**1.交流**，并且需要多轮对话，应当采用开放域对话模型，指完全趋于数据驱动的方式进行训练的聊天系统，直接使用语言模型，用大量已有对话进行模拟训练，其逻辑结构简单，因此可以扩展到各种开放领域的数据集进行训练；

**2.对于模型的输入**，由于模型无法处理结构化数据，首先进行知识文本的拼接，我们在其中加入字符“的“去连接主题实体和主题问题，加入字符“是”去连接主题问题和知识值，使之顺化语句并且符合人类对话得到字符序列str，如下列所示：

{＜主题实体＞，< 的＞，< 主题问题＞，< 是＞，<知识值＞｝

将历史的对话文本和知识信息文本拼接成加入带入特殊标记的符号的一个序列；

{[确认信息编码的开始]，知识信息文本信息，[链接两种不同意义]，}

**3.预训练文本生成模型有四个框架：**

（１）Ｔｒａｎｓｆｏｒｍｅｒ－ＥＤ

Ｔｒａｎｓｆｏｒｍｅｒ－ＥＤ是一种编码器－解码器架构，分别使用Ｔｒａｎｓｆｏｒｍｅｒ编码器和解码器来处理源语句和目标语句。它利用生成预训练模型（ＧＰＴ）来初始化编码器和解码器。ＧＰＴ基于从左到右的注意力，其中每个位置只能关注每个变压器块的屏蔽多头注意力层中的先前位置。Ｔｒａｎｓｆｏｒｍｅｒ－ＥＤ首先对对话历史进行编码，然后将编码器的最终输出输入解码器以生成响应。编码器使用双向注意，而解码器使用从左到右的注意。在之前的研究中已经注意到独立编码器和解码器潜在的问题，即编码步骤可以直接合并到解码器中，模型作为整体去更新参数。并且这是唯一倾向于生成安全响应的模型，例如“我不确定你在说什么”，“好的”，“谢谢”等回复。显著较低的Ｄｉｓｔｉｎｃｔ分数也揭示了这一现象。生成的响应几乎是平淡的，生活中使用频率高的词语，而且在生成时观察到Ｔｒａｎｓｆｏｒｍｅｒ－ＥＤ架构可能是冗余的，可能不需要单独的编码器，这也使模型很难去优化。

（2）Ｔｒａｎｓｆｏｒｍｅｒ－Ｄｅｃ

Ｔｒａｎｓｆｏｒｍｅｒ－Ｄｅｃ为源语句和目标语句使用Ｔｒａｎｓｆｏｒｍｅｒ解码器，是一种仅使用ＧＰＴ的解码器架构。它只使用从左到右的注意力对对话历史进行编码，与ＧＰＴ预培训中的方式相同。然而，仅使用左向右注意的事实可能会限制生成响应时的上下文范围。许多先前的研究表明，标记后的上下文对于标记的编码很有用，最好使用双向注意力编码，这能够产生最多样化的响应。可能归因于源端从左到右的关注，这可以为生成带来更大的灵活性，因为它没有来自右侧的约束。此外，当微调数据有限时，在数据集上都具有很好性能。

（３）Ｔｒａｎｓｆｏｒｍｅｒ- MLM

Ｔｒａｎｓｆｏｒｍｅｒ－ＭＬＭ使用Ｔｒａｎｓｆｏｒｍｅｒ解码器，该解码器在源端应用双向注意力机制，在目标端应用从左到右的注意，具有屏蔽语言模型目标，因为该架构不是明确的编码器－解码器结构，所以将其归类为解码器。源编码器和目标解码器采用不同类型嵌入和自我注意掩码：它们在源端使用双向注意，在目标端使用从左到右的注意。预先训练的ＢＥＲＴ是一种双向架构，可以使用ＭＬＭ作为其预先训练目标。同样Ｔｒａｎｓｆｏｒｍｅｒ－ＭＬＭ的编码器部分使用了相同的注意机制，但解码器部分使用从左到右的注意。在损失函数上，Ｔｒａｎｓｆｏｒｍｅｒ－ＭＬＭ使用ＢＥＲＴ模型参数进行微调，从而在目标端去屏蔽一定百分比的词汇并尝试预测它们。当微调数据有限时，在ＢＬＥＵ度量方面优于其他框架，这可能得益于源端的双向关注。总的来说，模型架构与微调预先训练的对话生成模型的成功密切相关

