# Open Domain Event Extraction Using Neural Latent Variable Models Note

# 使用神经隐变量模型进行开领域事件提取笔记

## 摘要

该文考虑的是开放域事件提取，即从新闻集中提取无约束类型的事件的任务。(实际上是关注的slot的分配。)作者构建了一种新的神经隐变量模型，该模型可扩展到非常大的语料库。收集并手动注释数据集，并设计特定于任务的评估指标。（得把相同事件的新闻放在一次，实体是通过规则找，）

## 以前的工作

从新闻文本中提取事件已引起过很多研究关注。该任务通常由两个子任务组成，即概要归纳（schema induction），用于提取为给定事件类型指定参数槽（argument slot）的事件模板（template）和事件提取，用于从一条新闻中识别带有固定槽（slot）的事件。（一个事件模板（template）与许多槽（slot）有关。比如一个事件由主谓宾组成，则不同事件模板会给主谓宾三个槽填上不同的东西。这个任务反过来就是模板填充。）

以前的建模总是基于给定的事件类型。而金融与安全等领域通常难以完整的给出事件。而且无论如何大多数情况任何机器学习任务我们总有走向无监督的动力。（不过本文的一大假设是事件的报道总是冗余的，如果关注度不高的事件只出现了很少的几次，是很有可能无法提取出来的）

以前的模式归纳工作总是基于人工选取的特征，而本文选取了神经网络中的隐变量(ELMO)进行更好的表征以及神经变分推断策略。这实现了一种新的图模型。这个模型 采用来自一个全局参数化正态分布（各向同性） 的每一个 事件类型隐向量，以及同一实体的文本冗余特性 进行建模。

采用的NPMI来度量槽的一致性。（无监督的工作一定文章中出现度量结果好坏的办法。）

MUC4（1992）是基于槽提取事件的经典方法，关心4种事件（纵火，攻击，轰炸和绑架）以及4个固定槽的组合（犯罪者，工具，目标和受害者）。输入一个新闻输出一个事件，而ODEE则是输入一个新闻集输出一堆事件，并且槽是不固定的。

最早的开创性的工作其实是从基于模式与事件链入手来实现模板提取。

而对MUC4最多的策略是采用概率生成模型，（预测与参数分发一起做。）（2014左右）

Ad-hoc聚类推导概要（2016-2018）

上述一般都是手工特征，不过2015左右也有神网模型。

## 挑战

在开放域下事件不再和固定的槽（数量以及种类）有关。这些槽的分布也是不同的。99

在不同新闻中出现的同一实体的情况应该被考虑进来。

惊了，居然他的实验里面有这样一句话。。看来以后不能直接跳过实验。。。：

We compare our work with Nguyen et al. (2015), the state-of-the-art model on MUC 4 representing each entity as a triple containing a head word, a list of attribute relation features and a list of predicate relation features. Features in the model are discrete and extracted from dependency parse trees. The model structure is identical to our *ODEE-F* except for the features.

我们将我们的工作与Nguyen等人进行了比较。(2015)，最先进的muc 4模型将每个实体表示为一个包含头字、属性关系特性列表和pr列表的三元组。编辑关系特征。模型中的特征是离散的，并从依赖关系分析树中提取。除特征外，模型结构与我们的ODEE-F相同。（和阮2015一样。。）

也做了采用其他特征在odee上。

———— 一言概括：ELMO牛逼。。不过这实验是真的难做。因为它的slot是不固定的，而之前的任务想套用在这上面。。最后只有靠贪心槽映射。整个实验，一定需要许多工具人才能把这个做好（标定数据，与做实验。），不然一个人明显太消耗时间了。

注意他的event是一个向量建模，所以他又把event向量拿来聚类了。

河内科学技术大学有个做文本挖掘的团队。这篇文章是北京理工大学。

话说我一直想看下open

Case study：

一个槽在三个事件里出现

事件丢失了槽。（我们对事件的推导只出于与同一谓词相关的足够的多的实体。）

Features in the model are discrete and extracted from dependency parse trees. The model structure is identical to our *ODEE-F* except for the features.

