

年货

No.1

HAPPY NEW YEAR

阅读

是一种生活方式。

技术文章合集：NLP系列

255页电子书
19个NLP相关话题
17.5M
84,500字

诸年大吉，诸事顺利

序

春节已到，年味浓烈。

到了我们献上技术年货的时候。

不久前，我们已经给大家分享了技术沙龙资料合集，汇集了过去一年我们线上线下技术沙龙

192 位嘉宾，194 个话题，160+ 小时分享。

陆续出场的，同样重磅——DataFun 社区技术文章合集。

2018 年，是 DataFun 社区正式成立的第一个完整年，微信公号（datafuntalk）的关注数

也超过了 **12,000** 人，社区用户大数据+算法工程师从业者占比超 **8** 成，年累计阅读 **280,000+** 次。

由衷地感谢大家一直以来对我们的鼓励和陪伴！

在 2019 年春节到来之际，我们再次整理了社区的系列回顾技术干货文章，制作成电子书呈

送给大家。

电子书主要包括 NLP、广告、搜索、推荐、CV、Fin Tech、大数据、HBase 等 8 个板块。

文章为社区小伙伴整理，难免出现错误，大家在阅读中如果发现 Bug、问题，欢迎扫描下

方二维码，通过微信公众号与我们交流。

也欢迎大家转给有相同兴趣的同事、朋友，一起切磋，共同成长。

最后祝大家，新春快乐，阖家幸福：



目录

智能写作：人工智能为媒体内容创作赋能.....	4
从语言智能到行业智能.....	19
强化学习在自然语言处理中的应用.....	29
旅游知识图谱的构建和应用.....	44
NLP 在网络文学领域的应用.....	51
让机器读懂人类：揭秘机器阅读理解技术及应用.....	64
自然语言处理中的多任务学习.....	71
金融知识图谱的应用与探索.....	103
猎户星空 NLP 技术进展及产品应用.....	119
智变中的美团客服.....	138
二手电商知识图谱构建以及在价格模型中的应用.....	152
阿里神马智能对话问答.....	164
对话机器人在瓜子的实践.....	171
五八同城智能客服系统“帮帮”技术揭秘.....	181
知识图谱在贝壳找房的从 0 到 1 实践.....	203
人机交互技术介绍.....	214
多轮对话提升自动化流程服务.....	221
开源节流的智能导购对话机器人实践.....	235
音乐垂域的自然语言理解.....	246

智能写作：人工智能为媒体内容创作赋能

作者：彭卫华 整理：马宇峰

百度知识图谱致力于构建最大最全最好的中文知识图谱，汇聚知识，连接万物。通过知识映射真实世界、理解世界，让复杂的世界更简单。今天我主要分享知识图谱部智能写作方向的相关研究工作和应用实践。

近几年国内外的各大科技公司与媒体公司都纷纷布局智能写作，例如国外的美联社，国内的新华社，技术公司 BAT 等等。为什么智能写作如此受到关注，它能为媒体内容创作带来什么样的价值，下面开始我们的分享。

背景：

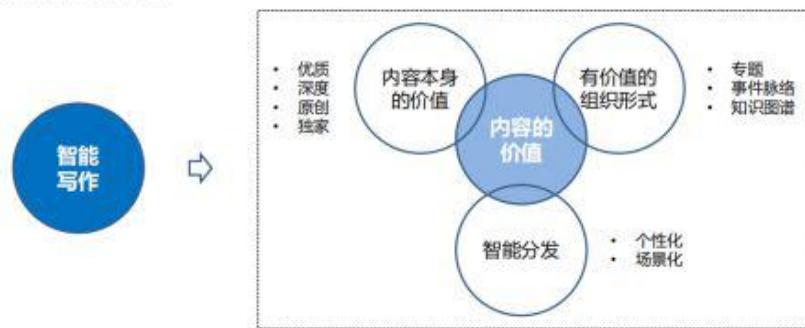
随着科技的发展，人工智能已经进入到认知阶段，AI 不仅仅被认为是一种算法、平台、解决方案，也是一种生态和生产力，可以大大推动传统产业的进步并改造它。

从最初的运算智能，到后期的感知智能（人脸识别、语音识别），再到当前探索的认知智能，有了长足的发展。然而，机器目前还无法与人类一样理解思考，也无法无中生有创造出新的知识，所以目前我们仍处在弱人工智能阶段。尽管如此，AI 已经展示出其强大的生产力，并已经涉足到我们的各个生活场景中，包括智能搜索、智能推荐、智慧医疗等，也包括我今天要介绍的智能写作。

写作任务大概可以从“采集、构思、表述”三个阶段来描述，无论是侧重于权威性的机构媒体，还是拥有独特风格的自媒体，都一直饱受创作效率的困扰。受限于主题选材、写作过程中出现的敏感词、错别字等因素，创作内容的成本一直居高不下。在自媒体领域，由于消费者注意力的马太效应，部分自媒体创作者逐利而去蹭热点，导致中长尾内容不足。例如文章配图，一些创作者找图片是直接在百度图片里面搜索，再选择粘贴到文章中，这样的创作效率是极为低下的，消费者的马太效应对整个内容的中长尾生态也影响较大，长期来看损害内容与流量的生态价值。

为什么需要智能写作 ?

- 当前媒体内容创作问题
- 内容的价值与智能写作



从内容价值的角度出发，可以简单理解为：

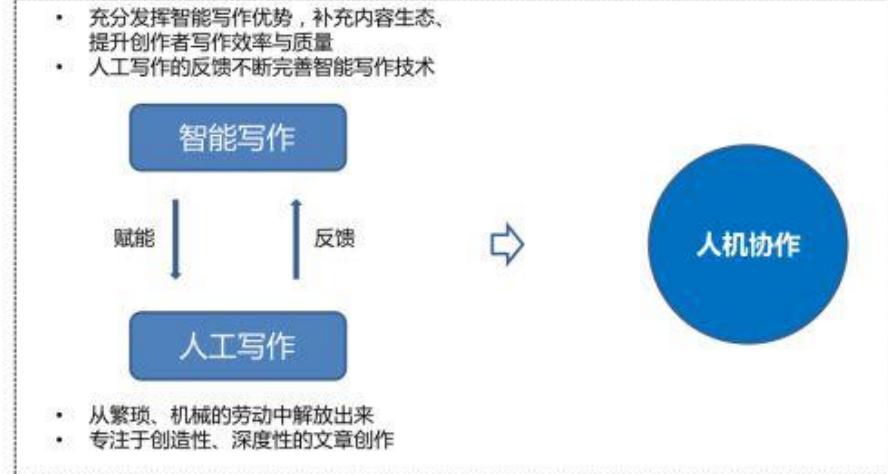
1. 内容本身的价值 (质量、深度等)；
2. 有价值的组织形式 (专题、脉络、知识图谱等)；
3. 内容的智能分发 (个性化、场景化)。

针对这三种场景，智能写作均可以发挥它的作用。创作过程中，可以提升效率；组织过程中，可以自动化组织；智能分发中，可以应用动态内容生成的技术，让用户对分发的内容更感兴趣。

现阶段，智能写作相比于人类，还有很大差距，人类擅长进行长文本、情感类的文章写作，写出高质量的有个性的文章。智能写作在信息与数据的处理上更占优势，可以大大提升聚合、时效类文章的创作效率。

在内容创作上以人机协作方式，智能写作与人工写作形成优势互补

- 充分发挥智能写作优势，补充内容生态、提升创作者写作效率与质量
- 人工写作的反馈不断完善智能写作技术



智能写作可以通过人机协作的方式，有效地将智能的效率优势与人工的创造性、深度性结合起来，降低人工繁琐、机械的劳动，不断补充优化内容生态。



从技术布局上来讲，主要分为两个部分，基础技术部分与智能写作部分。其中基础技术包括语言理解与生成、素材清洗与检索、知识认知与话题挖掘，并且需要有一定的质控保证。语言部分的技术是核心，延展开非文本类数据之外，是多模理解与多模生成。智能写作部分主要包括自动写作与辅助写作，前者主要用于数据写作、聚合写作，后者主要体现在创意激发、素材推荐、质量评估等场景上。

自动写作部分

快讯类

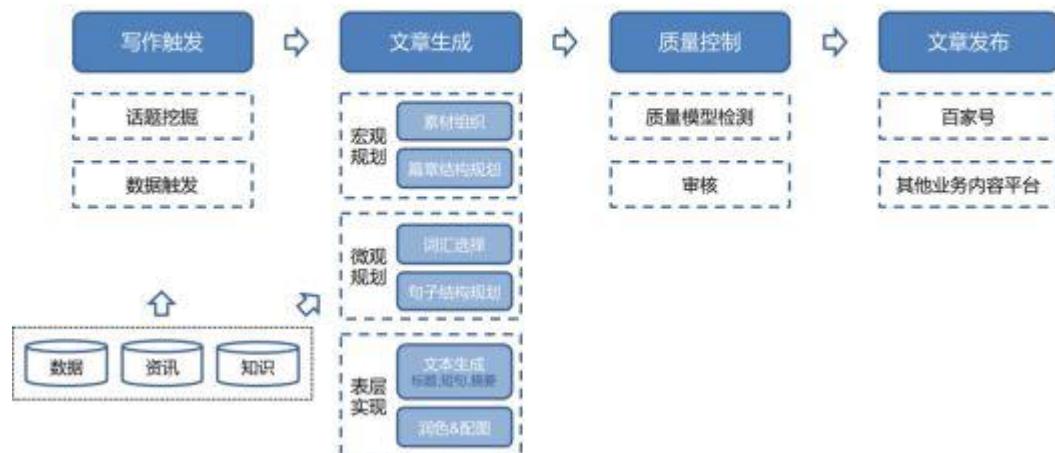
体育战报	财经快讯
中超联赛第27轮战报：上海上港 VS 山东鲁能 全场以比分4:2结束 小组赛单场最佳：胡尔克（上海上港） 上海上港 VS 山东鲁能 于10月28日19:30举行 全场比分为4:2结束	【6月20日】富达电子下午收盘价26.56元。涨幅9.09%！ 股市行情 6月20日，富达电子(602007)在北京时间下午 15:00:00 收盘，报 26.56 元，昨日 24.35 元。今开 24.4 元。总市值达到 106.27 亿元。流通市值 10.65 亿，市盈率 62.67%，市净率 7.52%，涨幅 +8.62%，成交量 5.29 万手，其中 外盘 2.26 万手，内盘 3.01 万手。成交额为 1.36 亿，每股收益 0.11 元，每股净资产 3.53 元，总股本 4.81 亿。流通股数 2.45 亿。 过去5日内该股资金总体呈流出状态，属于行业平均水平。5日共流出-1882.65 万元。根据统计，主力持仓占比10日前减少，已没有控盘。 龙虎榜日跌幅偏离值达7%的证券：日跌幅达20%的证券：日换手率达20%的证券。 【本日累计上榜5次】。买入金额最大的前5名，买入总计1832.95万元。占总成交比例4.7%；卖出金额最大的前5名，卖出总计3118.09万元，占总成交比例8.02%。买卖净差

聚合类

实体聚合文章	资讯聚合文章
你去过环境舒适的太平天国墟王府吗？江苏宜兴还有更多的景点值得游览 江苏宜兴还有更多的景点值得游览 太平天国墟王府 2018-10-29 10:00:00	每日科技_人工智能资讯精选小度在家一呼百应：人工智能的“国家队”板凳坐十年冷 每日科技_人工智能资讯精选 1. 小度在家一呼百应：人工智能家用化趋势明显 3月26日下午，百度在北京举办“百度不如见”发布会，发布国内首款智能音箱“小度在家”，而要创始人、董事长兼CEO李彦宏，百度度秘事业会总经理景鲲。小度在家CEO宋海涛出席，共同见证此次人工智能走向家用的标志性产品的诞生。该款搭载百度DuerOS系统，以自然语言实现跨终端对话的语音交互，并为用户提供了智能家居、看视频、美颜视频通话、百科搜索、生动画动等更好的AI体验。进一步实现了人工智能的家用化。百度作为“国家队”加持DuerOS，为AI技术落地到现实生活提供了先行条件。此次联合小度在家推出的首款智能音箱“小度在家”，也充分诠释百度在助力AI技术家用化

自动写作服务于内容生态，并已经在百家号、阿拉丁、地图等多个业务场景下落地。从产出的文章类型来看，主要分为快讯类、聚合类，此外还会包括科普类、视频转写的一些内容。

自动写作流程

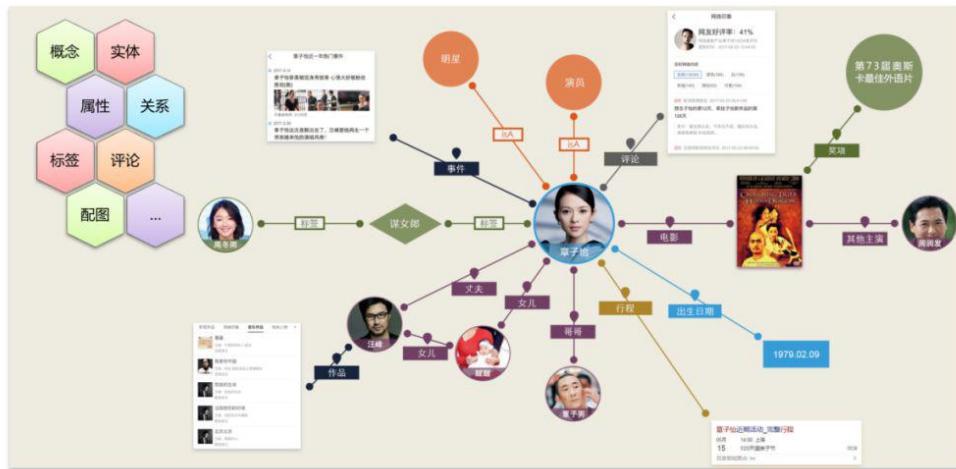


从写作的流程来看，首先是写作触发，接着文章生成，然后是质量控制，最后是文章发布。其中最开始的是写作触发，具体包括热点发现、主题分析、观点分析等，以生成满足用户需求的文章。此外最重要的部分是文章生成，分为下面三个部分：

1. 宏观规划，具体包括素材组织与篇章结构规划；
2. 微观规划，具体包括词汇选择与句子结构规划；
3. 表层实现，具体包括文本生成与润色配图等。

素材组织依赖于知识驱动产生的主题关联，文本生成则依赖于自然语言生成，结合通识知识图谱与行业知识图谱，以及包含事件等因素的复杂知识图谱，来完成文本到文本、数据到文本，以及多模到文本的文章生成。

知识图谱 - 知识的汇集、整理、以及再加工，基于语义的链接

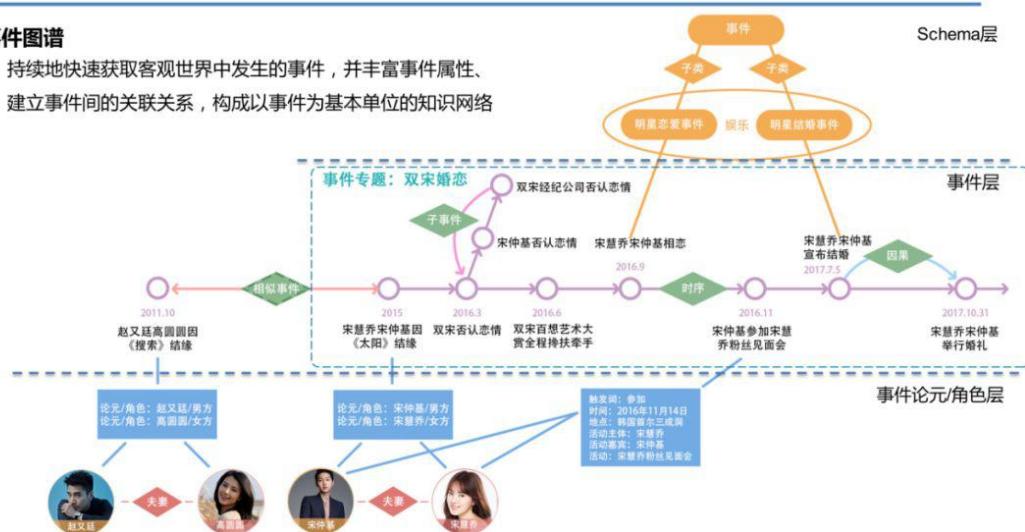


在深入写作关键技术之前，首先我们探讨下知识图谱的定义。简单来说，知识图谱就是知识的汇集、整理以及再加工，图谱中的每条边，均是基于语义的链接，是一个极其复杂的知识语义网络。事件图谱与传统知识图谱完全不一样，可以持续地动态地获取客观世界的事件，并丰富事件属性、建立事件间关联关系，构成以事件为基本单位的知识网络。目前百度知识图谱数据包含亿级别实体以及千亿级别的事实，以专家权威、百科实体、垂类挖掘与全网属性挖掘为组成部分，可以做到高时效性的秒级更新，在智能写作中扮演着核心角色，贯穿智能写作的全部流程。

关键技术 – 知识图谱

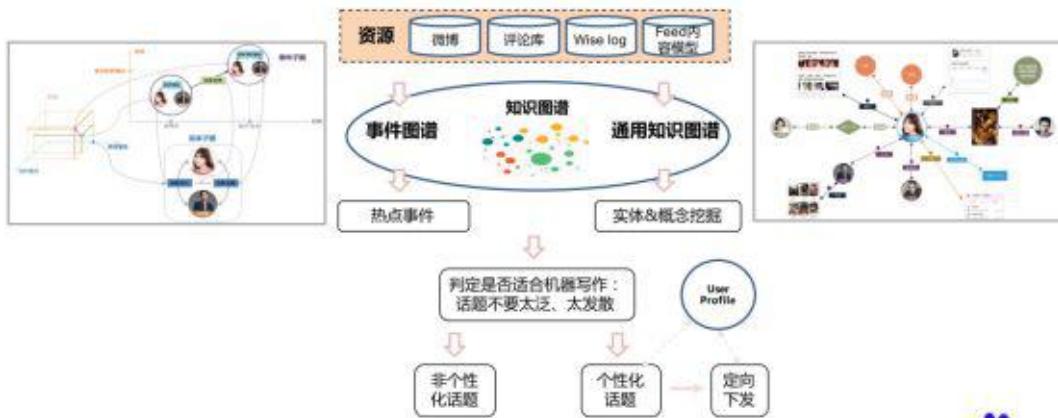
事件图谱

- 持续地快速获取客观世界中发生的事件，并丰富事件属性、建立事件间的关联关系，构成以事件为基本单位的知识网络



关键技术 – 话题挖掘

确定适合机器写作且有用户需求的文章主题



下面简单介绍下话题挖掘 ,话题挖掘是指挖掘提取出用户有需要的、且适合机器写作的主题。首先从微博、feed 内容等资源中，通过知识图谱提取、匹配出热点事件与概念，接下来判断是否适合机器写作，过滤掉太发散、太泛的话题；生成的话题包括非个性化与个性化的话题，其中个性化话题是通过用户画像进行定向下发。

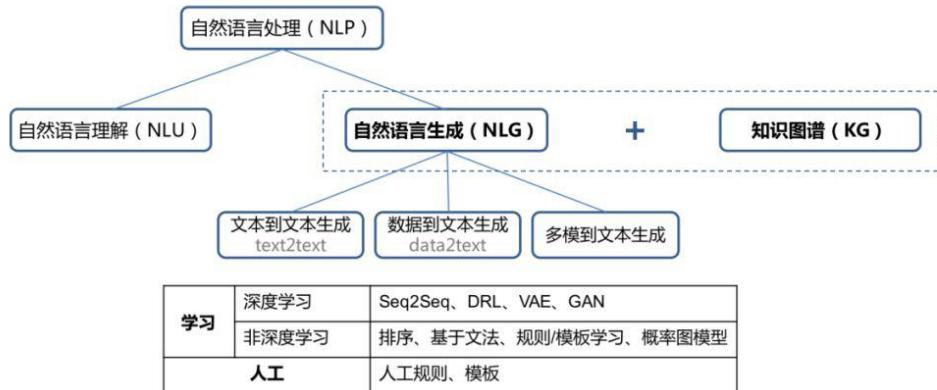
关键技术 – 素材组织

以财经类写作为例



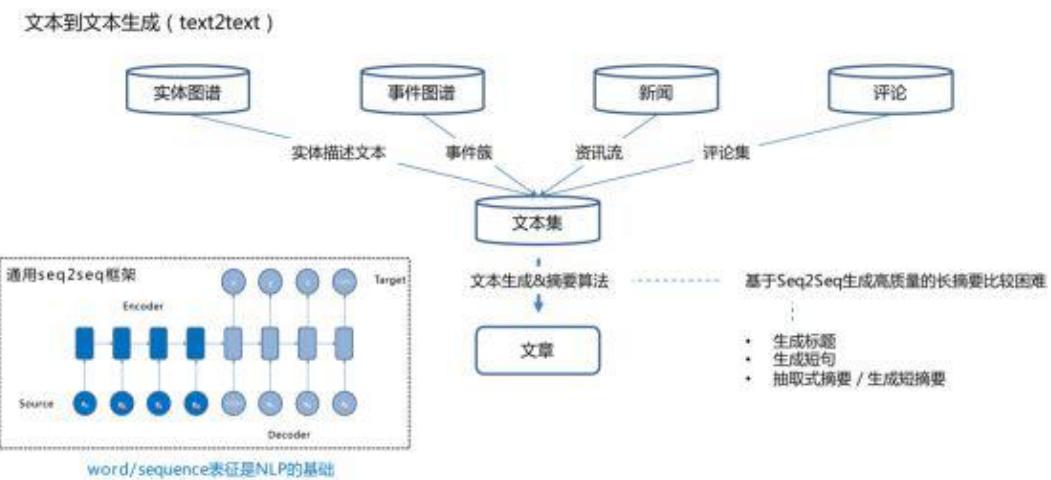
第二主要介绍素材组织。以财经类写作为例，传统做法是首先挖掘写作主题与相关关键词，通过检索关联素材，嵌入人工模板中，得到的文章往往零散而逻辑性不强，浮于浅层。在我们的做法中，主要通过知识图谱来驱动，通过事件触发，匹配财经、市场、板块等领域素材，进一步融合理论知识与权威评论，考虑一些归纳分析等方法，得到最终的素材关联与组织形式。

关键技术 – 文本生成



第三个关键技术是文本生成。文本生成的关键技术主要是自然语言生成 (NLG) 与知识图谱 (KG)。自然语言理解(NLU)与自然语言生成(NLG)是我们常用的自然语言处理(NLP)的两个主要方向。NLG 主要包含 text2text、data2text、多模到文本三种形式，考虑知识图谱作为先验知识进行相关生成。从人工方案角度讲，主要有人工规则与模板两种。从机器学习方法上来讲，深度学习方向主要包含：seq2seq、DRL、VAE、GAN 等相关技术，非深度学习技术方向包括：排序、基于文法、规则/模板学习、概率图模型等。

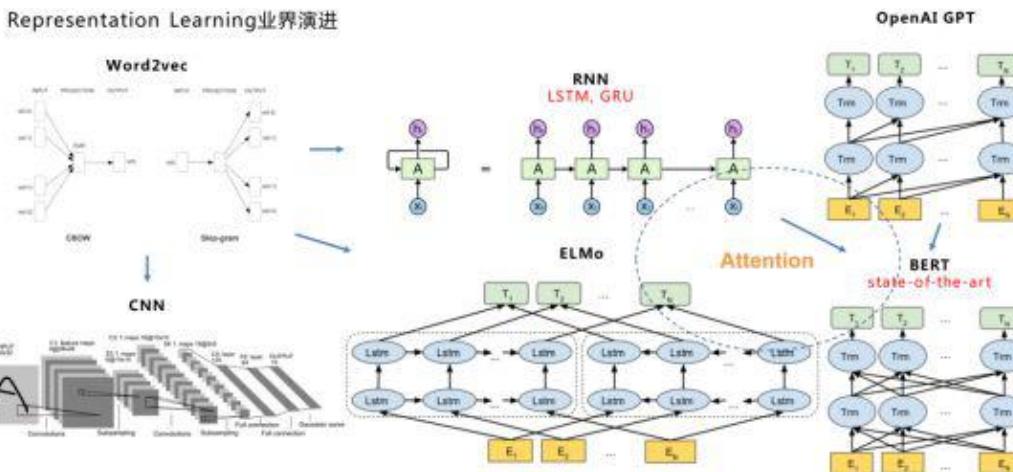
关键技术 – 文本生成



下面详细介绍文本生成的相关技术，主要是 text2text 的形式。首先通过实体图谱、事件图谱、行文、评论集等数据源获取文本集，接下来通过文本生成和摘要算法获取相应的文章。当然基于 seq2seq 的方法生成高质量的长摘要比较困难，但可以生成短句与标题。也可以通过抽取式摘要的方法，生成相关短摘要内容。可以看到 seq2seq 主要依赖于 encoder 与

decoder 两个步骤，贯穿其中的是 sequence 的表征，学习这种表征的方法我们称之为表示学习。

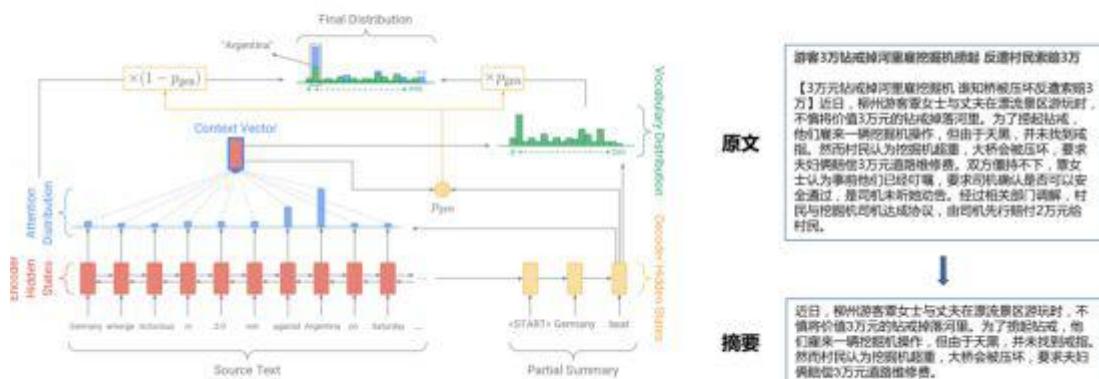
关键技术 – 文本生成



表征学习在近几年得到非常快速的演进。比较早期的 word2vec 模型，它可以有效地计算单词之间的语义相似度，但由于是词袋模型丢失了词的依赖关联关系；CNN 模型可以局部建模词的依赖关系，但无法解决长距离依赖问题，应运而生的是 RNN 模型，以及配套的 LSTM、GRU 方法。去年一个重大的突破就是 ELMo，提出解决一词多义的问题，突破了 word2vec 只有单一 embedding 的限制。然而基于 RNN 的方法其并行化做得不够，并且各种基于 RNN 的改进方案均无法表现出类似于人类的注意力感知机制，后续就诞生了 transformer 方法，诞生了 GPT 模型，然而其只考虑了单向学习。最终的集大成者是 BERT 模型，考虑了一些巧妙的创新，融合了前面的各种改进，得到了当前最佳的表示学习模型(计算复杂度相当高)。

关键技术 – 文本生成（探索）

自动文摘 - Pointer-Generator Model



在seq2seq+attention模型的基础上引入pointer network机制构造出了新的文本摘要模型。这个模型既能够从源文本中选择复制单词，同时还保留从固定词汇集中生成单词的能力。

原文

游客3万站成排河里推挖掘机堵坝 反遭村民索赔3万
【3万元站成排河里推挖掘机堵坝桥被压坏反遭索赔3万】近日，柳州游客蒋女士与丈夫在漂流景区游玩时，不慎将价值3万元的挖掘机掉落河里。为了捞起挖掘机，他们雇来一辆挖掘机操作。但由于天黑，并未找到挖掘机。然而村民认为挖掘机堵塞河道，大桥会被压坏，要求夫妇赔偿3万元道路维修费。双方僵持不下，蒋女士认为事情他们已经可管，要求司机确认是否可以安全通过，是司机未听她劝告。经过相关部门调解，村民与挖掘机司机达成协议，由司机先行赔付2万元给村民。

摘要

近日，柳州游客蒋女士与丈夫在漂流景区游玩时，不慎将价值3万元的挖掘机掉落河里。为了捞起挖掘机，他们雇来一辆挖掘机操作。但由于天黑，并未找到挖掘机。然而村民认为挖掘机堵塞河道，大桥会被压坏，要求夫妇赔偿3万元道路维修费。

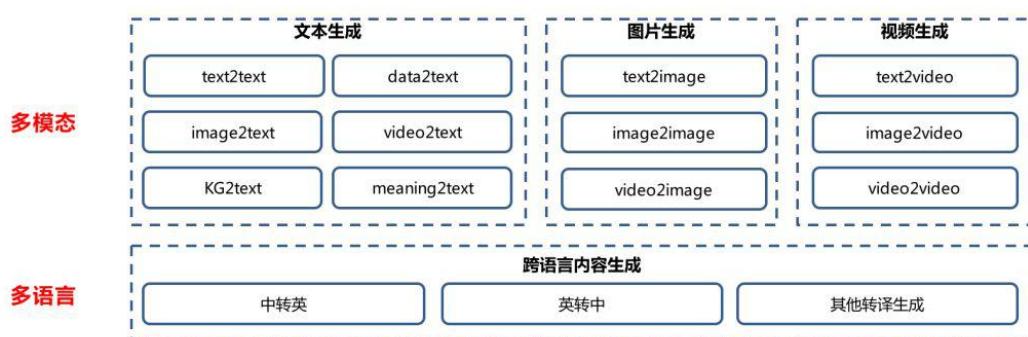
下面介绍目前我们在探索的摘要生成方法。其是在 seq2seq+attention 模型的基础上引入 pointer network 机制构造出了新的文本摘要模型。这个模型既能够从源文本中选择复制单词，同时还保留从固定词汇集中生成单词的能力，在 loss 上对重复出现的词进行打压，取得了不错的效果。

接着介绍我们是如何从事件脉络生成聚合类文章。针对嫦娥四号发射时间，首先从事件图谱中检索相关的时间点与事件，生成相应的事件脉络。之后通过篇章规划、自动文摘，生成相关的聚合文章，这个流程也可以用在娱乐明星的新闻生成上。

之后介绍的是 data2text 方法。主要还是基于模板的方法进行生成，首先通过对现有资讯中的文本组织形式学习，通过 bootstrap 算法自动生成相关的模板，再加以人工修正与设置触发条件。当有新的数据进入，则根据模板生成相应的文章。

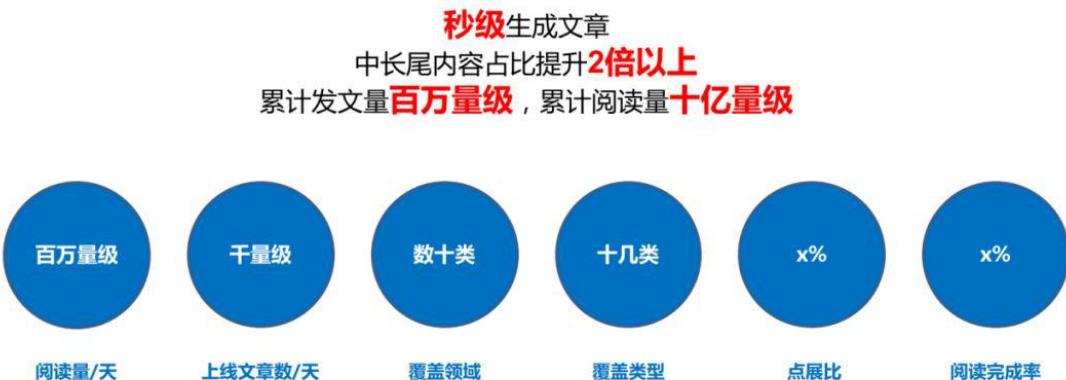
然后介绍的是多模到文本生成方法。主要依赖于知识图谱与视频理解技术，通过视频分析，从标题、关键帧、字幕等数据源，获取相应的多模实体解析。再通过知识图谱进行关联，进行联合推断产出相关的文字。

更多的自动内容生成技术



整体说，内容生成技术主要从两个角度考虑，多模态的理解与跨语言内容的生成。多模态包含各种数据到文本的技术，包括视频、图片、数据等。跨语言的内容主要包含各种跨语言转译的生成技术。

自动写作应用效果

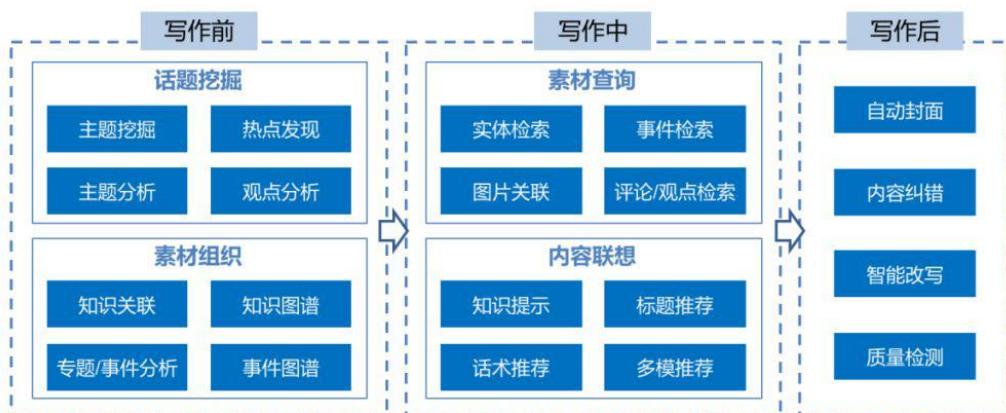


目前自动写作秒级生成文章，中长尾内容占比提升 2 倍以上，累计发文量百万量级，累计阅读量十亿量级，日均产出千级别文章，日均阅读数百万量级，覆盖数十类领域，点展比略好于人工，阅读完成率略差于人工文章。

辅助写作

辅助写作主要是指输出智能写作技术，赋能内容创作者提升写作效率与质量。已在百家号、若干媒体落地。与智能写作不同，主要面向于人配合完成写作相关步骤。

辅助写作功能



辅助写作主要作用在写作前、写作中、写作后，具体地说：

1. 写作前：话题挖掘（热点发现、观点提取等）、素材组织（专题分析、知识关联等）；

2. 写作中：素材查询（图片关联、观点检索等）、内容联想（标题推荐、知识提示等）；
3. 写作后：自动封面、内容纠错、智能改写、质量检测。

应用 – 热点话题推荐

The screenshot shows the Baidu Baike (Baidu Encyclopedia) interface. On the left is a sidebar with navigation links such as 首页 (Home), 创作风向标 (Trendy Topics), 发布 (Publish), 工具 (Tools), 分析 (Analysis), 收益 (Revenue), 设置 (Settings), and 发现 (Discover). The main content area has two sections: '关键词热度查询' (Keyword Heatmap Query) and '内容征集' (Content Collection). The '关键词热度查询' section includes a search bar and a button labeled '查看热度' (View Heatmap). The '内容征集' section displays three items: 1. '世界杯鲜肉 耶德瓦伊##世界杯' with a deadline of 2018年7月1日13时; 2. '旗舰5G扩展模块MOTO发布会' with a deadline of 2018年7月1日13时; 3. '特斯拉MODEL3 预订付订金增加' with a deadline of 2018年7月1日13时.

与自动写作不同，人工更愿意创作有深度的文章。因而第一个关键技术即是写作话题挖掘。通过新事件的发现、与长尾趣味话题的挖掘，提取一些话题源，并识别相关实体。接下来通过实体关注面的分析，获取该实体的用户关注点，并产出实体相关关系，与趣味话题。最终通过话题生成、稀缺性判断、领域类划分，获取到用户感兴趣的话题，最终进行相关话题投放。可以看到热点话题的一些具体页面：

应用 – 智能纠错

The screenshot shows a content editing interface with a toolbar at the top and a text area below. The text area contains the sentence '提升我国的综合国力是迫切需求' (Improving our country's comprehensive national strength is an urgent need). A red box highlights the word '综合国力' (comprehensive national strength). To the right, a detailed error correction interface is shown, divided into several sections: '效果如图:' (Effect as shown in the figure), '已拒绝, 拒绝理由: 标题 标题体验硬伤' (Rejected, rejection reason: Title, title experience hard hit), '标题: 这样好吗? 快餐店年轻男女当众激情舌吻, 不怕众人牌照观看, 还有孩子好吧, 能耐特点吗' (Title: Is this good? Fast food restaurant young men and women in public passion tongue kiss,不怕众人牌照观看,还有孩子好吧,能耐特点吗), '纠错提示: 1. 牌照=>拍照' (Correction tip: 1. License plate=>Take a picture), '已拒绝, 拒绝理由: 标题 标题体验硬伤' (Rejected, rejection reason: Title, title experience hard hit), '标题: 真没想到!共了这么个大逆转!刚入轨的解放军导弹弹被美军' (Title: Truly没想到! Shared so much a big reversal! Just entered orbit of the Chinese People's Liberation Army missile), '纠错提示: 1. 锁订=>锁定' (Correction tip: 1. Lock訂=>Locking), and a large red box containing the corrected text: '这样好吗? 快餐店年轻牌照观看, 还有孩子 1. 牌照=>拍照' (Is this good? Fast food restaurant young people license plate watch, there are children 1. License plate=>Take a picture).

应用 – 内容质量检测

注意了！一大波男神女神正在集结

近日，蒋劲夫被爆有家暴行为，微博上自己也承认对女友的伤害。算是坐实了家暴一事。许多人表示，根本不敢相信，多温柔的男人啊，怎么可能是这个样子？不相信的人，请看看女方的伤痕再说吧。请记住，人给你展示的永远只会是自己阳光的一面。也许蒋劲夫有千千万万的原因，但是打人、家暴这是洗不白的锅。以爱的名义去伤害爱人，这岂不是最大的讽刺，感动了自己，恶心了别人。明星也是人，也会有许多不为人知的一面，出轨、偷情、赌博、偷税，这些事情明星都有人干过。所以，永远不要被外在表现所迷惑。

蒋劲夫

这一月一直在忏悔和悔恨中度过。
对不起您们，我冲动的行为伤害了你和你的家人，不论什么原因，我都不应该动手，我为自己的行为感到羞愧，不做任何辩解。我会为自己的行为负起责任，接受惩罚。
在这里，诚恳地向大家道歉。对不起爸妈，对不起我的朋友。对不起一直以来关心和支持我的人，让你们失望了，对不起。
#超生文 ^

Instagram

吸毒,暴行,嫖娼

低质问题：
广告: 微信: xujhdj
图3: 头条号水印
疑似高危词:
吸毒,暴行,嫖娼
错别字词: 1个
委曲==>委屈
是否标题党: 是
质量打分[1/2/3分]: 2

辅助写作的第二个关键技术是智能纠错与质量检测，这里不进行技术介绍，在应用上通过各种提示提升作者的写作体验。

辅助写作应用效果，在话题推荐上，基于全网挖掘行业热点，每天发现覆盖 20+类领域的数千个热点事件，准实时热点发现。基于热点事件与知识图谱的理解、扩展能力，多角度挖掘话题，每天新增话题千量级。每天推荐话题被创作者采纳率 90%以上。

智能写作挑战

- | | | |
|---|--|--|
| 文本生成
连贯性 | 真实性 | 深度文章 |
| <ul style="list-style-type: none"> 自然语言生成难点 如何保证生成的文本无语病，如何检测辅助生成的文本是否存在语病 | <ul style="list-style-type: none"> 如何保证资讯的真实性 来源真实性保证 生成过程的逻辑 | <ul style="list-style-type: none"> 如何生成有深度的文章 需要更多领域相关的知识、关联的知识 知识图谱与认知推理 |
| 高质量 | 情感 | 辅助写作
评估 |
| <ul style="list-style-type: none"> 如何保证生成的文章是高质量的 写作全流程保证 文章内容质控功能 | <ul style="list-style-type: none"> 如何生成有情感、有观点的文章 用词造句融入情感 | <ul style="list-style-type: none"> 如何评估辅助写作的功能反馈 衡量标准 反馈通路过长 |

下面简单介绍下智能写作的挑战。具体包含如下六点：

1. 文本生成连贯性：如何辅助检测是否通顺；如何检测是否存在语病；
2. 真实性：如何保证资讯的真实性、来源的真实性、如何保证生成过程的逻辑性；
3. 深度文章：如何生成有深度的文章，知识图谱与认知推理足够么；
4. 高质量：写作全流程均需要保证，文章内容质控功能；
5. 情感：如何生成有情感、有观点的文章，用词造句融入情感；
6. 辅助写作评估：如何评估辅助写作的功能反馈，反馈通路过长。

总结

智能写作价值

- 解决媒体内容创作痛点
- 完善内容生态，提升内容价值

智能写作技术布局

- 智能写作基础技术
 - 语言、知识
 - 素材、话题、质控
- 智能写作应用技术
 - 自动写作：擅长快讯类、聚合类等类型文章写作，深度文章生成等技术仍面临较大挑战
 - 辅助写作：提升内容创作者的写作效率与质量，未来应用前景广阔

智能写作应用效果

总结：

当前来看，智能写作价值主要体现在，解决媒体内容创作痛点，完善内容生态，提升内容价值。从技术布局来讲，智能写作基础技术主要依赖于语言、知识；通过素材、话题、质控保证智能写作的顺利进行；智能写作应用技术主要体现在自动写作与辅助写作两方面，前者擅长快讯类、聚合类等类型文章写作，深度文章生成等技术仍面临较大挑战，后者提升内容创作者的写作效率与质量，未来应用前景广泛。

展望

用科技让复杂的世界更简单

继续深耕智能写作技术，深化影响内容产业，辐射到全行业自媒体与机构媒体。



展望：

后续展望过程中，我们希望继续深耕智能写作技术，深化影响内容产业，辐射到全行业自媒体与机构媒体。不仅是懂内容、写内容，更重要的是考虑创作者需求、用户需求，让智能写作更自动化、更智能化，让智能写作无处不在。

作者介绍：

彭卫华，百度主任研发架构师。硕士毕业于哈尔滨工业大学，百度知识图谱部主任研发架构师，目前负责复杂知识图谱、行业知识图谱、智能写作等知识构建与认知方向的研发工作。擅长搜索&推荐算法、机器学习、自然语言处理等技术，拥有 9 年以上相关的工业界实践经验。

内推职位：

[百度知识图谱部_知识图谱高级/资深工程师](#)

工作地点：

深圳

工作职责：

- 从事互联网数据挖掘、机器学习方向的研发工作，将海量的互联网数据结构化、图谱化；
- 应用数据挖掘等技术，对海量数据进行分析、建模 构建知识图谱并挖掘其潜在的应用价值；

- 负责数据挖掘、文本处理等算法开发，将结构化数据应用于知识图谱产品并提升产品效果；
- 基于知识图谱及相关技术创新，探索 AI 创新产品应用。

工作要求：

- 计算机相关专业，本科及以上学历，硕士博士优先；
- 熟悉 python/c / c + + / shell 等编程语言中的一门或几门，具有扎实的面向对象开发经验
- 熟悉数据挖掘、机器学习、文本处理等相关技术；
- 2 年以上互联网企业的研发工作经验；
- 具有较强的创新能力，乐于接受有挑战性的工作，具备优秀的分析问题和解决问题的能力；
- 擅长与产品经理等进行交流沟通及合作，能够准确、全面理解业务，根据业务要求给出合理的方案和分解计划。

从语言智能到行业智能

作者：吕正东 整理：Hoh

谢谢大家！很高兴能够有机会和大家聊一聊我们深度好奇最近的一些工作，和我们对自然语言处理与理解的一些思考，顺便说一下我们可能是唯一没有做智能客服的公司，下面开始我的演讲。

一、自然语言理解之难

1.1 自然语言解析

大家都知道自然语言理解是 NLP 中的核心的问题，是重中之重。那么我们就来聊一聊为什么自然语言处理这么有用，为什么又这么难。首先我们定义一下自然语言理解是什么，一直以来业界都没有一个关于它的很好的定义，那我们就用一个最简单的描述来解释它：我们把一段自然语言描述成一个机器可读的数据结构，它可以是一个图，也可以是一个逻辑表达式等等。只要后面的机器可以读这个数据结构，我们就认为它可以跟后面的业务逻辑进行对接，而这种程度的映射，我们就称它已经完成了自然语言的理解。自然语言理解里面有两个最基本的要求：首先是我们需要它有一个有足够的覆盖和同时不过于泛化的表示体系，它需要能够精确的执行我们希望它执行的东西；同时，关于映射本身，我们希望它可以通过有限的数据学习可以得到一个具有足够容错性的集合，也就是说这个映射本身也必须是可以被学习的。

1.2 自然语言处理难在哪

那自然语言处理为什么这么难呢？我对此做了几个最简单的总结，自然语言处理有四个最为核心的困难：

自然语言处理中有复杂灵活的表示方式；
自然语言中存在长距离的逻辑关联；
自然语言理解过程中存在对知识的大量依赖；
语义表示形态设计本身就很困难。

这里还有两个我们提出的核心观察：

第一个是：自然语言理解中的大量灵活性，很难通过传统的符号逻辑来充分表达；
第二个观察是：自然语言处理中的符号行为，很难通过传统的深度学习来解决。比如如果我们一边阅读，一边需要浅层或者较为有深度的逻辑推理，这就很难通过传统的深度学习来实现。

二、神经符号系统 : The way to go

2.1 学习范式的局限

由于当前语义分析的学习范式是存在很大局限性的，语义解析长期以来都是通过语法分析的形式如CCG来进行，这种严重依赖语法分析的方式存在较强的不确定性，它的训练数据往往也比较少无法有效的利用间接的监督信号。那如果我们大家现在都在做深度学习，可产生的效果也只是弥补了部分的问题，可它同时还有下面的不足：比如说不擅与处理语言中长距离的依赖关系、很难将人类常识或领域知识有效的加入进去、缺乏执行效率、缺乏严格性和可解释性缺乏符号性的泛化性能等等。

2.2 符号主义和联结主义

我们想说的一个核心的问题就是，我们要把符号智能和深度学习神经网络来做一个融合，这就是所谓的神经符号系统。符号系统用来处理离散的结构性的表示、操作具有清晰精确高效率的优点，这个符号系统指的就是我们传统的专家系统；而神经网络则是用来处理连续的表示、操作以及知识，其具有模糊可处理可学习、不确定、不可解释等特性，但它不善于处理图结构、变量、递归和指代等问题形式；通常我们会说端到端的系统也是如此，从头至尾，这个系统最后可能真的帮助你解决了某个问题，但往往你是不清楚他是怎么帮助你解决问题的。当我们把这个技术用到特定的领域比如说法律和金融时，这就会带来一定的问题：即我们虽然知道输入也知道输出，但是我们不知道系统的推理是不是符合知识和逻辑，也许只是在数据推动下得到的某一个特殊情况而已，这是它很大的缺陷；通常大家会说符号主义和联结主义的结合是一个蔓延了几十年的讨论，自从第一代的神经网络被提出之后，大家都在思考是不是可以把之前我们的传统意义上的AI的符号智能与新兴的神经网络结合起来；下图清晰的表示了符号主义和联结主义的多层面比较，这三个层面指出了符号主义和联结主义的主要区别和可结合的点：

	神经	符号
表示	连续表示，如固定维度的向量、矩阵或者张量等	<ul style="list-style-type: none"> 离散表示，如类别、字符串、图结构（包括逻辑表达式） 任何one-hot 表示，以及他们的复合结构
运算	可微运算 <ul style="list-style-type: none"> 常见的矩阵-向量运算，以及gating机制 可以直接对接基于梯度的优化，如后向传播算法等 	逻辑推理或者符号化运算，如 <ul style="list-style-type: none"> 基于规则的逻辑运算、算数运算等 改变图（包含树和链表）的拓扑结构
知识	神经系统的参数 <ul style="list-style-type: none"> 神经网络中联结权重 其他可以存储知识的可微数据结构，如外置记忆 	规则

图 1 符号主义和联结主义的比较

2.3 神经和符号融合的基本原则

既然二者都有各自的优缺点，那么接下来我们来讨论如何将它们很好的结合起来，我们认为基本上有三个原则：

第一个原则是我们要形成符号和神经交流的界面和闭环：如下图所示，比如说我们有一个向量，它经过一个神经网络比如说它是经过一个分类器，得到一个分类的结果放入一个规则引擎得到它的一个另外的符号表示最为一个输出，这个符号运算再经过嵌入层之后又得到了一个向量表示，这样我们就完成了一个闭环，我们要做的首先是建立符号和神经两边交流的界面，让神经可以调用符号、可以控制符号、可以读得懂符号，同时呢也要让符号能调用、控制、读得懂神经，这就是第一个原则；

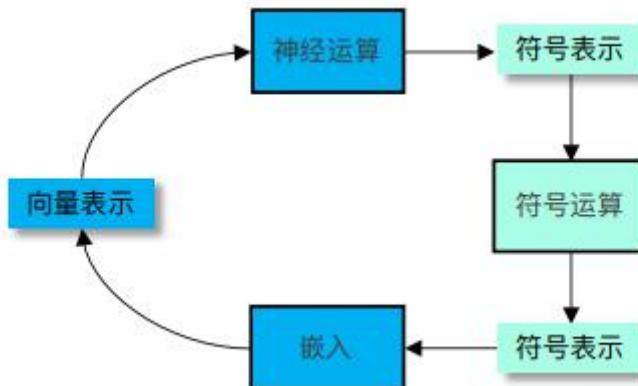


图 2 形成符号和神经的交流界面及闭环

第二个原则是我们要形成符号和神经间的并列及对偶：在我们复杂的系统中，经常会有大量的符号通入和这个神经通路的对偶，它们之间因为有各自的特点；如说，神经网络可以用BP 反向传播算法来学习，所以它是不是可以一边学习一边来教这个符号系统呢？即让符号系统明白什么是它应该做的事情。同样，符号系统在它的工作过程中也会完成总结反馈给神经系统，即刚才它学到了什么东西，这样就能够得到一个比单独使用两个通路都要好的一个结果。还有一种情况是我们有大量的符号知识，也就是我们人类总结出的各种各样的规则。那么问题来了，我们怎样去告诉神经网络我们人已经学习到了这么多有用的东西，是不是这些有用的东西可以直接告诉机器而不需它再次花费重复的时间去学习了呢？这个里面就有规则知识的消化和我们从符号知识与神经网络知识之间相互转化的过程；

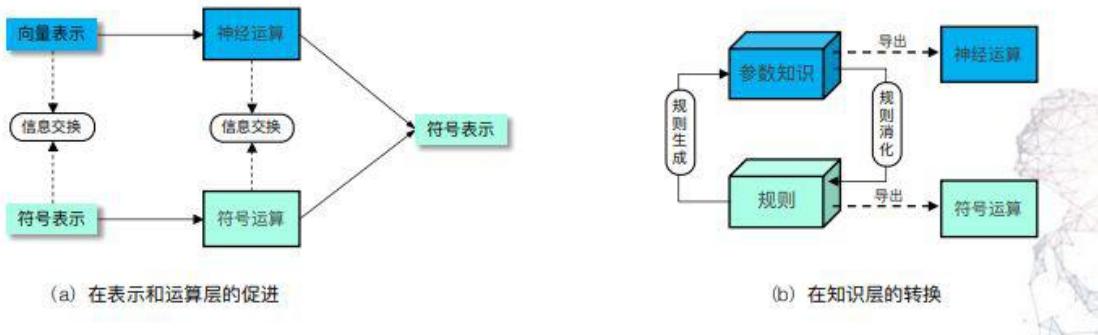


图 3 形成符号和神经之间的排列和对偶

最后一个原则是由于我们有这么多它们之间的交流对偶和并列关系 ,故若要满足第三个原则 我们就需要一个中央的调控机制去做选择控制和规划 :这是一个相对比较复杂的系统 ,因此 我们一定要保证中央调控机制的完备性只有这样才能很好的去选择、控制和规划。

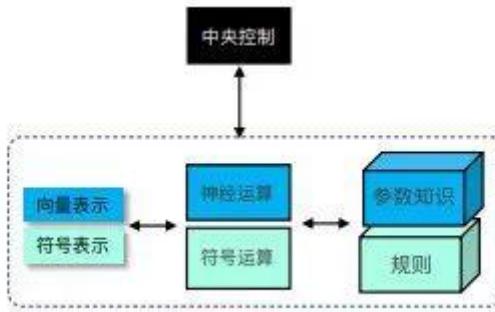


图 4 完备的中央调控机制去选择控制和规划

三、面向对象的神经规划 (OONP)

3.1 面向对象的神经规划概要

我们说了这么多的它们的这个能解决的问题和基础的设计原则 ,我们有没有一个好的实践把符号和神经网络结合起来呢 ?这是我下面要说的 ,也是我们深度好奇的研究小组在过去一年多以来的工作总结 :面向对象的神经规划 (Object-Oriented Neural Programming) 首先它是借用了借用面向对象编程思想 ,利用解析出来的实体组成对象和对象间的关系 ,构成结构清晰的图谱 ;那每个对象都是一个类的实例化 ,类的概念规定了其具有的内部属性和外部的关系和可执行的操作 ,以及与其他对象的链接关系类型。面向对象神经规划所要做的工作就是我们可以一边阅读、一边理解 ,总而言之 ,这是一个持续的决策过程 ,也是一个不断构建和完善图谱的过程 ,这和我传统的阅读理解是一个完全不同的过程。首先我们要做一个比较全的解析 ,也就是我们不是针对某一个问题或者是某一个点去做解析 ,而是我们要基本上复现整个文本的故事 ,把它完善成一个近乎信息完备的知识图谱 ;其次 ,这个过程是一边去读一边去构建的过程 ,我们之所以这么做 ,是因为 ,第一它是一个我们人可以做的过程 ,

第二是说当我们的故事足够复杂之后 ,当我们文本的叙述方式足够复杂之后这就成为了一个必须的过程。 我们不是要去构建单个的点 ,我们是要构建一个整个的复杂的图 ,那我们对图上面和文本中靠后的信息的构建和复原是需要靠前面的阅读和理解来完成的 :也就是说 ,当我们去理解一个复杂的故事的时候我们往往需要把这个故事前面的理解加到推理的过程中来去理解下面的故事。

3.2 面向对象的神经规划架构

这是一个基本的架构 ,下图可以看到这是一个比较复杂的事物 ,我们有一个中央的调控器 ,我们把它叫做 Reader ,它有三种不同的 Memory ,第一个是对象记忆 ,它是一个既有神经又有符号的基本格式 ;第二个叫做矩阵记忆 ,它是一个类似神经图灵机的这么一个可微连续的记忆 ;第三个是 Action History 因为他是一个决策过程 ,它会把过去的一些操作都记录起来 ,因此我们可以从它这些操作里面去理解我们对文章的结构和一些离散的这种划分是否合理。

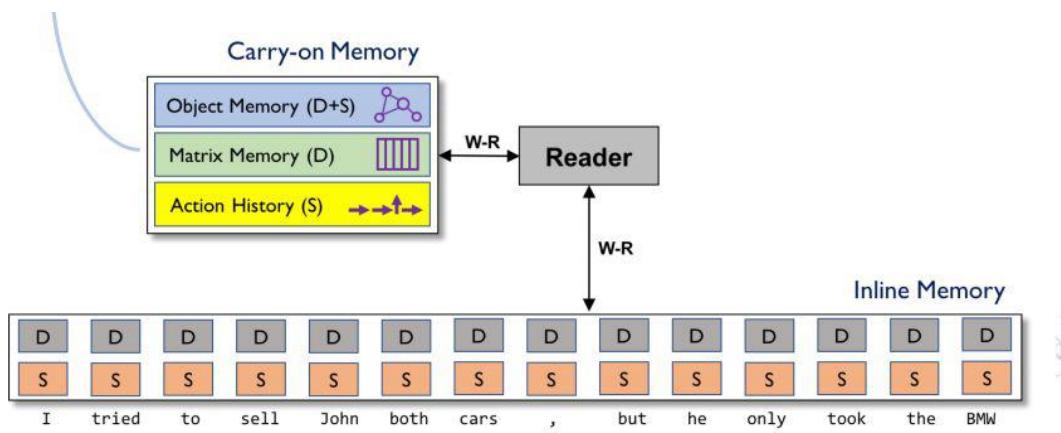


图 5 面向对象的神经规划之边阅读边理解的组织架构

我们进一步去看看中央控制器里面有什么东西 ,可以发现它里面既有这些偏符号的符号处理器同样也有神经网络的中央控制器 ,它们之间有复杂的通信关系。下面是一个实例 ,内容是 Tom 偷了两辆车 ,一辆白色奥迪和一辆宝马 ,他把这两辆车卖给 John 但是他只买了其中的一辆 ;这个动画就演示了系统一边读一边进行理解的过程 ,通过下图我们可以清晰的看到每在一些关键的点上它都会有一些关键的动作 ,这些动作是由 Action 进行触发的 ,它们会帮助系统不断的丰富知识图谱 ,等待系统读结束 ,知识图谱的构建也就相应的结束了。这个例子的自然语言理解共涉及了 22 步操作 ,它涉及了两个事件 ,一个是偷窃一件是销赃 ,然后两个人物 ,一个 Tom 一个 John ,两辆车 ,一辆奥迪一辆宝马所以说事件之间的关系也是蛮复杂的 ,宝马这个车即使被偷窃的东西也是被销赃的东西 ;

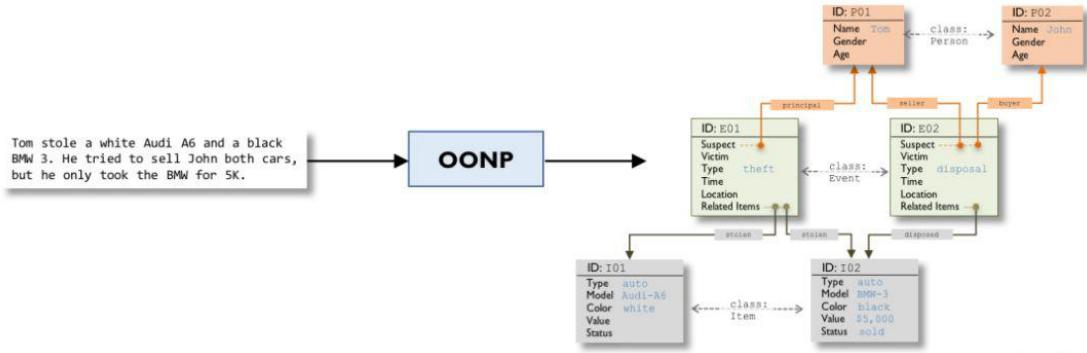


图 6 面向对象的神经规划边阅读边理解的实例过程

四、深度好奇：从技术到产品

4.1 深度好奇的技术布局

我们说了那么多很拽的技术，下面我们说说能够用它来干什么。我们深度好奇的战略布局是以整个自然语言理解为核心，同时我们也会有一个基于以 OONP 为框架的知识理解技术体系；它能够做的其实是大概三件事情：第一是复杂的文本理解；第二是对话系统。对话系统和复杂文本理解中最核心的一个是对话理解。同时我们也可以做文本生成，因为它可以看作是理解的逆过程。在此基础之上我们做了一系列产品：在公安领域，我们通过分析公安们在刑侦过程中的案情文本来记录和学习这个案件，在此之前公安可以拥有一些人工智能技术比如说视频和人脸识别，但这些信息都是一些片面的，不能够系统的去透彻的理解案情，我们在案情分析辅助决策方面为他们节省时间和提高效率；在金融领域呢，我们也有面向金融安全的基于人工智能的自然语言交互程序等。

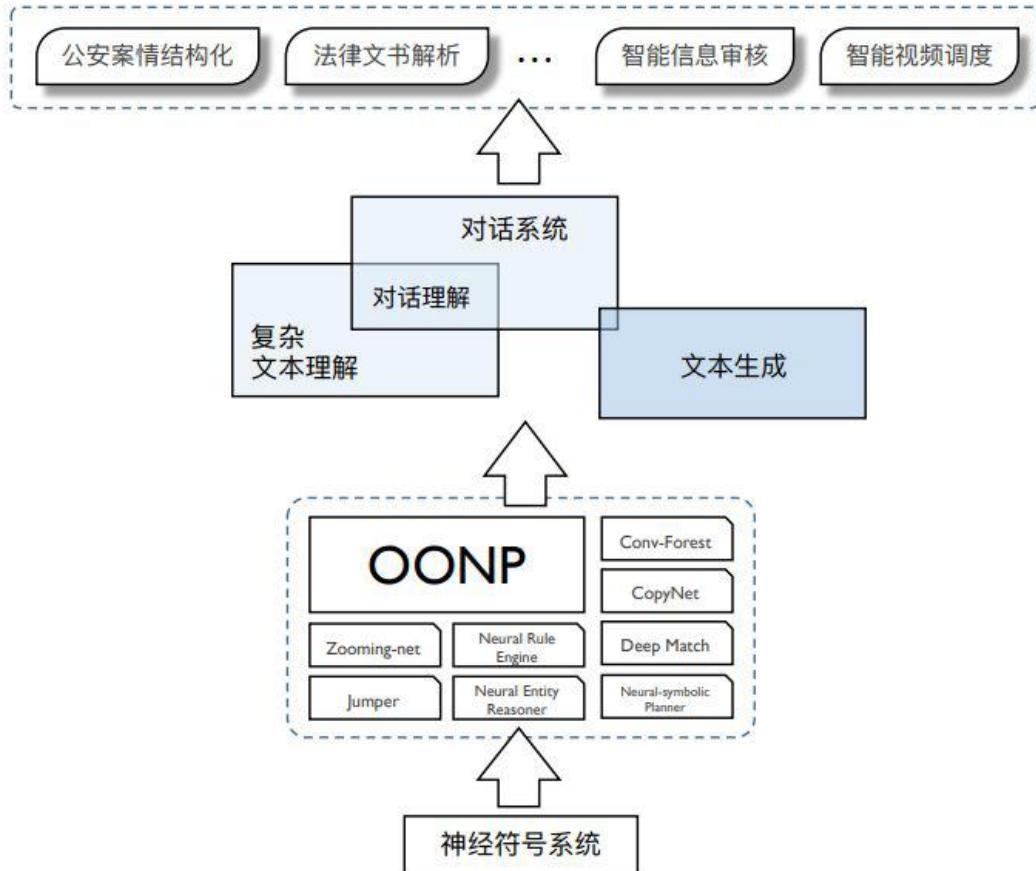


图 7 深度好奇的技术布局示意图

4.2 深度好奇项目案例分析

接下来看一些项目案例：

第一个案例是公安案情的结构化分析引擎，它可以生成人、事、物的知识图谱，以及多达 220 个标签，其准确率高达 95%，我们把此引擎接入某公安信息平台，提供串并案和犯罪预测的信息基础；第二个案例是视频平台的语音调度，提供语音视频的调度系统，该系统支持实时的高精度的对特定城市的摄像头的语音调用和语音控制，这其中也包括对视频内容和地理信息的推理和查找。



图 8 案情解析引擎

法律方面的案例是，裁判文书解析，这一块大家应该是比较熟悉的，在这一块我们人工智能的应用比较多，我们提供民事和刑事案件的判决文书解析和一些争议焦点进行提取，在这些基础之上，提供对多种形式的检索，经测试准确率高达 97%。



图 9 裁判文书解析

金融方面的案例是这个面向 P2P 的智能视频审核系统，我们开发用于视频审核的智能系统可以实现实时的无人自动对话和审核，对高风险的操作进行穷追不舍的追问，以便辅助发现各种、可能存在的欺诈；可替代用于审核的大部分人工的工作，并提供用于后期风控的基础数据，这显著的降低了骗贷和逃贷的风险。

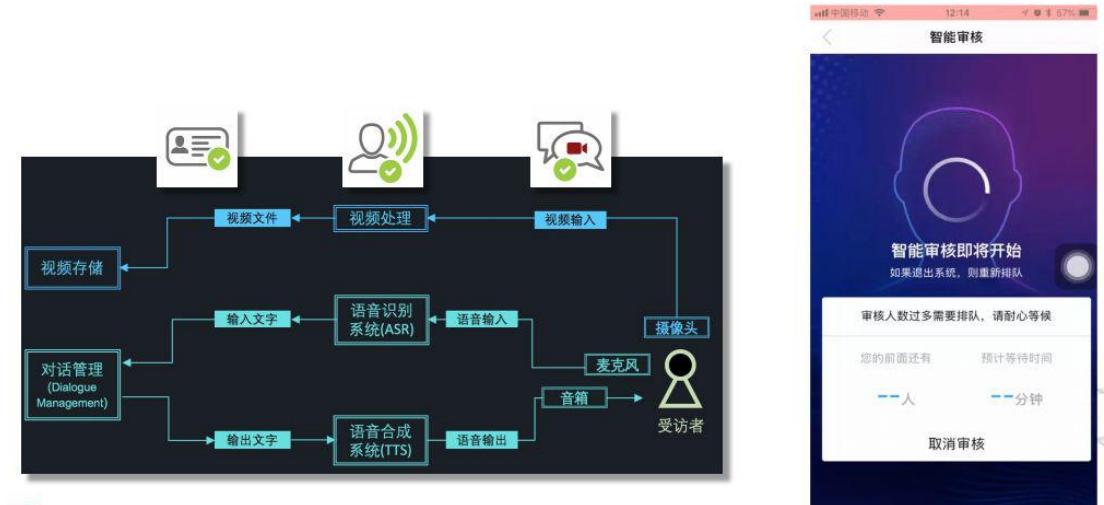


图 10 智能视频审核

五、总结

最后总结一下，我们认为自然语言理解在自然语言处理这个行业中的定位是一切智能产品融合的基础，同时我们也认为自然语言理解是一个非常非常困难的问题，因此它需要新的范式。我们正在孜孜以求的研究神经符号系统，则能够将它与我们熟悉的深度学习和类似规则的符号智能相结合，我们认为这种方案是实现复杂文本理解的唯一正确的道路。

我的分享就到这里。谢谢大家！

作者介绍 :

吕正东 博士 深度好奇™创始人 - CTO

- 留美计算机博士，新疆公共安全实验室首席专家，深度学习领域（尤其是自然语言处理方向）的国际权威。
- 2013年初创立华为诺亚方舟实验室的深度学习团队，从零开始建立软件及硬件平台，并在两年内带领诺亚方舟实验室在神经语言智能领域成为国际一流的研究机构。
- 2016年创立人工智能技术公司深度好奇，将包括神经符号模型在内的多项前沿技术应用于法律、公安、金融领域，大幅提升行业效能。其中，深度好奇的最新研究工作“面向对象的神经规划(OONP)”率先提出了复杂篇章理解的技术框架，获得学界和产业界的高度评价。
- 在2017年《人工智能杂志》关于神经语言智能的权威综述中引用的大中华区的十项工作中，吕博士及其团队的四项贡献获得了高度评价。
- 多项基于深度学习的自然语言处理技术专利的发明人，专利覆盖了语义匹配、问答、多轮对话和自动短信回复等。

内推职位 :

公司：深度好奇

Base：北京

职位：深度学习/自然语言处理算法工程师

邮箱：hr@deeplycurious.ai

深度学习/自然语言处理算法工程师

DeeplyCurious.AI

深度好奇（北京）科技有限公司，2016年设立于北京，成立初期即获国内两家顶级投资机构千万级天使轮融资。公司由语言智能领域领军人物吕正东博士领衔创办，致力于成为中国语言智能领域最具竞争力的创新技术公司。

工作职责：

- 参与自然语言理解相关算法（深度学习和传统方法）的设计及应用，包括但不限于：分类、抽取、多轮对话和知识图谱构建；

- 参与推理和决策相关算法的设计及应用，包括但不限于： 知识图谱上的推理、基于深度学习的推理和决策、预测模型等。

职位要求：

- 熟悉机器学习基础理论和常用算法，有深度学习和强化学习经验者优先；
- 熟悉自然语言处理基础理论和常用算法，有信息抽取、机器翻译等经典任务相关项目经验者优先；
- 熟练掌握 C/C++/python 至少一种语言，熟悉 tensorflow/pytorch/mxnet 等优先；
- 动手能力强，可以快速实现想法。

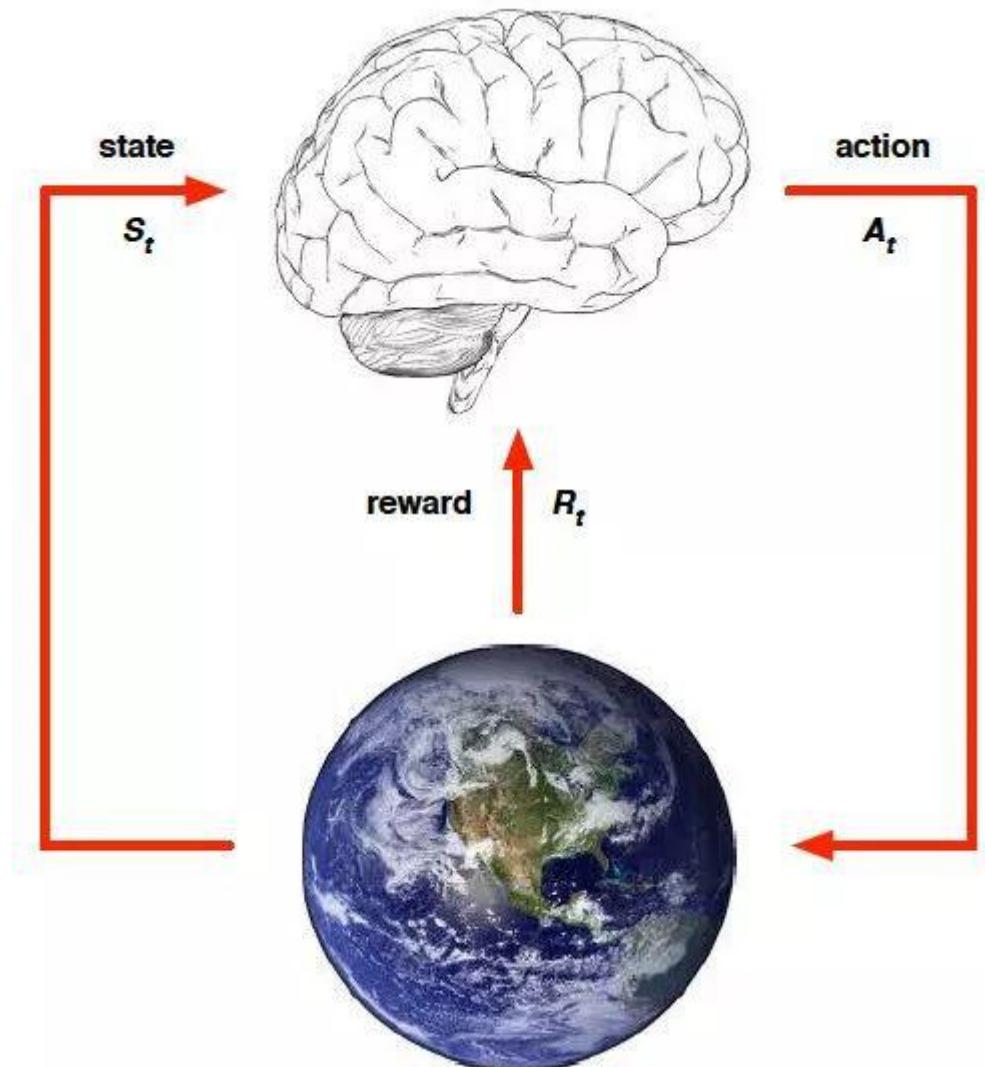
强化学习在自然语言处理中的应用

作者：黄民烈 整理：邓力

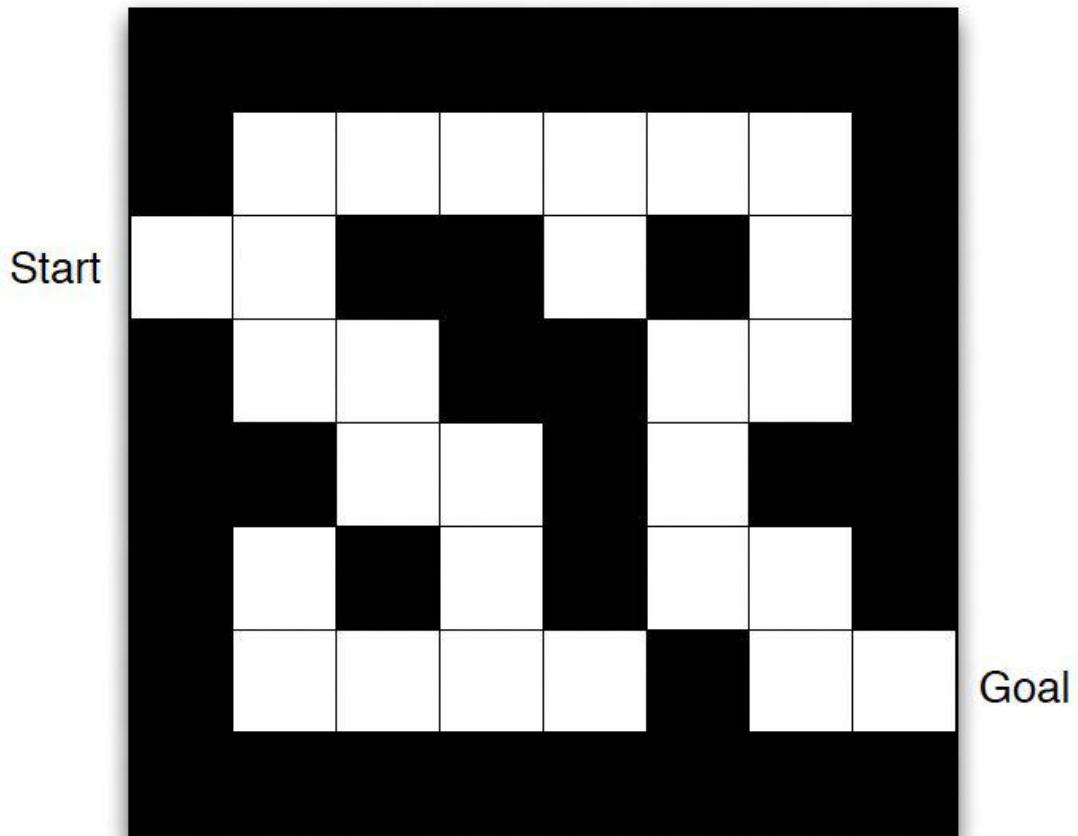
本文首先介绍了强化学习的概念和相关知识，以及与监督学习的区别，然后就强化学习在自然语言处理应用中的挑战和优势进行了讨论。

1. 强化学习

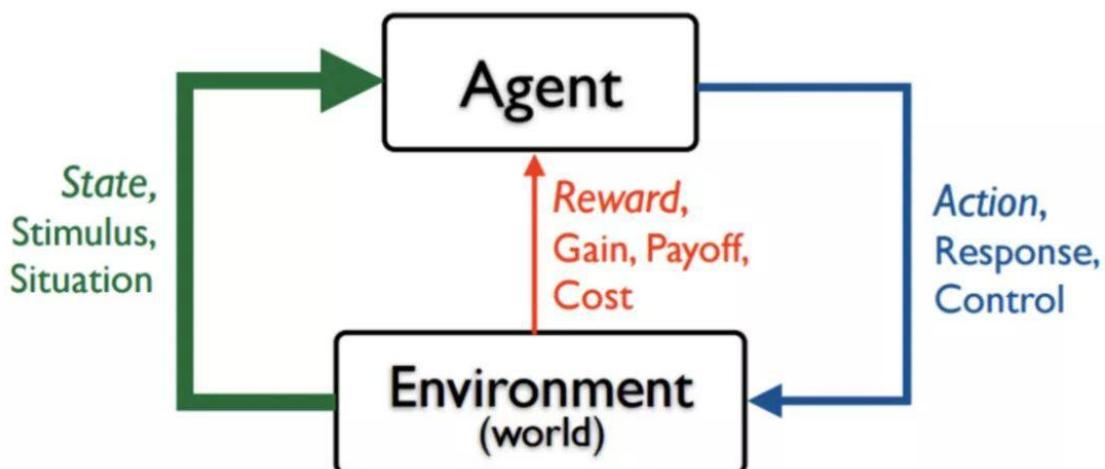
首先简单介绍一下强化学习的概念。强化学习是一种与监督学习不一样的学习范式，通过一个智能体与环境进行交互从而达到学习目标。其最重要的概念包括状态(state)，动作(action)，回报(reward)。智能体接收到环境的状态，对该状态做出一个动作，环境根据该动作做出一个回报。



以走迷宫为例，state 即为智能体所在的位置，action 就是向东西南北移动的动作，当智能体到达目标位置则给 100 的奖励，当走入死胡同则给 -100 的惩罚，每走一步给 -1 的惩罚（希望走的步数越少越好）。在该例子中，我们并没有告诉这个智能体该怎么做，只是当它做对了给它一个大的正分，当它做错了给一个大的负分。



随着深度学习的兴起，我们可以将深度学习与强化学习进行结合从而对问题进行更好的建模。深度学习可以用来刻画强化学习中的状态，动作和策略函数。二者结合的方法在很多领域都有应用，如自动控制，语言交互，系统运维等等方面。

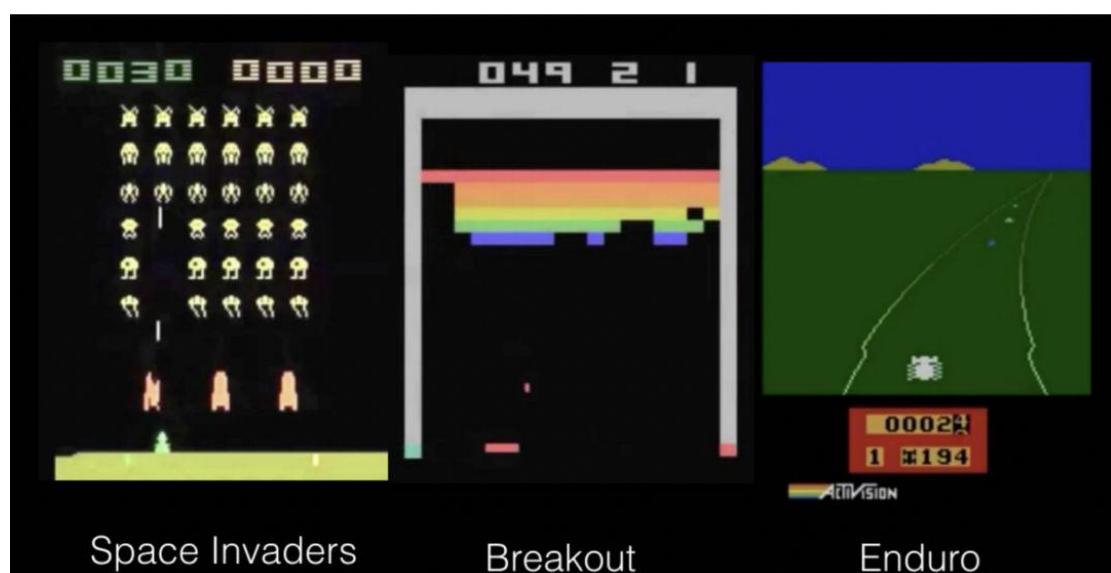


2. 强化学习与监督学习的区别

强化学习的特点：

- 1、序列决策，即当前决策影响后面的决策；
- 2、试错，即不告诉智能体怎样决策，让其不断试错；
- 3、探索和开发，即探索一些低概率事件，开发是利用当前的最佳策略；
- 4、未来收益，即当前收益可能不是最佳的，对未来来讲当前决策最佳。

监督学习就是给定一个样本集合 (x_i, y_i) 得到一个 X 到 Y 的映射。



以游戏举例，监督学习就会告诉智能体每一步应该怎么做，是向左还是向右，但在强化学习中，并不会告诉智能体应该怎么走，会让智能体自己试错，走得好的就给一个大的奖赏，走得不好就给大的惩罚。

3. 强化学习在自然语言处理中的应用

挑战

- 1、奖励的稀疏性问题；
- 2、奖励函数的设计；
- 3、动作空间维度高；
- 4、训练中的方差较大。

优势

- 1、适用于弱监督场景，问题中没有显性的标注；
- 2、不断试错调整，通过试错进行概率的探索；
- 3、奖励的积累，将专家系统或者先验知识编码进奖励函数。

1) 强化学习用于文本分类

(Learning Structured Representation for Text Classification via Reinforcement Learning)

如果做一个句子分类，首先要给句子做一个表示，经过 sentence representation 得到句子表示，把“表示”输入分类器中，最终就会得到这个句子属于哪一类。

传统的 sentence representation 有以下几个经典模型：

- 1、bag-of-words；
- 2、CNN；
- 3、RNN；
- 4、加入注意力机制的方法。

以上几种方法有一个共同的不足之处，完全没有考虑句子的结构信息。所有就有第五种 **tree-structured LSTM**。

不过这种方法也有一定的不足，虽然用到了结构信息，但是用到的是需要预处理才能得到的语法树结构。并且在不同的任务中可能都是同样的结构，因为语法都是一样的。

The actors are fantastic . They are what makes it worth the trip to the theater .