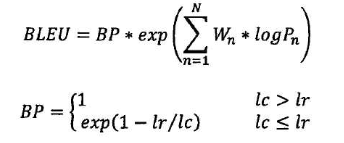
（４）Ｔｒａｎｓｆｏｒｍｅｒ－ＡＲ

Ｔｒａｎｓｆｏｒｍｅｒ－ＡＲ使用自回归目标的变压器由于与Ｔｒａｎｓｆｏｒｍｅｒ－Ｄｅ的唯一区别是Ｔｒａｎｓｆｏｒｍｅｒ－ＡＲ在源端应用双向注意，因此可以通过注意机制来解释性能上的差异。这表明源端的双向关注有助于模型更好对对话历史进行编码，从而产生更好的响应。人类评估结果还表明，Ｔｒａｎｓｆｏｒｍｅｒ－ＡＲ在使用百万级数据进行微调时表现良好。然而，对于小规模数据集，Ｔｒａｎｓｆｏｒｍｅｒ－ＡＲ并没有很强的优势，变得不如Ｔｒａｎｓｆｏｒｍｅｒ－Ｄｅ和Ｔｒａｎｓｆｏｒｍｅｒ－ＭＬＭ有效

**4.对于判断对话文本的质量也有多种指标**

（１）ＢＬＥＵ

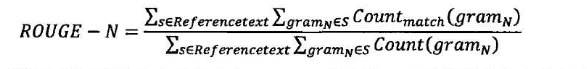
ＢＬＥＵ（ＢｉｌｉｎｇｕａｌＥｖａｌｕａｔｉｏｎＵｎｄｅｒｓｔｕｄｙ）用于衡量两个句子之间相似性的指标之一，它最初是为机器翻译提出的，将文本的候选翻译与一个或多个参考翻译进行比较。除此之外，ＢＬＥＵ还用于其他文本生成任务，如文档摘要、图像说明、人机对话和语言生成等，具体计算方法如公式所示：



其中ｌｃ代表模型生成文本的长度，ｌr代表最短参考文本长度。上面介绍了ＢＬＥＵ计算ｎ－ｇｒａｍ精确率的方法，但是仍然存在一些问题，ＢＬＥＵ与人类评估的相关性最强，但并没有显著优于表现最好的ＲＯＵＧＥ变体。尽管它可以作为机器翻译任务的一个很好的度量标准，但它与其他生成任务，如图像字幕或对话响应生成，与人类判断没有很好的相关性。而且Ｃａｃｃｉａ等人Ｍ发现，具有完美ＢＬＥＵ分数的生成文本通常在语法上是正确的，但缺乏语义或全局连贯性，从而得出生成文本的信息内容较差的结论

（２）ＲＯＵＧＥ

ＲＯＵＧＥ（Ｒｅｃａｌｌ－ＯｒｉｅｎｔｅｄＵｎｄｅｒｓｔｕｄｙｆｏｒＧｉｓｔｉｘｉｇＥｖａｌｕａｔｉｏｎ）可以看作是ＢＬＥＵ的改进版，专注于召回率而非精度。虽然主要用于评估单个或多文档摘要，但它也被用于评估短文本生成，如机器翻译、图像字幕和问题生成。ＲＯＵＧＥ包括大量不同的变体，包括八种不同的ｎ－ｇｒａｍ计数方法，用于测量生成的文本和参考文本之间的ｎ－ｇｒａｍ重叠：Ｒｏｕｇｅ－｛ｌ／２／３／４｝测量参考文本和生成文本（例如，摘要，对话）之间的ｕｎｉｇｒａｍｓ／ｂｉｇｒａｍｓ／ｔｒｉｇｒａｍｓ／ｆｏｕｒ－ｇｒａｍ单个标记的重叠。然而，ｒｏｕｇｅ对ｎ－ｇｒａｍ匹配的依赖可能是一个问题，特别是对于长文本生成任务，因为它不提供关于生成文本的叙事流、语法或主题流的信息，也不评估文本与生成文本的语料库相比的事实正确性。计算公式

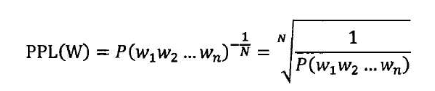


公式的分母是统计在参考文本中Ｎ－ｇｒａｍ的个数，而分子是统计参考文本与模型生成文本共有的Ｎ－ｇｒａｍ个数。如果给定多个参考译文Ｓｊ，ＲＯＵＧＥ－Ｎ也给出了一种计算方法，假设有Ｍ个译文Ｓｉ………ＳＭ，ＲＯＵＧＥ－Ｎ会分别计算模型生成文本和这些参考文本的ＲＯＵＧＥ－Ｎ分数，并取其最大值，公式



