一文详解知识图谱关键技术与应用

网易云有料　2018-09-10 11:06

讲师 | 桂洪冠

本课程从知识图谱的历史由来开展，讲述知识图谱与人工智能的关系与现状；知识图谱辐射至各行业领域的应用；在知识图谱关键技术概念与工具的实践应用中，本课程也会
建经验；以及达观在各行业领域系统中的产品开发和系统应用。

大家晚上好！我是达观数据的桂洪冠，负责达观的搜索技术团队。非常高兴今天晚上能给大家做一个分享，分享的主题是“知识图谱的关键技术和应用”。

达观数据是一家专注于文本智能处理的人工智能技术企业，我们为企业提供完善的文本挖掘、知识图谱、搜索引擎、个性化推荐的文本智能处理技术服务。

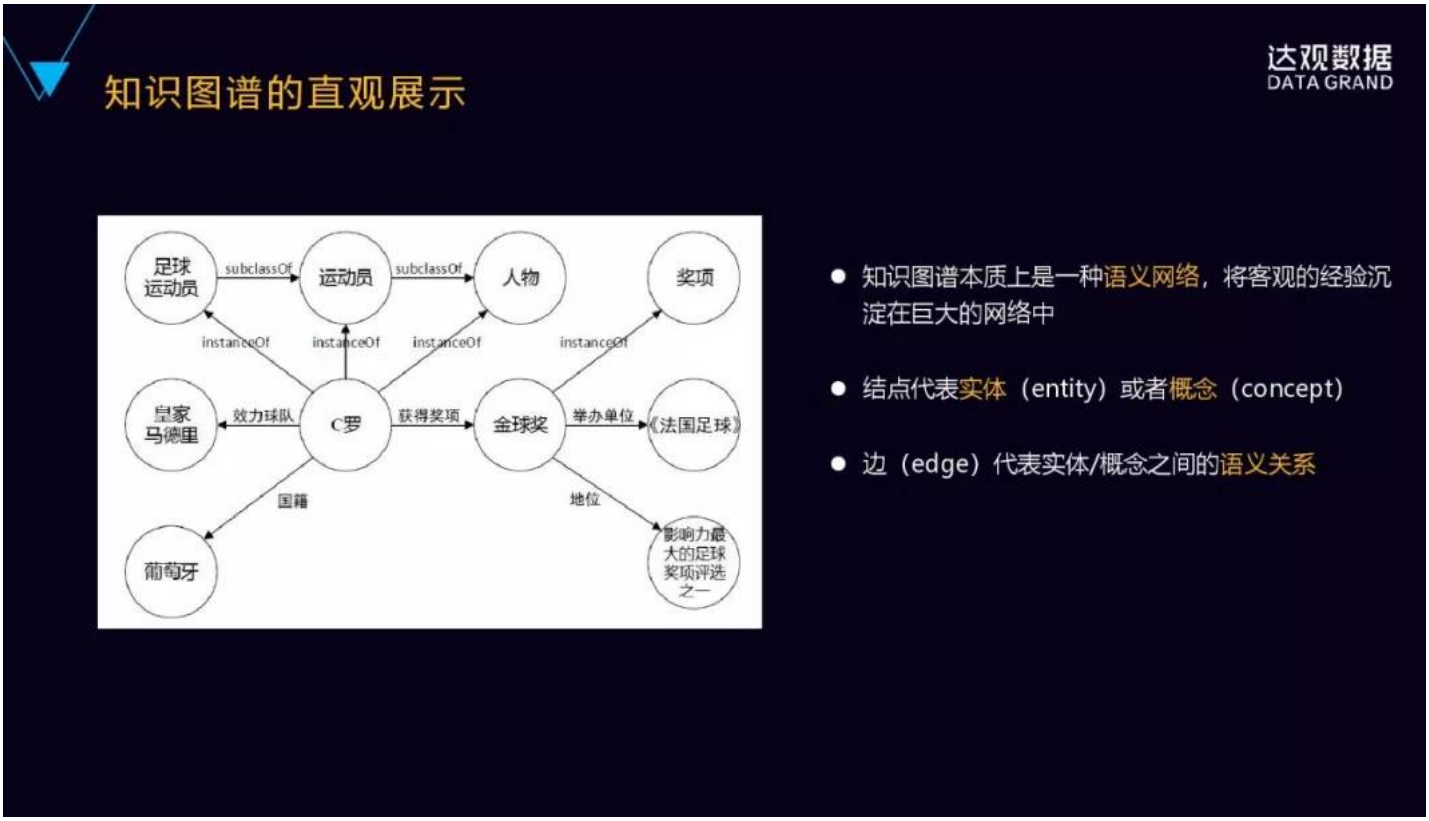
言归正传，进入今天的演讲环节。今天的演讲主题是“知识图谱关键技术与应用”，分成几个环节：

- 一、知识图谱的相关概述；
- 二、知识图谱的基本概念；
- 三、知识图谱行业方面的应用和场景介绍，着重讲一下知识图谱构建的相关技术；
- 四、达观在知识图谱构建方面的经验、心得和相关案例。
- 最后是与大家的Q&A互动环节。

一、知识图谱的概述

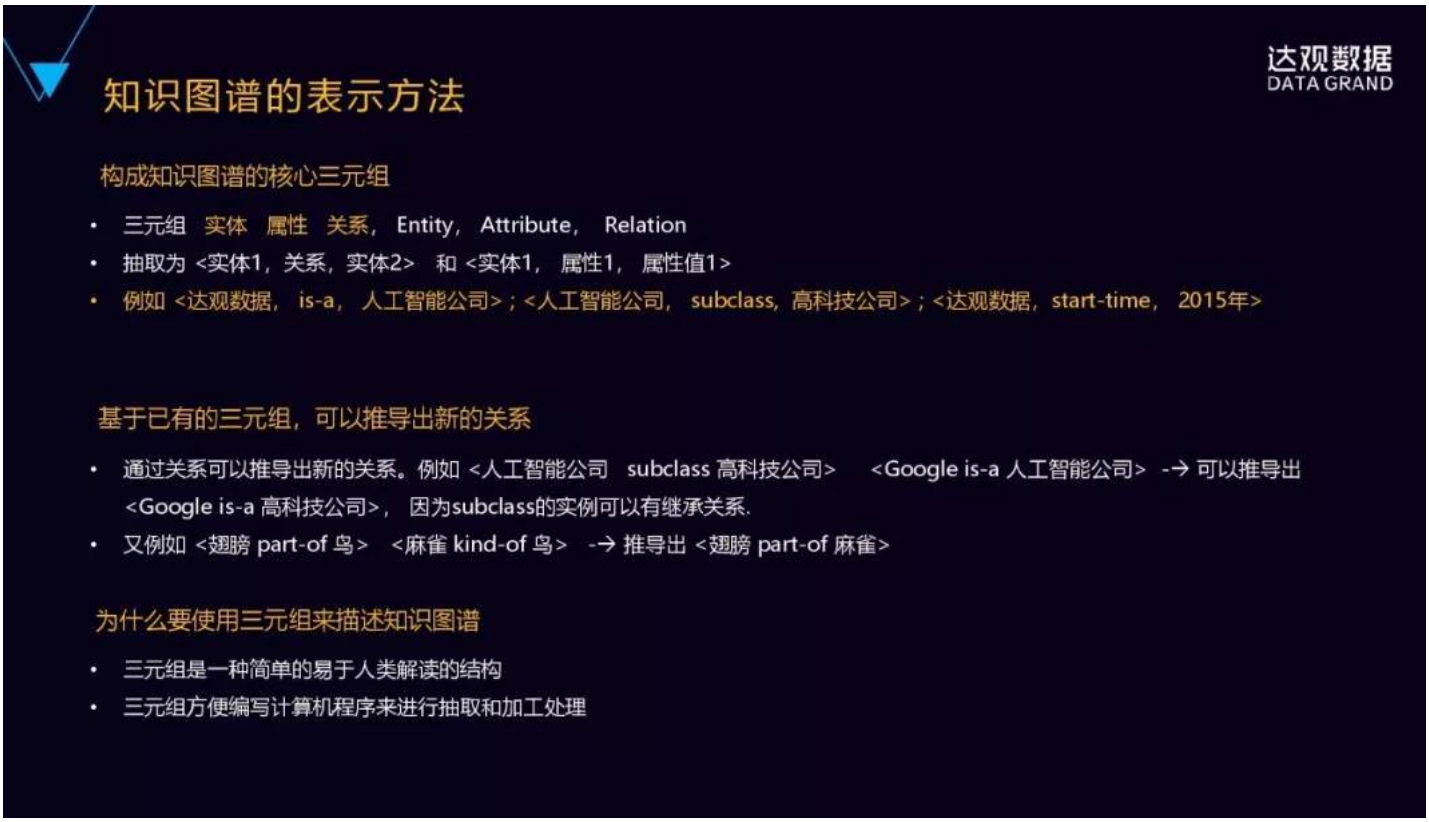
我们先直观的来看一下什么是知识图谱，下面有一张图，从这张图里可以看到，这个图里圆圈是节点，节点之间有一些带箭头的边来连成，这个节点实际上相当于知识图谱里
边连线表示实体之间的关系。

知识图谱本质上是一种大型的语义网络，它旨在描述客观世界的概念实体事件以及及其之间的关系。以实体概念为节点，以关系为边，提供一种从关系的视角来看世界。



语义网络已经不是什么新鲜事，早在上个世纪就已经出现了，但为什么重新又提到知识图谱?

知识图谱本质上是一种语义网络，但是它最主要的特点是一个非常大规模的语义网络，之前的语义网络受限于我们处理的方法，更多是依赖于专家的经验规则去构建，在规模域的数据。大规模网络，谷歌在2012年首先提出知识图谱的概念，在freebase的基础上扩展了大量来自互联网的实体数据和关系数据。据说目前实体的数据已经达到数十亿级实例关系，规模是非常巨大的。



我们再看一下，知识图谱背后是怎么表示的，我们看到的是一个巨大的语义网，背后是怎么存储或者表示的呢?

