基于生成式的聊天机器人

摘要　　聊天机器人是自然语言处理的一个重要的研究课题。本文梳理了生成式聊天机器人的构建步骤以及相关技术，对于相关的研究进行探讨。同时实际利用 tensorflow 构建了一个聊天机器人。同时对我构建的聊天机器人进行了效果评价，提出了一些改进措施。

聊天机器人是自然语言处理和人工智能领 S 域中的一个重要研究方向。近年来，聊天机器人系统的研究受到了广泛关注。综合运用各种人工智能技术，对推动人机对话的发展有重要意义。

关键词　聊天机器人　　自然语言处理　　深度学习　　TensorFlow

0 引言

在当前产品技术不断更新迭代，人工智能市场火爆的大环境下，一大批由于自然语言处理，语音识别，模式识别等自动化技术的大力推动下。聊天机器人系统得到了前所未有的发展，短短几年，各大厂商都推出了自家的聊天机器人系统如微软小冰，小米小爱，以及阿里的智能客服机器人小蜜。并且他们越来越强大，确实在各方面促进人机交互的发展。

研究聊天机器人的相关技术，并对其缺陷和未来的改进目标进行探究，可以使我们揭开这种技术的面纱。

1. 聊天机器人的分类、构建方式

1.1 根据输入形式划分

1）文本

2）语音

1.2 根据对话目的划分

1.2.1 任务型

为了解决任务；比如让 siri 定闹钟，打电话等。

1.2.2 闲聊型

对话就是类似于人类闲聊的聊天方式。

1.3 构建方式

1.3.1 基于检索的模型

检索式模型类似于一个搜索引擎，首先构建一个由大量查询-响应对构成的知识库。

使用倒排索引技术作查询匹配；TFIDF/BM25 用于提取关键词；余弦相似度、简单共有词、编辑距离用于相似度计算。使用这些技术是西安召回若干相关的候选回复。

但是由于没有考虑语义，所以直接使用检索分数来挑选最优回复效果很差。应该使用考虑语义的深度文本匹配模型来将这些候选回复与上下文相关联，这又涉及到交互式多轮对话模型。

1.3.2 基于规则的模型

如果问句中包含某个特定词，则机器人给予特定回答，按时这种方式需要精心编写规则，还要考虑到规则间的优先顺序。

1.3.3 生成式模型

使用 Encoder-Decoder 序列到序列模型实现聊天机器人。串联两个 RNN/LSTM，一个作为编码器，另一个作为解码器，可以记录记忆信息，与当前输入进行处理后输出。好处是可以覆盖任意话题的问句，缺点是生成的句子质量很差，可能出现语句不通顺等低级错误。通常

用于机器翻译

2.词向量

词语是 NLP 中的最细粒度。所以处理 NLP 问题时，怎么合理的表示词语就成了 NLP 领域中最先需要解决的问题。

语言模型：f(x)=y

在 NLP 中，我们把 x 看作是一个句子里的一个词，y 是这个词的上下文。这里的 f 就是语言模型，通过它判断（x，y）这个样本，是否符合自然语言的逻辑法则。

而词向量正是从这个训练好的语言模型中的副产物模型参数（也就是神经网络的权重）得来的。这些参数是作为输入 x 的某种向量化表示，这个向量就叫做词向量。

2.1 One-hot

One-hot 编码，使用 N 位状态寄存器来对 N 个状态进行编码，每个状态都有独立的寄存器位，在任意的时候，只有其中一位有效。

2.1.1 优点

解决了分类器不好处理离散数据的问题，并且编码也可以作为新的特征。

2.1.2 缺点

是一个词袋模型，不考虑词与词之间的顺序；其次，它假设词与词相互独立。得到的特征是离散稀疏的，这样的稀疏向量表达一个词效率并不高。假设状态位过多，可能会造成维度灾难。