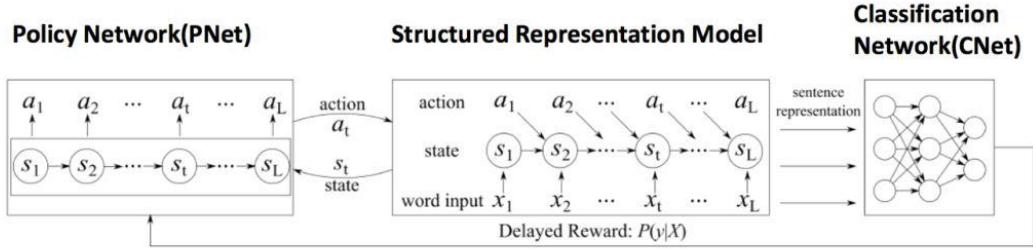
Sentence Representation



Classifier

所以我们希望能够学到和任务相关的结构，并且基于学到的结构给句子做表示，从而希望能得到更好的分类结构。但面临的挑战是我们并不知道什么样的结构对于这个任务是好的，我们并没有一个结构标注能够指导我们去学这个结构。但我们可以根据新的结构做出的分类结果好不好从而判断这个结构好不好。

可以使用强化学习来对该问题进行建模，使用策略网络来对文本从前往后扫描，得到 action(删除，切开)的序列，action 的序列即为该文本的表示，利用该表示再输入分类的网络进行分类。在该应用中，强化学习的 reward 信号来自于文本分类的准确度。



◎ Policy Network:

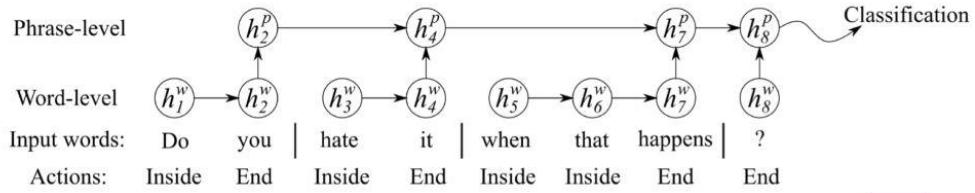
- ◆ Samples an action at each state
- ◆ Two models: **Information Distilled LSTM**, **Hierarchically Structured LSTM**

◎ Structured Representation Model: transfer action sequence to representation

◎ Classification Network: provide reward signals

第二种结构是层次的 LSTM 结构。

- ◎ Build a structured representation by discovering hierarchical structures in a sentence
- ◎ Two-level structure:
 - ◆ Word-level LSTM + phrase-level LSTM
 - ◆ Sentence representation: the last hidden state of phrase-level LSTM



先把字符切开连接得到短语，层层往上，所以是一种层次化的结构，其中 action 是(Inside , End)，状态就是当前的词与上一个词的组合，奖励就是当前类别的似然概率和结构化参数。

◎ Action: {Inside, End}

a_{t-1}	a_t	Structure Selection
Inside	Inside	A phrase continues at x_t .
Inside	End	A old phrase ends at x_t .
End	Inside	A new phrase begins at x_t .
End	End	x_t is a single-word phrase.

◎ States: $s_t = c_{t-1}^P \oplus h_{t-1}^P \oplus c_t^W \oplus h_t^W$

Word-level LSTM $c_t^W, h_t^W = \begin{cases} \Phi^w(\mathbf{0}, \mathbf{0}, x_t), & a_{t-1} = End \\ \Phi^w(c_{t-1}^W, h_{t-1}^W, x_t), & a_{t-1} = Inside \end{cases}$

Phrase-level LSTM $c_t^P, h_t^P = \begin{cases} \Phi^p(c_{t-1}^P, h_{t-1}^P, h_t^W), & a_t = End \\ c_{t-1}^P, h_{t-1}^P, & a_t = Inside \end{cases}$

◎ Rewards:

$$R_L = \log P(c_g | X) - \gamma(L'/L + 0.1L/L')$$

27

a unimodal function of the number of phrases (a good phrase structure should contain neither too many nor too few phrases)



实验数据和结果：

◎ Dataset

- ◆ MR: movie reviews (Pang and Lee 2005)
- ◆ SST: Stanford Sentiment Treebank, a public sentiment analysis dataset with five classes (Socher et al. 2013)
- ◆ Subj: subjective or objective sentence for subjectivity classification (Pang and Lee 2004)
- ◆ AG: AG's news corpus, a large topic classification dataset constructed by (Zhang, Zhao, and LeCun 2015)

Models	MR	SST	Subj	AG
LSTM	77.4*	46.4*	92.2	90.9
biLSTM	79.7*	49.1*	92.8	91.6
CNN	81.5*	48.0*	93.4*	91.6
RAE	76.2*	47.8	92.8	90.3
Tree-LSTM	80.7*	50.1	93.2	91.8
Self-Attentive	80.1	47.2	92.5	91.1
ID-LSTM	81.6	50.0	93.5	92.2
HS-LSTM	82.1	49.8	93.7	92.5

总结

这个工作中学习了跟任务相关的句子结构，基于句子机构得到了不同的句子表示，并且得到一个更好的文本分类方法，提出了两种不同的表示方法，ID-LSTM 和 HS-LSTM。这两个表示也得到了很好的分类结果，得到了非常有意思的和任务相关的表示。

2) 强化学习用于从噪声数据中进行关系抽取

(Reinforcement Learning for Relation Classification from Noisy Data)

任务背景

关系分类任务需要做的是，判断实体之间是什么关系，句子中包含的实体对儿是已知的。关系分类任务是强监督学习，需要人工对每一句话都做标注，因此之前的数据集比较小。

◎ Relation Classification (or extraction)

[Obama]_{e1} was born in the [United States]_{e2}.



Relation: BornIn

◎ Distant Supervision (noisy labeling problem)

[Barack Obama]_{e1} is the 44th President of the [United States]_{e2}.

Triple in knowledge base:<Barack_Obama, BornIn, United_States>



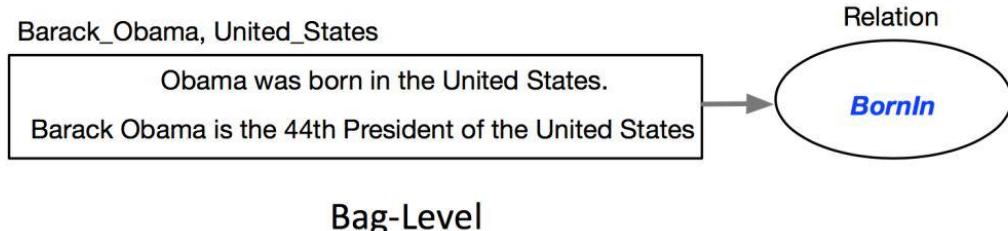
Relation: BornIn



之前也有人提出 Distant Supervision 方法，希望能利用已有资源对句子自动打上标签，使得得到更大的数据集。但这种方法是基于已有知识图谱中的实体关系来对一句话的实体关系进行预测，它的标注未必正确。

这篇文章就是用强化学习来解决这个问题。之前也有一些方法是基于 multi-instance learning 的方法来做的。

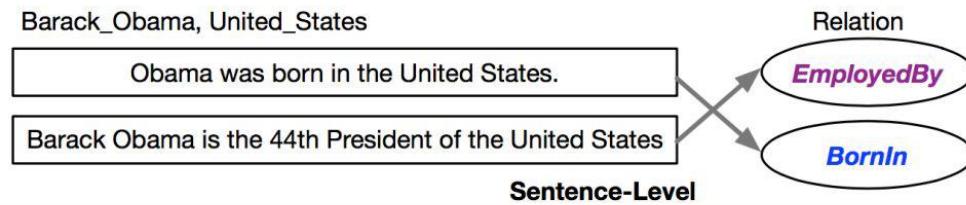
- Previous studies adopt multi-instance learning to consider the instance noises



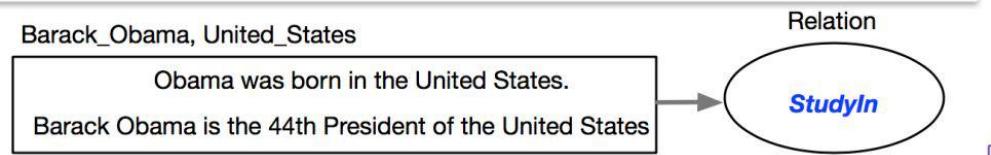
这篇文章就是用强化学习来解决这个问题。之前也有一些方法是基于 multi-instance learning 的方法来做的。这样做的局限性是不能很好处理句级预测。

- Two limitations of previous works:

- ◆ Unable to handle the **sentence-level prediction**

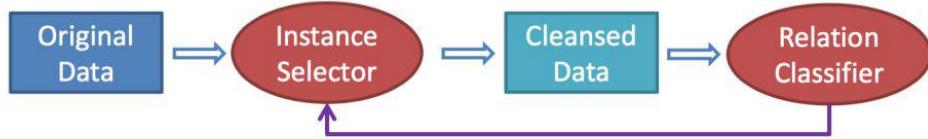


How can we remove noisy data to improve relation extraction without explicit annotations?



基于以上不足，这篇文章中设定了新模型。包括两个部分: Instance Selector 和 Relation Classifier。

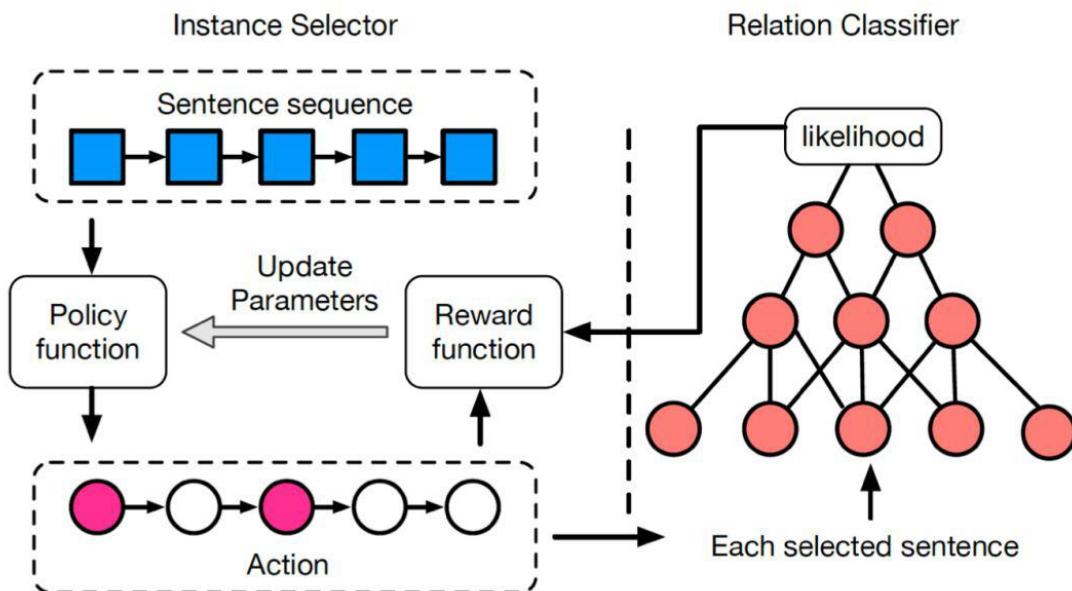
- The model consists of an **instance selector** and a **relation classifier**



- Challenges:

- ◆ Instance selector has no explicit knowledge about which sentences are labeled incorrectly
 - Weak supervision -> delayed reward
 - Trail-and-error search
- ◆ How to train the two modules jointly

这个模型有两个挑战，第一是不知道每句话的标注是否正确；第二个挑战是怎么将两个部分合到一块，让它们互相影响。



在 Instance Selector 中的“状态”就表示为，当前的句子是哪一句，之前选了哪些句子，以及当前句子包含的实体对儿。

⑤ Instance selection as a reinforcement learning problem

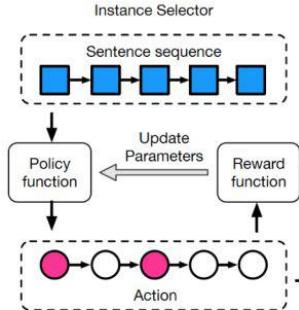
- ◆ **State:** $\mathbf{F}(s_i)$ the current sentence, the already selected sentences, and the entity pair
- ◆ **Action:** $\{0,1\}$, select the current sentence or not

$$\begin{aligned}\pi_{\Theta}(s_i, a_i) &= P_{\Theta}(a_i|s_i) \\ &= a_i \sigma(\mathbf{W} * \mathbf{F}(s_i) + \mathbf{b}) \\ &\quad + (1 - a_i)(1 - \sigma(\mathbf{W} * \mathbf{F}(s_i) + \mathbf{b}))\end{aligned}$$

- ◆ **Reward:** the total likelihood of the sent. bag

$$r(s_i|B) = \begin{cases} 0 & i < |B| + 1 \\ \frac{1}{|B|} \sum_{x_j \in \hat{B}} \log p(r|x_j) & i = |B| + 1 \end{cases}$$

40



Relation Classifier 是直接用了一个 CNN 的结构得到句子的表示。

实验以及 baseline :

⑥ Dataset

- ◆ NYT and developed by (Riedel, Yao, and McCallum 2010)

⑦ Baselines

- ◆ CNN: is a sentence-level classification model. It does not consider the noisy labeling problem.
- ◆ CNN+Max: assumes that there is one sentence describing the relation in a bag and chooses the most correct sentence in each bag.
- ◆ CNN+ATT: adopts a sentence-level attention over the sentences in a bag and thus can down weight noisy sentences in a bag.

◎ Sentence-Level Relation Classification

Method	Macro F_1	Accuracy
CNN	0.40	0.60
CNN+Max	0.06	0.34
CNN+ATT	0.29	0.56
CNN+RL(ours)	0.42	0.64

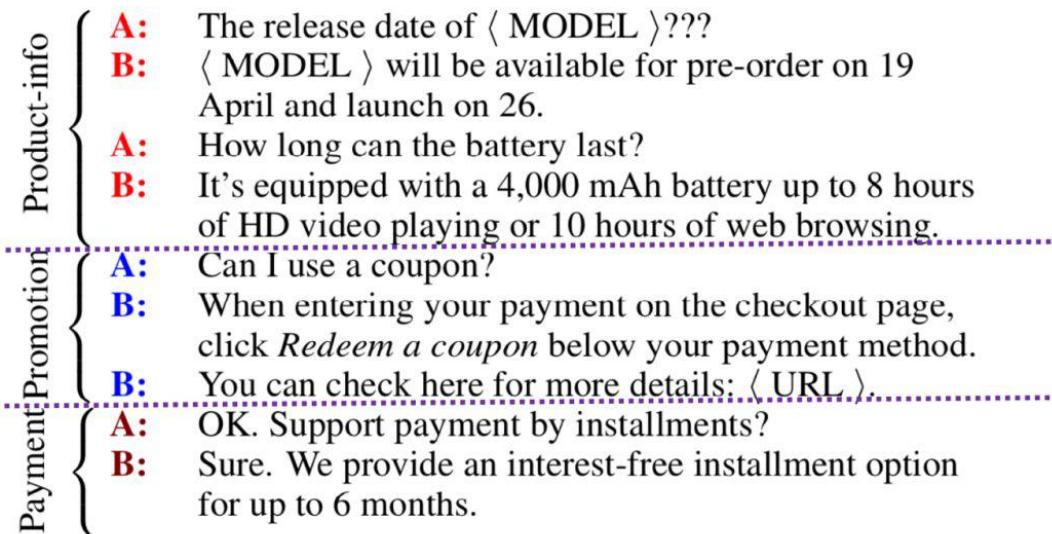
提出一个新的模型，在有噪声的情况下也能句子级别的关系分类，而不仅仅是 bags 级别的关系预测。

3) 强化学习用在面向目标的主题分割与标记的弱监督方法

(A Weakly Supervised Method for Topic Segmentation and Labeling in Goal-oriented Dialogues via Reinforcement Learning)

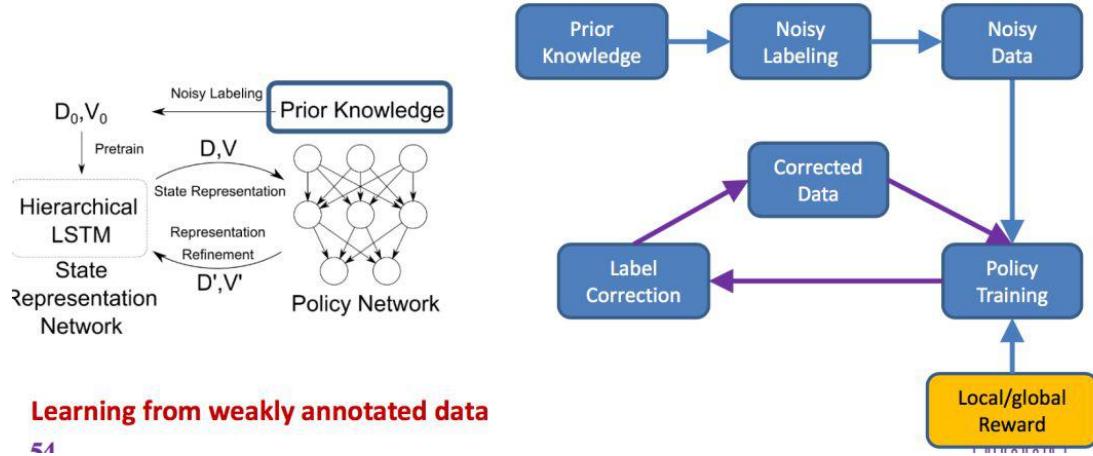
任务背景

- 1、客户服务对话经常出现在大型 Web 服务中；
- 2、主题分割和标记是一种粗粒度的意图分析，是对话理解的关键步骤；
- 3、对话结构分析是面向目标对话系统中的一项重要任务。



将上图中的对话数据自动切开并打上标签就是我们的具体问题。

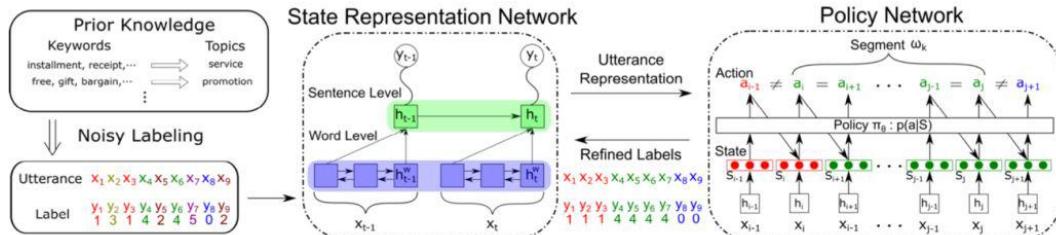
- Noisy labeled data → learn policies with reward → refine data → learn better policies → refine more data



基于先验知识对数据进行粗略的打标签，对打好的标签数据训练一个策略，根据该策略自动的纠正标签，再用纠正后的标注数据训练策略，这样不断的对数据进行纠正。

State Representation Network

Policy Network



训练一个状态表示网络用于状态的表示，基于状态表示网络训练一个策略网络。

- ◎ Local topic continuity: the same topic will continue in a few dialogue turns

$$r_{int} = \frac{1}{L-1} sign(a_{t-1} = a_t) \cos(\mathbf{h}_{t-1}, \mathbf{h}_t)$$

- ◎ Global topic structure: high content similarity within segments but low between segments

$$r_{delayed} = \frac{1}{N} \sum_{\omega \in X} \frac{1}{|\omega|} \sum_{X_t \in \omega} \cos(\mathbf{h}_t, \boldsymbol{\omega}) - \frac{1}{N-1} \sum_{(\omega_{k-1}, \omega_k) \in X} \cos(\boldsymbol{\omega}_{k-1}, \boldsymbol{\omega}_k)$$

56 

利用话题的连续性和全局主题结构作为奖励策略。

实验结果：

Experiment

(a) Topic Segmentation (MAE and WD)				
Model	SmartPhone		Clothing	
	MAE	WD	MAE	WD
TextTiling(TT)	13.09	.802	16.32	.948
TT+Embedding	3.59	.564	3.17	.567
STM	4.37	.505	8.85	.669
NL+HLSTM	8.25	.632	16.26	.925
Our method	2.69	.415	2.74	.446

(b) Topic Labeling (Accuracy)		
Model	SmartPhone	Clothing
Keyword Matching	39.8	31.8
NL	51.4	39.0
NL+LSTM	49.6	35.5
NL+HLSTM	52.6	40.1
Our method	62.2	48.0

57

Model	(a) # Keywords per topic		
	3	6	9
NL	45.0	51.4	48.0
NL+HLSTM	46.6	52.6	48.8
Our method	55.3	62.2	58.2

SubSets	(b)	
	KM	1-NN
Utterances	3,503	7,385
NL	78.7	38.4
NL+HLSTM	78.6	40.2
Our method	79.0	54.2

Model Setting	(c)	
	Segmentation	Labeling
	MAE	WD
RL + r_{int}	3.04	.449
RL + $r_{delayed}$	3.89	.490
RL + $r_{int} + r_{delayed}$	2.69	.415
		59.5
		60.4
		62.2

总结

- 从有噪声标签的数据开始（避免昂贵的完整标注）；
- 不删除有噪声的数据，而是使用强化学习对噪声数据标签进行纠正；
- 弱监督：我们需要的只是一组关键词和一些先验知识。

4. 强化学习在 NLP 中成功应用的关键

- 1、把一个任务描述成一个自然的顺序决策问题，其中当前的决定影响未来的决定；
- 2、当你没有充分的、强有力的监督时，记住试错的本质；
- 3、将任务的专业知识或先验知识编码进奖励中；
- 4、适用于许多薄弱的监控环境。

作者介绍：

黄民烈，清华大学计算机系副教授，博士生导师。研究兴趣主要集中在人工智能、深度学习、强化学习，自然语言处理如自动问答、人机对话系统、情感与情绪智能等。已超过 50 篇 CCF A/B 类论文发表在 ACL、IJCAI、AAAI、EMNLP、KDD、ICDM、ACM TOIS、Bioinformatics、JAMIA 等国际顶级和主流会议及期刊上。曾担任多个国际顶级会议的领域主席或高级程序委员，如 IJCAI 2018、IJCAI 2017、ACL 2016、EMNLP 2014/2011，IJCNLP 2017 等，担任 ACM TOIS、TKDE、TPAMI、CL 等顶级期刊的审稿人。作为负责人或学术骨干，负责或参与多项国家 973、863 子课题、多项国家自然科学基金，并与国内外知名企业如谷歌、微软、三星、惠普、美孚石油、斯伦贝谢、阿里巴巴、腾讯、百度、搜狗、美团等建立了广泛的合作。获得专利授权近 10 项，其中 2 项专利技术授权给企业应用。

旅游知识图谱的构建和应用

作者：鞠剑勋 整理：金媛

本文首先介绍了什么是旅游知识图谱，然后就旅游知识图谱的架构，构建，应用和未来几个方面展开讨论。

1. 旅游知识图谱

首先简单介绍什么是知识图谱。知识图谱是由 Google 公司在 2012 年提出的新概念。用信息可视化技术将知识以图的形式表示，图由节点和边构成，**节点**对应知识图谱的实体，自然界中的每个对象都可以称之为一个实体，例如人，公司，酒店，甚至酒店内的某个房间都可以称为实体；**边**对应知识图谱的关系，及实体之间的关系，比如**酒店位于北京市，“位于”就是**酒店和北京市之间的关系。

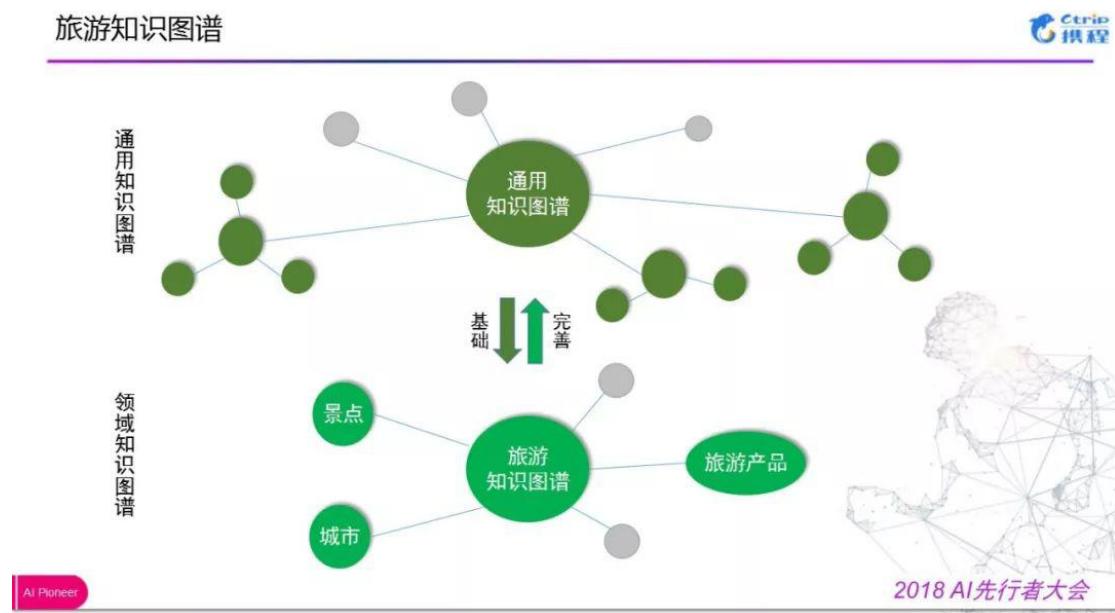


小问题：我们为什么要构建旅游知识图谱（旅游知识图谱的作用）？

传统的推荐系统会根据用户的历史行为，为用户打上隐形标签，并为用户推荐相关的产品。假设用户订购了普吉岛的旅游产品，比如自由行，用户的原因可能是喜欢旅游，喜欢海岛，于是推荐系统为该用户打上了“喜欢海岛”的标签，为该用户推荐了很多海岛的相关产品；有些用户可能喜欢普吉岛的某些服务，例如露天泳池，某家酒店等，推荐系统则引入酒店等一些特征加入推荐模型中；某些用户因为去普吉岛只需要落地签，不需要额外的手续，于是

推荐系统加入了签证相关的特征 ; 某些用户因为去普吉岛的机票打折 , 喜欢泰国的一些旅游景点 , 喜欢海鲜等 , 如果将这些特征全加入推荐系统 , 会发现推荐系统变得很困难。

酒店数据 , 机票数据 , 签证数据 , 景点数据等 , 每种数据都需要单独的数据库或者数据表去维护 , 将这些数据联合分析可能要做大量的 BI 工作 , 这些繁杂的分析都可以用知识图谱取代 , 这就是旅游业需要知识图谱的原因。



通用知识图谱可以看作是一套模板 , 利用领域知识向模板中填充内容 , 形成特定的领域知识图谱 , 例如旅游知识图谱 , 金融知识图谱 , 医药知识图谱 , 动物知识图谱等。通用知识图谱是领域知识图谱的基础 , 而领域知识图谱是通用知识图谱的扩充 , 二者相辅相成。



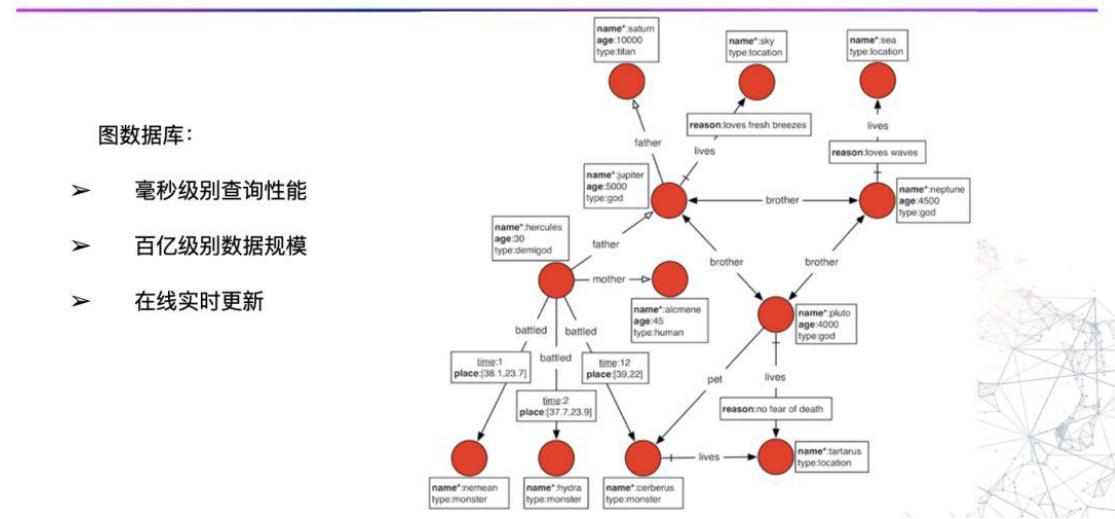
上图是旅游知识图谱的一个例子。以旅游产品为中心，扩散出与其相关的其他产品，比如酒店，机票，目的地，餐厅，签证，景点等。首先定义知识图谱的实体，酒店，景点，目的地，机票，餐厅等，以及它们都有哪些属性特征，例如酒店的星级，坐标，价格等，然后定义实体之间的关联，例如**酒店距离**景点多少米就是酒店和景点之间的关联，然后把具体的产品当作实体映射到本体上，以一个图结构去存储数据，建立知识库，形成知识图谱的简单架构。

2. 旅游知识图谱的架构

一套完整的旅游知识图谱架构：上层应用包括 QA 应用，推荐搜索，知识挖掘等方面的应用。QA 对话主要应用在以下几个方面：智能客服，智能导购，客服助手和对话机器人等，携程，淘宝等应用智能客服，智能家居则应用了智能导购系统，百度的智能音箱是类似对话机器人的一个产品。当进行 QA 问答时，智能回答者要通过知识图谱寻找答案，完成对话。图谱的构建包括 schema 本体管理，域管理等等，需要提前定义实体的类别属性等，还有数据自动化等构建，比如说知识的来源，大段的文章中抽取实体，关系等，同时对多个知识图谱进行融合，做一些补全，推理等操作，全局优化就是做一些一致性校检，智能更新等；底层是数据层，可以将现有数据库导入到知识图谱中，也可以从外部通用知识图谱收集知识，也可以从文章中，等非结构化数据中提取和采集知识来完善知识图谱。知识存储分为两个部分，分别是 rdf，类 rdf，比如 owl，还有一个是图结构的存储。此外还有一些机器学习，nlp 的一些算法等共同构建了知识图谱的架构。

owl 用来存储一些三元组，本体和本体之间的关系，好处：清晰的 schema 定义，丰富的类与类之间的关系，实现一些简单的推理，比如属性和属性之间是否存在相反关系，位于关系，比如 a 位于 b，b 位于 c，那么 a 位于 c 这样的传递关系，同时可以给每一个实体定义一个类型，给每一个类型提前定义一些属性，根据 schema 的 type 和属性往里面插入数据，本体 schema 可以认为是数据库表的列名，它已经限定数据库可以存哪些不可以存哪些东西。

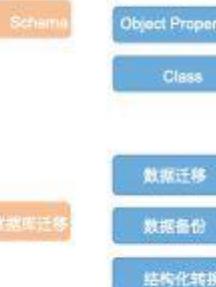
旅游知识图谱架构



除了 owl , 还有图数据库。图结构的好处就是毫秒级别的查询性能，容纳百亿级别的数据规模，可以在线实时更新，图数据库有很多种。

旅游知识图谱构建

- > schema 定义
- > 知识采集
- > 数据库迁移
- > 实时更新

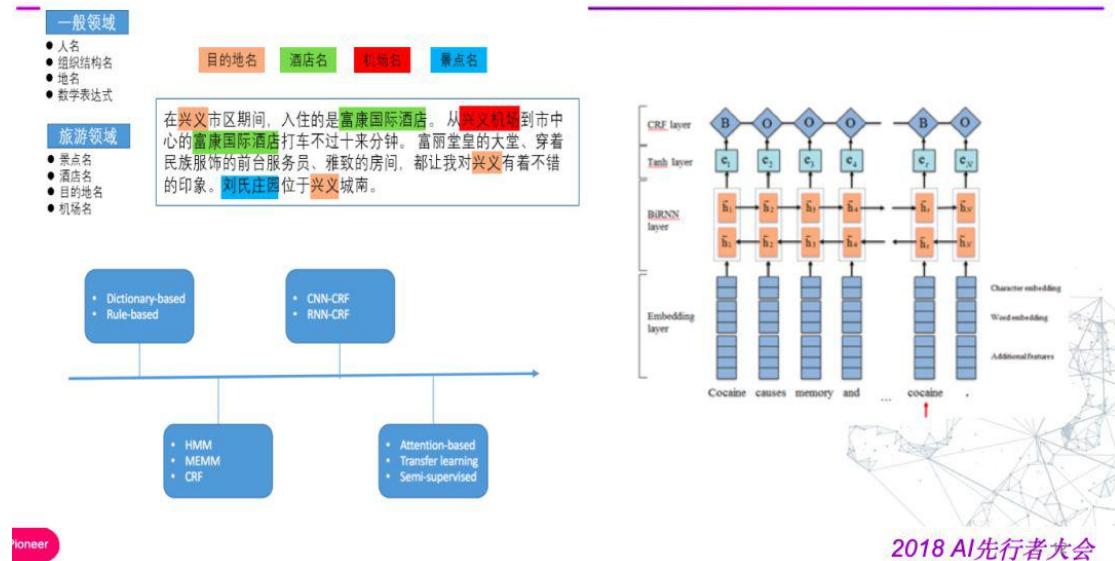


2018 AI 先行者大会

3. 知识图谱的构建方法

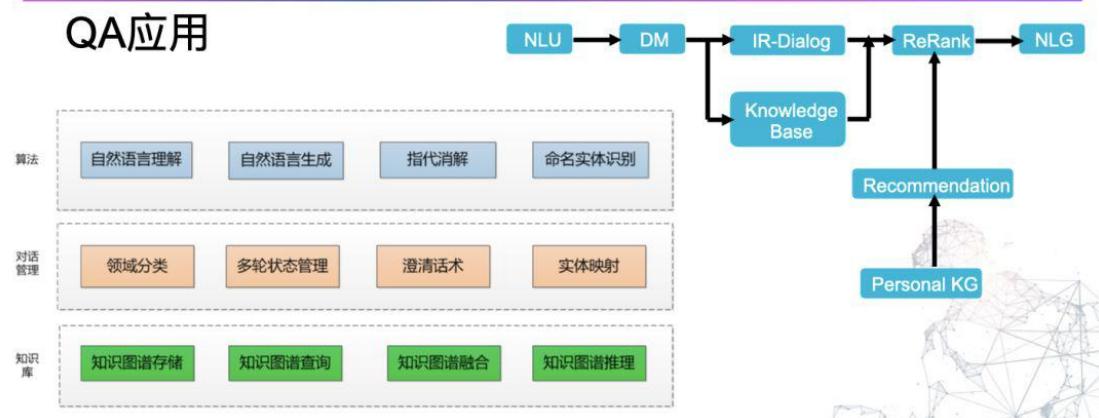
首先，定义 schema，比如实体的类型，数据类型，属性类型，类别等，然后做一些知识采集的操作，从文本中抽取关系，从外部知识库补充一些已有的三元组，接着是数据库迁移，从 sql 数据库中的数据迁移到知识库中，数据库备份，数据结构的转换，实时更新就是检查数据的一致性，对重复的内容做知识融合，比如 china 和中国，尽量保证实体的唯一性。

旅游知识图谱构建



在抽取实体时一般领域会抽取出人名，组织结构名，地名，数学表达式等；在旅游领域会抽取出旅游相关的实体，例如景点名，酒店名，目的地名，机场名等。比如这句话：“在兴义市区期间，入住的是富康国际酒店。从兴义机场到市中心的富康国际酒店打车不过十来分钟。富丽堂皇的大唐，穿着民族服饰的服务员，雅致的房间都让我对兴义有着不错的印象。刘氏庄园位于兴义城南”，在旅游领域抽取的实体有：目的地名兴义，酒店名富康国际酒店，机场名兴义机场，景点名刘氏庄园。命名实体识别一开始是基于规则或字典的方法抽取实体，接着发展到利用模型抽取实体，例如 HMM，HEMM，CRF 模型都可以用来做序列标注，从而识别实体。现在的做法一般是将卷积神经网络 CNN 或循环神经网络 RNN 与 CRF 结合的模型。

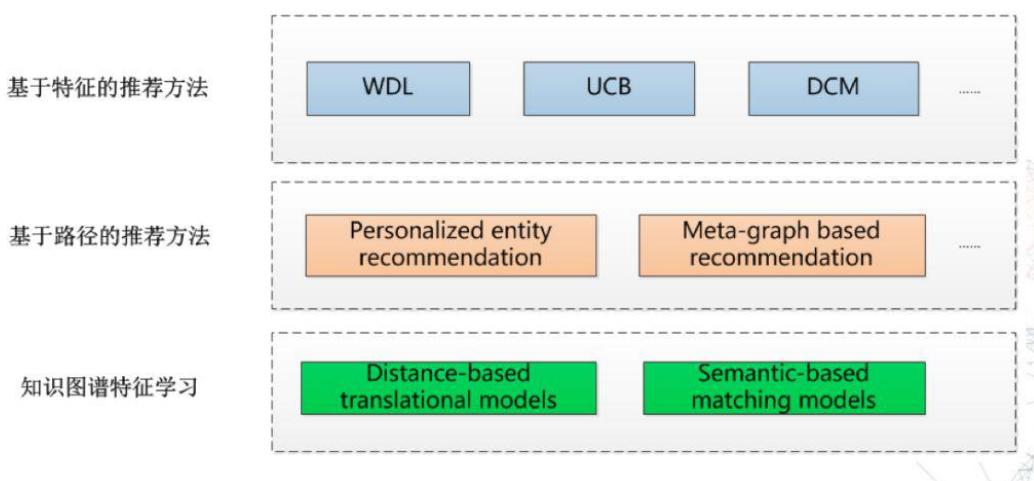
旅游知识图谱应用



4. 知识图谱的应用

QA 问答系统首先进行 NLU 语音识别，语音识别就是把语音信号转化为文本或者指令的技术并确定语音的意图，DM 会话管理是人机对话的核心，它主要用来维护和更新对话状态，当前的会话状态依赖于之前的系统状态和之前的系统响应以及当前时刻的用户输入。QA 系统的答案基本从知识图谱中获取，需要从句子中抽取出来实体，将句子的意图等映射到知识图谱中进行查询，提供答案。

推荐应用



知识图谱一般不用于基于特征的推荐系统，一般用于基于路径的推荐，分为两种 meta-path 和 meta-graph，用户喜欢普吉岛的酒店，喜欢 spa 服务和泳池，可以建立这样的一条路径作为推荐系统的特征，参与计算。缺点：需要提前设定这样的路径，不支持自动搜索路径。知识图谱特征学习，将特征转化为向量的形式，辅助推荐。应用于 embedding，协同过滤中只考虑 user 特征和 item 特征，可以利用知识图谱作一些特征，embedding 有很多方法，深度学习。

知识图谱还可以应用到搜索方面，传统搜索都是全文索引之类的搜索，没办法解析一些包含语义的句子，但是知识图谱可以解析出实体，筛选出一些答案，是基于语义理解方面的搜索。

5. 知识图谱的未来

将知识图谱的语义信息，图像输入到深度学习模型中，映射到知识图谱的三元组的实体，关系或者图上，将离散化的知识表示为连续的向量，从而使得知识图谱的先验知识能够称为深度学习对输入，参与模型的计算，加强模型，比如问答，翻译。离散知识转化为向量；同时，利用知识作为约束目标的约束项，从而指导深度学习模型的学习过程，通常是将知识图谱的知识表示为优化目标的后验证项。未来会在知识图谱中做多领域的融合，自动推理，自动

抽取，事件图谱，比如某人最近发生了什么事情，这是变化比较频繁的图谱，主要应用在开放域对话系统，旅游线路推荐系统，旅游生态规划和热点事件追踪等。

作者介绍：

鞠剑勋，携程旅游度假 AI 自然语言处理负责人。主导携程旅游知识图谱的整体构建，有五年的自然语言处理和知识图谱相关经验，专注于自然语言处理和知识图谱方面的应用和算法研发。

更多关于携程的技术文章欢迎关注：



携程技术中心公众号

NLP 在网络文学领域的应用

作者：马宇峰 整理：赵世瑜

一、业务背景



网络文学的发展已有 20 年的时间，阅文从发展之初的不太看好，再到现在发展为 400 亿港币市值的阅读平台和文学 IP 巨头。他的发展历程并不是那么一帆风顺，但却也契合了当前人们对物质文化的热切需求。目前很多网络小说已经改编为电影或电视剧，按每天每人阅读消费带动收入，可以说网络小说在泛娱乐场景下已无处不在。

1.1 发展历程

从中国加入国际互联网以来，网络小说一直是互相竞争的领域，几乎互联网的巨头都想要对这个领域进行渗透。其原因主要是小说是 IP (Intellectual Property) 的一个起源，为了争夺这样一个起源，很多公司都会在网络小说方面进行布局。2016 年阅文统一网络小说后，竞争仍在继续，如现在的爱奇艺文学以及头条传媒平台，都在做网络小说。

网络小说主要是创作式平台，作者在里面占有举足轻重的作用，所以永远不可能达成平台化的单方垄断，很难把所有的竞争对手都压制住，因而需要不断的提升作者与读者双方的生态体验。

1.2 产业状况

现代小说不再是作家单枪匹马进行创作，而是变成挖掘哪些元素比较受人喜欢，然后以一种比较快的方法去切入进行变现，并伴随一些商业衍生产品。如从网络小说衍生为国产动漫，电视剧、电影、自拍剧等，这些都是网络小说的一些变现方法。

网络小说是一种产业链生态，不仅仅是写小说、看小说这一件事。更多的是用户会参与其中，并告诉我们小说衍生的下一步应该如何走，是应该变成漫画还是变成影视剧。也正是这个原因，大量的付费阅读变成了免费阅读，希望把自己的作品扩展到其他领域，获取更多的收益。每家都有自己的网络小说平台，发展方式都是从明星作家到产业变现的方式。为了 IP 变现和影视流动，需要对网络小说作品做更深层次的理解。以前不太关注的点，如一篇长篇小说是不是适合改编成影视、游戏或者动漫，如何对改编的合理性进行评估，现在都需要有深层次的理解。

1.3 作者作品

网络小说头部流量作品主要有玄幻、奇幻、科幻、仙侠、武侠、都市、历史、灵异和游戏。每种类型的网络小说都有自己的代表作，如武侠类别的代表作为英雄志。

不同类型的小说有不同的表现形式。如玄幻小说和武侠小说是完全不一样的作品，玄幻小说比武侠小说有更夸张的表现形式，如手一挥，星球就爆炸了，这种在武侠小说中仍然不存在。随着种类的变化词的意义也不同，如“吓死了”，很多时候不是死了或者要死了，而是情感的一种表达方式。这也是 NLP 之所以困难的一个原因。也就是说 NLP 是由共识而来，也是会随共识而变。对于一个词的语义，在不同的文章中、不同的上下文中都在不停的变化，不可能有一个标准的方法来处理一切 NLP 问题。而且热门的网络小说类型也在不停的变化。

1.4 写作套路

网络小说的写作有自己套路，一般表现在书名、等级、打斗、装备、悬念及世界观等方面。如书名要么狂、要么 low，总之要贴近小白和草根；等级设定要完备、可以无限升级、做到一山还比一山高；打斗要么跨级逆袭，要么扮猪吃虎；装备则需要变废为宝、随手捡来的垃圾也得是个宝物；明线暗线要留足吊胃口的悬念，例如要报仇、要找爹妈、要复活老婆等；世界观都非常大，如玄幻仙侠中，可涉及地球、星系、异界、多宇宙、平行宇宙、混沌等。网络小说到最后已经不是在写文章，而是写一种体系和架构。作者会驱动自己把文章变成架构体系。文字风格需要使用夸张的手法，如一吼之下，让好几座山峰都炸开。

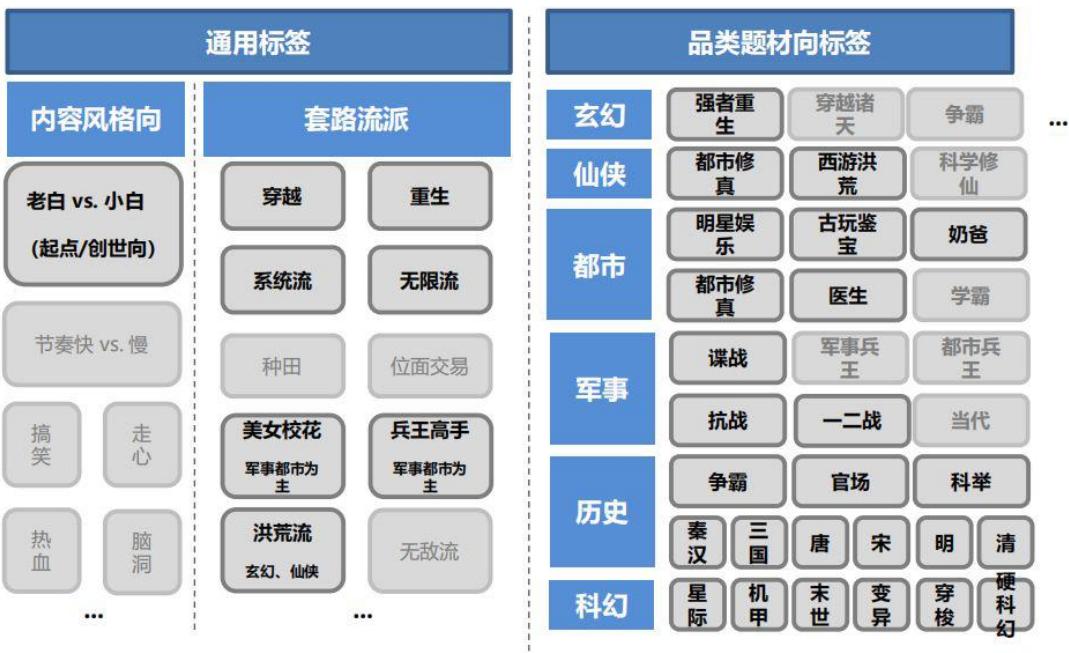
1.5 网文结构化-标签维度

这里说的网文结构化，主要从标签维度考虑。而在这之前需要明确什么是标签以及标签如何进行定义，如何把标签描述清楚等。一个好的标签体系是后续工作的基础。

对网络文章进行结构化，主要是通过技术结合人工进行确定。标签标记大致流程为。首先通过运营、编辑结合技术手段确定标签体系，再通过用户填写标签，以及通过技术判断标签下的候选作品集，运营和编辑对候选结果进行判断后与作家沟通确认（但不许作家随意改动）后，形成最终的用户标签。如果后续需要补充标签，如编辑或者技术提议补充新标签，或者用户标记了新标签后，需要运营对这些新增的标签进行确认，然后在重复标签标记的工作。流程如下图所示。



标签主要分为通用标签和品类题材标签，通用标签主要强调通用性，而品类主题标签主要是结合品类进行更加细化标签。甚至内容风格、套路流派、主角身份及故事元素等方面都需要制定细化的标签体系。



为什么要这样做这么多、这么细的标签，其原因在于网络文章的推荐是不同于短资讯类的推荐，短资讯喜欢不喜欢，看一眼马上就已经明确了。但网络小说需要看比较长的篇幅才能确定喜欢还是不喜欢，如果推荐不准确，用户看了两小时后不喜欢这部小说，会导致用户对推荐非常反感。因此需要对网络小说的标签进行细化，建设更多维度的标签。

内容风格向	套路流派	主角身份	主角个性	故事元素	其余分类
小白	穿越	帝王	傲娇	青梅竹马	发展背景
爆笑	重生	总裁	腹黑	白领浪漫	角色关系
虐恋	变身	黑帮	闷骚	平凡生活	升级体系
甜文/宠文	种田	明星	多智	姐弟恋	情节/文风
治愈	神豪	医生	杀伐果断	别后重逢	历史时代
暧昧	废柴流	校花	放荡不羁	欢喜冤家	IP类
...

二、技术架构

内容挖掘目标：

持续提升内容价值转化。最简单的是确定用户喜欢不喜欢、但更重要的是要转化到其他场景中去，需要深挖，把不同的场景循环起来。这才是一个比较好的内容挖掘平台。网络小说内容挖掘主要存在三方面的问题：

- 1、内容挖掘算子分散不集中、不可互相促进；
- 2、需求来源散乱、整理代价大、不可复用；
- 3、内容挖掘后的使用渠道单一。

解决方案：

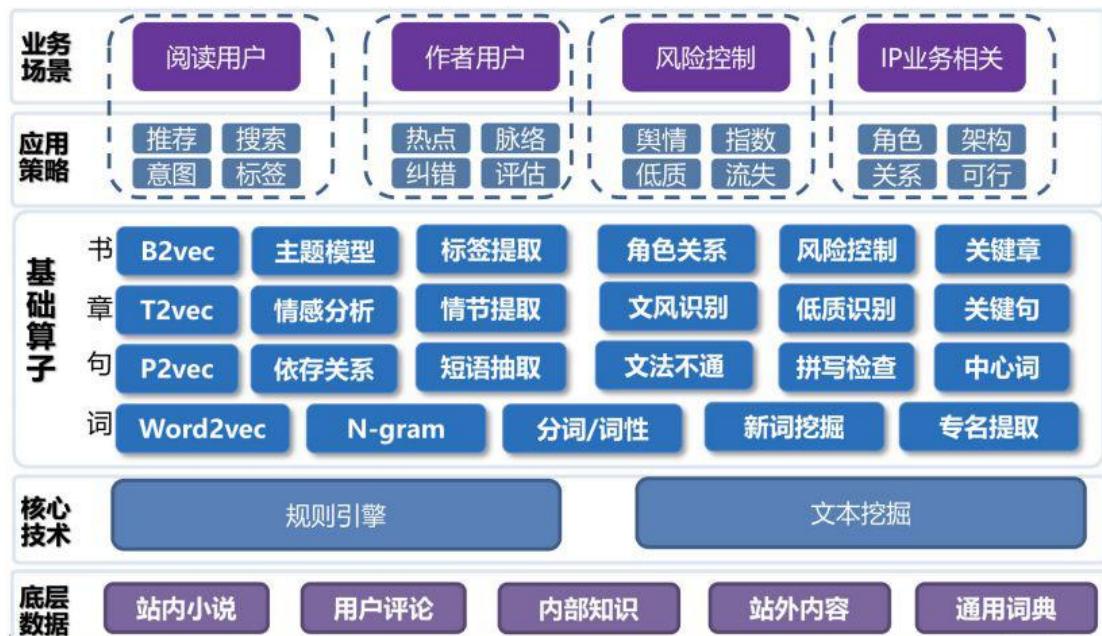
- 1、内容挖掘平台需要闭环。即平台需求、挖掘算子和业务反馈需要形成闭环。
- 2、不同平台之间需要链接。即业务观察、内容生产、挖掘平台及应用场景之间需要形成很好的链接关系。

2.1 内容挖掘平台-赋能业务

内容挖掘平台的主要任务是挖掘内容价值、赋能作者，提升内容流传效率。其目的是赋能业务，不同的业务需要挖掘不同的内容。如果是IP变现，需要预测内容的目标群体以及转化的可行性分析，应该转化为动漫、电视剧还是游戏等。如果是针对阅读用户，用户提供一些明确的信息，则需要提供推荐理由、标签和结构信息等辅助用户进行消费决策。针对作家，可以引导作家写什么样的内容，用户更感兴趣。针对内容审核可以提示一些审核风险等。通过趋势指数、候选标签的指导编辑进行内容方向的判断等。

2.2 技术全景

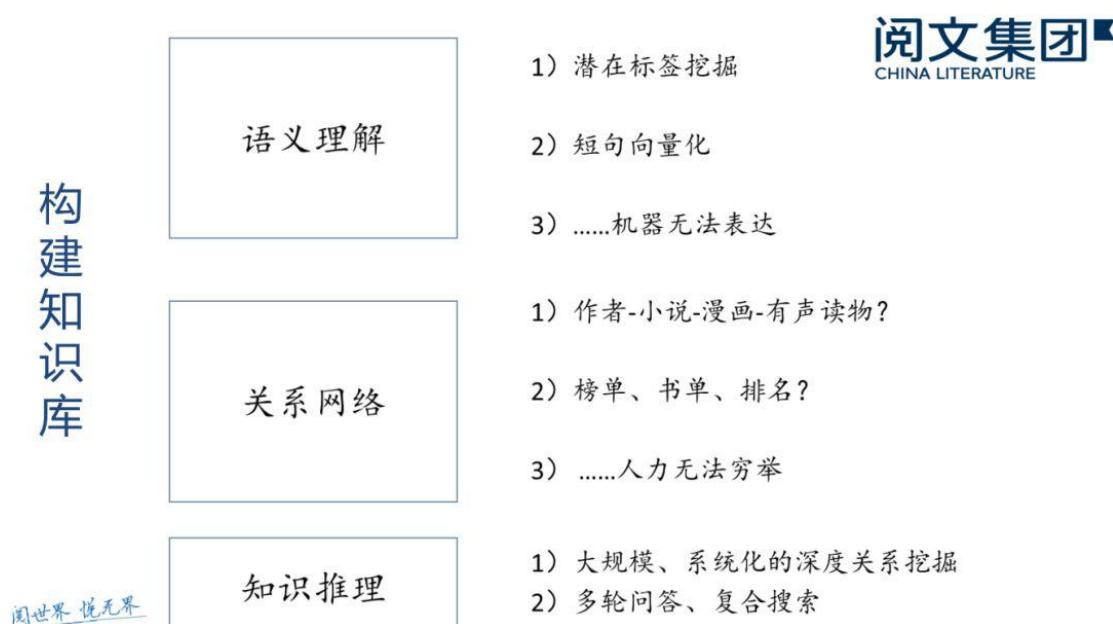
技术主要包含底层数据、核心技术、基础算子、应用策略和业务场景五层。如下图所示。



需要说明的是，基础算子中的书层次，需要依赖段落与章节粒度的分析，却又与这两个粒度的分析方法不同，是独立的端到端模型。

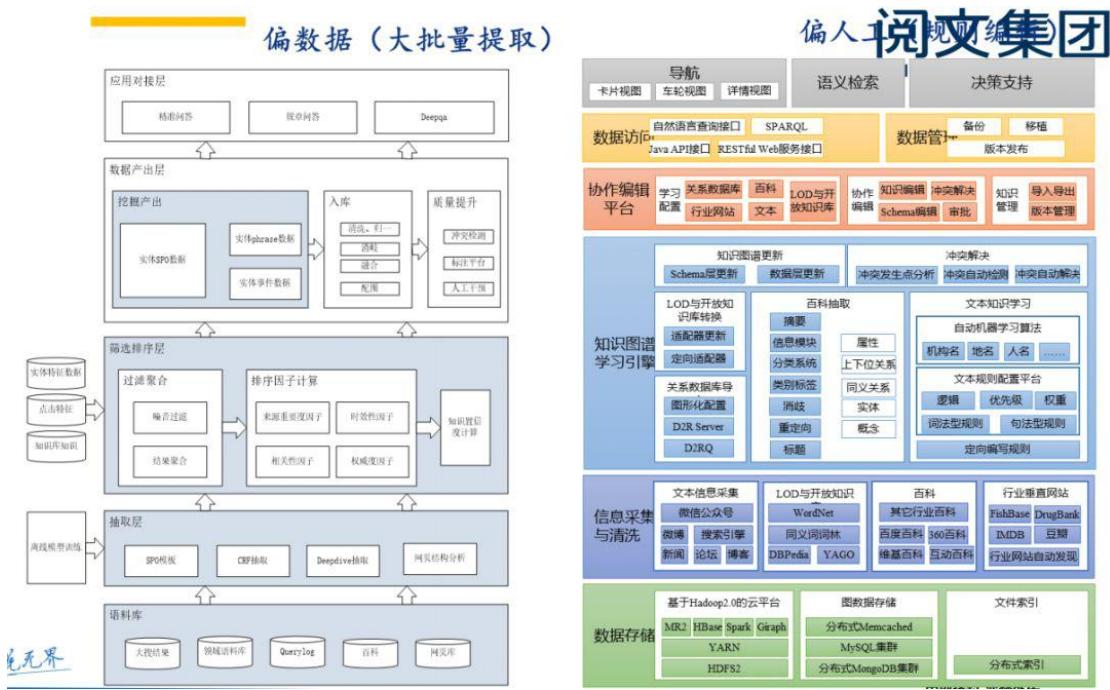
2.3 知识库构建

知识库主要用于辅助语义理解、关系网络构建和知识推理。知识库可以辅助网络内容进行语义理解，并希望把这些知识库固化下来进行迭代更新。以及相应的角色与角色之间的关系，把关系网络建立起来。当需要大规模、系统化的深层关系挖掘时，可通过知识库来支持知识推理。



2.4 知识库构建方法

知识库的构建方法主要有两种，一种是基于数据推理，另一种是基于人工构建。人工构建方法比较简单，而基于数据推理的方法则需要大量的算法辅助。



三、落地实践

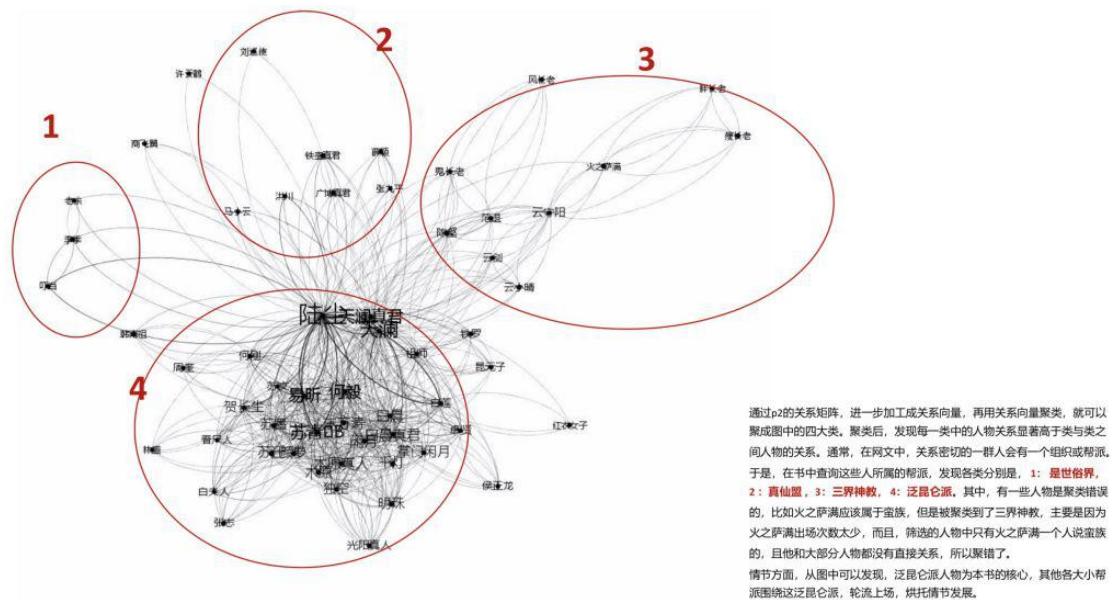
业务落地场景主要有角色分析、标签建设、推荐语生成、色情鉴别和抄袭鉴别五种场景。

3.1 角色分析

角色分析主要通过 NER 加关系抽取进行分析。NER 主要是书籍主角识别，关系为人物关系和书籍角色关系。书籍主角名识别最简单的一种方法是通过关键词+词性+百家姓来分析角色，这种简单的方法就可以达到很高的准确率 (95.6%)。另外主角的出现次数是远多于其他角色，其他角色的次数呈现阶段性下降，通过这种方式可以确定主要人物、重要人物、一般人物等。

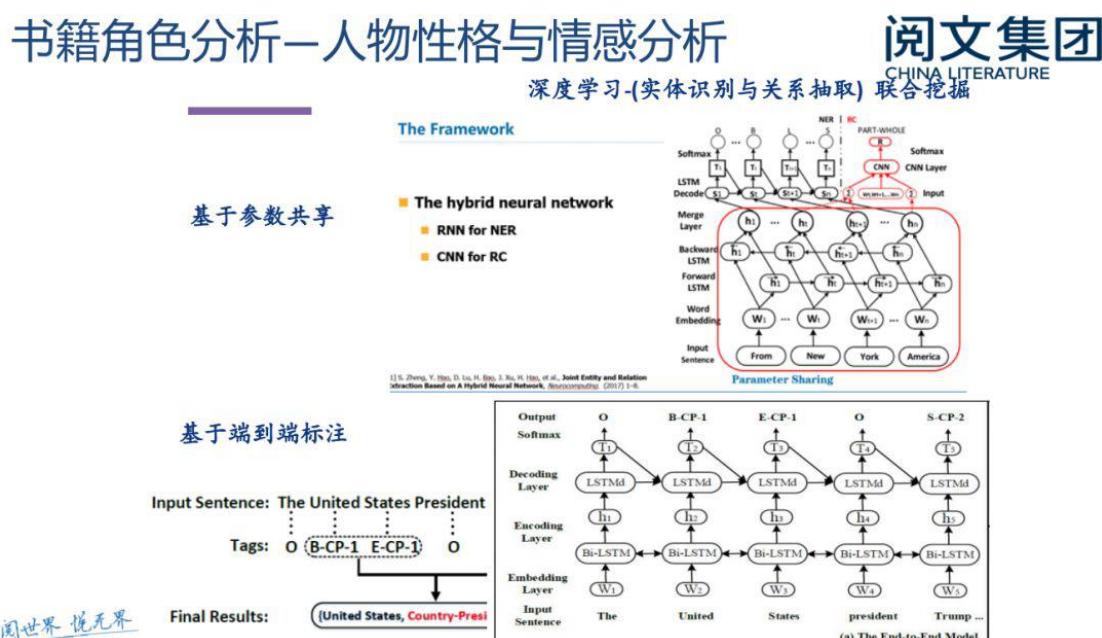
人物关系通过社交关系反应。所谓社交关系，定义为两个人有过对话、打斗，即存在社交关系。社交广泛程度通过社交比例进行量化，与人物 A 有社交关系的所有人除以书中人物总数，即为人物 A 的社交比例。将每一次对话、打斗记为一次关系(可以累加)，可以通过这种关系构建人物关系矩阵。有了这个关系矩阵，就可以进一步构建人物关系图并分析人物关系。

从人物关系矩阵中，可以发现每个人物之间存在的一些联系和冲突，然后通过统计人物贡献周围的一些词是正向还是负向来判断人物是正面人物还是反面人物。通过人物关系矩阵，进一步加工成关系向量，再用关系向量聚类，就可以聚成图中的四大类。



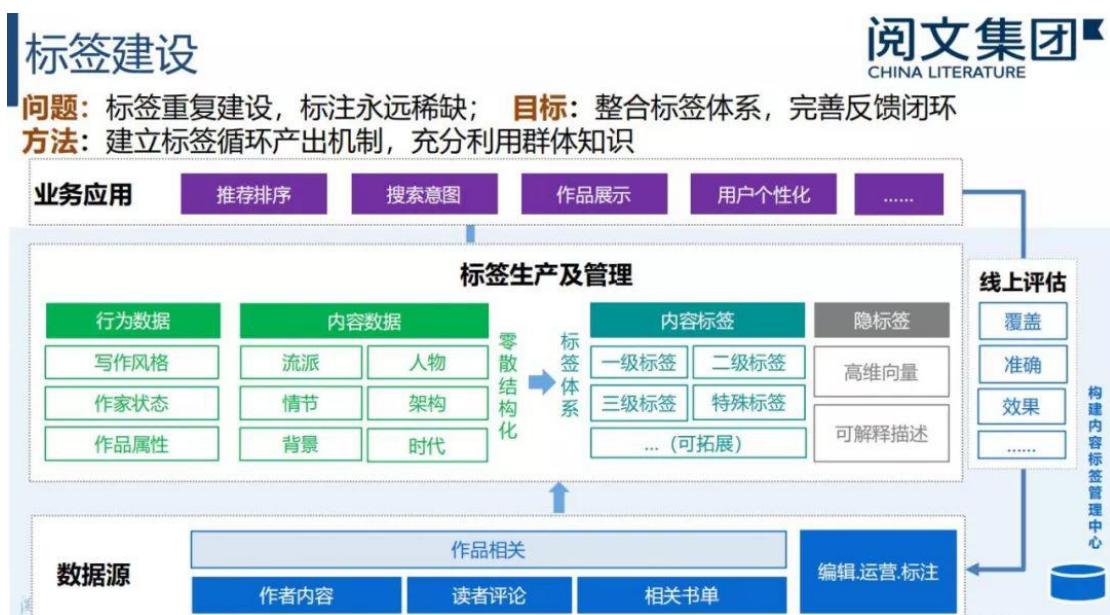
聚类后，发现每一类中的人物关系显著高于类与类之间人物的关系。通常，在网文中，关系密切的一群人会有一个组织或帮派。与此同时，也会出现少量的聚类错误。

通过与主角的对话等，结合情感挖掘方法进行情感分析和预测，使用基于参数共享和端到端标注的深度学习方法对实体识别与关系抽取进行联合挖掘，分析其他角色的人物性格与情感分析。



