链滴

闲鱼是怎么让二手属性抽取准确率达到 95% + 的?

作者：zhaozhizheng

原文链接：https://ld246.com/article/1614066561276

来源网站：链滴

先上效果

null

图 1 - 二手属性抽取算法效果 Demo(个护美妆)

背景

闲鱼作为一款 C2X 的 app，站在商品发布的角度，闲鱼商品相对于淘宝商品的特点有：

- **轻发布导致商品信息不足**

闲鱼采用图文描述的轻发布模式，迎合了用户快速发布的体验，但也导致了商品结构化信息不足的问。如果平台希望更理解商品到底是什么，就需要算法去识别用户描述的图片和文本。

- **商品具有独特的二手属性**

- 不同于淘宝新品的一手属性（例如品牌、型号、规格参数等），二手属性指的是在商品入手一段时后，能够反映商品折损/保值情况的属性，比如商品的【使用次数】、【购买渠道】、【包装/配件是完整】等。

- 不同类目有该类目独特的二手属性，比如个护美妆有【保质期】，手机有【屏幕外观】、【拆修情】，服装类有【是否下过水】等。

问题和难点

二手属性抽取在 NLP 领域属于信息抽取（Information Extraction）问题，通常的做法是拆解为命名体识别（NER）任务和文本分类（Text Classification）任务。