2.1.3 改进

1)将编码由整形改为浮点型，可以在整个实数空间表示。

2)将原来稀疏的巨大维度压缩嵌入到一个更小维度的空间,也即词嵌入。

2.2 使用 DNN 训练词向量

神经网络 DNN 来训练出词向量。一般采用三层神经网络结构，分为输入层，隐藏层，和输出层（softmax 层）。

输入是某个词,一般用 one-hot 表示该词(长度为词汇表长度)，隐藏层有 N 个神经元(表示想要的词向量的维度)，输入层与隐藏层全连接。输出层的神经元个数和输入相同，使用 softmax 计算隐藏层到输出层每个位置(不同的单词)的概率。该模型中我们想要的就是经过训练以后，输入层到隐藏层的权重作为词向量。

2.2.1 简单介绍

1)输入层:

为词汇表中某一个词，采用 one-hot 编码,长度是词汇表长度。

2)隐藏层:

从输入层到隐藏层的权重矩阵词汇表长度 X 词向量维度的矩阵，其中每一行就代表一个词向量。这样词汇表中所有的词都会从高维维的 one-hot 编码转变成为低维的词向量。

3)输出层:

经过神经网络隐层的计算,这个输入的词就会变为低维的词向量，再被输入到输出层。输出层就是一个 sotfmax 回归分类器。

2.2.2 缺点

DNN 模型的处理过程非常耗时。我们的词汇表一般在百万级别以上，这意味着我们 DNN 的输出层需要进行 softmax 计算各个词的输出概率的的计算量很大。

2.3 CBOW 与 Skip-gram

2.3.1 CBOW 模型

它的输入是某特征词的上下文中 N 个词对应的词向量，输出是特征词的词向量。由于 CBOW 使用的词袋模型，所以不考虑上下文中的词和特征词之间的距离大小. 我们的输入是上下文中 N 个词的词向量，输出是所有词的 softmax 概率，对应的 CBOW 神经网络模型输入层有 N 个神经元，输出层有词汇表大小个神经元。隐藏层的神经元个数自定义。通过 DNN 的反向传播算法，我们可以求出 DNN 模型的参数，同时得到所有的词对应的词向量。这样当我们要求出上下文中对应的最可能的输出中心词时，我们可以通过一次 DNN 前向传播算法并通过 softmax 激活函数找到概率最大的词对应的神经元即可

2.3.2 Skip-Gram 模型

它的输入是特定的一个词的词向量，而输出是特定词对应的上下文词向量。我们输入特定词， 输出是 softmax 概率排前 8 的 8 个词，对应的 Skip-Gram 神经网络模型输入层有 1 个神经元，输出层有词汇表大小个神经元。隐藏层的神经元个数我们可以自己指定。通过 DNN 的反向传播算法，我们可以求出 DNN 模型的参数，同时得到所有的词对应的词向量。当我们要求出某个词对应的最可能的 N 上下文词时，通过 DNN 前向传播算法得到概率大小排前 N 的 softmax 概率对

应的神经元所对应的词即可。

2.4 Word2vec

Word2vec 同样使用 CBOW 与 Skip-Gram 来训练模型与输出词向量，但是对 DNN 模型做了很多优化。

2.4.1 霍夫曼树

Word2vec 使用霍夫曼树代替隐藏层和输出层的神经元，霍夫曼树的叶子节点起到输出层神经元的作用，叶子节点的个数即为词汇表的小大，词频作为节点的权。而内部节点则起到隐藏层神经元的作用。

权重高的叶子节点越靠近根节点，权重低的叶子节点会远离根节点，高权重节点编码值较短，保证树的带权路径最短，且符合信息论。

在 Word2vec 中，左子树编码为 1，右子树编码为 0，同时左子树的权重不小于右子树的权重。

2.4.2 Negative Sample(NEG)

Negative Sampling 是 Noise-Contrastive Estimation（NCE，噪声对比估计）的简化版本. 词典中的词在语料 D 中出现的次数有高有低，对于高频词，被选为负样本的概率更大. 本质就是带权采样问题. 优化目标为：最大化正样本的概率，同时最小化负样本的概率。