首先，它是由三元组构成的，构成知识图谱的核心其实就是三元组，三元组是由实体、属性和关系组成的（由Entity、Attribute、Relation组成）。

具体表示方法为，实体1跟实体2之间有某种关系，或者是实体属性、属性词。

举个例子，“达观数据是一家人工智能公司”，其实就可以表示成这样的三元组：

<达观数据，is-a，人工智能公司>。

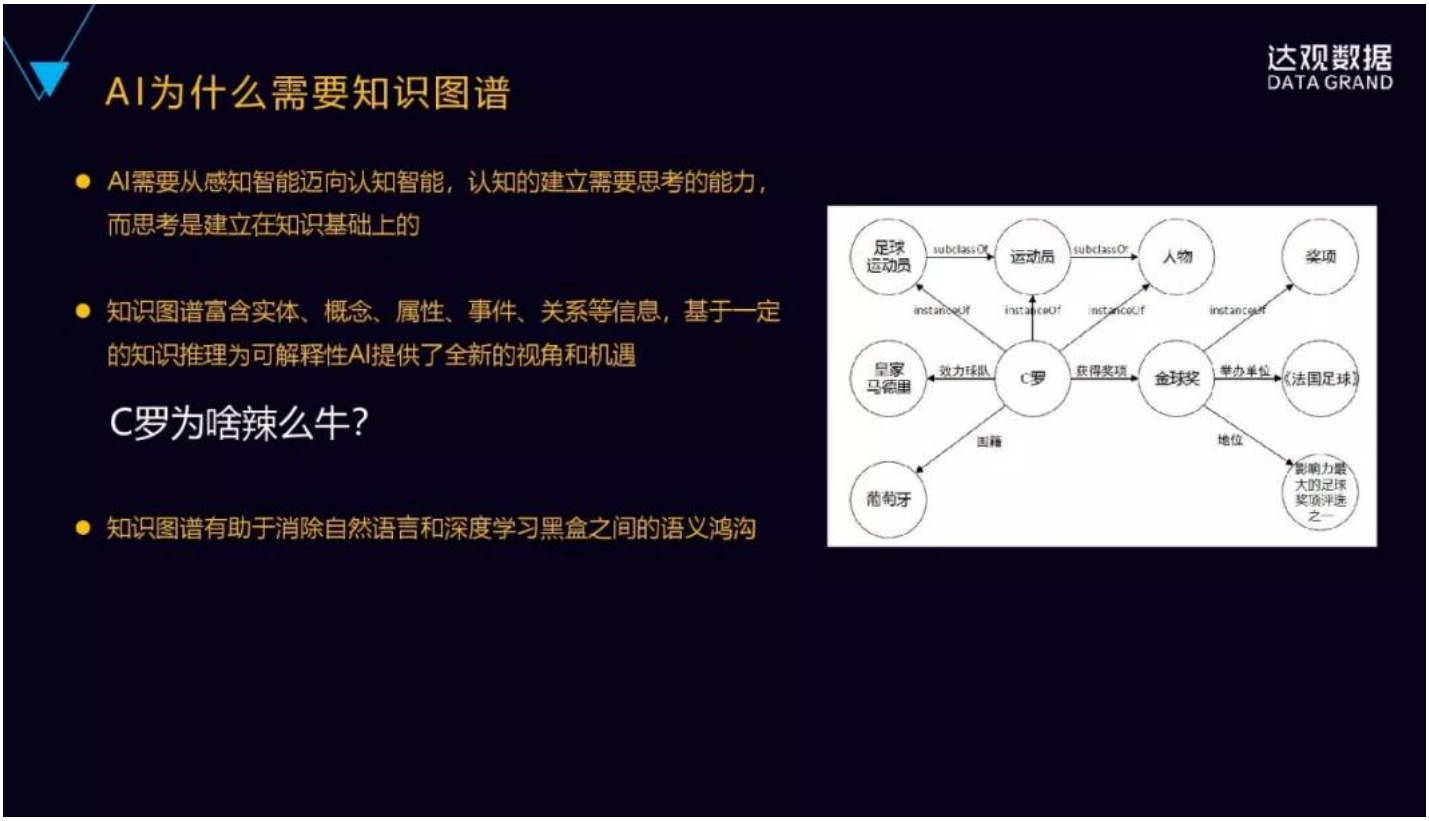
“人工智能公司是一种高科技公司”可以表示成：

<人工智能公司，subclass,高科技公司>。

“达观数据成立于2015年”，也可以把这个属性表示成一个三元组，就是：

<达观数据，start-time，2015年>。

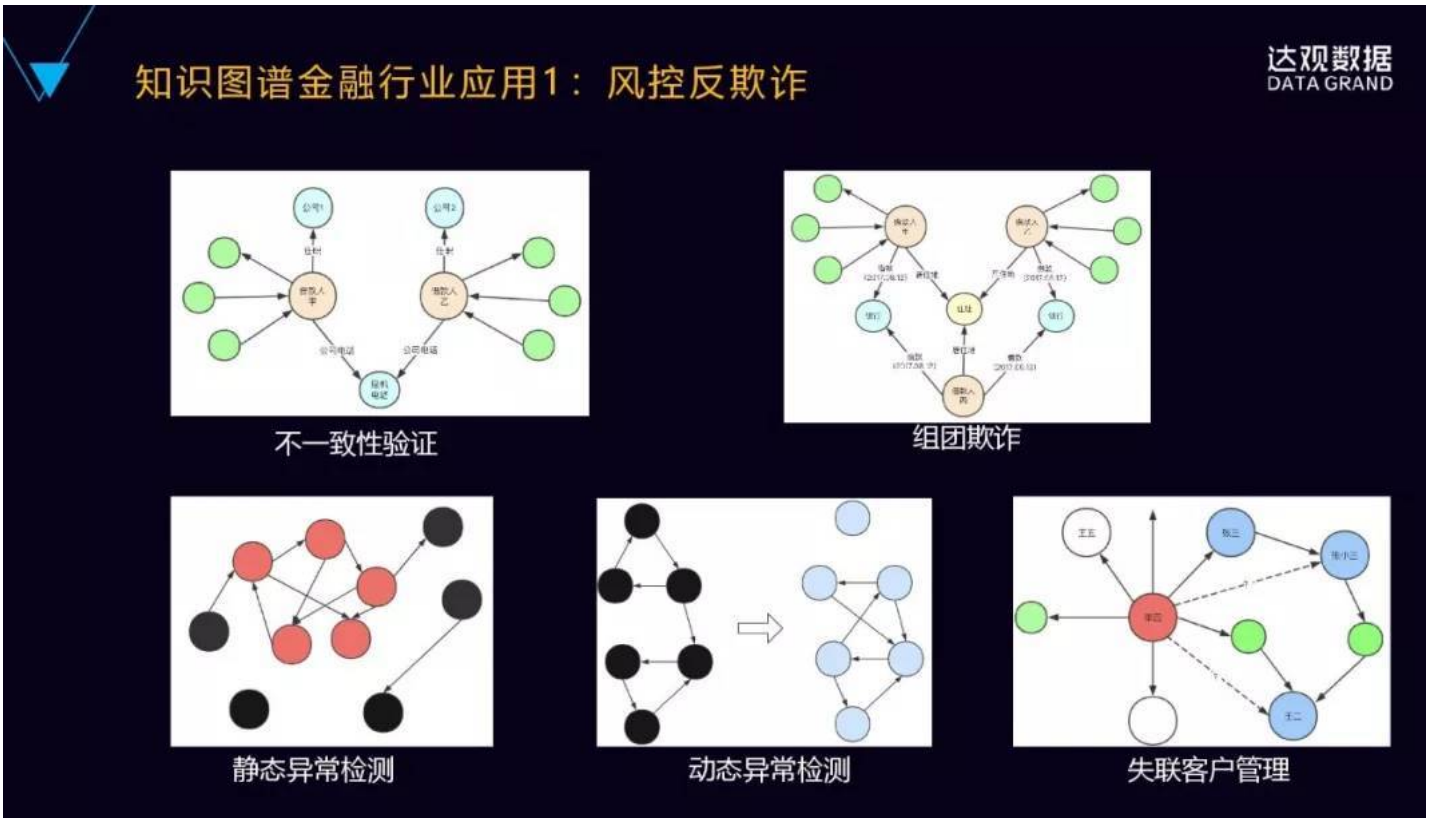
基于已有的三元组，它可以推导出新的关系，这个对构建知识图谱来说是非常重要的。我们知道，知识图谱要有丰富的实体关系，才能真正达到它实用的价值。完全靠人工去
的，所以内部一定有一个自动推理的机制，可以不断的去推理出新的关系数据出来，不断的丰富知识图谱。
来看一些具体的例子。
“人工智能公司是一种高科技公司”，subclass的关系。
还有一个三元组是谷歌是一家人工智能公司，<Google is-a人工智能公司>，可以由这两个三元组推导出谷歌是一家高科技公司，<Google is-a高科技公司>。因为subclass的实
承的关系。
<翅膀part-of鸟>，<麻雀kind-of鸟>，可以推导出<翅膀part-of麻雀>。
为什么要用三元组来描述知识图谱?
三元组是一个人和计算机都易于理解的结构，人是可以解读的，计算机也可以通过三元组去处理，所以它是一个既容易被人类解读，又容易被计算机来处理和加工的结构，而
单，如果说你扩充成四元组、五元组，它整个结构就会变得比较复杂，那是综合的一种复杂性和人的易理解性、和计算机的易出理性来综合的考虑，决定用三元组的结构来去
储。
那么，AI为什么需要知识图谱?
人工智能分为三个阶段，从机器智能到感知智能，再到认知智能。
机器智能更多强调这些机器的运算的能力，大规模的集群的处理能力，GPU的处理的能力。
在这个基础之上会有感知智能，感知智能就是语音识别、图像识别，从图片里面识别出一个猫，识别人脸，是感知智能。感知智能并非人类所特有，动物也会有这样的一些感
再往上一层的认知智能，是人类所特有的，是建立在思考的基础之上的，认知的建立是需要思考的能力，而思考是建立在知识的基础之上，必须有知识的基础、有一些常识，
考，形成一个推理机制。



AI需要从感知智能迈向认知智能，本质上知识是一个基础，然后基于知识的推理，刚好知识图谱其实是具备这样的一个属性。
知识图谱其实是富含有实体、属性、概念、事件和关系等信息，它能够基于一定的推理。且比较关键的是，它能够基于一定的推理为AI的可解释性，带来全新的一个视角。
可解释性已被一些领域AI大规模使用，比如医疗领域，AI进行癌症的诊断的结果，如果没有给出一个合理的一个理由，或者是给出一个解释的一个方法，医生是不敢贸然的用
的结果去给病人直接做下一步的措施。包括金融领域也一样，AI如果给投资人推荐了一个投资的方案，但是没有给出任何的一个解释跟说明的话，也会存在巨大的一个风险。
也是一样，用AI进行判案，AI给一个案件判定一个结果，但是没有给出任何的一个解释，也是不能作为结果来采用的，因为司法强调的就是一种可解释性，对法律的解释性、
为什么说知识图谱可以做这样一个可解释性呢?
举个例子，我们问“C罗为什么那么牛？”
C罗为什么那么牛？这个是一个问题，要解释回答这个问题，人通常是怎么样去回答这样的问题呢？上图，通过知识图谱的简单的推理，就可以回答这样一个问题，因为C罗获
跟金球奖之间的关系是获得奖项的一个关系，金球奖跟影响力最大的足球评选奖项之一有这样一个地位的关系，它具有这样的一个非常高的地位，C罗又获得过这个奖项，所
是很牛的。这是一种知识图谱来解释、来回答这样一个“为什么”的一个问题。
同样还有一些问题，比如，“鳄鱼为什么那么可怕？”
人类是有一样这样的常识，所有的大型的食肉动物都是很可怕，这是个常识。鳄鱼是一种大型的食薄动物，鳄鱼跟大型食肉动物概念之间是一种instance的关系。通过这样的
间的关系，可以推导出鳄鱼是很可怕的。同样的，“鸟儿为什么会飞？”因为它有翅膀，鸟儿这个实体它的属性是有翅膀，利用一个实体跟属性之间的关系，可以做这样一个推
之前微博上关晓彤跟鹿晗非常的火，经常被刷屏，这是为什么?
因为关晓彤跟鹿晗之间是男女朋友这样的关系，明星之间的男女朋友的关系就最容易被大家追捧，也最容易被刷屏。这个就是通过关系也好，通过实体的属性也好，通过实体
以去解释、去回答一些问题。这些是知识图谱在AI在可解释性方面的一些具体的例子。
深度学习的可解释性非常差的，深度学习里面内部的语义表达、向量的表达都是一些浮点数，人类是非常难以理解的。深度学习出来的结果，它的可解释性也是非常少的。
尽管我们现在在研究可视化的技术，把中间的它的结果呈现出来、可视化出来，但是真正能达到对人有效的解释性进展还是比较缓慢的。知识图谱实际上是有望能够消除人类
学习黑盒之间的语义鸿沟。也就是深度学习的底层的特征空间和上层的人的自然语言空间这种巨大的语义鸿沟，通过深度学习跟知识图谱结合起来，有望能够消除。这也是为
图谱的一个原因。

二、知识图谱的典型行业应用介绍

1. 金融行业的应用。



知识图谱在金融行业里面比较典型的应用就是风控反欺诈。

（1）. 知识图谱可以进行信息的不一致性检查，来确定是不是存在可能的借款人欺诈的风险，比如第一个图里面的借款人甲和乙来自于不同的公司，但是他却非常诡异地留下话号码，这时审核人员就要格外留意了，有可能会存在欺诈的风险。

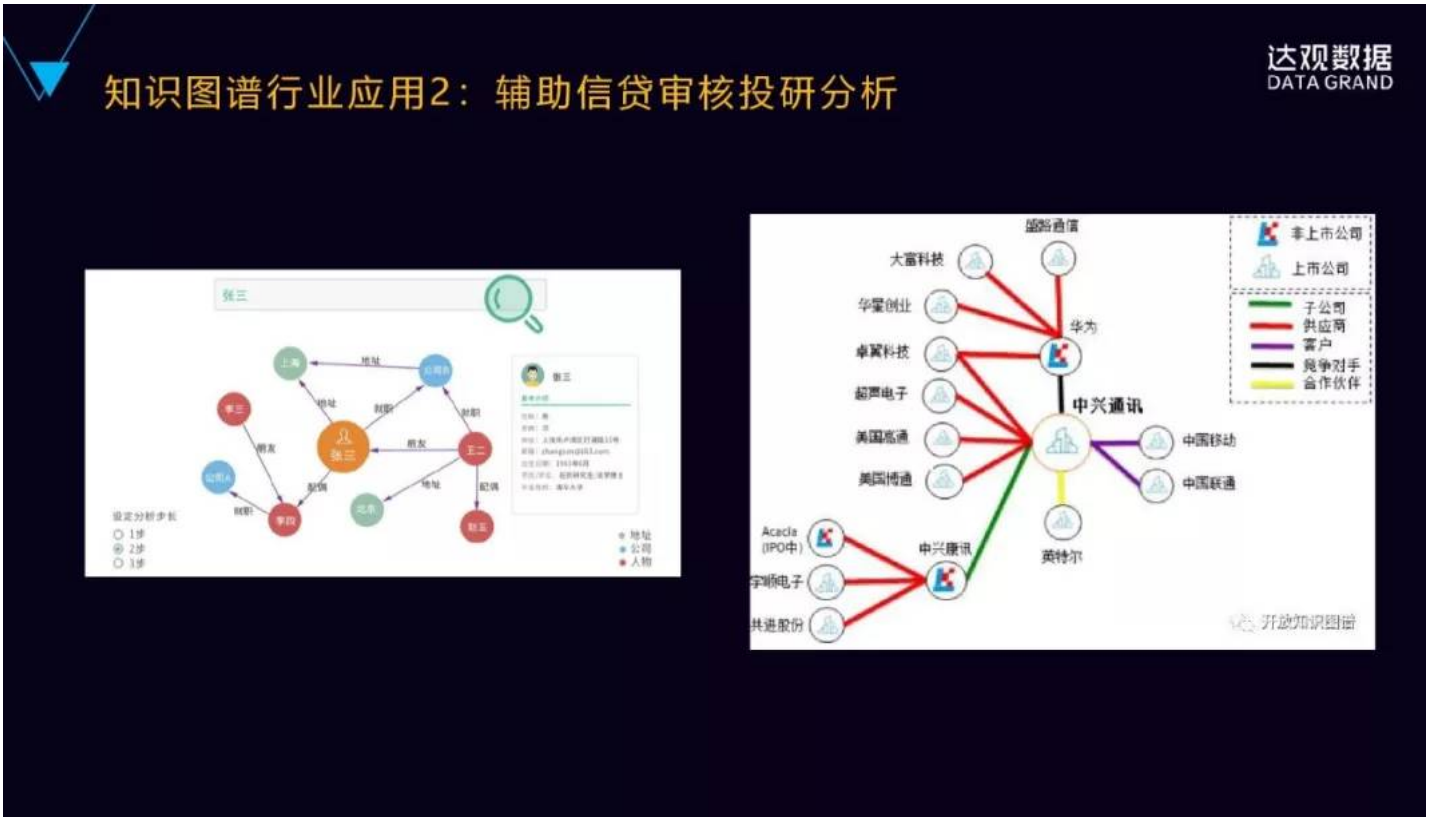
（2）. 组团欺诈，甲乙丙三个借款人同一天向银行发起借款，他们是互不相关的人，但是他们留了相同的地址，这时有可能是组团的欺诈。

（3）. 静态的异常检测，它表示的是在某个时间点突然发现图中的某几个节点的联系异常的紧密，原来是互相联系都比较少、比较松散的，突然间有几个点之间密集的联系，组织。

