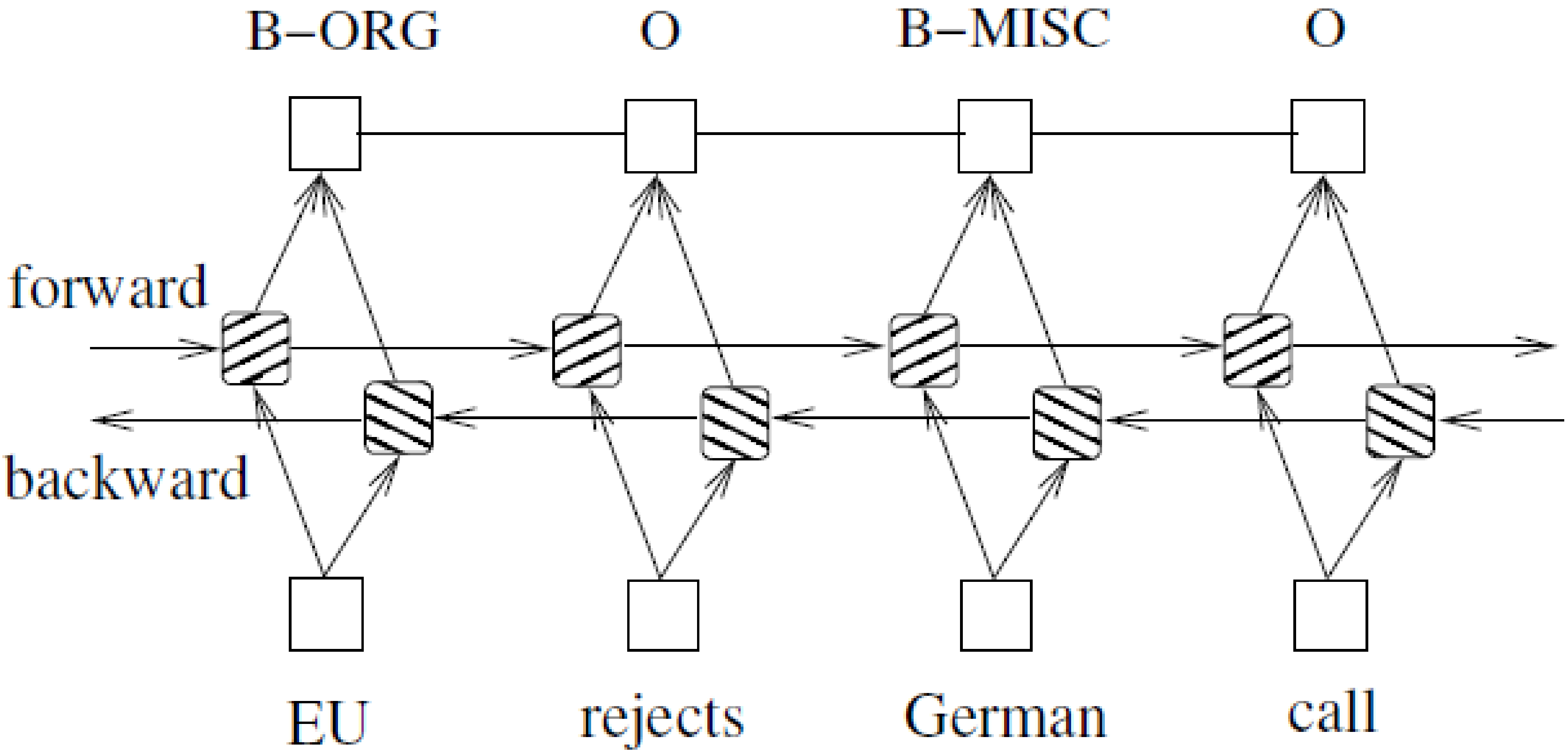
A bidirectional LSTM sequence tagging model