3.2 标签建设

标签能有效给予读者锚点，让筛选的成本进一步降低，但每本书的标签都是不一样的。与段内容的标签不同，段落中存在一些标签，但是很有可能不置信。网文标签变换非常快，2016年热门标签是校花、兵王，2018年热门标签变成了神豪、奶爸。重要的问题是不太清楚热门标签会不会变化，而且每年都会有新标签出现，如何才能快速对新标签进行融合。第二个问题是标签因为某些书籍而诞生，需要后续慢慢发展而填充进来，很可能在那个时间点样本是相当有限的（就算长期来看，某些标签的样本总量也极低）。由于这些问题，需要对标签进行重复建设，但是数据标注永远稀缺。因此需要整合标签体系，完善反馈闭环。具体的做法是充分利用群体知识，结合已有行为数据和内容数据的标签，通过标签生产和管理生成一些不太确定标签，然后在通过编辑、运营进行标注，再进行标签生成和管理，形成标签产出机制进行循环迭代。



标签的生成主要有两种方法。一种是基于规则产出，缺点是规则不好定义，规则中的词存在歧义，在不同的场景和上下文中有不同的意思。

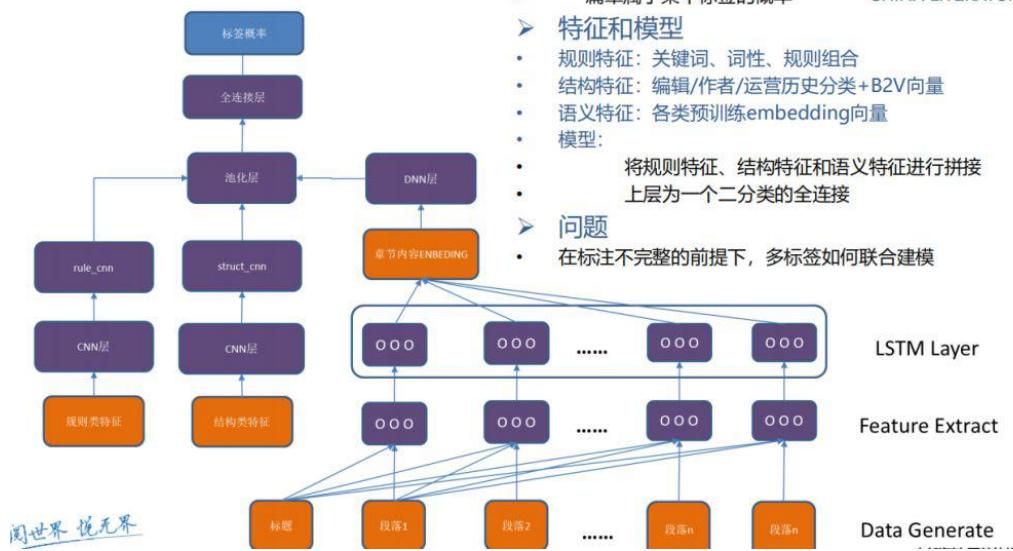


另一种方法是基于相似度产出，这里的相似度主要有两类，一类是语义相似度，包含标签语义向量生成和书籍语义向量生成；另一种是B2V行为特征向量相似度，通过用户行为的相关性对标签进行预测。



结合规则特征、结构特征和语义特征，使用深度学习进行建模。但存在在标注不完整的情况下，多标签如何联合建模的问题。

标签建设



3.3 推荐语生成

生成推荐语的目标是需要覆盖推荐池内的数据，提升转化。解决推荐理由相对单调，信息量低的问题。

结合标签和用户行为数据，推荐语生成有两种方案。一是基于结构化内容模板生成，另一种方案是基于书单已有的推荐语作为训练语料，使用 data2seq 模型生成推荐语。

除推荐语生成外，还可以结合推荐文章生成，热门作家、台词和时间模板等进行更好的推荐。让用户看到不仅仅是推荐、更是一个 AI 的应用场景。

3.4 色情鉴别

色情鉴别主要判断内容是否涉黄、涉政及涉黑等，鉴别方法包含关键词召回和模型召回两种。关键词召回需要定义风险召回关键词和黑名单等。模型召回使用的特征包括规则粒度特征、结构特征和语义特征。规则特征在不同的条件语境下，不同的代词会有不同的指代对象，此时需要很多规则去列举。如不同的穿着和形容词等，有不同的组合，定义好特征规则后，再接入模型进行判断。也可以使用 word2vec 进行特征扩展，但同时也会引入大量的噪音。



3.5 抄袭鉴别

抄袭一般会对关键词和命名实体进行替换。基于这种原因，在做抄袭鉴别时，把句子中的部分关键词和命名实体识别去除，只提取常用词词典中的词，减少命名实体、时间名词的干扰。具体的算法有：

章节拆分：以句子为最小单位，判断不同章节中句子是否有重复。

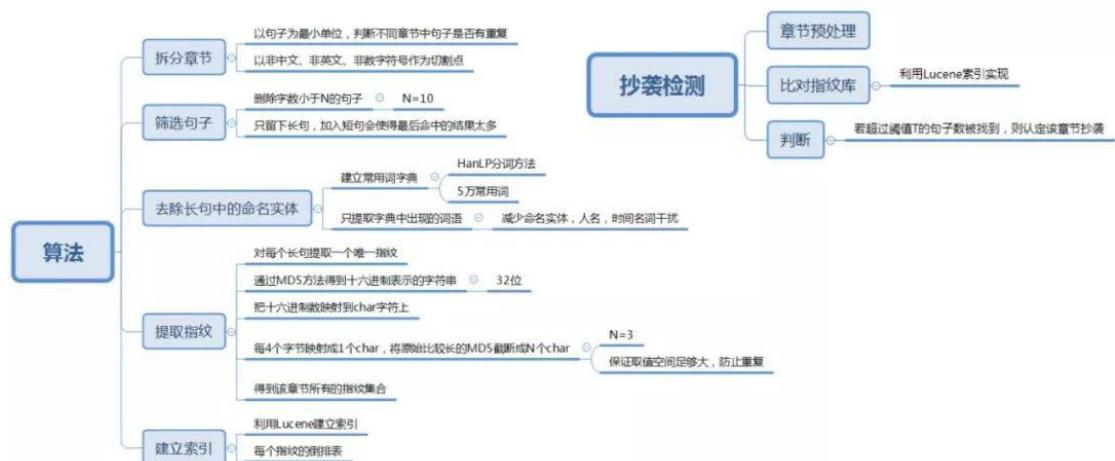
句子筛选：删除短句，只保留长句。原因是加入短句会使得最后的命中结果太多。

去除长句中的命名实体：保留常用词，减少实体词的干扰。

提取指纹：通过 MD5 等，对每个长句提取唯一的指纹，得到该章节的所有指纹集合。

建立索引：通过 Lucene 对指纹建立倒排表。

鉴别时，先对章节进行预处理，利用 Lucene 索引对比指纹库，如果被找到的句子数超过一定的阈值，则认定该章节为抄袭章节。



四、实践总结

技术如何与业务结合。很多时候不能避免返工，但要保证这件事或者方向正确，要对业务问题非常清楚。

如何快速构建正负样本。样本标注不是硬标注，应结合技术手段尽可能减少标注的工作量（例如谷歌流体标注改造），标注尽可能使用二值判断的方式，避免使用从多个选项中选择一个的方式。另一点是配套监控与记录、校验，确保整个标注过程可控。

如何充分利用用户行为。不要觉得用户行为是无效的，用户行为能提供很多信息。文本本身是通过共识达成的，而用户行为记录的是更本质的共识系统。如果业务上会产出用户行为，则优先考虑用户行为贡献的知识。用户行为表明两个 item 相关，就不要单纯从 NLP 语义上去判断说不相关。将行为融入到 NLP 分析模型中，也是后续的发展方向。

作者介绍：

马宇峰，阅文信息 资深研发工程师 内容挖掘平台技术负责人。前百度高级研发工程师，研究方向主要包括知识图谱、用户理解、推荐系统。曾获 2014 百度知识图谱竞赛第 1 名。

内推信息：

岗位名称：文本挖掘工程师

Base：上海

内推邮箱：mayufeng@yuewen.com，欢迎加入阅文集团。

工作职责：

- 1、参与研究自然语言处理中基于词、句、篇章语法与语义分析算法，包括但不限于分词、命名实体识别、角色识别、情感分析等，提供文本分析基础技术；
- 2、运用机器学习与自然语言处理算法开发文本分类、文本自动聚类、信息提取、知识图谱等功能；
- 3、针对阅文海量小说文本，建立小说内容智能分析平台，实现内容智能理解与推荐。

让机器读懂人类：揭秘机器阅读理解技术及应用

作者：邬霄云 整理：Hoh

我们在今年百度举行的中文阅读理解大赛上拿了冠军，而且比第二名高出三个点左右；听到过许多的反馈，好多人都说想要听机器阅读理解的技术方面的内容；今天要讲的东西呢，主要分成三部分，第一部分是介绍问答系统和阅读理解的基本概念，然后跟大家介绍一下比较传统的模块化的问答系统；第二部分呢，是讲一些比较新的一个端到端的系统技术，我们在百度的阅读理解竞赛中夺冠正是使用的这项技术；除此之外本次我还会分享一些我们公司在语音对话交互领域的落地应用和本行业趋势。

一、自动问答与阅读理解

第一个是自动问答和阅读理解，其实从 2017 开始斯坦福大学就开始做这个了，它把这个竞赛变成了一个类似于 ImageNet 一样大家可以刷榜的竞赛，所有的人都在争取第一，但这也同时变相的促进了技术的进步；最早 MSA 周铭老师的队伍，后来科大讯飞和哈工大的老师一起做，成绩也都不错；SQuAD1.0 的时候榜单上最好的成绩是 83% 左右，但是后来斯坦福的专家将数据集进行了更新，进入了 sQuAD2.0 时代，在换了一些似是而非的答案后，很多解决方案都一夜回到解放前，准确率一度讲到百分之六十多；最近不知道大家有没有关注 Google 出了个模型叫做 Bert，我们内部笑称它是大力出奇迹，它的具体原理很简单就是找一个 model 能够融合我所有的数据，然后不管让它自己训练去，Google 资金充足算力强，需要等两周，但对于我们普通公司来说，会等待更长的时间。NLP 是一件比较综合的事情，做好还是比较难的，我们这一行有两个不太好的状况我们现在拿到的 GPU 最早是给 Image 设计的，都是四四方方的，它对图像处理是得天独厚的 NVIDIA 对图像的处理积累了 20 多年，但对 NLP 的作用就不是那么大的，DNN 对 Image 领域的受益是最大的，我们语言不是一个规则的东西，做结构化分析和链式分析都不是很好做，这些事情在 GPU 上运行是非常非常难受的，需要做很多额外的预处理操作；回到 Bert，它真的是大力出奇迹，它就是硬算，不考虑其他任何东西，然后结果竟然非常好。

1.1 模块化问答系统

问答系统在自然语言处理里其实已经有很多很多年了，他是一个 NLP 的标杆性应用，我们在实际的生活中也选用问答来查看对象是否真的理解，比如说老师对学生的问答，可以查看学生们的理解水平；现在问答在 NLP 里面有很多作为核心技术的应用，比如说智能客服和手机上的虚拟助手，比如说苹果 Siri 或百度的小度、微软小冰等等；下面简单介绍一下问答系统，我们可以把问题简单的分成几类，一类是简单的事实，比如地球直径多大；第二类是定义的事实类，第三类是列表类，第四类是长答案类，我们可以看到网上有许多很长的答案，最后一类为是非类，看着很简单实际上这是最难的一类问题，下图是业界常见的系统。

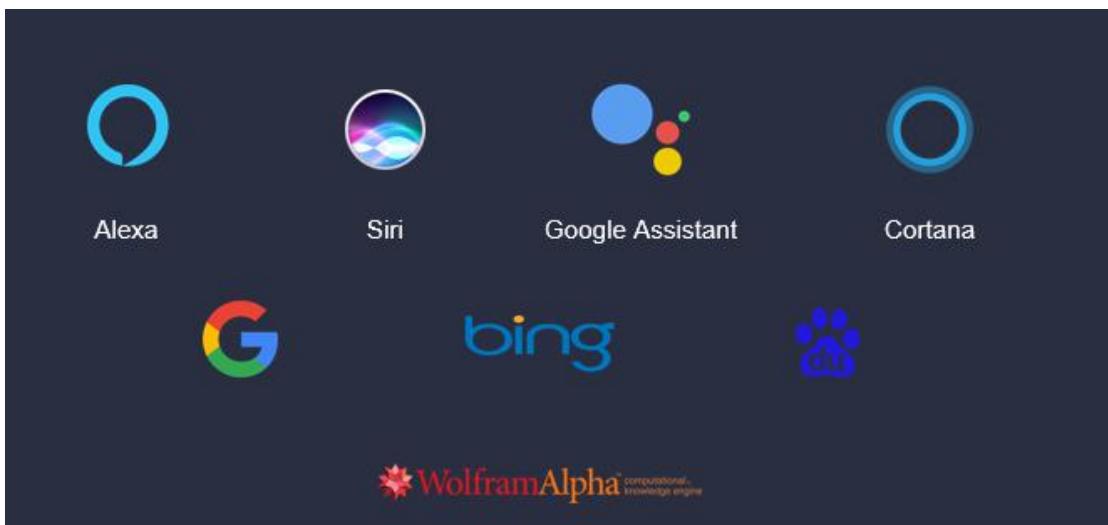


图 1 实用问答系统的举例

1.2 传统化回答方法

下面我们看看模块化问答系统的回答方法，它们有很多种，比如说有特制服务的问答，比如说有一个叫做 WolframAlpha 这个专门基于数学问题的网站，你可以在上面搜索任何数学问题。

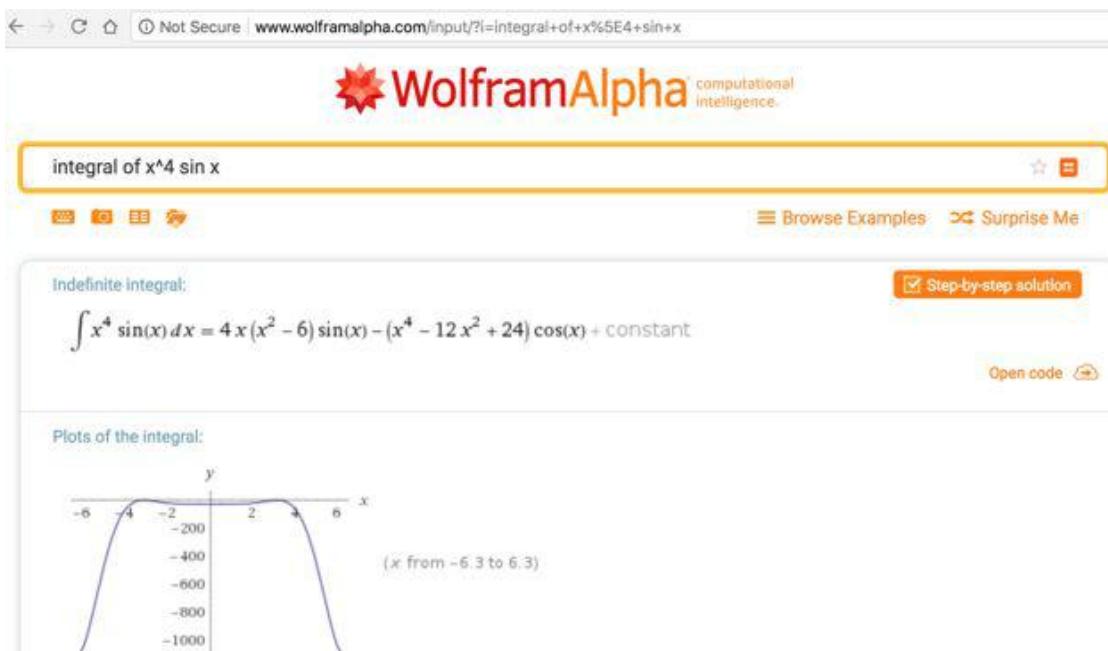


图 2 数学计算网站的特制问答

最近还有基于知识图谱的问答和基于搜索的自动问答，比如说微软的这个应用，用户问一个问题“明天需要带伞吗？”然后 Bing 除了去搜索文档外还会有一个专门回答问题的 bot，将把你当前位置的地理位置的天气给显示出来，但是这个问答系统的更新方式是非常落后

的，需要经常维护更新知识库。然后我们介绍一下基于搜索的问答系统的结构，下图是一个比较传统的结构它是一个模块化的系统，第一步，有了用户的 query 以后，我们并行计算一边对 query 进行分析，另一边将其送入 google 的搜索引擎找到相应的 docs，然后我们在其中根据 query 匹配答案；阅读理解跟基于搜索的问答非常像，只不过这个问答不用你自己找，这个文章是已经作为另一个输入给系统了，等于说是先给定几篇文章，然后给出问题，让系统在文章中找出答案，这里的做法与前面很像仅仅知识把搜索引擎去掉了。

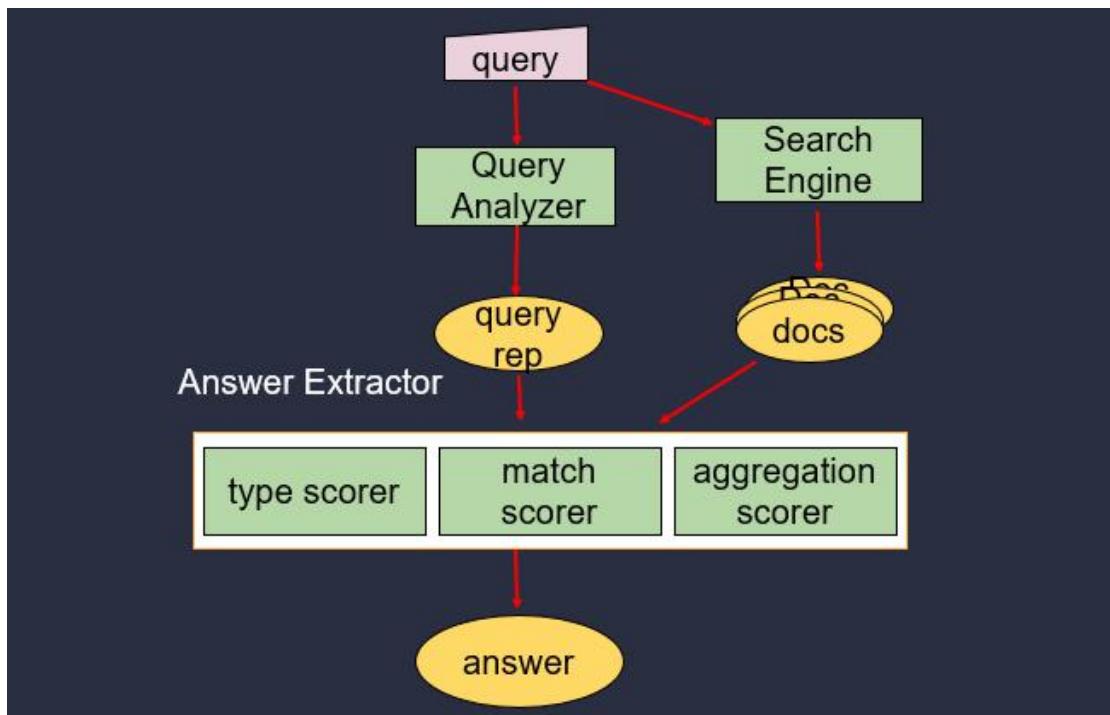


图 3 基于搜索的自动问答

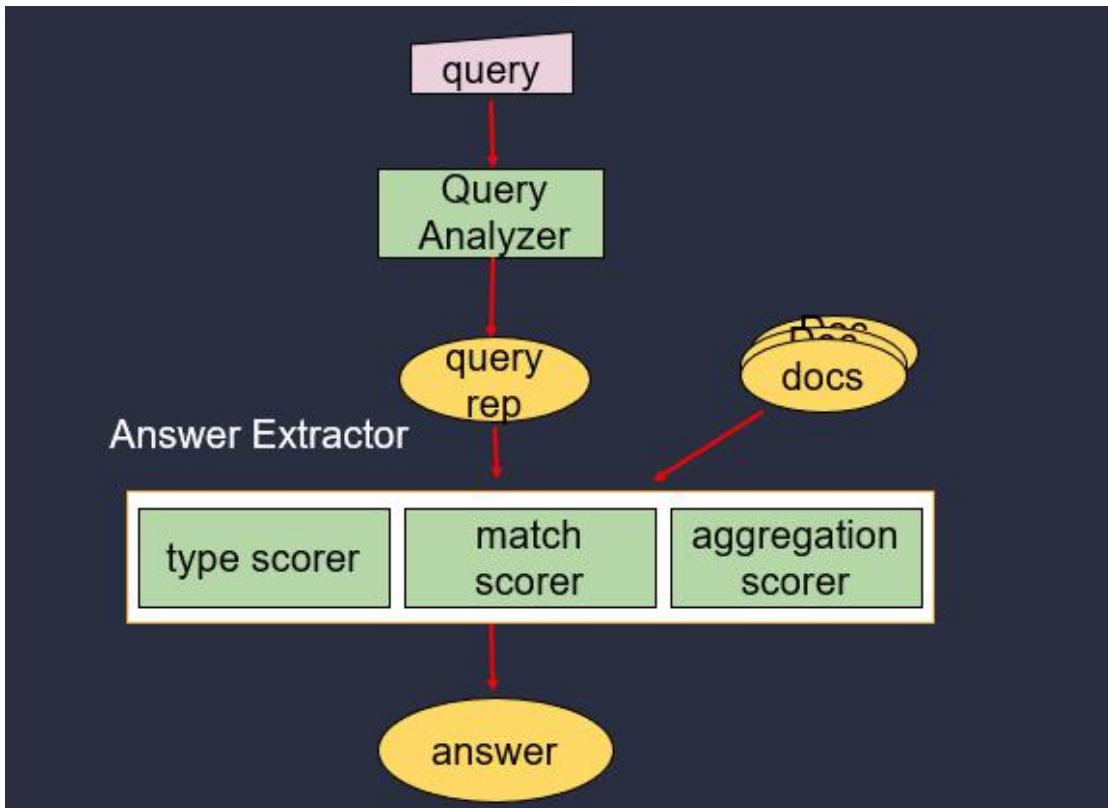


图 4 传统阅读理解框架流程图

二、端到端阅读理解

2.1 阅读理解数据集

接下来是端到端阅读理解，阅读理解数据集 sQuAD 我之前已经讲过，它是以 Wikipedia 页面中的段落作为来源文档，从这里拿过来以后，根据文档人工编写问题，这个文档我们可以问什么问题，可能怎么问？制造了比较大的数据集，答案是文档里面的一个连续片段（span）。还有一个不太有名也比较复杂的数据集是 MS MARCO，它的问题来自搜索引擎真实用户提出的问题，相关文档也是信息检索系统从真实网页得到的段落，每个问题对应多个段落，因为标注答案是人工根据文档总结撰写，而成这就增加了复杂度，他和 sQuAD 的区别是他的答案不是一个 span，还需要做一些提取和综合答案，因为比较复杂，所以做的人不是很多，百度前段时间在这个上面拿了第一，但是普适性不是很好。具体怎么做呢？

2.2 端到端阅读理解系统

首先我讲一下端到端的阅读理解系统的基础架构，如图所示前面是一个模块，后面我们把整个的各种各样的模块都放进一个神经网络里面进行训练；

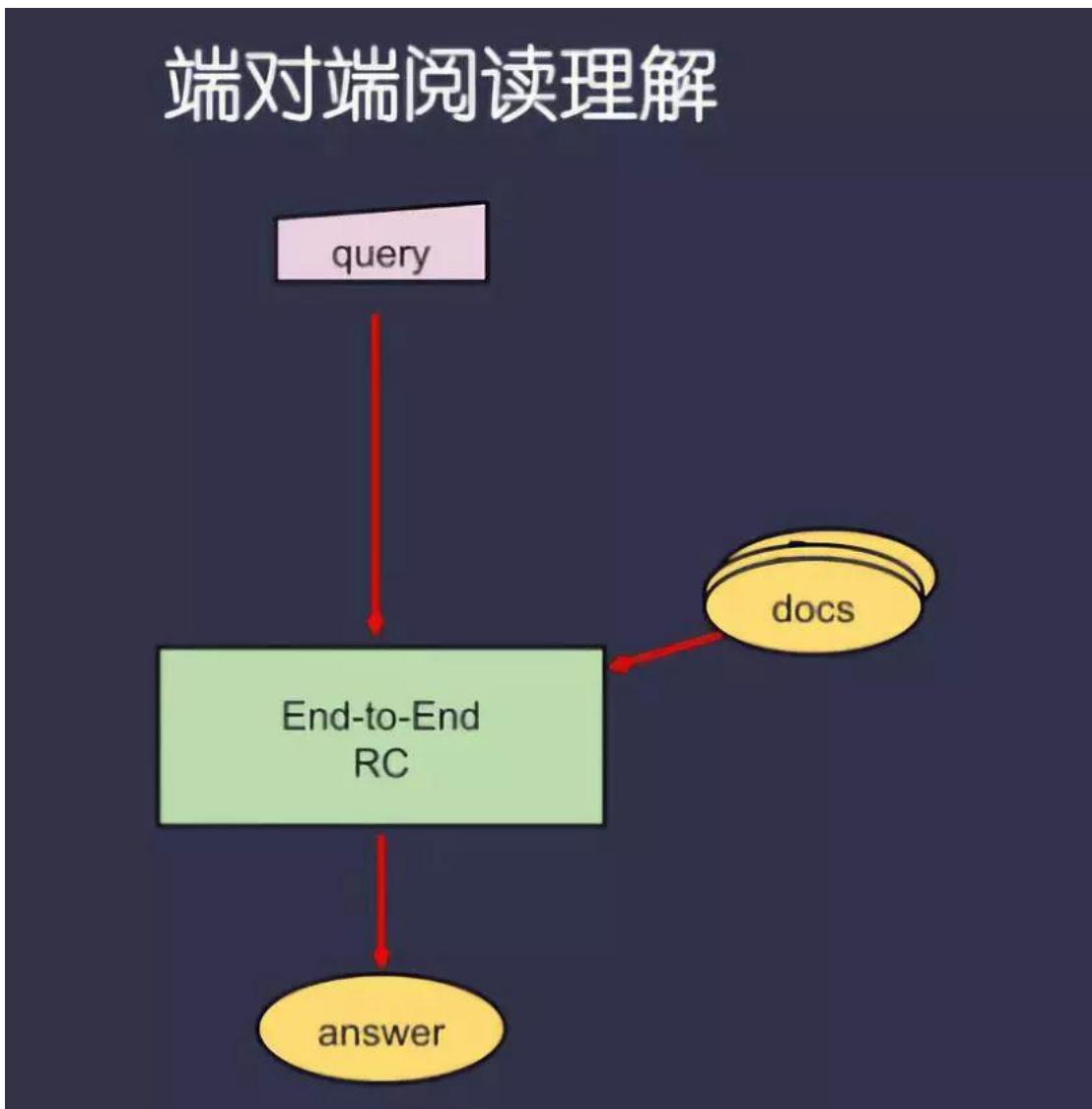


图 5 端到端阅读理解流程图

比如用户问个问题“什么酶可以分解淀粉？”回答这个问题时，第一步，找到问题中的焦点词 Focus words，焦点词分为显性和隐性两种，它声明了答案的类型；第二步骤是在文章中将那些有可能是答案的东西给找出来，这里有个概念是回答的类型，它与焦点词是非常相关的，它主要是找到焦点词所指的类型是什么，这里我们要知道如何运用算法才能识别这个预期答案类型，同样的问答类型的识别可以用粗颗粒答案类型和直接使用 query 中的焦点词作为答案的类型。

2.3 模型整体结构

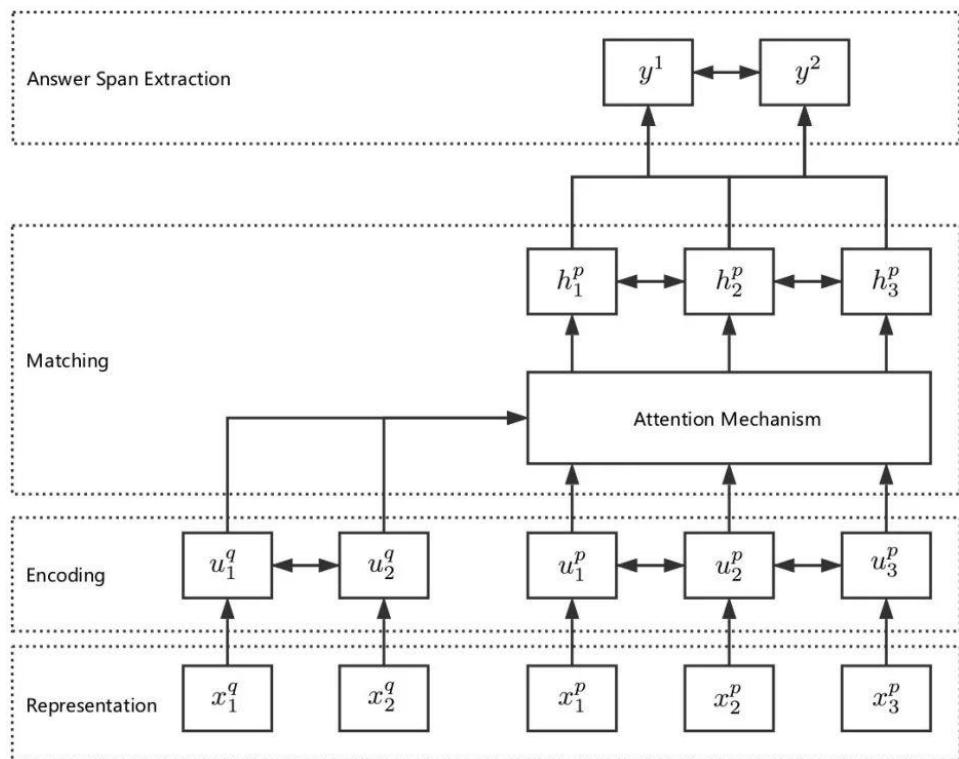


图 6 模型整体结构示意图

我们来看一下模型的整体结构，我不负责写代码，我从宏观上介绍一下，几乎所有的模型分成四块：最底层是 Representation（特征表示层），作用是看这个词在这个场景中是什么意思，确定出问题的类型，将问题和篇章的词语转化为向量化特征表示并进行此行的标注；接下来是编码层，主要是衔接底层的特征；然后是匹配所有问题的匹配层，要想找到所有的答案，我们要在 model 中把问题的信息通过这个机制对每一个字进行重新表示，利用注意力机制融合问题和篇章信息，我们用的模型是 Match-LSTM、BiDAF 和 DCA 等；最后我们再通过一种对应准则把每一个字的新表达 span 给找出来，也就是答案片段抽取层负责的工作，这时我们就可以利用两个步的指针网络对组成答案的 Span 进行相应的提取操作，相应的细节大家可以再 paper 上面找，我这里就不详细讲了。如果大家想做这个方面的问答技术实现，我建议大家先跑通这个模型，然后再做一些微调，阅读大量文献找到感觉后再进行创新和改进，从一个好的点到一个更好的点是需要循序渐进的，我认为这样的效率会更高些。

三、语音交互技术和行业发展趋势

我们公司要做的事情和现有的大家的想法可能不太一样：虽然大家都知道人工智能的三大要素是算法+数据+算力，不过我认为用户体验是最重要的粘合剂，有了这个粘合剂，才能使

AI 真正的落地。对话的用户体验是一个更自然的体验，我对和个对话用户体验很认同。因为对任何人来说最重要的事情就是时间；因为这个时间都是 24 小时，时间过去了再也不会回来，我们想一下，假如我们想要吃一个麻辣烫，这个东西我们都知道怎么做，这是个很简单的问题，但是几乎每一个人第一次用美团都需要相当长的适应时间，对于这个学习成本是很高的；也就是说我们是知道自己要做什么，但我们希望语音对话直接帮忙完成相关操作，学习 App 如何使用其实是我们并不关心的。我们年轻人来说还好，但是对于我们的父母和更老的爷爷奶奶来说这件事情还是很有挑战的，我们真正关心的事情是我饿了，我想买东西吃，我们并不关心 App 是在哪里？怎么用？如何选取优惠现在慢慢的大家都开始关注小程序了，小程序是一个小热点，非常 popular 的东西，人们终于发现了 App 并不具有太大的吸引力了，有了小程序，可以慢慢的替代 App 了，人的一辈子是单次路程，我们小学毕业一次，中学毕业一次，大学毕业一次，我们去某地旅游，去一个餐馆吃一次饭，可能就吃这么一次，再也不会去了。同理，我们手机上安装六七十个 App，但我们一天能用几次呢？会不会嫌麻烦不用了呢？我们公司的愿景就是期待帮助当用户知道自己要做什么时，我们可以让他们通过一句话搞定复杂操作，而不需要去想怎么操作图形界面而浪费时间和精力。

语音交互两部分，一个是本身，二是商务逻辑；我们公司本身，关注的是商务逻辑，如果想要语音交互和图形界面交互一样产生很棒的体验，那么就要实现任何一个程序员都可以很方便的把他擅长的领域中的体验给做出来，这一点很重要，但是过去国内外企业对这方面积累比较少；这个语音交互其实对社会和人类产生很多好处，不过它是不好实现的技术细节比较麻烦；音交互将会成为未来主流的人机交互方式之一，技术更新迭代大爆发即将来临，我希望我们公司能够成为时代的弄潮儿。关于语音体验的优化，第一件事从表达到操作，就像头痛和头疼其实是一件事，关键在于我们如何将表达同一含义的说法进行泛化？第二件事是我根据这些信息，如何让商业逻辑容易的表达出来，如何高效的用声音来控制图形界面操作和业务，我们会有很多很多后续的技术分享和成果发布，请大家保持关注。

作者介绍：

邬霄云，Naturali 奇点机智创始人兼 CEO，纽约州立大学计算机博士，拥有 1 年雅虎实验室、8 年谷歌研究院工作经验，回国创业前负责美国应用搜索公司 Quixey.com 搜索部，专攻自然语言处理、深度学习、互联网大数据、分布式计算领域。

内推信息：

对 NLP 领域感兴趣的朋友，欢迎投简历到 jobs@naturali.io，2018 机器阅读理解技术竞赛冠军团队期待你的加入！

自然语言处理中的多任务学习

作者：邱锡鹏 整理：靳韓贊

本次报告内容的题目是自然语言处理中的多任务学习，报告主要分为四个部分：

- 1、基于深度学习的自然语言处理；
- 2、深度学习在自然语言处理中的困境；
- 3、自然语言处理中的多任务学习；
- 4、新的多任务基准平台。

首先简单介绍一下实验室情况，课题组主要聚焦于深度学习与自然语言处理领域，包括语言表示学习、词法/句法分析、文本推理、问答系统等方面。开源自然语言处理系统 FudanNLP，并将在 12 月中旬推出全新的 NLP 系统：fastNLP。

研究组介绍



- } 主要聚焦于深度学习与自然语言处理领域，包括语言表示学习、词法/句法分析、文本推理、问答系统等方面。
- } 主要成果
 - } 近几年发表国际顶级会议/期刊（IJCAI、ACL、AAAI、EMNLP等）论文 50余篇，ACL2017杰出论文
 - } SQuAD2.0第二，SQuAD1.1多次第一
 - } 开源自然语言处理系统：
 - } FudanNLP：国内最早的开源NLP系统之一
 - } fastNLP：一个全新的系统！



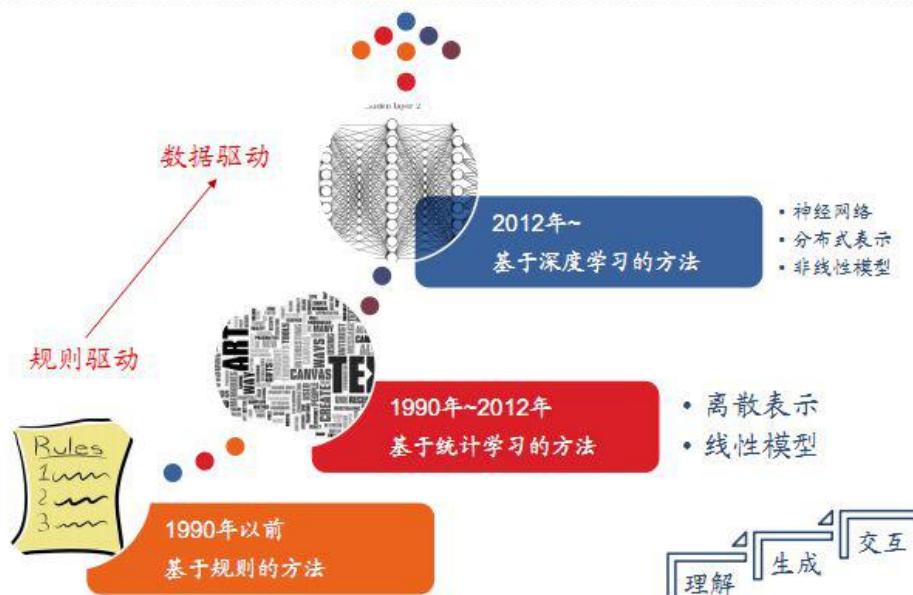
一、自然语言处理简介

自然语言处理就像人类语言一样，与人工语言的区别在于它是程序语言，自然语言处理包括**语音识别、自然语言理解、自然语言生成、人机交互**以及所涉及的中间阶段。下面列举出了自然语言处理的基础技术、核心技术和一些应用：

基础技术：词法分析、句法分析、实体识别、语义分析、篇章分析、语言模型；
核心技术：机器翻译、自动问答、情感分析、信息抽取、文本摘要、文本蕴含；
应用：智能客服、搜索引擎、个人助理、推荐系统、舆情分析、知识图谱。

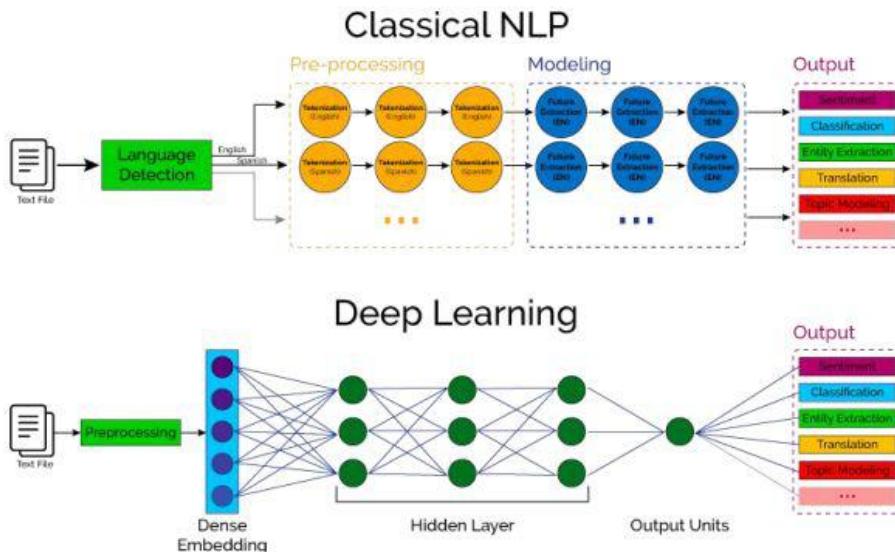
自然语言处理最初由规则驱动，逐步发展为数据驱动。

发展历程



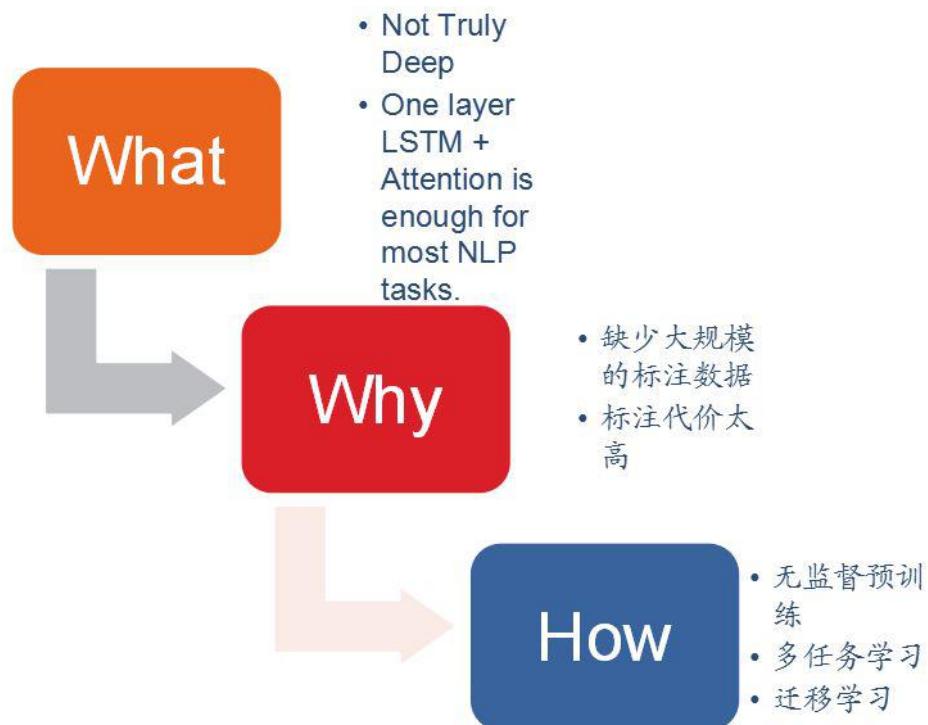


NLP技术路线



二、深度学习在自然语言处理中的困境

由于缺少大规模的标注数据或者标注代价太高，目前大部分用在 NLP 上的神经网络都不是很深，一般情况下，一层 LSTM+Attention 就足够完成大部分 NLP 任务。解决问题的方法包括有无监督预训练、多任务学习和迁移学习。今天我们主要介绍多任务学习。



1、无监督预训练

首先我们来介绍一下 NLP 中非常重要的无监督预训练，早期有很多研究者使用词向量等词级别的模型，后来发展为使用句子级别的模型，例如最近出现的 ELMo、OpenAI GPT、BERT 等，人们从最初学习更好的词的表示转变为学习更好的句子的表示。



无监督预训练

} 词级别

↳ 语言模型

↳ Word2Vec (CBOW and Skip-Gram)

↳ GLOVE

↳ FastText

} 句子级别

↳ Skip-Thought

↳ Paragraph Vector

↳ ...

↳ ELMo: Embeddings from Language Models

↳ OpenAI GPT

↳ BERT

论文 Deep Contextualized Word Representations 主要描述的是 ELMo 问题，通过建立两个双向的 LSTM 来预测一个前向、正向的语言模型，然后将它们拼起来，这个模型是一个非常好的迁移模型。

NAACL 2018 Best Paper
Deep contextualized word representations

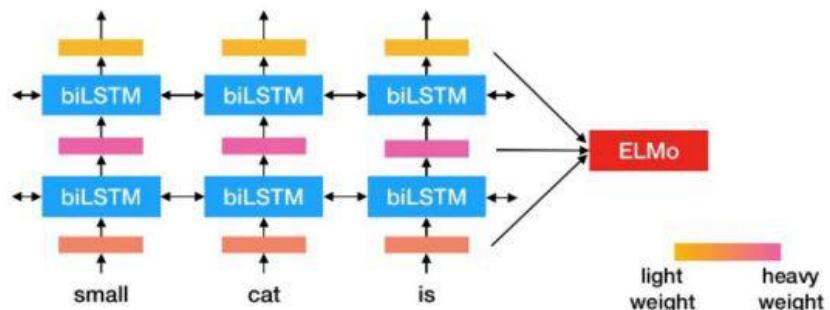
Matthew E. Peters[†], Mark Neumann[†], Mohit Iyyer[†], Matt Gardner[†],
 {matthewp, markn, mohiti, mattg}@allenai.org

Christopher Clark*, Kenton Lee*, Luke Zettlemoyer^{†*}
 {csquared, kentonl, lsz}@cs.washington.edu

[†]Allen Institute for Artificial Intelligence
 *Paul G. Allen School of Computer Science & Engineering, University of Washington

ELMo: Embeddings from Language Models

$$\text{ELMo}_k^{task} = E(R_k; \Theta^{task}) = \gamma^{task} \sum_{j=0}^L s_j^{task} \mathbf{h}_{k,j}^{LM}.$$



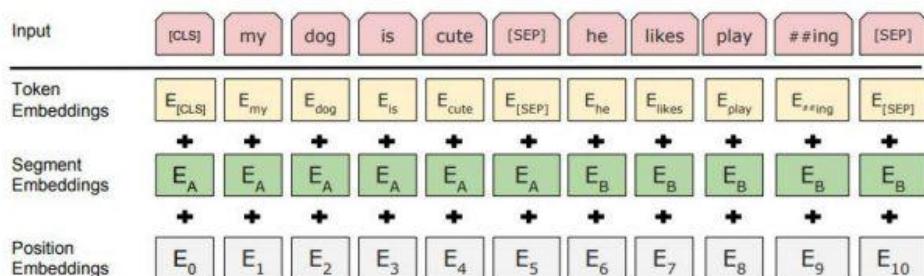
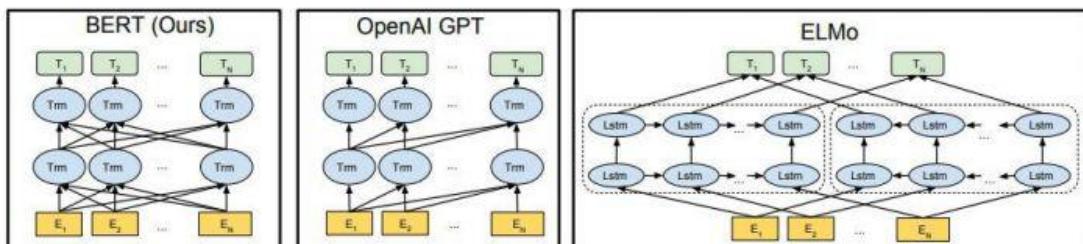
<https://arxiv.org/pdf/1802.05365.pdf>

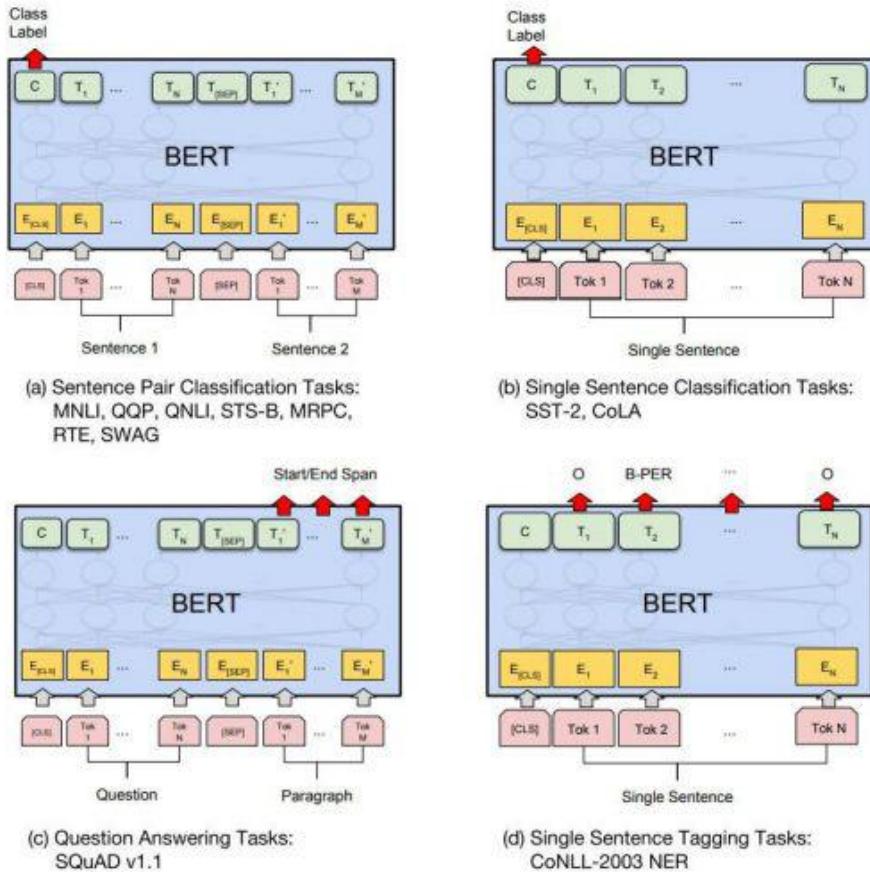
谷歌新推出的 BERT 是将机器翻译中的常用模型 transformer 的双向训练用于建模 , 它在很多任务中取得了较好的效果。

BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding

Jacob Devlin Ming-Wei Chang Kenton Lee Kristina Toutanova
 Google AI Language

{jacobdevlin, mingweichang, kentonl, kristout}@google.com





这些模型证明在 NLP 中表示学习依然十分重要，表示学习是从文本输入到计算机内部的一种表示，对于 NLP 任务，表示学习是指将语义信息表示成稠密、低维的实值向量。表示好之后送到分类器中，好的表示是一个非常主观的概念，没有一个明确的标准。一般而言，好的表示具有以下几个优点：

- 1) 应该具有很强的表示能力，模型需要一定的深度；
- 2) 应该使后续的学习任务变得简单；
- 3) 应该具有一般性，是任务或领域独立的。



在NLP中，表示学习依然十分重要！

} 什么是好的文本表示？

 } “好的表示”是一个非常主观的概念，没有一个明确的标准。

} 一般而言，一个好的表示具有以下几个优点：

 } 应该具有很强的表示能力。

 } 模型需要一定的深度

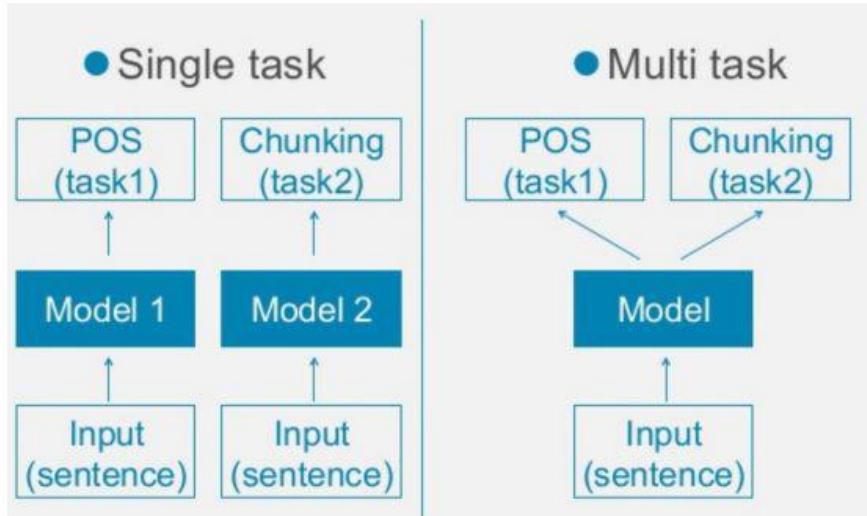
 } 应该使后续的学习任务变得简单。

 } 应该具有一般性，是任务或领域独立的。

2、多任务学习

下面给出一个多任务学习的例子，对于两个单独的任务训练两个模型，对于任务 1 训练一个模型 1，对于任务 2 训练一个模型 2，多任务就是将两个任务放在一起用一个模型来处理。

A NLP Example



多任务学习最早在 97 年被提出，多任务学习隐含着从其他任务中学习一种共享的表示，共享表示可以作为一种归纳偏置，归纳偏置可以看做是对问题相关的经验数据进行分析，从中

归纳出反映问题本质的模型的过程，不同的学习算法（决策树、神经网络、支持向量机）具有不同的归纳偏置，在学习不同的任务过程中使用共享表示，可以使在某个任务中学习到的内容可以帮助其他任务学习的更好。

Multitask Learning*

Machine Learning 1997

RICH CARUANA

School of Computer Science, Carnegie Mellon University, Pittsburgh, PA 15213

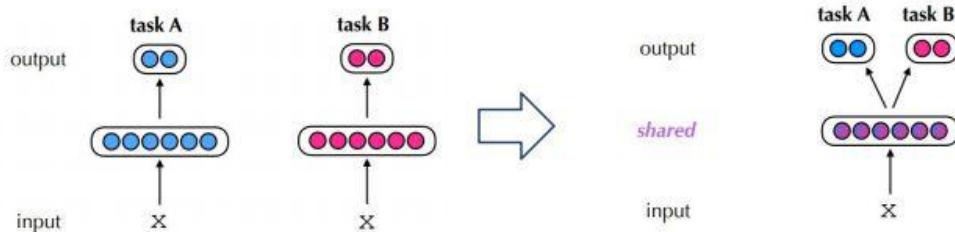
caruana@cs.cmu.edu

Multitask Learning is an approach to **inductive transfer** that improves **generalization** by using the domain information contained in the training signals of related tasks as an **inductive bias**. It does this by learning tasks in parallel while using a **shared representation**; what is learned for each task can help other tasks be learned better.

由于传统 NLP 的表示空间是离散的，MTL+NLP 在传统的 NLP 模型是非常难实现的，随着深度学习的应用，整个 NLP 的表示空间变为连续的，使得任务实现更加容易。例如下图中 taskA 和 taskB 两个任务可以共享同一个模型。



多任务学习+深度学习

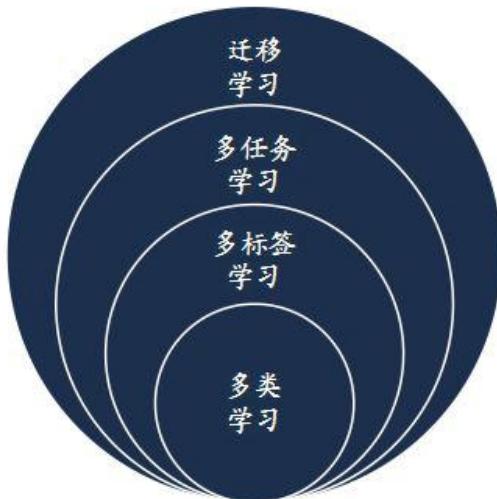


Neural network based approaches make MTL particulary attractive/easy

不同学习范式之间的关系：多任务学习之上有迁移学习，之下有多标签学习和多类学习。



不同学习范式之间的关系



- } 迁移学习 Transfer Learning
 - } 在源领域上学习模型
 - } 泛化到目标领域上
- } 多任务学习 Multi-Task Learning
 - } 同时建模多个相关任务
 - } 不同任务有不同的数据和标签
- } 多标签学习 Multi-Label Learning
 - } 一个样本可以有多个标签
 - } 建模标签之间的关系
- } 多类学习 Multi-Class Learning
 - } 一个样本只能属于一个标签

损失函数：假设有 m 个任务，多任务学习的损失函数是将各个任务的损失函数相加求得联合损失函数 joint loss。



损失函数

} 损失函数

$$-\frac{1}{m} \sum_{i=1}^m \sum_{j=1}^k \left(y_j^{(i)} \log \hat{y}_j^{(i)} + (1 - y_j^{(i)}) \log (1 - \hat{y}_j^{(i)}) \right)$$

Joint Losses?

训练方式：首先进行 Joint Training , Training 之后进行 Fine Tuning。



训练方式

} **Joint Training:** The training is achieved in a stochastic manner by looping over the tasks:

- } 1. Select a random task.
- } 2. Select a random training example from this task.
- } 3. Update the parameters for this task by taking a gradient step with respect to this example.
- } 4. Go to 1.

} **Fine Tuning:** After the joint learning phase, we can use a fine tuning strategy to further optimize the performance for each task.

23

多任务学习工作的优点：

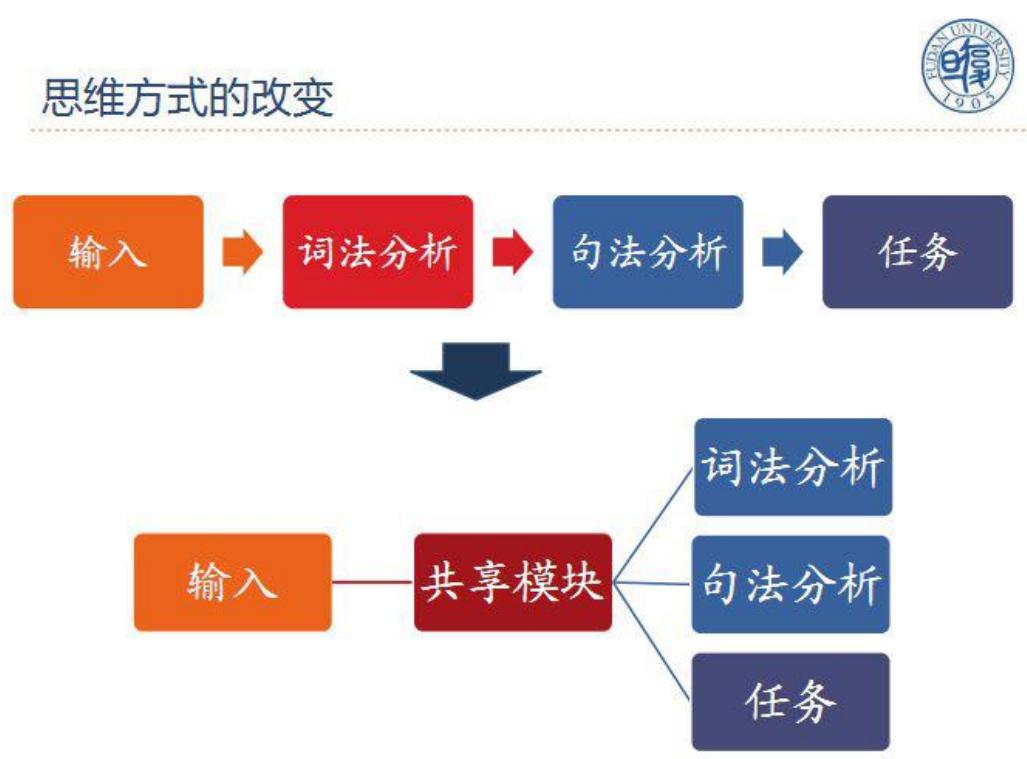
1) 隐式的数据增强 :一个任务的数据量相对较少，而实现多个任务时数据量就得到了扩充，隐含的做了一个数据共享。

- 2) **更好的表示学习** : 一个好的表示需要能够提高多个任务的性能。
- 3) **正则化** : 共享参数在一定程度上弱化了网络能力 , 防止过拟合。
- 4) **窃听** : 某个特征很容易被任务 A 学习 , 但是难以被另一个任务 B 学习 , 这可能是因为 B 以更复杂的方式与特征进行交互或者因为其它特征阻碍了模型学习该特征的能力。通过 MTL , 我们可以允许模型窃听 , 即通过任务 A 来学习该特征。

目前 NLP 中每个任务只做其中的一块 , 如果我们把这些任务拼起来会取得更好的效果。

三、自然语言处理中的多任务学习

下面介绍几种多任务学习的方式 , 传统的自然语言处理在输入端输入文本 , 之后进行词法分析和句法分析最后完成任务 , 这种方式很难实现 , 在有了多任务学习之后 , 不同的任务可以共享词法分析和句法分析模块 , 自然语言处理的方式得到了简化。



自然语言中的多任务学习包括有 : **多领域任务、多级任务、多语言任务、多模态任务等**。



自然语言中的多任务学习

多领域 (Multi-Domain) 任务

- Multi-Domain Text Classification
- Multi-Domain Sentiment Analysis

多级 (Multi-Level) 任务

- part-of-speech (POS)
- tagging, named entity recognition (NER)
- semantic role labeling (SRL)

多语言 (Multi-Linguistic) 任务

- Machine translation
- Multi-lingual parsing

多模态 (Multi-Modality) 任务

- Visual QA
- Image Caption

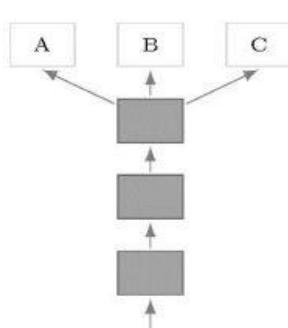
深度学习+多任务学习有硬共享、软共享、共享-私有等多种模式。

硬共享模式：在下面层共享，上层根据自己不同的任务做不同的设计；

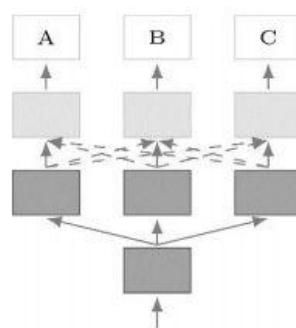
软共享模式：每个任务都有自己的流程，从信息流来看就是从输入到 A 有自己的体系流程，还可以从其他任务的表示方法中拿一些东西过来；

共享-私有模式：一部分共享，一部分私有的信息传递机制。

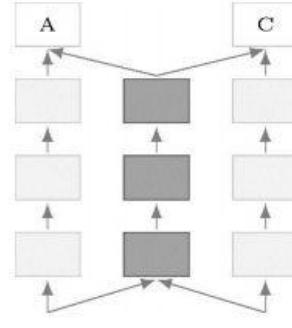
深度学习+多任务学习



(a) 硬共享模式



(b) 软共享模式

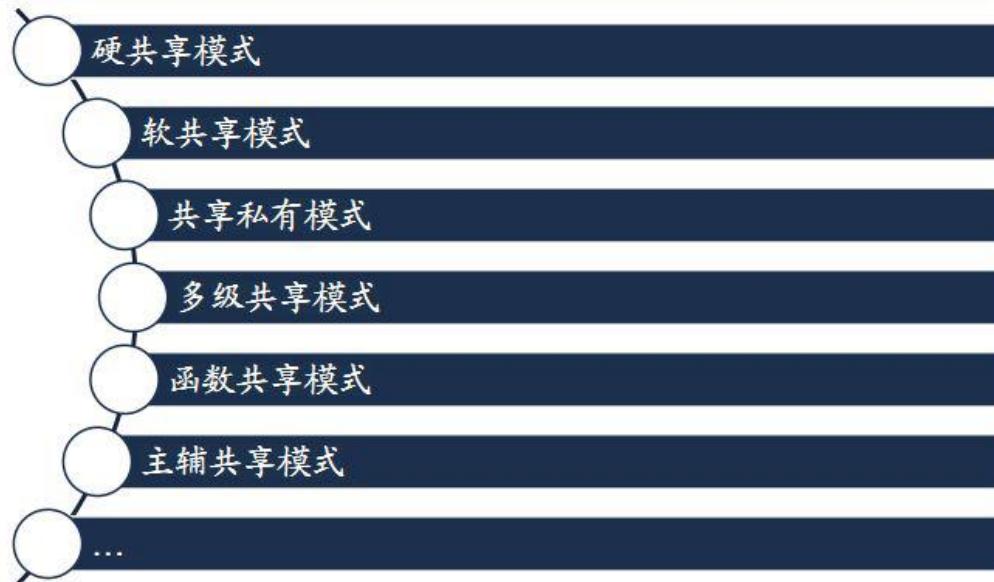


(c) 共享-私有模式

此外还有多级共享、函数共享、主辅共享等多种共享模式，下面将一一介绍。



共享模式



1、硬共享模式

硬共享在下面层共享，上面根据自己的不同的任务来做不同的设计，这种方法最早在 2008 年由 Ronan Collobert 在论文 A Unified Architecture for Natural Language Processing : Deep Neural Networks with Multitask Learning 中提出，应用到了很多与语义相关和语法相关的方面，例如机器翻译、文本分类等。

A Unified Architecture for Natural Language Processing: Deep Neural Networks with Multitask Learning

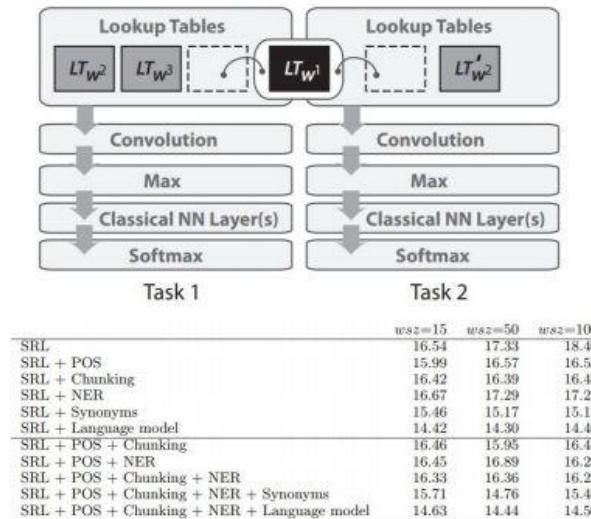
Ronan Collobert

Jason Weston

NEC Labs America, 4 Independence Way, Princeton, NJ 08540 USA

COLLOBER@NEC-LABS.COM

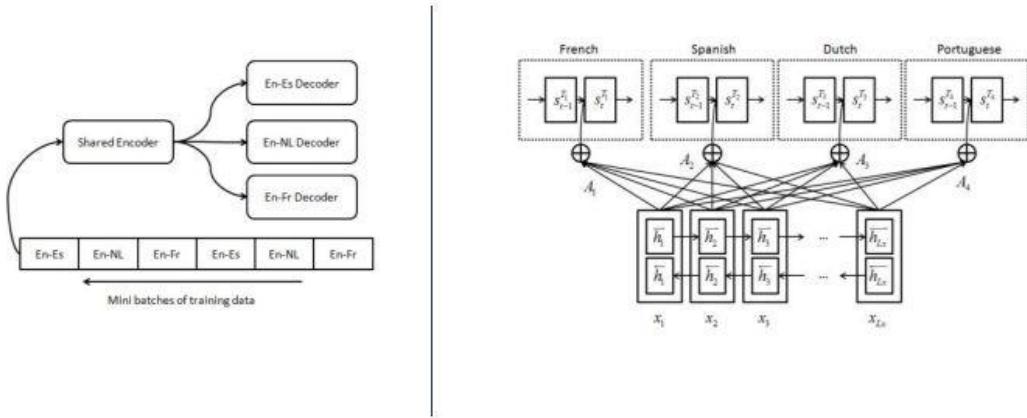
JASONW@NEC-LABS.COM



Multi-Task Learning for Multiple Language Translation

ACL 2015

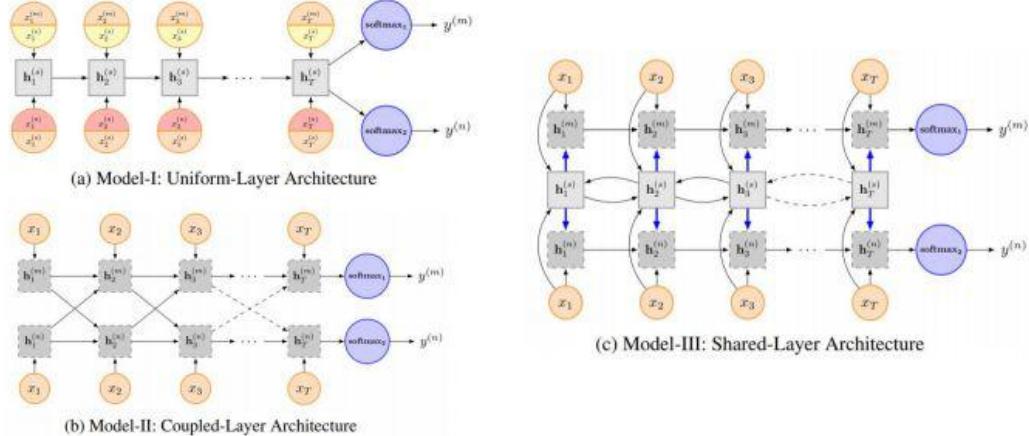
Daxiang Dong, Hua Wu, Wei He, Dianhai Yu and Haifeng Wang
Baidu Inc, Beijing, China



Recurrent Neural Network for Text Classification with Multi-Task Learning

Pengfei Liu Xipeng Qiu* Xuanjing Huang

Shanghai Key Laboratory of Intelligent Information Processing, Fudan University
School of Computer Science, Fudan University



<https://arxiv.org/pdf/1605.05101.pdf>

后来人们将注意力机制模型用于共享模式，注意力机制不需要使用所有的信息，只需要将其中部分信息选择出来，人们基于注意力机制做了共享模式。

Same Representation, Different Attentions: IJCAI 2018 Shareable Sentence Representation Learning from Multiple Tasks

Renjie Zheng, Junkun Chen, Xipeng Qiu*
Shanghai Key Laboratory of Intelligent Information Processing, Fudan University
School of Computer Science, Fudan University

The infantile cart is easy to use,

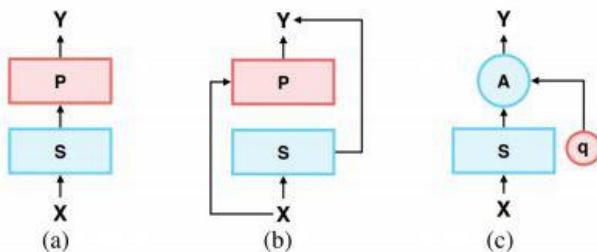


Figure 1: Three schemes of information sharing in multi-task learning. (a) stacked shared-private scheme, (b) parallel shared-private scheme, (c) our proposed attentive sharing scheme.

<https://arxiv.org/pdf/1804.08139.pdf>

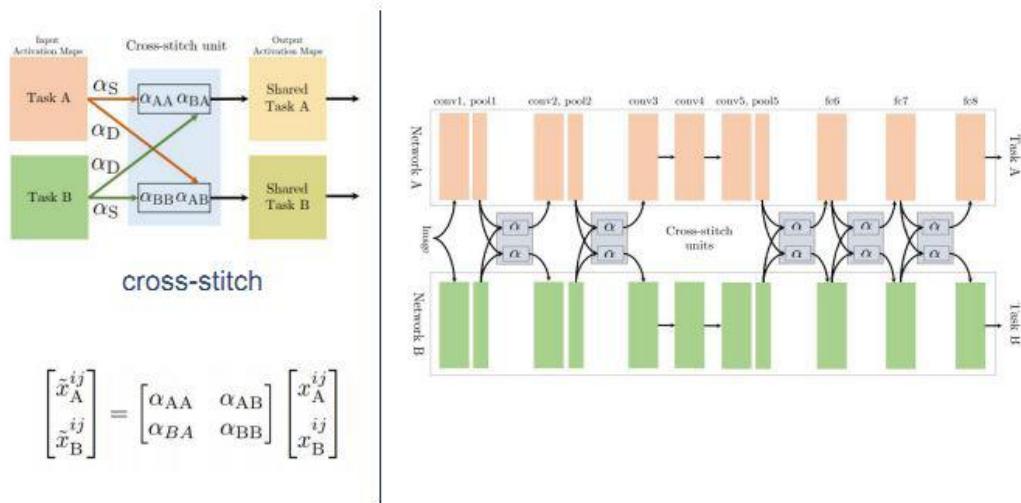
原来的多任务学习如图 a 所示，下面的 s 是共享层，p 是不同任务自己的设计。现在我们将原有的算法转换大图 c 的形式，所有的表示函数共享，在输入到具体任务的时候使用一个和任务相关的查询 Q 去 s 中选择任务相关的信息。虽然表示方式是一样的，但是针对不同的具体任务，会根据每个任务关注点的不同来选择相应的信息。

2、软共享模式

在软共享模式中没有严格规定共享层。经典网络 cross-stitch 结构中，上面是 taskA，下面是 taskB，在中间部分两个任务有交互， α 是权重系数，表示 taskA 中有多少信息从自身流过来，有多少信息从 taskB 中流过来，这样两个任务就由两路，四个系数构成一个矩阵做权重组合，如果用到神经网络就类似于下图中右边的这种形式，这种网络最初应用于机器视觉领域，后来被人们用于 NLP。

CVPR 2016 (Spotlight)
Cross-stitch Networks for Multi-task Learning

Ishan Misra* Abhinav Shrivastava* Abhinav Gupta Martial Hebert
The Robotics Institute, Carnegie Mellon University



<https://arxiv.org/pdf/1604.03539.pdf>

3、共享-私有模式

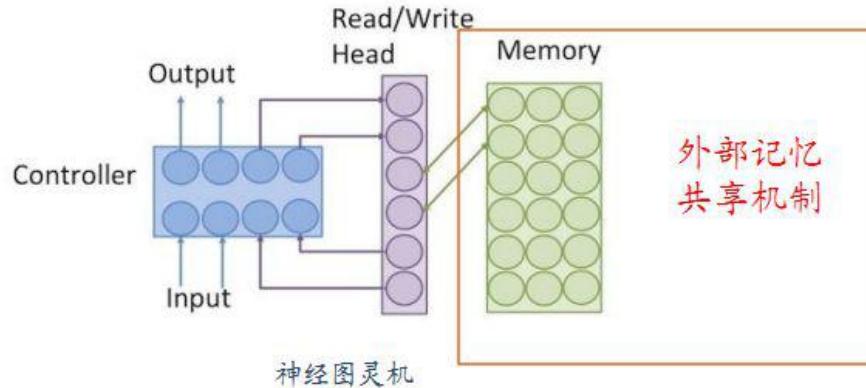
在共享-私有模式中部分网络模块在所有的任务中是共享的，通过设置外部记忆共享机制来实现信息共享，神经图灵机就是在神经网络中引入一个 memory 模块，整个框架就是用神经网络实现的一个控制器，加读写头和外部输入。图灵机全部由神经网络搭建而成。

EMNLP 2016

Deep Multi-Task Learning with Shared Memory

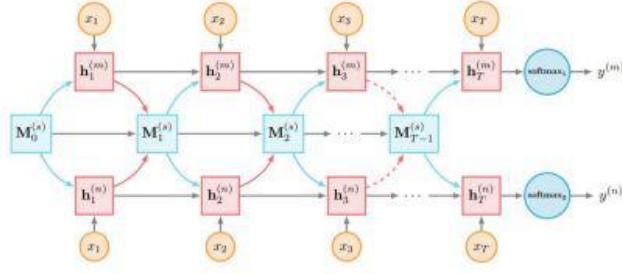
Pengfei Liu Xipeng Qiu* Xuanjing Huang

Shanghai Key Laboratory of Intelligent Information Processing, Fudan University
School of Computer Science, Fudan University

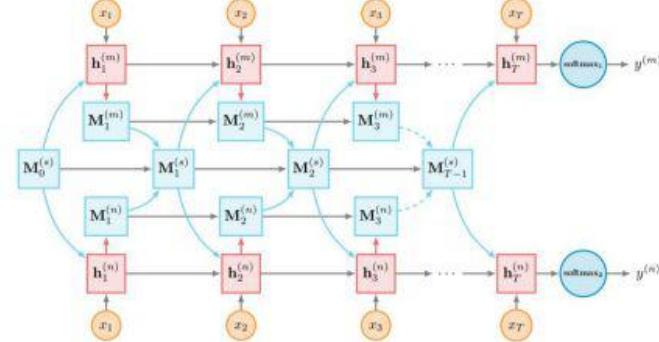


<https://aclweb.org/anthology/D16-1012>

基于神经图灵机的想法我们可以做一个多任务学习 ,每个任务我们都可以看做是一个单独的图灵机 ,外部的 memory 在所有的任务中共享。在下图中 M 是外部记忆 ,外部记忆由两个任务共享 ,每个任务都会把共享信息写到外部记忆中 ,这是一种非常简单的共享方式。



(a) Global Memory Architecture



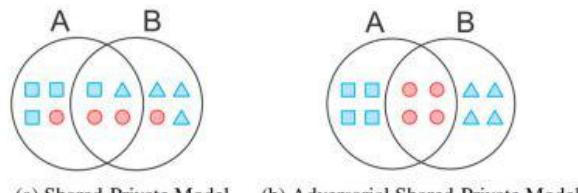
(b) Local-Global Hybrid Memory Architecture

ACL 2017

Adversarial Multi-task Learning for Text Classification

Pengfei Liu Xipeng Qiu Xuanjing Huang
 Shanghai Key Laboratory of Intelligent Information Processing, Fudan University
 School of Computer Science, Fudan University

The **infantile** cart is simple and easy to use.
 This kind of humour is **infantile** and boring.