2.4.3 Hierarchical Softmax

Hierarchical Softmax 是一种对输出层进行优化的策略，输出层从 DNN 的利用 softmax 计算概率值改为了利用 Huffman 树计算概率值。先使用词汇表作为叶子节点，词频作为节点的权，构建 Huffman 树。

Hierarchical Softmax 利用 root 到指定叶子节点的路径路径来计

算指定词的概率。把 N 分类问题优化为 Log(N)次二分类。

3.语料处理流程

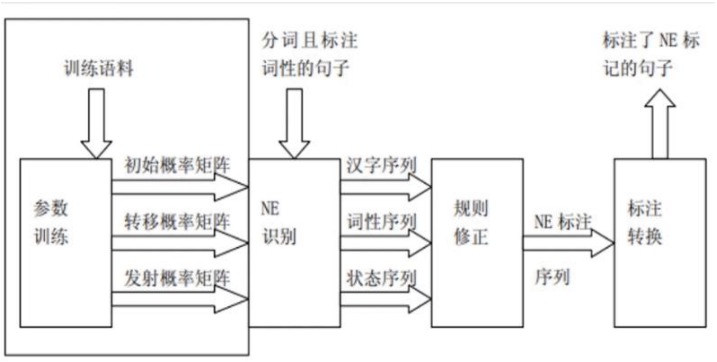

图 1：语料处理流程

3.1 语料收集

1)聊天记录，微博评论。

2)电影对话, 以及台词。

3)各种开放的数据集。

3.2 语料清洗

3.2.1 要清洗的内容

1)多余的空格。

2)不正规的符号。

3)多余的字符, 英文。

3.2.2 清洗的方法

1)正则化清洗。

2)对语料进行切分。

3)对好坏语句判断。

3.3 语料问答对的构建

对语料进行处理，构建问答对，并做简单的筛选处理以及相应的拆分。

3.4 句子向量的编码化

原始文本不能直接训练，将句子转化为向量再将向量转换为句子。

3.5 语料模型的保存

使用 pickle 来保存模型，生成 pkl 格式，利用 pkl 格式进行语料的训练。最后通过深度模型打包。

4.seq2seq

我们的目标是给定输入句子 X，希望通过 Encoder-Decoder 框架来生成目标句子 Y。X 和 Y 可以是同一种语言，也可以是两种不同的语言，因此 enc-dec 也常用于机器翻译。

在实际实现聊天系统的时候，一般 Encoder 和 Decoder 都采用 RNN, LSTM 或者 GRU 模型. 对于句子比较长的情形，LSTM 和 GRU 模型效果要明显优于 RNN 模型。但当句子长度超过 30 以后，LSTM 模型的效果会急剧下降，一般此时会引入 Attention 模型，这是一种体现输出 Y 和输入 X 句子单词之间对齐概率的神经网络模型，对于长句子来说能够明显提升系统效果。

4.1 Encoder 和 Decoder

4.1.1 Encoder

Encoder 对可变长度的信号序列 X 进行编码，将输入句子通过非线性变换转化为固定长度的中间语义向量 C.

4.1.2 Decoder

Decoder 根据句子 X 的中间语义表示 C 和之前已经生成的历史信息来生成 i 时刻要生成的单词 yi;每个 yi 都依次产生，整体而言系统根据输入句子 X 生成了目标句子 Y。

4.1.3 对话系统

对于聊天机器人来说，X 是用户输入语句(Message)，Y 是聊天机器人的回复(Response)。当用户输入 Message 后，经过 Encoder-Decoder 框架计算，先由 Encoder 对 Message 进行语义编码，形成中间语义表示 C，Decoder 根据中间语义表示 C 生成了 Response。当用户输入不同的 Message，聊天机器人形成也输出 Response，形成了一个对话系统。