（4）. 动态的异常检测（第二行中间图），是随着时间的变化，它的几个节点之间图的结构发生明显的变化，原来它是比较稳定的，左边黑色的上三角、下三角，然后中间连间之后，它整个图的结构变成了右边的这样结构，此时很可能是异常的关系的变化，会出现一个欺诈组织。

（5）. 客户关系管理。怎么样去做失联客户的管理？图中的例子有一个借款的用户，银行可能现在没有办法直接找到他，甚至通过他的直接联系人也没办法找到他，那这个时一步的通过他的二度联系人来间接的来找到他？通过这样的图结构是可以快速找到他的二度联系人，比如张小三或者是王二，再去联系他们，尝试把李四这个人给找到。

2.辅助信贷审核和投研分析



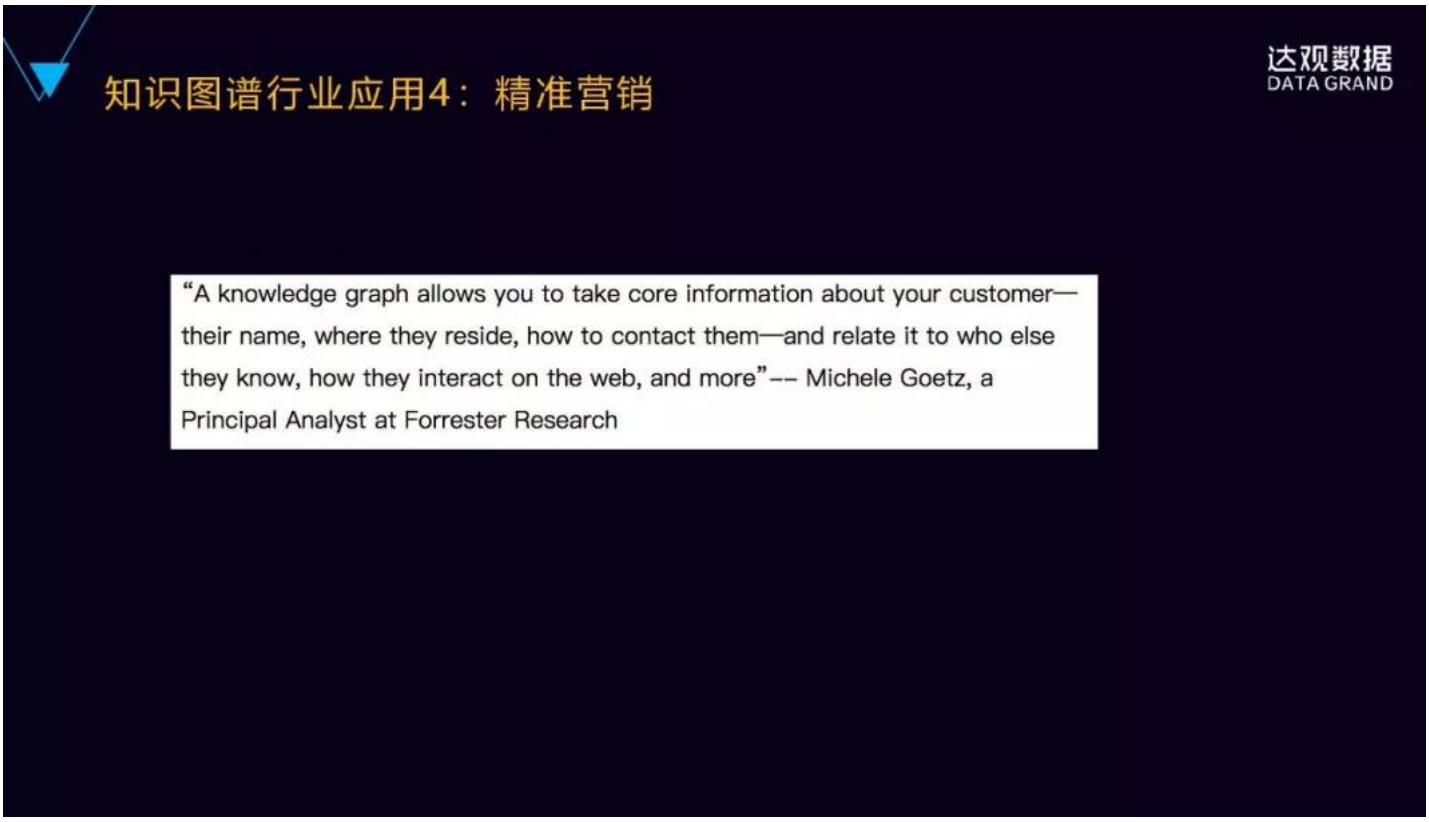
左边是辅助审贷。

知识图谱会融合多个数据源，从多个维度来维护关联人员的信息，来避免数据不全与数据孤岛，把它整合到一个大的网络结构里面去，借助知识图谱的搜索，审核人员可以快
请人张三的相关的信息，住址、配偶、就职公司、他的朋友等等。这比原来到各个异构且散落的数据源去进行搜集的效率要高得多，且能够从整体上来看到关键实体相互之间
第二个是用于辅助投研的。

知识图谱能够实时地串联起来这个公司相关的上下游公司，供应商的关系、竞争者的关系、客户的关系、投融资那些关系等，然后进行快速实时的定位。中信通讯这家公司前
行合规性审查，这个时候投研人员通过知识图谱搜索到中兴通讯公司实体，进而可以非常快地得到跟中兴通讯相关的上下游公司实体，包括关联的子公司、供应商、客户、竞
伴，有助于投研人员快速的做决策。

3.精准营销应用

知识图谱能够比较全面的记录客户的非常详细的信息，包括名字，住址，经常和什么样的人进行互动，还认识其它什么样的人，网上的行为习惯、行为方式是什么样的，这样
掘出更多的用户的属性标签和兴趣标签，以及社会的属性标签，形成全面的用户洞察，基于知识图谱就可以进行个性化的商品或者活动的推送，或者基于用户的分群分组做定
精准营销。



4. 知识图谱在搜索引擎里面的应用，最典型的就是在谷歌搜索引擎里面应用。

谷歌是在2012年率先提出来知识图谱的概念。提出这个概念的最主要的目的就是改善它的搜索引擎的体验。从这个图就可以看到，用户搜索的是泰姬陵，泰姬陵是印度的非常八大奇迹之一的景点。

不一样的地方，在搜索引擎的右侧，会以知识卡片的形式来呈现跟泰姬陵相关的结构化的信息，包括泰姬陵的地图、图片、景点的描述、开放时间门票等等，甚至在下面会列或者相关联的景点，比如中国的万里长城，同样是世界的几大奇迹，还有金字塔等等。同时，它还可以进行知识的扩展，比如泰姬陵不光是印度的景点，它还可以是一张音乐外某城市的街区街道。这样通过知识图谱可以不断的去探索发现新的非常新奇的东西，让用户在搜索引擎里面能够不断的去进行知识的关联和发现，激发起用户的搜索的欲望擎讲究的是说快速的找到它的结果，然后关掉就完了，谷歌通过知识图谱，实际上是把搜索引擎变成了知识的探索和发现引擎，这是概念和理念上的非常大的变迁与升级。

5.知识图谱应用于推荐系统

我们比较熟悉的是个性化推荐，即所谓的千人千面，比如根据游戏来推荐游戏的道具。对于小白用户和骨灰级的用户，推荐的东西显然是不一样的，这是个性化的推荐。个性场景化的推荐，比如用户购买了沙滩鞋，存在用户可能要去海边度假这样的场景，基于这样的场景可以继续给他推荐游泳衣、防晒霜或者其它的海岛旅游度假的产品。

任务型的推荐。比如用户买了牛肉卷或者羊肉卷，假设他实际上是要为了做一顿火锅。，这时候系统可以给他推荐火锅底料或者是电磁炉。

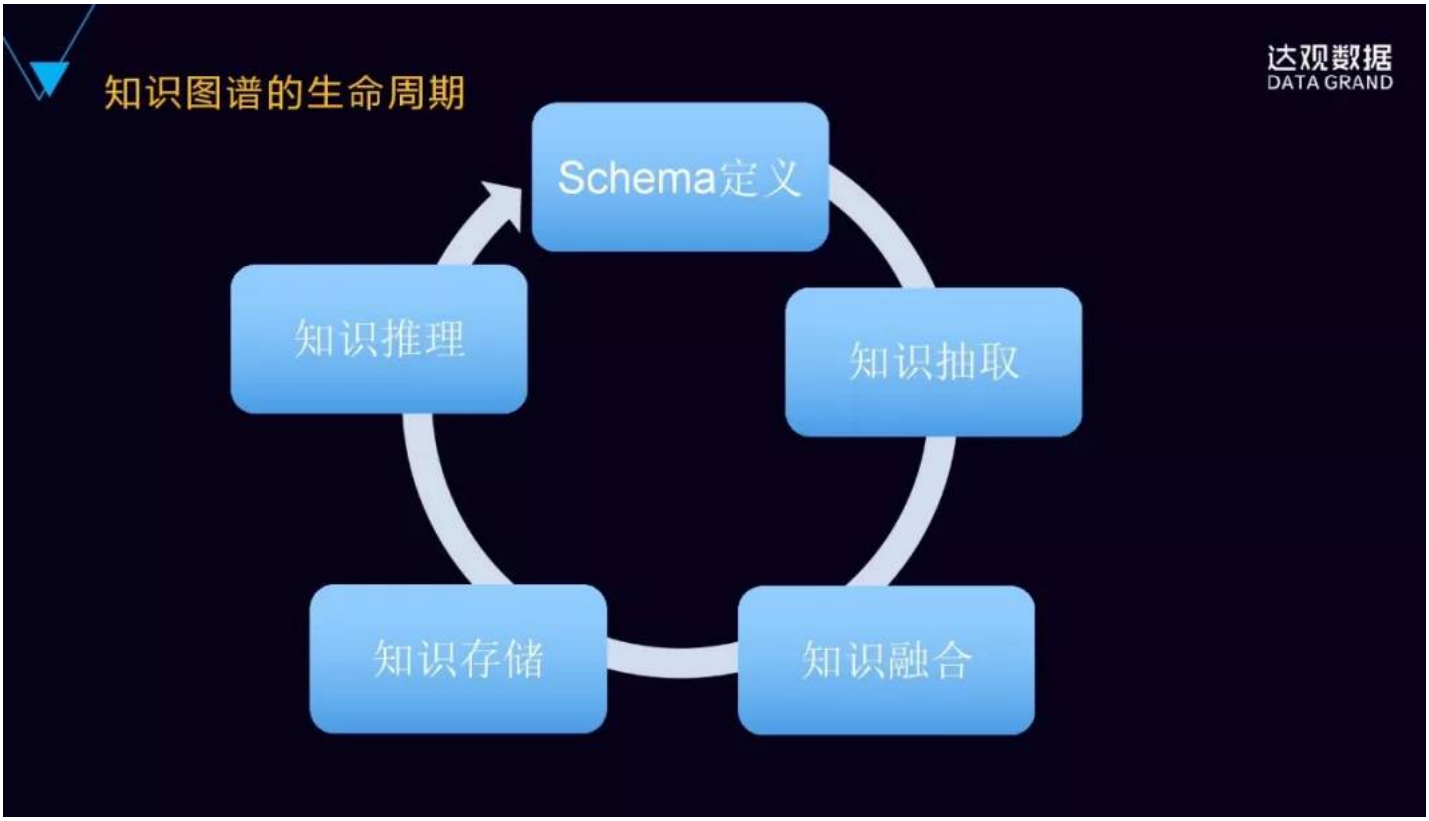
冷启动问题。推荐系统的冷启动一直是比较难以处理的问题，通常的做法是根据新用户的设备类型，或者他当前的时间位置等等，或者外面的关联数据来做推荐。可以基于知标签进行推荐，比如旅游和摄影实际上是语义相近的两个标签，再比如相同的导演或者相同演员的电影在语义上也是比较相近的。

跨领域的推荐问题。微博的信息流里会推荐淘宝的商品，然而微博和淘宝是两个不同的领域，它是怎么做到的呢？新浪微博有些用户会经常去晒黄山、九寨沟、泰山等这些照就知道他有可能是一位登山的爱好者，这个时候淘宝就会可以给他推荐登山的装备，登山杖、登山鞋等等这些装备，利用这些背景知识，能够打通不同的平台之间的语义鸿沟