(a) Shared-Private Model (b) Adversarial Shared-Private Model

Two sharing schemes for task A and task B.
 The overlap between two black circles denotes shared space.
 The blue triangles and boxes represent the task-specific features
 The **red circles** denote the features which can be shared.

<http://aclweb.org/anthology/P/P17/P17-1001.pdf>

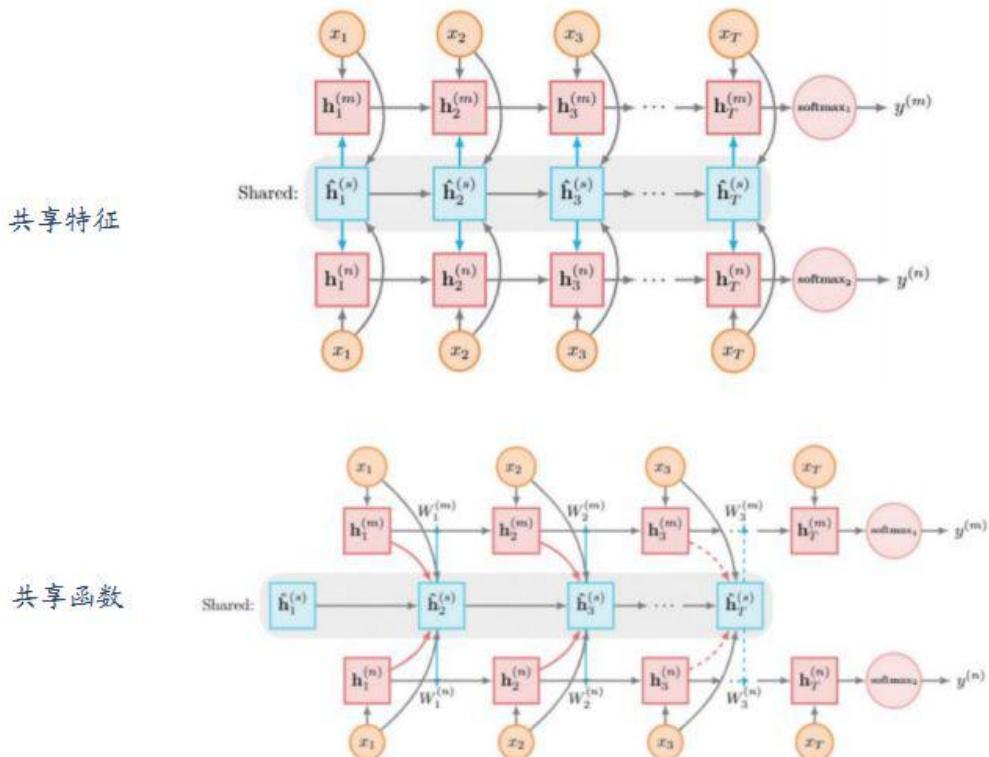
为了避免上图中的负迁移 negative transfer，就需要判断哪些内容是和任务相关的，这就引入了近两年流行的对抗学习，在对抗学习中，中间的 LSTM 共享层有一个判决器来区分共享特征从哪个任务传递过来，在送入 LSTM 之前会包含有特征的来源信息。因此我们希

希望训练一个和判决器对抗的网络，在共享的 LSTM 层中尽可能让判决器不能区分任务来源。这样就去掉了特征的源信息，保证了共享 LSTM 学到的是与源无关的共享价值信息，这些叫做对抗信息。

下面我们将介绍几种未来研究的方向：

1、函数共享模式

之前我们了解的多任务学习都是特征共享，在函数共享中我们学的不再是共享特征而是共享函数，来生成一些参数或模型，这里我们将 feature 级的共享迁移到函数级的共享，下图中第一幅图是特征共享，中间蓝色的是共享层，它将学到的特征送到上下两个任务中，第二幅图是函数共享，函数共享中共享层的输出不是直接送到上下两个分类器中，而是决定了上下两个分类器的参数。通过修改分类器来有效利用这些信息。



2、多级共享模式

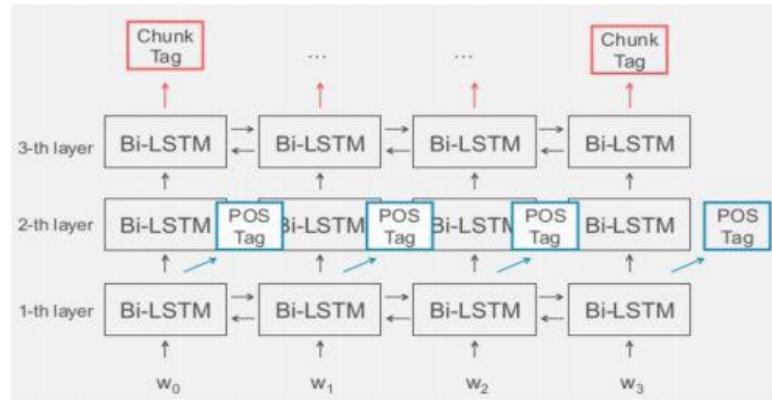
2016 年 Anders Søgaard 等人在论文 Deep Multi-task Learning with Low Level Tasks Supervised at Lower Layers 中提出在低级的网络层次输出低级的任务，在高级的网络层次输出高级的任务。例如在第一层输出词性标签 POS tag，在第三层输出 chunk tag，将 NLP 任务按照不同的级别来设计共享模式。

ACL short 2016

Deep multi-task learning with low level tasks supervised at lower layers

Anders Søgaard
 University of Copenhagen
 soegaard@hum.ku.dk

Yoav Goldberg
 Bar-Ilan University
 yoav.goldberg@gmail.com



	LAYERS		DOMAINS			
	CHUNKS	POS	BROADCAST (6)	BC-NEWS (8)	MAGAZINES (1)	WEBLOGS (6)
BI-LSTM	3	-	88.98	91.84	90.09	90.36
	3	3	88.91	91.84	90.95	90.43
	3	1	89.48	92.03	91.53	90.78

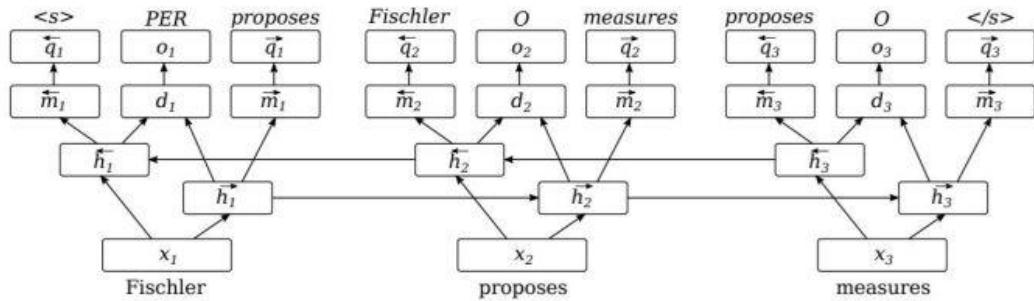
<http://anthology.aclweb.org/P16-2038>

3、主辅任务模式

在做任何一个主要任务的同时都可以引入一个辅助任务。如下图，我们对每个任务引入一个辅助的语言模型，每个任务都使用左右两个语言模型，对所有任务进行这种拓展就形成了主辅任务模式。

Marek Rei

The ALTA Institute
 Computer Laboratory
 University of Cambridge
 United Kingdom



<https://arxiv.org/pdf/1704.07156.pdf>

4、共享模式搜索

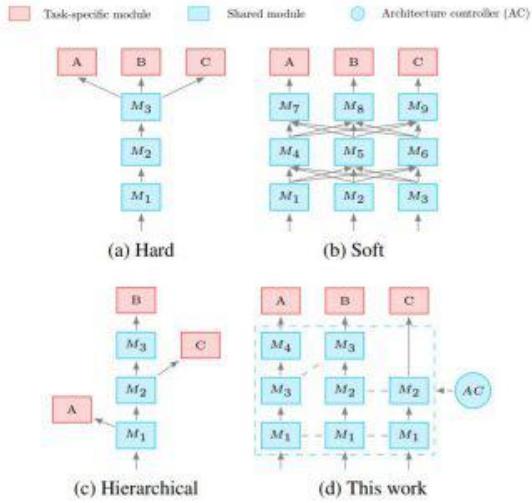
共享模式搜索是让计算机自动搜索这些共享模式，具体做法如图 d 所示，我们希望设计一种灵活的框架，在共享池中放入很多不同的模块，每个任务在完成过程中可以从共享池中挑选一些模块来组装自己的 guideline。示例中任务 A 挑选了 4、3、1，任务 B 挑选了 3、2、1，这就隐含了 A 从 M4 出来，而 B 从 M3 出来，C 从 M2 出来，这样一种层次化的共享模式设计。它本身也可以实现 hard 和 soft 的两种表示方式，因此是一种非常灵活的表示方式。

Exploring Shared Structures and Hierarchies for Multiple NLP Tasks

Junkun Chen*, Kaiyu Chen*, Xinchi Chen, Xipeng Qiu[†], Xuanjing Huang

Shanghai Key Laboratory of Intelligent Information Processing, Fudan University

School of Computer Science, Fudan University

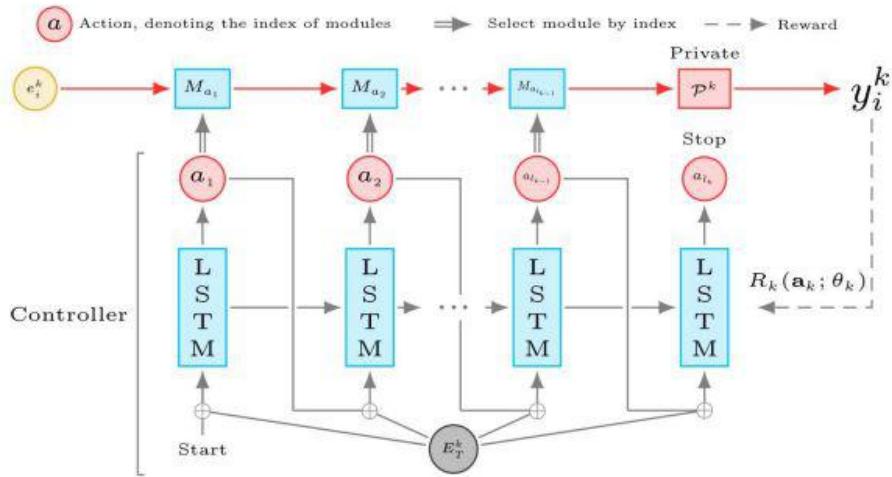


<https://arxiv.org/pdf/1808.07658.pdf>

在面向 NLP 的神经网络架构搜索中，从共享池中挑选 M_{a1}, M_{a2} 等模块来组成不同的模型，将模型带入任务中去训练，得到正确率作为 reward 反馈给分类器从而选择更合适的组合方式来完成任务。

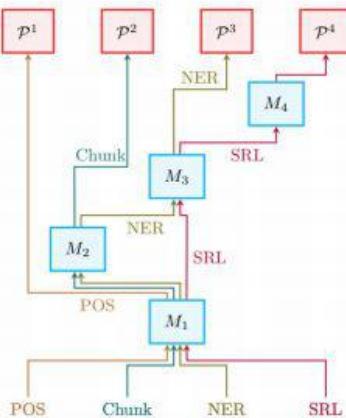


面向NLP的神经网络架构搜索

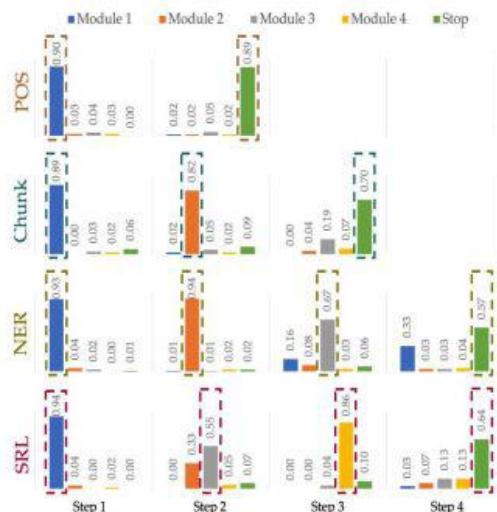


下面给出的例子就是对不同的任务挑选的不同的组合方式，其中有些组合方式非常类似。

面向NLP的神经网络架构搜索



自动选择的共享模式

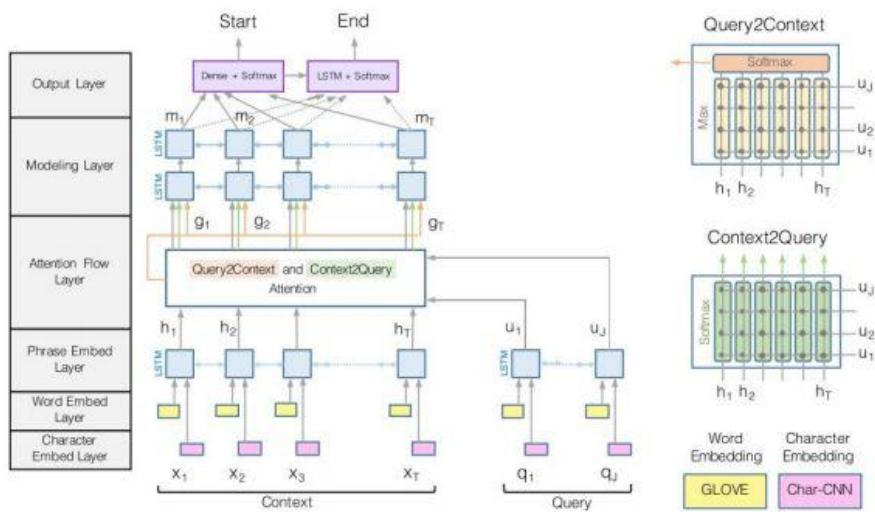


自动选择过程

四、新的多任务基准平台

首先介绍一下机器阅读理解，机器阅读理解是在阅读一篇或多篇文档后，回答一些相关问题。由机器来生成答案，答案可能在原文中出现也可能不在原文中出现，目前机器阅读理解大部分都假设答案在原文中出现，我们用的一个主要框架是 Bidirectional Attention，同时给你 context 和 query，做一个双向的注意力交互，最终确定两个位置，一个是答案开始的位置，一个是答案结束的位置，大部分的问题都可以通过这个框架来解决，这个框架具有通用性。几乎 NLP 所有任务都可以转化成阅读理解任务通过该框架解决和完成。

Bidirectional Attention (Seo et al., 2016)



The Natural Language Decathlon: Multitask Learning as Question Answering

Bryan McCann, Nitish Shirish Keskar, Caiming Xiong, Richard Socher
Salesforce Research

Examples

Question	Context	Answer	Question	Context	Answer
What is a major importance of Southern California in relation to California and the US?	Southern California is a major economic center for the state of California and the US...	major economic center	What has something experienced?	Areas of the Baltic that have experienced eutrophication .	eutrophication
What is the translation from English to German?	Most of the planet is ocean water.	Der Großteil der Erde ist Meerwasser.	Who is the illustrator of Cycle of the Werewolf?	Cycle of the Werewolf is a short novel by Stephen King, featuring illustrations by comic book artist Bernie Wrightson .	Bernie Wrightson
What is the summary?	Harry Potter star Daniel Radcliffe gains access to a reported £320 million fortune...	Entailment.	What is the change in dialogue state?	Are there any Eritrean restaurants in town?	food: Eritrean
Hypothesis: Product and geography are what make cream skimming work. Entailment, neutral, or contradiction?	Premise: Conceptually cream skimming has two basic dimensions – product and geography.		What is the translation from English to SQL?	The table has column names... Tell me what the notes are for South Australia	SELECT notes from table WHERE 'Current Slogan' = 'South Australia'
Is this sentence positive or negative?	A stirring, funny and finally transporting re-imagining of Beauty and the Beast and 1930s horror film.	positive	Who had given help? Susan or Joan?	Joan made sure to thank Susan for all the help she had given.	Susan

今年新发布的一个 NLP 通用的多任务学习系统叫做十项全能，选取了十个典型的 NLP 任务转化成阅读理解的形式，例如左下角的情感分类问题，将这些任务转换到阅读理解问题后采用 Bidirectional Attention 框架去处理。由于这些问题的答案不一定出现在背景文档中，因此需要对 Bidirectional Attention 框架进行改进。

The Natural Language Decathlon: Multitask Learning as Question Answering

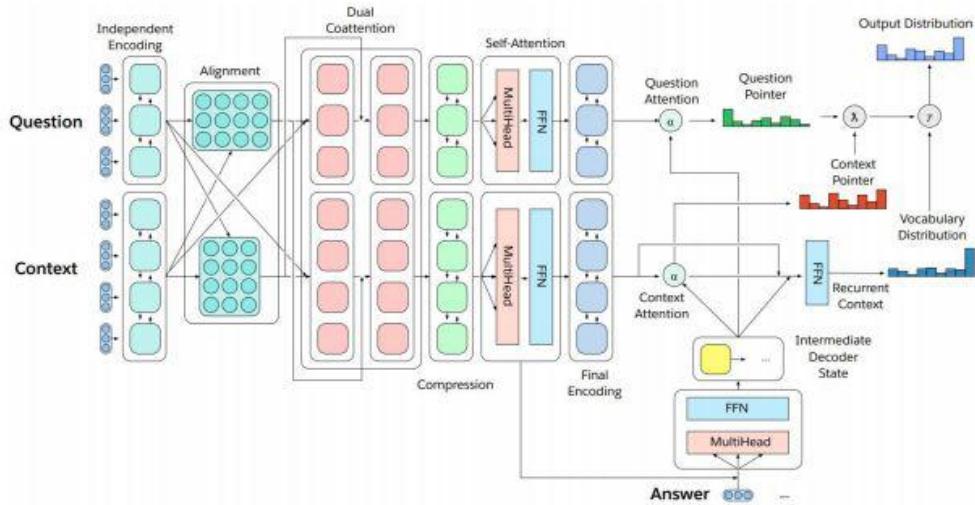
Bryan McCann, Nitish Shirish Keskar, Caiming Xiong, Richard Socher
Salesforce Research

Task	Dataset	# Train	# Dev	# Test	Metric
Question Answering	SQuAD	87599	10570	9616	nF1
Machine Translation	IWSLT	196884	993	1305	BLEU
Summarization	CNN/DM	287227	13368	11490	ROUGE
Natural Language Inference	MNLI	392702	20000	20000	EM
Sentiment Analysis	SST	6920	872	1821	EM
Semantic Role Labeling	QA-SRL	6414	2183	2201	nF1
Zero-Shot Relation Extraction	QA-ZRE	840000	600	12000	cF1
Goal-Oriented Dialogue	WOZ	2536	830	1646	dsEM
Semantic Parsing	WikiSQL	56355	8421	15878	lfEM
Pronoun Resolution	MWSC	80	82	100	EM

<https://arxiv.org/pdf/1806.08730.pdf>

The Natural Language Decathlon: Multitask Learning as Question Answering

Bryan McCann, Nitish Shirish Keskar, Caiming Xiong, Richard Socher
Salesforce Research



<https://arxiv.org/pdf/1806.08730.pdf>

还有一个较大的框架是 GLUE，也是将很多 NLP 任务转化成一个统一的形式。下图中是三个任务：单个句子任务、计算两个句子相似度、表示两个句子之间的蕴含关系。这些任务都可以做成 encoder 和 decoder 模式。

GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding

Alex Wang¹, Amanpreet Singh¹, Julian Michael², Felix Hill³,
Omer Levy², and Samuel R. Bowman¹

¹New York University, New York, NY

²Paul G. Allen School of Computer Science & Engineering, University of Washington, Seattle, WA

³DeepMind, London, UK

Corpus	Train	Dev	Test	Task	Metric	Domain
Single-Sentence Tasks						
CoLA	10k	1k	1.1k	acceptability	Matthews	linguistics literature
SST-2	67k	872	1.8k	sentiment	acc.	movie reviews
Similarity and Paraphrase Tasks						
MRPC	4k	N/A	1.7k	paraphrase	acc./F1	news
STS-B	7k	1.5k	1.4k	sentence similarity	Pearson/Spearman	misc.
QQP	400k	N/A	391k	paraphrase	acc./F1	social QA Questions
Inference Tasks						
MNLI	393k	20k	20k	NLI	acc. (match/mismatch)	misc.
QNLI	108k	11k	11k	QA/NLI	acc.	Wikipedia
RTE	2.7k	N/A	3k	NLI	acc.	misc.
WNLI	706	N/A	146	coreference/NLI	acc.	fiction books

<https://arxiv.org/pdf/1804.07461.pdf>

五、总结

最后，我们对今天介绍的内容做一个总结。今天主要介绍了自然语言处理简介、基于深度学习的自然语言处理、深度学习在自然语言处理中的困境、多任务学习和新的多任务基准平台。总的来说多任务学习的难度会比迁移训练低而效果比预训练要高一些。



总结

- } 自然语言处理简介
- } 基于深度学习的自然语言处理
- } 深度学习在自然语言处理中的困境
 - } 无监督预训练
 - } 多任务学习
- } 自然语言处理中的多任务学习
 - } 硬共享模式
 - } 软共享模式
 - } 共享-私有模式
 - } 函数共享模式
 - } 多级共享模式
 - } 主辅任务模式
- } 新的多任务基准平台

另外，在今年 12 月中旬，我们将发布一个模块化的开源自然语言工具 fastNLP。



fastNLP : 一个模块化的开源自然语言工具

这个工具包括 Spacy 高级接口、AllenNLP 自定义模块、AutoML 自动调参。将训练好的模型开放出来供大家直接调用。



fastNLP=Spacy + AllenNLP + AutoML + ...



为实现模块化，我们将 NLP 分为四个构成组件：

- 1、**编码器**：将输入编码为一些抽象表示，输入的是单词序列，输出是向量序列；
- 2、**交互器**：使表示中的信息相互交互，输入的是向量序列，输出的也是向量序列；
- 3、**聚合器**：聚合信息，输入向量序列，输出一个向量；
- 4、**解码器**：将表示解码为输出，输出一个标签或者输出标签序列。



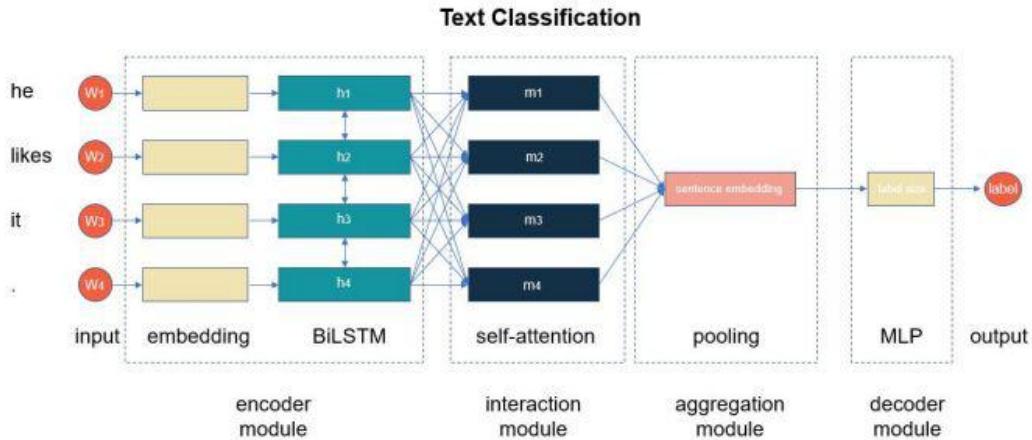
设计理念：模块化

- } 大部分NLP模型可以由4个组件构成：
 - } 编码器 (encoder)：将输入编码为一些抽象表示，输入的是单词序列，输出向量序列。
 - } 交互器 (interactor)：使表示中的信息相互交互，输入的是向量序列，输出的也是向量序列。
 - } 聚合器 (aggregator)：聚合信息，输入向量序列，输出一个向量。
 - } 解码器 (decoder)：将表示解码为输出，输出一个标签（文本分类）或者输出标签序列（序列标注）。

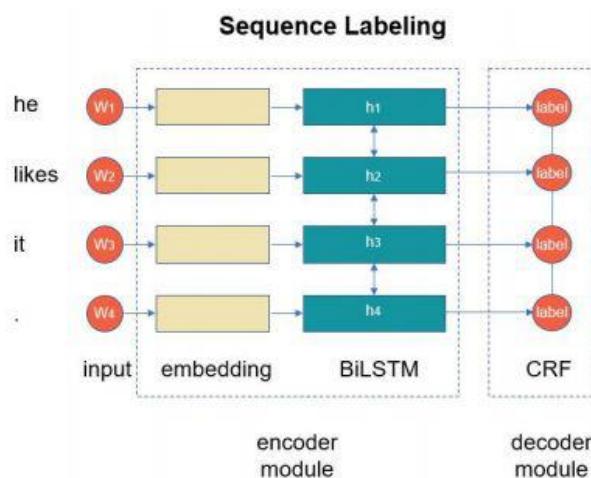
这里我们给出了两个示例，分别是文本分类和序列标注。



示例：文本分类



示例：序列标注



可以应用的场景主要包括：

- 1、直接调用；
- 2、模型开发；
- 3、自动化学习。



小结：三个应用场景

- } 直接调用（通用需求）
 - } 分词、词性标注、实体名识别、句法分析
 - } 提供state-of-the-art模型
- } 模型开发（具有NLP背景的研发人员）
 - } 高复用的模块化编程，快速实现NLP模型
- } 自动化学习（无需NLP背景的产品经理）
 - } 自动模型选择、超参优化
 - } 自动模型训练、评价、校验、发布

作者介绍：

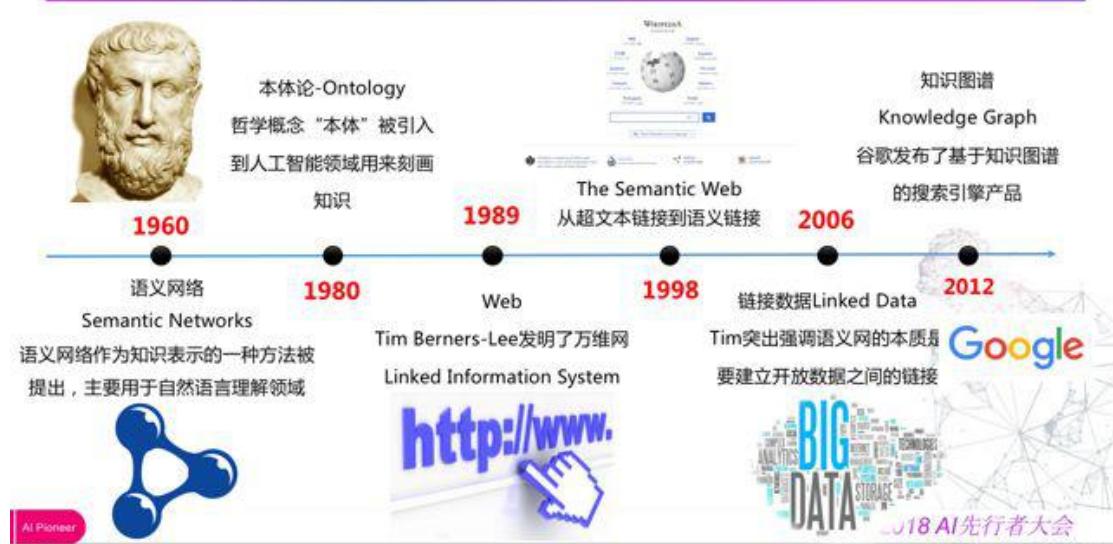
邱锡鹏，复旦大学计算机科学技术学院 副教授，博士生导师，于复旦大学获得理学学士和博士学位。中国中文信息学会青年工作委员会执委、计算语言学专委会委员、中国人工智能学会青年工作委员会常务委员、自然语言理解专委会委员。主要研究领域包括人工智能、机器学习、深度学习、自然语言处理等，并且在上述领域的顶级期刊、会议（ACL/EMNLP/IJCAI/AAAI等）上发表过 50 余篇论文。自然语言处理开源工具 FudanNLP 作者，2015 年入选首届中国科协青年人才托举工程，2017 年 ACL 杰出论文奖。

金融知识图谱的应用与探索

作者：张秋剑 整理：程峰

今天的分享分为以下几个部分：知识图谱的实现基础、理论基础和技术基础，还包括三个案例，跟金融都是相关的。

知识图谱的实现基础



首先讲的是实现基础，这个时间轴贯穿于知识图谱的前世今生，发展到今天大概 50-60 年，其中最早诞生的是语义网络。语义网络可以理解为，现存的词汇都是可以串联起来的，比如说“麻雀是鸟，朱鹮也是一种鸟，朱鹮又是国家一级保护动物，一级国家保护动物包括扬子鳄、大熊猫。大熊猫本身又是哺乳动物。”这样就可以对某一领域的知识甚至是全领域的知识，可以通过网络的方式进行链接，这样就构建了一个语义网络，它是对知识表示的奠基。

到了 80 年代，人工智能领域又把哲学中的本体论引入进来，刚才讲到的“麻雀”、“鸟”和“朱鹮”等等，需要标识哪些是主体，这样就引进了本体论。接下来互联网的诞生，从万维网诞生到超文本的链接，像维基百科，使得互联网把数据链接起来，出现了数据链接这样非常重要的跨越。

这样非常重要的三段历程，语义网络本体论，以及从文本链接到数据链接，成为构成知识网络的基础。Google 在 2012 年推出了全世界第一款知识图谱的产品。

知识图谱的意义 — 洞察语义

- 2012年5月16日，Google为了提升搜索引擎返回的答案质量和用户查询的效率，发布了知识图谱 (Knowledge Graph)
- 有知识图谱作为辅助，搜索引擎能够洞察用户查询背后的语义信息，返回更为精准、结构化的信息，更大可能地满足用户的查询需求
- Google知识图谱的宣传语 “**things not strings**” 给出了知识图谱的精髓，即，不要无意义的字符串，而是获取字符串背后隐含的对象或事物



google 对于知识图谱有一个非常重要的定义，things not strings。过去人们通过搜索引擎获取大量信息，其中相当多是我们不想要的，当然也包括广告，甚至有一些噪音。比如搜索“贵州茅台”，实际上关注的是这只股票，希望在检索的时候更精准的告诉我们想要的，不要有太多臃肿的知识。Google 在自己的知识图谱里就是这样去构建的。ppt 右下角，在检索贵州茅台，会有一个简单的知识库的总结，包括实时股价，归属，总资产规模，包括子公司等等。更加精准定义我们想要的东西，展现字符串背后隐含的对象和事物。我们的目标就是洞察语义。

知识从何而来？



简单回顾了历史，讲了一下知识图谱的实现基础，接下面回顾知识图谱的理论基础。知识图谱中很重要的一点就是知识，知识从何而来。过去知识的获取主要有三种方式。第一种是进

化，更多的是人与自然的互驯，在相互驯化的过程中，适者生存，把最先进的知识传承下去。第二种是经验，经验是日常生活中司空见惯的方式，例如“一朝被蛇咬，十年怕井绳”，这就是一个典型的经验。第三种是文化传承，从古代的图腾到近现代的文字、书籍和影像资料等，更广泛的让我们获取知识和传承知识的方式。

知识从何而来？



到了近现代，除了前三种方式之外，又有了计算机这种新的方式，计算机能帮我们获取知识、存储知识、传播知识、理解知识。理解知识就是广义上讲的机器学习，包括人工智能。

计算机如何发现新知识？



在计算机去发现知识的方向，过去被广泛研究的主要有五种方式。第一种方式是填补现有知识的空白，比如填字游戏，根据字母的排列关系把答案填补上去。第二种方式是模仿大脑，

例如现在比较火的神经网络，用机器去构建神经元。第三种方式是模拟进化，主要用在机器人的领域，让机器人通过自学习自迭代的方式去成长。第四种方式是系统性的减少不确定性，说白了就是统计学，ppt 右上角是典型的贝叶斯定理。第五种方式是注意新旧知识之间的相似性，类似 svm，精准的去找到一个二分类的方法。

机器学习的五大学派

知识发现	学派	学科依据	方法论	主要算法	应用场景
系统性的减少不确定性	贝叶斯学派	统计学	概率推理	朴素贝叶斯、分类器	风控
注意新旧知识之间的相似性	类推学派	心理学	行为类推主义	支持向量机、内核机	推荐
模拟进化	进化学派	进化生物学	遗传算法	遗传编码	机器人
模仿大脑	联结学派	神经科学	希望从大脑运行方式得到启发	反向传播	深度学习
填补现有知识空白	符号学派	逻辑学、哲学	相信填补现有知识的空白的	逆向演绎	专家系统

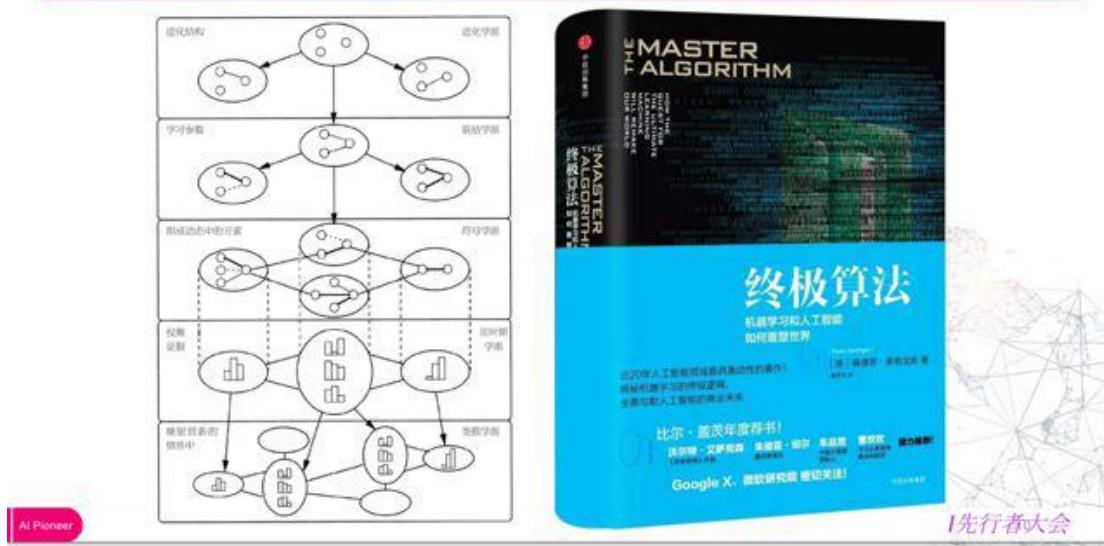
AI Pioneer

2018 AI 先行者大会

做一下总结，发现知识有五种方式，相对应的是五大学派。统计学就是贝叶斯学派，svm 就是类推学派，模拟进化就是进化生物学学派，模仿大脑的就是联结学派，神经网络分支。填补现有知识空白是符号学派。其中的理论依据，还是根据基础学科去做借鉴，比如说统计学、心理学、生物学、脑科学和哲学，同样用到了很多算法，比如贝叶斯分类器，内推学派主要是 svm 内核机，进化学派主要是遗传编码，神经学派主要是反向传播，符号学派是逆向演绎。

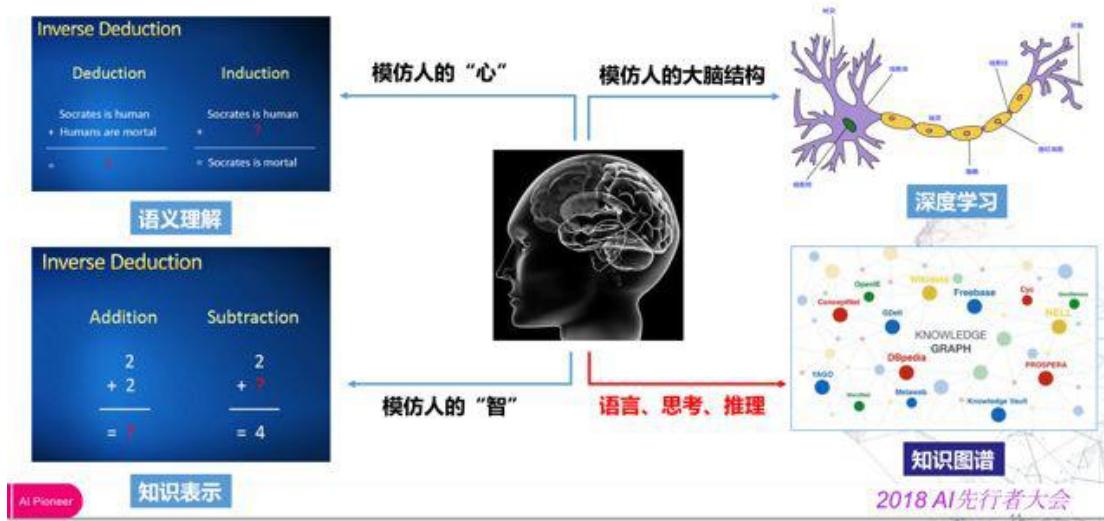
对应的应用场景，统计学用在风险控制的场景，支持向量机用在推荐类的场景，生物学主要是机器人场景，神经网络用在深度学习，符号学派有很多专家系统的应用。

机器学习的五大学派



关于五大学派，有一本书叫终极算法，不同的学派在某个领域去解决不同的问题，有没有一种终极算法把这五个合并在一起？可以参考下这本书。

构建“知识图谱”的理论基础



五个学派，模仿人的大脑，模仿人的心，模仿人的智。其实知识图谱关注的是人类的语言思考以及推理，如何通过机器的方式来实现，构成了知识图谱的理论基础。

知识图谱的三大基础



刚讲了理论基础，这里讲讲技术，这里用了 nlp 的图片，知识图谱主要还是在自然语言处理的领域。

知识图谱的“技术栈”



这是我总结的知识图谱全栈，从底层到上层有四层。

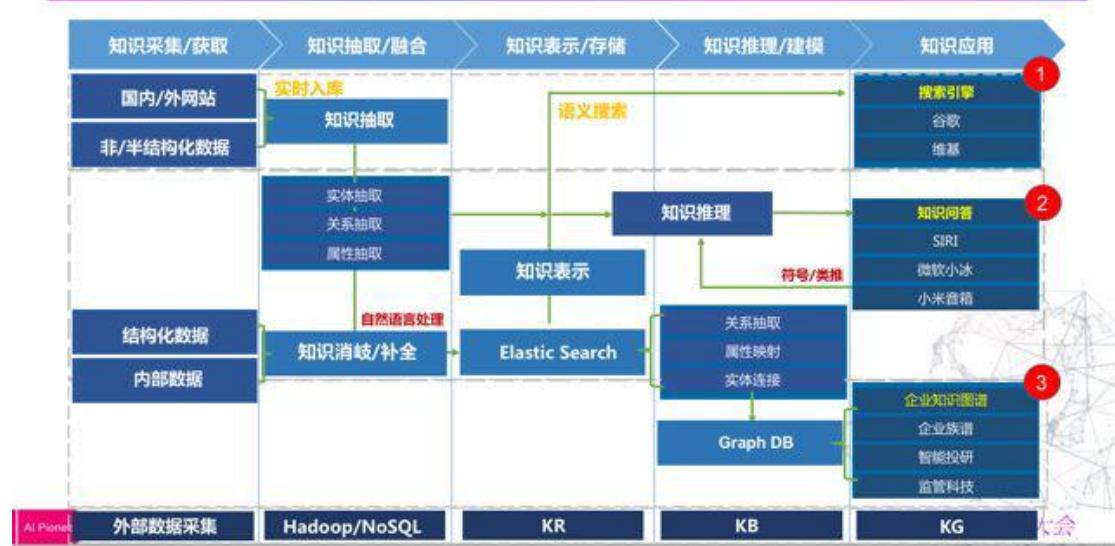
最底层是理论层，理论层就是刚才提到的几个比较关键的点，比如说本体论、语义网络、语义超链接、数据链接以及知识库。

在数据层，举例的都是一些通用的数据源，比如说 freebase，这个是 google 收购了，knowledge vault，这个是 google 开放的知识图谱库，包括维基数据，imagenet 等，这些都是公开的，大家从网上能够查到。

实现层，大概分成六个步骤，分别是知识获取、知识抽取、知识融合、知识存储、知识推理、知识建模和知识发现，知识获取是获取外部数据的方式，包括爬虫和实时入库的技术方法；知识抽取就是，对三元组进行知识的抽取，包括实体抽取、关系抽取和属性的抽取；知识融合就是，抽取出来之后，存在很多的数据冗余和噪声，要做实体的消歧，数据的整合；知识存储，刚才讲了，实际是要构建一个三元组 RDF 的数据结构，如果把所有的顶点和边构造出来之后，要对他进行图数据库的存储；知识推理，刚才也讲到了，如果要做一些深层次的知识问答，就要做很多的训练，无论有监督的还是半监督的；知识建模更多的是去理解语义，涉及到属性的映射，实体的连接；知识发现，两大主要的应用是知识的检索和知识的问答。这些构建了知识图谱的实现层。

再往上就是应用领域，大概分成两个方向，一个是通用领域，比如搜索引擎、机器人和物联网等等。在专业领域基本都是行业，例如交通、能源、金融，包括医疗健康。

知识图谱解决方案



刚才讲的是技术的全栈，这里是解决方案构建的实现路径。首先就是知识的采集和获取。现在的数据无非两块，内部数据和外部数据。对于外部数据，入库后要做知识的抽取，主要是对三元组的抽取，实体关系和属性的抽取。对抽取的知识可以去构建一个简单的搜索引擎应用。把自然语言处理结合进来之后，就要对知识进行消歧和补全，如果有一些行业属性数据，要从这里去做补全。融合之后的数据，首先放在类似 ES 的存储里边，通过知识表示，一方面去构建搜索引擎，再一个就是结合知识推理，对知识问答类的应用去产品化，例如 siri、微软小冰和小米音箱。在知识推理这块，更多的用到了符号学和类推学的算法去实现。对知

识表示化后的数据进行深加工，去做关系抽取、属性映射、实体连接，可以把顶点和边全部结构化，存储在图数据库里。构建了自己的图数据库，可以为行业做一些专用的知识图谱，比如企业族谱、证券的智能投研和监管科技。

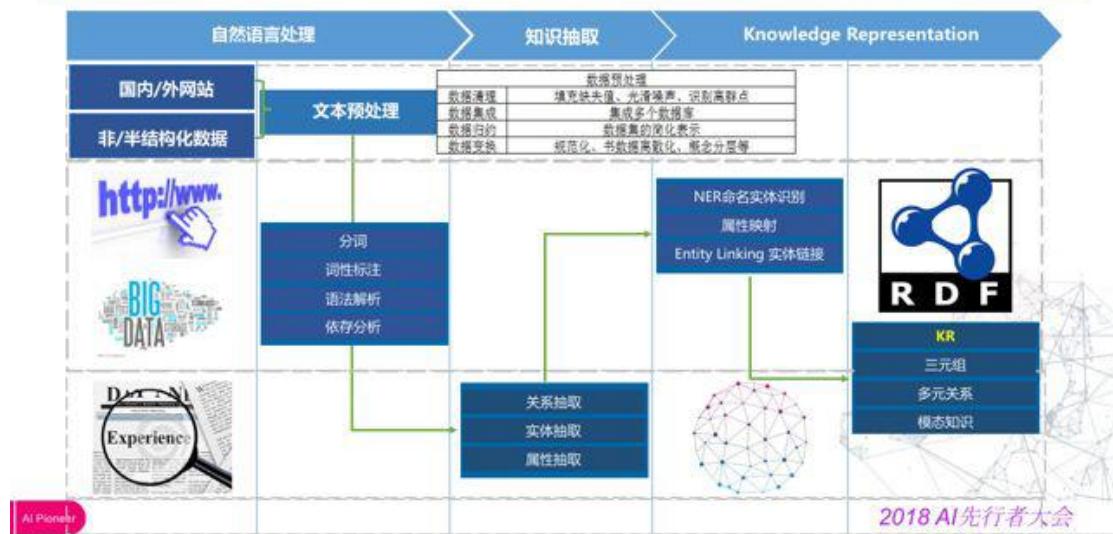
知识图谱的数据源

知识图谱库名称	机构	特点, 构建手段	应用产品
FreeBase	MetacWeb(2010年被谷歌收购)	<ul style="list-style-type: none"> 实体、语义类、属性、关系； 自动+人工：部分数据从维基百科等数据源抽取 而得到；另一部分数据来自人工协调编辑 https://developers.google.com/firebase/ 	Google Search Engine, Google Now
Knowledge Vault (谷歌知识图谱)	Google	<ul style="list-style-type: none"> 实体、语义类、属性、关系； 超大规模的数据；源自维基百科、Freebase、《世界名胜记实手册》 https://research.google.com/pubs/pub45634 	Google Search Engine, Google Now
DBpedia	莱比锡大学、柏林自由大学、OpenLink Software	<ul style="list-style-type: none"> 实体、语义类、属性、关系； 从维基百科抽取 * 	DBpedia
维基数据(Wikidata)	维基媒体基金会(Wikipedia Foundation)	<ul style="list-style-type: none"> 实体、语义类、属性、关系，与维基百科紧密结合； 人工（协同编辑） 	Wikipedia
Wolfram Alpha	沃尔夫实验室(Wolfram Research)	<ul style="list-style-type: none"> 实体、语义类、属性、关系，知识计算 部分知识来自于Mathematica，其它知识来自于各个科普网站 	Apple Siri
Bing Satori	Microsoft	<ul style="list-style-type: none"> 实体、语义类、属性、关系，知识计算 自动+人工 	Bing Search Engine, Microsoft Cortana
YAGO	马克斯·普朗克研究所	自动：从维基百科、WordNet和GeoNames提取信息	YAGO
Facebook Social Graph	Facebook	Facebook 社交网络数据	Social Graph Search
百度知识图谱	百度	搜索结构化数据	百度搜索
搜狗知立方	搜狗	搜索结构化数据	搜狗搜索
ImageNet	斯坦福大学	<ul style="list-style-type: none"> 搜索引擎 亚马逊 AMT 	计算机视觉相关应用

2018 AI先行者大会

刚才提到了知识图谱的一些数据源，这里也摘录了一些信息，比如 freebase，Google 的 knowledge vault 等。

知识抽取 – NLP+KR



刚才讲了知识图谱的构建步骤，对几个比较抽象的步骤做展开。第一个就是知识抽取，知识抽取就是自然语言理解和知识表示的结合。刚才提到了自然语言处理两个非常重要的步骤，第一个就是文本的预处理，涉及到数据的清理、降噪、数据的集成、数据的离散化；第二个

步骤，就是做分词、做标注，更深入一点的是做语法的解析和依存度的解析，这个层面实现后做三元组的抽取。把关系、实体和属性抽取出来。再后边就是知识表示，实现关系、实体和属性之间的关联，构建三元组。

知识表示 - 数据结构 - RDF

- 实体：指的是具有可区别性且独立存在的某种事物。如某一个人、某一个城市、某一种植物等、某一种商品等等。世界万物由具体事物组成，此指实体。如图1的“中国”、“美国”、“日本”等。实体是知识图谱中的最基本元素，不同的实体间存在不同的关系。
- 语义类（概念）：具有同种特性的实体构成的集合，如国家、民族、书籍、电能等。概念主要指集合、类别、对象类型、事物的种类，例如人物、地理等。
- 内容：通常作为实体和语义类的名字、描述、解释等，可以由文本、图像、音视频等来表达。
- 属性：从一个实体指向它的属性值。不同的属性类型对应于不同类型属性的边。属性值主要指对象指定属性的值。如图1所示的“面积”、“人口”、“首都”是几种不同的属性。
- 属性值：主要指对象指定属性的值，例如960万平方公里等。
- 关系：形式化为一个函数，它把 k 个点映射到一个布尔值。在知识图谱上，关系则是一个把 k 个圆节点(实体、语义类、属性值)映射到布尔值的函数。



AI Pioneer

2018 AI 先行者大会

刚才多次提到三元组，到底什么是三元组，一部分是一些概念、实体、语义、内容、属性。我们对于语言的理解，主要还是通过主谓宾的方式去构建，主谓宾就是典型的一个三元组，把它应用到知识图谱就是 RDF。RDF 有非常多的构建方式，下面举了两种例子。一种是实体、关系、实体的方式，一种是实体、属性、属性值的方式。举个例子，某某法人京东，构建了一个非常简单的三元组，可以理解某某是京东的法人。

RDF举例 — 某证券 “公司与现任高管三元组”

数据建模

- 从任职关系表中抽出(group by)高管信息作为图数据库中高管的节点
- 将上市公司信息表所有信息作为图数据库中公司的节点
- 将任职关系表中的高管个人信息与高管节点进行关联，将任职关系表中的公司代码与公司节点进行关联，构造图数据库中的任职关系

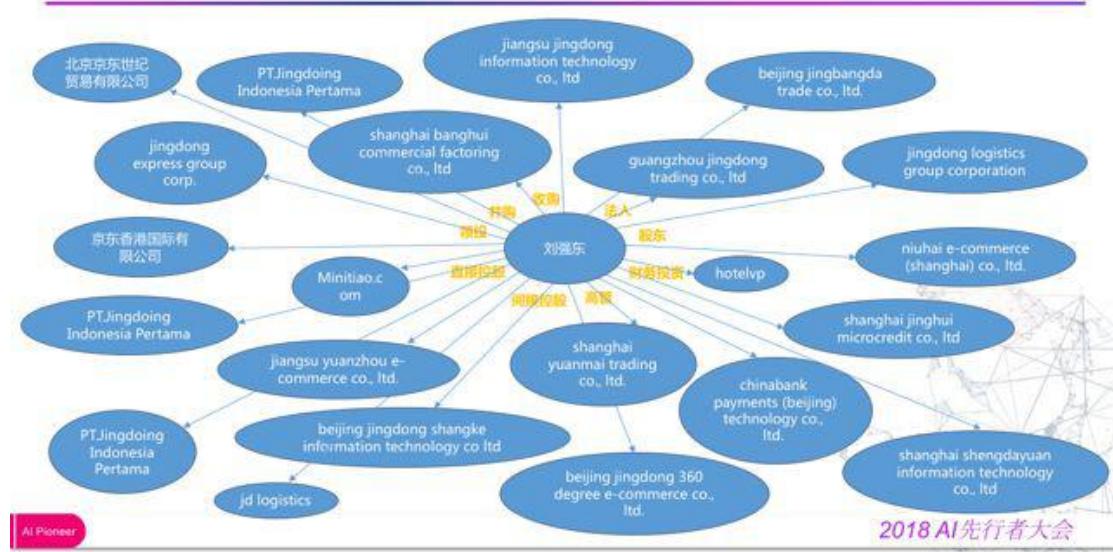


AI Pioneer

2018 AI 先行者大会

举一个小小的案例，这是我们帮一个券商做的企业族谱，如何把上市公司任职的高管关系全部遍历出来，从源数据库抽取三张表，分别是高管信息表、任职表、公司信息表。这样就可以把对应的高管字段、高管任职的字段以及所在的公司属性字段抽取出来，构建成高管任职关系的三元组。

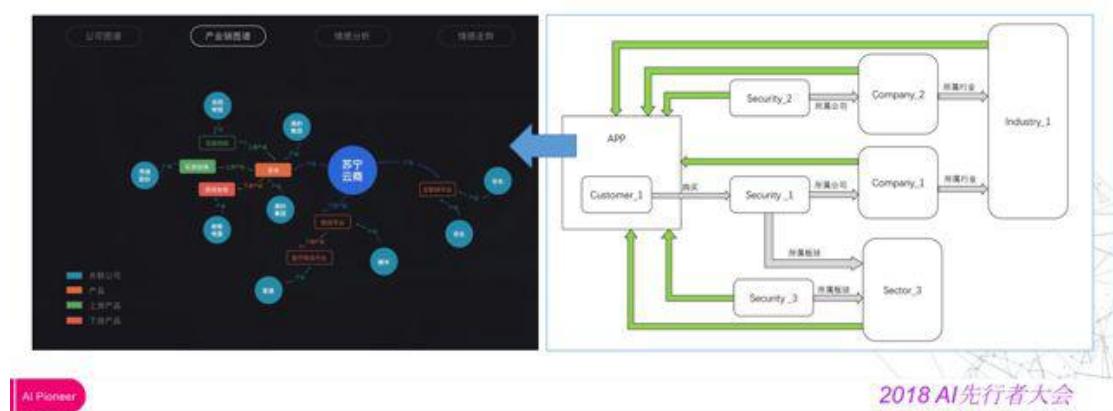
知识表示 – 数据结构 – RDF



某某和所有有资金关联的公司，全部都在上面，有些是法人关系，有些是财务投资，有些是并购，有些是控股。

知识表示举例 – 某证券 “消息精准推送”

以某客户购买某只证券为例，通过构建的三元组关系可以查询到与该证券关系深度为n的所有节点及其关系，并选择性向客户推送查询到的节点的更新消息



这是我们帮券商做的应用，每个人都有自己的股票 app，当我们去购买某一个个股的时候，个股对应的公司所在的行业，对应的关联公司，以及个股所在的板块，板块对应的上市公司

个股，都可以构建为知识图谱通过 app 推送给客户。假设用户购买的是“苏宁云商”个股，可以看到行业属性里，会把“京东”个股关联进来，它们都属于互联网电商行业。“苏宁”物流这块就会跟顺丰关联起来。“苏宁”本身所处的板块有家电背景，会跟美的、格力关联，蓝色的是关联公司，橙色的是产品，绿色的是上游，棕色的是下游。

知识存储- 图数据库

图计算/图检索

- 兼容SQL，采用SQL语法的简单扩展实现图计算
- 大规模拓扑图分析(100亿+点，10000亿+边)，线性扩展
- 高度容错的计算引擎
- 支持点、边CRUD
- 秒级图检索查询

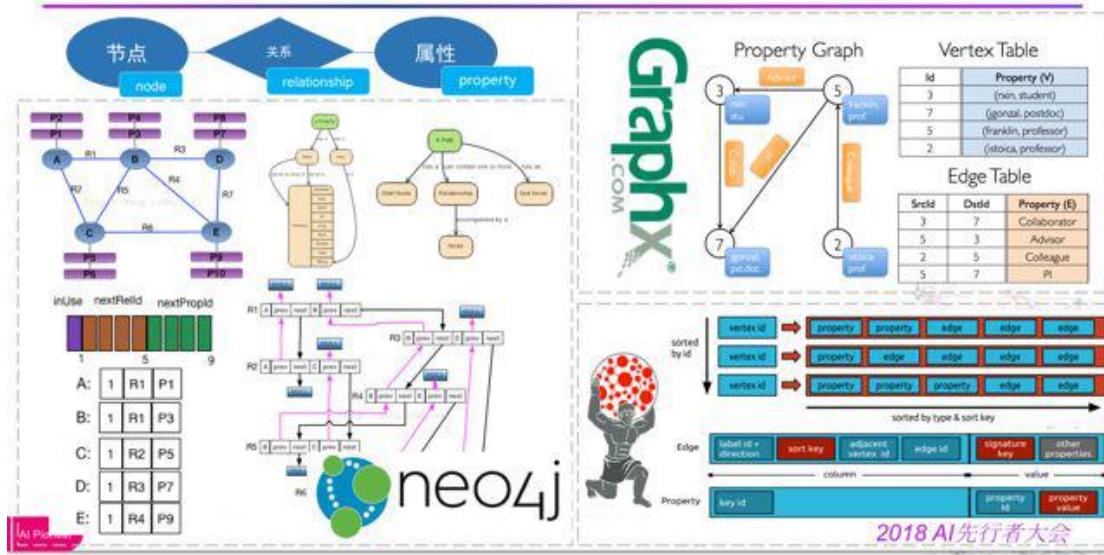
Rank	DB	Model	Database Model	Score
1.	1.	Neo4j	graph database	36.26 ±0.00 ±0.00
2.	2.	OrientDB	graph model	5.85 ±0.00 ±0.00
3.	3.	MySQL	graph model	5.76 ±0.00 ±0.00
4.	4.	ArangoDB	graph model	2.43 ±0.00 ±0.00
5.	5.	Virtuoso	graph model	2.32 ±0.00 ±0.00
6.	6.	Graph	graph model	1.08 ±0.00 ±0.00
7.	7.	MarkLogic	graph model	0.84 ±0.00 ±0.00
8.	8.	GigaGraph	graph model	0.75 ±0.00 ±0.00
9.	9.	Redis	graph model	0.75 ±0.00 ±0.00
10.	10.	InfogridGraph	graph model	0.75 ±0.00 ±0.00
11.	11.	GridGain	graph model	0.75 ±0.00 ±0.00
12.	12.	HyperGraph	graph model	0.75 ±0.00 ±0.00
13.	13.	Graph	graph model	0.75 ±0.00 ±0.00
14.	14.	PinotDB	graph model	0.70 ±0.00 ±0.00
15.	15.	HyperGraph	graph model	0.68 ±0.00 ±0.00
16.	16.	HyperGraphDB	graph model	0.68 ±0.00 ±0.00
17.	17.	HyperGraphDB	graph model	0.68 ±0.00 ±0.00
18.	18.	GraphDB	graph model	0.68 ±0.00 ±0.00
19.	19.	Spanner	graph model	0.68 ±0.00 ±0.00
20.	20.	HyperGraph	graph model	0.68 ±0.00 ±0.00
21.	21.	AgostiniGraph	graph model	0.68 ±0.00 ±0.00
22.	22.	Amara Server	graph model	0.68 ±0.00 ±0.00

图2 2016年的图数据库的综合排名 (截止到2017年1月)



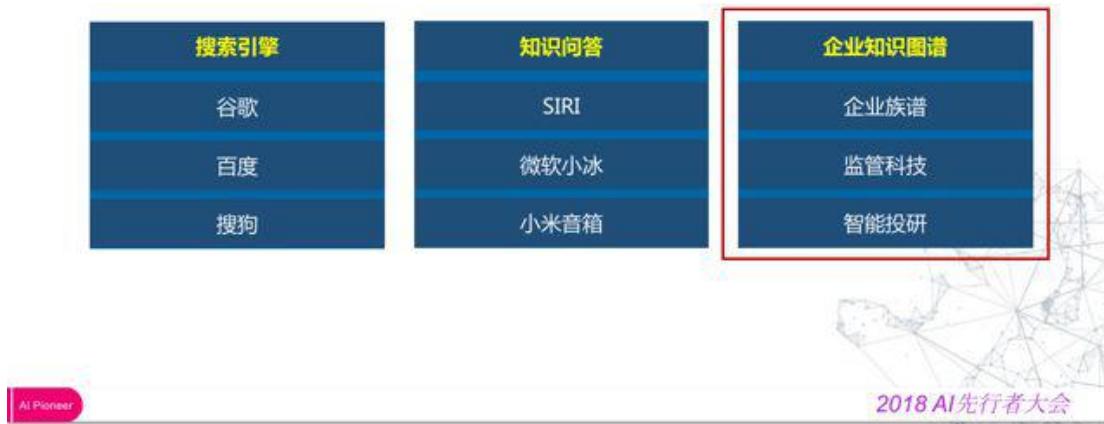
前面讲了知识抽取和知识表示，这里简单讲一下图数据库。最近两年图数据库非常火，像neo4j、StellarDb、GraphX、TITAN、OrientDB。图数据库是知识图谱非常重要的技术架构，对图数据库的存储、检索要求越来越高，希望更多去兼容sql语法。现在很难去构建多层属性，更多的去平铺，平铺开之后有个问题，点和边会非常的多，现在遇到一些案例，上百亿个点，上千亿条边。海量的数据，计算引擎怎么设计，比方说现在比较好的分布式计算架构是不是能更好的去优化，包括点边是否能实现增删改查，对图的遍历能否做到秒级返回，这是我们关注的一些点。

知识存储- 图数据库



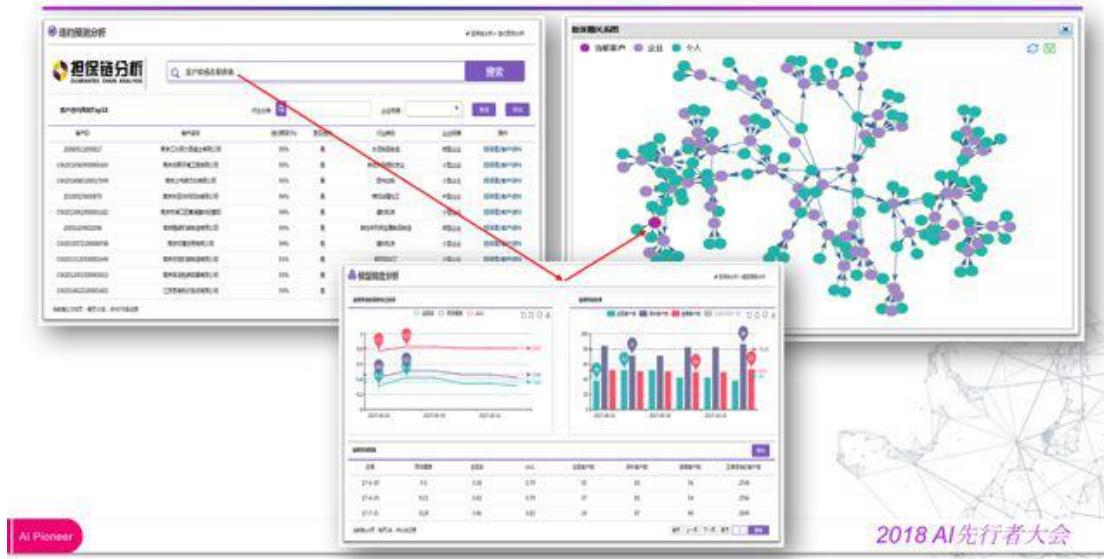
这里对图数据库的存储结构做简单的展示，目前主流的图数据库是通过 RDF 的方式去构建，GraphX 把顶点和边分别存储，属性和属性值和点边产生依赖。TITAN 把顶点、边和属性全部拆分开去构建。Neo4j 是 RDF 去构建，通过指针跳转的方式去连接，各有千秋。

案例场景



最后一部分讲下案例，今天来的很多讲师来自一线互联网公司，更多关注的是搜索引擎和知识问答这些方面，星环是专门做企业级市场的公司，更多关注的是企业级知识图谱。

案例：银行对公信贷分析



第一个例子是银行的案例，银行信贷里担保链的分析。用户检索出目前某一个授信企业客户目前的信贷状况，可以看到信贷的不良率、是否有违约以及逾期的状况，这是一个非常基础的信息报表的展示。当我们发现该企业有疑点的时候，就可以去检查它某一次的授信里面的关联关系。比如该次授信，企业的担保人是不是存在一些问题。右边的知识图谱中，红色是我们查阅的实体，实体与它产生的担保关系就可以全部遍历出来，和给该企业做担保的上级是否存在担保关系。这样全部都能遍历出来，在遍历出来的图谱可以看到企业是不是存在闭环，或者出现双向或者交叉的图形。在过去，人工的方式很难做到，尤其一些体量比较大的银行，企业的经营范围面向全国的时候。目前构建企业担保类的知识图谱非常快，遍历一个大企业能做到小时级。

案例：银监会 — 企业监管图谱



这是一个监管科技的应用，这是我们帮银监会去做的案例，在福建省银监做的银监眼的案例。福建省有七个地市，抓一些关键性的监控指标，比如存款指标、贷款指标以及不良率的指标、流动性指标。这是一个应急看板，可以看到有一些关键性的数据，比方说不良率、地区的存款分布、房产贷款，横坐标是地市。

案例：银监会 — 企业监管图谱

检索是相对比较复杂的，有疑点提示，指标概览，当我们去关注某一个疑点的时候，可以做一些筛选，像资金流向、资金空转、失信被执行等监管科技比较关心的指标，当我们去筛选的时候，把有疑点的一些企业和客户抓取出来。

案例：银监会 — 企业监管图谱

深入点击进去之后，就可以对该企业形成关系图谱，或者叫对公客户的客户画像。比方跟该企业相关联的交易关系，可以通过知识图谱展示出来。空心就是实体，绿色就是跟企业产生交易的，全部都是有向图，箭头指向就是交易的流向。

智能投研 — 企业图谱



第三个案例是证券，帮券商去构建的投研平台，当去搜索个股的时候，除了个股 F10 的信息之外，还会有研报信息和新闻热点信息都可以在看板展示。在左下方，帮助个股构建了四类图谱，第一个是公司图谱，主要对企业内部，跟企业相关的高管、法人以及股东关系。

智能投研 — 行业图谱



产业链图谱，包括物流、家电、电商等。还有所处行业都会做展示。

智能投研 — 舆情分析



跟投资相关会比较关注热度，第一个就是情感分析，比如雪球指数、新浪、股吧。红色表示反向，蓝色表示中性，绿色表示正向。

右边是情感走势，可以看到个股在每一个互联网平台热度的变化。这样就是智能投研的知识图谱。

作者介绍：

张秋剑，星环科技金融事业部总监。上海师范大学计算机科学技术硕士，资深大数据专家和金融行业技术专家。 现任星环科技金融事业部总监，大数据技术架构行业顾问专家，云析学院发起人， AICUG 社区联合发起人，曾在 IEEE 等期刊发表多篇论文。目前主要为银行、证券和保险等行业客户提供大数据平台及人工智能平台的整体规划和项目建设等工作。

猎户星空 NLP 技术进展及产品应用

作者：韩伟 整理：靳贛贛

本次分享的主要内容包括以下四个方面：首先介绍了人机交互的相关背景；然后是猎户在 NLP 提供的能力和具体技术实现环节；最后介绍了猎户 NLP 的产品落地情况。

一、人机交互

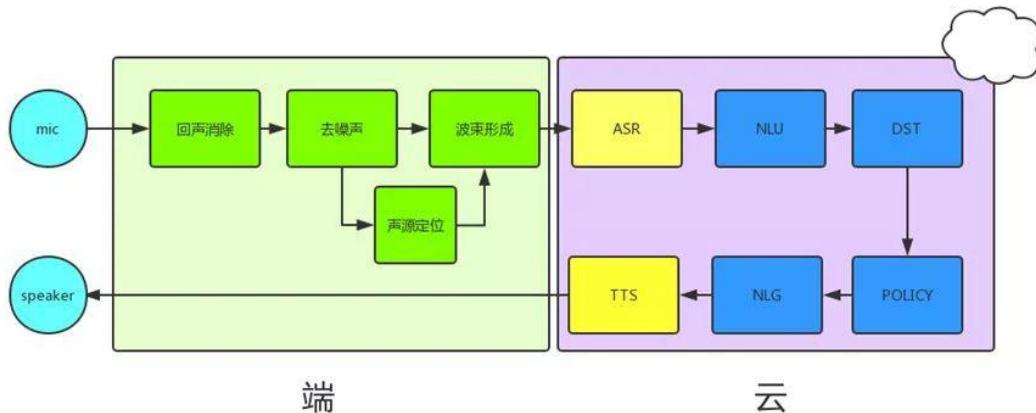
首先来介绍一下人机交互的发展史，最初人们通过键盘输入命令行进行交互，随着苹果和 windows 推出图像界面，我们可以通过鼠标点击来实现人机交互，在 ipad 出现之后，大家可以通过触摸的方式来与机器进行交互，随后的 iphone4s 上出现了跨时代的产品 siri，人们可以直接通过语音来实现人机交互，直接引爆了语音行业的革命。



人机交互中语音交互的整个链路如下，主要可以分为两个部分：端和云，端上的主要工作包括一些信号层面的算法处理，云上主要进行与语音识别相关的一些自然语言处理，对话管理以及和资源相关的工作。首先我们通过麦克风来进行收音，然后通过回声消除和去噪过程将背景噪声以及机器自己产生的声音消除以取得一个干净的信号，由于现在多采用麦克风阵列，所以可以进行波束形成和声源定位。将端上获取的音频信号通过互联网传到云端之后，首先进行的第一步是 ASR，即将音频信号转换成文本，以便于在后端的大脑层来进行决策。之后通过 NLU 理解说话人的意图以及用户说话中的关键信息。DST 对话状态跟踪模块属于

上下文功能，通过 Policy Ranking 决策出用户的意图以及相应的 action，最后由 NLG 模块生成机器要表达的想法并通过 TTS 把文本转化为音频并利用 speaker 播放出来。

语音交互链路



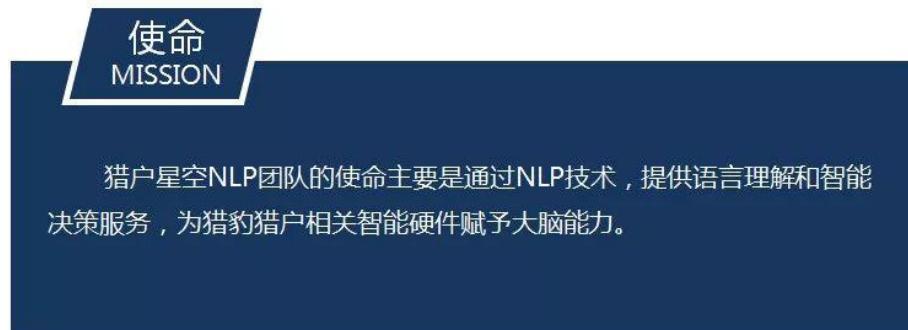
但是在实际的语音交互中存在很多难点，主要可以分为五个方面：

- 1、**用户使用的环境和口音**（用户在使用过程中，环境噪声和口音给语音识别和前端信号处理带来了非常大的困难）；
- 2、**不同的垂直行业，对理解的差异化非常大**（在理解和交互层面，不同的垂直行业知识差别非常大，对于 ASR 和 NLP 如果没有相应的垂直行业知识预练，效果往往会比较差）；
- 3、**不同的硬件设备，对理解有不同的需求**；
- 4、**语义理解和知识图谱等认知技术没有通用的框架**；
- 5、**用户语言表达的多样性，随意性，隐喻，上下文等等**。

二、猎户 NLP 能力

猎户星空 NLP 团队的使命主要是通过 NLP 技术，提供语言理解和智能决策服务，为猎豹猎户相关智能硬件赋予大脑能力。

猎户NLP团队的使命



目前，猎户星空 NLP 团队提供了这样一些能力矩阵：首先是理解，在得到用户的 query 后，对用户的意图做一个深层次的理解，挖掘出用户的意图，即用户想获取的信息，然后结合世界知识，垂直行业知识，企业知识生成一个综合的决策。此外，我们提供用户画像这样一个基础性的服务，不同的用户在交互中有不同的诉求，我们可以结合用户的信息来实现最终的交流目的。