4.2 引入 Attention 的作用

1)减小处理高纬度输入数据的计算负担,通过结构化的选取输入的子集,降低数据维度

2)让任务处理系统更专注于找到输入数据中显著的与当前输出相关的有用信息.从而提高输出的质量

3) Attention 模型的最终目的是帮助类似编解码器这样的框架,更好的学到多种内容模态的相互关系,从而更好的表示这些信息,克服其无法解释从而很难设计的缺陷

4)注意力机制是在序列到序列模型中用于注意编码器状态的最常用方法,它同时还可用于回顾序列模型的过去状态

5)注意力机制不仅能用来处理编码器或前面的隐藏层,它同样还能用来获得其他特征的分布

6)可以把输入内容加入到解码的过程中，对于当前输出的词，每一个输入给与的注意力是不一样的。

4.3 Beam-Search

decoder 的过程有两种主要的解码方式。

4.3.1 贪婪解码

Decoder 时，将在上一个时间步预测的单词 feed 给下一步的输入，来预测本个时间步长的最有可能的单词。但如果有一个 cell 解码错误，那么错误便会一直累加。

4.3.2 beam-search

在 decoder 阶段，某个 cell 解码时不只是选出预测概率最大的 symbol，而是选出 k 个概率最大的词（例如 k＝5，我们称 k=5 为 beam-size。在下一个时间步长，对于这 5 个概率最大的词，可能就会有 5V 个 symbols（V 代表词表的大小）。但是，只保留这 5V 个 symbols 中最好的 5 个，然后不断的沿时流图，你就可以使用 Tensorflow。你来构建图，描写驱动计算的内部循环。我们提供了有用的工具来帮助你组装“子图”（常用于神经网络），当然用户也可以自己在 Tensorflow 基础上写自己的“上层库”。定义顺手好用的新复间步长走下去。这样可以保证得到的 decode 的整体的结果最优。

5．TensorFlow

5.1 简介

TensorFlow 是一个使用数据流图进行数值计算的开源软件库，是谷歌基于 DistBelief 研发的第二代人工智能学习系统，命名来源于本

身的运行原理。Tensor(张量)意味着 N 维数组，Flow(流)意味着基于数据流图的计算，TensorFlow 为张量从流图的一端流动到另一端的计算过程。TensorFlow 将复杂的数据结构传输至人工智能神经网络进行分析和处理。

5.2 数据流图

数据流图用包含“结点”和“线”的有向图来描述数学计算。“节点”用来表示数学操作，也可以表示数据输入的起点或输出的终点。“线”表示“节点”之间的输入/输出关系。这些数据“线”可以传输多维数据数组，即“张量”(tensor)。当输入端的张量就绪时，节点将会被执行分布式异步并行运算。

5.3 TensorFlow 的特点

5.3.1 高度的灵活性

TensorFlow 不是一个严格的“神经网络”库。只要你可以将你的计算表示为一个数据流图，你就可以使用 Tensorflow。

5.3.2 可移植性

Tensorflow 支持在 CPU 和 GPU 上运行，也可以指运行在台式机、服务器、手机移动设备等。

5.3.3 自动求微分

基于梯度的机器学习算法会受益于 Tensorflow 自动求微分的能力。

5.3.4 性能最优化

Tensorflow 提供了线程、队列、异步操作等支持。

6. 模型的评价以及提高

6.1“好”的问答系统的特点

1)针对用户的回答或者聊天内容，机器人产生的应答句应该和用户的问句语义一致并逻辑正确。

2)聊天机器人的回答应该是语法正确的。采用生成式对话技术的机器人是一个词一个词生成的句子，句子的语法有缺陷。

3)聊天机器人的应答应该有趣、多样,避免安全回答。。

4)聊天机器人的基本背景信息以及爱好、语言风格等方面应该一致。

6.2 多轮会话中的上下文问题

我们使用的 Encoder-Decoder 框架可以根据用户当前输入 Message，聊天机器人自动生成应答 Response，形成了一个有效的问答系统。

但是人类聊天不止是一问一答，大多数时候我们会根据对话场景的上下文来决定我们的回复。

多轮对话就是当前的输入问句，会根据之前场景的上下文来决定输出。在多轮会话中，一般将上下文称作 Context，当前输入称为 Message，应答称作 Response。