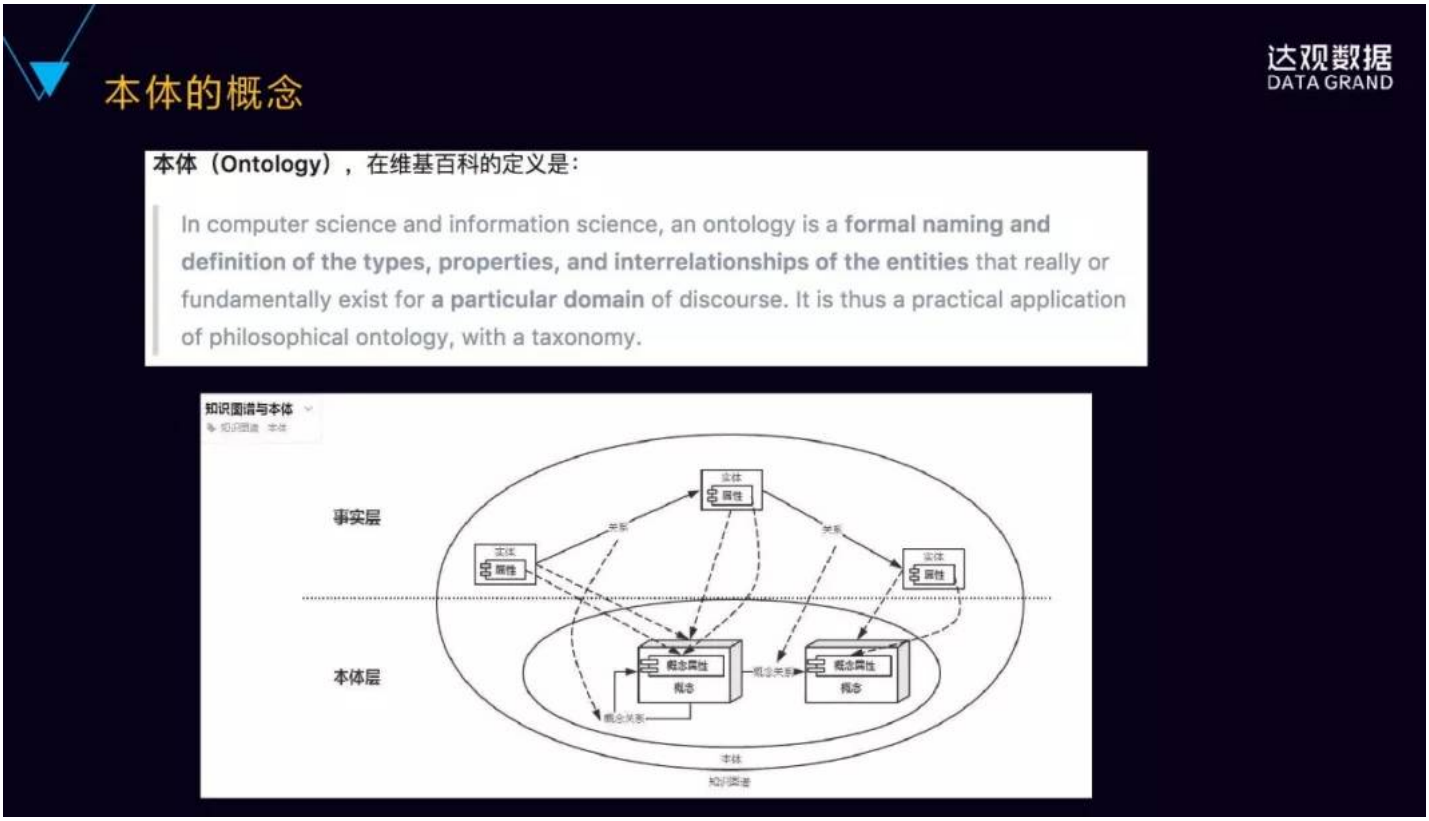
知识型的推荐，是基于知识的。比如清华大学、北京大学都是顶级名校，复旦大学也同样是，这个时候是可以推荐复旦大学，再比如百度、阿里和腾讯都属于BAT级互联网公司
里就可以推荐腾讯。
有了知识图谱以后，我们可以从基于行为的推荐，发展到行为跟语义相融合的智能推荐。

三、如何构建知识图谱

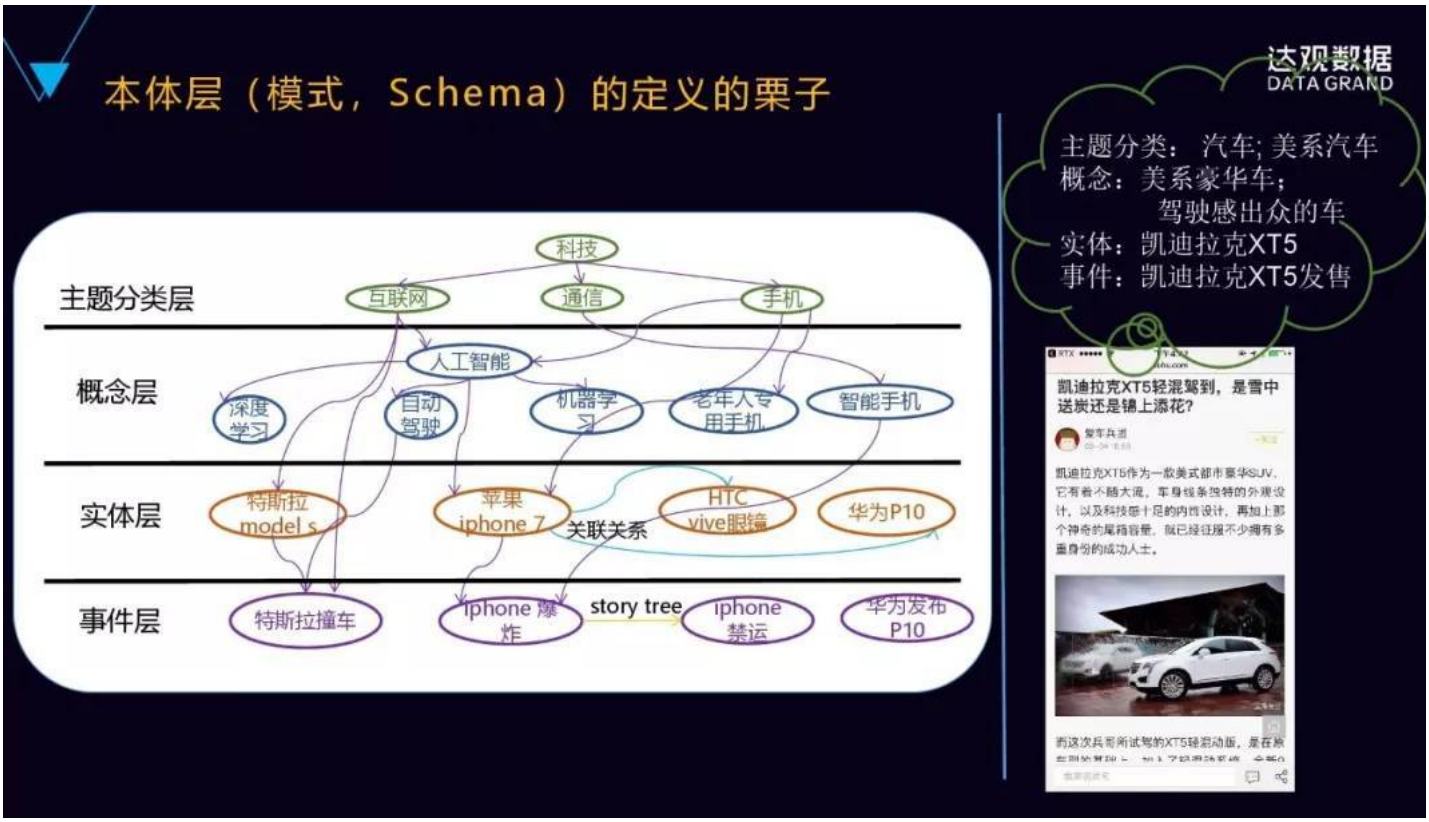
构建知识图谱是包括这样的生命周期或这样的部分，包括定义、知识的抽取、知识的融合、存储、知识的推理、知识的应用，这样的循环迭代的过程。
我们先来理解一下本体的概念，本体是用于描述事物的本质的，维基百科里面对于计算机科学领域当中的本体给出的定义是这样的，即：对于特定领域真实存在的实体的类型
之间的相互关系的一种定义。



我们再来看看知识图谱和本体的关系，下面这张图，我们看到有两个层，就下面是本体层，上面是事实层，那本体层是基于特定领域的概念的定义，包括概念的属性，概念之
念之间的关系是一种父子关系，也有叫做上下位的关系。事实层是具体的真实存在的实体，包括实体的属性以及实体之间的关系，每个实体都会映射到本体层相应的概念。面
熟悉，给大家举例说明一下，本体层的概念就好比面向对象里面的类的概念，然后事实层的实体就好比面向对象里面的对象，对象是从类当中派生出来的，同时继承了类的一
就是本体的概念。



为什么要讲本体的概念？一个更直观的例子，就是要对知识图谱来进行模式或者Schema的定义。这里的领域是科技，在这科技领域下面是我们划分的若干个主题，比如互联网主题、手机主题，在每个主题下面又有若干的概念，就是下面的概念层，比如互联网里面有深度学习、人工智能等等这些概念，然后通讯里面有智能手机这些概念。在概念下面是实体层，比如特斯拉modelS、苹果iPhone7、华为P10等等，这些都是具体的实体，然后实体之间有关系，它继承的概念之间的关系。最下面一层是事件层，这就是栗子。

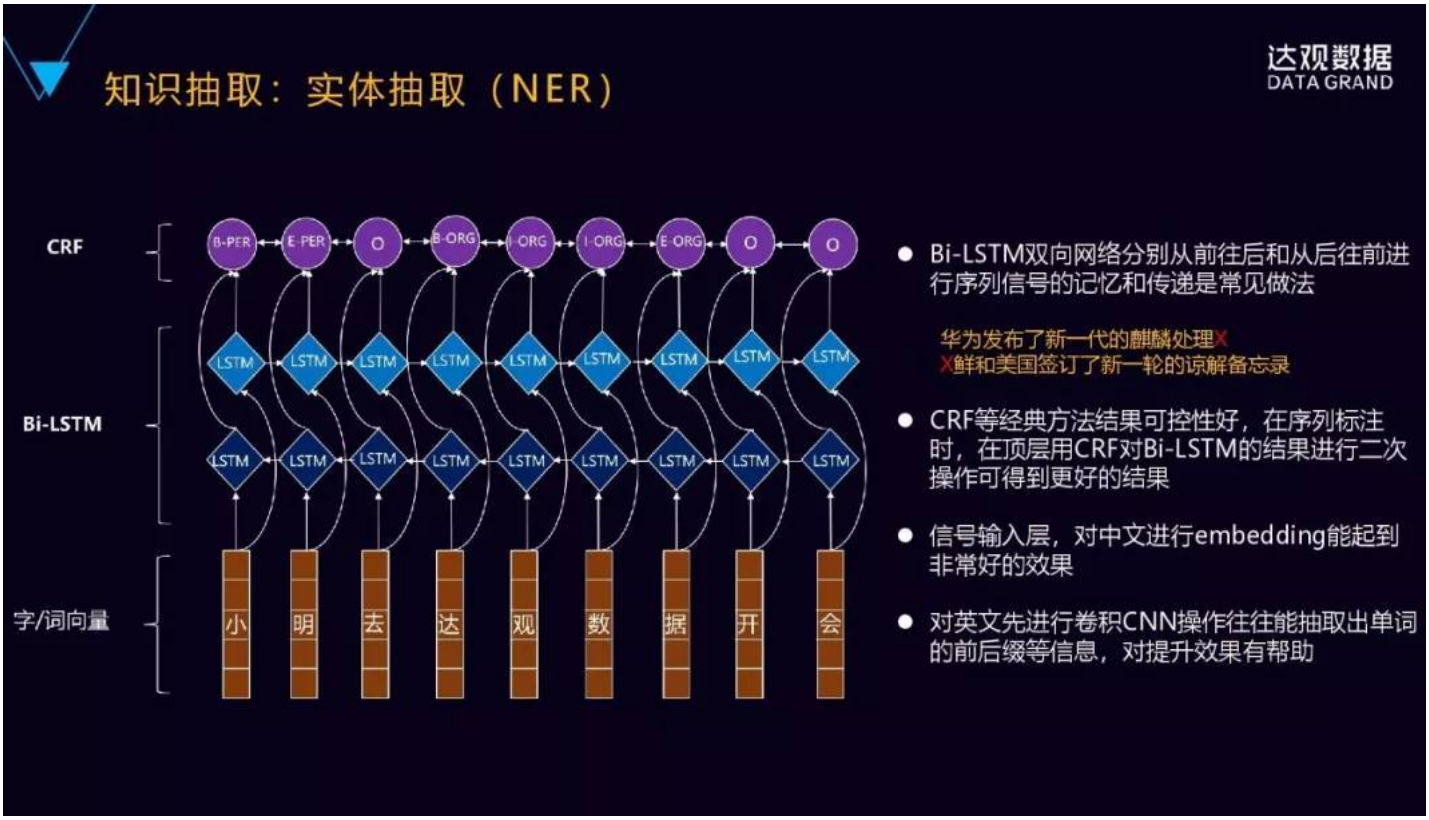


给大家介绍一款比较好的开源本体编辑工具，叫Protégé。这个工具是斯坦福大学开源的，它的功能非常强大，也是目前最流行的本体编辑工具，有网页版和桌面版，桌面版是直接下载下来就可以试用。