二手属性抽取任务的难点有：

- 不同的类目、不同的二手属性/属性簇，需要构建不同的模型。
- 如果使用有监督学习（Bert 家族），打标工作会非常的繁重，开发周期会变得很长。

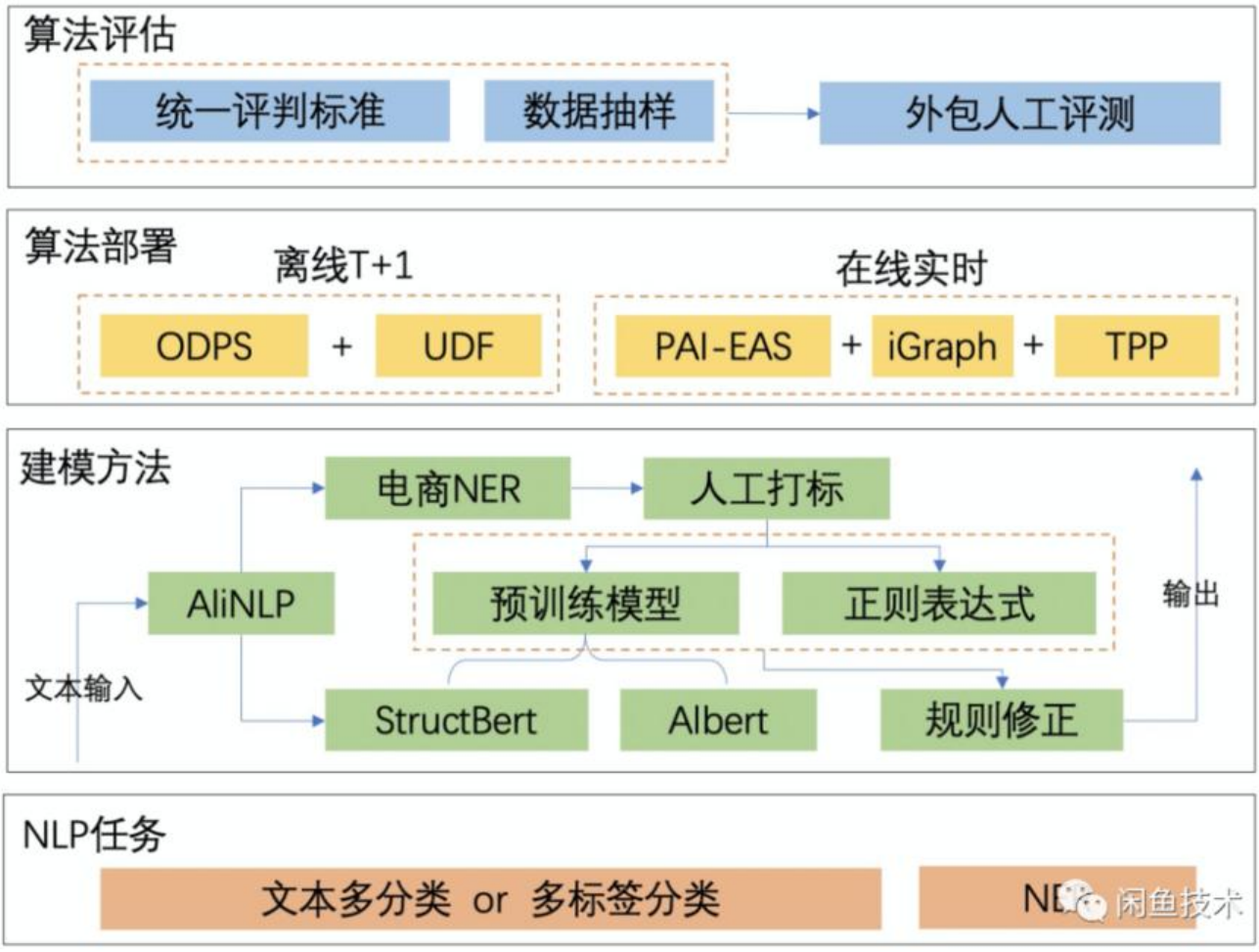
解决方案

方法论

在当今 NLP 环境，依旧是 Bert 家族（或 Transformer 衍生的各种算法）大行其道，霸榜 GLUE、CLE 等各大 NLP 榜单，信息抽取任务也不例外，所以笔者在本方案中的某些场景也使用了 Bert 家族。过笔者认为，没有一种算法在各种场景下都是全能的，只有在给定领域、指定场景下最适用的算法。外，笔者总结了自己的一套属性抽取的方法论：

- 句式相对固定，或者句式受模板限制，如文本描述模板是典型的时间+地点+人物+事件（某事某地人做了啥事），用 NER，建议方法：CRF、BiLSTM+CRF、Bert 家族、Bert 家族+CRF 等。
- 句式不固定，但领域/场景关键词相对固定，或者有一些关键词模板、俗称、行话等，用文本分类：
- 近义词、近义表述不是特别多的情况（≤几十种到上百种），关键词呈对数正态分布/指数分布（即很多高频且集中的关键词），建议方法：正则表达式+规则。
- 近义词、近义表述非常多的情况（≥几百种到上千种），典型的如地名识别，建议方法：用 Bert 家。
- 句式和词语都不固定，典型的如社交评论/聊天的情感分析，建议方法：用 Bert 家族。

方案架构



原文链接：闲鱼是怎么让二手属性抽取准确率达到 95%+ 的?

null

图 2 - 二手属性抽取方案架构图

NLP 任务

如前所述，将不同的二手属性识别需求拆解为文本多分类、多标签分类以及 NER 任务。

● **文本多分类**：即“n 选 1”问题，比如根据文本判断商品是否包邮（二分类）。

● **多标签分类**：即同时进行多个“n 选 1”问题，比如同时判断某手机商品的屏幕外观（好/中/差）机身外观（好/中/差）。多标签分类通常的做法是对不同标签共享网络层，并将损失函数以一定权重加，由于多个标签之间有一定程度的联系，效果有时候会比做多个单独的“n 选 1”问题更好，同时于是多个属性（属性簇）一起建模，在训练和推断的时候也会更省事。

● **NER**：即命名实体识别。

建模方法

1. 人工打标阶段

由于打标的人工成本比较高，需要设法利用集团的 AliNLP 进行辅助。方法是，首先利用 AliNLP 的商 NER 模型对输入文本进行解析。然后进行拆解，对属于 NER 任务的二手属性，如保质期/保修期/量/使用次数/服装风格等，可以直接定位到相关词性或实体的关键词进行 BIO 标注；对属于分类任务其它二手属性，则可以在电商 NER 的分词结果基础上打标，提高人工标注的效率。

2. 算法训练阶段

此为方案核心，本方案训练算法主要通过 3 种途径：

（1）使用 Albert-Tiny：建模采用主流对预训练+finetune 的方案。由于该模型推断速度更快，用对 QPS 和响应要求非常高的实时在线场景。对于 NER 任务也可以尝试在网络最后面接一层 CRF 或接。