猎户NLP能力矩阵



猎户 NLP 提供的云服务包括：[智能交互服务](#)、[自然语言理解](#)、[智能对话](#)、[问答聊天系统](#)、[用户画像服务](#)和[机器翻译](#)等。

猎户NLP提供的云服务



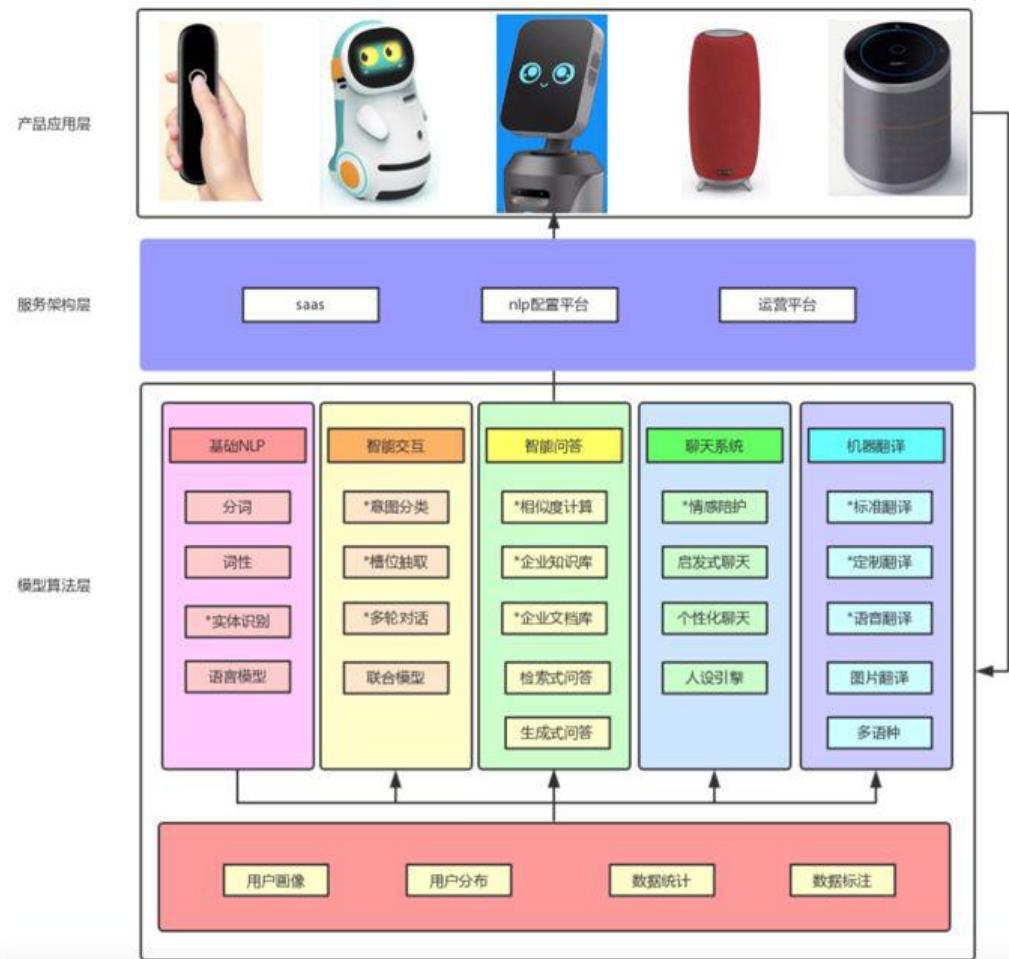
猎户NLP团队提供的云服务

- 1、智能交互服务** : NLP 服务对外的平台架构服务 , 调用 NLU , chat , QA , 开放平台 , 第三方技能服务 , 做综合决策 , 并支持业务线配置 , 用户自定义技能的开放平台 。
- 2、自然语言理解** : 理解用户说的每一句话 , 指导硬件做出正确的动作 , 比如 : “我想听一首欢快的歌叫小苹果” , “北京明天可以洗车吗” 。
- 3、智能对话** : 针对用户对机器的多轮交互 , 维护上下文状态 , 理解用户当前 query 在上文条件下的准确意图 , 执行正确的动作。
- 4、问答聊天系统** : 给出正确答案 , 比如豹小秘中 : “猎豹移动的老板是谁 ? ” 豹小秘回答 : “猎豹移动的董事长兼 CEO 是傅盛。” 对用户的闲聊问句 , 给出具备情感和亲和力的答案 , 增强用户粘性。
- 5、用户画像服务** : 通过用户的交互行为 , 建模用户的特征 , 比如 : 有的用户喜欢听粤语歌 , 有的用户的活动时间在早上 8 点和晚上 10 点等等 , 用于个性化聊天和其他用户相关应用 。
- 6、机器翻译** : 支持中英日韩的相互翻译 ; 基于深度学习技术构建翻译模型进行准确的翻译。

三、猎户 NLP 技术

目前猎户 NLP 整体技术架构如下 , 总共分为三层 , 底层是模型算法层 , 提供用户画像、用户分布、数据统计、数据标注等任务 , 这些数据从产品流过来。在基础算法层之上我们搭建了服务架构层 , 通过服务架构层把所有底层算法能力以平台方式输出 , 业务方或需求方可以很好的通过平台去使用服务。最上层是产品应用层。

猎户NLP技术架构



接下来主要介绍技术实现的细节：

1、自然语言理解

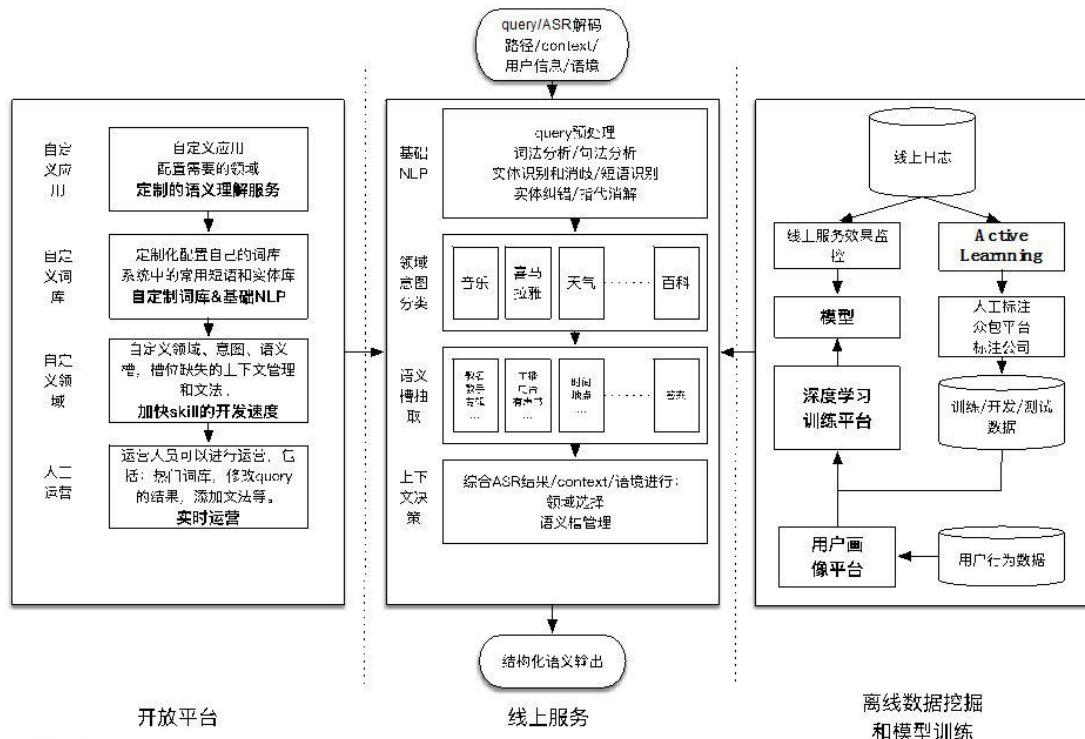
任务：**理解用户说的话，输出结构化语义表示**。目前自然语言处理的**难点包括用户说话的多样性、口语歧义、语义边界模糊、修改语义复合语义以及上下文相关联等。**

自然语言理解难点

- ◆ 说法的多样性
 - ◆ 预约周日的临时保洁。/周末可以帮我叫一个家政阿姨吗？/我想找个钟点工星期天来家里打扫卫生。
- ◆ 口语歧义
 - ◆ 帮我定下周二的会议。/我要去拉萨。
- ◆ 语义边界模糊
 - ◆ 明天适合穿短袖吗？明天适合穿短裤吗？明天适合穿短裙吗？明天适合穿超短裙吗？
- ◆ 修改语义，复合语义
 - ◆ 25号下午三点提醒我去开会，哦不是，26号。
 - ◆ 纠正纠正，不是8288，是8188
- ◆ 上下文
 - ◆ 我想订去上海的机票→那边的天气怎么样
 - ◆ 我想订机票→【你想去哪里？or 你想从哪里出发？】→上海。

基于此任务目前我们的技术框架如下：

自然语言理解技术架构



中间部分是线上服务，输入是 query/ASR 解码路径/context/用户信息/语境等内容，首先在基础 NLP 工作中进行 query 预处理/词法分析/句法分析/实体识别和消歧/短语识别/实体纠错/指代消解等。之后进行领域意图分类，利用浅层语法分析结果判断用户在哪个领域以及他在这个领域中到底在问哪个方面的问题，接下来进行语义槽抽取模块，在当前领域下对用户 query 的槽位进行抽取，对领域和槽位进行综合管理得出用户真正的意图做一个结构化的输出。对于线上服务存在很多离线工作需要完成，右侧是离线数据标注和挖掘，线上的设备会积累大量的线上日志，线上日志具有两个作用：1) 随机抽取一些样本来做一个每日的效果监控，本质上是对用户提供服务，所以要对用户体验层的效果进行监控。2) 通过 active learning 筛选出系统最薄弱的环节，对最薄弱环节的数据进行人工标注后混入训练模型，可以明显提升效果。左侧是开放平台，开放平台希望能够通过用户自定义的一些数据、词典配置属于自己的领域。

首先介绍数据部分，数据主要分为两个部分，一是词典资源数据，二是线上数据，在这里需要强调的一点是对于 NLU 这部分词典资源数据的清洗是非常重要的，我们一般直接拿过来的数据无论是音乐的数据还是有声节目的数据都带有非常大的噪声。这部分工作会消耗大量的人力，而且非常重要，如果资源清洗的不好，所有的线上 NLU 效果都会变差，必须通过规则对数据进行大量清洗。此外，我们要对资源进行分级，因为无论是多媒体资源、歌曲、视频还是有声节目，资源名之间都会有大量的冲突和复用情况，我们需要确定该资源属于常用资源还是有歧义的资源，以及有些资源会与常用语例如星期天有冲突。

数据

- ◆ 线上数据标注
- ◆ 词典资源数据清洗与分级

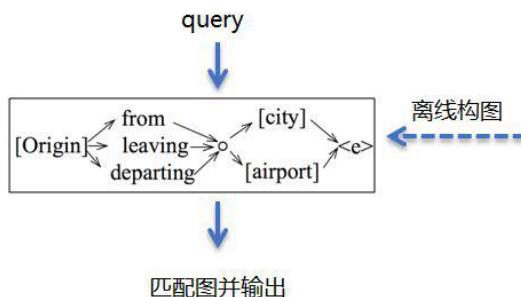
原始名称	类型	播放量	清洗后名称	分级
夜曲	music_song	115684908	夜曲	LOW
童话(钢琴演奏版)	music_song	110954635	童话	LOW
轻松	music_song	100448281	轻松	LOW
流行歌曲	music_song	100418716	流行歌曲	LOW
那个	music_song	100382305	那个	COMMON
等 (Live)	music_song	105102478	等	COMMON
火	music_song	100763471	火	COMMON
鬼故事	fm_album	38427000	鬼故事	LOW
儿童歌曲(精选)	fm_album	28269055	儿童歌曲	LOW
养生	fm_album	6415000	养生	COMMON
国画	fm_album	10537000	国画	COMMON
幻想曲	fm_track	1331366	幻想曲	LOW
新时代	fm_track	1058166	新时代	LOW
谢谢	fm_track	1320333	谢谢	COMMON
【直播回听】秘密!	fm_track	1210205	秘密	COMMON

在得到数据之后我们开始做自然语言理解，目前是基于 CFG 搭建的一个端到端的语义理解引擎，支持通配、量词，上下文等。直接输出领域、意图和槽位。端到端的文法语义理解用于领域在冷启动阶段的语义解析，NLU 的新需求大多基于此进行快速开发。在积累了大量的线上数据之后，唯一的途径是通过数据标注加模型训练来提升效果。

自然语言理解

● 端到端的文法语义理解

- 基于CFG规则引擎进行改进，
端到端、支持通配、量词，上下文等
- 直接输出领域、意图和槽位



```

1 #include"utils.gra"
2 #include"hci_dict.gra"
3 #include"location.gra"
4 #include"time.gra"
5 [main]
6 >---[motion_main]{domain=motion}
7 ;
8
9 [motion_main]
10 >---[move]{intent=move}
11 >---[turnaround]{intent=turnaround}
12 ;
13
14 [move]
15 >---([forward]{direction=前进} *[verb] [number]{distance})
16 >---([backward]{direction=后退} *[verb] [number]{distance})
17 ;
18
19 [turnaround]
20 >---([turn_prefix] 左{direction=左} *转 *动 [degree]{angle})
21 >---([turn_prefix] 右{direction=右} *转 *动 [degree]{angle})
22 >---([turn_prefix] 后{direction=后} *转 *动 [degree]{angle})
23 >---([turn_prefix] 前{direction=前} *转 *动 [degree]{angle})
24 >---(左{direction=左} 转 *动 [degree]{angle})
25 >---(右{direction=右} 转 *动 [degree]{angle})
26 >---(后{direction=后} 转 *动 [degree]{angle})
27 >---(前{direction=前} 转 *动 [degree]{angle})

```

自然语言理解

● 端到端的文法语义理解

- 用于领域在冷启动阶段的语义解析
- NLU新需求大多基于此进行快速开发

● 积累了大量的线上数据之后，如何提升效果？

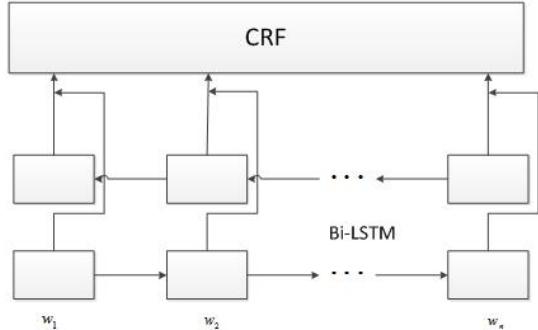
- 数据标注
- 模型训练

首先我们将领域分类模块替换为模型，这里我们基于 LSTM 做了一个模型层面的意图和领域分类模型，输入是两维特征：字向量和实体识别特征，作为所有意图分类模型基础输入。模型由一个双向的 LSTM 加一个单向的 LSTM 构成，最后通过 softmax 做一个领域的选择。意图分类同样基于此模型，在领域分类上准确率可以达到 93%，在意图分类上准确率可以达到 96%。

自然语言理解

●槽位抽取

- 基于 pattern 的抽取
- 基于模型的抽取
序列标注问题
BLSTM 分类模型 (计算概率) +
CRF (选择最优合法序列)
- 线上总体准确率 92%



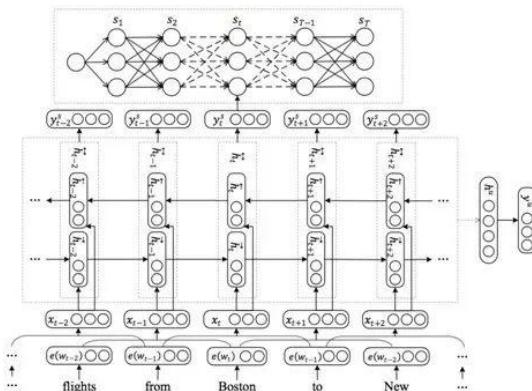
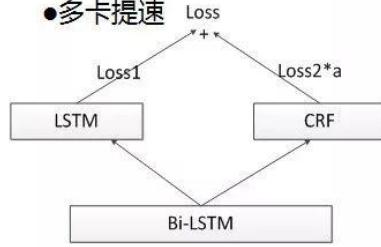
槽位抽取分为两种，一是基于 pattern 的抽取，二是基于模型的抽取，模型采用比较成熟的双向 LSTM 加 CRF，线上总体准确率可达 92%。

双向 LSTM 内层特征可以做一些共享和互补，这对系统的整体效果是有帮助的，我们借鉴了一些公开的论文做了一个联合学习模型，将 slot 抽取和意图分类最终做了一个统一的损失函数输出，获得一个统一的模型。联合学习模型在全局进行意图分类和槽位抽取，槽位抽取采用 CRF 做最终打标，意图分类也是 softmax，但在最上层将两个 loss 进行相加，相加之后反向传播所有的误差来更新模型的参数。

自然语言理解

● 联合学习模型

- 全局意图分类 + 槽位抽取
- 槽位抽取采用 CRF 做最终打标
- 多卡提速



A Joint Model of Intent Determination and Slot Filling for Spoken Language Understanding
Xiaodong Zhang and Houfeng Wang, IJCAI2016

相对于意图分类和槽位抽取模型，联合学习模型取得了较大的效果提升。

模型	domain	intent	slot
线上	90.79%	93.91%	91.59%
联合	95.74%	98.23%	94.08%
提升	4.95%	4.32%	2.49%

2、智能问答

我们还有智能问答这个方向，本质上是用户提出问题，回复正确的答案，技术点包括问句分析，通过对用户 query 的理解解析出 query 是在做哪些方向数据库的查询。此外还有知识库构建和答案生成。实现方法有三种：**1) 基于检索式 QA 系统；2) 基于知识图谱的 QA 系统；3) 文档阅读理解。**

- 智能问答任务
 - 给出问题，回复正确的答案，例如：“中国最长的河是什么河？”，“黄河”。
- 技术点
 - 问句分析
 - 知识库构建
 - 答案生成
- 方法
 - 基于检索式QA系统
 - 基于知识图谱的QA系统
 - 文档阅读理解

检索式问答的难点有两个：**知识库构建和句子相似度计算**，包括有难以建立准确全面的知识库，相似 query 不同义以及同义 query 差别大等。

检索式问答难点

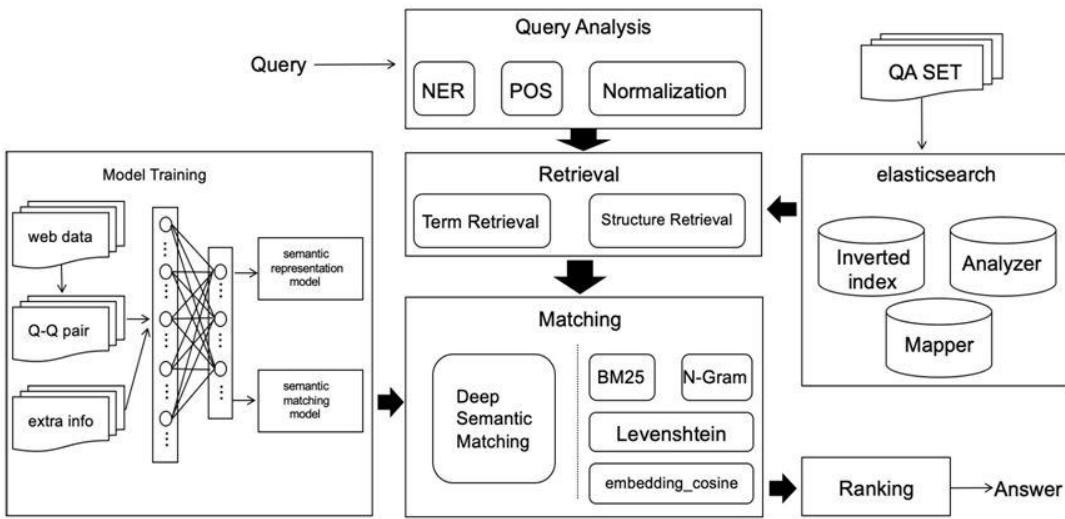
- 知识库构建+句子相似度计算
- 难点
 - 准确全面的知识库
 - 相似query不同义
 - 小宝宝生病怎么办？狗宝宝生病怎么办？
 - 食人鱼能吃吗？食人鱼能吃人吗？
 - 同义query差别大
 - 我的睡觉时间怎么黑白颠倒？为什么我白天很困但晚上睡不着？
 - 我的头发越来越少咋办？我有点秃头怎么处理？

目前我们线上跑的系统主要是基于检索式的智能问答系统，query 进来后先进行 normalization，根据离线构建的知识库构建出与用户最相似的 query，做语义相似度计算，经过 rank 后得到 answer。此外，由于 ASR 存在许多识别错误，我们会做一些拼音层面的模糊匹配，实体相关的模糊反查这样一些工作。

检索式智能问答架构



检索式智能问答架构



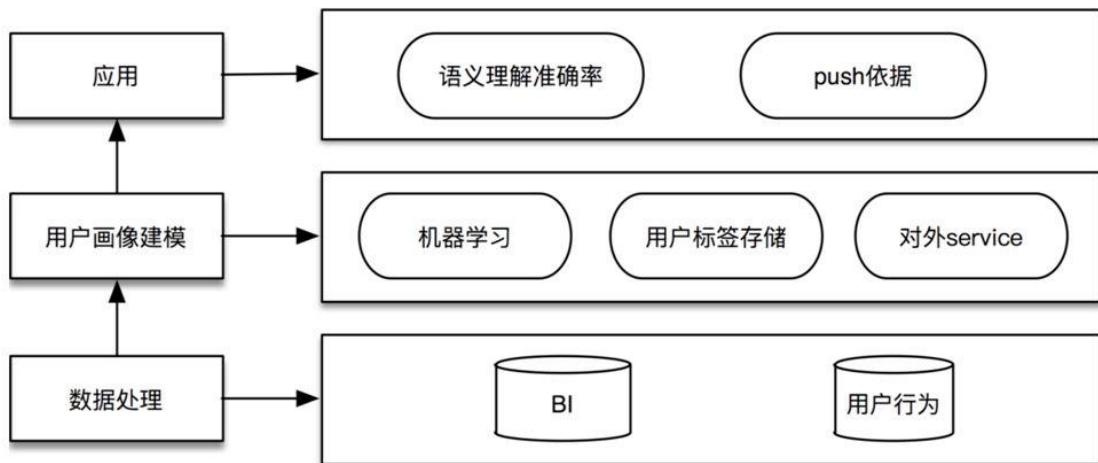
从细节来说分为三个部分：

- 1) 离线模型训练；
- 2) 在线的 query 分析和相似度计算；
- 3) QA 库的构建。

3、用户画像

我们还有用户画像这个模块，在离线积累大量的 BI 数据和用户行为数据，数据处理后通过标签的归一，机器学习的一些建模方法把相关的 BI 数据构建出用户的多维度特征，最终拿用户画像这个基础 service 在上层做一个语义理解以及内容推送，广告系统的一个应用。

用户画像



用户画像根据设备 id 和声纹来唯一指定一个用户，例如音箱在家庭场景下可能会有多个使用者，所以通过设备 id 和声纹两个信息来唯一描述音箱的一个属性。因为我们会依据用户主动 query 及点播情况来刻画用户，我们描述的主要是一些相关的有声内容的维度以及一些用户的基本信息，比如用户的地点、性别、职业等。用户交互的频度，用户每个 query 的分析统计情况，她的资源完成情况，在每种类型下资源的耗时情况，以及每种耗时会在置信度上计算一个权重，用户收藏内容的情况，用户使用音箱时的时间分布，对于不同的人群，使用音箱的时间分布是不一样的。

用户画像

- 根据设备id和声纹来唯一指定一个用户
- 依据用户主动query及点播情况来刻画用户
- 主要domain: music, fm, children_song, children_story, news, nature_sound
- 基本维度
 - user_info
 - query count
 - res count
 - res dwell time
 - res duration
 - weight
 - collection_info
 - user_habits
 - time_distribution
 - total_info

在用户画像这个 service 单元之后我们的应用包括：

- 1) 内容自动化推送；
- 2) NLU 综合决策；
- 3) 内容串起 fm,music；
- 4) 用户大数据统计。

用户画像

- 内容自动化推送
 - user-based, content-based, knowledge-based
tag expand ; Cosine similarity
 - 提升30%的有效转化
- NLU综合决策
 - 个性化解析结果，对NLU结果进行重排
 - 给我放点儿歌
- 内容串起 music, fm
 - 模板代入：小豹发现你喜欢【style】的歌，给你推荐【song】
- 用户大数据统计

4、聊天系统

聊天系统基本上是模仿人类的风格，对自然语言的文本，做出相应的回复。目前我们的聊天系统具有两个特色：

- 1) 支持不同的业务线，每个业务线都有一套人设，现在我们有萌妹型、大叔型以及职场型等人设。
- 2) 用户情感及机器情感，将机器看做是一个人，内部装有机器情感引擎。用户在交流过程中是有情感的，例如用户可能会说：“今天我不开心。”机器人会回复：“不开心，那我们出去玩吧！”但是如果用户在不断地谩骂机器人，机器人就会接受用户的每个 query 并且在每一个情感维度上去做一个权重的累积，如果用户一直谩骂下去机器就会在一定程度上不再理你或者进行反击。

聊天系统

- 任务定义
 - 模仿人类的风格，对自然语言的文本，做出相应的回复
 - 每个业务线都有一套人设

Aini:Hi!!	你好:You
Aini:你好，我叫小雅，来自银河系猎户座，是一个人工智能美少女哦	你多大了:You
Aini:小雅年芳十八，正是青春好年华。	我爱你:You
Aini:我能想到最浪漫的事，就是和你慢慢变老	我帅不帅:You
Aini:帅，惨不忍睹的那种	

最后，给出 NLP 方向的一些实用建议：

- 1) 冷启动阶段只能文法规则先上；
- 2) 一定关注线上真实用户的数据情况，关注 badcase；
- 3) 一定要增加运营干预手段；
- 4) 一定要强化对资源数据的清洗和挖掘理解；
- 5) 线上数据标注+模型，提升整体性能。

四、猎户 NLP 相关产品

猎户NLP支撑的产品



目前猎户星空 NLP 能够支持的一些相关产品应用包括有：

豹小秘：五星级服务接待机器人，能够完成前台接待、主动问候、人脸跟随、视频通话、拍照合影、远程控制等多种功能。

豹豹龙：豹豹龙是一只来自于“猎户星座 参 (shēn) 宿 (xiù) 四号星球”的 9 岁小恐龙，具备超凡的听觉、视觉和运动细胞，同时性格温顺而又具有爱心。他利用猎户星空流传下来的先进科技帮助地球小朋友变得更聪明！

豹小贩：会行走的自动贩卖机，利用人脸识别、动态跟踪等进行分析判断，利用云端数据库处理，智能分析真实需求，提供智能服务。

豹咖啡：豹咖啡是依托猎户机械臂技术的机器人咖啡亭，使用“香格里拉酒店”同款咖啡豆，保留新鲜豆粒精华，运用 Arm OS 精确模拟 WBC 咖啡大师手法，利用基于深度学习的视觉识别技术智能抓取、精准判断，只需一键点单，就能让机械臂为你做一杯好咖啡。

小豹 AI 音箱：首创主动交互，听声识人，AI 私教，还有 100+ 生活小功能。

小豹翻译棒：支持中英、中日、中韩一键翻译，满足国人 75% 出境游需求。

作者介绍：

韩伟，猎户星空自然语言处理平台技术负责人。研究包括：语义理解，智能问答，机器翻译，多轮对话，聊天系统等技术方向，相关技术在豹小秘，豹豹龙，小豹智能音箱，小雅智能音箱，小豹 AI 翻译棒等多个智能硬件上落地。

内推信息 :

- (1) NLP 产品经理 : 1.负责公司 NLP 服务平台的产品设计和规划 , 跟踪接口的效果评估和功能迭代 ; 2.负责语音交互等场景的 NLP 产品规划和设计 ; 3.负责跟踪 AI 类前端产品方向并规划 NLP 相关产品及应用 ;
- (2) 后端工程师 ;
- (3) 前端工程师 ;
- (4) 自然语言处理高级工程师

工作地点 : 北京

- 1、负责人机交互系统中自然语言处理相关核心技术研发 ;
- 2、负责人机交互系统在线服务的维护和相关业务的开发 ;
- 3、负责基于大数据分析相关算法建立内容分类模型和用户画像 ;
- 4、探索自然语言处理前沿技术 , 并应用于语音人机交互系统中的语义解析、对话系统、个性化推荐等系统等 ;

有同岗位经验的同学优先哈。

有意向者欢迎投递简历至韩伟老师邮箱 : hanw@ainirobot.com

智变中的美团客服

作者：刘学梁 整理：赵富旺

当 NLP 遇上客服系统确实会发生一些美妙的事情，下面我分享的内容将围绕美团客服系统中一些比较前沿的技术展开，希望能对大家有所启发。

今天分享的内容主要分为以下几个方面，第一部分首先对美团的客服系统进行简单介绍，第二部分是本次分享的重点，包括我们采用了什么样的技术方案，技术方案的具体的技术组件是怎样实现的，背后有着怎样的思考等等，第三部分展示我们所做的工作在美团客服系统的落地效果，第四部分对本次分享做一个总结。总体上来说，客服不是一个纯靠人可以解决的问题，也不是一个纯靠算法可以解决的问题，而是需要人机协同解决的问题。同时，它也不是一个静态的系统，它需要不断地进化不断地运营。

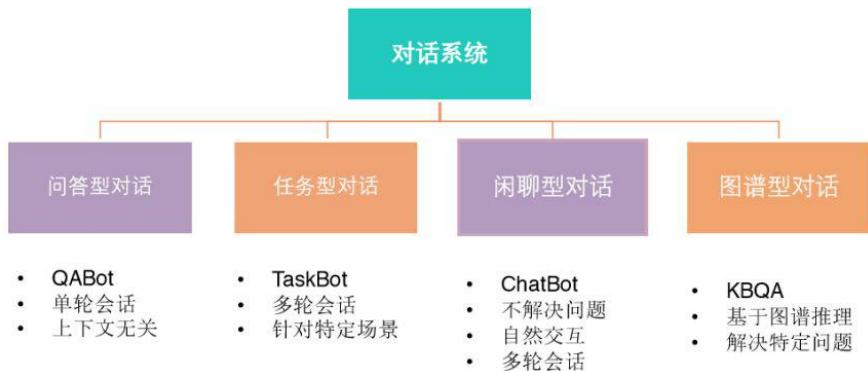
1. 客服系统简介

1.1 演变中的客服系统

智能化是客服系统演变的方向，智能客服通过算法、人机协同、自主学习可以极大地提高人效和体验。



首先我们回顾一下客服系统的演变历史。客服系统的原始阶段是语音呼叫中心，这种客服系统纯靠人工服务，且支持语音电话，效率低成本高。第二阶段进化到了网页在线客服，这种客服系统基于网页会话，服务形式支持文本和语音，同时还利于对流量数据进行挖掘。随着移动互联网的兴起，便有了 SaaS 客服系统，这种客服系统支持多渠道接入，有了丰富的辅助功能和知识库管理。如今客服系统进化到了智能客服，它最大的特点就是人机协同，许多简单问题都可以由机器自主解决，这个系统可以自主学习不断进化。回顾客服系统的演变历史可以发现，智能化是客服系统的一个演变趋势。



然后我介绍一下对话系统。对话系统主要包括四类：问答型对话、任务型对话、闲聊型对话、图谱型对话。在问答型对话中，我们使用 QABot 机器人完成简单任务，这种对话通常是与上下文无关的单轮对话。在任务型对话中，TaskBot 机器人完成特定场景下的复杂任务。另外还有 ChatBot 闲聊型机器人，这种对话通常不以解决实际问题为目的，我们的客服系统也有用到这种机器人。最后是图谱型机器人 KBQA，这种机器人可能更多地用在金融、医疗等领域，但还未有成熟的系统，这方面我们也还处于探索中。

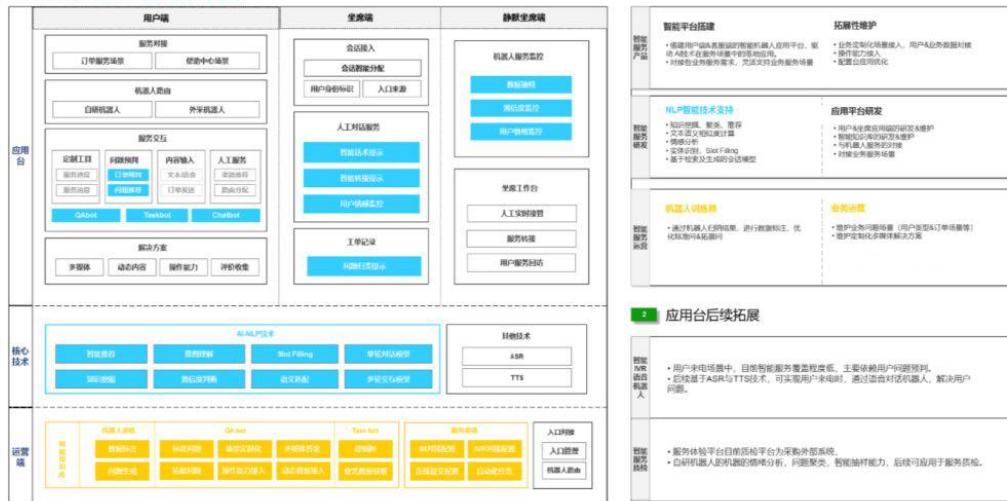
1.3 智能客服机器人



根据智能客服机器人的智能水平，可以将其分为四个档次：简单检索机器人、语义识别机器人、场景导向机器人、智慧机器人。简单检索机器人只支持特定类型的检索，只要说法稍微一变可能就不能正确识别，匹配性较差。语义识别机器人基于知识库，可以更智能地理解所检索的问题。场景导向机器人根据不同场景量身定制机器人，机器人的聪明程度与场景有关。

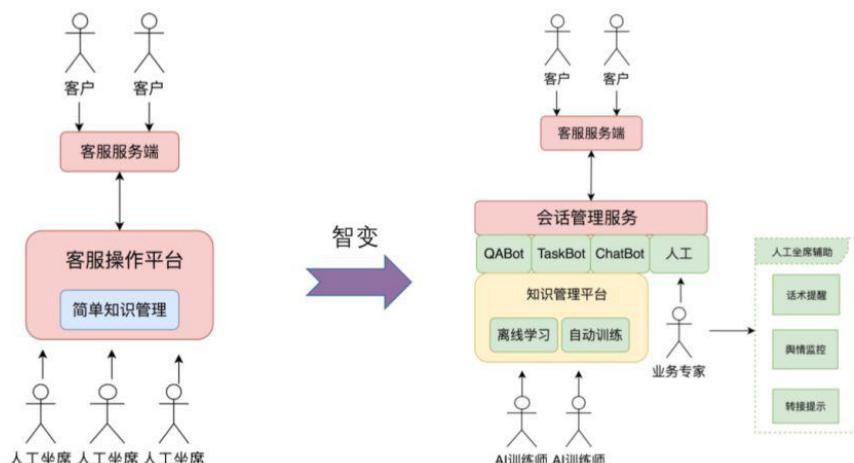
智慧机器人是智能程度最高的机器人，甚至可以达到拟人的程度。现在来看，大多数机器人还只是停留在第二个阶段，能达到第三个阶段的还是少数。

1.4 美团客服系统



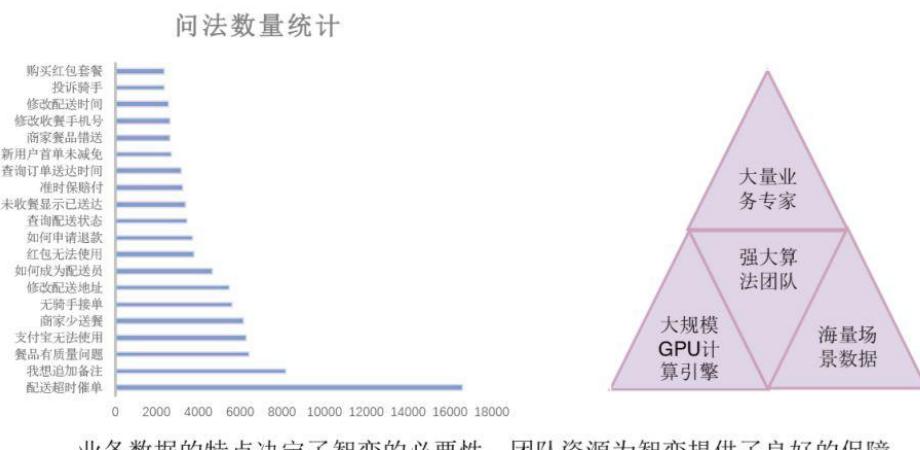
美团的业务种类繁多，不同的业务所需要的客服系统也不尽相同，这无疑给我们提出了严峻的技术挑战。上图是客服系统的整体框架图，蓝色部分代表了前面提到的 nlp 技术对原来客服系统的改进。在用户端我们设置了 QABot、TaskBot、ChatBot 等机器人，在座席端运用了话术提示、转接提示、情感分析等技术。

2 智变之路



下面进入本次分享的第二部分：智变之路，主要聚焦于我们在这个过程中做了哪些工作以及我们背后的思考。

原来的客服系统中客户将请求传送到客服服务器，然后客服操作平台就会分配相应的人工客服处理相应的客户请求，客服操作平台只具备简单的知识管理功能。这种客服系统最大的问题就是效率低，需要的人力成本高。对于这样的客服系统来说，实际上需要的人工客服数目和订单数目是成正比的。美团现在的业务正处于飞速的发展过程中，现在就有近万名客服人员，如果不对客服系统进行改进，可以想象未来这个队伍还会扩充很多倍。基于原来客服系统的这些缺点，我们对这个架构进行了改进，增添了会话管理服务，后面连接着 QABot、TakBot、ChatBot 等机器人以及人工服务，有一个专门的知识管理平台来支撑 QABot、TakBot、ChatBot，AI 训练师对知识管理平台设计离线学习和自动训练的算法。除此之外我们还设计了话术提醒、舆情监控、转接提示等模块来辅助人工客服。

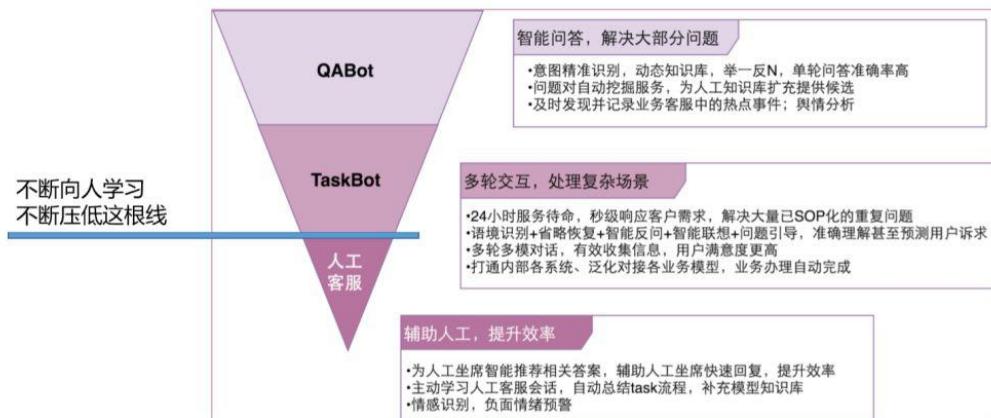


业务数据的特点决定了智变的必要性，团队资源为智变提供了良好的保障

我们也研究过做字典的必要性。以上图片来源于对外卖的日志数据分析，可以看出只是“外卖配送超单”这一个问题对应的问法就有 16000 多种，原客服系统的简单检索很明显不能满足这种需求，让检索系统具有一定的语义识别能力十分必要。

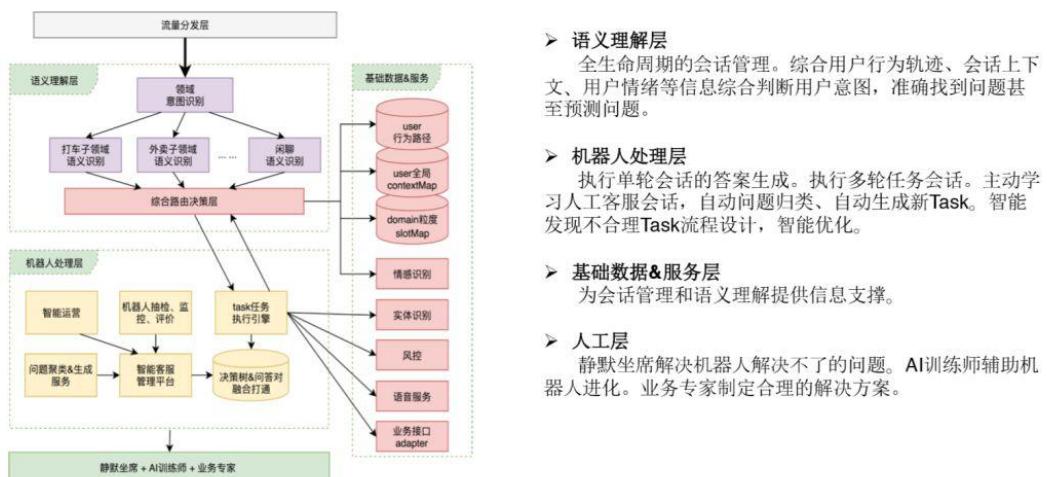
要做好一个 AI 系统必须满足这样的条件：要有大量业务专家、要有强大算法团队、要有大规模 GPU 计算引擎、要有海量场景数据，这些条件在美团都是满足的，这无疑给了我们使智能客服系统落地的信心。

2.2 智变途径



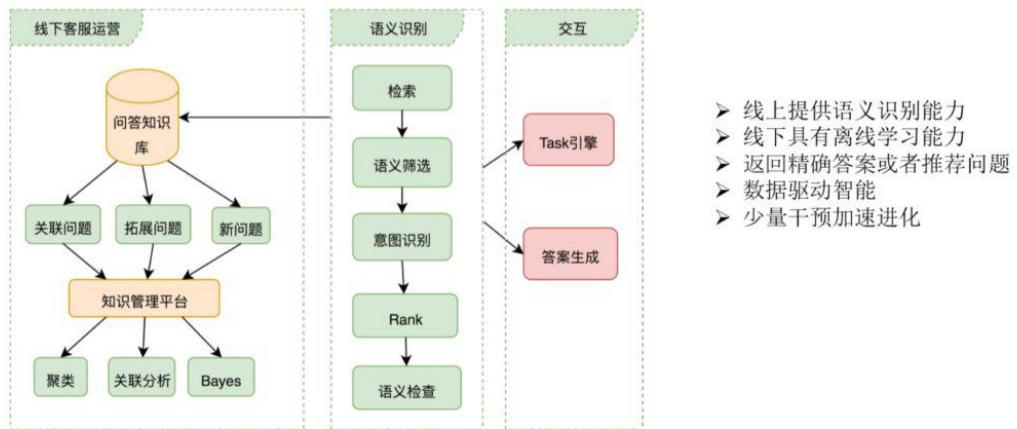
这是我们智变的思路，我们采用三级漏斗的方式，把问题分为了三类，第一类简单咨询问题由 QABot 解决，第二类高频相关的复杂场景下的问题由 TaskBot 解决，最后 TaskBot 解决不了的任务再借助人工客服来解决。我们算法的目标就是随着时间的演进，不断地把更多的任务转向机器解决。

2.3 系统架构



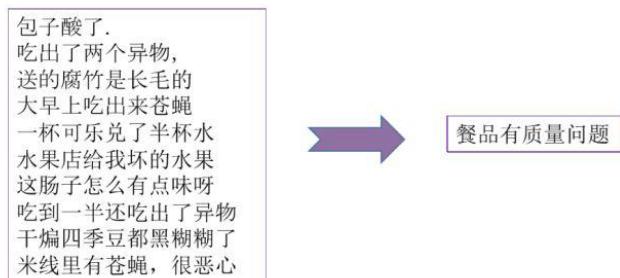
这是我们的系统架构，当一个请求来临时，先进入我们的意图识别领域，然后被识别到对应的业务领域中，把对应的知识点识别出来。如果识别出来的是一个比较简单的问题，直接检索就可以得到答案，如果它是和用户订单状态或者行为有关系的复杂问题，需要根据场景生成不同的答案。比如对于催单问题，订单状态不一样所需要的回应也不一样，如果用户刚下订单就要催单，此时告诉用户骑手的位置比较合适，但用户是在等了一个小时的情况下催单，此时需要先安抚客户的情绪。而对于多轮的任务，如客户支付宝支付无法使用，这种任务需要调用好多层才能完成，此时就会调用 TaskBot。

2.4 单轮会话机器人QABOT



接下来介绍一下单轮会话的 QABot。它主要由两部分构成，其中一部分是线上的语义识别，另一部分是线下的客服运营，运营的目的是发现更多的标准问题以及更多的关联问题。另外后面还有一个交互层，交互层有可能触发 Task 或者直接生成答案。

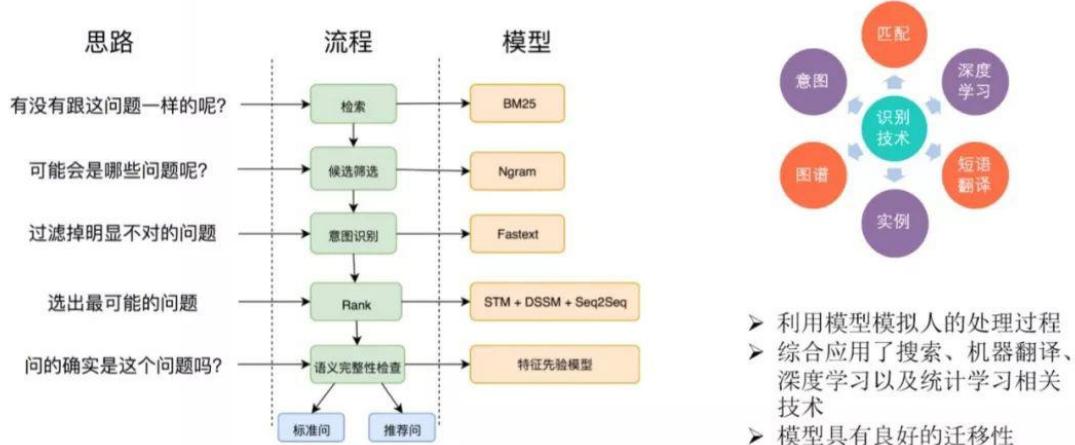
2.4.1 语义识别



- 理解用户的问题
- 综合利用搜索技术、翻译技术、图谱技术、深度学习和统计学习技术

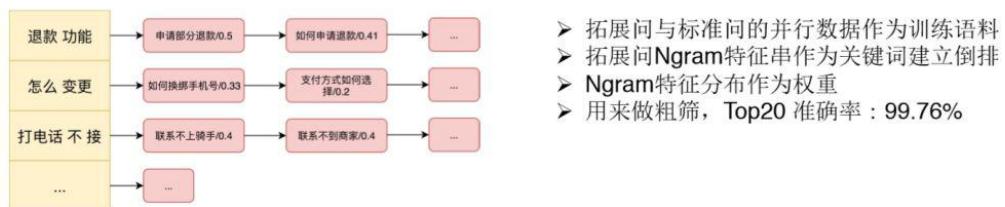
语义识别主要指把不同的问法规整到标准的问法上去，感受一下上面的例子，不同的细节表述其实都是在说餐品有质量问题，而语义识别的目的就是要找到检索问题的标准表述。为了解决这个问题，我们综合运用了搜索技术、翻译技术、图谱技术、深度学习和统计学习技术。

2.4.2 语义识别流程



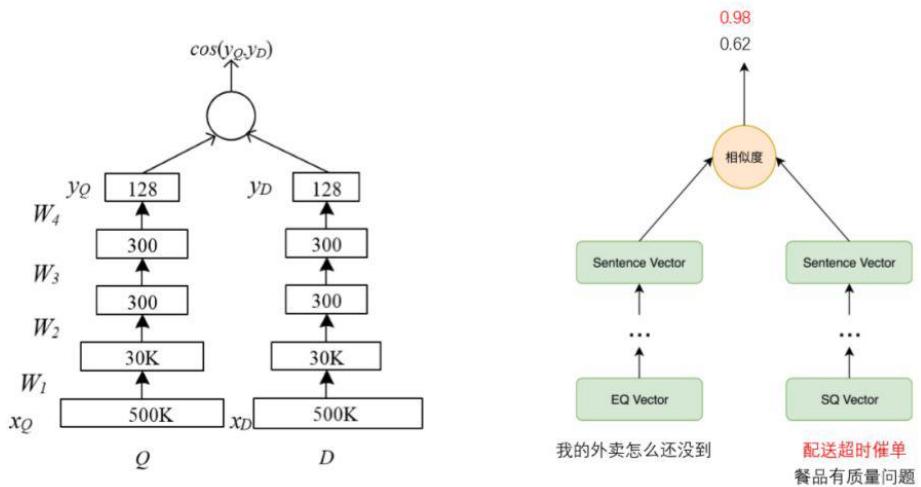
语义识别的流程模拟了人解决问题的思路。人类在解决一个问题的时候会首先考虑以前有没有类似的问题，对应于语义识别中我们也首先采用搜索检索的方法。在找寻相似问题的过程中，人类通常会考虑：“我们要找的到底是哪些问题呢？”，对应于语义识别中，这是一个对问题进行候选筛选的过程。当人类发现一些问题明显和意图无关，通常要把他们去掉，对应于语义识别中，这就是要进行意图识别从而筛选问题。面临最后挑选出的几个问题，人类通常要对其进行优先级排序，对应于语义识别中，这就是 Rank 的过程。最后人类可能还会对结果进行检查：“问的确实是这个问题嘛？”，这就是语义识别中的语义完整性检查。这整个过程中运用了各种各样的模型来达成语义识别的目的。

2.4.3 基于匹配的识别



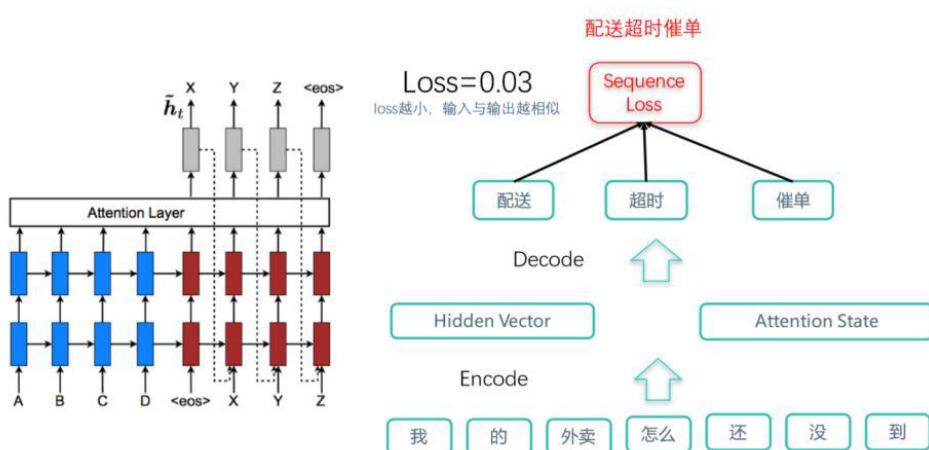
先介绍一下基于匹配的识别。基于匹配的识别利用拓展问与标准问的并行数据作为训练语料，把拓展问 Ngram 特征串作为关键词建立倒排，Ngram 特征分布作为权重，在外卖的例子中只用 Top20 就能达到 99.76% 的权重。这一步的筛选为后面模型节省了很多时间。

2.4.5.1 基于深度模型的识别--DSSM



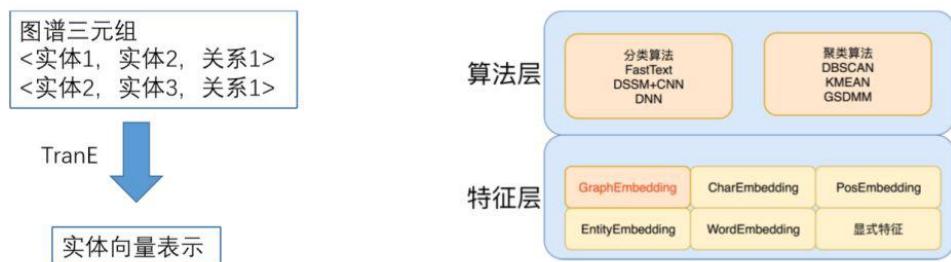
下面介绍一种深度模型—DSSM 模型。DSSM 模型是一个双塔模型，它在句子 embedding 上效果很好，因此我们也借鉴了这一模型。我们把标准问与拓展问语义相同的句子对作为正例，把标准问与拓展问语义不同的句子对作为反例，训练了 DSSM 模型。模型训练后，对于任给的一个问题，可以得到它的 embedding 结果，和其他标准问 embedding 结果相对比就可以算出相似度。比如拓展问“我的外卖怎么还没到”可以计算出一个 Sentence Vector，而标准问“配送超时催单”和“餐品有质量问题”分别有一个 Sentence Vector，通过计算可得，“我的外卖怎么还没到”和“配送超时催单”的相似度为 0.98，我的外卖怎么还没到”和“餐品有质量问题”的相似度是 0.62，所以可以把“我的外卖怎么还没到”的语义识别为“配送超时催单”。

2.4.5.2 基于深度模型的识别--Seq2Seq



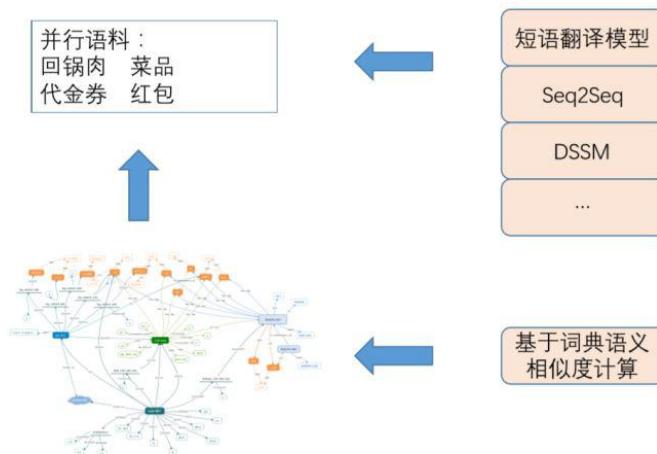
还有一个效果很好的深度模型—Seq2seq 模型。Seq2seq 模型本身是一个生成模型，但是我们把它用来计算句子之间的相似度，我们把它 encoder 和 attention 的结果和不同的候选做 loss 计算，把 loss 作为一种度量结果，Loss 越小代表输入和候选越接近。以上两个深度模型是我们尝试过的深度模型中效果最好的两个。

2.4.6.1 基于知识图谱的识别-隐式



我们还采用了一种基于图谱的识别方法。因为美团积累了很多图谱信息，美团大脑项目里面有亿级的实体数量，相互之间的关系超过 5 亿，实体与实体之间的关系可以做一个 embedding，这样实体就可以变成一个向量表示，而这种向量可以作为一种特征和其他显式特征联合在一起提供给聚类分类算法，这样会使算法效果有很大的提升。

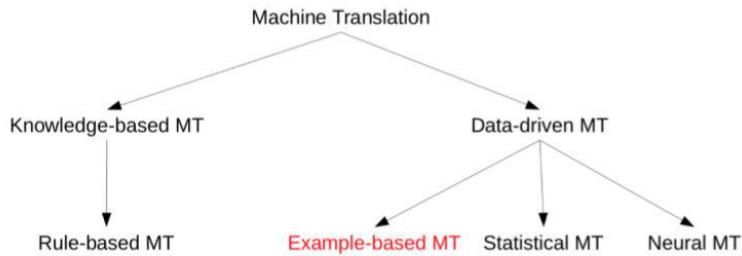
2.4.6.2 基于知识图谱的识别-显式



还有一种对知识图谱的用法，我们从图谱中抽取一些近义词上下位等关系作为并行语料放入一些模型中去，这样对原有语料进行了很好的扩充。

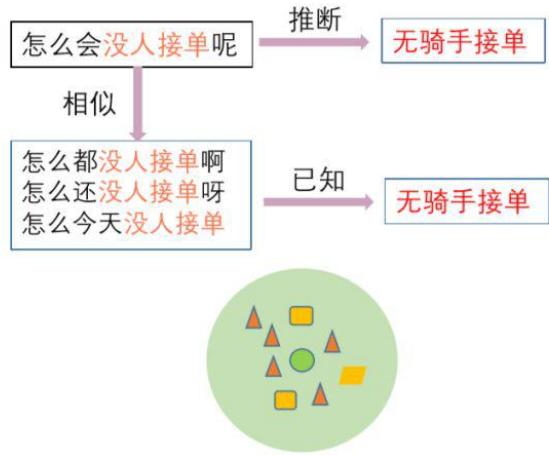
2.4.7.1 基于实例的翻译

Man does not translate a simple sentence by doing deep linguistic analysis.
--Nagao 1984



还有一种模型：基于实例的翻译。Nagao1984年时提出过这样一个理论：人类在翻译简单句子时不会进行深层次的语义分析。这是一种很简单的思想：从已知的实体中找到和问题最接近的，把它的答案作为问题的答案。

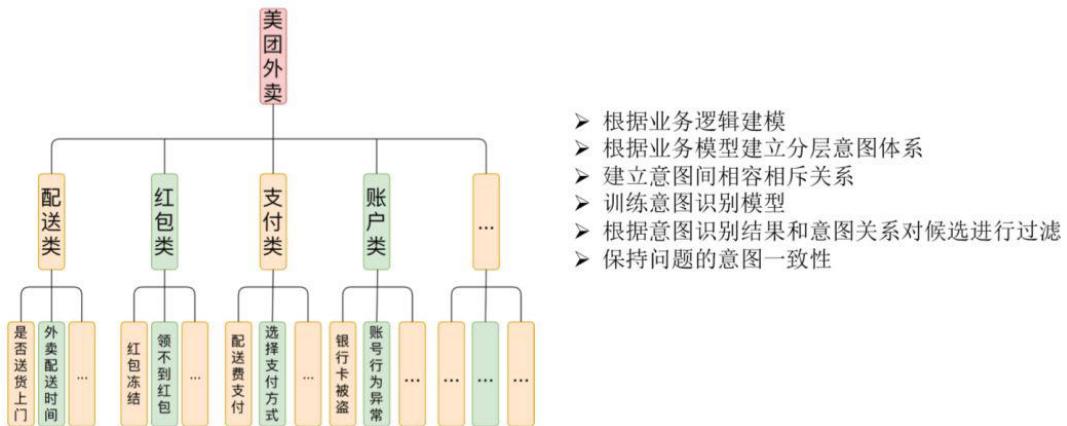
2.4.7.2 基于实例的识别



基于片断拼凑和句法分析计算相似度
特定相似度邻域内的样本进行投票决定待预测问题的答案

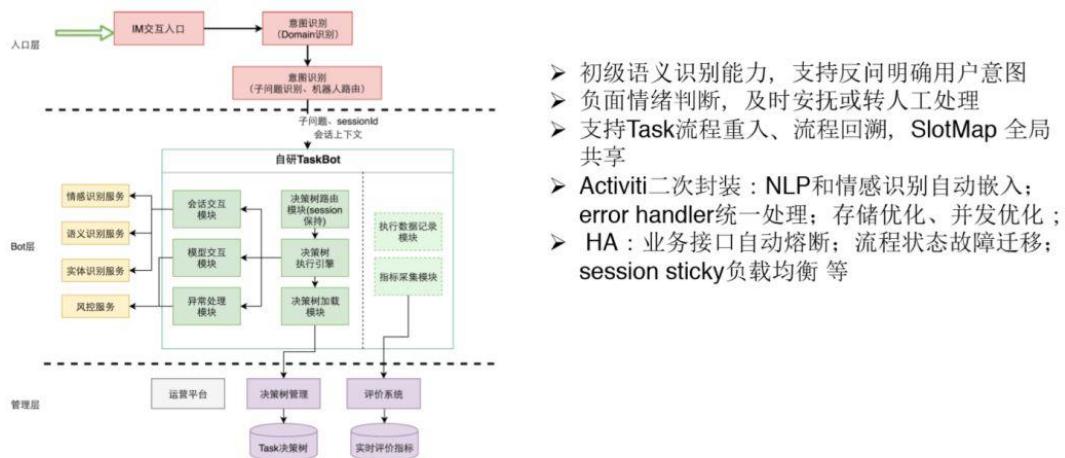
这是一个简单的例子。我们的语言库里面已经有“怎么都没人接单啊”、“怎么还没人接单呀”、“怎么今天没人接单”等句子，库里面这些问题的答案都是“无骑手接单”，此时来了一个新的问题“怎么会没人接单呢”，通过相似性可以推断出它的答案也应该是“无骑手接单”。

2.4.8 基于业务模型的意图过滤



下面介绍一个进行意图识别的模型。我们根据业务逻辑进行建模，对于外卖来说，可以把问题分为几个大类，比如配送类、红包类、支付类、账户类等等，不同大类之间的问题关系可以依据业务逻辑分别被定义为互斥或者相容，根据这些关系可以对候选问题进行语义层面的筛选。根据筛选结果对候选问题进行过滤。

2.5 TaskBot整体框架



上面介绍的都是 QABot 用到的模型，下面介绍一下 TaskBot 的整体框架。TaskBot 在我们这里被定义为一个执行引擎，在架构上它是这样的，请求来临后，先进行意图识别，出发对应的 Task，Task 把对应的决策树 load 到内存里边，然后它会对决策树的节点状态进行记录，之后调用情感识别、语义识别、实体识别等服务进行分析决定节点之间的状态流转。

2.6 闲聊机器人ChatBot



- 用户交流情感，不以解决实际问题为目的
- 主动问客户问题，收集信息以提供更好服务
- 不同客户机器人间切换的平滑剂
- 检索式：构建一个闲聊库，检索给出答案
- 生成式：从闲聊库学习生成模型

接下来介绍闲聊机器人。ChatBot 不以解决实际问题为目的，主要用来和用户进行情感交流。我们采用两种方式来做 ChatBot，第一种是检索式，构建一个闲聊库，检索给出答案，第二种是生成式模型，从闲聊库中学习生成模型。生成式比较具有挑战性，因为客服系统是一个比较严肃公开的平台，必须保证会话的可控性。

3.1 落地效果--QABot

实时响应、24小时永不离线、轻松应对高峰咨询量、比人工更有耐心



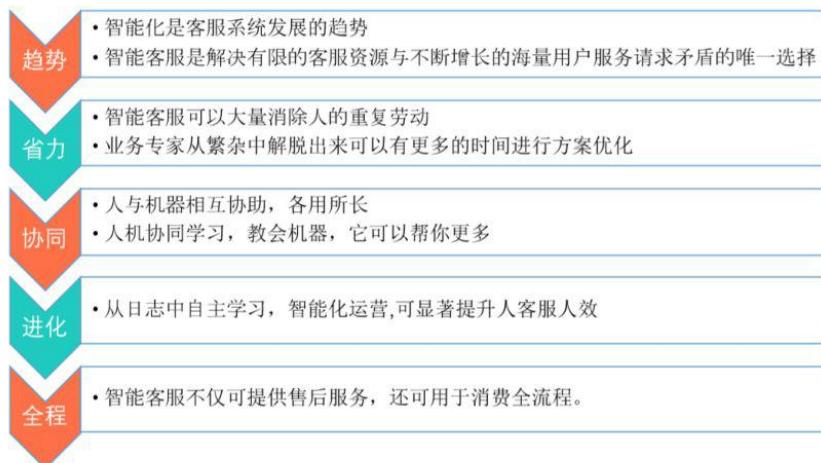
这是我们 QABot 的落地情况，数据来自于外卖场景，每天解决 72,000 个问题，算法离线准确率达到 92%，在线智能解决率 83%。

3.2 落地效果--TaskBot



这是 TaskBot 的落地效果，业务场景是打车领域，针对打车司机平台派单少这一问题，TaskBot 的上线方便了司机自助地解决问题，大大提升了这一问题的智能解决率和智能解决量。

4. 总结



最后，对本次分享做一下总结。智能化是客服系统发展的趋势，是解决有限的客服资源与不断增长的海量用户服务请求矛盾的唯一选择。实践证明，智能化客服确实可以大量消除人的重复劳动，业务专家也可以从繁杂中解脱出来，可以有更多的时间进行方案优化。最重要的一点，智能化客服系统不是一个纯人的系统，也不是一个纯算法的系统，也不是一个静态的系统，它需要人机协同，自主学习不断进化。还有一点，客服系统现在还主要用于售后方面，

我们现在也在售前的相关研究，后面我们也会把它用于智能营销、导购等流程中，这方面我们也在探索。

作者介绍 :

刘学梁，美团 AI 平台部 NLP 中心客服算法团队负责人，研发的智能客服系统已上线服务于外卖、打车等领域。曾就职于微信，从事机器翻译、语音识别相关基础算法研究工作。

团队介绍 :

美团点评 AI Lab-NLP 中心是负责美团点评人工智能技术研发的核心团队，使命是打造世界一流的自然语言处理核心技术和服务能力，依托 NLP (自然语言处理) 、 Deep Learning (深度学习) 、 Knowledge Graph (知识图谱) 等技术，处理美团点评海量文本数据，打通餐饮、旅行、休闲娱乐等各个场景数据，构建美团点评知识图谱，搭建通用 NLP Service ，为美团点评各项业务提供智能的文本语义理解服务。我们的团队既注重 AI 技术的落地，也开展中长期的 NLP 及知识图谱基础研究。目前项目及业务包括美团点评知识图谱、智能客服、语音语义搜索、文章评论语义理解、美团点评智能助理等。

内推信息 :

职级： p2-1 到 p3-2 , 如下方向： NLP (在知识图谱、智能客服、搜索引擎、推荐系统等领域有实际经验) 、智能助手和客服机器人、知识图谱等算法开发岗， Base 北京、上海。欢迎加入学梁老师的团队，简历请投递至学梁老师的邮箱：liuxueliang03@meituan.com

二手电商知识图谱构建以及在价格模型中的应用

作者 : 张青楠 整理 : Hoh

一、知识图谱概述

这次的分享主要从以下四个部分 : 知识图谱概述、知识图谱构造、转转二手电商知识图谱、在价格模型中的应用。

1.1 什么是知识图谱

知识图谱是谷歌在 2012 年提出来的，最初目的是优化其搜索引擎。在现实世界中是存在很多的实体的，各种人、物，他们之间是相互联系的。知识图谱就是对这个真实世界的符号表达，描述现实世界中存在的一些概念，以及它们之间的联系。具体来说是一个具有**属性的实体**，通过关系连接而成的**网状知识库**。

1.2 知识图谱的基本组成

在电商的知识图中，包括用户、商家、商品，他们带有各自的属性，彼此之间又互相联系。知识图谱的基本组成三要素：**实体、属性、关系**。实体-关系-实体三元组；实体-属性-属性值三元组，在电商的知识图谱中，用户和商品都是实体。

在知识图谱中，有一类特殊的实体叫做本体，也叫做概念或语义类。它是一些具共性的实体构成的集合。比如说，比尔盖茨和乔布斯都是人，微软和苹果都是公司。

二、知识图谱构建

目前的知识图谱分为两类。一类是开放域的知识图谱，另一类是垂直领域的知识图谱。比如谷歌为搜索引擎所建立的知识图谱就属于开放域的。垂直领域的知识图谱，比如说金融的，电商的。

首先就是要先处理数据。互联网上的数据基本上都是结构化的，非结构化的和半结构化的。结构数据一般就是公司的业务数据。这些数据都存储到数据库里，从库里面抽取出来做一些简单的预处理就可以拿来使用。半结构化数据和非结构化数据，比如对商品的描述，或是标题，可能是一段文本或是一张图片，这就是一些非结构化数据了。但它里面是存储了一些信息的，反映到的是知识图谱里的一些属性。所以需要对它里面进行一个抽取，这是构建知识图谱中比较费时费力的一个工作。

从数据里需要抽取的其实就是之前所提到的实体、属性、关系这些信息。对于实体的提取就是 NLP 里面的命名实体识别。这里相关的技术都比较成熟了，从之前传统的人工词典规则的方法，到现在机器学习的方法，还有深度学习的一些使用。比如说，从一段文本里面，我们提取出来比尔盖次这个实体以及微软这个实体，然后再进行一个关系提取。比尔盖次是微软的创始人，会有这么一个对应的关系。另外还有属性提取，比如比尔盖茨的国籍是美国。在这些提取完成之后都是一些比较零散的信息，然后在再加之前用结构化信息所拿到的东西以及从第三方知识库里面所拿到的信息做一个融合。

另外还需要做的是实体对齐和实体消歧。

关于实体对齐。举例来说，比尔盖茨这四个字是中文名称，Bill Gates 是他的英文名称，但其实这两个指的是同一个人。由于文本的不一样，开始的时候导致这是两个实体。这就需要我们对它进行实体对齐，把它统一化。

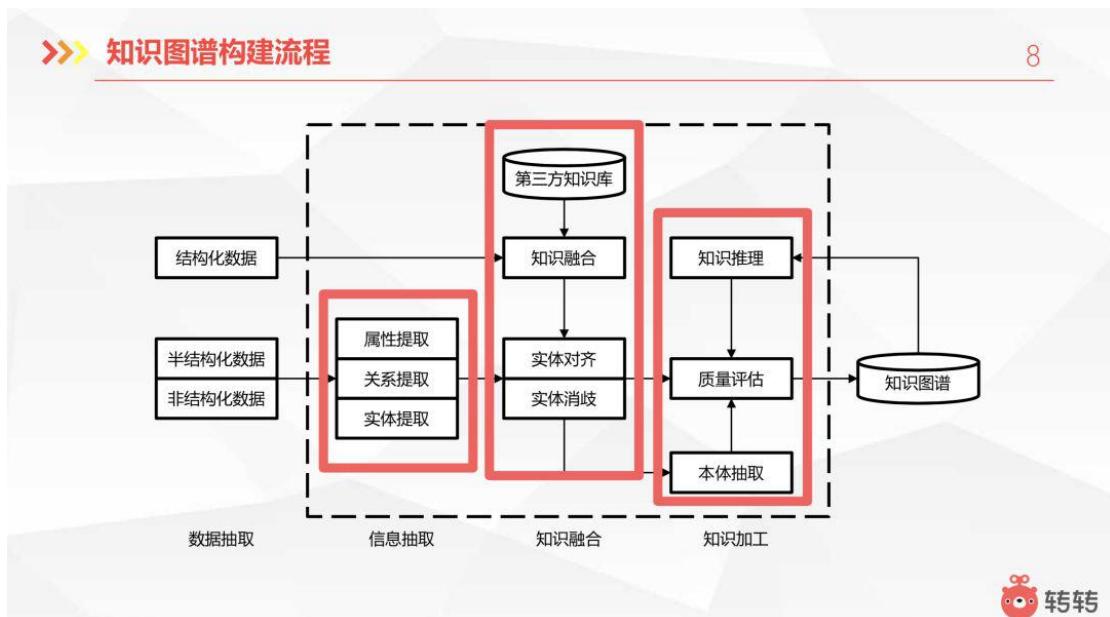
另外是实体消歧。举例来说，苹果是一种水果，但是在某些上下文里面，它可能指的是苹果公司。这就是一个实体歧义，我们需要根据上下文对它进行实体消歧。

在完成了以上步骤之后，接下来就是本体抽取。比如之前提到的微软和苹果，它们的实体是公司。从文本里面可能无法直接提取出来，它们是公司。那么需要一些方法对他们进行抽取。然后搭建出本体库，比如说公司是一个机构，它是有这种上下流的关系的。对于平级的也需要计算一个他们的相识度，比如比尔盖茨和乔布斯在实体层面，他们是相似的。他们都属于人这个实体。他们跟公司的差别还是挺大的，所以需要一个相似度的计算。

在以上步骤完成之后需要对知识库进行质量评估，这是一个避免不了的人工步骤。在做完质量评估以后，最终形成知识图谱。形成知识图谱以后，有些关系可能是无法直接得到的，然后需要进行知识推理，这可以对知识图谱进行扩展。比如，猫是猫科动物。猫科动物是哺乳动物。这就可以推理出来，猫是哺乳动物。但是这个推理也不是随便就可以推出来的。比如，比尔盖茨是美国人，比尔盖茨创建了一个公司，但这个公司并不一定是美国的。

>>> 知识图谱构建流程

8



三、二手电商知识图谱

主要从以下四个部分阐述：业务理解、知识图谱设计、算法、开发。

3.1 二手电商特性

搜索优化和个性化推荐是我们最开始所做的初衷。主要去做一些意图识别或是自动化查询这些。个性化推荐这里，我们利用知识图谱做一些召回源以及推荐排序模型特征。在电商运营这里，主要是帮助后台运营组货。在垂直业务这块，主要是做一些价格模型和供需关系分析。

二手电商不同于一手电商。首先就是数据源的质量。二手电商平台上面的商品都是个人发布。商品的描述信息不像商家那样完整。我们提供给他们的可选项，也都不一定会被完整的填写。