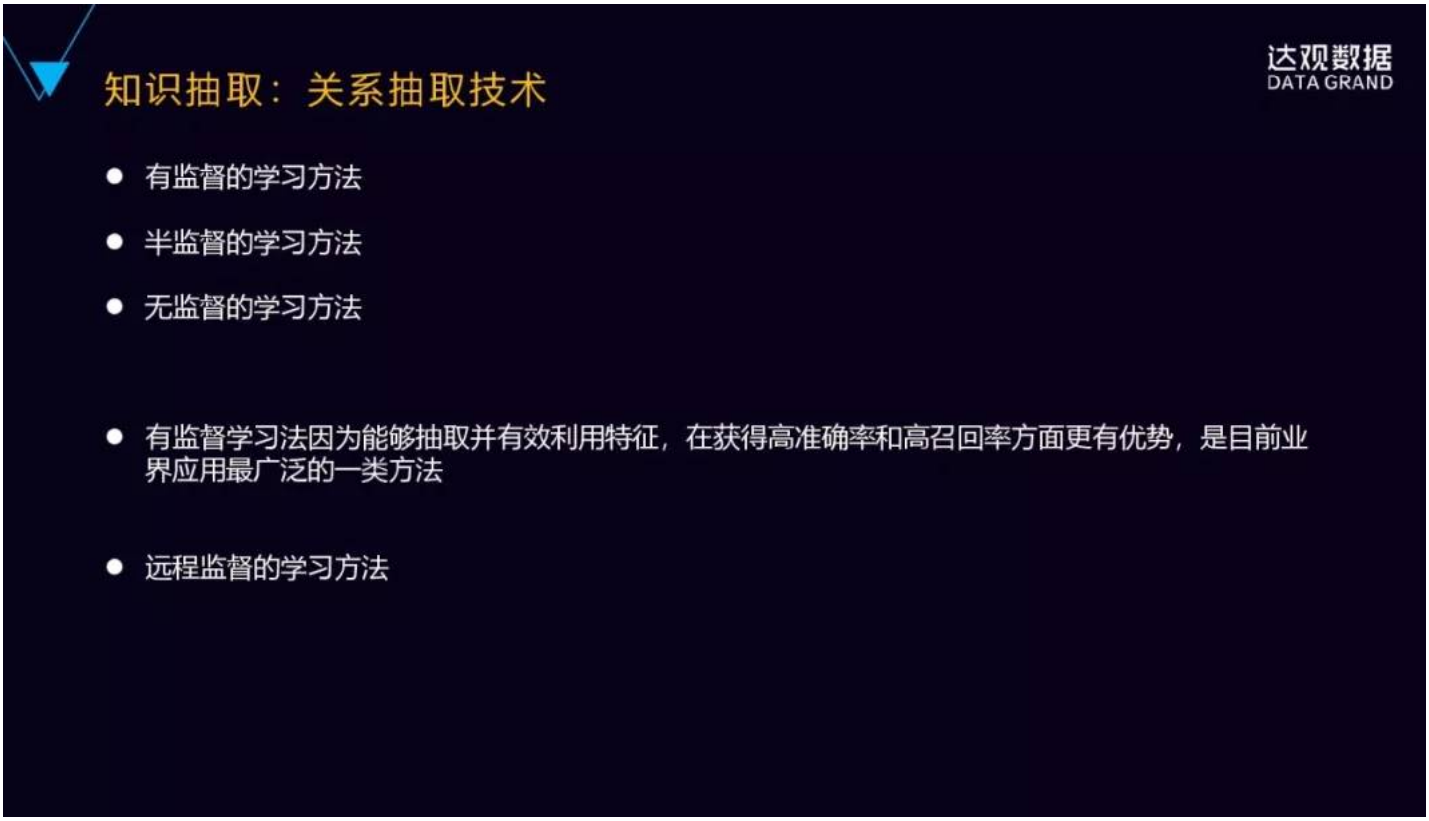
它的好处是什么？它屏蔽了具体的本体描述语言，用户只需要在概念层次上面进行本体的模型构建，同时也比较灵活，能够支持各种插件来扩展特定的功能。比如推理的功能扩展。不过这个工具对中文的支持不是很友好。

下面一个非常重要的关键的步骤就是知识的抽取，首先要抽取实体，然后是实体之间的关系。我们看一下NER实体的抽取，我们知道NER可以转化为序列标注的问题，传统的CRF等都可以做，而且CFR做的效果还是不错的。不过CRF通常只能学习到相邻词位置比较近的上下文的特征，它无法获取整个句子甚至更长的上下文的特征。

目前业界比较主流的，包括学术界比较主流的一种做法是什么？是深度循环神经网络加上结合CRF，这样的做法更多的是双向的循环神经网络，它可以分别从前往后以及从后去学习上下文的特征，然后进行序列信号的记忆和传递，这是一种比较常见的做法。



华为发布了新一代的麒麟处理，通过从前往后就可以推理出最后面的文字，然后从后往前也可以推理出最前面的文字，第一个字"朝"实际上是从后往前去进行预测推理。顶层网络的结果进行约束就可以，对这个输出进行更好的控制。然后输入层通常是词向量或者字向量，embedding能可以把单个字或者词转化为低维的稠密的语义向量。

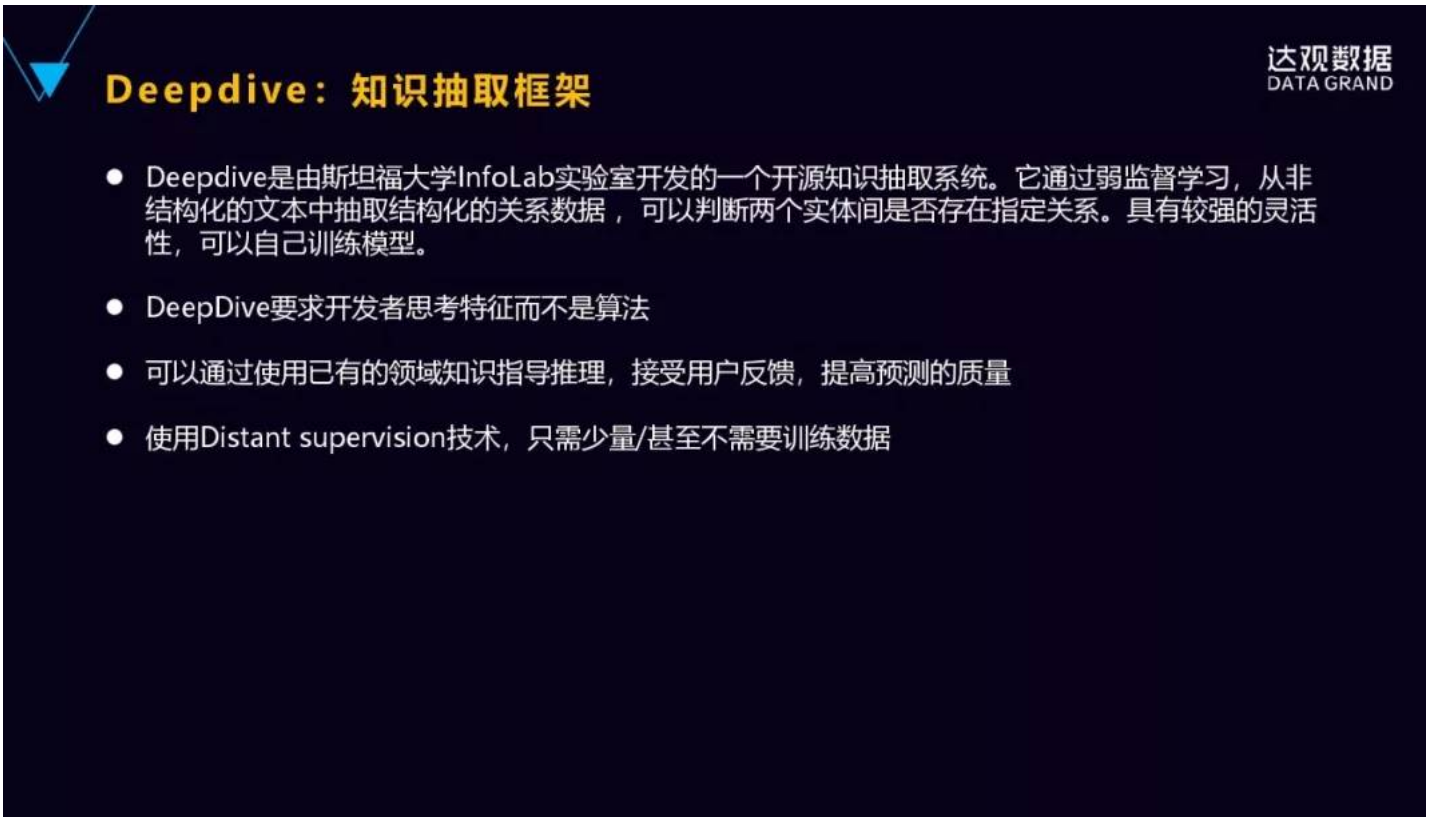


下面讲一下关系抽取的技术。它有几种方法，第一种是基于有监督的方法，把关系抽取当做分类问题来看待，根据训练数据设计有效的特征，来学习各种分类模型，这是传统
不好的地方，是需要大量的人工标注的训练语料，语料的标注是非常的耗时耗力的。所以在有监督的基础上，又提出了一种半监督的方式，半监督的方式主要采用方式进行关
说就是要对于要抽取的关系，首先基于手工设定若干的种子的实例，然后迭代性的从数据当中抽取关系对应的关系模板和更多的实例，通过不断迭代的方式来抽取越来越多的
还有一种是无监督的方法。无监督的方法本质上是一种聚类的方法，用拥有相同语义关系的实体，它拥有相似上下文的信息是它的假设，因此它可以利用每个实体的上下文的
语义关系，对实体进行语义关系的聚类。

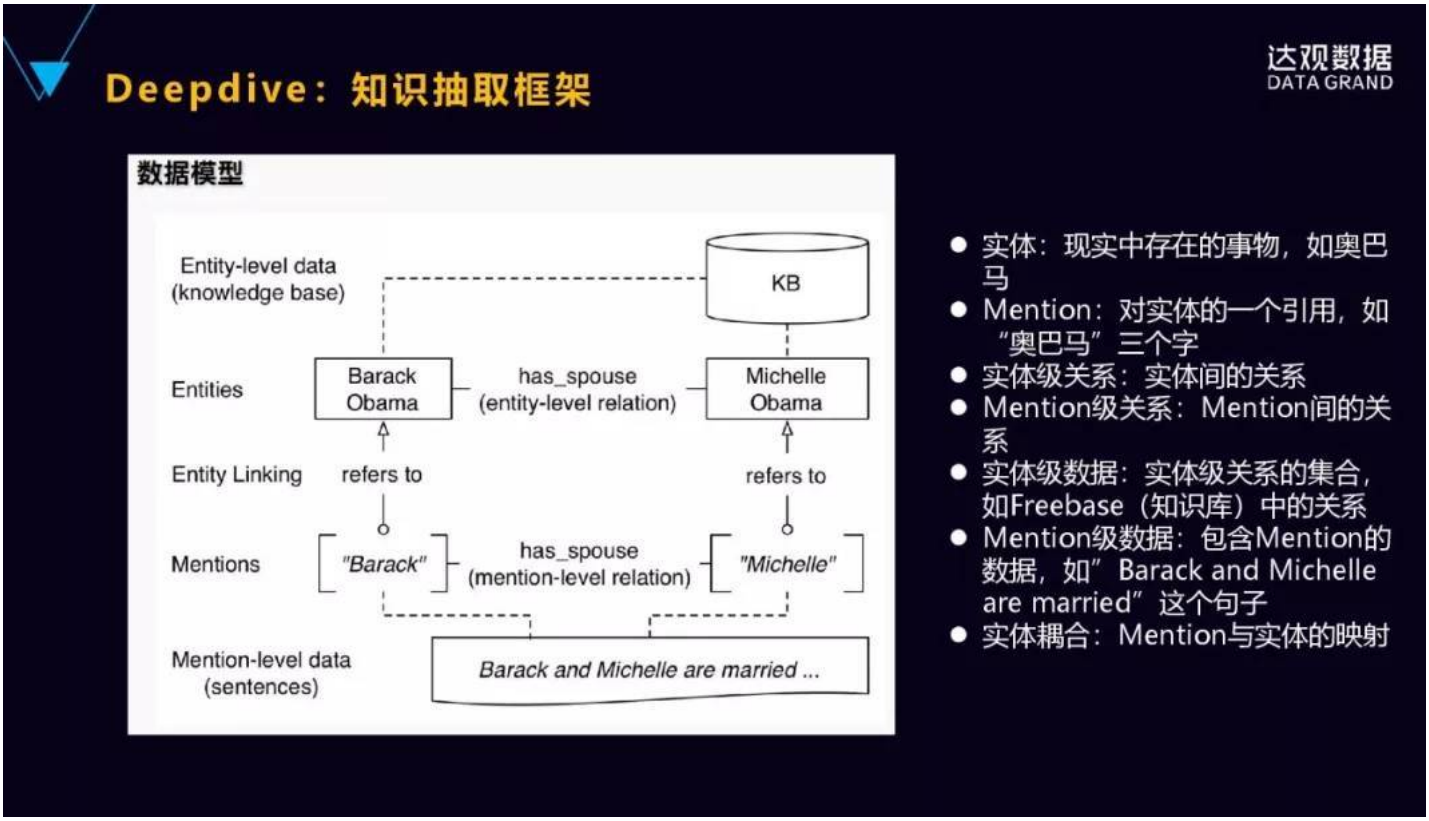
这三种方法当中，有监督的方法能够抽取出有效的特征，然后在准确率和召回率方面是更有优势的，半监督和无监督的方法一般情况下，效果都不是特别的好，所以业界现在
有监督的学习的方法。

我们刚刚提到有监督学习方法，比较困难的地方就是怎么样获取大量分类的训练样本，完全通过人工去标注的方式显然不是比较好的方式。

有什么样好的处理的方法？用远程监督的一种方法，典型的工具Deepdive，也是斯坦福大学InfoLab实验室开源的知识抽取的系统，通过弱监督学习的方法，从非结构化的文本
结构化的关系的数据。开发者不需要理解它里面的具体的算法，只要在概念层次进行思考基本的特征就可以了，然后也可以使用已有的领域知识进行推理，也能够对用户的反
进行实时反馈的一种机制，这样能够提高整个预测的质量。背后用的是也是一种远程监督的技术，只要少量的运训练的数据就可以了。