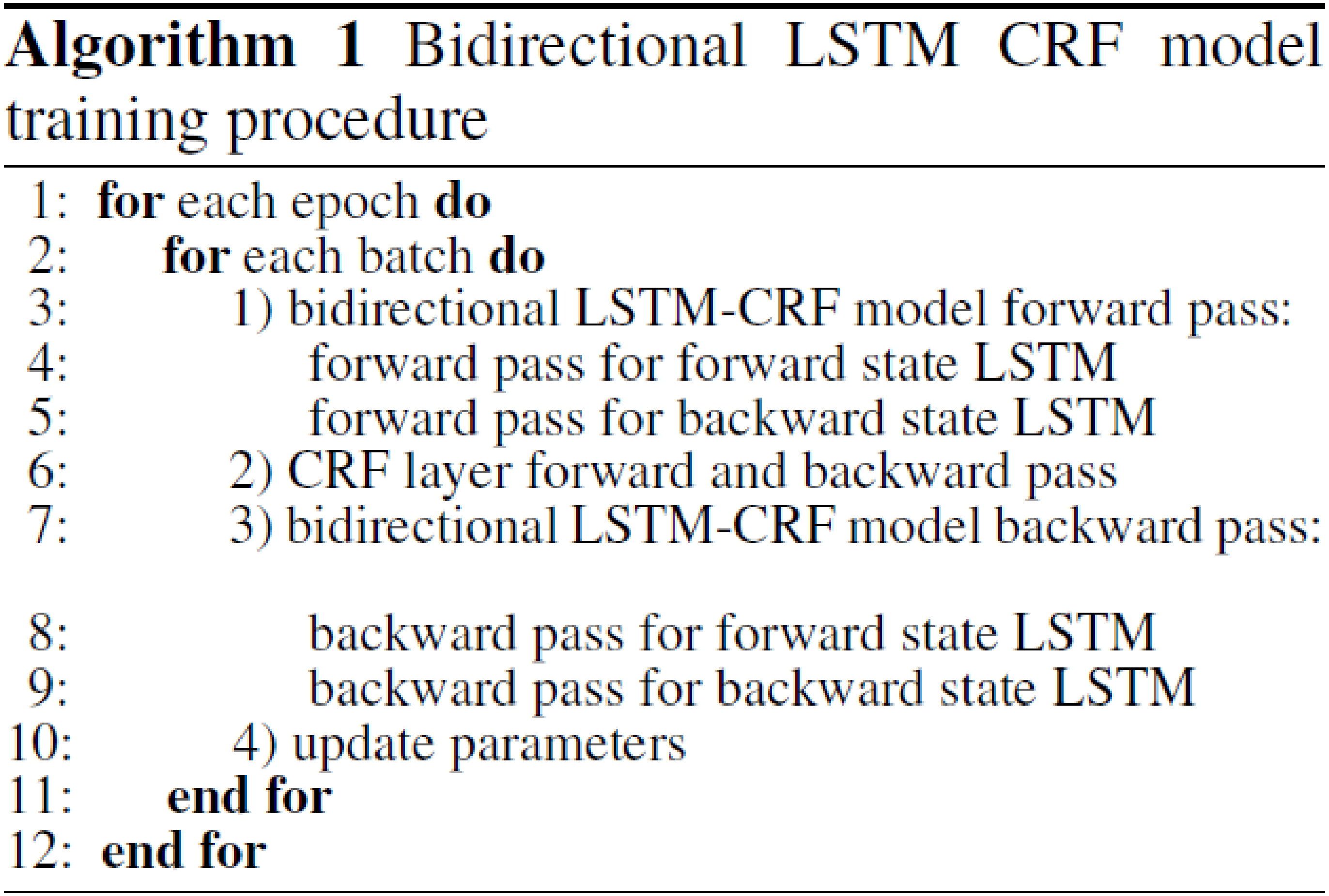
LSTM-CRF model



BiLSTM-CRF Model



BiLSTM-CRF Model

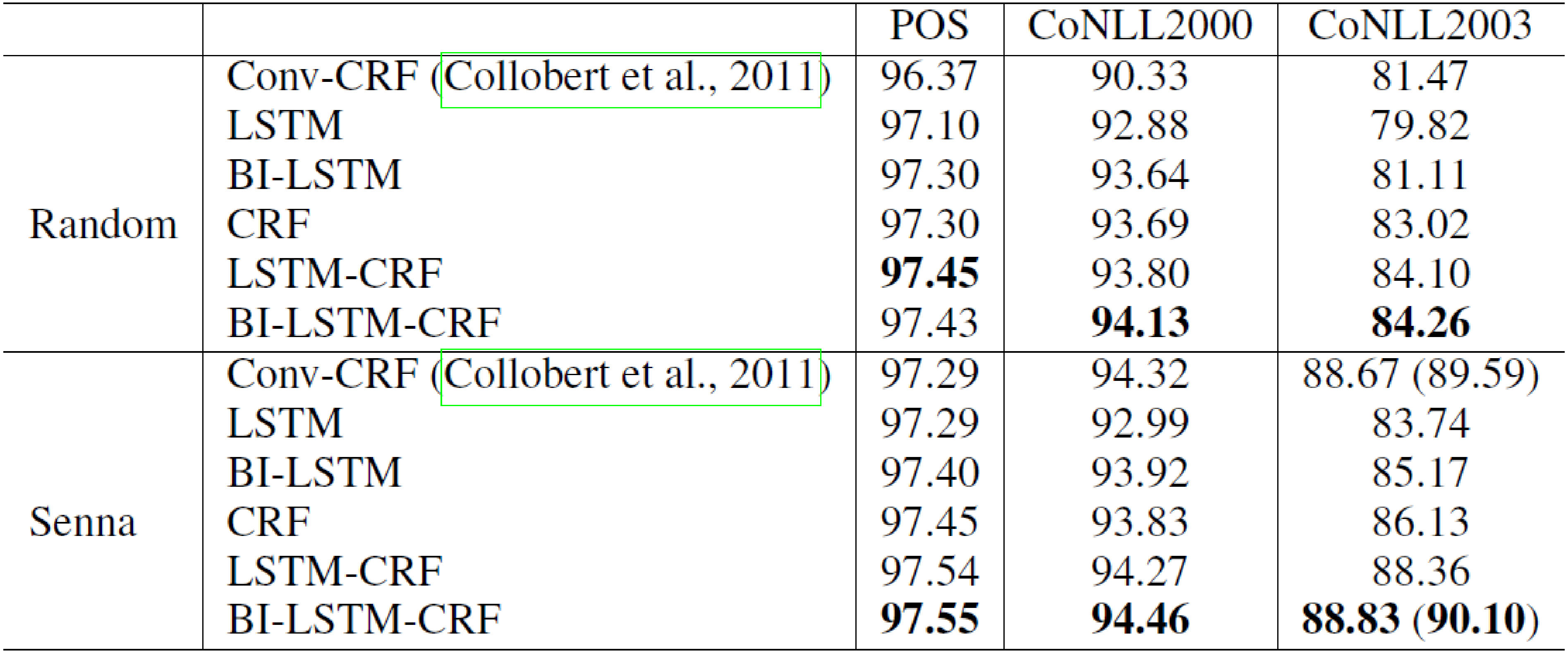


Algorithm 1 Bidirectional LSTM CRF model training procedure

```
for each epoch do
    for each batch do
        1) bidirectional LSTM-CRF model forward pass:
            forward pass for forward state LSTM
            forward pass for backward state LSTM
        2) CRF layer forward and backward pass
        3) bidirectional LSTM-CRF model backward pass:

            backward pass for forward state LSTM
            backward pass for backward state LSTM
        4) update parameters
    end for
end for
```

Zhiheng Huang, Wei Xu, Kai Yu: **Bidirectional LSTM-CRF Models for Sequence Tagging.** CoRR abs/1508.01991 (2015)

Comparison of tagging performance for various models



		POS	CoNLL2000	CoNLL2003
Random	Conv-CRF (Collobert et al., 2011)	96.37	90.33	81.47
	LSTM	97.10	92.88	79.82
	BI-LSTM	97.30	93.64	81.11
	CRF	97.30	93.69	83.02
	LSTM-CRF	**97.45**	93.80	84.10