基于实体消歧的生成事件模式归纳

一个槽跳到了另一个事件。（解析树错误传播）

蒸馏学习。

A feature is just a set words, it can be a sentence or a paragraph. A head word is basically the most important word of a feature. More specifically, the head word yields the syntactic category of that sentence; it’s the word that gives the most meaning and structure.

<https://en.wikipedia.org/wiki/Head_(linguistics)>

In linguistics, the **head** or **nucleus** of a [phrase](https://en.wikipedia.org/wiki/Phrase" \o "Phrase) is the word that determines the [syntactic](https://en.wikipedia.org/wiki/Syntax" \o "Syntax) category of that phrase.

在语言学中，**头**或**核** a的[短语](https://en.wikipedia.org/wiki/Phrase" \o "短语)是确定单词[句法](https://en.wikipedia.org/wiki/Syntax" \o "句法)该短语的类别

短语：

*boiling hot water* is the [noun](https://en.wikipedia.org/wiki/Noun" \o "Noun) *water*

*混合词：*

*handbag* is *bag*,

*[exocentric](https://en.wikipedia.org/wiki/Exocentric" \o "Exocentric)* ("headless") phrases and compounds (if they exist) lack a clear head.（无核的）

Head-initial phrases are right-branching, head-final phrases are left-branching, and head-medial phrases combine left- and right-branching.

需要把相同事件的新闻放在一起作为输入，事件类型在N下，而slot是一个事件的。等同于变成了slot的聚类问题，外加规则推断事件类型（至少这个规则蛮好用。）。最大贡献在于能够学习模式（集合式的模式）。

如何摄入监督信息？？我需要一个钥匙。。。聚类结果要如何和具体的情形对应上去。。Lda能聚出具体主题吗？而不是我们需要把主题一样的放在一起。

在传统的拉丁语和希腊语（及其他）语法中， government是[动词](https://en.wikipedia.org/wiki/Verb" \o "动词)和[介词](https://en.wikipedia.org/wiki/Preposition" \o "介词)对其他词的语法特征选择的控制

*A* governs *B* if and only if:[[3]](https://en.wikipedia.org/wiki/Government_(linguistics)" \l "cite_note-3)

* *A is a governor (a lexical head),*
* *A* [m-commands](https://en.wikipedia.org/wiki/M-command" \o "M-command) *B*, and
* no barrier intervenes between *A* and *B*.

所以事件是怎么来的：

我们把同一共引链（一个实体）中的多个实体提及的谓词合并为一个谓词集，把每一个谓词集中的谓词统计它和多少个实体相关，最多的n个输出。

coreference chain：

**government** or **rection** refers to the relationship between a word and its dependents.

2015年那篇是**Generative Event Schema Induction with Entity Disambiguation，基本上是第一篇单纯通过加元，方法是gibbs采样，超参数用的2017年那篇**

**伪装的无监督：要把一些相同的绑在一起。（如关注的事件。相关的实体）**

事实上虽然是个降维到映射的模式这种伪装的高大上的模式，

但是当你成果比别人好时即可。

# 通过实体消歧得到概要归纳（2015）

其思想是根据这些实体与谓词之间的关系的相似性，将对应于事件模板中相同角色的实体组合在一起。

## 把 与谓词之间的关系 相同的实体聚在一起作为一个slot

不过一般是一个软聚类，算这些实体对应不同slot的概率。

Slot和事件角色的概念很近只是一般不包括trigger本身。

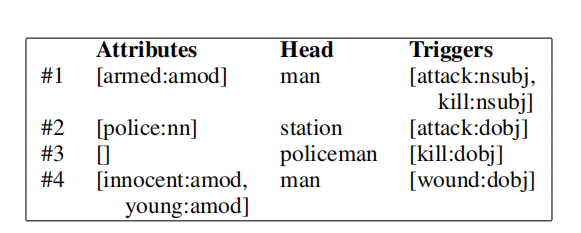
例如，在有关恐怖袭击的语料库（如MUC4）中，可以将作为杀死，攻击动词的对象的实体组合在一起，并以角色来表征 名为 VICTIM。