6.2.1 使用拼接策略来改进

使用深度学习解决多轮对话需要将上下文信息引入到 Encoder-Decoder 模型中去。很简单的思路是把 Context 引入到 Encoder 中，也就是把 Context 和 Message 拼接起来作为输入提供给 Encoder，这样

就把 Context 融入模型中了。

但是问题在于，对于 RNN 来说，线型序列长度越长，模型效果越差。因此简单地拼接策略效果很差。

6.2.2 使用多层前向神经网络改进

通过将 Encoder 用多层前向神经网络来代替 RNN 模型，神经网络的输出代表上下文信息 Context 和当前输入 Message 的中间语义 C 表示，而 Decoder 依据这个中间表示 C 来生成对话 Response。这样做既能够将 Context 和 Message 通过多层前向神经网络编码成中间语义表达，又避免了 RNN 对于过长输入敏感的问题。

6.2.3 使用层级网络(HNN)

这种思路是使用层级神经网络（Hierarchical Neural Network，简称 HNN）。

HNN 本质上也是 Encoder-Decoder 框架，主要的改进是 Encoder 使用了二级结构。

第一级对 Context 中每个句子先用 Sentence RNN 对每个单词编码形成每个句子的中间表示。

第二级的 RNN 将第一级的结果按照上下文中句子出现先后顺序序列进行编码，这级 RNN 模型可被称作 Context RNN。

最后把所有 Context 和 Message 一起编码，以此作为 Decoder 的输入之一，这样就可以在生成 Response 时把上下文信息考虑进来。

6.2.4 综述

深度学习解决多轮会话的上下文信息，都是在 Encoder 阶段把上

下文信息 Context 及当前输入 Message 同时编码，以促进 Decoder 阶段上下文信息对生成 Response 的影响。

6.3 解决“安全回答”问题

安全回答问题即，无论什么输入，聊天机器人都用少数非常常见的句子进行回复，比如“是吗”，“呵呵”。想象你和一个人类聊天，她的回答总是“呵呵”，无疑让人抓狂。

当聊天训练数据中存在很多这种“安全回复”，就很容易出现安全问题。这种情况尤其容易出现在采用 Encoder-Decoder 模型构建的开放领域生成式聊天机器人系统中。

6.3.1 使用 MMI 作为优化目标

通过改进传统的使用 Sequence-to-Sequence 框架来进行模型训练时的优化目标，可以一定程度解决这个问题。通过使用最大化互信息（MMI）来代替传统的最大似然法（MLE），MMI 同时最大化 Message 生成应答 Response 的概率，以及一个反向优化目标，最大化应答 Response 产生 Message 的概率，通过 lamda 控制两者权重的超参数。

采用 MMI 作为目标函数可以显著解决很多“安全回答”问题。

6.4 个性信息一致性

对于聊天助手，聊天机器人会被视作一个虚拟人，它的个人资料比如（年龄、性别、爱好）应该保持一致。即在不同的上下文中，个人信息也不会受到 Encoder-Decoder 模型的影响。但是 Sequence-to-Sequence 模型训练的都是单句 Message 对单句 Response 的映射关系，所以并不能保证每次相同的问题能够产生完全相同的应答。

6.4.1 改进 Decoder 阶段

基本思路是把聊天机器人的个性信息引入到Decoder的输出过程中，在采用 RNN 的 Decoder 生成 Response 时，神经网络节点除了 RNN 输入外，也将个性化 Word Embedding 信息一起输入。