第二点就是数据稀疏的问题。二手电商的商品大多都是无标品。相比于一手店上电商来说，数据较为稀疏。

第三点是具备一些二手属性。二手店电商的商品都有很多二手属性。比如说成色、外观、屏幕划痕、是否换屏、是否翻新等等。

最后是价格差异。商品进行折旧以后，他们的价格会有一些差异。二手商品的价格是具备很强区分度的特征。



3.2 二手电商知识图谱构建

先构建商品的知识图谱。商品的知识图谱是类似树的形态。树由一级一级的节点组成，最后的叶子节点是商品实体，它的下面是一些商品的属性。

遵循业务需求循序渐进。在制作知识图谱的过程中，是边做边用的过程，而不是花费了很长的时间来做的很完整后才去使用。我们是根据具体的需求将知识图谱拆成几个步骤，然后进行持续的输出。

那么怎么拆分？根据之前提到的树的形态的知识图谱，首先要做的是先描点。先把图中的节点标好，然后再去挖掘属性中一些 K-V 信息，得到一些零散的点边关系，接着再把这些零散的点和边的关系串起来形成一张图，变成知识库。最终，再把商品挂上去。

>>> 二手电商知识图谱构建

13

- 先构建商品的知识图谱
- 商品知识图谱是类似树的形态
- 遵循业务需求循序渐进



首先，是 term 层面的一些应用。提取物品词，完成本体构建。然后，K-V 层面就是连接点和边。提取 tag 词，完成属性抽取。接着，在图的层面。tag 词树结构化，完成知识库构建。最后，商品粒度。将商品挂靠上去，完成实体抽取。

3.3 商品理解——物品词

首先从商品中提取出它的物品词，然后根据用户的行为数据得出用户偏好物品词，接着根据这个用户偏好物品词进行召回或是排序特征。

那么具体的实现方案：

先是物品词库的构建，不断地挖掘当前都有哪些东西，以及以后还打算做哪些东西。这部分的数据大部分是从我们自有的结构化数据那里拿到的，也有一部分是从外部爬去得到的，还有是从命名实体识别得到的。

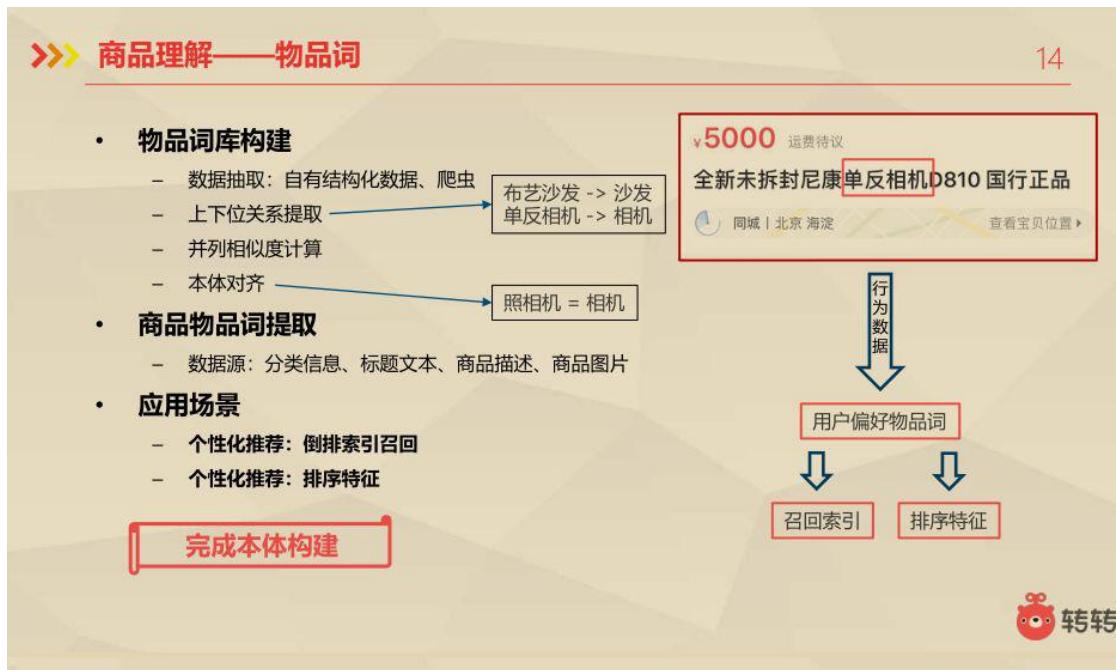
接着是上下位关系提取，沙发是个实体，布艺沙发也是个实体。布艺沙发也是沙发的一种，它们是一个上下位的关系。

然后是并列相似度计算。像布艺沙发和皮质沙发的相似度是比较高的，而沙发和相机的相似度就比较低。还有是文本对齐。类似于同义词，比如，相机和照相机其实是指代的同一个东西。

当以上完成以后，就构架出了一个物品词库。接着就是商品层面，商品物品词提取，使用到的数据源有：分类信息、标题文本、商品描述、商品图片。

应用场景主要就是：

- 个性化推荐：倒排索引召回
- 个性化推荐：排序特征



3.4 商品理解——tag 词

后面做了一个商品理解的 Tag 词，这是物品词的演进，这是服务的升级，刚才我们提到的是用户感兴趣的东西，但是人往往不会局限于对这个东西感兴趣，还有可能对这类物品有很多的要求。所以需要从属性的角度去挖掘用户的兴趣，比如右下角的例子。对该商品提取出更多的属性。那这个套路和刚才的物品词比较相似。这里需要注意的是，一手化的数据可以从自有结构的数据，爬虫，文本抽取中可以拿到，但是二手数据只能从文本挖掘中抽取。还有属性对齐。还有商品 Tag 词的提取，他的数据源来源于结构化数据，标题文本，商品描述，商品图片等。应用场景和物品词一致。最后就完成了属性抽取。



3.5 Tag 词树结构化

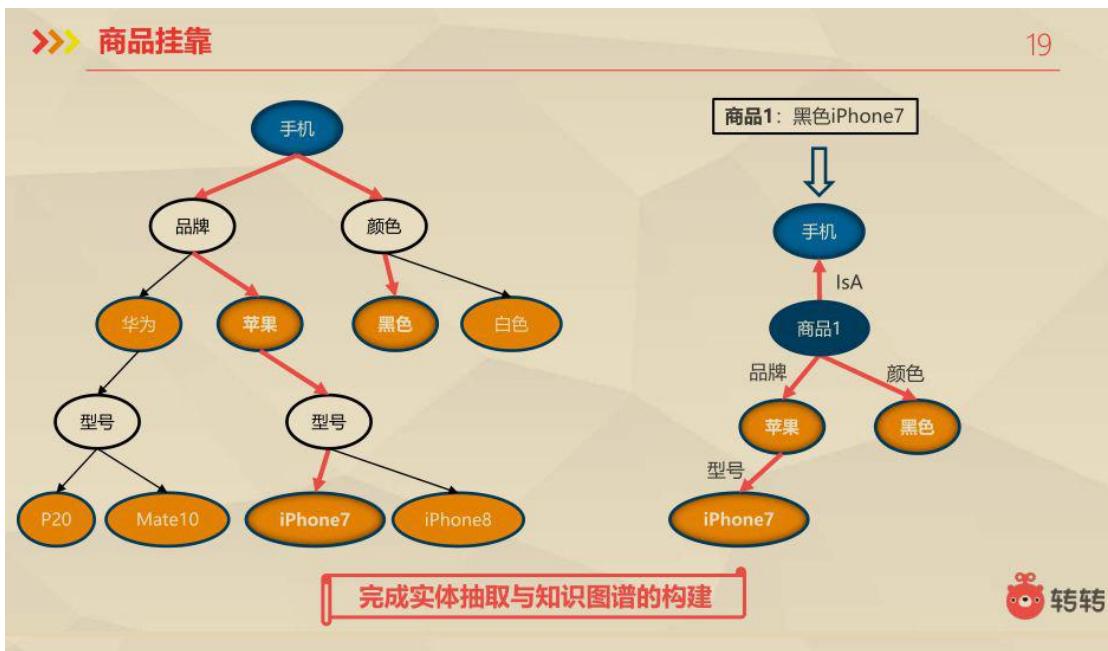
上面做完之后，我们发现提取出的 key-value 属性，都是各自离散存在的。然后会出现数据质量的问题，所以把之前挖掘出的 term 给提取出来组成一个树，下面是例子。从这个树里面可以追溯到他的所有信息。

这样的做法还提供了 query 结构化，对 query 进行理解，他的应用场景有三部分，个性化推荐和智能搜索，这一块截止，做完了商品库的知识库的构建。后面就是商品挂靠。

3.6 商品挂靠

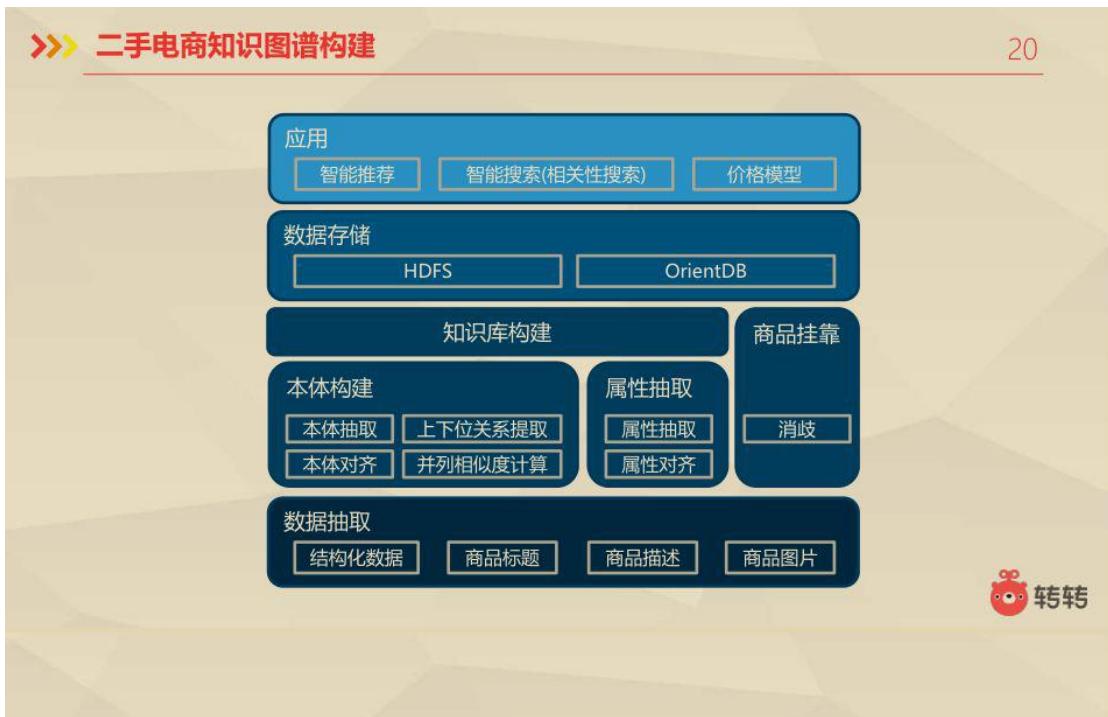
商品挂靠指利用分类信息、商品标题、商品描述、商品图片等数据，对本体库（Tag 词树型结构）中的节点进行匹配和生成商品知识路径。同时消歧有可能一个商品会匹配到本体库中的多个本体（物品词）和对属性节点赋予权值，选取匹配权重最高的本体。

这还是刚才的例子，商品挂靠之后生成一个实体（右侧）这一块做完之后完成实体的抽取与知识图谱的构建。目前我们有一些关于知识推理论和知识图谱的应用，优先级并不是这样的，目前还没有发力去做。



3.7 二手电商知识图谱构建

根据场景去介绍就可以构建出下面的架构，首先是数据抽取，在进行本体构建和属性抽取，在进行知识库的构建，最后完成商品的挂靠，把这些数据存储在 HDFS 或者 OrientDB 中，就可以进行智能推荐和智能搜索以及价格模型的构建。这里有一个消歧的概念，他主要是做根据树的权重的加和 权重较高的路径他的置信度就越高。消除一些无效的路径和属性。



四、在价格模型中的应用

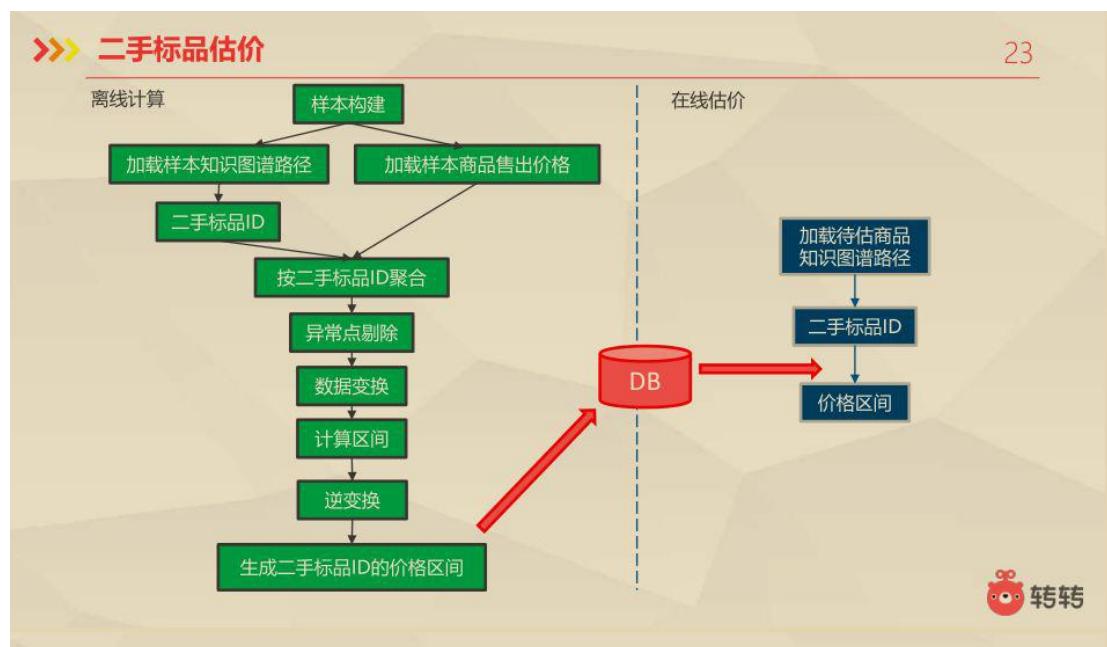
它的应用很多，这里说一下在价格模型中的应用，为什么做这件事情，对于二手商品的来说，很难去定一个合理的价格，所以我们这边希望提供一个定价的能力。

4.1 二手标品化

首先需要二手标品化，先做知识谱图商品挂靠，然后去筛选出价格敏感的二手属性。举个例子，我的二手手机屏幕碎了，这是很影响定价的一个因素。但是另外一个手机仅仅是划痕，这个属性对二手手机的定价不是明显的。所以需要去筛选一些对价格影响的属性。在同本体、同一手属性值和同价格敏感二手属性值下归纳二手标品，把这个 ID 作为这个实体新的属性打到知识图谱图谱上。我们假设这个标品的商品价格是同分布。针对这个假设，我么做了统计方法做估计价格区间和生成二手标品 ID 到价格区间的映射。最后得到的结果是可以支撑这个假设的。

4.2 二手标品估价

这块就是整个流程，前面要进行样本构建，然后在加载样本知识图谱和样本商品售出价格，在开始离线计算二手标品商品 ID 的价格区间。由于我们也没有二手商品的真正的价格，所以这里需要另外一个假设，我们认为大部分成交的二手商品的成交价是合理的，因为这是买家和卖家讨价还价之后的结果，并且基本上满足了双方的心理预期。所以我们收集已成交商品的价格，在按照二手标品 ID 聚合，对异常点删除，在进行数据变化。计算价格区间。最后生成二手标品 ID 的价格区间放到数据库中，在线估价的时候，首先加载待估商品知识谱图路径，然后定位二手标品 ID，最后确定价格区间。



4.3 非二手标品商品估价

上面仅仅说了二手标品 ID 的估价，这里还有非二手标品商品估计。手机很好说，但是衣服的话，从一手状态就不太好标品化，这有一套另外的解决方案，首先还是基于知识图谱制作，查找图谱中最近的 TopN 个出售商品，在聚合出售的价格，删除异常点，进行数据变化，计算价格区间，最后进行逆变换，生成商品价格区间。



4.4 数据变换

对于价格来说，他的分布有明显偏态的，但是区间估计需要分布是无偏的，为了能更精准地通过控制置信度来调整区间大小，最好无偏正态化。类似于左下角的分布，拿对数变换或者平方根变换就可以变换成近似正态分布，但实际数据的情况会复杂多样一些，为了能很好得无偏正态化，我们采用 Box-Cox 变换。对数变换和平方根变换是其特例。

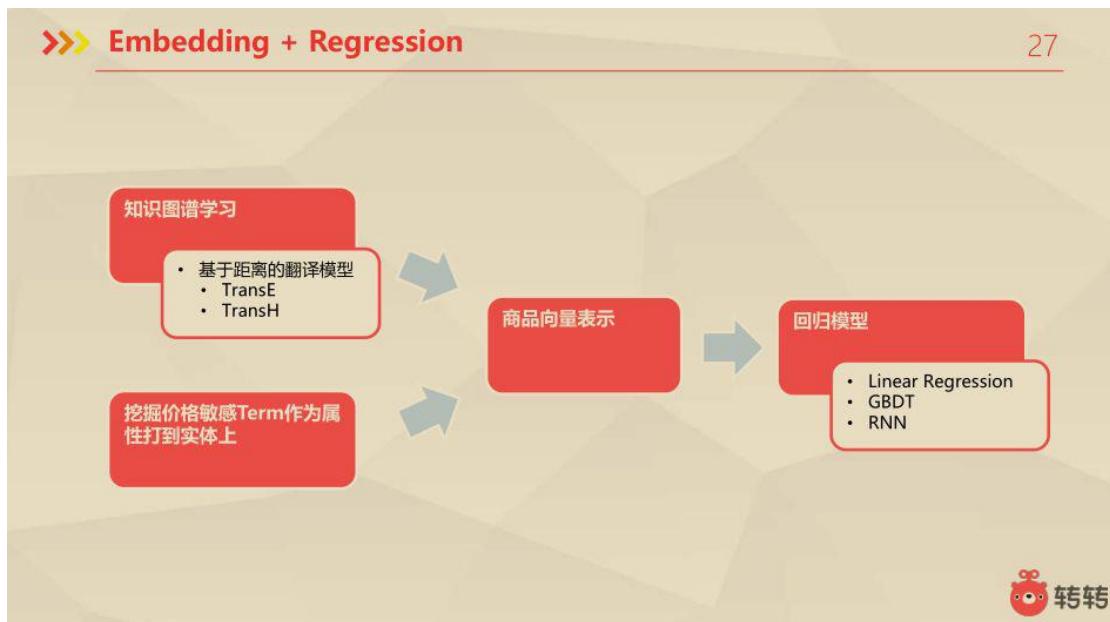
- **无偏正态化**
 - 价格的分布往往是具有明显偏态的
 - 区间估计需要分布是无偏的，为了能更精准地通过控制置信度来调整区间大小，最好无偏正态化。
 - Box-Cox变换

4.5 区间划分

有了正态分布之后，我们可以做区间的划分，首先我们希望这个区间可以涵盖大多数的商品，可以求均值，标准差，根据不同业务的需要，计算出价格区间，然后将计算出来的区间的上下限，做 Box-Cox 逆变换。这样才是真正的价格区间。

4.6 Embedding + Regression

刚才所说的是基于统计的方法，后面还有另外一种能够做法，基于回归的方法。先进行知识谱图的学习，挖掘出价格敏感的 term 作为属性打到实体上，在把商品用向量表示，做回归模型。然后可以用回归的方式去预测出商品的基本的定价。



作者介绍 :

张青楠，算法架构师，转转算法部基础模型团队负责人。主导了整套电商基础模型体系的建立。曾就职于当当推荐部，任资深推荐算法工程师。

内推信息 :

如下岗位 (50-100W 年薪) : 转转-算法工程研究员 (承担转转 搜索、风控 和 流量增长 3 个团队 的算法和策略开发 , 直接为各种指标负责) 转转-自然语言处理研究员(整体负责自然语言处理 Topic 的团队)转转-机器学习研究员(承担转转机器学习平台的开发 , 支持整个公司的算法业务)对转转感兴趣的同学欢迎投递 , 邮箱 :

zhangqingnan@zhuanzhuan.com

阿里神马智能对话问答

作者：亚楠 整理：Hoh

信息泛滥的互联网

手机发展到今天，上面的东西越来越多，不仅仅手机，以后我们还有物联网。以后我们的物品都是智能的，信息都是交互的，服务都是在线的。我们处在这么多物品，信息，服务之中，我们与他们的怎样的一种交互方式更加合适，更加自然，更加方便，这是做语音交互的重要的原因。不可否认，人类作为地球上最智能的生物，可能在一些方面还不如一些其他动物。但同时我们有更好的语言交互能力，这也是我们最高效的处理方式。必然会想到我们通过技术的手段处理一系列的问题，而这其中最难的其实是自然语言处理。

天猫精灵上通用的问答，技能，服务都是由神马搜索提供的。因为这需要全网的一个不同类别信息的支持，不同类别信息的一个服务。神马是做搜索引擎的，所以会有一些数据和技术的一些积累。开发的语智能助手是类似于度秘，Siri 的语音助手，目前并没有去刻意推广，只是作为一个实验 demo，后续的话我们会在 UC 上或者是阿里其他的一些产品上上线类似的一些功能。

对话问答系统

我们主要做的是一个对话的平台，它既要支持集团内部很多不同的业务，同时也会做自己的一些产品，所以我们的平台并非是一个简单的对话机器人。如果只是简单的对话机器人，一个人用大概一周的时间也能做出一个，但是这也需要一些知识和技术能力。另外我们知道音箱上，手机上对很多的问题的处理是不一样的。假如现在新拍了一个电视剧《琅琊榜》，在手机和电视上，你可能会是想看电视剧，而如果是在音箱上，你可能是想听一首里面的歌。对话需要考虑不同的场景，不同的人，以及不同领域内的知识，所以想做一个好的对话平台对技术的要求是比较高的。对话平台之下的核心其实是一套对话系统（SDS），对话系统的下面和通常的分类一样，有任务型的对话，比如今天天气怎么样？上海的天气怎么样？还有就是知识型的问答。其实人的对话并不像使用搜索引擎，人们在使用搜索引擎的时候大多是带着任务来的。而实际中我们每天说的这么多话当中只有极少的部分是问题，在日常中人们更多的是交流，而并非像审讯犯人那样。所以聊天对于对话平台也是很重要的，也是前面很多重要能力的一个联合技术。但是现在大多数的系统并没有一个很好的融合。包括现在主流的一些系统，在对话系统这里可能只是做了一个分类。这样做可能就会导致人在做一些穿插的任务的时候，在执行的时候会遇到一些困难。所以我们的对话系统是类似一个大脑的形式，下面会有一些具体的执行形式。

1、丰富的数据内容

做人工智能光有技术是不行的，还需要有大量的数据。在我们的平台上做人工智能还是有一些好处的。比如我们有大量的数据，因为我们有自己的搜索引擎，所以会有网上比较全的网页数据。在我们的搜索引擎上也有好几百个门类的不同行业领域的数据。我们也是很多重要厂商的流量入口，我们跟这些厂商都是很好的合作关系。还有就是我们有自己的知识图谱，我们知识图谱的建立已经有三年的时间了，在国内也是有着 top2 的水平；再就是我们有自己的用户画像，阿里的用户画像可能是国内最全面的用户画像了；还有大鱼号等等诸多合作的媒体资源；还有不可或缺的就是用户的日志，我们有着这么多的数据，同时我们也有一个强大的运营和生产数据的能力。也可以针对我们的智能所需生产各种各样的数据，这就是我们在这方面的优势。

2、对话系统 SDS

关于对话系统，我们都应该知道里面最重要的三个部分就是 NLU、DM、NLG。这是一个非常传统的架构，这个架构已经存在了几十年了。NLU 是指对话理解，NLG 负责将概念转化为自然语言，DM 相当于大脑，CM 是上下文的管理。

2.1 NLU

NLU 其实就是把一个非结构的自然语言转换为结构化的可执行的函数。如果是人来处理这个信息就很简单了，但是交给计算机去处理就会涉及各种编码、字符等问题。

首先我们是需要做一些预处理，这里大部分是在搜索引擎中完成的，然后进行一些口语化处理，过滤掉一些废话的信息。对于 NLU 来说其实就是要识别问题的领域以及意图，这其实是一个分类的问题。再有就是识别它的参数，即识别它的 slot。关键点是我们怎么能够把这些参数给对应上，目前有各种各样的方法，有神经网络的方法，也有非常复杂的方法。但考虑到它的一个性能效率等方面的原因，在业内大家其实还是使用的是模板。使用模板的好处：第一，准确率很高；第二，速度很快。但是光有模板不行，除了模板以外，还需要一些细化的东西，比如匹配，检索，模糊匹配等一些的处理。有的时候可能进行完这些处理之后仍然不行，这就需要一些更细化的模型，涉及机器学习的模型。

2.2 模板模糊匹配

通过深度学习分类的方法，将这些模板进行优化

2.3 联合模型

联合模型的意图就是既能识别它的槽位（slot）也能识别它的意图（intent）。为什么要用联合模型呢？先用一些简单的方法，如果仍然有比较模糊的需要进一步处理的，再使用联合

模型。这其中既有 softmax 也有加了 attention 的 CRF , RNN 的效果会好一点。在实际的应用当中会根据性能和要求做一个调整。

3、DM 对话管理

大多是人工配的流程图 , 好一点的是加个有限状态自动机。就像买票一样 , 就知道了起始点 , 终点的信息。这些简单流程都会面临一个问题 , 很多领域交叉的时候变得很复杂也难以维护 , 通用性会很差。通用性是一个必须的要求 , 我们怎么才能做到通用性 , 我们用到了一个 blanstek 方法 , 上面会有一个本体的知识库 , 有点像知识图谱 , 准确的来说知识图谱是本体的一种相对静态的一种支持。本体中也会有一些逻辑的关系。

3.1 Intent Slot 联合模型

这是一个导航的 DM , 它首先会有很多领域 , 比如是开车打电话 , 开车听音乐的。在领域的下面又会有一些功能 , 类似机构或者公司的架构。其中的 root 就是老大 , 什么都能做。我们想做什么事 , 在这棵树中都是可以找到的 , 另外就是像这种总分的结构也比较容易构建。它的引擎部分是栈的结构 , 我们人类聊天的话其实就是栈的结构 , 我们最关心的话往往是上一句话 , 可能是最近的一句话。这其中还有一个焦点词 , 比如人在做一件事的过程中可能会切换到另一件事 , 然后回来以后希望能够继续之前的行为。焦点词就是为了解决这种情况 , 中间接收到的任务会插入栈 , 然后结束以后再平滑的过渡回去。

在结合上面做的层次关系 , 刚开始在导航里 , 后来打电话 , 他会插到中间 , 从上一轮的信息查找 , 做到信息的平滑过渡。这种方式做对话 , 会解决我们在做对话中常见的问题。另外我们做的通用的引擎 , 所以需要做通用的工作 , 比如说 , 指代、省略。我们人类对话是多种方式的。我们每一个人说话不可能把所有从出生到现在所有的信息表述出来 , 前面的已经说过了 , 别人大脑里存储的有。对于对话系统来说 , 也是一样的 , 他们存储的知识树之间的相应的关系 , 背后需要知识图谱的配合 , 也需要本体树的配合 , 才可以理解人说的话。

还有就是错误的处理 , 对于语音来说 , 当然也包括打字等输入的方法 , 也会有输出错误的情况 , 大家可以看到许多纠错的产品 , 所以错误怎么处理 , 错误往往是与领域相关的 , 在一个引擎里面做出通用的错误处理方法。

3.2 DM 实现

大家认为前面就是一套规则 , 现在很火的是增强学习 , 深度学习 , RNN 对话系统之类的。他们之间有什么关系 , 这里阐述一下。在学术上 , DM 分为 DST 和 policy 两大部分 , DST 是对话状态的识别 , 类似于 NLU , 但是现在加入了机器学习的方法 (目前相对于传统的基于词典等方法没有明显的优势) , policy 就是根据现在的对话状态生成策略 , 基于规则的

方法相对比较死板，虽然可以加一些随机的路径，而 Policy 可以根据一些策略学习，比如增强学习的方法。

3.3 增强学习用于对话策略

刚才我们说到有限状态自动机，它可以自动优化策略，这就是马尔科夫（MDP）；如果有一些不确定性的话，就是贝叶斯网络，进行推理；由于对话不是显性的东西，如何获取这句话有没有错误、背后的真正意思是什么和处理过后的数据？所以 POMDP 应用于不确定性和自动策略优化的情况。目前又推出了一个基于深度学习的增强学习的方法 DRL。比如阿尔法狗，用深度学习的方法将不同的状态建模，应用不同的 Policy。

3.4 增强学习面临的挑战

我们拿对话与阿尔法狗进行对比，阿尔法狗在 19×19 的棋盘做出那么多的决策，对于对话而言，他的状态是无限的，还有就是评判，围棋的结果很明确只有输和赢，而对于对话而言，它的结果是无限的，很难去测量。另外难以收集有效训练数据：标注成本高、数据需求大。

3.5 利用模拟器收集数据 M2M

近些年出现了一些解决方案，比如谷歌今年的 I/O 大会，其中有一款可以代替人去打电话的产品，主要的方法是机器人与机器人聊天来收集数据，但问题在于不是真实的数据，所以加了一些真实的对话。两者结合会产生大量好的数据，为后面的机器学习工作奠定了基础。其中有一部分是任务型对话，这种类型也比较好评判结果。还有他的参数是有限的，也比较容易让机器采集数据。所以他的优点：应用于任务型对话，有明确目标，可枚举的 slot 类型和取值和技术与语义良好结合，降低了数据生产成本，并提升了效率。不足：不能适用于所有类型对话，需要各个领域分别构建，且无法应对闲聊以及模拟器构建需要很多设计和参与工作，收集的数据缺乏新颖。

3.6 NLG (自然语言生成) 与实现方法

自然语言生成包含内容规划、句子规划和表层生成三部分，其中内容规划包括内容确定和框架规划；句子规划包括关键词选取、内容聚合和指代生成；表层生成包括结构实现和语言生成；类似于高考作文，先构造一个大纲，在写内容，最后根据内容添加修饰。

NLG 实现方法在学术界也是热点讨论的，在工业界主要还是基于模板，因为模板有便于控制，准确率高，容易实现这些优点，当然也有缺点，比如：生硬死板，扩展性差，编辑成本高。而机器学习在一些地方还是有应用的，但不是很成熟，这需要一个过程。当然机器学习

也有它的优点变化多样，扩展性强，编辑成本低；他的缺点是不易调试，错误率较高，技术复杂。

3.7 TaskBot (不同的服务)

第一个是任务型技能，这些都应用了 NLP 和 NLU。只不过在一些具体的任务中，会做一些细化。我们看一下智能问答，他会有四种不同的服务，一般在学术的讨论中，会把智能问答分为基于知识图谱的问答和基于非知识图谱的问答。但是这里分为四种，第一种精准问答是基于知识图谱的问答，重点在于 query 结构化，准确率高但覆盖率低。第二种是高质量问答，是运营人员编辑过的，难点在于优质 query 挖掘和同语义 query 扩展，第三种是通用问答，难点在于答案质量判断、同语义问题匹配、长答案自动摘要，第四种是百科问答，较简单，重点在于意图识别和口语化处理。其中第二种和第三种是非结构化数据问答。

(1) 优质 QA 挖掘生产

我们这里说一下生产能力，我们的天猫精灵的日志过来，我们从里面找到一些比较好的问题，好的答案挖掘出来，然后经过人工的审核，或者经过运营生产的编辑。在实际的工作过程中会有很多的问题要处理，比如说什么是好的问题，如何定义？互联网上的问题大多都是大数据量的，一个母婴的问题，在互联网上就可以抓取几亿条数据，这些问题重复的很多，去重之后也会很多。我们需要知道哪些问题是问得比较多的，这个时候搜索引擎的作用就显得很重要了。另外一个大问题是去重，在大数据量的情况下，如何快速的去重，也是有难度的，这也是一个传统的问题，有各种各样的方法。这都需要搜索引擎来帮助我们建立重复的事例。还有就是敏感问题，不是所有的问题都是适合回答的，这些问题如何识别，也是需要很大的资源和积累才可以做的。所以整体的流程是天猫精灵日志过去，然后问题分类分为 CNN 闲聊类型和疑问词分类模型，同时划分高质量问答 query、敏感问答 query 和其他问答 query。然后在经过 UGC 生产、外包生产和问答 url 挖掘最终确定高质量问答、众包审核和挖掘审核。最后挖掘出来的如何确定是好答案，需要通过答案交叉验证（答案与答案之间、答案与权威数据之间以及答案与知识图谱之间），答案质量模型和 Qt 相似度模型。

(2) 问答匹配模型与答案质量模型

答案质量判断根据用户行为数据包括用户的点赞、点击、回答者的等级等信息，另外一方面也会深度学习纯文本的特征，可以把一些明显的答非所问的问题给去掉。我们这里的模型是比较简单的，因为数据量很大，我们也尝试过复杂的模型但是综合性能的话，简单的模型相对优势大一些。

(3) 基于 RNN 的答案自动摘要

另外在智能音响上或者机器人上，我们都不希望答案特别长，如何把答案自动的摘要压缩，这也是一个问题。我们用 RNN+attention 的方式来处理这个句子选取的问题。先选取 RNN 的 query，然后在做一个 attention。之后在算他的分数，选择出分数较高的。另外在召回方面也做了很多的工作。我们不希望一个回答省略到关键的词或者句子。

3.8 ChatBot

前面我们主要说了问答。这里我们出要说一下聊天，前面所说的问答和任务型对话只是很小的一部分。因为聊天没有边界所以相对较难一些，经过我们的探索，聊天应该分为这四个阶段，相关包括对话理解和相关匹配；有趣：多样性、个性化和场景化；可持续包括用户画像、情感分析、推荐引导和对话策略；可信赖包括高情商、察言观色、三观一致、趣味相投和体贴备至。我们这里说一下生成式模型，因为是开源的，加上开源的对话的语料，所以可以很容易把它跑起来。但是实际上效果非常不好，最大的问题是他们的对话没有什么价值，可以通过一些手段来优化，比如加一些 attention，来优化他的回答。在 decoder 过程中加一些重要的词，让他的回答更加按照我们的意思来进行生成。但是没有一种方法彻底来解决这个问题，现在通过的方法只有两种，第一个是通过 seq2seq 加 attention 的方式来优化，第二种是优化他的 loss function。不断的优化目标函数。提升回答的多样性。

现在很多是基于 CAVE 的方法来提升多样性和鲁棒性。当然也有很多种方法，这只是其中一种。多了一个分布，他在训练的时候会生成一个类分布，在最终使用的过程中，会从中采样出来。根据采样在生成结果。它的结果也是不一定的，因为每次采样是不用的，当然我们可以控制，比如上下文加起来，控制他的输出。从我们的实验来看，加进来之后，他的相关性和多样性会有提升，这些方法在业界也不是主流的方法，这一块是比较有前途的方向。

3.9 聊天系统

在技术上，分为三部分 CHAT AGG、排序触发和语料模型。CHAT AGG 包含了综合出发排序、对话和问答三部分，排序触发包括了基础文本，VSM BM25、翻译模型、DL 相关性、生成式模型、置信度识别 LTR 综合排序。语料模型包括了模版剧本、对话对索引、机器人运营、垂直领域知识本体和多轮个性化 session 用户画像等。

4、总结与规划

我们这边会做很多种模型，比如置信度，相关性等。然后再利用我们这边的资源比如用户画像、知识图谱、对话的剧本、大规模的深度学习的平台。这些都可以帮助我们解决我们实际遇到的问题。我们最终的目标是构造一个全面通用的平台。目前除了支持集团各项服务，神马计划建设完整的聊天系统，兼具问答实用性和聊天趣味性，形成完整的能答会聊的对话系

统。目前业内还没有真正实现这一目标的系统，我们将借助阿里和大文娱的流量和内容助力，形成新的突破。

内推信息：



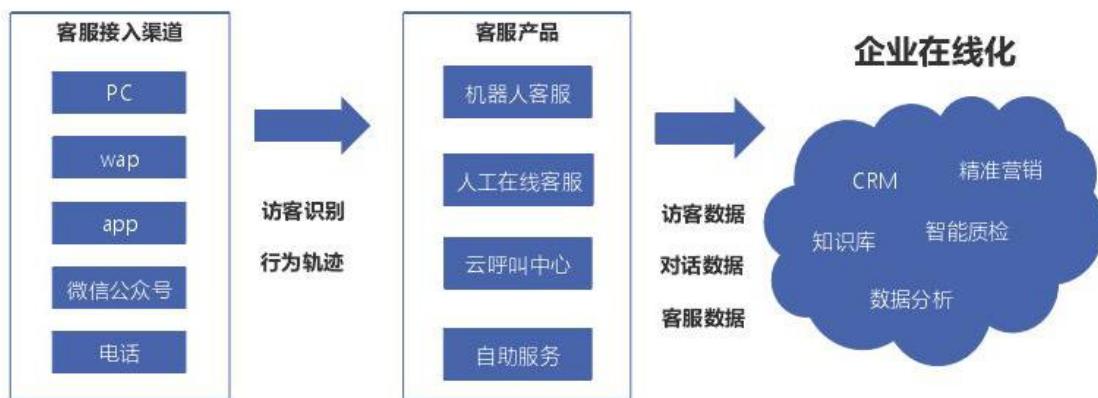
扫一扫二维码，加入亚楠老师团队。

对话机器人在瓜子的实践

作者：王文斌 整理：Hoh

今天主要分享以下几个方面，首先介绍下什么是对话机器人，然后讲一下技术选型的过程，设计了怎样的算法架构和系统架构，最后分享下线上的效果以及在瓜子中面临的一些挑战。

背景



目前对话机器人很火，是有多方面原因的：第一，图灵在定义智能时就将对话机器人作为人工智能的一个标志；第二，深度学习技术越来越成熟，对话机器人在工业界已经达到一定水平；第三，对话机器人由于有智能客服的积累，有很多公司在做这方面的东西。上面是一个智能客服设计图，左边是接入渠道，登录进来，会提供一些客服产品，如机器人客服、人工在线客服、云呼叫中心，以及用户依据产品做一些自助服务。聊天过程中用户会将其数据留下来（反馈数据、对话数据、人工客服数据），利用这些数据就可以做分析，如客服数据可以做质检，用户数据可以做营销工作，与 CRM 接入打通。

背景

为什么要有对话机器人？



缩减人力和培训成本	沟通可追踪可优化
7*24小时在线	差异化服务
服务质量可控	精细化运营
提升服务效率	推动企业在线化

接下来讲一下为什么要对话机器人，开始瓜子目标就是提高效率，用机器替代人，达到缩减人力和培训成本、7*24 小时在线服务、质量可控的目标。在发展的过程中概念慢慢升华到一个在线化的概念，就是数字化、数据化和智能化。数字化就是将用户和企业交互的数据都记录下来，将数据结构化，做成算法可用的数据叫数据化，有了数据化就可以用建模等一系列智能化手段做一些智能化提升。在线化做后可以做到整个沟通可追踪、提供可优化、差异化的服务以及精细化运营，最终推动企业在线化。

在线机器人是在线聊天的一部分，既是整个服务闭环的入口也是出口。用户可以在聊天中表达和解决相应的诉求，而搜索、推荐更像是一个被动的过程，IM 是一个主动表达诉求的门户。

什么是对话机器人 - 分类

分类	角色
开放域：微软小冰、度秘	产品驱动、数据驱动
任务导向：订机票、询问天气	算法是关键 架构是主体

用户视图

- 客户视图： 对话内容、对话框、对话框外推荐信息
- 客服视图： 对话上下文、客户画像、背景信息、订单画像
- 管理者视图： 控制台、知识库

对话机器人的分类：开放式的有微软小冰、度秘；任务导向的有订机票、询问天气。从角色定位角度，如提供 IM 通道其实就是架构，只有有了骨架才能做相应地应用，算法在里面是一个关键的作用，后期其实更多的是偏产品化的东西。对话机器人技术是透明的，区别在于谁做的细节更完善。开发的角度就是完善三个视图，客户视图：对话内容、对话框、对话框外推荐信息；客服视图：对话上下文、客户画像、背景信息、订单画像，管理者视图：控制台、知识库。

对话机器人经典流程：语音唤醒，告诉你要干嘛，唤醒之后经过语音识别转化为文本，这时候可以做语义理解（其中可能需要知识库交互），将语义理解的结果通过对话管理引擎拿到用户对应的话术，将对应的话术转化为文本，最后转化为语音输出。

什么是对话机器人 - 核心概念

帮我订一张明天上午10点从北京到上海的机票

瓜子业务下的问题：

- 泉州车过户到厦门，会不会很麻烦

意图 (Intent) = “订机票”

槽位 (Slot) :

- 起飞时间 = 明天早上10点
- 起始地 = 北京
- 目的地 = 上海

说明下对话机器人的核心概念，如“帮我定一张明天上午10点从北京到上海的机票”，这句话的意图就是“订机票”。槽位就是如果要完全理解一句话并且能够返回结果信息还需

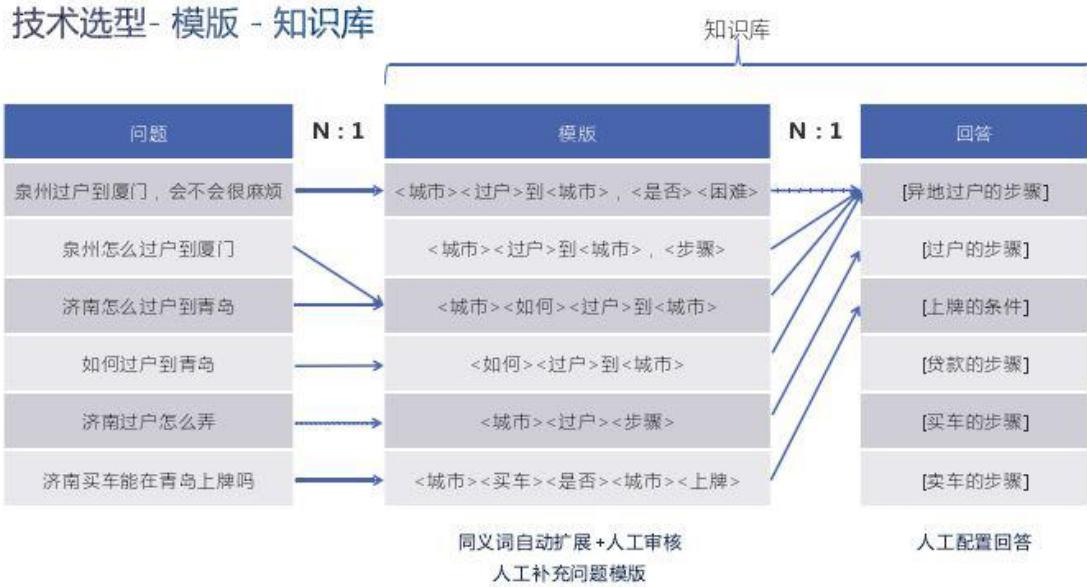
要什么属性，这句话槽位信息有：起飞时间 = 明天早上 10 点，起始地 = 北京，目的地 = 上海。

技术选型



接下来讲一下技术选型，这是对话机器人选用技术调研的过程。对话机器人开始是基于关键词，然后就是模板技术，目前很多公司还在使用，优点是质量可控、准确率高，其缺点就是泛化能力比较弱。随着功能不断迭代，模板很大程度依赖于人工，不能自主提升自己的泛化能力。然后有了基于搜索的对话机器人，有很强的业务适应能力，其缺点就是准确率低。最近几年深度学习火起来后，利用深度学习替换原来的模型进行意图识别，意图识别相对传统方法准确率提升很大，但是缺点就是对数据质量要求较高。

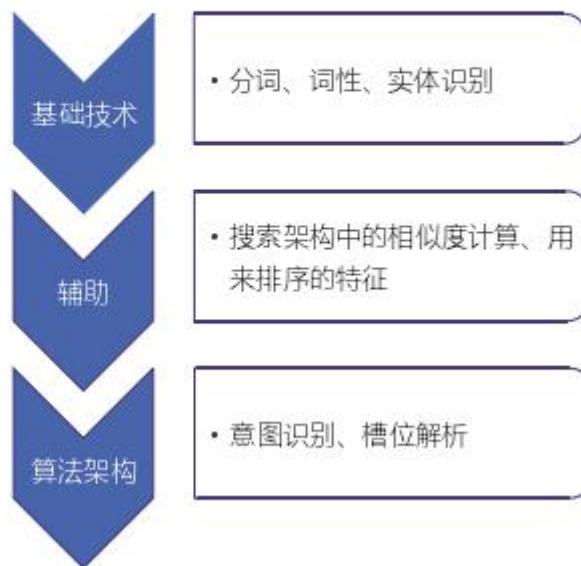
模板可以部分自动生成，如果上线也需要应用方自己审核与补充，话术也需要应用方自己去配置。搜索技术更多的是先用意图识别做一个路由器的功能，然后路由到一些小的 robot，每个 robot 做一类事情。深度学习与传统分类方法做的事情类似，也是在做多意图的意图分类，确定意图后会通过一对一或其他配置方式将其关联回答。



下面是一个模板算法，“泉州过户到厦门会不会很麻烦”，这个模板在前面没有出现过，就无法匹配，需要将模板提取出来，固化到知识库、模板库中。

对话机器人发展到后面越来越注重运营，有一个管理平台，就是知识库。固化知识，给运营提供管理入口。前面例子就是维护问题到模板以及模板到回答的映射关系，人工需要做很多审核以及一些校对的工作。而搜索方案，将 query 经过预处理打散成 terms，进入搜索系统，如果按照原始结果会得到一个排序“泉州到厦门过户问题，泉州到厦门远吗，泉州到厦门怎么坐车，泉州到福州过户问题，泉州到厦门过户问题”，最后得出结果与查询一致，将最相近的 query 回答返回。而解决排序不正确的方法就是需要海量数据。

技术选型- 深度学习 - 算法



接下来讲一下深度学习的算法架构，深度学习应用很多，以对话机器人而言，基础技术如分词、词性、实体识别都可以用深度学习，数据好的话会比传统方法好。还有搜索架构中的相似度计算、用来排序的一些特征也可以用到深度学习的方法。我们是从整个结构来看就是一个深度学习的架构，这也是学术界研究的热点。

深度学习知识库我们解决就是意图与答案一对一的关系，回答对话本身要求很严格，几乎是一个纯人工的过程，有很多人参与业务运营。如果是单轮就是一个多分类问题，更重要的是如何建立一种机制将问题积累过程与上线后模型的演进过程变得更加自动化、质量更可控。除了刚才它谈到的技术还有其他方法如生成模式，学术界较火，主要是应用于闲聊。我们最后选用深度学习模式，考虑的原因有以下几个方面，就是不再需要人去抽取大量的特征。语义理解的流程，包括快速识别、模型识别、搜索识别、相似问题，在这个流程中应用了很多技术。我们采取的是一个漏斗方式，开始是快速识别（需要实时解决），在快速识别弄一个白名单用关键词或模版匹配立刻纠正，原则是必须准确率要高。90%的问题是依据模型框架，准确率也在90%以上，有了前面两步，后面是在补充召回的过程，通过搜索系统借助文本相似度的匹配将一部分数据召回，尽量让用户更多的问题被识别。

算法架构 - 多轮

多轮是个工程问题，需要做好3件事：

1. 填槽
2. 场景管理
3. 可配置：
 - 多轮的逻辑是在知识库里配置的
 - DM是业务无关的，只需要按配置的解析结果执行即可

接下来介绍下多轮，我理解多轮是一个更偏工程的过程。里面更多的算法是在做槽位解析，需要做好三件事，第一个就是填槽，如果对话过程中槽位未补全，在下轮对话过程中引导用户补全槽位信息。再者就是场景管理，需要维护海量用户的聊天信息。第三点就是可配置，多轮最后面都是一个业务问题，开发一个可配置的界面，让运营自行配置其需要的对话。多轮的逻辑是在知识库里配置的，DM 是和业务无关的，只需要按配置的解析结果执行即可。按照上面设计还是会出现风险，常见的五个风险有：任何算法的选择都只是满足当前的需求，数据是历史数据，算法是当前反馈，业务演化过程不可知；模型互搏，各种模型都要去做 A/BTest 确定哪种好那种坏，之前更多的判断是从原理上判断；意图爆炸，目前知识库是基于意图回答一对一关系，业务相对收敛，但是未来发展速度可能导致意图不可收敛；主观标准的反复，很多过程都由人工参与，每个人评判标准不一；模型更新滞后于业务发展，技术发展较快。解决方案就是永远保持主动，提前应对。

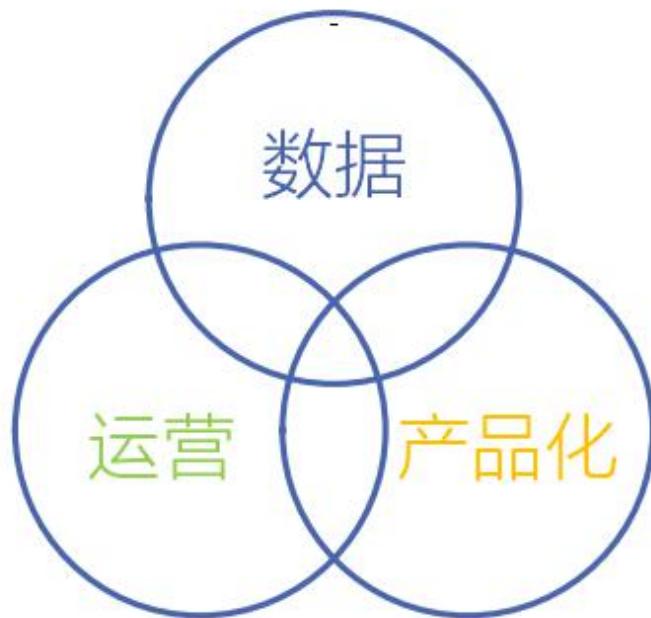
系统架构：前端有一个对话框和消息服务器，类似于 IM 基本架构，消息服务器会将消息路由到对话管理模块（中控）。用户聊天文本会在中控识别意图和槽位，通过意图在知识库中获取对应的话术。知识库有一个控制台，与外部交互的界面，对话管理也会访问后端云服务，比如通过 ip 地址获取其属于哪个城市，除此外还有语义理解、CRM 服务等。

线上效果



线上效果 , 左边是一个单轮对话能力 , 无论问如何贷款都能准确识别 , 右边是一个动态 API , 类似于知识图谱想要完成的工作。

在瓜子的挑战



在瓜子遇到的挑战 : 首先是数据 , 不管做什么都需要数据。运营 , 这方面主要是对话机器人自学习的能力 , 如何设置一些机制使运营能够满足当前业务效果 , 跟上业务发展速度。最后是产品化 , 如何将产品细节做得足够好。

举例：第一个就是数据来源，以一定规则构造数据，或利用非结构化数据通过迁移学习训练 embedding 向量，将向量作为意图识别的原始输入，或模型产生数据反哺模型，不断迭代。第二个就是话术的确认流程，编辑发起修改，业务反馈，编辑确认，审核，法务，上线，这是一个理想的模式。人与人之间的平衡：回答的标准，新增意图的标准，产品和算法的平衡：意图预判、suggest、相似问题、下一个问题，业务和技术的平衡：卡片消息，就是在线化，后台服务如何让用户利用起来。

不只是客服 - 意图预判



不只是客服 - 精准营销

我想买一辆三四万的车	太远了
打算买个马6	离得太远怎么办
问一下2016款奔驰	怎么看车的，异地的怎么办
五菱宏光我想买	车在外省，本人不想去
.....

用户主动表达诉求

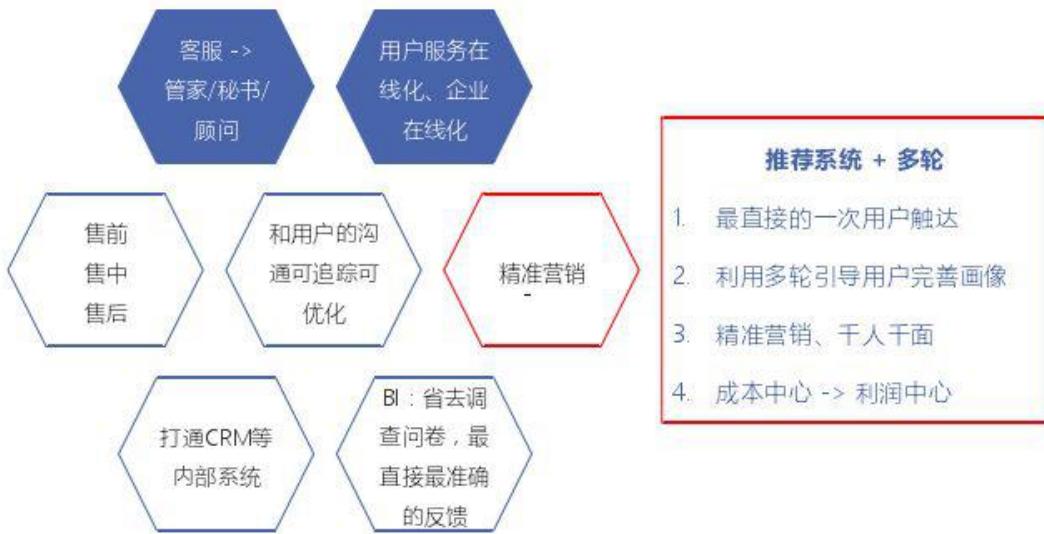
-- 对话中的推荐

识别用户潜在需求

-- 回答后的推荐

用户从不同的业务入口进来看到的问题列表是不同的，从不同业务阶段进来看到的问题列表也是不一样的。后续希望做到不仅根据业务状态还要基于历史数据做一些推荐。对话机器人可以做很多事情，如目前我们正在做的精准营销，通过多轮对话完善用户诉求，给出更加精准的推荐。

不只是客服



下面更多是一种理念，打通 CRM 等内部系统，可以利用数据做商业智能，覆盖售前、售中、售后所有场景，用户沟通可追踪可优化，精准营销，从客服转化为专家顾问，实现用户服务在线化和企业在线化，最终实现整个企业的智能化。

作者介绍：

王文斌，车好多 NLP 方向负责人。硕士毕业于北京大学，曾就职于美团、百度等公司，在编译器、浏览器、IM、大数据等复杂系统研发上有实践经验，并在搜索推荐、知识问答、数据挖掘、机器学习、NLP 等算法方向有丰富的积累。加入车好多后发起了智能 IM 项目，实现了对话机器人的成功落地。

团队介绍：

瓜子 NLP 团队，以 chatbot 等产品，增加人效，提高服务质量，让瓜子逐步加大服务线上化的比例。团队承载瓜子服务线上化的重任，是未来瓜子发展的重要基础能力之一。

内推信息：

在看 NLP 算法工程师、数据挖掘工程师机会的小伙伴，欢迎加入文斌老师的团队，内推邮箱：wangwenbin2@guazi.com.

五八同城智能客服系统“帮帮”技术揭秘

作者：詹坤林 整理：Hoh

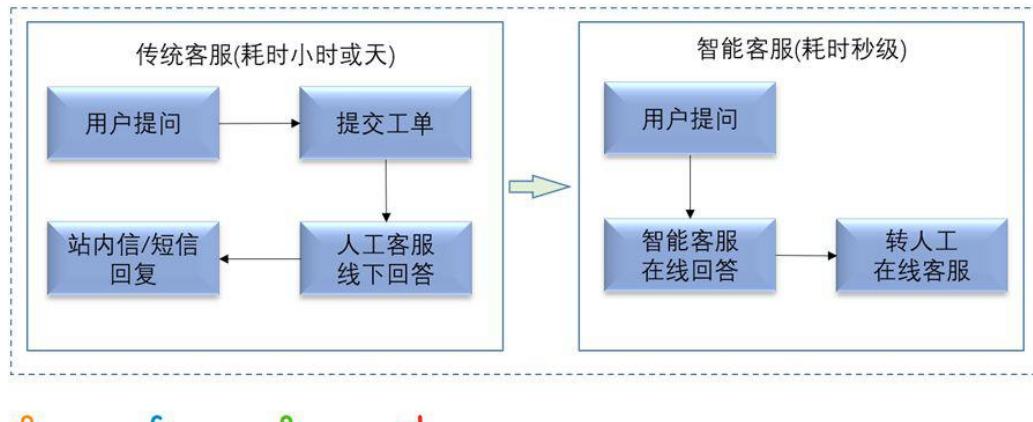
首先简单介绍一下 58 同城，58 同城是一个生活服务平台，平台连接着 B 端商户和 C 端用户，B 端商户在平台发布帖子信息，平台将这些帖子信息分发给 C 端用户供其浏览。在 58 同城 APP 或网站上，用户可以通过搜索和推荐的方式获取到帖子信息，例如用户可以通过搜索框搜索信息、进入列表页筛选信息、在猜你喜欢和相关推荐等推荐位浏览信息。58 同城提供租房、二手房、找工作、二手车、黄页等信息，这些业务分布于房产、招聘、二手车、黄页等不同业务部门，不同业务部门都有各自独立的客服团队，我们的目标是设计一套通用的智能客服平台来解决所有客服问题，以提高客服效率。

今天的分享将从以下几个方面展开：首先介绍智能客服的背景，然后介绍总体技术架构、算法和工程架构，最后做一下总结。主要想通过这次分享使大家了解到智能客服系统中的技术全貌，希望对大家有些启发。

背景

让 生 活 简 单 美 好 58

- 传统客服的局限：操作繁琐 & 问题解决周期长
- 智能客服的优势：提高人效 & 提升用户体验



传统客服工作模式包括客服网站和电话客服两种：

(1) 公司提供一个客服网站给用户，用户通过网站提交问题反馈，这些反馈信息会通过一个系统展示给客服人员，客服人员每天逐个解决这些问题，解决后通过站内信或者短信回复

用户。这种模式下，用户在客服网站上的操作往往较繁琐，并且问题解决流程周期长，可能会耗时数小时甚至数天，用户体验差。

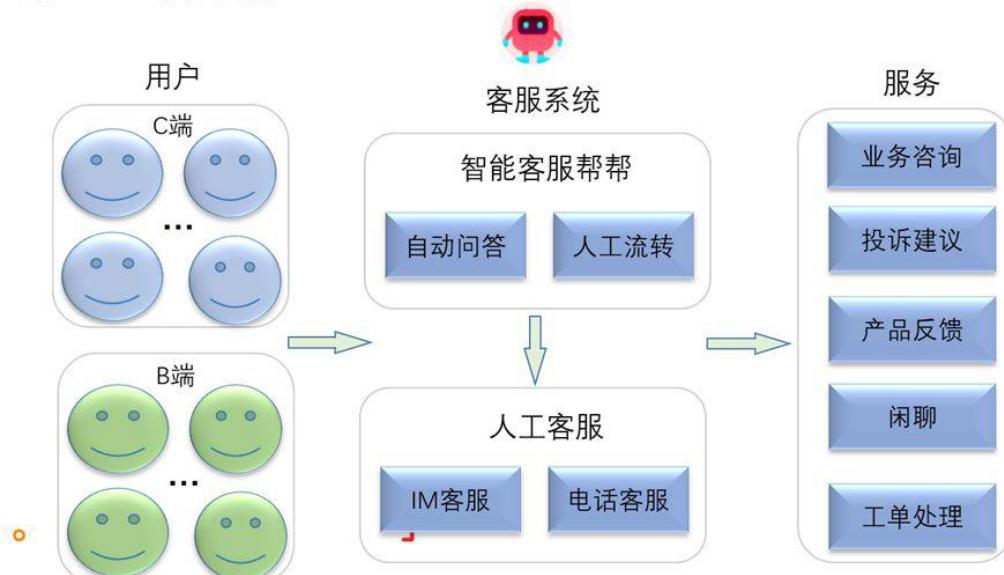
(2) 公司提供一个客服电话给用户，用户通过电话咨询客服。这种模式尽管直接，但是可能存在问题描述不清、沟通成本高的问题，例如客服在解决某个问题时往往需要用户提供额外信息，一通电话会持续较长时间。一般一个客服每天能完成 60-80 个电话的接线，服务效率较低，而且客服人力成本高。

大部分客服问题其实是高频重复问题，这些问题往往都有标准的答案，这可以利用机器去解决，可以构建一套智能问答系统去自动回答用户的提问，当用户对答案不满意时，他可以再寻求人工客服的帮助。这种机器自动问答和人工客服辅助的模式下，大部分客服问题通过机器解决了，只有少部分机器解决不了的复杂问题才会由人工客服来解决，这不仅提升了用户体验也提高了客服人员的人效。

背景

让生活简单美好 58

● 五八客服体系



58 同城旧有客服体系就是通过客服网站和客服电话来提供客服服务，我们需要重塑这种模式，构建一套新的客服体系。在新的客服体系下，用户所有的客服咨询首先都会经过智能客服系统“帮帮”，由“帮帮”来自动回答用户的问题，若用户对答案不满意，他可以转接人工客服。人工客服包括旧有的电话客服和新设计的 IM (即时通讯) 在线客服，IM 在线客服是指通过 IM 聊天的方式提供客服服务，用户可以和客服人员通过聊天窗口直接一对一进行沟通，智能客服和 IM 在线客服会无缝整合在同一个聊天窗口中。转接人工客服时我们会首

先转接到 IM 在线客服上 ,若用户仍不满意才会通过电话的方式解决问题。新的客服体系下 , 用户可以获取到业务咨询、投诉建议、产品反馈、闲聊以及工单处理等客服服务。

这种新的客服模式相比旧有模式的优点有 :

(1) 用户体验好。传统客服网站的方式用户获取答案周期长 , 这是因为客服人员需要手动解答客服网站上收集的每个用户问题 , 由于每日问题量大而且客服人员数量有限 , 大部分用户的问题不能即时得到解答。新的模式下用户可以通过 IM 聊天窗口咨询问题并即时获取答案 , 简单高效。

(3) 客服人效高。“帮帮”能够自动回答大部分问题 , 人工客服只需要利用 IM 在线客服聊天工具去解答少部分复杂问题 , 机器和人工处理问题的比例大约是 8:2。每个 IM 客服人员一天大约能处理 120-150 个用户的咨询 , 这远比电话客服每天处理 60-80 个用户的咨询要高 , 因此我们会尽量让用户咨询先流转至 IM 在线客服 , 只有最复杂的问题才会流转至电话客服。通过这种智能客服到 IM 在线客服再到电话客服的方式 , 我们可以利用有限的客服人员处理更多的用户咨询。

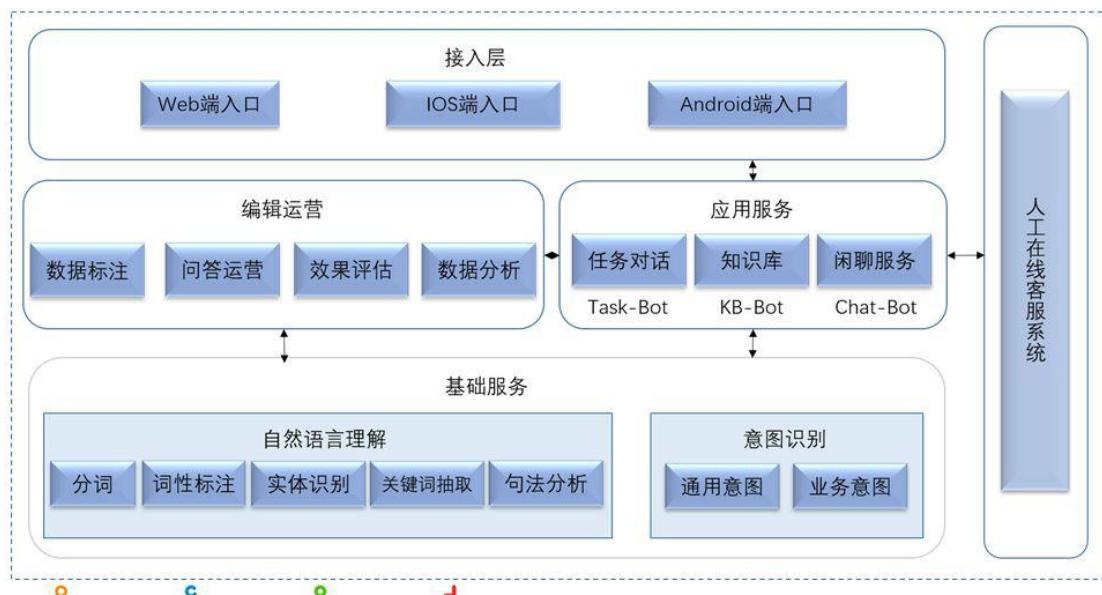


“帮帮”智能客服系统是一套基于深度学习和自然语言理解技术实现的自动问答对话机器人 , 产品界面如图所示 , 用户通过聊天窗口的形式和“帮帮”进行对话。对话机器人一般分为业务咨询类、任务类和闲聊类三种 , “帮帮”也支持这三种功能 : 最主要的是提供业务咨

询功能，帮助用户解决业务类问题；其次支持任务类型的回答，用户可以实现查询帖子被删除原因、注销账号等任务；此外，为丰富“帮帮”的功能，也支持闲聊功能，用户可以在聊天窗口与机器人寒暄闲聊。

整体架构

让生活简单美好 58



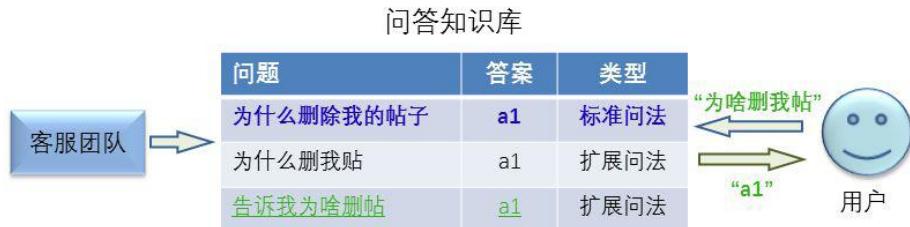
“帮帮”整体技术架构如图所示，包括基础服务层、应用服务层、编辑运营层、接入层以及在线客服系统。基础服务层提供对话系统的基础技术能力，系统需要对用户输入的一段语句进行理解，这里需要自然语言理解模块，对语句进行分词、词性标注、实体识别、关键词抽取和句法分析等；同时需要识别用户的意图，包括通用意图和业务意图，通用意图是指用户是来做业务咨询还是闲聊，业务意图是指若用户是做业务咨询，具体咨询什么业务，这里会使用文本分类的技术去识别用户意图。基础服务之上是应用服务层，这一层具体实现了KB-Bot 基于问答知识库的机器人、Task-Bot 任务对话型机器和 Chat-Bot 闲聊类型机器人，这是“帮帮”系统的三种核心能力。编辑运营层是指有一个编辑团队支撑着“帮帮”的算法策略迭代，主要完成数据标注、问答运营、数据分析和效果评估的工作，这些工作输出会作用到基础服务层和应用服务层。基于应用服务层，对外提供通用的接口服务以便于业务方接入，我们支持 Android、iOS 和 web 端的接入。此外，机器不是万能的，用户有很多复杂的问题仍需要人工解决，这里有一套在线客服系统提供了人工在线客服的能力，应用服务层会和这套在线客服系统做无缝对接。

核心功能-业务咨询服务 KB-Bot

让 生 活 简 单 美 好 58

- FAQ型问答

- 问答知识库构建
- 问题匹配



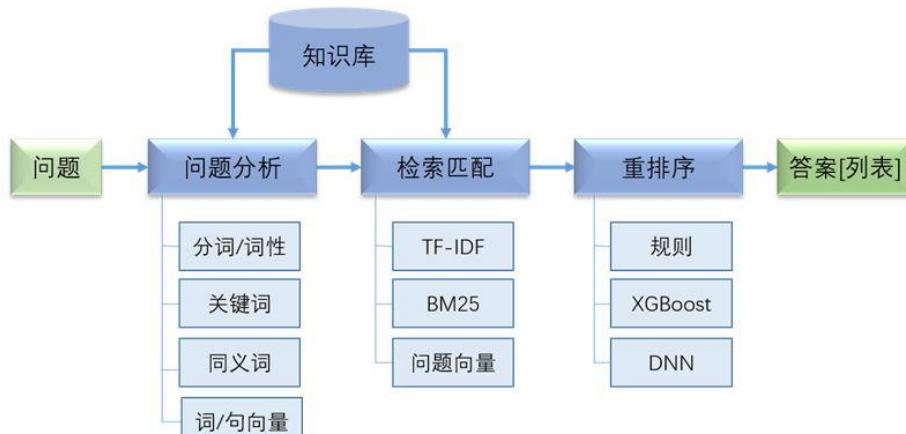
“帮帮”系统的核心是提供 KB-Bot、Task-Bot 和 Chat-Bot 三种能力，下面分别介绍下这里使用到的技术。KB-Bot 是指基于问答知识库的对话机器人，它主要实现了“帮帮”最重要的能力——提供业务咨询类服务。58 的用户使用帮帮主要是来进行业务咨询，例如询问账号为何被锁、帖子为何被删、如何购买帖子置顶服务等等。业务咨询类的回答需要基于问答知识库来实现，这里的问答知识库是一个包含众多问答对的数据集。我们将问题划分为标准问题和扩展问题，例如“为什么删除我的帖子”这个是一个标准问题，语句表达很标准，它会有一个标准答案，其近似的问法我们称之为扩展问题，例如“为什么删我贴”、“告诉我为啥删帖”等，这些都表达的是一个意思，这些问题同样对应的是相同的标准答案。有了问答知识库，用户来询问时就是一个问题匹配的过程了，只需要将用户输入的问题和知识库中的问题做匹配，得到意思最相近的那条问题，然后将对应的答案返回给用户，这就完成了一次问答操作。问答知识库的构建非常关键，这里会首先对客服团队历史积累的问题数据进行抽象，形成标准问题，然后结合算法和标注对标准问题做扩展，形成初始问答知识库，在系统上线后，对新产生的数据又会进行挖掘，不断扩充知识库。

核心功能-业务咨询服务 KB-Bot

让生活简单美好



● 检索式回答



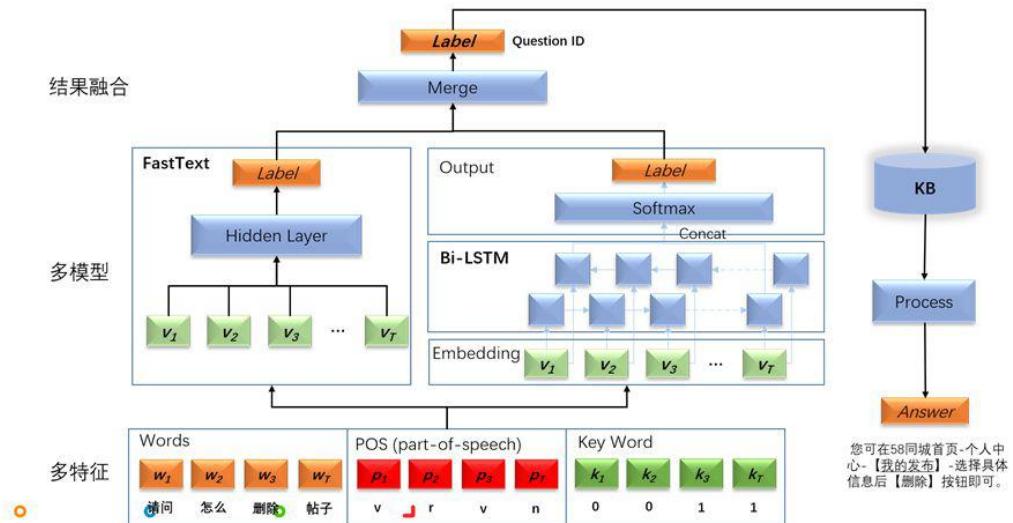
基于知识库的问答可以使用检索或者分类模型来实现。检索式回答的流程是：首先对用户的输入问题做处理，如分词、抽取关键词、同义词扩展、计算句子向量等；然后基于处理结果在知识库中做检索匹配，例如利用 BM25、TF-IDF 或者向量相似度等匹配出一个问题集合，这类似推荐系统中的召回过程；由于我们是一个问答系统，最终是直接返回给用户一个答案，因此需要从问题集合中挑出最相似的那个问题，这里会对问题集合做重排序，例如利用规则、机器学习或者深度学习模型做排序，每个问题会被打上一个分值，最终挑选出 top1，将这个问题对应的答案返回给用户，这就完成了一次对话流程。在实际应用中，我们还会设置阈值来保证回答的准确性，若最终每个问题的得分低于阈值，会将头部的几个问题以列表的形式返回给用户，最终用户可以选择他想问的问题，进而得到具体的答案。