这个识别操作的输出是一组集合，其中的成员既是词又是关系，与它们的概率相关联(参见图4后面的示例)。这些集群没有被标记，但它们中的每一个都表示一个事件时隙。

核心在于邻近上下文有用有助于消歧，可以更确定它的slot

文档级别通常是主题建模的中心概念，但在我们的生成模型中并未使用。

其中从槽生成观察值，该槽由实体，谓词和句法属性上的概率分布定义。



将通用流程与数量有限的领域特定规则（Freedman 等人，2011）或示例（Grishman and He，2014）混合在一起，是一种减少使系统适应另一个领域所需的工作量的方法。

按需信息提取方法(Hasegawa等人，2004年；Sekine，2006年)和先发制人信息提取方法(Shinyama和Sekine，2006年)试图克服另一个困难。利用查询所选择的具有代表性的文档所导出的模板的方法。

基于知识结构的

一个实体表示为一个三元组，它包含：一个单词 h，

属性关系列表 A 和触发器关系列表 T。

A trigger relation is composed

of a predicate (*attack*, *kill*, *wound*) and a depen

dency type (subject, object). An attribute rela

tion is composed of an argument (*armed*, *police*,

*young*) and a dependency type (adjectival, nomi

nal or verbal modififier).

A head word is extracted if it is a nominal or

proper noun and it is related to at least one trig

ger; pronouns are omitted. A trigger of an head

word is extracted if it is a verb or an eventive noun

and the head word serves as its subject, object, or

preposition.

一个 主词可以有多个触发器。这些多重关系可以 来自单个句子内部的句法协调（例如，和），例如在说明 示例的第一句中就是这种情况。它们也可以 代表共同引用的跨句子链接

## 隐式语义可以在显示时通过有监督的方法实现语义映射。从而我们实现了半监督学习。

## 泛化能力强（不是特定于任务的）的显语义迁移能力更强

## 此外，我们还在模型学习的插槽和注释中的插槽之间执行插槽映射。继之前的 MUC 4（隐语义到显语义的映射）

（这个一般是有监督的。。）

## 工作（钱伯斯, 2013; Cheung 等。, 2013; 阮等。, 2015; Sha 等人。, 2016; 安, 2017），

## 我们实现自动贪心槽映射。每个参考时隙被映射到学习的时隙，该时隙根据 GNBusinessDev 上的 F1 得分度量排名最佳。

一般不直接单点映射的原因是因为概率中总是多点的，因为我们是把单点映射到了多点，

We use POS

tags and parse trees produced by the Stanford de

pendency parser (Klein and Manning, 2003) to ex

tract the predicate for the head word of each entity

mention. The following rules are applied: (1) if

the governor of a head word is *VB*, or (2) if the

governor of a head word is *NN* and belongs to the

*noun.ACT* or *noun.EVENT* category of WordNet,

then it is regarded as a predicate.

We merge the predicates of entity mentions in

the same coreference chain as a predicate set. For

each predicate *v* in these sets, we fifind the entities

whose predicate set contains *v*, treating the entities

as arguments of the event triggered by *v*. Finally,

by ranking the numbers of arguments, we obtain

top-N open-domain events as the output *Ec*.

一般采取Slot（类别）个（高斯或者多项式分布）混合。（然后做slot mapping）（实际等效于降维后映射。半监督。）（多项式分布如何算梯度？？）

双指一的逻辑

唯一夙愿，往生物意识上靠近，能往那边仿真。