Albert：Albert 意指“A lite bert”，名副其实，它的优点在训练速度快。Albert 的源码相比 Bert 源码基本如出一辙，但网络结构有几点重要的区别：

● Word Embedding 层做了因式分解，在词向量上大大减少了参数量。设词表大小为 V，词向量长为 H，对 Bert，词向量参数量为 V**H；对 Albert，先把词向长度量缩减为 E，再扩充为 H，参数量为 V**E+E*H，由于 E 远小于 H，H 远小于 V，用于训练的参数量得到锐减。

● 跨层参数共享：以 albert-base 为例，albert 会在 12 层之间共享每层的 attention 参数或全连接层 ffn 的参数，默认是两者都共享。源码中通过 tenorflow.variable_scope 的 reuse 参数可以轻松实现参数共享进一步减少了需要训练的参数量。

除此之外，Albert 还有一些训练任务和训练细节上的优化，此处按下不表。

Albert 依据网络深度不同分为：

● Albert-Large/xLarge/xxLarge：24 层

● Albert-Base：12 层

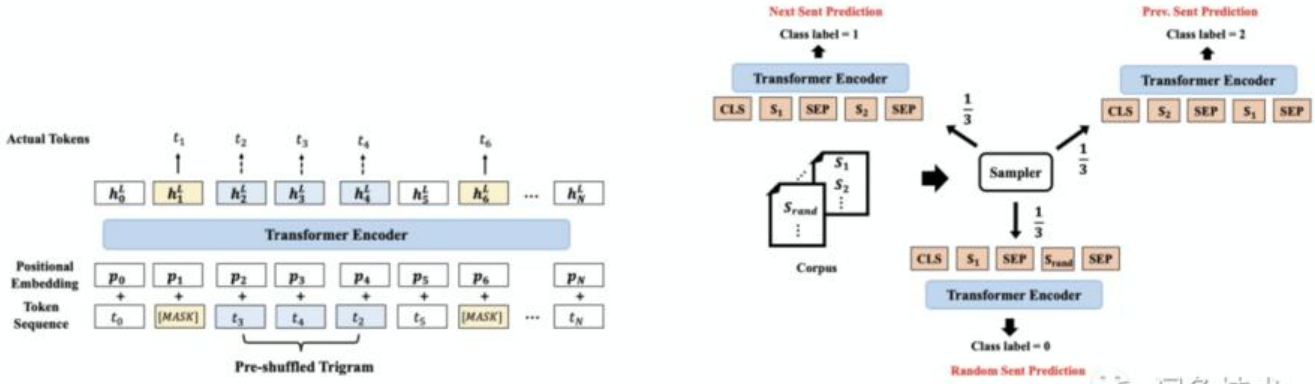
● Albert-Small：6 层

● Albert-Tiny：4 层

一般来说，层数越多，训练和推断耗时都会越久。考虑到线上部署的实时性要求更快的推断速度，本案选择了最小的 Albert-Tiny。其中文推断速度相对 bert-base 提高约 10 倍，且精度基本保留（数引用自 github/albert_zh[1]）。

（2）使用 StrutBert-Base：建模采用主流对预训练+finetune 的方案。经测算，在二手属性识别上它比 Albert-Tiny 准确率相对提升约 1%到 1.5%，可用于离线 T+1 场景。对于 NER 任务也可以尝在网络最后面接一层 CRF 或不接。

StructBert：为阿里自研算法，优点在精度高，GLUE 榜单[2]上已经排到第 3 名。StrutBert 论文相比 Bert 的主要优化点在预训练任务的两个目标上，如图 3 所示：