核心功能-业务咨询服务 KB-Bot

让 生 活 简 单 美 好



● 分类模型：多模型融合

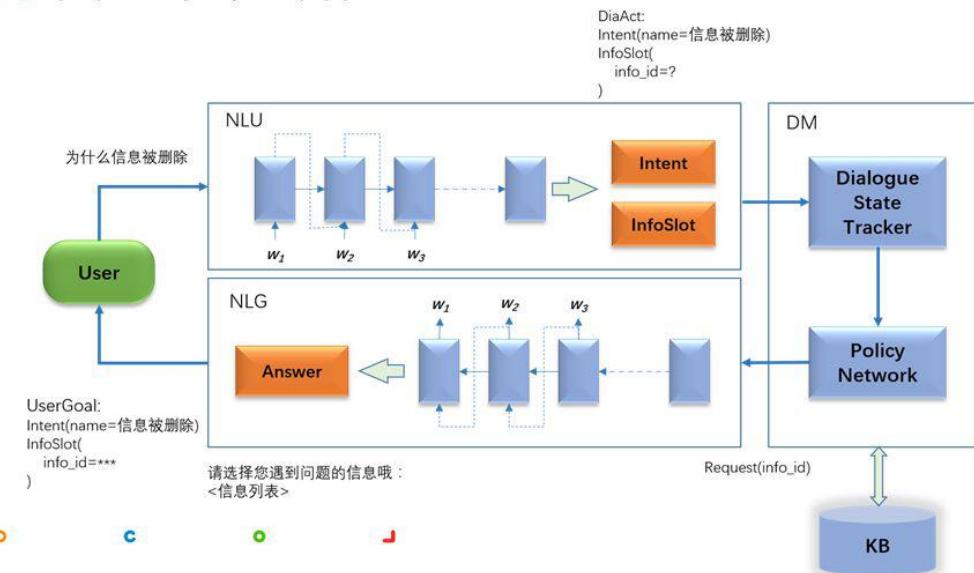


这里还可以使用分类模型来实现问答，一个标准问题有多种扩展问法，每个标准问题可以看做是一个分类，将用户的输入映射到标准问题上即可完成回答，因此可以将问答看做是一个大规模短文本分类的问题。我们采用了多特征、多模型、多分类结果融合的方式来完成短文本分类，在特征层尝试使用了单字、词、词性、词语属性等多种特征，在模型层应用了 FastText、TextCNN 和 Bi-LSTM 等模型，各模型的结果输出最终会做融合得到最终分类结果。

核心功能-任务对话服务 Task-Bot



● 任务型多轮会话

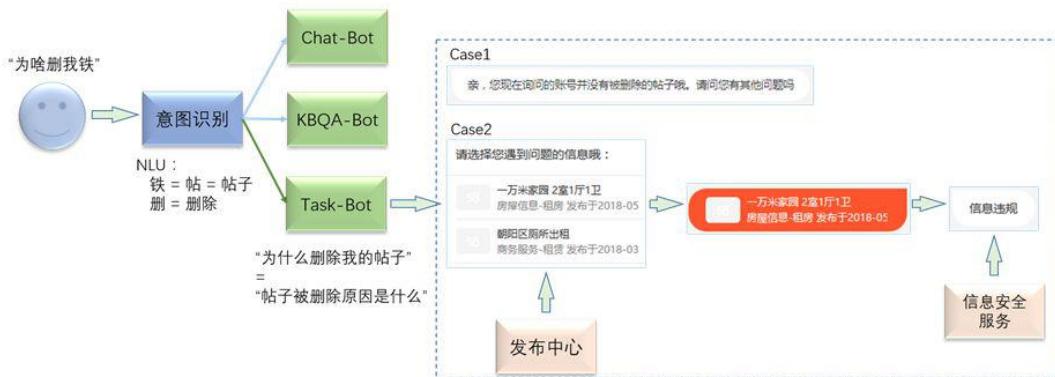


Task-Bot 任务型机器人是在特定条件下提供服务，为了满足带有明确目的的用户，例如查天气、查物流、订机票等任务型场景。用户的需求一般较复杂，通常需要机器人和用户做多轮互动以帮助用户明确目的。我们实现了一个标准的多轮会话系统，首先自然语言理解模块会识别出当前输入问题的意图和槽位，然后输入到对话管理器去决定下一步的回答动作，最终再通过自然语言生成模块生成答案返回给用户。

核心功能-任务对话服务 Task-Bot



● 应用实例

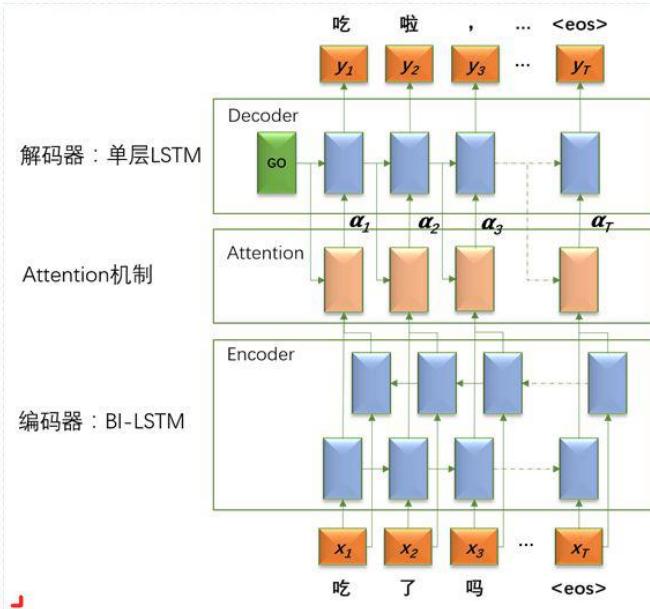


这是一个具体的应用实例，用户输入“为啥删我贴”，经过自然语言理解处理后，意图识别模块会将其识别为任务类型的服务，用户是想询问删除帖子的原因，通常情况下问答系统会反问用户，要求用户提供帖子 ID 才能查询，这里我们通过另一种设计来完成：首先调用发布中心接口拉取用户已发布的帖子列表展示给用户，让用户去自主选择相应的帖子，用户点击具体帖子之后，帖子 ID 会传递给问答系统，问答系统会再调用相关接口查询到帖子删除原因返回给用户。这一整套流程是用户的自助查询过程，相比以往用户需要查询自己的帖子 ID 给客服人员，客服人员登录相关系统并输入帖子 ID 查询结果要高效很多。

核心功能-闲聊服务 Chat-Bot

让 生 活 简 单 美 好 58

- 模板匹配
- 检索式
- SeqSeq生成式对话



闲聊服务是基于一个闲聊语料库，采用模板匹配、检索式回答以及生成式对话等多种技术来实现的。模板匹配使用了 AIML 和正则表达式匹配；检索式回答类似 KB-Bot 中的方式首先检索然后利用模型排序；当模板匹配和检索式回答都不能给出闲聊回答时，我们会采用 SeqSeq 生成式对话，我们使用了一个标准的 Seq2Seq 模型，问题会首先输入到一个双向 LSTM 编码器，然后加入 Attention 机制，最终使用一个单层 LSTM 做解码，从而得到结果输出。生成式对话往往会生成一些让人难以理解的答案，这也是业界难以解决的问题。

人工在线客服转接支持

让 生 活 简 单 美 好 58

- 人工在线客服无缝转接、IM一对聊天支持

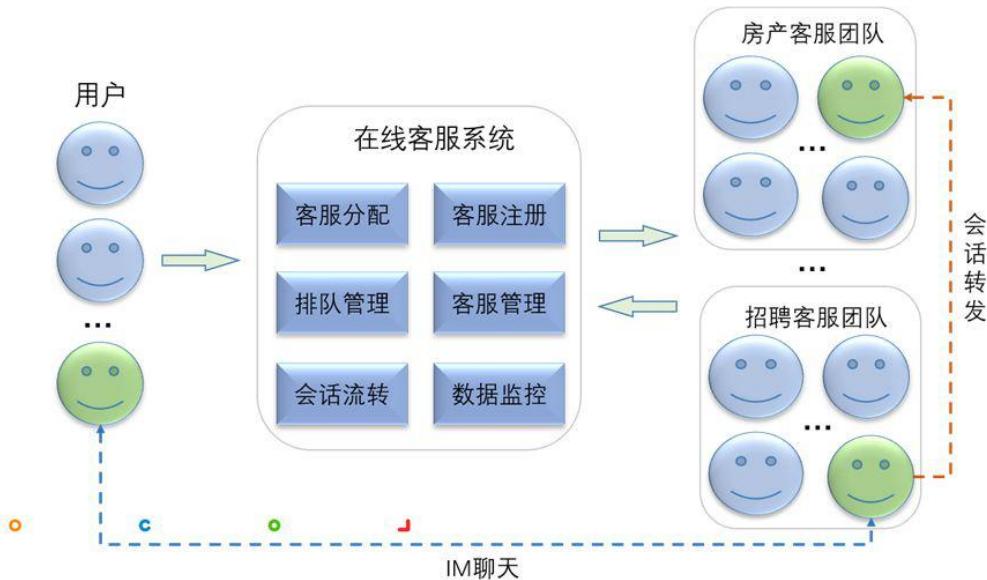


当“帮帮”给出的答案用户不满意时，用户会寻求人工服务。“帮帮”支持人工在线客服的无缝转接，用户只需在聊天窗口一键点击按钮便能连接到IM人工在线客服，实现一对聊天。在转接人工客服成功后，人工客服会在客服工作台中通过一个类似微信的聊天窗口和用户沟通。虽然用户在前端操作简单，其实后面是有一套功能复杂的在线客服系统在支撑。

在线客服系统

让生活简单美好 **58**

- 支持多客服团队接入、客服分配、会话流转



在线客服系统是用户和客服人员沟通的桥梁，在 58 业务场景下，它支持多个业务部门的不同客服团队注册使用，不同客服团队可以管理自己的客服人员。当用户在智能客服窗口点击转接人工客服按钮时，智能客服会识别出用户转向的目标客服团队，在线客服会分配一名客服人员和用户进行沟通。在线客服系统支持用户排队功能，当同时转接人工客服的用户较多而客服人员人力有限时，用户便会进入等待队列。智能客服识别用户业务意图往往存在一定错误率，有时候客服人员在和用户沟通一段时间后会发现用户的业务问题需要其他客服团队来解决，此时客服人员会将会话转交给其他业务团队，因此在线客服系统还需支持会话流转的功能。此外，沟通过程中的数据是非常重要的，例如可以根据人工的沟通记录去优化自动问答的答案，因此数据监控也是必须必备的功能。

评价体系——人工评价

让生活简单美好 **58**

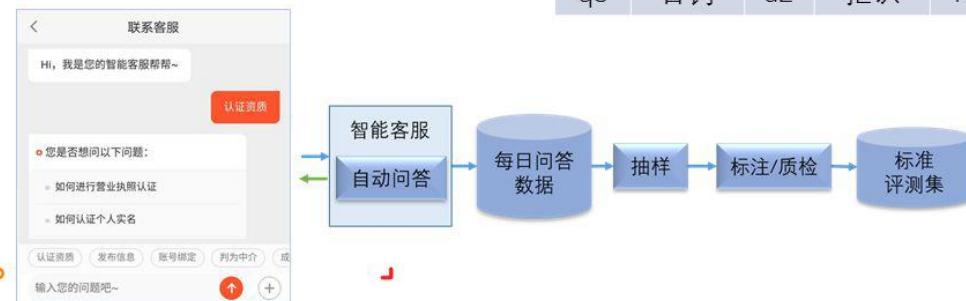
● 评价指标

- 有结果率、拒识率、召回率、准确率

● 评价方法

- 使用人工标注评测集做效果评估

问题	回答类型	答案	实际类型	实际答案
q1	闲聊	a1	闲聊	a1
q2	咨询	a2	咨询	a1
q3	咨询	a2	拒识	NULL



智能客服系统需要有一个完备的评价体系去评价它的好坏，在我们的评价体系中有基于人工标注的评价和基于用户反馈的评价两种方式：

(1) 基于人工标注的评价。“帮帮”能够自动回答业务咨询、任务和闲聊类型的回答，业务咨询类是基于问答知识库来回答的，系统的回答能力受限于知识库的丰富程度，因此并不能回答用户的所有问题，系统最佳的状态是将能回答的全部回答准确，不能回答的全部拒识，即拒绝回答。因此这里的评价指标包括有结果率、拒识率、召回率和准确率等，我们的目标是让系统的有结果率无限接近数据的真实有结果率，召回率和准确率尽量高。这里我们是通过标注标准评测集来计算系统的各项指标，我们会从每日的全量数据集中抽样出一个小数据集，保证小数据集的数据分布尽量符合全量数据集，然后由标注团队对数据集做标注，标注出每个问题的实际答案，一般标注完成后还有质检的环节，以保证标注结果尽量准确，这样便生成了每日数据的标准评测集。基于该标准评测集我们会去评价系统的好坏，并且每次做新模型迭代时都会使用标准评测集去评价新模型，只有新模型的效果好了才允许上线。

评价体系——用户反馈评价

让生活简单美好 **58**

● 如何评价一个用户的咨询真正被解决

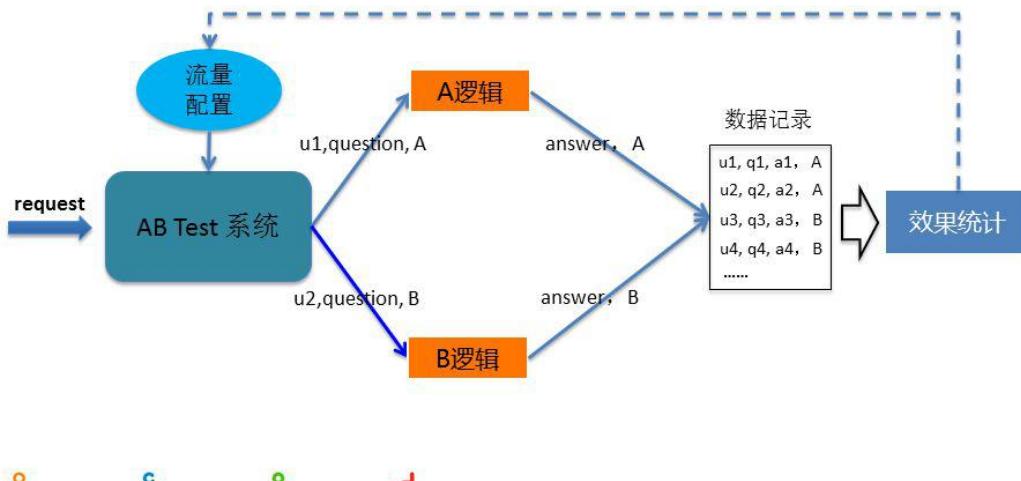
- 促进用户主动评价：智能客服 + 在线客服
- 相关指标：参评率、满意比例、不满意比例



(2) 基于用户反馈的评价。人工评价能够评价智能客服系统的准确率，但是答案是否合理，能否为用户解决问题，需要用户去反馈评价，整个智能客服系统的最终目标是帮助用户解决问题。我们会在产品上设计智能客服和在线客服的评价功能，例如会让用户评价智能客服的每个答案或者某次会话，在和人工客服聊天完毕会发送评价卡片给用户去评价满意度。最终我们会统计参评比例、满意度等指标，这些指标能够真正反应智能客服系统的好坏。实际中往往用户参评比例低，我们会使用各种方法去刺激用户评价。

算法模型迭代：在线ABTest实验

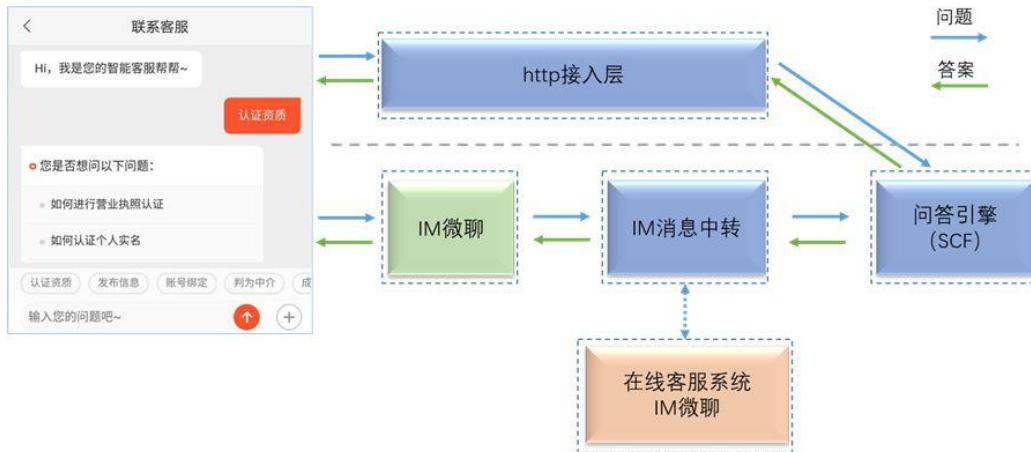
● 分流实验、数据监控



上述内容介绍了“帮帮”智能客服系统中的技术和评价体系，我们在做算法策略迭代时会不断优化评价指标。首先在离线模型迭代时，会基于标准评测集计算离线指标，只有指标提高了才允许模型上线。上线时会做 ABTest 上线，首先将新模型小流量上线，然后看数据效果，若效果好会切换更多的流量进行上线。

帮帮后台总体架构

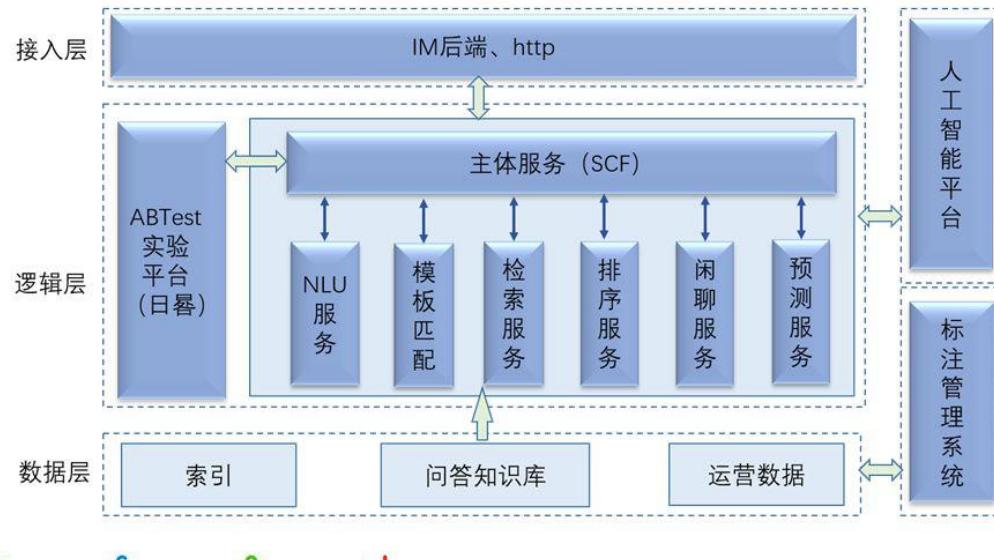
让生活简单美好 **58**



"帮帮" 后台系统总体架构如图所示，"帮帮" 前端页面是一个 IM 聊天窗口，用户在聊天窗口中可以和 "帮帮" 即时对话。这里的具体实现分为两种：第一种是通过微聊（58 同城 TEG 自研的 IM 即时聊天工具）来实现，用户在前端的提问会被当做一条消息发送给微聊，我们有一个 IM 消息中转模块从微聊接收消息，并将消息转发给问答引擎，问答引擎是一个 RPC 服务，使用 SCF 框架（五八同城 TEG 自研的服务通信框架）实现，问答引擎给出答案后返回给 IM 消息中转模块，中转模块将答案组装成消息发送给微聊，最终微聊返回消息给用户。这种方式的实现需要使用我们的微聊通道。还有一些业务方不希望通过微聊来获取 "帮帮" 自动问答功能，只希望我们提供一个接口，业务方输入问题，接口能够返回答案即可，针对这种方式我们在问答引擎之上封装了一层 http 服务，业务方只需要调用该服务即可。

问答引擎后台架构

让生活简单美好 **58**

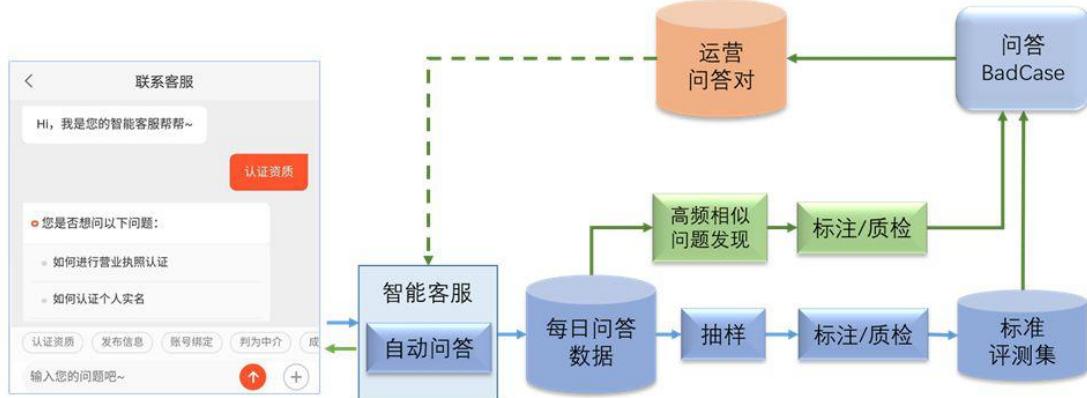


下面介绍下问答引擎的后台架构，问答引擎分为数据层、逻辑层和接入层。数据层包括问答知识库、标注和运营数据以及构建的问答索引。逻辑层里各个功能模块都基于 SCF 框架封装成微服务，包括 NLU 服务、模板匹配服务、检索服务、排序服务、预测服务、闲聊服务、主体服务，主体服务负责对外提供通用接口，接收问答请求，调用各个子服务完成问答逻辑以得到答案，并将答案返回给接入层。这里我们会做 ABTest 实验，主体服务会请求 ABTest 平台“日晷”（自研的包括请求分流和数据监控功能的 ABTest 平台）获取具体分流实验信息。此外，我们的所有算法迭代都是通过自研的人工智能平台来实现，标注和运营数据由 Web 标注管理系统来提供。

结合数据运营提高问答效果

让生活简单美好 **58**

- BadCase发现和运营上线



我们还会通过运营来提高问答效果，针对问答系统的高频 badcase 回答，我们会进行人工修正，并即时同步到线上系统，以保证回答准确。“帮帮”每天产生的问答数据，我们会抽样一部分去做标准评测集的标注，从标注结果中我们可以看到哪些问题回答错误了，我们会将这些问题标上正确答案并即时上线。这是因为线上问答模型的更新周期较长，一般是数天或者一周，通过人工运营可以快速将 badcase 给去掉。标准评测集数据较少，只会包含少量的 badcase，我们还会挖掘每日的全量数据，发现高频相似问题，并交由标注同事标注，若回答错误，也会进行标注运营上线。通过这种结合人工运营的方式，我们可以提高“帮帮”的回答准确率。

结合产品设计提高问答效果

让 生 活 简 单 美 好 58

● 输入提示

- 召回率提升 4%+，准确率提升 8%+



我们还会通过产品设计来提高问答准确率，“帮帮”最主要的功能是解决业务咨询，这是基于我们构建的问答知识库做回答的。因此，可以设计一个输入提示的功能，在用户输入问题时去问答知识库中匹配相关的问题，若匹配到，用户直接选择相关问题即可，此时我们的回答逻辑就是在知识库硬匹配，而不用走算法模型匹配，可以大大提高回答准确性。在实际应用中，我们发现有很大一部分问题会从输入匹配中匹配到，这种方式最终给回答准确率带来了 8% 的提升，效果非常可观。

提高人效：联通内部业务系统

让 生 活 简 单 美 好 58

● 用户自助服务

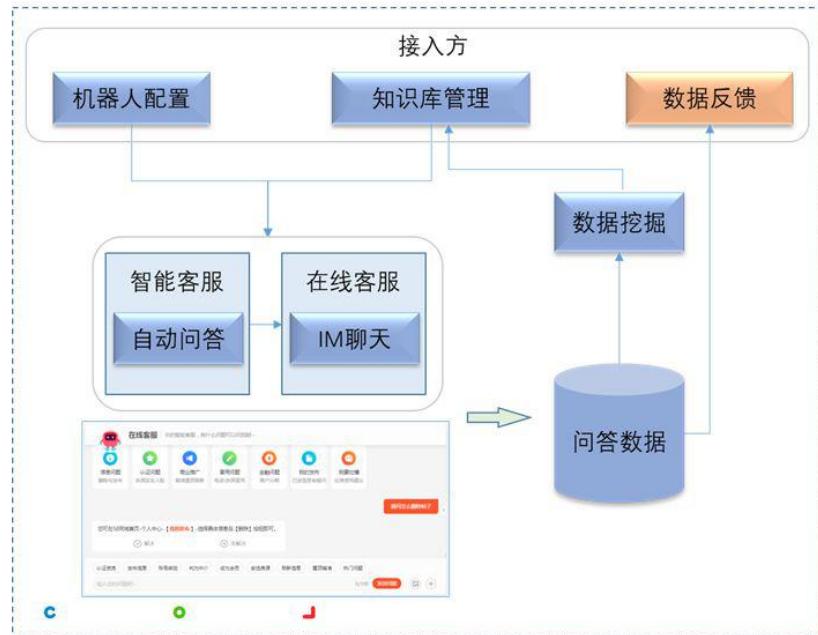
- 查询帖子删除原因
- 彻底删除帖子
- 账号注销
- 申诉举报
- ...



智能客服系统有一个主要目标是提高人效，我们会将很多较复杂的咨询服务做到聊天窗口中，让用户去自助完成，而不是像原来那样用户和人工客服沟通，人工客服去操作内部各个系统以得到答案返回给用户。这样可以减轻我们客服人员的压力，并能提升用户体验。例如用户需要彻底删除自己发布的帖子，旧模式下必须让客服人员去操作，新模式下，只要用户在“帮帮”界面上问到了该问题，我们便会向用户返回他的发布列表，他可以选择某条帖子，直接点击彻底删除按钮即可完成删除。

智能客服Web接入平台

让生活简单美好 58



“帮帮”是一个通用的智能客服平台，需要对接 58 集团内多个业务方，为了提高接入效率，我们设计了一个通用的 Web 接入平台。业务方注册登录接入平台后，只需要简单配置机器人和导入知识库即可获得智能客服能力，例如配置机器人欢迎语、热门问题、配色等，平台会自动生成一个前端页面的链接，业务方可以嵌入到相关入口上。智能客服上线后，我们会将线上数据反馈给接入方，接入方可以在 Web 平台上查看统计数据和明细数据。另外要强调的一点是，我们将知识库的管理开放给接入方来管理，接入方可以导入和更新自己的问答知识库，我们也会对问答数据做分类、聚类、主题抽取等操作，将相关中间结果提供给业务方，业务方基于此来更新知识库。

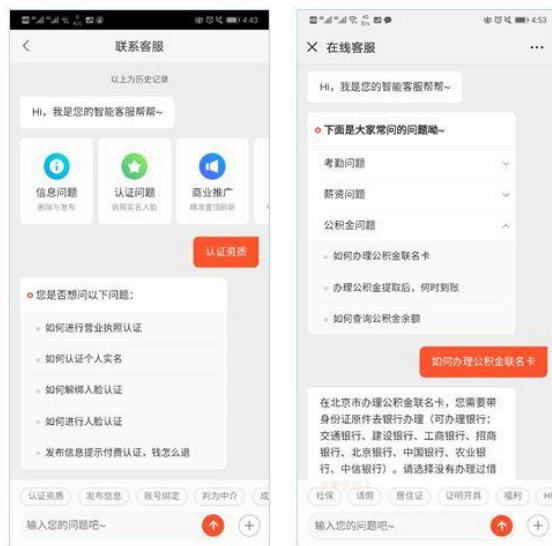
效果数据

让生活简单美好 **58**

- 集团各业务线已接入智能客服，日均解决数万用户咨询

- 五八/赶集/安居客共计30+接入场景
- HR / 行政 / 运维 智能客服

- 召回率90%，准确率85%



“帮帮”已接入了五八集团内五八、赶集和安居客三大平台的三十多个业务场景，每日可以解决数万用户的客服咨询，此外，“帮帮”也被应用于公司内部的HR、行政和运维系统之中，以提高内部工作人员的办公效率。经过我们持续开展算法策略迭代，目前“帮帮”问答系统召回率达到了90%，准确率达到了85%。

作者介绍：

詹坤林，58集团AI Lab负责人，算法高级架构师，负责推动AI技术在58生活服务行业的落地，为集团打造全面AI能力。曾任腾讯高级工程师，负责腾讯微博/腾讯新闻推荐算法研发。

团队介绍：

58集团AI Lab人工智能实验室隶属于58集团TEG架构平台线，旨在推动AI技术在58生活服务行业的落地，驱动各产品业务在人效、用户增长、用户体验等方面的提升。目前主要产品包括人工智能平台、智能客服对话机器人、智能外呼电话机器人、智能写稿、推荐系统和推送系统等。

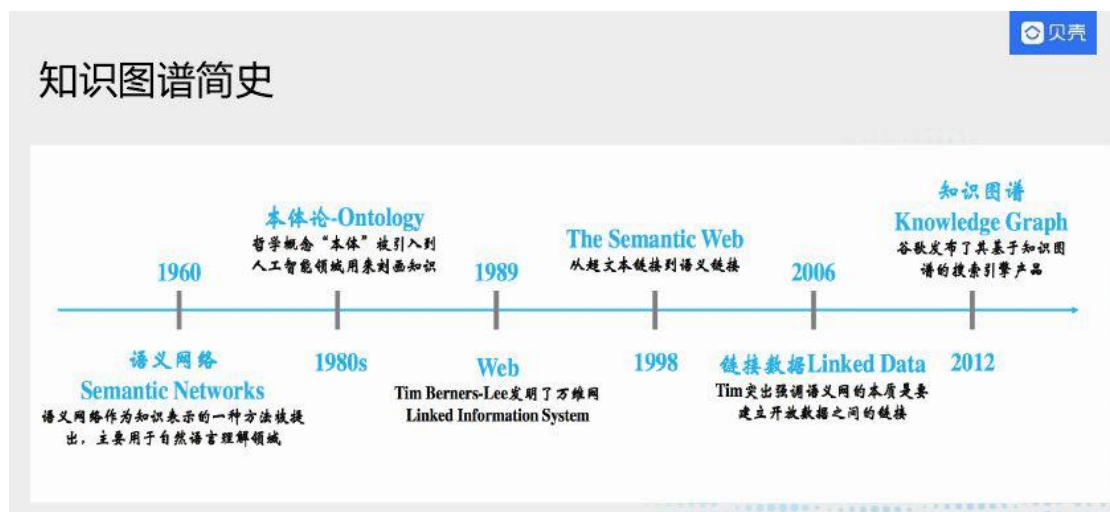
内推信息：

求算法工程师，包括自然语言理解、对话、推荐、语音识别、语音合成、图像识别等方向；求Java后台开发工程师；有意者发送简历至邮箱 zhankunlin@58ganji.com

知识图谱在贝壳找房的从 0 到 1 实践

作者：王贺青 整理：Hoh

今天分享的内容有四个方面，第一个介绍下知识图谱，第二个讲为什么“可以&要”在贝壳找房中落地知识图谱。第三个就是贝壳找房中的知识图谱落地应用，最后讲一下存在挑战和未来展望。



首先介绍下知识图谱的简史。1960 年提出了语义网络，作为知识表示的一种方式，主要是帮助自然语言的理解，典型的就是 WordNet，从不同维度表达词与词之间的语义关系。1980 年提出了本体论，先定义一些本体再定义它们相关的关系，成熟的应用就是专家系统。1989 年提出了万维网，然后 1998 年从超文本连接到语义网络，就是将每一个网页加一个语义含义。到了 2006 年提出了链接数据的概念，将互联网上的数据建立一些联系，如姚明的页面出现他妻子叶莉的信息，会给“叶莉”加一个链接。在 2012 年 Google 提出了知识图谱，目的是提升整个搜索效果。

知识图谱旨在描述真实世界中存在的各种实体或概念，以及他们之间的关联关系。它的每一个实体用全局唯一确定的 ID 来标识，就如每个人都有一个身份证号码；第二个就是用属性-值对来刻画实体的内在特性，用关系来连接两个实体，刻画他们之间的关联。如刻画姚明：属性-值<姚明+身+2.26 米>，关系型<姚明+妻子+叶莉>。

知识图谱优势



从知识图谱的一个发展史及其定义可以看出知识图谱具有 5 个视角优势，首先是 Web 视角，建立数据之间的语义链接、支持语义检索，第二个就是 NLP 视角，对文本进行结构化语义抽取。然后就是 KR 视角（知识表示），利用计算机的符号表示和处理知识。再者就是 AI 视角，利用知识图谱辅助理解人的语言，目前 AI 大部分是在垂直领域落地，会建立自己的知识库，帮助垂直领域人工智能的实施。DB 视角，就是以图的方式存储知识。知识图谱展开其实是一个很大的知识网络，存储时会利用各种的图数据库存储。



目前知识图谱应用场景分为两块，一种是通用领域，一种是垂直领域。通用领域如 Google 的搜索领域，国内的百度和搜狗也在搜索领域应用；还有些聊天领域，如智能机器人、智能手表。这种应用有一个特性就是依赖通用知识图谱，构建依赖国外维基百科，国内有百度百科、搜狗百科，将页面中结构化数据抽取出来构建知识图谱来支撑通用领域的问答和搜索。垂直领域应用越来越多，如金融、电商、公共安全、农业、电信等，如金融里面的反欺诈，

公共安全领域的追捕犯罪分子。不管是通用领域还是垂直领域落地有几个共性条件，第一个必须有一个结构化的数据，这个数据还要高质量，尽可能的海量数据；第二在数据基础上会抽象出一个本体库，从本体层面去定义实体类型，以及表示他们的关系，第三就是有可以利用数据和本体库的智能应用场景，依据知识图谱具有的优势和现有条件来确定业务场景是否需要知识图谱。目前知识图谱支撑的领域有搜索、问答、推荐、图数据关系挖掘。

为什么“可以”在贝壳找房中落地知识图谱

- 丰富数据可利用

来源	数量级	数据类型	覆盖描述
楼盘字典	亿级别实体、10亿+三元组	结构化	房源、客源、小区、学区、地铁站、带看、成交等数据
经纪人与用户对话数据	亿+级别	非结构化	用户找房、咨询经纪人、委托、带看、成交过程中产生的对话数据
用户问答及百科文章	100万+级别	非结构化	用户关于房产领域的问答及百科知识数据

接下来讲一下知识图谱为什么能在贝壳找房中落地。首先我们有丰富的数据可以利用，从两个维度数据：结构化和非结构化。结构化有楼盘字典，数据亿级实体，10亿级的三元组。覆盖类型有房源、客源、小区、学区、地铁站、带看、成交等数据。还有非结构化数据，经纪人与用户对话数据，级别是亿级，主要是用户找房、咨询经纪人、委托、带看、成交过程中产生的对话数据。第三种是用户问答及百科文章，量级是100万+，非结构化数据主要是用户关于房产领域的问答及百科知识数据。

结构化数据楼盘字典覆盖类型有房源、客源、经纪人体系，还会涉及些客户和业主，都会作为链家的参与人。

为什么“要”在贝壳中落地知识图谱

丰富的智能应用场景

- 智能搜索&推荐：提升用户的找房效率
- 数据可视化：分析用户行为，挖掘数据之间关联
- 智能问答：做经纪人的助手



WE ARE BEIKE, 2018 BEIKE ALL RIGHTS RESERVED

然后讲一下为什么要在贝壳找房中落地知识图谱，因为贝壳找房有丰富的应用场景，如智能搜索&推荐：提升用户的找房效率；数据可视化：分析用户行为，挖掘数据之间关联，智能问答：做经纪人的助手潜在客户找房咨询。右边的图是贝壳找房业务中的商业转化漏斗，这个过程涉及用户找房、到咨询经纪人、委托经纪人带看，最后成交几个环节。我们的目标就是帮助获得更多更优质的商机，以及提升他们的服务能力，帮助用户快速找到合适的房子，了解购房知识，拓宽这个漏斗图。

为什么“要”在贝壳中落地知识图谱

• 智能搜索

借助知识图谱AI+Web视角，提升搜索意图理解

● 楼栋 ● 学校 ● 房源 ● 单元 ● 楼盘 ● 员工 ● 地铁口

• 智能推荐

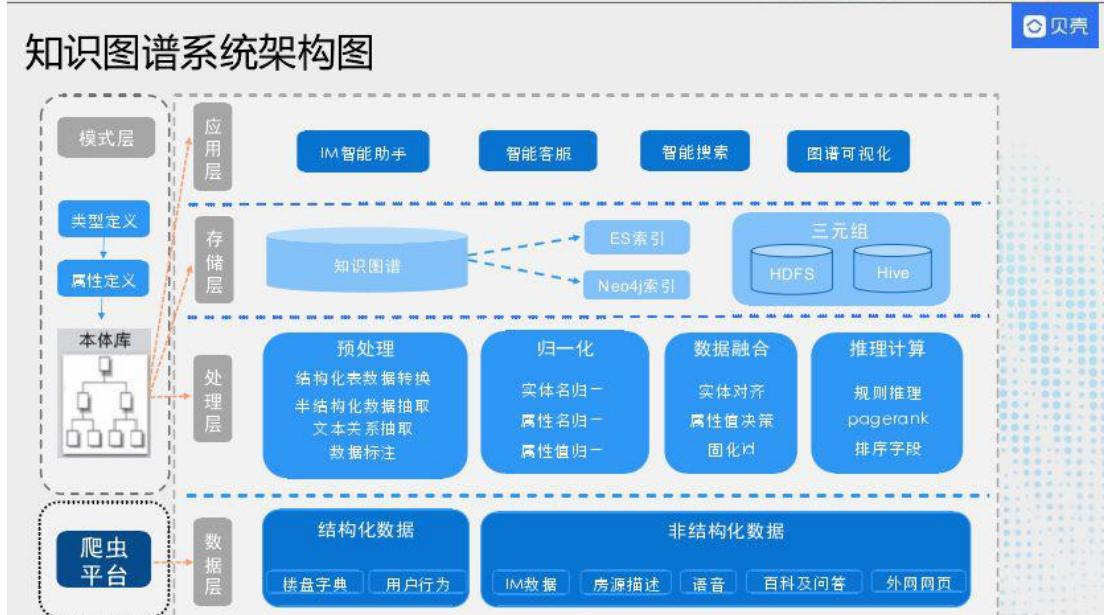
借助知识图谱的AI+DB视角，推荐相关的房子



• 智能问答

借助知识图谱NLP+AI视角，回答经纪人有关房子的问答

在智能搜索方面借助知识图谱 AI+Web 视角，提升搜索意图理解；在智能推荐方面利用知识图谱的 AI+DB 视角；推荐相关的房子，在智能问答方面利用知识图谱 NLP+AI 视角，回答经纪人有关房子的问答。右图我们可以看到，在搜索一个学校时，我们可以看到与这个学校相关联的房源和学区等实体信息。

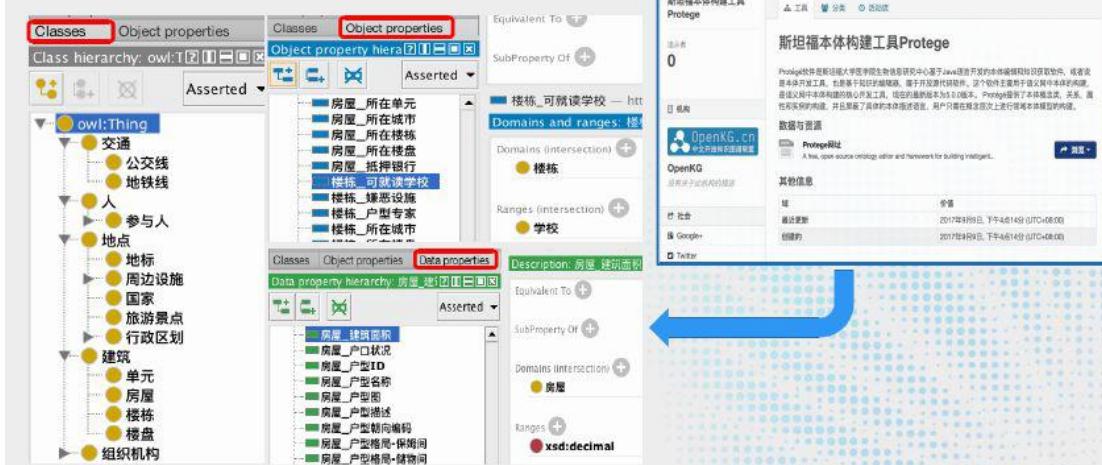


接下来讲一下贝壳找房中的知识图谱落地应用。一个知识图谱系统构建流程通常有五个部分，第一个是定义具体的业务问题，第二个数据搜集与处理，第三个是知识图谱的设计，第四就是知识图谱的存储，最后是应用开发及系统评估。

我们的知识图谱系统架构有五层，在数据层会有外网数据爬虫平台，大部分数据是内网数据，数据分为两块结构化数据和非结构化数据。处理层有预处理、归一化、数据融合以及推理计算。预处理方面结构化数据转换、半结构化数据抽取、文本关系抽取、数据标注，在处理完后会做一些实体名归一、属性归一、属性值归一。数据融合中会做一个实体对齐，因为实体会来自于不同的源，但是表达的是一个实体，可能存在交集或并集，或者一个属性有多个值，会做一个属性值的决策。推理计算会基于现有的数据做一些规则推理补充、pagerank、排序字段。整个生成后会形成一个知识图谱，建立ES索引或者neo4j索引，然后也会在hdfs或者hive里面进行备份，支持不同业务方调用数据。应用层有IM智能助手、智能客服、智能搜索、图谱可视化。左边是我们的模式层，从类型定义、属性定义，最后构建本体库。本体库会支撑数据层、存储层、应用层。

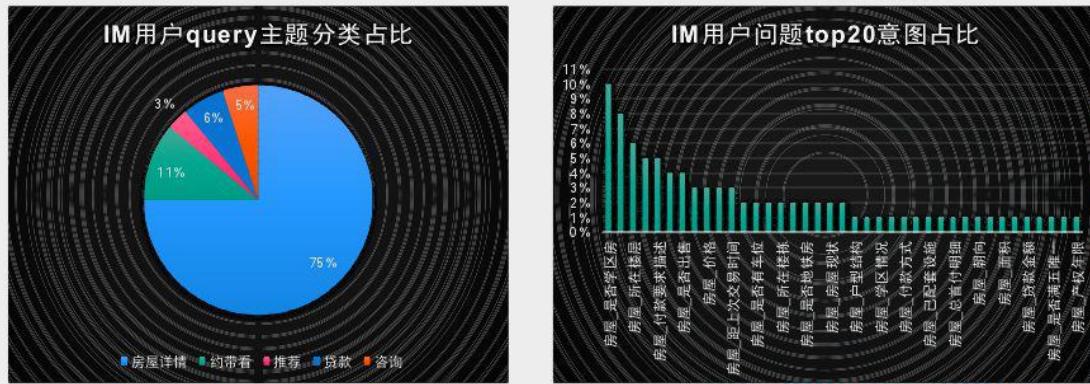
本体库构建 – protege 工具使用

- 结合业务及数据，自下而上与自上而下相结合构建本体库



知识图谱构建第一步工作就是本体构建，通过业务中涉及的问题抽象出相应的类型，我们抽象出四大类型交通、地点、人、组织机构，下面会再划分小的类型，大概有三十多种类型。构建本体库使用构建工具 protégé，主要有三块，第一块是 class，第二部分是 objects，这是个关系型的属性，描述两个实体关系的属性。Data property 是内在属性，描述实体本身的属性。正常本体库构建是自下而上，从数据出发。但在我们的业务中会需要一些自定义的属性，加入一些特殊的约束，因此采用自下而上与自上而下相结合构建本体库。在构建的过程中也加入了对属性关系的自定义约束，如是否加密、显示顺序、是否归一化等约束。

调研分析及问题定义

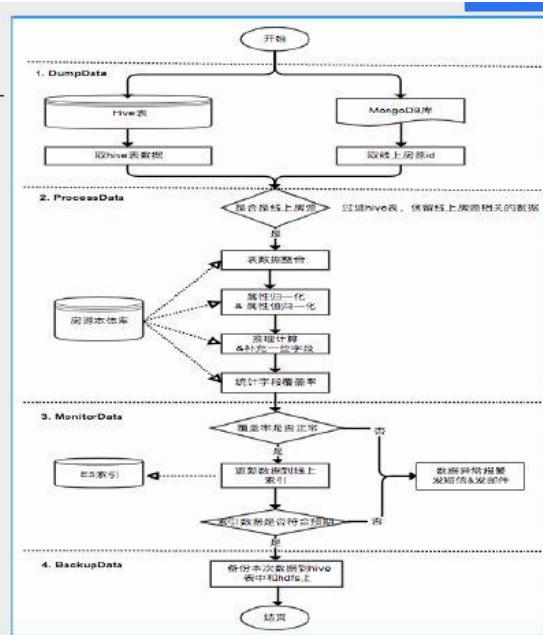


接下来介绍下知识图谱在 IM 智能助手中落地，应用场景是更快解决用户问题，提升经纪人的作业效率。依据历史经纪人与用户聊天数据进行了一个分类，用户主要问五种类型问题，75%是于房屋详情问题、10%是约带看、6%是贷款、3%是推荐。右图是整体 75% 问题中 top20 的意图占比，问的最多的是学区房问题。从 0 到 1 做这个事情，我们优先解决了 75% 房屋详情问题。

数据收集及处理

- **DumpData (取数据)**
从hive取相关房源数据表
- **processData (处理数据)**
表数据合并、属性及属性值归一化、推理计算
- **MonitorData (数据监控)**
监控覆盖率是否正常：更新数据到ES索引
监控索引是否正常：发送报警邮件和短信
- **BackupData (数据备份)**
备份本次数据到hdfs和hive中

WE ARE BEIKE, 2018 BEIKE ALL RIGHTS RESERVED



针对这些问题进行数据搜集与处理流程，首先从楼盘字典中获取所需的房源数据，然后进入知识图谱构建流程，整个统计完成后会计算其覆盖率，最后进入 ES 索引里面，最后会判断数据是否符合预期，符合后才会建立索引，也会对异常进行一些监控报警。



IM 智能助手房源详情检索架构，首先用户问了一个问题，经过 NLU 模块中的分词、然后通过 NER 模块，DA 做实体解析，然后意图识别。意图识别后进入检索模块，生成 SQL，通过 ES 索引查询字段，查询到的字段进行结果的生成。结果生成阶段有个话术设计，让回答结果更人性化，还有个结果拼接。后期会针对不同经纪人自己定义话术模板。



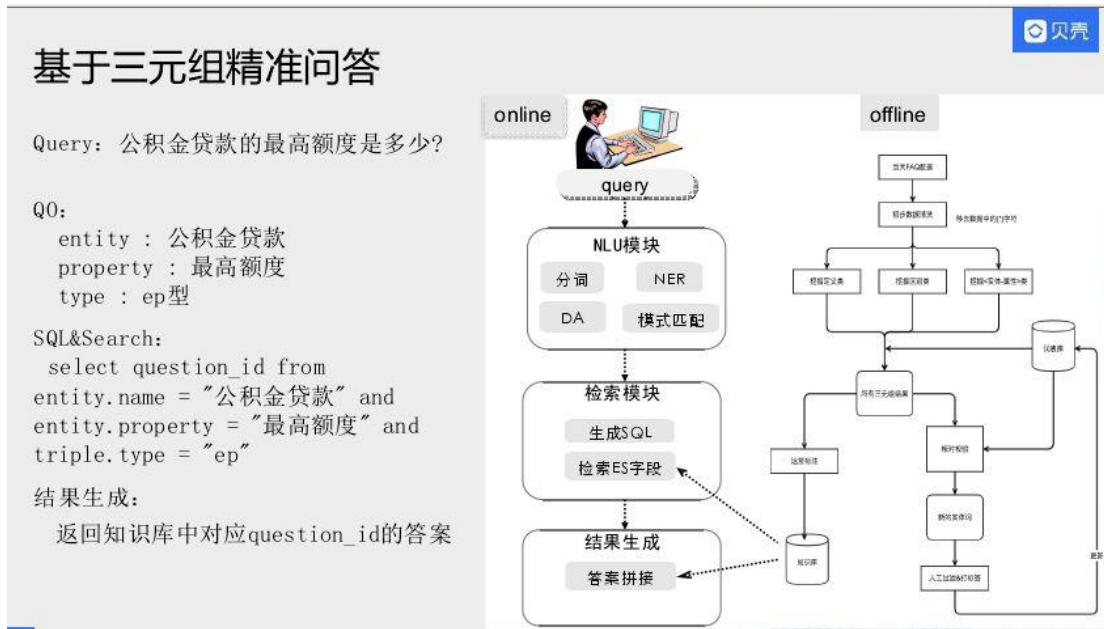
IM 中还有关于知识的问答，但是会存在有些问题答案不匹配，不是很精准，那么如何解决呢。我们对用户经常问的问题进行了分类，用知识图谱方式将问答对表示成三元组和事件三元组。遇到一个事情该怎么办和为什么，得到 how 和 why 事件型三元组；还有实体三元组，分为实体是什么和属性是什么，定义为 what 和 ep 型。第三种是区别的，which 型：实体 A 和实体 B 有什么区别。通过历史 log 随机抽样，符合三元组模式 10%。

目标：精准问答

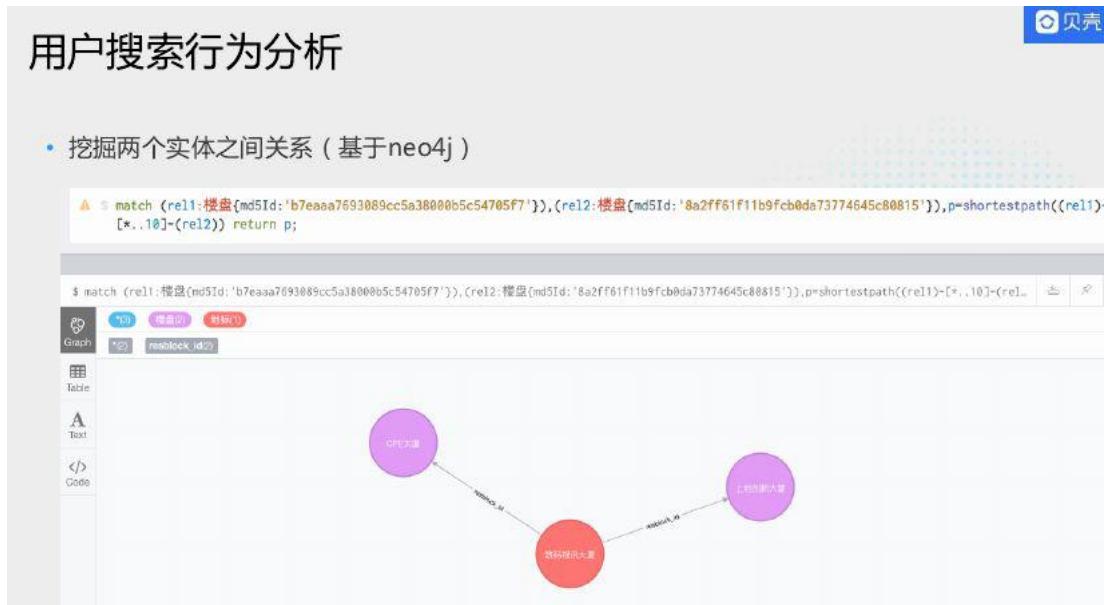
如何精准理解用户搜索意图，找到精准答案

- query1: 公积金贷款的条件？
=> tag: [公积金贷款] [条件] type: ep型
entity property
- query2: 砌体结构墙是什么？
=> tag: [砌体结构墙] [定义] type: what型
entity define
- query3: 签错定金类型了怎么办？
=> tag: [签错定金类型了] [怎么办] type: how型
event property

如何精准理解用户搜索意图，找到精准答案。如“公积金贷款条件？”我们会把公积金作为一个实体，将条件作为一个属性，这种就是 EP 型，还有“砌体结构墙是什么？”这种就是 what 型。“签错定金类型了怎么办？”，将“签错定金”定义为 event，“怎么办”定义为属性，这种归为 how 型。是一种基于语义的搜索而不是传统基于关键词或者语义相似度做计算。

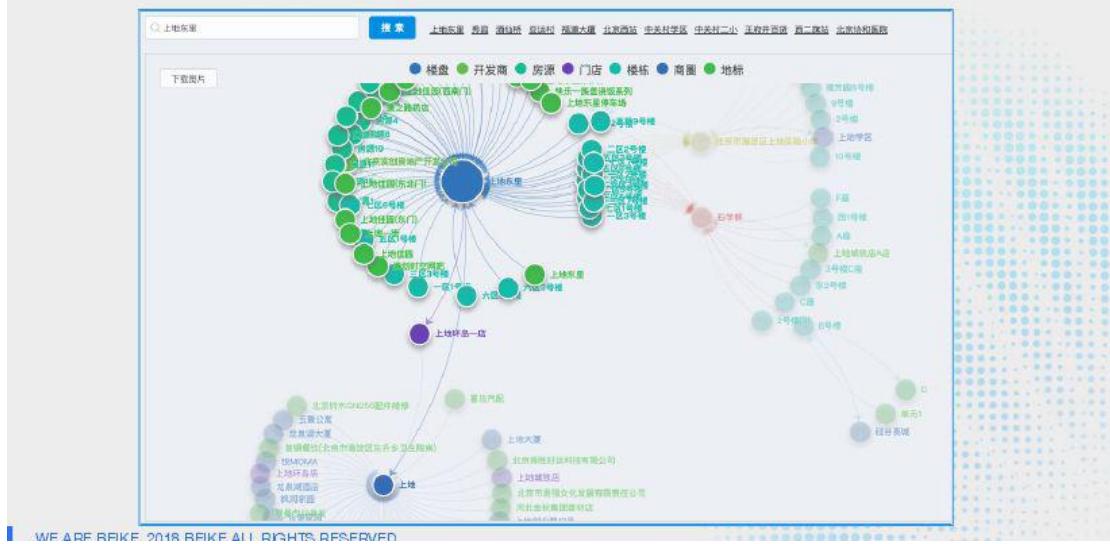


上图是基于三元组的精准问答，分为 online 和 offline。offline 部分我们会离线的去从 FAQ(Frequently Asked Questions , 常见问答库)里面做三元组的抽取和挖掘工作，将历史 FAQ 中符合三元组的问题抽取出相应的三元组。Online 部分同样一句用户 query 去识别意图，进入知识库进行相应的检索，最后返回一个唯一符合的三元组形成答案。当然挖掘得到的三元组需要进行人工标注，目的就是要确保回答的精度和知识库的质量。



知识图谱还可以在很多场景进行应用优化，我们建立知识图谱后，可以通过检索挖掘两个实体间的关系，通过挖掘同一用户不同搜索间的关系，可以更好地做搜索优化、搜索召回优化。当我们搜某一个小区附近的房子，但是没有相应房源，推荐小区附近小区的房子。

图谱数据的可视化



上图是我们自己研发的图数据可视化平台，将贝壳找房所有涉及的实体进行可视化展示。目的是展示内部数据间关联关系，后期让 C 端用户能够更好的找房子而不是直白的搜索列表，返回什么就看什么，可以主动的在知识图谱知识网络中漫游。

总结下今天分享的内容，首先是知识图谱优势五大视角：Web + NLP + KR + AI + DB。知识图谱落地充分必要条件三个方面：数据 + 本体库 + 智能应用场景。贝壳找房中知识图谱落地，1.房产行业数据：结构化楼盘字典数据 + 非结构化的用户文本数据房产；2.行业本体库：支持图谱数据构建 + 智能应用；3.智能应用：智能搜索 + 智能问答 + 智能推荐 + 图谱数据可视化。

最后和大家分享下我们实践过程遇到的问题：1.构建本体库需要房产行业专家的加入及规范，2.房产行业实体词类型、实体词需要规范化及挖掘，大量非结构文本数据亟待结构化。未来的挑战：打造出房产领域最权威的知识图谱。提升 B 端智能应用效果，逐步推向 C 端。将知识图谱深入结合贝壳找房业务场景深入结合，将图谱与地图找房、VR 找房应用相结合。

作者介绍

王贺青，贝壳找房高级知识图谱工程师，现负责贝壳找房房产知识图谱的研发及落地应用。曾就职于搜狗，参与搜狗搜索通用领域知识图谱研发及应用。

团队介绍

贝壳找房智能搜索团队是贝壳找房基础技术中心的核心团队，负责贝壳找房搜索平台、智能搜索、智能客服等多个项目，同时负责打造房产行业数据仓库。在这里，你有机会和算法专

家一起专研最新技术:包括知识图谱、NLP、数据挖掘，利用链家网数亿级房源信息、百科数据、UGC 数据构建房产知识图谱，让房源、客户、经纪人之间的数据互联更加智能；也有机会将技术应用在最前沿的业务领域:智能问答、个性化推荐、智能客服、语音质检等。

内推信息

如下职位：数据挖掘工程师、自然语言处理工程师、知识图谱研发工程师，有意加入贝壳的小伙伴可直接投递贺青老师的邮箱：wangheqing001@ke.com。

人机交互技术介绍

作者：翁嘉颀 整理：Hoh

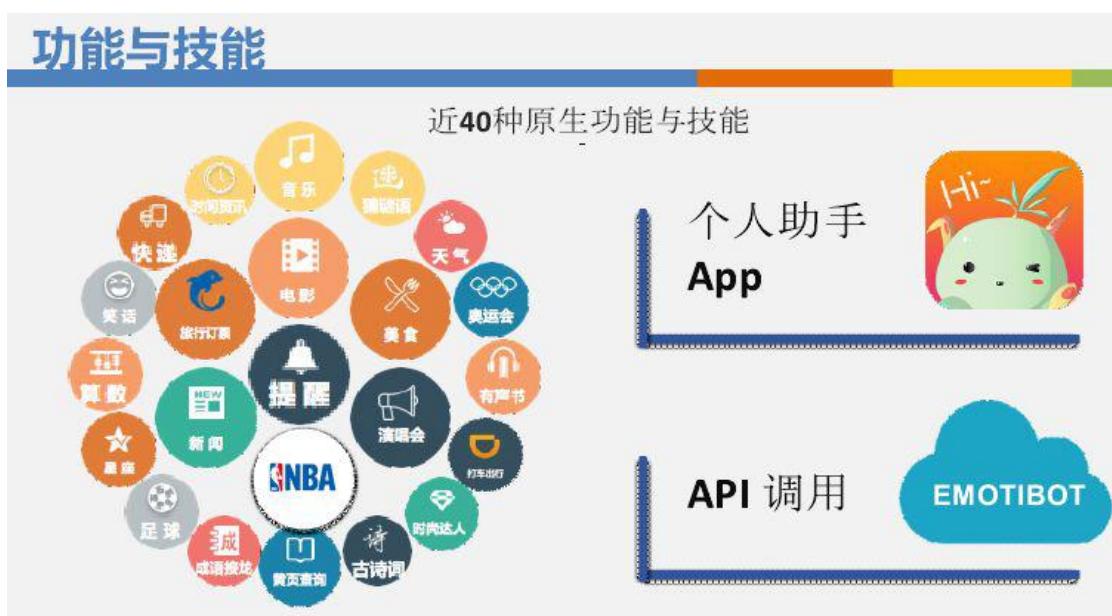
前人工智能时代是基于关键词模板，只能接受固定命令，能力无法持续提升，不能识别用户情绪，没有用户记忆。如现在的智能音箱，你如果对音箱说“我现在吃很饱”关键词是“吃饭”，音箱就会为你推荐附近的菜馆。当然我们的期望不是这样，期望可以用比较好的方式来做。如“我不喜欢吃牛肉面”，看到句子是一个否定陈述句，这不是一个命令句和意图，所以可以避开定外卖这个意图。利用自然语言理解意图，自然语言目前分为三个层次，第一个是nlp，语句分词；第二层叫意图理解，“肚子好饿”和“我想吃东西”这两个意图相近；第三层就是结合场景、理解用户情绪，实现意图识别。



接下来讲一下情感计算的设计与应用，我们做了 22 种文字情绪，计算出你寂寞和无聊的情感要干嘛。但是仅仅做负、正、中情绪是不够的，图中红色代表愤怒，蓝色是讨厌、难过，紫色是害怕，都是负面情绪，但是这三种负面情绪是不一样的，因此机器反馈也是不一样的。除此之外还做了 9 种人脸表情、4 种语音情绪，将“文字+表情+语音”形成多模态情感。这样的意义是什么呢，举个例子如“高考我考了 500 分”这个时候你是应该恭喜还是安慰呢，但是如果加入说话的语气就完全不一样，语音情绪表达更多信息。再加上人脸表情，如面带微笑说“你死定了”这个时候很难判断意图，结合上下文，如果都是微笑那就是开玩笑，如果两者在吵架，这个时候就是威胁。



上面是一个多模态情感例子，上面是人脸表情，中间是语音情绪，下面是文字情绪，图片的正上方是其总情绪。



有了情绪接下来是做一个好的机器人，那么一个好的机器人如何制作呢。目前市面上的聊天机器人智商在 2-3 岁，竹间科技的机器人在 5-6 岁，它可以查天气、查快递、查股票、成语接龙等 40 种原生功能与技能。

知识图谱

- 少数机器人一定胜过人类的部分
- (1) 基本三元组模型 (A 的 B = C)
 - 姚明的身高 = 226cm
 - 姚明的老婆 = 叶莉
 - 叶莉的身高 = 190cm
- (2) 知识图谱的推论
 - 一阶层 → 姚明跟叶莉是什么关系？
 - 一阶层 → 姚明的老婆有多高？
 - 二阶层 → 谢霆锋跟陈小春有什么关系？
 - 多阶层 → 陆奇加入百度之后股价涨还是跌？
- (3) Bi-attention flow
 - 利用深度学习模型
 - 读取一段文字 / 一篇文章，在模型内形成知识图谱数据结构
 - 询问问题，由知识图谱深度模型寻找答案
- 缺点：不可控

再者就是知识图谱，机器人很多用户是小朋友，针对同一个问题需要识别不同问法，识别什么是疑问句、感叹句。然后做一些推论，如“姚明的老婆有多高”先找到姚明的老婆，然后推出叶莉的身高 190cm。“谢霆锋跟陈小春有什么关系”通过知识图谱可以推断谢霆锋的前妻的前男友是陈小春，还有“陆奇加入百度之后股价涨还是跌”，知识图谱需要找到“陆奇”是谁，百度是什么，陆奇是哪一天离开的，哪一天百度的股价是多少等等信息然后推断，这些是机器少数能替代人的地方。还有就是 Bi-attention flow，利用深度学习模型，询问问题，由知识图谱深度模型寻找答案，这个存在缺点是不可控。

对话主题

- 根据目前主题，(1) 决定答案的主题 (2) 主动跳转对话主题
- 主题有阶层关系
 - 例如：你喜欢英超哪支球队？
 - 主题：运动 → 足球 → 五大联赛 → 英超
- 回答1：其实我喜欢巴萨 (运动 → 足球 → 五大联赛 → 西甲)
- 回答2：我比较喜欢看 NBA (运动 → 篮球 → 职业篮球 → NBA)
- 回答3：我喜欢吃蛋炒饭 (美食)
- 机器人主动引导话题 -- 根据 memory，根据 user profile，根据前面的对话
- 机器人主动推荐

闲聊不是随便的聊，好的闲聊需要用主题控制，主题还有阶层关系，如“你喜欢英超哪支球队？”该问句的主题是运动下面的足球底下的五大联赛的英超，如果回答“我喜欢巴萨”或者“我喜欢蛋炒饭”这些都是不对的。现在对话的主题是体育里面的足球，因此回答应该更强烈的选择与足球相关的，依据主题做对话的控制。那如何做话题的跳转呢，机器人主动

引导话题跳转，依据你对话中的相关属性来主导话题，或者根据 memory，根据 user profile，根据前面的对话来主导话题。

上下文理解

- 人，不会每次都讲完整的句子
- Q1：明天我们去看电影好不好?
- A1：明天有事不行
- Q2：**那后天呢？**
- Q1：明天上海会不会下雨?
- A1：明天上海小雨，气温 10 ~ 18度
- Q2：**那后天呢？**
- Q1：你们有卖净水器吗？
- A1：有啊
- Q2：**占不占地方啊？**
- A2：不会，很小的
- Q3：**怎么卖啊？**

黑色基本上定义为没有任何可见光进入视觉范围，和白色正相反。白色是所有可见光光谱内的光都同时进入视觉范围内。

接下来讲一下上下文理解，人不会每次都讲完整的句子，如“Q1：明天我们去看电影好不好? A1：明天有事不行，Q2：那后天呢？”，那后天就代表后天我们去看电影好不好，这种就是第一种主谓宾的补全。第二种就是指代消减，如“我喜欢大张伟，我也喜欢他”这个他就是指大张伟。第三种是话题式，如“Q1：你们有卖净水器吗？A1：有啊 Q2：占不占地方啊？ A2：不会，很小的 Q3：怎么卖啊？”，根据目前的话题进行上下文的补全。

一个好的机器人还需要一些记忆能力，长时记忆，如“我不喜欢吃辣的”，那么下次推荐餐馆就避免推荐辣的餐馆。永久记忆，我今天肚子不舒服，那么就不能聊大姨妈的事情。短时记忆，一般是 48 小时到 72 小时，如“明天要去苏州见张先生”，晚上询问明天要去哪里，回答明天要去苏州。这种就是短时知识图谱，放在用户里面，在你问问句时回答你。

Prediction & Generation

根据上下文，去预测下一句话

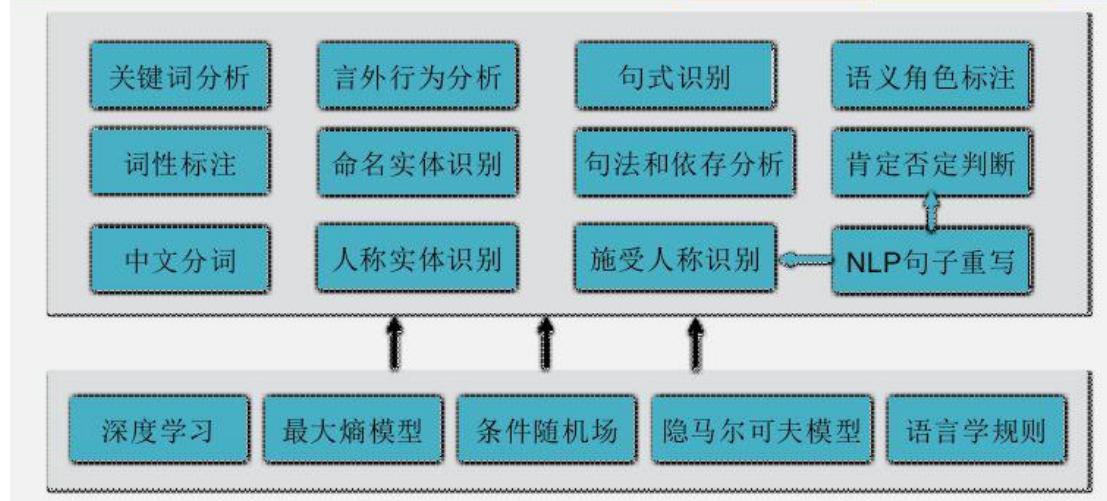
- 预测 主题
- 预测 意图
- 预测 Keywords
- 预测 句型
- 预测 情感变化
- . . .