	BI-LSTM-CRF	97.43	**94.13**	**84.26**
Senna	Conv-CRF (Collobert et al., 2011)	97.29	94.32	88.67 (89.59)
	LSTM	97.29	92.99	83.74
	BI-LSTM	97.40	93.92	85.17
	CRF	97.45	93.83	86.13
	LSTM-CRF	97.54	94.27	88.36
	BI-LSTM-CRF	**97.55**	**94.46**	**88.83 (90.10)**

命名实体识别的评价

	actual class (标准答案)	
predicted class (预测结果)	**tp** (true positive) Correct result	**fp** (false positive) Unexpected result
	fn (false negative) Missing result	**tn** (true negative) Correct absence of result

命名实体识别的评价

- 准确率 Precision(P)

$$P = \frac{tp}{tp + fp}$$

- 召回率 Recall(R)

$$R = \frac{tp}{tp + fn}$$

- F1度量 F1-measure(F1)

$$F1 = \frac{2PR}{P + R}$$

Useful Toolkits

- Maximum Entropy
  - https://github.com/lzhang10/maxent
- Conditional Random Fields
  - Mallet http://mallet.cs.umass.edu/fst.php
  - CRF++ https://taku910.github.io/crfpp/
  - LSTM-CRF https://github.com/abhyudaynj/LSTM-CRF-models

Data Set

- CONLL 2003 NER
  - https://www.clips.uantwerpen.be/conll2003/ner/
- BioNLP 2004 NER
  - http://nactem.ac.uk/tsujii/GENIA/ERtask/report.html