(a) Word Structural Objective　　(b) Sentence Structural Objective

null

图 3 - StrutBert 的预训练任务目标（引用自 StrutBert 论文）

● Word Structural Objective：StrutBert 在 Bert 的 MLM 任务基础上，加上了打乱词序并迫使其构正确词序的任务：论文中是随机抽取一个三元词组(trigram)进行打乱，然后加上了如下公式作为 M M 损失函数的约束。StrutBert 的这个灵感也许来自于网上的一个段子：“研表究明,汉字序顺并不定影阅响读，事证实明了当你看这完句话之后才发字现都乱是的”。

$$\arg\max_{\theta} \sum \log P(\text{pos}_1 = t_1, \text{pos}_2 = t_2, \ldots, \text{pos}_K = t_K | t_1, t_2, \ldots, t_K)$$

null

图 4 - Word Structural 的目标函数（引用自 StrutBert 论文）

● Sentence Structural Objective：与 Bert 的 NSP 任务不同，对一组句子对(A,B)，它不是预测 B 否 A 的下一句（二分类），而是去预测 B 是 A 的下一句、上一句还是随机抽取的（三分类）。对于(A B)这样的句子对，它在训练集中让这三种情况出现的频率各为 1/3。

本方案之所以选择 StrutBert，是因为集团内有该算法在电商领域专属的预训练模型（接口），它依网络深度不同分为：

● StrutBert-Base：12 层

● StrutBert-Lite：6 层

● StrutBert-Tiny：4 层

在离线 T+1 场景下，追求精度更高而对实时性无太大要求，因此本方案选择了 StrutBert-Base。

（3）使用正则表达式：优点：速度最快，比 Albert-Tiny 还快 10-100 倍以上；且在许多句式和关词相对固定的二手属性上，准确率比上面两种算法更高；且易于维护。缺点：非常依赖业务知识、行经验和数据分析对大量正则模式进行梳理。

3. 规则修正阶段

● 识别结果归一化：对于 NER 任务，许多识别出来的结果不能直接使用，需要做“归一化”，例如件男装衣服的尺码识别出来为“175/88A”，那么应该自动映射到“L 码”。

● 某些二手属性之间可能会存在冲突或依赖，因此在算法识别之后，需要对识别结果依据业务规则进一定修正。比如某商品卖家声称是“全新”，但是同时又表明“仅用过 3 次”，那么“全新”会自动级为“非全新”（99 新或 95 新，不同类目分级略有不同）。

算法部署

● 离线 T+1 场景：通过 ODPS(现名 MaxCompute)+UDF 的方式进行部署，即算法会通过 Python 成 UDF 脚本，模型文件则作为资源上传到 ODPS 上。

● 在线实时场景：模型通过 PAI-EAS 进行分布式部署，数据交互通过 iGraph（一种实时图数据库）和 TPP 完成。

算法评估

对每个类目的每个二手属性，制定好评测的标准，然后抽样一定量级的数据，交由外包进行人工评估评估工作通过对比人工识别的结果和算法识别的结果是否一致，给出准确率、精确率、召回率等。

最终效果

准确率

本方案识别结果经过人工评估，每个类目无论是准确率、精召率都达到了非常高的水平（98%+），误差值均远小于上线限制，并已经上线应用在闲鱼主要类目的商品上。

效果展示

null

图 5 - 二手属性抽取算法效果 Demo(手机)

应用场景 &后续展望

二手属性抽取的结果目前已应用的场景包括：

- 定价场景
- 聊天场景
- 优质商品池挖掘
- 搜索导购
- 个性化商品推荐

后续展望：

- 目前二手属性抽取总共覆盖闲鱼主流类目商品，随着开发进行，后续计划覆盖到全部类目。
- 目前二手属性抽取主要依赖于文本识别，闲鱼商品是图文描述，后续可以考虑在图片上下功夫，通图像算法完善商品的结构化信息。
- 利用和分析商品二手属性，形成优质商品标准，扩充优质商品池。

参考：

Albert 论文：https://arxiv.org/abs/1909.11942StructBert论文：https://arxiv.org/abs/1908.0457Albert_zh 源码：https://github.com/brightmart/albert_hGLUE排行榜：https://gluebenchmark.com/leaderboard

References:

[1] github/albert_zh: https://github.com/brightmart/albert_zh

[2] GLUE 榜单: https://gluebenchmark.com/leaderboard

本文转载自：闲鱼技术（ID：XYtech_Alibaba）

原文链接：闲鱼是怎么让二手属性抽取准确率达到95%+的?