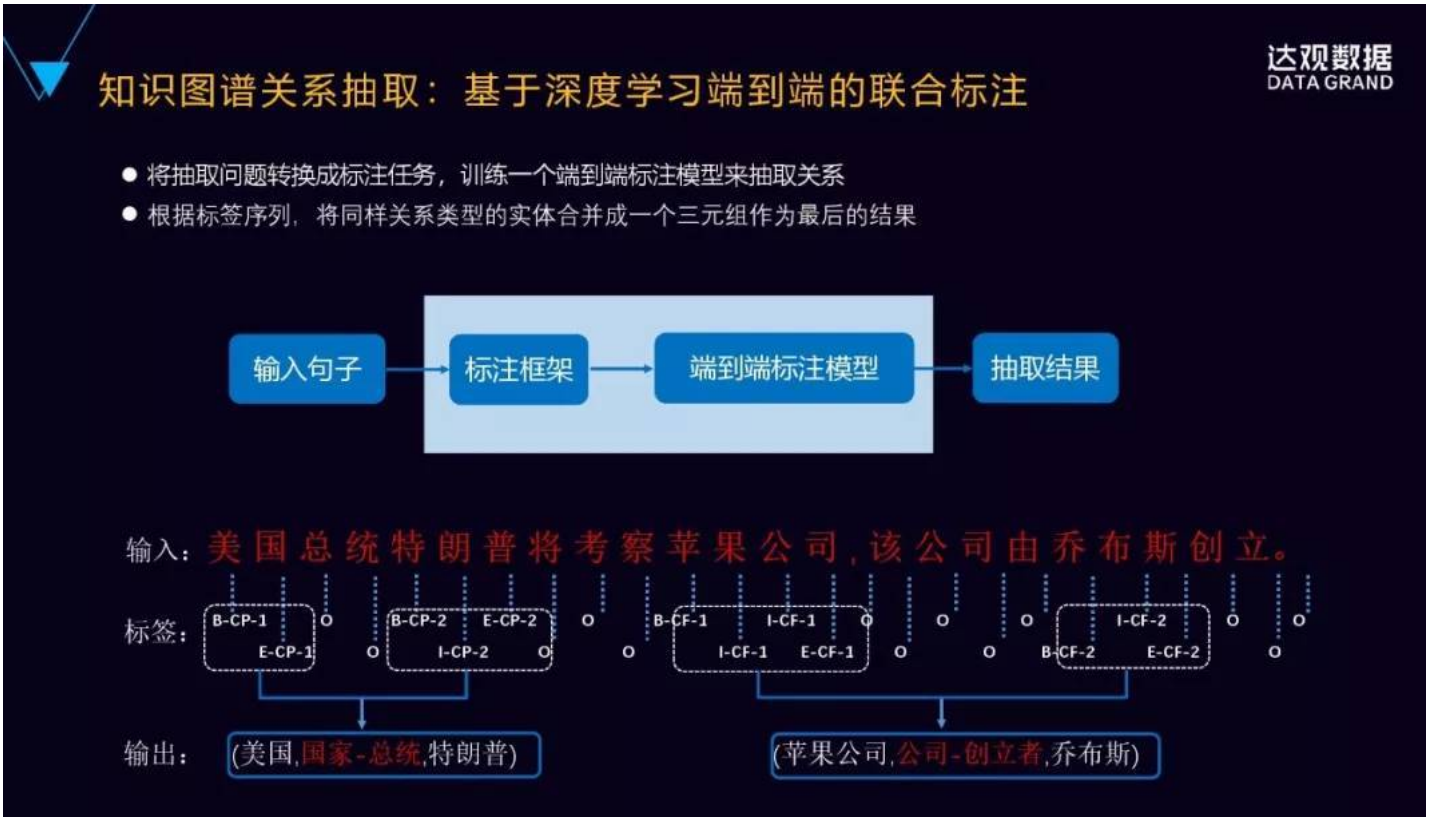


我们来具体来看一下它是怎么样来做这样一件事情的。首先，下面看到Mention的句子就是“奥巴马和米歇尔结婚”，它是句子，但Mention就是这些词的标记，奥巴马米歇尔都
要推测它之间是不是配偶的关系。这个时候要对Mention：奥巴马和米歇尔，去对应到知识图谱里面具体的实体，看一下这两个实体在知识图谱里面是不是存在着配偶的关系？
把它拿过来作为正的训练样本，如果不是，它就是负的样本。前提假设就是知识图谱里面的它的实体之间的关系都是正确的，以这个为依据，去做样本的标注。



目前进行实体关系抽取有两大类方法，有一类是基于流水线式的抽取，输入一个句子，首先抽取这实体，再对实体进行两两组合，然后再进行分类，最后识别出实体之间的关
缺点的，第一，它容易造成错误的传播，比如在第一步的时候，实体如果识别错误了，后面的关系肯定也是错误的。第二，会产生没有必要冗余的信息，因为要对实体进行两
进行关系的分类，很多配对之间实际上就没有这样的关系，就是它会带来了这样非常多的冗余的信息，错误率也会被放大、被提升。

现在用的比较多的另一种方法，是联合学习的方法，输入一个句子，通过实体识别和关系抽取的联合模型，可以直接得到有效的三元组。通常我们是基于神经网络的联合标注
面涉及到两个关键的工作，一个是模型的参数共享的问题，还有一个就是标注策略，怎么样进行有效的标注。模型共享是指的是在实体训练的时候能够进行实体识别和关系分
会通过反向传播来匹配来进行关系的分类，然后同时来实现这两个参数之间的依赖，两个子任务之间的依赖。参数共享的方法，它本质上还是两个子任务，只是说它们这两个
共享有了交互，而且在训练的时候还要预先进行实体识别，识别实体之后再根据预测信息进行两两匹配，来进行关系的分类，所以仍然会产生无关系时候冗余的信息存在。



现在我们新提出了一种端到端序列标注的策略，把原来涉及到序列标注和分类的两个任务变成了完全的端到端的序列标注的问题，通过端到端的神经网络模型，可以直接得到
实体关系的三元组。

新的标注策略是像下面这张图里面有三个部分来组成的，第一部分是实体中的词的位置，比如b是表示开始，i表示是在实体的内部，e表示是实体的结尾，s表示是单个的实体。
系类型的信息，预定义的关系类型的编码，比如里面的CP、CF，CP是国家总统，CF是公司创立者，这样两种的关系。还有实体的角色的信息，它表示是实体1还是实体2？
其它的这些字符都用O来表示，这样就进行了实体的标注。

知识融合：实体链接和实体合并

达观数据 DATA GRAND

- 实体对齐：旨在发现具有不同标识实体但却代表真实世界中同一对象的那些实体，并将这些实体归并为一个具有全局唯一标识的实体对象添加到知识图谱中。
- 实体对齐主要通过计算实例之间相似度：具有相同描述的实体可能代表同一实体（字符相似）；具有相同属性-值的实体可能代表相同对象（属性相似）；具有相同邻居的实体可能指向同一个对象
- 充分考虑数据源的可靠性以及不同信息在各个数据源中出现的频度等因素来决定最终选用哪个类别或哪个属性值。
- 利用来自如LOD (linked open data)中已有的人工对齐标注数据（使用owl:sameAs关联两个实体）可以作为训练数据学习发现更多相同的实体对。
- 无论何种自动化方法都无法保证100%的准确率，这些方法的产出结果将作为候选供人工进一步审核和过滤。

接下来我们讲一下实体的融合，这里最主要就是实体的对齐。

首先是实体对齐，也叫实体归一化，是把具有不同的标识的实体，但是在现实世界当中可能是表达同样意思的，把它做归一化。比如中华人民共和国、中国和China，这个三个体，尽管它的表述方式不一样，就把它归一化为具有全局唯一标识的实例对象，然后添加到知识图谱当中去。

现在实体对齐普遍采用的还是一种聚类的方法，关键在于定义合适的相似度的阈值，一般从三个维度来依次来考察的，首先会从字符的相似度的维度，基于的假设是具有相同能代表同实体。第二个维度，是从属性的相似度的维度来看的，就是具有相同属性的和以及属性词的这些实体，有可能会代表是相同的对象。第三个维度，是从结构相似度的假设是具有相同邻居的实体更有可能指向同对象。

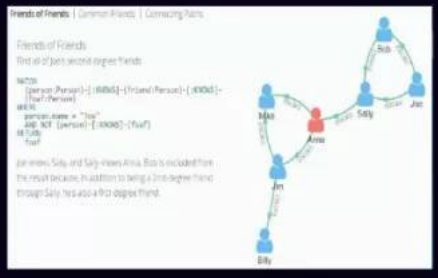
进行融合的时候要考虑各个数据源的数据的可靠性，以及在各个不同数据源当中出现的频度来综合决定选用哪个类别或者哪个属性词。还有一种方法就是是用来自LD（Linked多人工标记好的数据，是非常准确的，其中有种关联叫owl:sameAs，它表示前后两个是同实体的，利用这个作为训练数据来发现更多相同的实体对，是比较好的方法。最后要哪种方法，都不能保证百分之百的准确率，所以最后也要有人工审核和过滤。

知识存储：数据库选择

达观数据 DATA GRAND

数据库层面的选择有：图数据库、NoSQL数据库、关系数据库

- 若KG结构复杂，且关系复杂，连接多，建议使用图数据库，如Neo4J等
- 若KG侧重节点知识，关系简单，连接少，可以使用传统关系数据库
- 若考虑KG的性能、扩展性和分布式等，可以使用NoSQL数据库
- 根据实际情况，也可以多种数据库融合使用（ES和Neo4J）



知识存储，就是如何选择数据库，从选择层面，我们有图数据库，有NoSQL的数据库，也有关系型数据库，数据库有很多选择。具体什么样的情况下选择什么样的数据库？通谱的关系结构非常的复杂、关系非常的多，这时候建议使用这个图数据库，比如Neo4J这样的数据库。另外一种就是它的关系并不是很复杂，关系可能也就是1度、2度的关系常多的属性的数据，这个时候可以考虑关系式数据库，或者是ES这样的存储。如果要考虑到知识图谱的性能、可扩展性、可分布式，是可以结合NoSQL的数据库，比如TiTan况，一般是综合起来使用的，根据我们的经验，我们会结合Neo4J和ES来综合来使用，同时还结合这关系型数据库MySQL等等，根据不同的数据的特点来进行选型，而不是说据库。

知识存储：数据库选择

达观数据 DATA GRAND

	TiTan	Graph Engine	Neo4J
是否开源	是	是	是
License	Apache License 2.0	MIT	GPL（开源）、AGPL（商业）
平台	Linux	Windows	Windows/Linux/Mac OS
数据量级	千亿	百亿	百亿
查询语言	Gremlin	LINQ	Cypher
API	Java	C#	Java/Python/Ruby/JS/Go/Php/.Net/C++/Spring等
Java版本	1.8以上	不支持	1.8以上
存储后端	Cassandra/Hbase/Berkeley DB	RAM	嵌入式、基于磁盘的专有文件系统
分布式	支持	支持	支持，但较弱

顺便了解一下目前主流的几款图数据库，Titan、Graph Engine、Neo4J这个三个分别都是开源的，然后Titan是Apache旗下的，Graph Engine是MIT的License，Neo4J是GPL版，也有也有开源免费版。然后它们的平台，像Titan是Linux，Graph是windows。而数据的支撑量级，像Titan是后端存储，基于Cassandra/Hbase/BDB这样的分布式存储引擎的数据量，千亿级的数据量级；Neo4J商业版也可以支持到百亿级的，但是它的非商业版在数据量级比较大的时候，一般是在几千万级的时候就可能会出现一些问题。

知识推理：基于符号的推理

达观数据 DATA GRAND

RDF idea

- Use (directed) graphs as data model

- "Resource Description Framework"

RDFS:Class and Instance

- Classes: sets of instance
- Example：人工智能公司
- Classes can have hierarchy
  - Example: 人工智能公司是高科技公司