有了句型，有了关键词

- 能否直接造句？
- 能否根据上下文生成？
- 能否根据不同的用户说话习惯来生成？

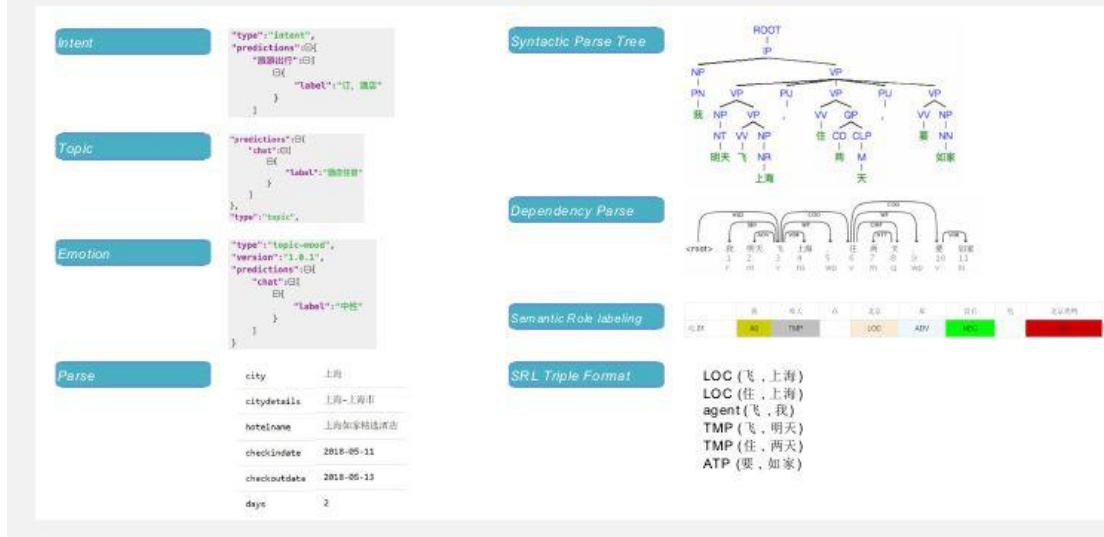
人机对话还有一些“Prediction & Generation”，根据上下文去预测下一句话，去预测其主题、意图、keywords、句型、情感变化等。有了句型、关键词，进行造句，根据上下文生成或者根据不同的用户说话习惯来生成。那么如何实现呢，通过外包请人聊天获取数据，通过数据清洗等操作然后做预测。

NLP 的模块架构



接下来讲一下 NLP 模型架构，其中最核心的是中文分词，分词不对语义理解肯定错误；然后是词性标注，名词、形容词、副词等标注；还有就是句式识别，如“人民广场怎么走？”，“你喜不喜欢吃苹果？”这些都是上下文对话的基础，有的是询问信息，有的是问你个人喜好等。“你在北京买衣服花了好多钱？”和“你上个月在北京买衣服花了好多钱”，一个是一般疑问句一个是感叹句，说的意思也是完全不同。

Example : 我明天飞上海，住两天，要如家



如“你好可爱”，如果分词为“你好”和“可爱”，会认为你是和名叫可爱的人打招呼。那我们怎么做呢，举例说明，如“我明天飞上海，住两天，要如家”，核心动词“飞上海”、“住两天”、“要如家”，核心是“飞、住、要”，“住两天”知道意图是要订酒店。第一种做法是将整个句子丢到一个黑盒子中模型训练得出意图，这种需要大量的基础数据。第二种将句子进行拆分，在丢到模型中训练，这样会简单很多。

如何利用 NLP 的基础信息呢，如“上周买衣服花了多少钱？”，首先知道是一个数量问句，核心动词“花钱买衣服”类别是衣服，时间是上周，通过拆解进行判断。

对话式 & 主动式

- 67% 的用户决定转人工，不是因为匹配错误，而是【答案看不懂】
- 25% 的用户决定转人工，不是因为匹配错误，而是【新问题】
- 只有 8% 的用户决定转人工，是因为算法匹配错误
- 对话方式的交互，才能解决主要问题！

- 机器人是否不再被动？
 - 根据你的 image, 主动跟你交谈
 - 根据你的 profile, 主动跟你交谈
- 机器视觉 + 人机对话交互

接下来看一下人机交互下一步的变化，目前情况 67% 的用户决定转人工，不是因为匹配错误，而是【答案看不懂】，25% 的用户决定转人工，不是因为匹配错误，而是【新问题】，

只有 8% 的用户决定转人工，是因为算法匹配错误。因此应该采用交互的方式，才能解决主要问题。不是所有的问题都是多轮，只要在某一区域的前二十个问题做到多轮就有很大的提升。第二个机器人不再被动，主动和你聊天，根据你的 image，主动跟你交谈，根据你的 profile，主动跟你交谈。还有就是依据机器视觉实现人机对话交互。

结合用户画像和用户的多轮对话，作为条件制定策略，进行商品或服务的推荐。

多轮之间的切换 – 中控中心

- Q1：我想要订酒店 → (意图：订酒店，找上[订酒店机器人](#))
- A1：您要订哪个都市的酒店？
- Q2：上海的 → (还在订酒店的场景，[订酒店机器人](#)继续回应)
- A2：请问您什么时候入住？住几天？
- Q3：后天上海会不会下雨？→ (还在订酒店场景，但是同时命中[天气机器人](#))
- A3：后天上海天气晴
- Q4：那我后天入住，住两个晚上 → (回到[订酒店机器人](#)，继续未完成的场景)
- A4：请问您是否有指定的酒店名称？
- Q5：我失恋了！！
- A5：我无法理解您的意思，请问您是否有指定的酒店名称？

还有一个目前比较潮流的就是多轮之间的切换-中控中心，如订酒店，找到订酒店机器人，然后询问上海是否下雨，还在订酒店场景，但是同时命中天气机器人，然后询问“那我后天入住，住两个晚上”（回到订酒店机器人，继续未完成的场景）。依据多轮对话实现不同机器人的切换。

作者介绍

翁嘉颀，竹间智能 CTO、联合创始人。熟悉算法、编程语言、搜索引擎、网络安全以及邮件安全，使用过的语言超过 35 种。作为 AI 领域的技术专家，他带领团队负责竹间在 AI 领域产品研发与技术规划，领域主要涵盖对话机器人、计算机视觉、金融科技等领域。

内推信息

机器学习算法工程师、自然语言处理专家、语音识别工程师，如上方向，有对竹间感兴趣的欢迎投递翁老师邮箱：phantomweng@emotibot.com，base 上海徐汇区宜山路。

多轮对话提升自动化流程服务

作者：王海良 整理：Hoh

今天主要分享两部分，一部分是技术分享，第二部分是介绍一下 Chatopera 提供的企业聊天机器人应用解决方案。目前开发企业聊天机器人很麻烦，需要用大量的数据，依赖机器学习和熟悉自然语言处理的专家，成本比较高，我们能提供快速落地、稳定的、低成本实现聊天机器人的方案，下图是我们的解决方案。

关于我们



图：企业对话应用解决方案

Chatopera 面向企业业务人员发布了**多轮对话设计器**，用于设计满足企业需求的聊天机器人，从多轮对话设计器中可以导出**对话应用**。对话应用可以导入到**智能问答引擎**中，智能问答引擎是面向企业 IT 人员的，它可以管理聊天机器人，包括多轮对话、知识库、意图识别和监控接口使用情况。**智能问答引擎**暴露接口对外集成，也包含基于 Web 的管理控制台，这套方案可以支持将企业内部流程和在客服、营销过程中的话术转化为聊天服务，接入到企业微信、微信公众号等，从而提供企业的智能化和自动化。闲话不多说了，下面开始分享一些自然语言处理的知识。

语言模型

统计语言模型，也被称为语言模型 (language model)，是通过统计方法来计算一个句子的概率的模型。假设 C 表示一个语料库，D 表示 C 中一个文档，D 由一组顺序排列的单词 (w_1, w_2, \dots, w_t) 组成，t 是句子长度，则 D 在 C 中出现的概率可以表示为：

$$P(D) = P(w_1, w_2, \dots, w_t) \quad (1)$$

$$= P(w_1)P(w_2|w_1)P(w_3|w_2, w_1)\dots P(w_t|w_1, w_2, \dots, w_{t-1}) \quad (2)$$

应用

语言模型用于各种自然语言处理的任务中，如机器翻译、自动分词、语音识别和文本纠错等，从原则上讲，图灵测试就是语言模型的要解决的问题之一。

语言模型是计算一个句子出现的可能性，比如我问机器人一个问题，那么它的不同回答出现的可能性是什么样的。自然语言处理经历过很多阶段，不同发展时期有不同的特点，经历过经验主义主导和理性主义主导的时期，前者主要是语言学家推出的一些方法，后者则大量使用统计学方法。自然语言处理又应用于很多任务中，比如机器翻译、信息检索和阅读理解，近年来取得重大突破的主要是基于统计学方法的。用大量的数据统计出语言的特征，从中找到规律，从而对给定输入预测出结果。第一位提出用数学解决语言问题是香农，他是信息论的开创者，也是人工智能之父。现在很多用于人工智能的方法更多的是来源于通信领域，比如最大熵、最大似然都来自于信息论。

语言模型

ARPA 格式介绍

机器学习中，常使用很大的语料库训练 N-gram 语言模型，并且记录为 ARPA 格式的文件，ARPA 格式被很多自然语言处理的工具包支持。使用 ARPA 格式文件，计算一个字符串的出现概率方法如下：查找这个字符串是否在模型中，如果在直接返回；如果不在，则用下面公式。

$$P(\text{word}_N | \text{word}_{N-1}, \dots, \text{word}_1) \quad (6)$$

$$= P(\text{word}_N | \text{word}_{N-1}, \dots, \text{word}_2) \times \text{backoff-weight}(\text{word}_{N-1} | \text{word}_{N-2}, \dots, \text{word}_1) \quad (7)$$

公式7就是将这个字符串出现的概率分解成两部分，然后分别求，如果左半部分对应的 backoff-weight 不存在，则使用 1.0 作为默认值。利用上述方式迭代右半部分，就得到了解。

检验一个语言模型的好坏是通过“困惑度”来衡量，就是你说了一个字，下一个字有几种可能，可能数越少说明语言模型越准确。描述一个语言模型的格式常用 ARPA，很多工具都支持这种格式，用很大的语料训练形成一个文件，文件带有 N-Grams 的 token 列表，第一列是 token，第二列是它出现的概率，通常是以 Log10 函数换算后的结果，第三列是 backoff 加权，主要用来平滑。计算一句话出现的可能性用途有很多，比如纠错中看那个字是错误的。公式 7 就是具体计算一句话可能性的，它将目标分成两个部分，然后分别求，右半部分是对应的 backoff 权值，如果它不存在，就使用 1.0 作为默认值，左半部分如果在语言模型中也不存在，就迭代计算。

最大熵学习算法

最大熵模型的求解过程可以表示为带有约束的最优化问题，按照习惯思路，将求最大值问题改写为求等价最小值问题，这样目标函数为凸函数，方便使用**凸优化** (convex optimization) 的方法求极值，则有：

$$\min_{P \in C} -H(P) = \sum_{x,y} \tilde{P}(x) \cdot P(y|x) \cdot \ln P(y|x) \quad (8)$$

$$s.t. \quad E_P(f_i) = E_{\tilde{P}}(f_i), \quad i = 1, 2, \dots, n \quad (9)$$

$$\sum_y P(y|x) = 1 \quad (10)$$

如果解决一个翻译任务，就可以用历史数据去训练模型，我们的目的是得到一个函数，通过它预测未来输入的句子的翻译结果。我们不能知道完整的特征空间，但是基于大数定理，我们认为在训练数据很大的情况下，训练后的模型与完整特征空间是一致的，这就是用已知数据去拟合完整的特征空间，利用数学原理，我们能得到的解不一定是最优解，但可以保证是局部的最优。最大熵原理是指导这个求解过程的重要思想，它的核心就是对未出现的事件，认为是等可能的，这样保证熵最大。因为熵增定律认为系统总是朝着最无序，最混乱的方向发展，那么保证熵最大，可以最接近真实情况。上图可以用来描述最大熵模型，首先定义计算熵的公式，然后目标是最小化它的对偶问题（公式 8），然后描述它的限制条件，限制条件为若干特征函数，特征函数的构造一般是输出值为 0 或 1 的函数，再一个条件就是对于一个输入 x ，各种输出的概率和是 1。然后将目标函数和限制条件通过拉格朗日乘子法建立方程组，对每个方程求偏导，进行求解。这是一个凸优化问题。

中文分词

基于 HMM 的由字构词分词系统

HMM 可以用来求解解码问题，通过维特比算法可以根据观测状态序列计算出对应的最优状态序列，这个思路与由字构词的思想相结合，就提供了一个分词系统的实现方案。

状态 (*states*)：(*B, M, E, S*)，文字在词中出现的位置。

观测状态 (*observations*)：(人, 们,, *OOO*)，训练集的独立的文字，其中，*OOO* 代表 Out of Observations，是为那些未曾在训练集内出现的文字定义的。

初始概率 (π)： $B : 0.5, M : 0, E : 0, S : 0.5$ ，对于一个句子的第一个字，要么属于 *B*，要么属于 *S*，假设二者出现的频率一样。

中文分词、词性标注、命名实体识别是自然语言处理中的“三姐妹”，是其它任务的基础。中文分词这些年来出现了很多方法，在九几年的时候主要通过字典和人为指定的规则完成，比如 MMSEG 算法，提出语素自由度概念，前向或后向的算各种分词情况下的分数，然后确定最优解。2000 年以后，多用机器学习算法，准确率也大幅提升，解决了识别新词等困难，有些分词器准确率能达到 96% 以上。隐马尔可夫模型是很经典的模型，也很简单，很多分词器基于它实现。使用隐马尔可夫模型可以解决三类问题：

概率计算问题：计算可观测序列出现的概率；

预测问题：根据可观测序列找到最有可能的隐藏状态序列；

学习问题：估计隐马尔可夫模型的参数。

隐马尔可夫模型包含五个参数：不可观测的**状态**，可观测的**观测序列**，初始状态概率向量，观测概率矩阵和状态转移概率矩阵。在中文分词中，由字构词法就是将状态分为(*B,M,E,S*)，它们分别代表一个字出现在词汇的开始、中间、结尾或独立成词。近年来，基于条件随机场的分词效果超过了隐马尔可夫，主要是因为条件随机场能更充分的利用一个字的上下文关系，能更好的描述序列化任务，因为隐马设定了比较强的依赖，只是利用了前一个字。

相似度计算

Word2vec: C-BOW 的原理

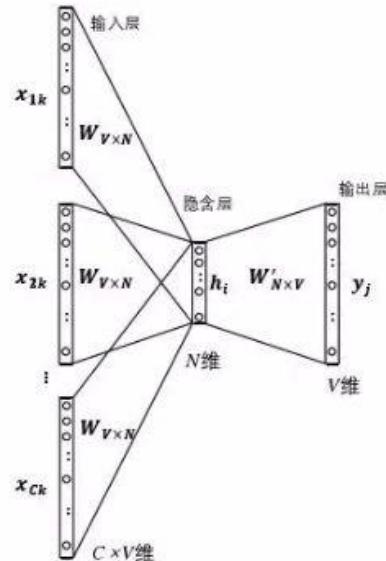


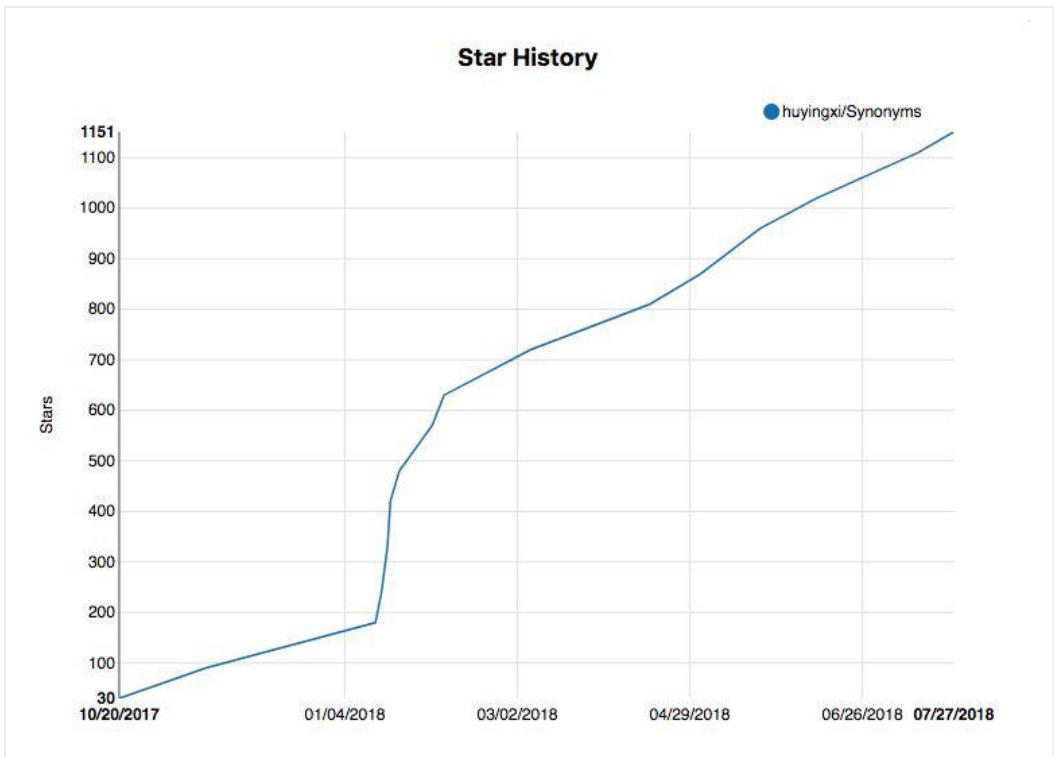
图: C-BOW 模型

在很多自然语言处理任务中，比如搜索、摘要和关键词提取都与相似度计算关系很大，尤其是两个句子和两个词之间的相似度。目前基于词向量的相似度计算用的比较多的是 Word2vec，它的网络设计是很简单的，比如 C-BOW 模型，就是利用前后词去预测当前词。Synonyms 是一个开源的可以用于计算相似度的库，它融合了语义上的距离以及匹配，利用开源的算法和开放的 wikidata 数据制作的。

<https://github.com/huyingxi/Synonyms>

在最近受到了一些关注，在 Github 上，star 数量在稳定增长，我是这个项目的作者之一，欢迎大家使用和提 Issue。

Synonyms star 趋势

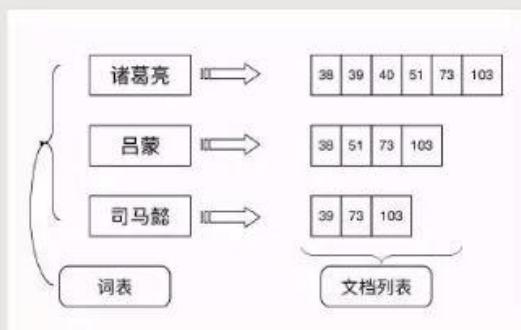


常见的信息检索系统就是基于倒排索引，在 2011 年以后，召回后进行排序时，又多用深度学习技术，比如 Pointwise, Pairwise 等方法。

搜索引擎

倒排索引

倒排索引 (inverted index)，也常被称为反向索引，是一种索引方法。我们已经了解到单词和文档需要通过索引相互映射，索引需要存储每个词被包含的文档的列表。每个文档也使用序号作为标识。



图：词和文档通过索引映射

倒排索引是一个词可能出现在很多文章中，那么就将词建一个列表，然后出现它的文章都建一个 id，然后这些文章按照递增的顺序关联到这个词上。在查询时，根据不同条件得到文章集合，使用归并算法输出。

搜索引擎

Lucene 查询语法

请求 *query* 是支持 Lucene 查询语法的查询条件，由 Lucene 的查询解析器进行解析，下面介绍一些基本的语法。

代码 1：针对字段进行搜索

```
1 title:张飞
2 title:张飞 ~
3 title:张?
4 title:"张飞 AND 关羽"
5 +title:张飞 -post:关羽
6 +title:张飞 OR -post:关羽
7 +title:张飞 AND (-post:关羽 OR -content:刘备)
```

Apache Lucene 是帮助实现信息检索系统的开源项目，它的 query 语法很丰富，性能也很出众。Lucene 的查询语法也给 Chatopera 团队实现聊天机器人的对话引擎很大的启发。另外在搜索时，Lucene 也支持使用近义词，通过简单扩展、简单收缩、简单映射和姻亲扩展让检索更智能。

搜索引擎

相关度计算

```
score(q,d) = ①
    queryNorm(q) ②
    · coord(q,d) ③
    · ∑ (          ④
        tf(t in d) ⑤
        · idf(t)^z ⑥
        · t.getBoost() ⑦
        · norm(t,d) ⑧
    ) (t in q) ⑨
```

图: elasticsearch 中计算相关度

<https://www.elastic.co/guide/en/elasticsearch/guide/current/practical-scoring-function.html>

另外一个知名开源项目 - Elasticsearch 也是面向企业快速搭建信息检索系统的，是实现搜索引擎的非常棒的项目，它的文档相关度计算如上图，它不但使用了 TF-IDF、也支持对一个词进行加权，还根据文档长度进行规范化。

搜索引擎

准确率、精确率、召回率和 F1

准确率 (Accuracy) 代表在给定的测试数据集中，分类器正确分类的样本数与总样本数之比，计算方法如公式 11：

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{P} + \text{N}} \quad (11)$$

精确率 (Precision) 代表实际被预测为正例的样本中，实际值也是正例的比例，计算公式 12：

$$P = \frac{\text{TP}}{\text{TP} + \text{FP}} \quad (12)$$

召回率 (Recall) 又被称为查全率，代表实际正例的样本，有多少被预测正确了，计算公式 13：

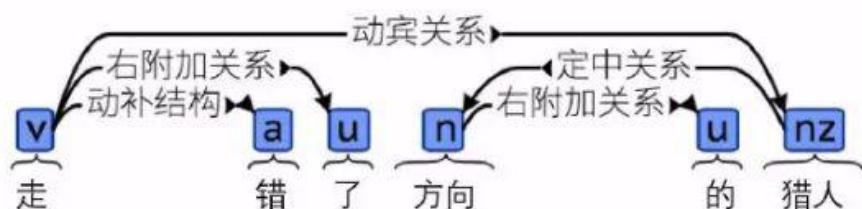
$$R = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad (13)$$

F1 是最常见的一个评测指标，F1 的值越高，代表模型效果越好。

$$F1 = \frac{2 \times P \times R}{P + R} \quad (14)$$

搜索引擎的评测标准有很多指标，比如 MAP、MPP，基于混淆矩阵的准确率和召回率是经典的方法。准确率和召回率是矛盾的，所以，常使用二者结合起来计算的 F1 值评价，F1 值越高，效果越好。

依存关系分析



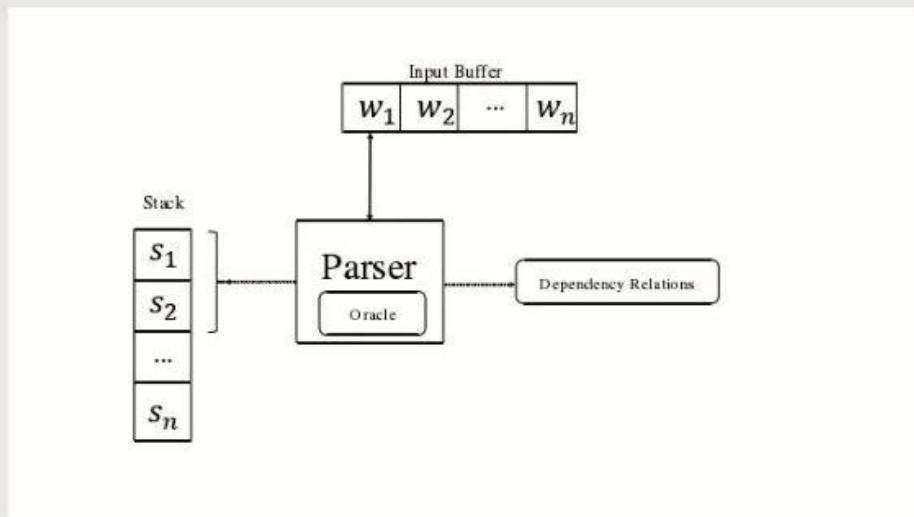
图：依存句法分析

在自然语言处理中，另一个比较难的点就是分析词与词之间关系，在分词后，一个词在句子中充当什么成分。依存关系分析依赖于分词的结果，也依赖于词性标注的结果。标注依存关

系数据集也是很难的任务，词之间的关系会有几十种。我个人认为，从近几年学术上的成果看，依存分析分析已经处于从学术向实际应用过渡的节点了，2017 年在依存关系分析任务上，就出现了准确率 95% 的论文。

依存关系分析

transition-based



图：基于 transition 的依存关系分析实现原理

目前，用于依存关系分析的流行算法有基于 Transition-based 和 Graph-based 两种。Transition-based parser 基本理念就是将待分析的句子放入 buffer 中，然后顺序进入 parser，oracle 是一个分类器，它会给出针对这个词作出哪种行为，比如 SHIFT, 右附加关系，左附加关系。下面是有关依存关系分析的两个开源项目：

```
https://github.com/Samurais/text-dependency-parser # 经典的 transition-based parser
```

```
https://github.com/elikip/bist-parser # 使用神经网络实现 Oracle 的 Parser
```

以上内容是自然语言处理的部分，但是本次分享的主题是“多轮对话提升自动化流程服务”，目前信息检索服务不断向更智能的方向发展，搜索引擎公司越来越了解我们，但是在企业里，随着业务的发展，它更需要的是通过多轮对话完成流程服务，一问一答的服务解决不了太多问题。企业里对业务流程引擎有着极大的热情，业务流程引擎也在几十年的发展中，不断标准化，完善化。比如当前的标准 BPMN2.0 用 50 多个元素去帮助业务人员建模，编排流程服务。结合对业务流程引擎和自然语言处理的理解，Chatopera 开发了**对话引擎**，并基于它开发了**多轮对话设计器**和**智能问答引擎**。

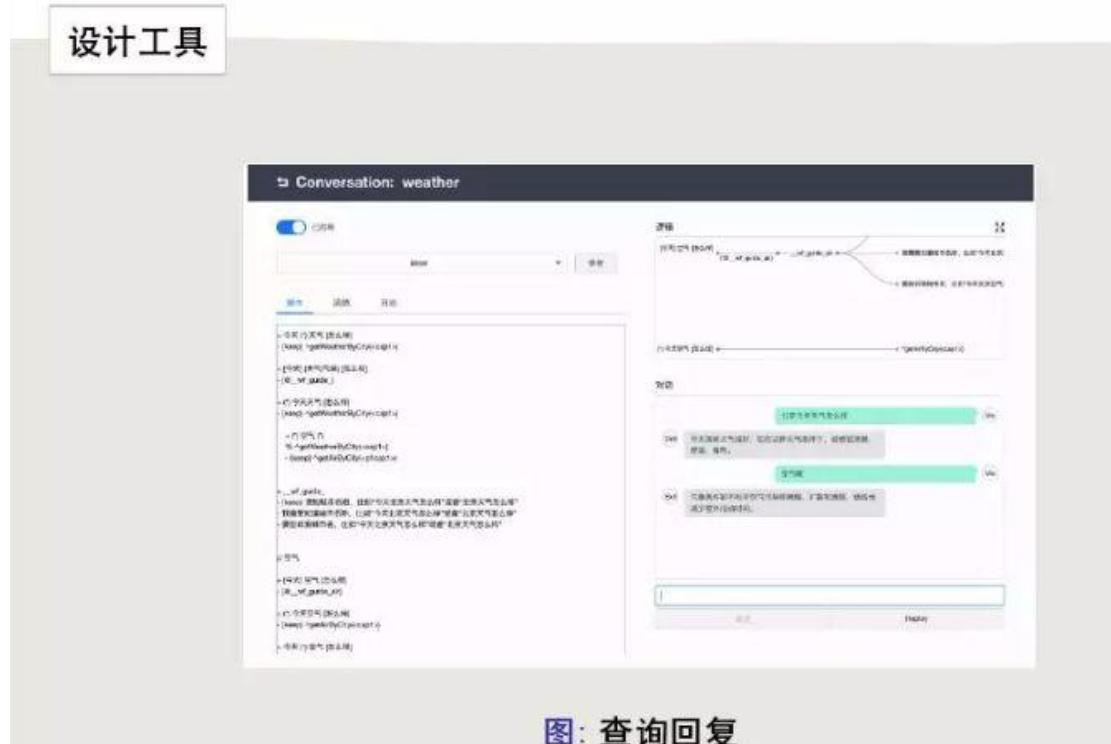
对话引擎

脚本语法

```
13 // 天气
14
15 + 今天 (*) 天气 [怎么样]
16 - {keep} ^getWeatherByCity(<cap1>)
17
18 + [今天] (天气|气候) [怎么样]
19 - {e_wf_guide_}
20
21 + (*) 今天天气 [怎么样]
22 - {keep} ^getWeatherByCity(<cap1>)
23
24     + (*) 空气 (*)
25     % ^getWeatherByCity(<cap1>)
26     - {keep} ^getAirByCity(<p1cap1>)
```

对话引擎是以自然语言为输入的，描述对话的，它根据这些脚本，梳理规则，形成机器人的思维逻辑导图。对话引擎也有自己的一套强大的语法规则，能让机器人更加智能。

对话引擎



上图是多轮对话设计器的对话编辑窗口，左侧写脚本，右脚上是实时渲染的机器人思维逻辑导图，右下角是测试窗口。多轮对话设计器通过**函数**完成系统集成，比如在对话中依赖 CRM 或订单管理数据等，**函数**和**脚本**可以相互调用，这样能满足企业的各种业务需求，并且灵活调整。下面的链接提供一个具体的例子，怎么通过多轮对话设计器实现一个查询天气的机器人。

<https://github.com/chatopera/conversation-sampleapp>

在 Chatopera，自然语言处理和机器学习主要应用于帮助业务人员快速实现聊天机器人，比如客服人员输入“订单下单后怎么查看物流状态”，那么这说话有哪些不同的说法，基于对话引擎，它的规则是什么样的，它的回答有几种可能。也就是，Chatopera 利用先进的技术帮助业务人员写机器人的对话脚本。在 Chatopera 看来，这是企业实现聊天机器人过程中，一个重要的环节。那么，怎么做呢？一方面，要熟悉业务流程引擎的设计理念，比如流程状态机、流程的编排、企业业务人员的工作习惯等，另外一方面，要对信息检索、分词、近义词挖掘、依存关系和机器阅读理解等自然语言处理任务不断研究，持续创新。

今天分享的内容包括一些经典的、成熟的，同时，这些方法也能启发大家创新，比如分词的技术应用于机器阅读理解、Lucene 的查询语法应用于多轮对话技术等。期待于大家进行更深度的交流，我今天的分享就到这里，想要进一步了解 Chatopera 的产品请访问：

<https://docs.chatopera.com/>

作者介绍：

王海良，Chatopera 创始人&CEO 微软人工智能最有价值专家。毕业于北京邮电大学，后加入 IBM 工作四年，先后工作于软件开发实验室和创新中心。从 2016 年开始工作于创业公司，三角兽 AI 算法工程师，呤呤英语 AI 产品负责人，负责智能对话系统研发。具有丰富的项目落地经验，熟悉机器学习，搜索引擎，自然语言处理，业务流程引擎。

开源节流的智能导购对话机器人实践

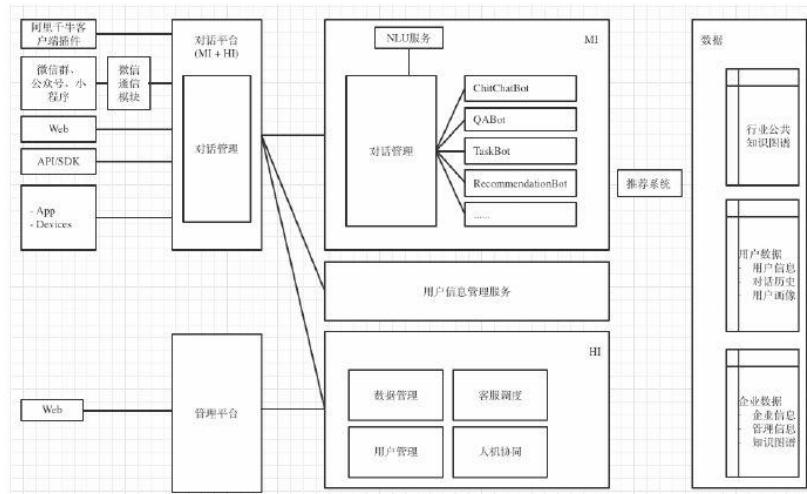
作者：莫瑜 整理：Hoh

首先介绍一下“智能一点”，它是为电商客户提供智能导购服务，比较偏向于售前客服。上面是项目背景，今天主要从以下几个方面讲述，智能导购机器人建设情况，从节流（提升客服效率）和开源（提高客户转化率）两个角度来讲解。



上图是应用场景截图，有单轮交互，如闲聊，单轮推荐。还有多轮交互帮助用户完成一个任务，还有一个交互推荐等内容，后续会详细讲解。

工程师眼中的智能导购对话机器人



接下来讲一下应用后台服务，中间是对话机器人核心部分，主要包含一个中控部分，在接到客户端请求后调用自然语言理解服务对用户输入进行语义理解，然后分发到很多小的机器

人，这些小机器人包括闲聊的、问答的、任务型机器人还有推荐型机器人。由于目前技术原因机器人无法完全取代人，因此采用一种人机协同方式。当机器人无法响应用户请求时，将任务分发，由人工客服去解决用户的问题。对话机器人通过不同的渠道去触达不同的用户，如阿里千牛、微信公众号、web 或者 app。



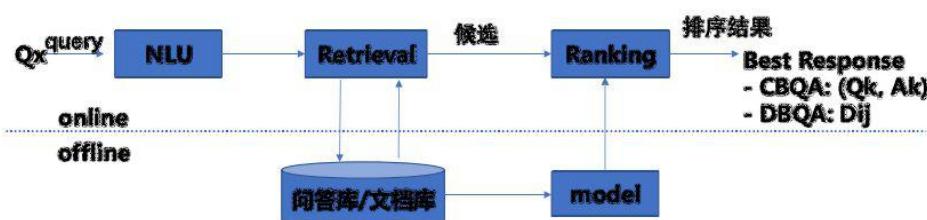
问答导向的对话交互(单轮)

- CBQA/cQA (Community based Query Answering)
 - 已知问答库：(Q1, A1), (Q2, A2), …, (Qn, An)
 - 输入：Qx
 - 输出：(Qk, Ak)
- DBQA (Document based Query Answering)
 - 已知文档库：(D11, D12, …), (D21, D22, …), …
 - 输入：Qx
 - 输出：Dij
- KBQA (Knowledge base Query Answering)
 - 已知知识图谱：Knowledge Base
 - 输入：Qx
 - 输出：Answer

接下来从节流和开源两个方面介绍我们做了那些工作，首先节流方面，做的最多的是问答机器人。单轮问答，暂时不考虑上下文，或者说在多轮交互过程中通过一些处理后再回答用户问题。有三种形式：一种是基于问答库的形式，事先有了很多 Q&A 对，当用户提出一个问题后，如何找到一个答案回复用户。第二种是在文档库中找到一句话回答用户的问题，这个与阅读理解有差异，阅读理解是将文档中的实体回复给用户。第三种是基于知识图谱的形式，构建知识图谱，依据用户的问题在知识图谱中获取相关的信息回答用户。



- 基于检索的问答框架



大部分的智能问答都是基于上述类似框架，首先基于用户的问题进行语义理解，依据 retrieval 模块获取候选集，最后依据 Ranking 模块排序，架构模块与搜索引擎很相似。在 retrieval 和 Ranking 模块也可以用到在搜索引擎中提升 matching 和 ranking 的方法。

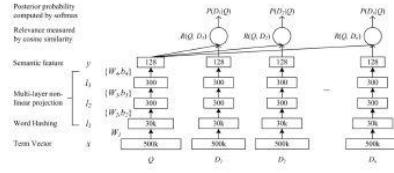


CBQA/cQA

- 问题 : $Q_x \sim (Q_{11}, A_{11}), (Q_{12}, A_{12}), \dots, (Q_{1n}, A_{1n}) \rightarrow (Q_{1k}, A_{1k})$
- 核心问题 : 语义相似度 $F(Q_x, Q_y) = ?$
- 算法

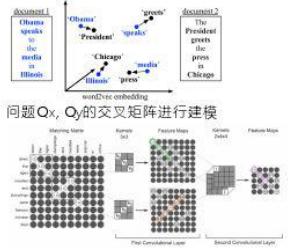
基于语义表示的语义相似度

- $f_i(Q_x, Q_y) = \text{similarity}(\text{Representation}(Q_x), \text{Representation}(Q_y))$
- 语义表示方案
 - BOW
 - 无监督句子向量表示 : Weighted average、skip-thought
 - 监督句子向量表示
 - DSSM、CDSSM、DSSM-LSTM



基于交叉特征的语义相似度

- $f_i(Q_x, Q_y) = f(\text{Interaction}(Q_x, Q_y))$
- 交叉特征方案
 - 文本相似度 : Q_x, Q_y 的 Jaccard 距离、编辑距离
 - WMD (Word Mover Distance, 借鉴 EMD 概念)



问题 Q_x, Q_y 的交叉矩阵进行建模

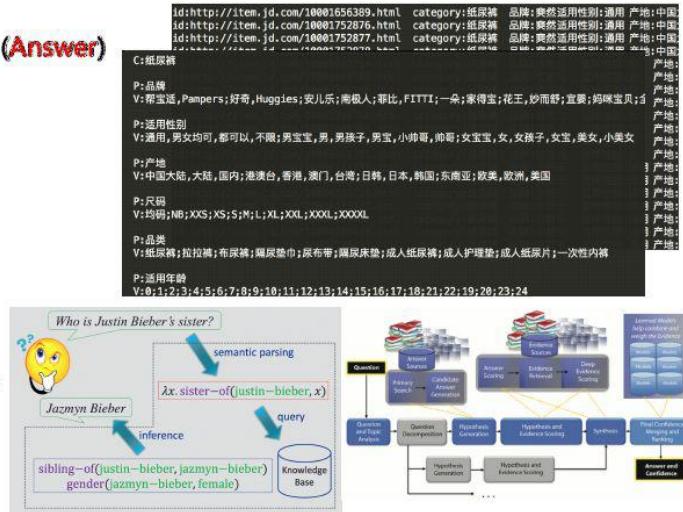
第一种方式基于问答库的智能问答，其核心是给定两个问题，其语义相似度是多少。除去预处理可以从两个维度考虑，首先是要从语义表达的方式来表示问题的含义，然后根据这两个语义表示计算相似度。比如可以将问题转化为词袋模型，也可以使用一些无监督语义表示方法或者有监督语义表示方式。基本上就是将问题转化为语义表示，然后求解语义表示的相似度。第二种是直接建模计算两个问题间的相似度，比如 Jaccard 距离、编辑距离、WMD (Word Mover Distance, 借鉴 EMD 概念)，或者将其转化为图的形式。我们可以结合不同模型最后来解决这类问题，看这两个问题含义是否一样。

第二种就是从文档中找相应的句子来回答，首先我们会建模实现问题的表示和句子的表示，也会采用一些传统特征，比如考虑答案中是否包含地点，考虑问题是否是咨询地点等传统特征，然后结合深度学习的特征再去建立一个模型。



KBQA

- 问题 : Q_x KnowledgeBase \rightarrow (Answer)
- 核心问题 :
 - 知识图谱构建
 - 结构化数据
 - 半结构化数据
 - 非结构化数据
 - 语义解析 Semantic Parsing
 - 句法解析
 - LAT(Lexical Answer Type)
 - 问答
 - 模式1
 - 知识图谱查询 KB Query
 - 知识图谱推理 KB Inference
 - 模式2
 - Deep QA: 检索 + 排序



但是现实应用场景中，用户的提问是多种多样的，比如用户会问很多关于尺码的问题，如果参数展开会是指数级。还有明天有没有活动，今天问和明天问是不一样的。如果使用前面的基于问答库的问答方式无法支持，因此基于知识图谱的问答就显得比较重要。第三种方式一般涉及三类问题，第一个是知识图谱如何构建，怎么从结构化数据、非结构化数据还有半结构化数据中构建知识图谱；第二个是如何理解用户的问题输入；第三个是如何生成回复答案。对于后面两个问题，一般有两个常见解决方案：第一种解决方案是首先对句子进行理解然后将其转化为数据库查询，根据查询结果回答；第二种是 IBM Watson 采用的模式，重点关注问题答案的类型是什么，然后通过检索和排序方法支持。



综合方案

- 问题建模
 - CBQA ($F(Q_x, Q_y), F(Q_x, Q_y, A_y)$)
 - DBQA
 - KBQA
- 数据:
 - 标注数据
 - 未标注数据 (问答数据挖掘)
- 特征: 多维度特征
 - 传统文本特征
 - 深度学习特征
- 模型: LR (融合多个子模型)

我们的解决方案是对前面三种方案进行了融合，比如在做智能问答时，最简单的方案是只考虑问题含义是否一样。其实我们可以首先判断问题含义是否一样，也可用第二种方式，这个答案是否回复了这个问题，因此可以考虑更多的信息。然后利用 LR 模型融合更多的特征。



语义理解 - 意图识别

• 问题建模

- $F(uu) = ?$
- $F(\text{UserContext}, uu) = ?$

• 方案

- 分类问题
- 模型
 - 非参数模型 (kNN)
 - 模型 (LR, CNN, RNN)
- 特征
 - 传统文本特征
 - 深度模型特征

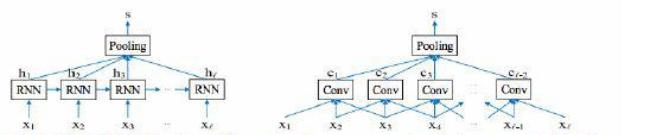


Figure 1: RNN (left) and CNN (right) architectures for generating the vector representation s of a short text $x_{1:t}$. For CNN, Conv refers to convolution operations, and the filter height $h = 3$ is used in this figure.

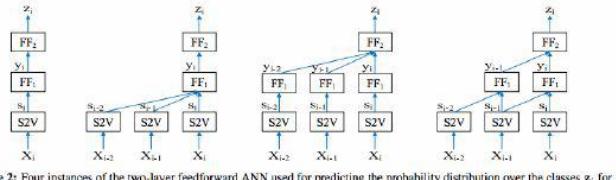


Figure 2: Four instances of the two-layer feedforward ANN used for predicting the probability distribution over the classes z_i for the t^{th} short-text X_t . S2V stands for short text to vector, which is the RNN/CNN architecture that generates s_i from X_t . From left to right, the history sizes (d_1, d_2) are (0, 0), (2, 0), (0, 2) and (1, 1). (0, 0) corresponds to the non-sequential classification case.

Sequential Short-Text Classification with Recurrent and Convolutional Neural Networks, NAACL 2016



语义理解 - 槽值抽取

• 问题定义

- 输入：文本(Text)
 - 如：我家宝宝2个月，穿多大码纸尿裤？
- 输出：(槽, 槽属性值) (Slot, Slot Value)
 - 如：Informable slots: age = 2个月, productType = 纸尿裤, Requestable slots: size

• 方案

- 基于规则
- 基于序列标注

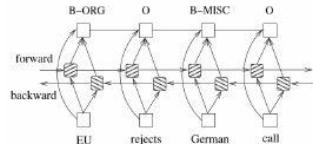
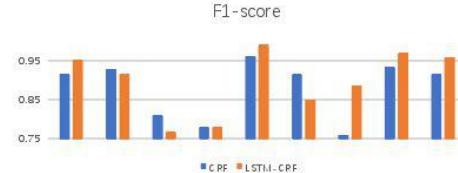


Figure 8: A BI-LSTM-CRF model with MaxEnt features.



其实很多情况下是多轮对话，如果要支持多轮对话核心是如何维护对话状态。需要结合上下文理解用户意图是什么，使用两大类方式，一种是非参数模型，基于近似数据进行标注；再一种就是使用模型的方式。前者可控性更强，后者泛化性更强。在应用出现的问题有很多，比如历史训练数据是系统还未上线前的交互数据，上线后用户对话情景会发生变化。实体抽取有两种方式，一种是基于规则方式，另一种是基于序列标注。

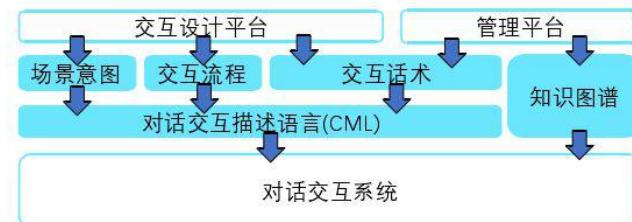


Frame based architecture

- 对话状态定义
 - 槽值组合
- 状态管理(Dialog State Tracking)
 - 对话交互设计
- 策略优化(Dialog Policy Learning)
 - 对话交互设计
- 文本生成(Natural Language Generation)
 - 对话交互设计



- 设计和实现解耦



对话管理一般分为两大类，一种是预先自定义的图架构，通过事先自定义进行维护，对话的状态可以用 id 来定义，每个节点就是一个状态，每个节点的转移或者要回复的文本都是系统预先定义好的。这种方式系统难维护，尤其状态较多时。第二种是 Frame based 架构，将状态表示为槽的组合，策略状态优化、文本生成等都是通过相关配置完成，从而实现设计与现实的解耦。比如导购机器人，开发人员并不是一个好的对话机器人设计师，导购专家可以通过这个平台实现他们的想法。目的是让导购专家无需了解技术实现面向用户体验的平台设计，交互平台支持配置要实现哪些意图、如何交互以及交互方式是怎样。管理平台维护商品活动信息，所有的相关信息都会进入知识图谱形成图谱。设计统一的描述语言，要求表达能力强，对话交互系统依据描述语言和知识图谱进行最终交互。技术人员着重解决如何构建表达能力强的描述语言，设计人员着重于提升用户体验。电商存在潮汐现象，平常和活动差异很大，在平常对导购机器人有更高的需求，需要如何提升客户转化率。

- 为什么需要个性化推荐 ?

- 对话交互有利于信息的获取,不利于信息的展现
 - GUI > CUI
 - 信息展示、操控精度
 - CUI > GUI
 - 信息获取
- 对话交互需要细粒度的个性化

提升客户转化率不单单帮用户省钱而且还帮其赚钱 , 如何将交互和推荐结合来提高转化率。在 CUI 之前 , GUI 比 CUI 有很多优点 , 比如 : GUI 信息展示更好 , GUI 操作精准度更高 , 基本 100% , 但是 CUI 很难做到百分之百。但是 CUI 比 GUI 更容易信息获取 , GUI 服务于头部信息 , 对于长尾信息很难获取 , 比如在在支付宝找免密支付是很困难的。从用户角度是要结合两者的优点。另外 , 对话交互场景的个性化需求很强。综合来看 , 将个性化推荐融入对话场景中显得非常重要。



- 对话交互推荐的独特问题?

- 传统推荐系统

- 多依赖于隐式反馈 , 如 : 浏览、点击、收藏、购买、评论等用户行为
- 不(少)考虑时间、地点、场景、情绪、活动状态等上下文
- 问题建模
 - $F(\text{User}, \text{Item}) = ?$
 - $F(\text{Context}, \text{User}, \text{Item}) = ?$

- 对话交互推荐系统

- 结合更丰富的信息 (包括显式反馈)
- 更多地考虑上下文 , 如 : 时间(早中晚 / 星期)、地点、情绪、环境 (商家情况) 等



智能导购中的交互推荐(1)

- 热门推荐
- 相关推荐
- 主动推荐
 - 提醒 / 催单
 - 商品 / 活动推送
 - 优惠券推送



在对话交互过程中做推荐有什么不同。传统推荐很多依赖隐式反馈，比如在看今日头条、抖音等，会根据你的浏览、点击、购买等信息来判断是否对某一产品感兴趣。但是在对话交互过程中可以结合更多丰富的信息，可以获得一个显式反馈，比如去餐馆吃饭，服务员问你要吃什么，这是一个获得显式反馈的。在导购过程中的推荐有：热门推荐、相关推荐，还有主动推荐，比如提醒/催单，还有最近相关活动推荐，还有优惠券推荐。比如客户买了 90 片 S 码纸尿裤，使用 S 码纸尿裤的小孩的纸尿裤每天用量大，那么，我们可以在第七天或第八天对他进行纸尿裤优惠券推荐；而对于 L 码时，就要在 30-40 天之后再发起相关推荐和优惠券推送。



智能导购中的交互推荐(2)

- 交互推荐
 - 双向信息交换
 - 商家信息
 - 用户偏好
 - 一种形式
 - Q20
 - 迭代过滤：推荐 - 交互 - 再推荐



在对话过程中其实信息是不对称的，因此通过对话交互方式实现信息交换。比如用户并不知道商家有什么信息，商家也不知道用户有什么偏好，因此需要如何通过信息交互弥补这种不对称，促成交易。主体上采用一种 Q20 方式，首先给用户一个推荐，然后进行交互再推荐。

比如用户想买一个手机，User: 我想买个手机，有什么推荐？， Bot: 亲，你想看什么价位的？， User: 1000 以下吧， Bot: 这个价位的手机，有华为，小米，魅族，你想看哪个品牌的？， User: 小米的吧，Bot: 推荐你看看红米(Url)和红米 Pro(Url)。



- 基本流程

- Set constraint = {}
- While (condiction)
 - Recommend based on constraint
 - Query attribute selection
 - NLG – User response - NLU
 - Set constraint = constraint + new constraint

- 商品类型

- 确定型：Apple iPhone X (A1865) 64GB
银色 移动联通电信4G手机
- 可配置：组装商品
- 规划型：自驾游计划

- 显式反馈

- GUI：评分 / 赞、偏好选择、对比选择、答题
- CUI：属性限制

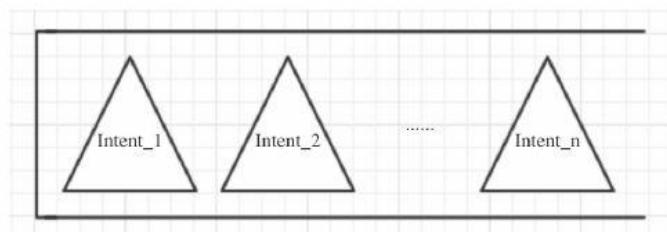
- 问题

- 如何提升交互推荐效率？
 - 选择信息量最大的问题？
 - 静态 / 动态策略？
 - 个性化、引入用户画像

基本流程是开始没有任何限制，然后在没有满足终止条件前，基于限制做一个推荐，第二步，选择下一次交互的参数，需要交互什么东西，然后生成用户回复，经过语义理解发现一个新的限制，然后依据新的限制进行结合不断进行迭代。在商品类型中可以有不同类型，最简单是确定型商品，但是有不同属性，也有可能是一个组装产品，包含不同部分。还有就是反馈部分，以前 GUI 就是通过点赞/评分、偏好等，CUI 就是依据属性咨询。在这其中有一些问题是影响转化效率的，我们如何去和用户推荐，如果交互太多，用户会反感，你为什么问我这么多问题，选择信息量最大的做推荐，可以采用静态策略和动态策略，静态策略就是对所有人先问 A 属性再问 B 属性，动态策略比如对于 A 不同的反馈采取不同的行为。还可以考虑个性化，如一个人每次来都买某个品牌的产品，就不需要再沟通他的品牌偏好了。

- Discourse Goal Stack Model

- 维护所有未结束意图对话状态（下图以三角形表示）
- 对用户输入进行意图识别
 - 如果为新意图，栈内压入新意图对话状态
 - 如果为老意图，根据意图对话设计交互，并在交互结束后出栈



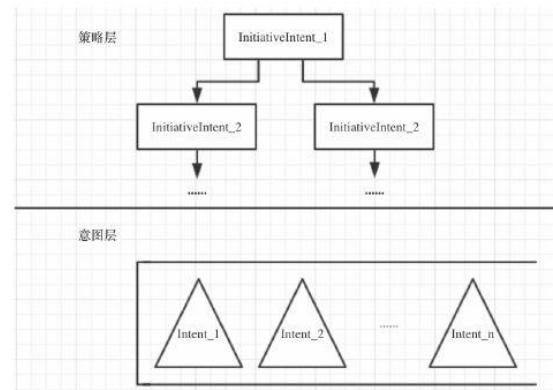
如 “User: 刚出生的宝宝，有没有什么纸尿裤推荐？， Bot: 咱们家纸尿裤有好几个系列，您是要超薄的，还是超柔的？， User: 超薄和超柔有什么不同？• Bot: 超薄...，超柔...，User: 看看超薄的？， Bot: 推荐...” 对话会发生返回，这种情况很普遍，这其实是多个意图的交叉或偏移，解决方案就是上图三角形是交互平台如何去维护一个场景活动，可能有很多没有结束的产品，在用户输入时判断是否是新的意图，新意图就创建新的场景，将新场景加入堆栈中，如果是老场景，判断是哪一个老意图，然后去交互，完成交互之后将对应场景从栈里拿出，这种方式支持不同意图切换。还有情况是交互过程中无法回复，这种情况是在交互过程中避免说的太过于技术语言。这种涉及到评论挖掘，抽取用户对商品的标签或描述。



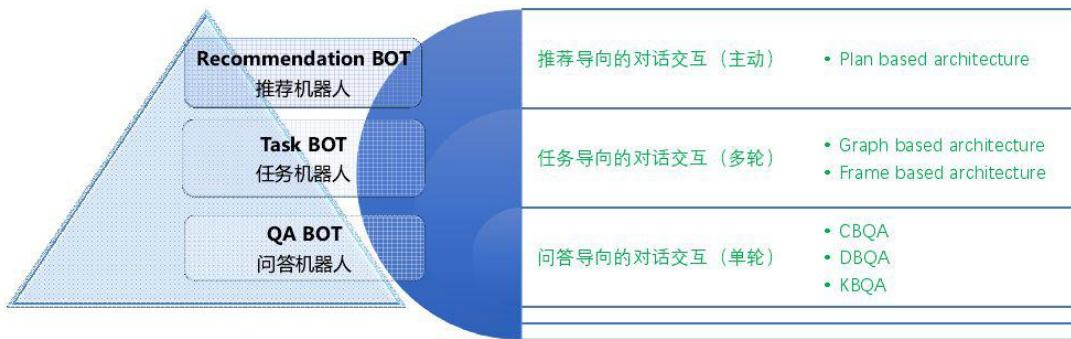
- 主动发起 + 被动对话

- while (!终止条件)

- 根据策略层发起主动意图对话
 - while (用户回复处于主动意图)
 - 根据主动意图流程设计交互
 - while (用户回复处于被动意图)
 - 根据被动意图流程设计交互



很多情况是用户发起，但是有时候是需要机器人发起，如 “Bot: 最近咱们家上线了一款新产品，您要不要了解一下？• User: 你们家用什么快递？• Bot: 通过天猫商城菜鸟仓随机快递发货哦。• Bot: 新款产品，了解下？• User: 好啊• Bot: balabala• User: balabala” ，这种解决方案就是分为策略层和意图层。机器人依据一定策略发起对话，发起主动意图，当用户回复在主动意图里面，按主动意图脚本执行，如果用户意图不在主动意图里面，依据被动意图去交互。被动意图交互完依据一定策略继续主动意图，通过这种方式实现机器人主动发起和被动对话的结合。



最后小结一下，我们将导购机器人分为三个机器人，最简单是问答机器人——单轮情况；在此基础上实现多轮对话，多轮对话会依据一定策略转化为单轮情况；在多轮基础上实现主动对话和交互推荐。依据这三个机器人不同的三个属性（单轮、多轮、主动）实现不单单帮用户节省人工还能提升转化率的目的。

作者介绍：

莫瑜，智能一点科技有限公司联合创始人 & CTO。中山大学硕士，曾任职于微软搜索技术中心，海豚浏览器。先后参与 Bing 搜索引擎技术研发、支持千万曲库的音乐检索和哼唱搜索算法研发、带领团队研发支持多国语言的信息流推荐系统等。畅销书《编程之美》作者之一。

音乐垂域的自然语言理解

作者：秦斌 整理：Hoh

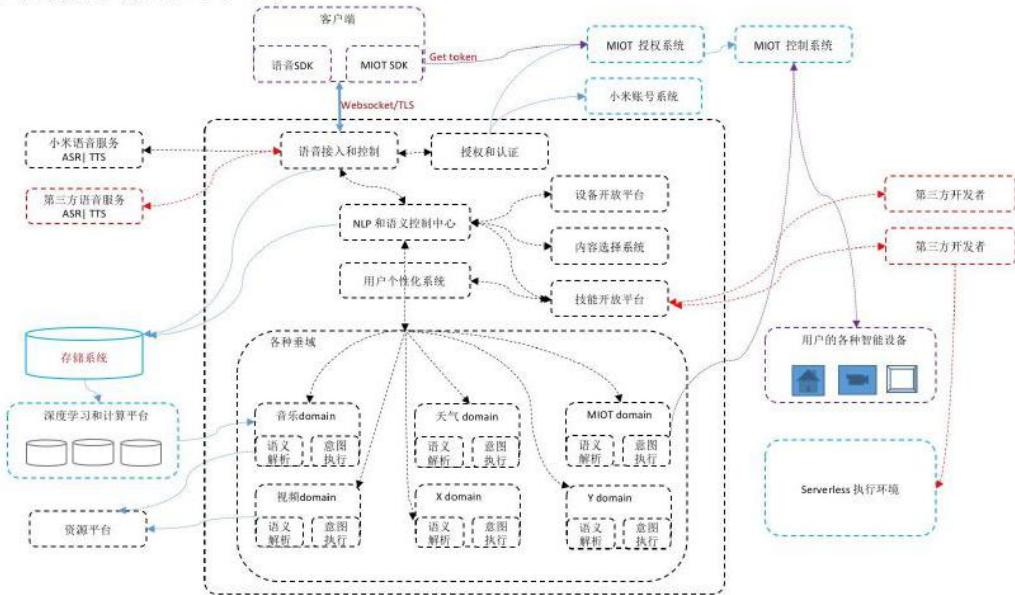
▶ 概括



- 背景
- 功能介绍
- 问题挑战
- 音乐理解
 - 意图抽取
 - 知识库
 - 字段抽取
 - 路径选择及打分
 - 线上用户反馈

今天分享的内容有项目研究背景、实现了那些功能，在做音乐领域时有哪些独有的问题与挑战，还有就是“小爱”项目具体的实现。

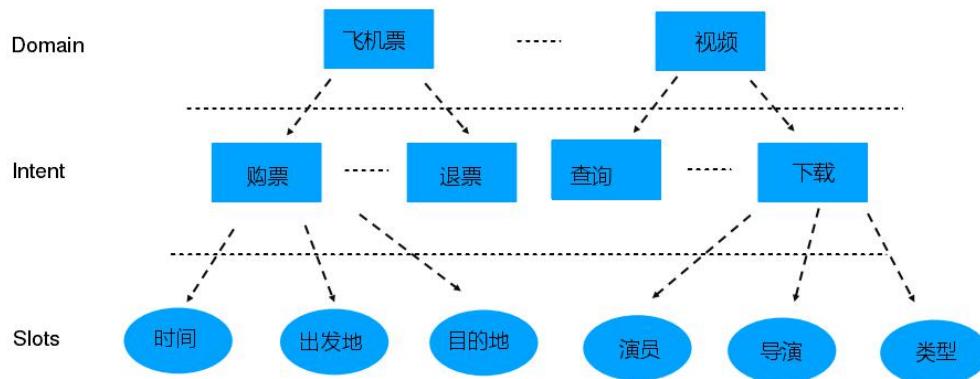
▶ 小爱语音交互平台



上图是整个小爱语音交互平台的后台服务架构，小米大脑定位的是一个平台，能够处理各种数据。在最外部给各种厂商封装了 SDK 接口，目的使厂商能够很快的接入，降低你接入成本，如果你要操作小米相关智能设备就要 MIOT 授权验证。后续会有小米语音服务 ASR，语音识别都在客户端，由于平台特性，在云端接入 ASR 厂商，如微软、百度、科大讯飞、猎狐星空等，部署于云端便于控制和优化，可以额外做一些文本选择等功能。语音转化为文本就会进入 NLP 模块，在 NLP 中控部分会做一些个人训练计划、公共训练计划、还有一些 query 概率，然后将其发布到精品垂域，采用分而治愈的思想，每一个垂域将这个领域的语

料、知识、常见说法给建立起来，由中控选择最终的垂域。最外部有一个设备开放平台 oivs，方便各种硬件设备厂商接入。后续还有一个技能开放平台，第三方技能开发者能够在平台上很简单的实现一个技能，如打开成语接龙或闲聊，将 query 转给第三方技能。周边就是机器存储、机器学习平台等资源平台。

► 垂域语义理解



接下来介绍下，一个垂域要做那些事情。如飞机票垂域，要理解用户的意图，是要购票还是退票，音乐就是你要找歌手、听歌、还是找一个推荐；在意图理解后抽取 Slot（一个个的字段），如时间、出发地、目的地。整体就是将一个纯文本机构化，将相关信息抽取出来，这是做语义理解经常要做的事情。

► 音乐基本功能-搜索意图

- 从Query中抽取“歌手/歌名/专辑/标签”四类字段(slot)信息
 Q: 我想听周杰伦的歌 A: artist=周杰伦
 Q: 来一首周杰伦的简单爱 A: artist=周杰伦 song=简单爱
 Q: 播放范特西里面的简单爱 A: album=范特西 song=简单爱
 Q: 播放英文版柠檬树 A: song=柠檬树 tag=英文
- 对“歌手/歌名/专辑/标签”字段的字段消歧
 Q: 三生三世十里桃花的歌 A: album=三生三世十里桃花 (优先播放song:凉凉)
- 对“歌手/歌名/专辑/标签”字段的“同音/补全/乱序”纠错
 Q: 播放李文的歌 A: artist=李玟
 Q: 播放大王叫我 A: song=大王叫我来巡山
 Q: 长江滚滚东逝水 A: song=滚滚长江东逝水
 Q: 韩磊南山南 A: artist=张磊 song=南山南

接下来介绍下音乐领域实现了那些功能，第一个就是用户的个性化推荐，如随便放首歌、歌单等。再往后就是一个搜索意图，比如我要听周杰伦的歌，周杰伦的简单爱，抽取“歌手/歌名/专辑/标签”四类字段(slot)信息。字段消歧，如“三生三世十里桃花的歌”，其实这是一个专辑，同时也有首歌叫三生三世十里桃花，通过用户原始信息知道应该是专辑而不是歌名。ASR 不可能完全准确还有用户发音问题，因此需要纠错，纠错太多召回存在问题，

一言不合就放歌，把握不好就会觉得你太笨，对“歌手/歌名/专辑/标签”字段的“同音/补全/乱序”纠错。如歌手同名问题，歌词错误，歌名与歌手对应错误等。

▶ 音乐特色功能



- 音乐意图的上下文继承,对字段信息进行信息补全或指代消解
Q1: 我想听刘德华的歌 Q2: 播放他的笨小孩
A: artist=刘德华 song=笨小孩
- 用户情感分析,识别对指定字段的否定意图
Q: 我想听岁月神偷不要听金玟岐的
A: artist!=金玟岐 song=岁月神偷
- 识别指定播放模式(顺序/随机/单曲循环)的听歌的意图
Q: 循环播放刘德华的歌 A:artist=刘德华 mode=顺序
Q: 循环播放笨小孩 A:song=笨小孩 mode=单曲循环
- 根据歌词内容识别歌曲
Q: 海草海草 A: song=海草舞
- 收听历史
Q: 昨天听过的歌 Q: 我听过的儿歌

还有一些音乐特色功能：(1)音乐意图的上下文继承,对字段信息进行信息补全或指代消解。如我想听刘德华的歌，完了又说要播放他的笨小孩，那就是“刘德华的笨小孩”；(2)用户情感分析。如“我想听岁月神偷不要听金玟岐的”，就是对Artist的一个否定；(3)指定播放顺序；(4)根据歌词内容识别歌曲，如“海草海草”识别到是“海草舞”；(5)收听历史歌曲，需要提出时间信息。

面对的主要挑战



▶ 实体名的复杂性

- **实体名形式多样**：实体名数量众多且形式自由，没有固定、清晰的组成规则，尤其是歌曲名
- **知识库数据量大**：音乐领域历史悠久，涉及古今中外的庞大的知识数据，QQ音乐有1500w，网易云音乐500w，阿里系有300w。
- **要求高质量的知识库数据**：原始数据存在着大量的噪音，缺少字段信息。如何建立完整的音乐知识库，清洗数据，打上准确的标签等都是繁杂有挑战的工作。

▶ 半结构化文本

- **输入文本形式多样**：允许输入形式为半结构化的文本，不完全符合自然语言规则，比较简短，没有固定形式

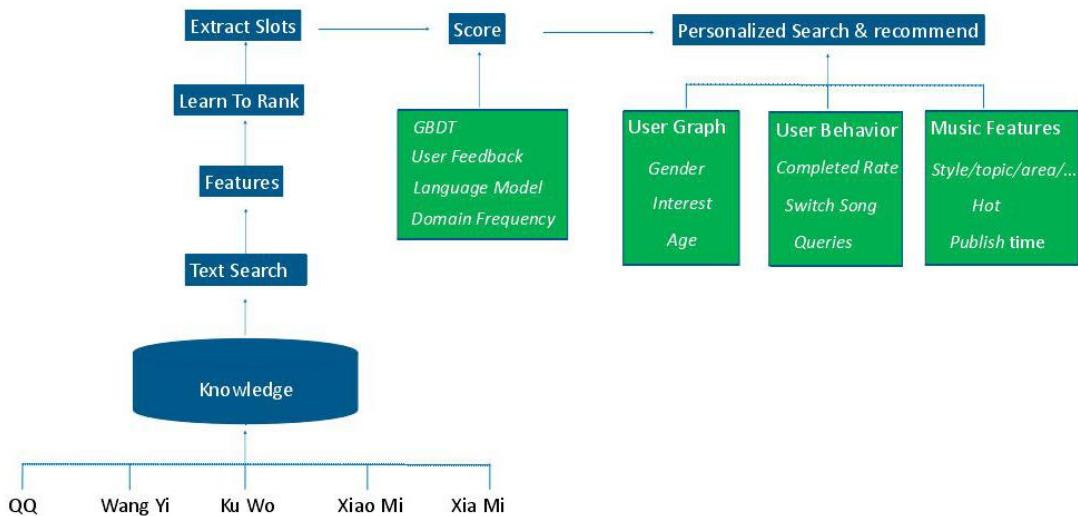
▶ 名实体纠错

- **纠错形式多样**：由于实体名自身的复杂性和多义性，存在着同音纠错，方言纠错，乱序纠错等多种纠错情况

上面介绍音乐要实现的功能，接下来介绍遇到的挑战。首先(1)实体名太过复杂，形式多样，没有固定的组成规则；知识库数量巨大，如QQ音乐是千万级，网易云音乐数百万，阿里系也有数百万。由于歌数量巨大，很多都垄断，要识别这些歌曲需要建立知识库，但是原始数据存在很大的噪音，缺少字段信息，歌名不规范。需要将海量数据爬取下来，判重。

(2) 用户说法很乱 , 不符合自然语言 , 比较简单 , 没有固定形式 , 半结构化文本。 (3) 实体名纠错。由于实体名自身的复杂性和多义性 , 存在着同音纠错 , 方言纠错 , 乱序纠错等多种纠错情况。

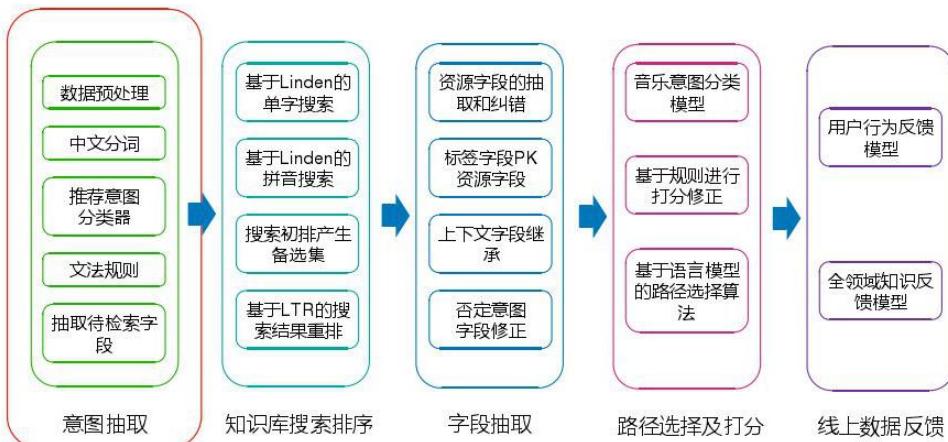
► 音乐理解



知识库很多 , 实体名很复杂 , 单纯用词表不能很好地解决这个问题 , 采用了知识库加搜索的方案。搜索能很容易解决数百万知识库 , 不存在性能问题 , 搜索排序算法技术比较成熟 , 搜索保留了歌的物理信息 , 如 “王菲红豆” 就能知道是红豆这首歌 , 但是如 “韩红红豆” 就不是歌了。歌来源有爬取的也有合作的 , 进入知识库后会有一个文本搜索 , 抽取 feature , 利用 learn to rank 排序 , 通过 query 确定用户想听那首歌 , 确定歌名 , 抽取 slot , 利用 GBDT 模型打分。还有 User feedback , language model , domain frequency (利用用户行为反馈 , 改善向量效果) 。后面就是个性化的搜索和推荐 , 如用户的性别、行为 , 歌一放出来就切换 , 还有音乐的热度、风格、发行时间等。大致流程会对 query 做一些前置理解 , 后续从知识库中找到与之相关的歌 , 抽取 slot , 然后进行语义消歧。然后打分 , 然后利用语言模型判断是否符合常用规则。



► 音乐NLU整体架构



资源字段：指歌手，歌名，专辑名称这三类属性信息

数据是核心，资源主要来自资源方、垂直网站还有人工运营平台。获取数据后对数据进行归一化，做相应的映射，打一些标签。还要排重，一家网站一首歌也会存在很多版本，但是我们只需要原始数据忽略版本。后续就进行 DB、内容评审、构建索引等，清洗数据会花费大量时间。

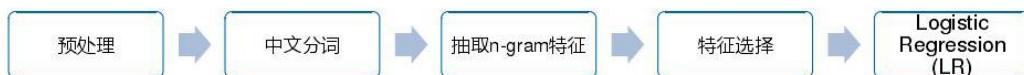
音乐 NLU 整体架构，分为意图抽取、知识库搜索排序、字段抽取、路径选择及打分、线上数据反馈。意图抽取会对 query 进行预分析，数据预处理，然后中文分词还有推荐意图分类器，还有文法规则，最后找到一些主干 query 供后续搜索，先理解 query 语义上的可能的一些倾向性的方向判断，找到主干 query 供第二部使用。



► 意图抽取-推荐意图分类算法

- 意图分类目标：
 1. 识别出音乐泛推荐意图
 2. 识别出包含曲风，场景等标签信息的指定音乐内容推荐意图

- 基于传统机器学习方法



- 基于深度学习的方法（待上线）
 1. Word Embedding：word2vec 能够捕捉更多语法和语义特征
 2. Convolutional Neural Network(CNN)

接下来介绍一下推荐意图分类器，在获取 query 时，增加了一个二分类判断，判断是否是推荐意图，目的是简化搜索，节省时间，同时为策略判断给出一些特例。具体做法是中文分

词后抽取 n-gram 特征，如单词特征、二元特征、三元特征，最后利用逻辑回归。后续会使用 Word2vec，因为 word2vec 能够捕捉更多语法和语义特征。

▶ 意图抽取——文法规则匹配



意图文法	专辑类文法	标签文法	特殊文法
<input type="checkbox"/> 正则句式	<input type="checkbox"/> 影视类关键词	<input type="checkbox"/> 通用标签匹配句式	<input type="checkbox"/> 排除类文法
<input type="checkbox"/> 意图词+意图分类	<input type="checkbox"/> 专辑类关键词	<input type="checkbox"/> 强标签匹配句式	<input type="checkbox"/> 信息查询类文法
<input type="checkbox"/> 切词白名单	<input type="checkbox"/> 专辑规范化文法	<input type="checkbox"/> 标签趋势句式	<input type="checkbox"/> 黑名单文法
<input type="checkbox"/> 停用词文法			<input type="checkbox"/> 白名单文法

接下来就是文法规则匹配，自定义了一套文法规则，支持变量的定义、文法的组合、文法替代，目的是对 query 语义做倾向性判断。还有影视类、标签类的词典，标签文法对标签进行了强分类（强标签、弱标签）。

第二部分局势知识库搜索排序，基于 Linden 做的搜索。Linden 是基于开源的 Lucene 包，Linden 就是将其服务化，集群管理，加入类似 SQL 语言，便于集群查询。单词搜索主要针对用户说法很乱，如果进行切词可能无法召回，但是出现的问题是召回太多。最后会对初排后结果 learn to rank，进行重排。

▶ 知识库搜索排序——索引建立



知识库内容维护	知识库索引设计	知识库搜索初排算法
600万+数据 每周评测更新索引 人工审核 top query	支持歌手、歌名、歌手别名、专辑和影视类字段的综合搜索 支持单字和拼音两类搜索方式	连续命中匹配加权 支持特定模式词匹配降权

知识库中有数百万的数据，每周都需要评测更新一半的索引，上线有严格的流程，线上的策略改动、数据改动都要经过评测才能上线，还有人工审核 top query。知识库索引设计支持歌手、歌名、歌手别名、专辑和影视类字段的综合搜索，支持单字和拼音两类搜索方式。知识库搜索初排也会支持降权，在搜索时就进行降权。

▶ 知识库搜索排序——排序算法



问题描述和模型选型	模型特征	算法提升效果
<p>给定搜索备选集合下的排序问题，相关文档数量为 1</p> <p>正负样本比例失衡，适用于 Learn to rank 算法</p> <p>LambdaMart</p>	<p>字段匹配类别、相似度 字段匹配长度 字段纠错类别 文档热度 50多个相关特征</p>	<p>替代繁杂的人工规则逻辑，同时提升排序稳定性 属性准确率提高一个点 有效解决字段歧义问题</p>

在初排后进入中间层会利用排序算法进行二次排序，如从 top80 里面选择一个最相似的文档。正负样本比例失衡，适用于 Learn to rank 算法，最后选用 LambdaMart，采用决策树模型会学习到很多特征组合。模型特征有字段匹配类别、相似度，字段匹配长度，字段纠错类别，文档热度等 50 多个相关特征。

▶ 字段抽取模块



50多个相关特征

超过**10**次迭代的规则优化

标签体系：**3**个级别，**13**种级别

数十条阈值分支判断

上百条优先级判定规则

通过相关特征选择一首歌后，进行字段抽取。涉及资源字段的抽取和纠错，标签字段 PK 资源字段，如判断青春是歌手还是歌名，上下文字段继承，否定意图等。字段抽取是基于纯规则，有些热门歌曲唱错时也会将其选择出来，如一个歌手没唱过这首歌，但是如果是热门歌曲也会选择出来。



▶ 路径选择及打分模块

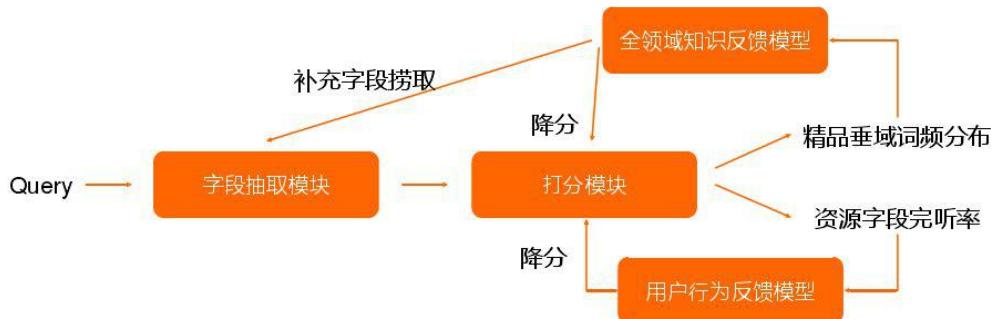
问题描述和模型选型	模型特征	算法提升效果
典型二分类问题，缺少其他 domain 打分信息	字段匹配长度、相似度、纠错类别	替代繁杂的人工规则逻辑，同时提升排序稳定性
结合少量必要规则，替代原有规则系统	字段文档热度	属性准确率提高近1个点
最终选定 GBDT 算法	语言模型特征 歌手职业特征 APP 搜索日志重要性特征	通过样本生成策略，增强了模型鲁棒性

打分是基于意图分类和规则打分，还有基于语言路径的用户选择。由于语言的歧义性可能会选择多条路径，典型二分类问题，缺少其他 domain 打分信息结合少量必要规则，替代原有规则系统。最终选定 GBDT 算法，利用各种特征，如职业特征，郭德纲、岳云鹏都唱过歌，但是职业是相声演员，单说歌手就会将其推电台。还有一个优势就是 APP 搜索日志，小米音乐大都是 key-words 搜索，如果在日志中找到也是属于音乐特征。

▶ 线上数据反馈模块



- 用户行为反馈模型：利用线上的低完听率数据，区分有歧义的实体名
- 全领域知识反馈模型：利用线上各垂域的词频分布信息，辅助纠错



文本搜索有用户点击信息，但是语音很难找用户行为特征信息，尤其命令型语音，得不到用户反馈。但音乐可以获取用户听没听这首歌，听了多久。利用的是全领域知识反馈模型，利用 DFTF 思想，如“大王叫我来巡山”在音乐领域用的很多，当出现大王，就很容易召回。另外还有完成率，就是用户完成行为反馈模型，用户听了多少歌，首条完成率，即第一首歌听完的概率，还有 30 秒切歌率等指标。



▶ 当前主要问题和挑战

- 音乐的过召回问题
- 音乐slot抽取准确性仍需提高
- 知识库的准确性和完整性存在不足
- 聊音乐
- 端到端

项目目前存在的问题，音乐过召回严重，音乐 slot 抽取准确性仍需提高，知识库的准确性和完整性存在不足，聊音乐，想做电台式音乐，这样显得智能，一首歌结束会引出小爱的评价，引出下一首歌。端到端，利用 click model 训练端到端模型，知道一些 query 在历史上选为音乐意图，当一个新 query 来后与历史 query 比对，如果词向量相近，直接返回结果，简化操作。

作者介绍：

秦斌，小米智能云小爱智能应用组负责人。毕业于武汉大学，曾就职于人人网、百度，目前在小米智能云工作，负责小爱同学垂直领域的自然语言理解、数据挖掘、功能建设。



[关注社区公众号 : DataFunTalk , 后台回复【沙龙年货】 , 下载沙龙 PPT+Video 资料合集。](#)

或识别下方二维码 , 查看 DataFun 沙龙年货合集文章 , 了解详情 :