可以使系统倾向输出有个性的特征信息。

7.总结

本文介绍了使用深度学习构建聊天机器人采用的框架技术以及其中遇到的问题和一些前人提出的解决方案。

相对基于检索模式或者规则模式而言，基于 Encoder-Decoder 深度学习框架的聊天机器人具有以下优点：

1)构建过程是端到端（End-to-End）数据驱动的，只要给定特定领域的训练数据即可训练出特定领域的聊天系统，省去了很多传统 NLP 的工作如句法分析，语义分析等，可以提高系统的开发效率。

2)语言无关，可扩展性强。对于不同的语言，只需要使用相应的语料进行训练，不需要针对某种语言进行其他优化，大大提高了系统可扩展性的可扩展性。

3)持续改进。可以通过不断增加新的语料能持续提升聊天机器人的效果。

本文介绍了深度学习聊天机器人的以下缺陷和改进的方法。但是对于 Encoder-Decoder 模式或者其他形式的聊天机器人，还有很多方面可以去改进：

1)聊天机器人的评价标准。

目前常用的标准包括机器翻译的评价指标 BLEU、语言模型评价标准困惑度，以及图灵测试。但是还没有一种特别合适的的用于聊天机器人的评价标准，大大阻碍了聊天机器人的发展。

2)语料的缺乏。深度学习模型优点在于可以利用新的语料持续改进聊天机器人的效果。目前都是通过采集 Twitter、微博评论或者使用电影字幕等作为语料。语料库的匮乏也是聊天机器人发展的一大阻碍。

参考文献：

[1]王浩畅,李斌.聊天机器人系统研究进展[J].计算机应用与软件,2018,35(12):1-6+89.

[2]Turing A M .Computing machinery and intelligence [M] Computation & intelligence．American Association for Artificial Intelligence，1950:433-460．

[3]Weizenbaum J． ELIZA—a computer program for the study of natural language communication between man and machine [J]．Communications of the Acm，1966，9(1) :36-45．

[4]CuayahuitlH.Simpleds:Asimpledeepreinforcementlearning dialogue system [M] Dialogues with Social Robots. Springer Singapore,2017:109-118.

[5]Lifeng Shang, Zhengdong Lu, and Hang Li.2015. Neural responding machine for short-text conversation. In ACL-IJCNLP,pages 1577–1586.

[6]Oriol Vinyals and Quoc Le. 2015. Aneural conversational model. In Proc. of ICML Deep Learning Workshop.

[7] Alessandro Sordoni, Michel Galley,Michael Auli, ChrisBrockett, Yangfeng Ji, Meg Mitchell, Jian-Yun Nie,JianfengGao, and Bill Dolan. 2015. A neural network approach to context-sensitivegeneration of conversational responses. In Proc. of NAACL-HLT.

[8]Iulian V Serban, Alessandro Sordoni,Yoshua Bengio,Aaron Courville, and Joelle Pineau. 2015. Building end-to-enddialogue systems using generative hierarchical neural network models. In Proc.of AAAI.

[9]Jiwei Li, Michel Galley, Chris Brockett,Jianfeng Gao,and Bill Dolan. 2015. A diversity-promoting objective function forneural conversation models. arXiv preprint arXiv:15.

[10] Jiwei Li, Michel Galley, ChrisBrockett, Jianfeng Gao and Bill Dolan. A Persona-Based Neural ConversationModel. arXiv preprint arXiv:16.

[11]Xiang Li,Lili Mou,Rui Yan and MingZhang. StalemateBreaker: A Proactive Content-Introducing.

[12]孙立茹,余华云.基于深度学习的生成式聊天机器人算法综述[J].电脑知识与技术,2018,14(23):227-228.

[13]曹东岩. 基于强化学习的开放领域聊天机器人对话生成算法[D].哈尔滨工业大学,2017.