人工智能公司 subclass 高科技公司

RDFS:Reasoning

From

Google RDF:type 人工智能公司

and

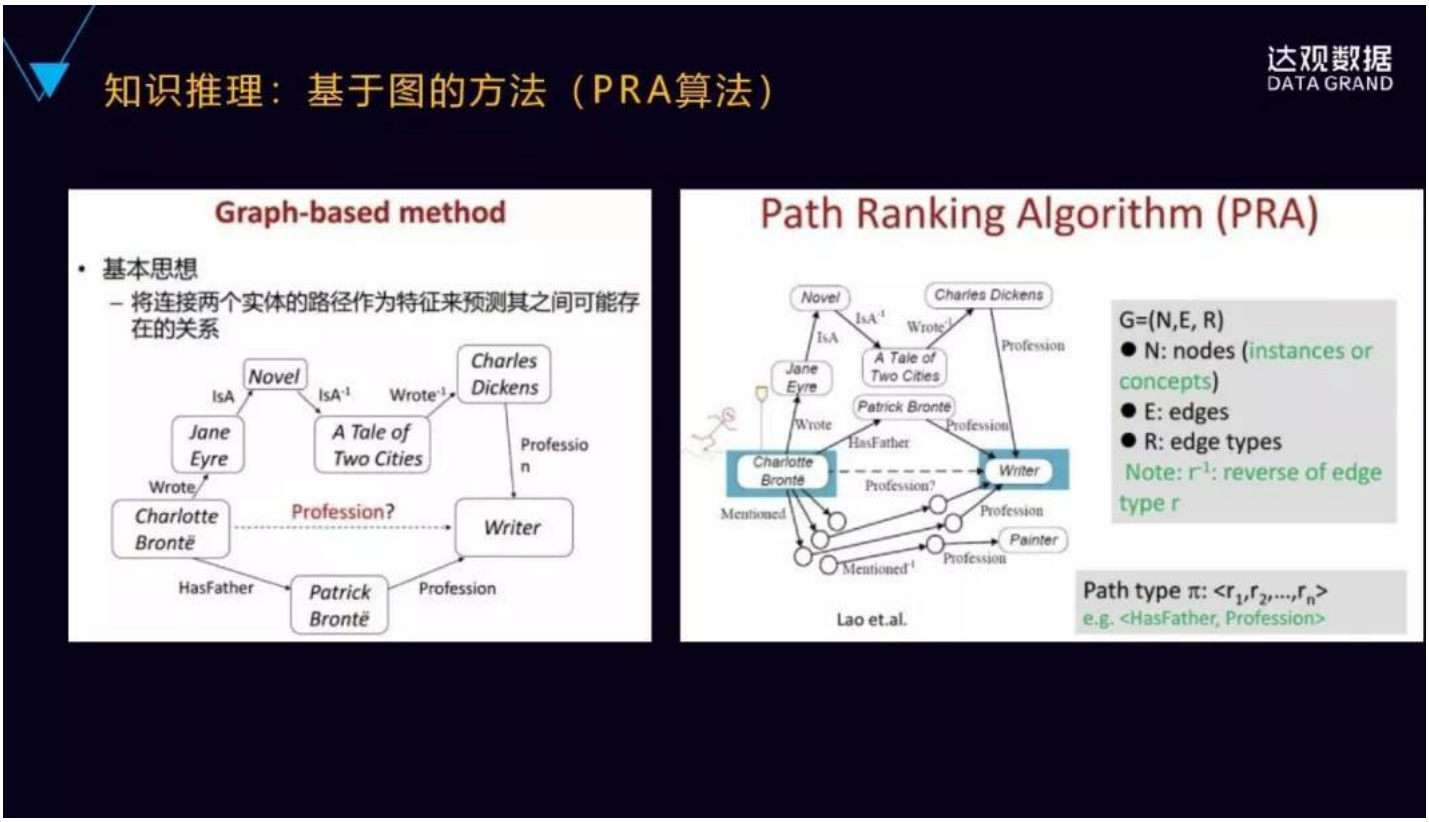
人工智能公司 subclass 高科技公司

we can infer

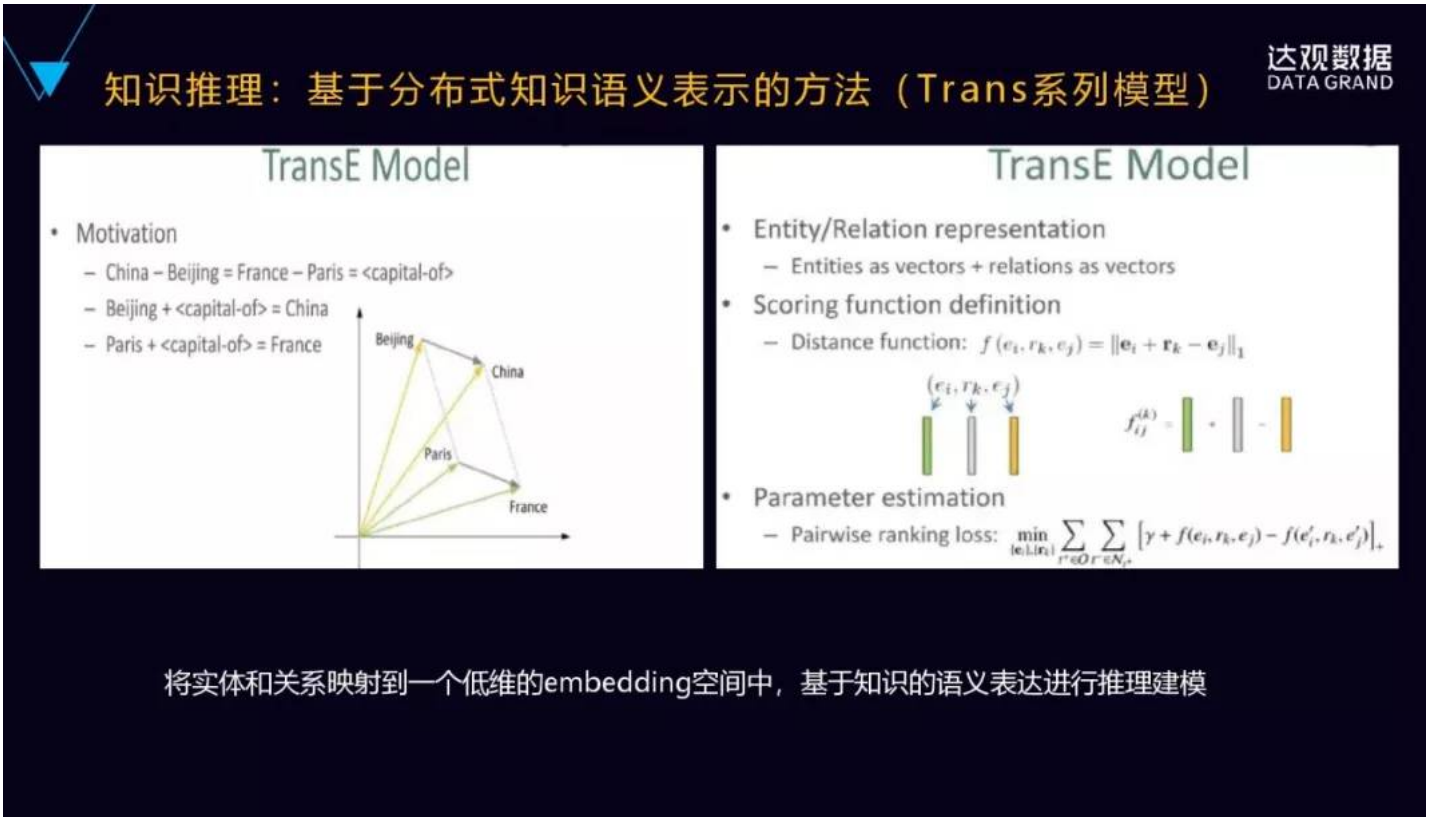
Google RDF:type 高科技公司

再看一下知识推理，知识推理这边有几种方法，首先是基于符号推理，我们上面说的三元组的结构，比如左边的RDF，有概念，然后基于概念符号进行推理。

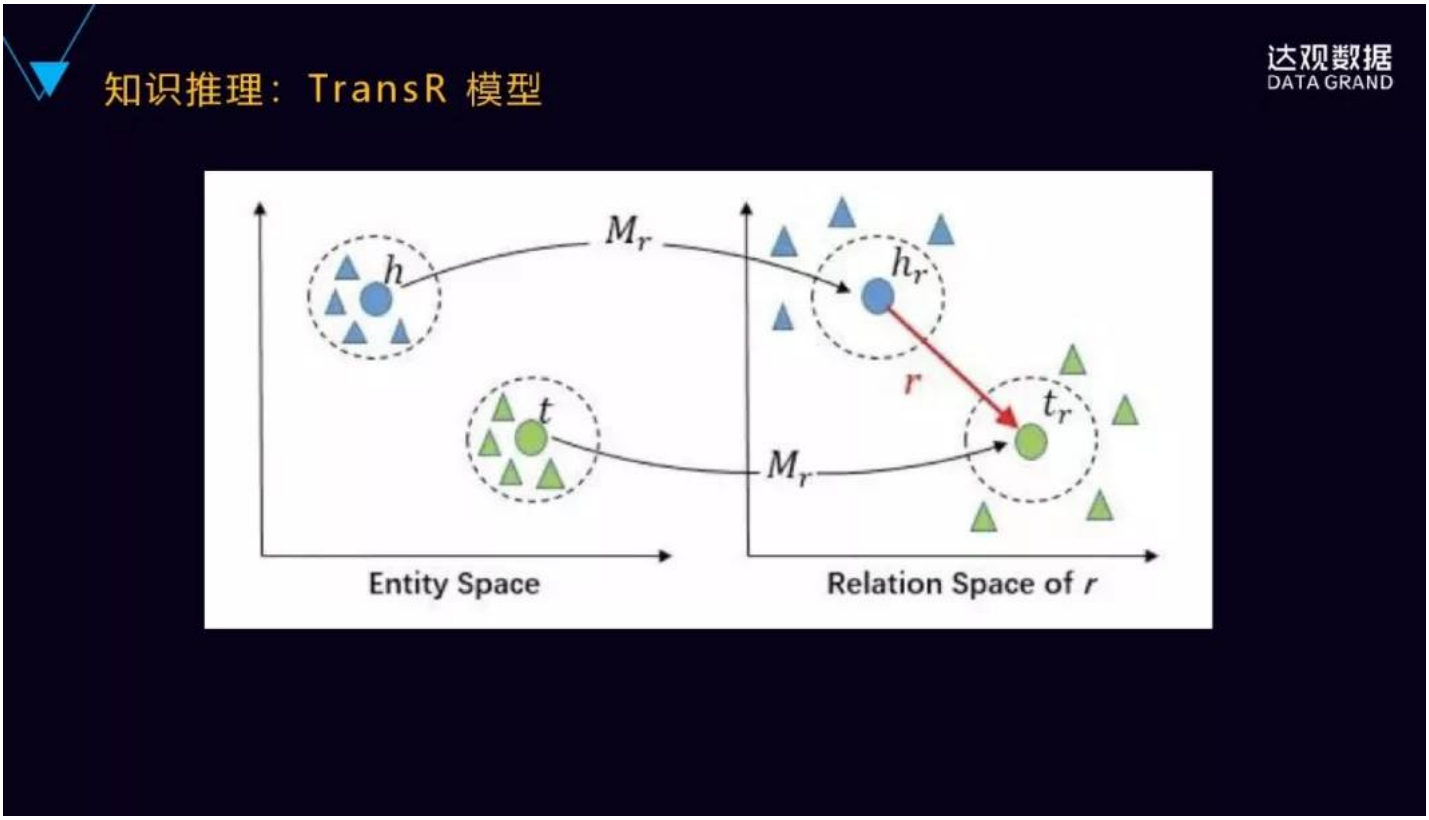
这个是基于OWL进行本体推理的例子，这个背后是基于OWL本体的推理，最常见的OWL推理工具是Jena, Jena 2支持基于规则的简单推理，它的推理机制支持将推理器(infere
入Jena，创建模型时将推理器与模型关联以实现推理。



还有一种是基于图（PRA）的推理的方法，更直观的一种方法，思想是比较简单的，就是以连接两个实体的已有路径作为特征构建分类器，来预测它们之间可能存在的潜在关
比如左边这个图里面Charlotte Bronte，我们要预测他的职业是不是作家。在这个图里面已知存在关系是什么呢？他写过一篇小说，它写过一篇小说Jane Eyre，然后双城记也是
斯是写了双城记这部小说，狄更斯是作家，同时它下面还有他的父亲职业也是作家，所以基于这样图之间的关系，就可以较大概率的推理出Charlotte Bronte的职业很有可能就
于图之间的关系的特征构建分类器来进行预测的。 PRA提取特征的方法主要有随机游走、广度优先和深度优先遍历，特征值计算方法有随机游走probability，路径出现/不出现
径的出现频次等。PRA方法的优点是直观、解释性好，但缺点也很明显，有三个主要缺点：首先，很难处理关系稀疏的数据，其次，很难处理低连通度的图，最后，是路径特
耗时。

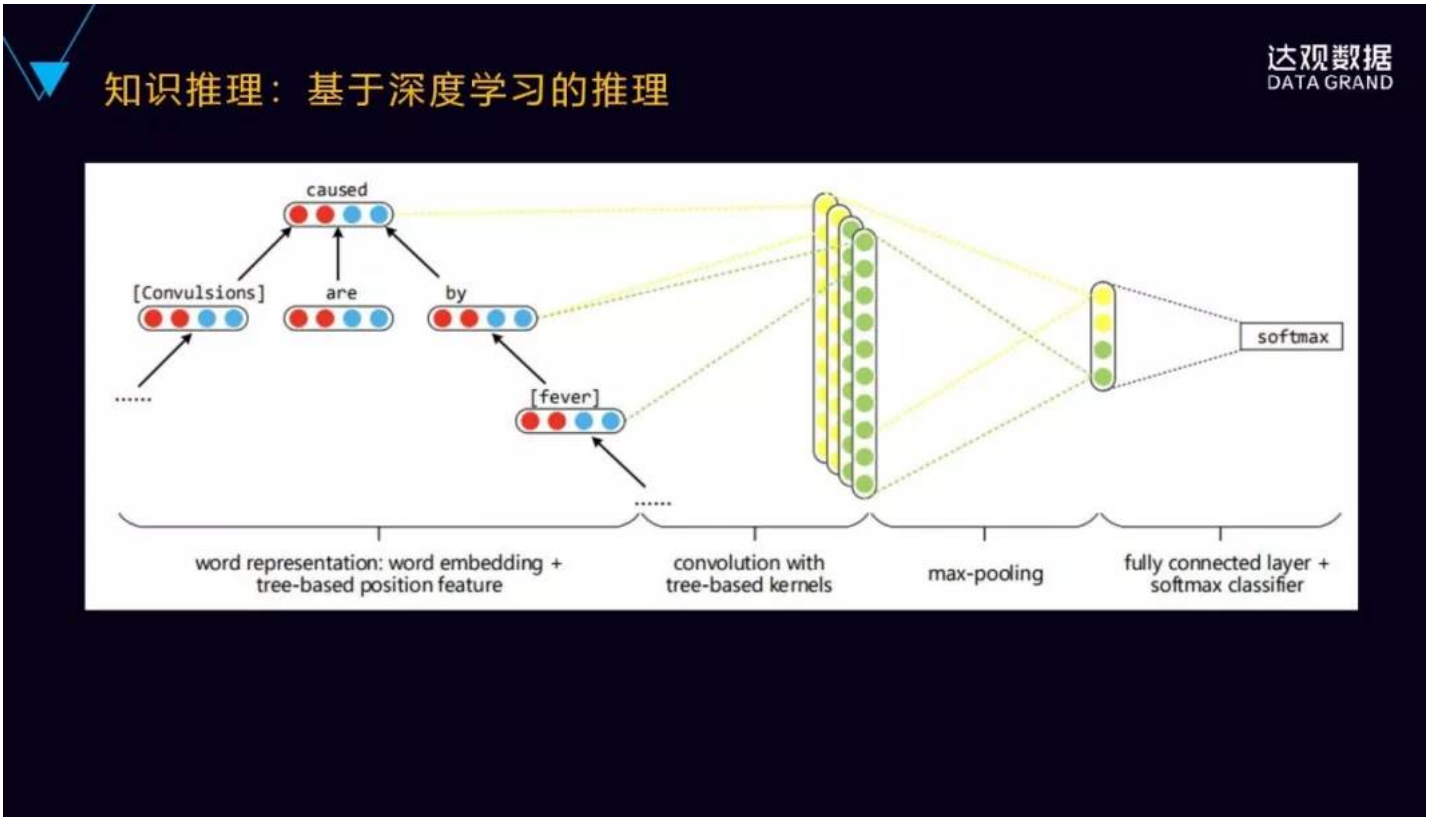


还有是基于分布式的知识语义表示的方法，比如像Trans系列的模型，在这个模型基础上进行语义的推理。TransE这个模型的思想也比较直观，它是将每个词表示成向量，然后
种类比的关系。比如上面这个图里面的北京中国，然后类比巴黎法国，就是北京加上首都的关系就等于中国，然后巴黎加上capital的关系等于France。所以它是无限的接近于
embed]ding。这个模型的特点是比较简单的，但是它只能处理实体之间一对一的关系，它不能处理多对一与多对多的关系。



后来提出了TransR的模型了，TransR实际上是解决了上面提到的一对多或者多对一、多对多的问题，它分别将实体和关系投射到不同的空间里面。一个实体的空间和一个关系
实体空间和关系空间来构建实体和关系的嵌入，就对于每个元组<h,r,t>首先将实体空间中实体通过Mr向关系空间进行投影得到hr和Tr，然后Hr加上r是不是约等于或者近似的等
关系空间里面的距离，来判断在实体空间里面，H和T之间是不是具有这样的关系？

除了TransE、TransR，还有更多的Trans系列的，像TransH、TransN、TransG等等这些模型。清华大学的自然语言处理实验室发布了一款这个叫openKE的平台，openKE它
学习的平台，是基于Tanserflow的工具包来开发的。它整合了Trans系列的很多算法，提供统一的接口。它也是面向了预训练的数据来表示模型的。



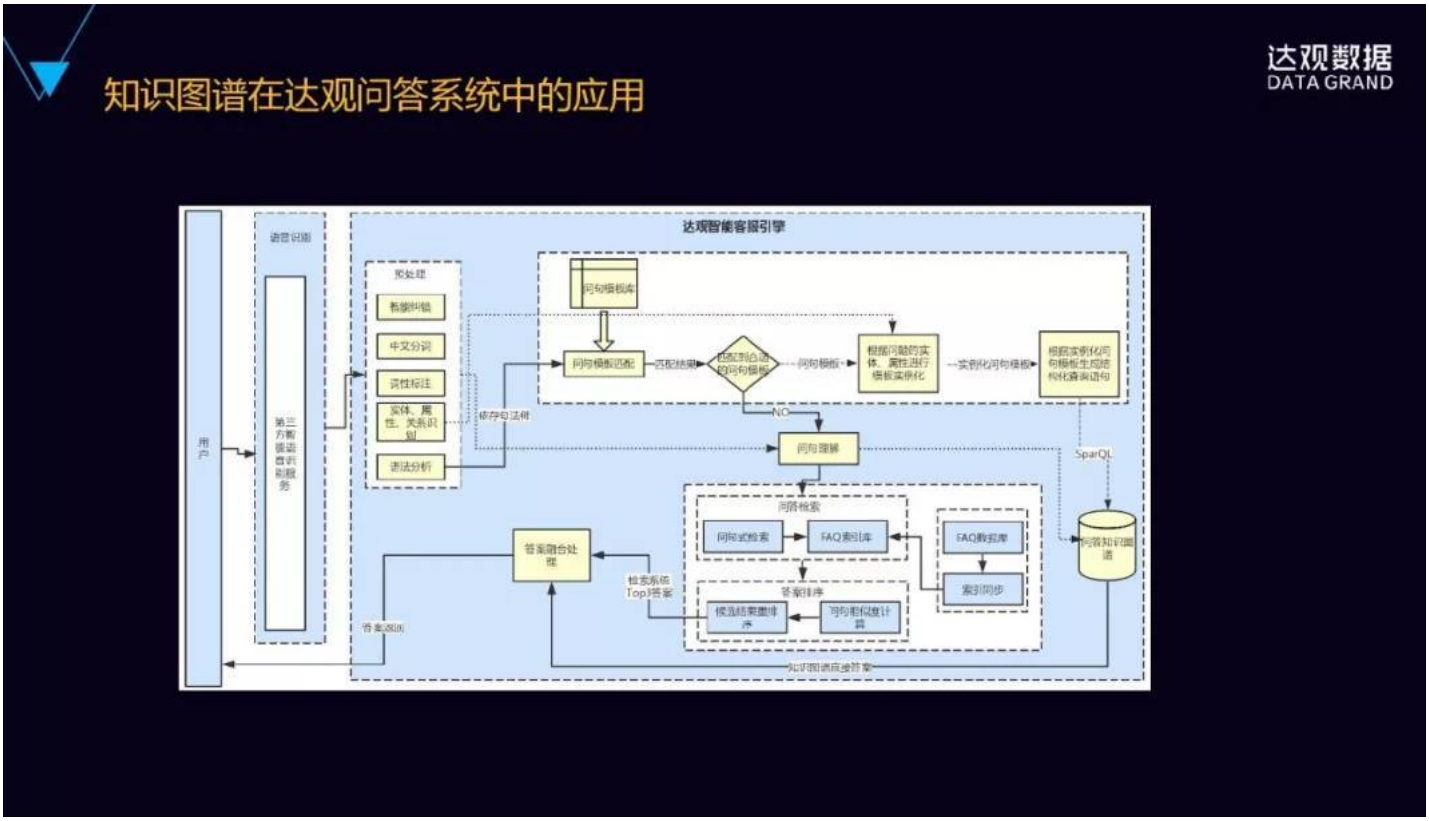
最后介绍一种基于深度学习的推理模型，这个模型利用了卷积神经网络对实体进行关系的分类的，把句子的依存树作为输入，就是将词在树中的不同的位置的嵌入式的表示拼
中来学习，同时对面相树结构设计了独特的卷积核。这种方法在实体分类的任务上，相较于未使用位置关系的信息，效果会有一定的提升。

四、达观经验与案例

1.知识图谱在达观的知识问答当中的应用

我们的智能问答是融合了是知识图谱问答和基于检索的问答两种方法进行融合。

首先，左边会把用户的语音经过语音转成文字以后，进行一个预处理，预处理主要是做了分词、纠错、词性标注、实体属性的识别，对这个句子进行依存句法树的结构分析。



预处理完了以后，引擎会首先尝试根据问句的句法结构进行问句模板的匹配，如果说能够匹配到合适的问句模板，这个时候再根据在预处理阶段得到的问句的实体属性和关系模板进行实例化，然后再根据实例化以后的问句模板来生成知识图谱的图数据库的查询语言，然后在图数据库里面就可以把这个答案查出来了。

另外一种情况是没有匹配到合适的问句模板，这个时候会进入到基于检索的问答模块，最后是把基于知识图谱和基于检索的两种的结果进行融合。

2. 在HR人岗精准匹配里面的应用

HR创建的JD能够自动的匹配到简历库里面最合适的候选人的简历，也即把JD的内容和简历库里面简历的内容做语义的匹配。

这里有一个问题，JD对技能的描述和不同的人的简历中对技能的描述存在很多表达方式造成的语义方面的差异。我们分别对JD和简历构建技能图谱，通过技能图谱的子图匹配地来解决语义匹配的问题。

我们在知识图谱建设当中的一些经验：

- 第一，界定好范围，就是要有一个明确的场景和问题的定义，不能说为了知识图谱而知识图谱。如果没有想清楚知识图谱有什么样的应用的场景，或者能解决什么样的图谱是比较难以落地的。一些明确的场景，比如解决商品数据的搜索问题，或者从产品说明书里面做相关问题的回答。
- 第二，做好schema的定义，就是上面讲到的对于schema或者本体的定义。第一步确定好场景和问题以后，就基于这样的场景或者问题，再进行相关领域的schema的定域里概念的层次结构、概念之间的关系的类型，这样做是确保整个知识图谱是比较严谨的，知识的准确性是比较可靠的。知识的模型的定义，或者schema的定义，大部这个领域的知识专家的参与，自上而下的方式去定义的。
- 第三，数据是知识图谱构建基础。数据的梳理就比较重要，最需要什么样的数据？依赖于我们要解决的问题是什么，或者我们的应用场景是什么？基于问题和场景，梳理题、相关的数据，包括结构化的数据、半结构化数据、无结构化的数据，结合百科跟这个领域相关的数据，领域的词典，或者领域专家的经验的规则。
- 第四，不要重复去造轮子，很多百科的数据和开放知识图谱的数据，是可以融合到我们的领域知识图谱中。
- 第五，要有验证和反馈机制，需要有管理后台，用户可以不断的和知识图谱系统进行交互，不断的进行确认和验证，确保知识图谱每一步推理和计算都是准确的。
- 第六，知识图谱构建是持续迭代的系统工程，不可能一蹴而就。

最后给大家介绍中文开放知识图谱，达观数据也是中文开放知识图谱发起单位之一，这里面有很多开放的数据和开源的工具，其中的文章也是非常好的学习材料。

下面是自由提问环节。

Q：用知识图谱来做反欺诈，和深度学习的方法相比，或者是其它机器学习的相比，知识图谱是否有优势？

A：反欺诈是整个风控流程中的一个非常重要的环节。其主要难点在于如何基于大数据把多个不同来源的数据（内部、外部，格式化、非格式化）有机整合在一起，并对这些数杂且动态变化的关系进行建模，从而构建起一个反欺诈引擎。知识图谱作为关系的直接表示方式，可以提供一种非常直观的可视化的手段以及内部的推理机制来有效地分析各存在的潜在风险。

我们知道深度学习的表示是基于一种低维稠密的数值向量，模型本身是一个“黑盒”，我们无从知道内部的各种特征的非线性组合是如何形成的。相比于知识图谱，深度学习的不的缺点。此外深度学习要真正达成效果，需要依赖已经标注过的大样本数据进行训练，但在很多领域（包括反欺诈）这样的数据获取门槛极高。

当然，深度学习和知识图谱也是不完全互斥的，知识图谱的构建中的实体、属性、关系抽取等关键过程也会用到深度学习的循环神经网络等自然语言处理技术。

Q：知识图谱的查询跟关系数据库的查询感觉都一样，这两个有什么样的关系？

A：在关系型数据库里面，如果要联合多个表执行复杂查询，特别在数据量比较大的情况下是非常慢的。如果涉及到多种很复杂的关联关系，图数据库是比较好的选择。图数据历，内置对图搜索的优化算法，可以非常快速进行子图的搜索。相较于传统的关系数据库，其性能会有多个数量级的提升。

Q：纯文本怎么进行抽取？

A：关于抽取，我在前面讲到了有很多的方法，包括实体抽取里面有NER的方法，NER有传统的CRF的方法，有基于循环神经网络+CRF的方法，也有端到端的联合标注的抽取同时输出实体和实体之间关系的三元组。

Q：知识图谱怎么和其它算法相结合，应用在搜索推荐上面？

A：在搜索里面，更多的是基于知识图谱去回答问题，它可以对query所表达的实体和属性进行识别，比如查询“华为P10手机的内存是多少?”这就是实体属性值的查找。华为P1P10这个实体的属性，对应到知识图谱里面，可以快速定位到这个实体叫P10，就可以直接把P10的内存（属性值）返回出来，就可以直接给出64G或32G这样的答案。

Q：实体有哪些属性是怎么定义的？

A： 从如何定义的角度看，主要有两种方法，一种是自上而下的方法，比如我们进行模式（Schema）定义的时候，首先会对其中的各种概念进行定义，概念有哪些属性，概念么样的。后面抽取的各个实体都会映射到模式层的一个概念上去，实体会自动继承所属概念的属性。还有一种是自下而上的方法，实体的属性是从原始数据中不断抽取提出来源的属性与属性值进行融合（归一化）。实际使用中，这两种方法往往是同时使用互相补充的。

Q：知识图谱中的时间和空间关系怎么表达？

A：知识图谱表达的是动态的数据，比如美国总统是特朗普对吧？现在是的，但三年前不是，五年后也不一定是。随着时间的推移，它的三元组的关系是会有变化的，那这个时达？通常是在三元组的基础上再扩展一个维度形成四元组这样的做法，但是它会大大增加系统处理的复杂度。也有一些其他的方法，比如可以是跟关系数据库相结合的方式进

Q：如何衡量一个知识图谱建立以后的效果？如何有效的进行学习改进？

A：我们建立一个知识图谱，建立一个机器学习的算法模型，对它的效果进行衡量度量是非常关键的。在知识图谱这里，一定是基于某个场景具体的应用，然后再看这个场景应用使用了知识图谱以后跟原来的做法来做一个比较，或者有一个管理后台，人可以定期对知识图谱系统的推理和计算的结果做一些检测和验证，然后对这些验证的结果再反馈去，让系统重新来进行学习、改进，建立人跟系统之间不断的反馈、验证和自动学习的过程。

Q：可不可以用知识图谱进行分类？

A：这个问题很有意思。上面提到，一方面我们可以基于分类的方式来做知识图谱中实体关系的学习。另一方面，我们是否可以利用知识图谱辅助进行文本的分类？这样方面我对于文本分类中的特征工程，我们的做法是通过知识图谱进行语义的关联扩充以获取更多的语义特征，尤其是对于短文本，对分类效果能有比较好的提升。

< 上一篇 MultiVAC：基于可信分片计算的弹性高性…

下一篇 从骁龙845到5G时代，高通

推荐博客

【译文】移动端的输入

Python数据可视化编程实战（八）：为项目设置matplotlib参数

理解Binder通信原理及常见问题3

让机器读懂用户--大数据中的用户画像

Android权限管理原理（4.3-6.0）上篇

【大数据之数据仓库】kudu性能测试报告分析

自备干货！如何有效的做竞品迭代分析

Spring Data Rest实战

知物由学 | 内容平台、社交媒体如何应对虚假新闻?

【重磅发布】最风骚的走位，最撩人的峰会，裂变！变！变！变！变！变！抢！