（3）ＰＰＬ

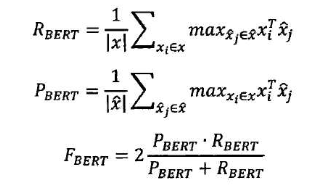
ＰＰＬ（ｐｅｒｐｌｅｘｉｔｙ）也可以用来比较两个语言模型在预测样本上的优劣。困惑度越高的句子人们理解或阅读会觉得不合逻辑或者杂糅难懂，造成这种现象可能是语法错误或者指向不明，但也有可能意味着句子复杂，多重逻辑，而非简单的无意义序列，困惑低代表句子语义流畅，语言模型的建模能力强。具体计算公式如所示



这里之所以除以Ｎ，是为了考虑长短句的平衡问题，也可以单纯的不评价模型，而考虑生成句子的质量。这个时候的概率就不是ｄｅｃｏｄｅｒ解码过ＳｏｆｔＭａｘ之后的概率了，而是语料库里面的ｎ元组：

（4）ＢＥＲＴＳｃｏｒｅ

最近的大规模预训练语言模型（ＰＬＭｓ），如ｅｌｍｏ和ｂｅｒｔ使用上下文化词嵌入来表示句子。使用ｂｅｒｔ模型来获得句子表示，在顶层使用基于距离的相似度度量作为学习信号，如类似于ｄｓｓｍ的余弦相似度。但是两者在计算上都比ｄｓｓｍ要昂贵得多，因为它们使用了更深层次的神经网络架构，并且需要针对不同的任务进行微调。该模型可用于各种ＮＬＧ任务的评估。专注于机器翻译任务，还可以从ｂｅｒｔ嵌入计算句子表示（没有微调然后使用其参考文献的平均召回率等指标计算翻译文本与其参考文献之间的相似性。在大量丰富的实验验证鲁棒性和有效性，公式



**5.模型结构**

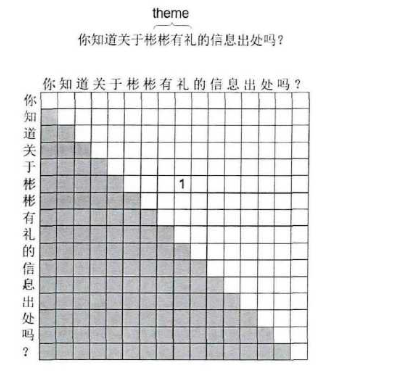
（１）句子编码器：句子编码器负责将对话上文和主题问题进行拼接将其编码为对应的语义表示向量ｘ，并分别输入到问题解码器和主题解码器中。对于问题解码器，只取Ｘ第一个字符的向量表示作为输入，而对于主题解码器，使用ｘｋ向量用于初始化隐状态。

（２）主题解码器：主题解码器根据对话上下文的信息，计算主题概率分布，为每一条候选主题赋予不同的分数。

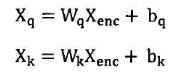
（３）问题解码器：问题解码器对Ｘ的第一个向量进行编码，与标准标签做对比计算损失

**6.主题解码器**

主题解码器主要是识别出对话数据中的对话主题，但对话上下文中包含大量与对话主题相关的实体信息，同时，可用于回复生成的主题实体数量很少，大多数情况只有一个，很少有多个主题的情况，因此需要对候选所有主题进行过滤与选择。简单理解就是从可选的主题标注序列中，选择概率最大的一个序列，因此可以通过有监督学习的方式训练一个分类器作为实体选择器，因此知识实体选择可以当作命名实体识别任务（ＮＥＲ）看待，即把主题当作实体去识别出来，属于序列标注任务。主流的方法一般采用ｂｅｒｔ＋ｂｉ－ｌｓｔａ＋ｃｒｆ５４１模型去识别文本中的实体，但其在使用情况下有很大的局限性，必须实体的所有标注都预测正确才算正确，所以在面对语音转写文本（ＣＲＳ）的语料时，当实体语音部分转写错误得到错误的实体表达，模型可能无法检测出实体，并且针对文本多个实体时，无法得到实体优先级信息。针对以上情况，本实验采用全局指针网络（ＧｌｏｂａｌＰｏｉｎｔｅｒ）做实体识别模型，原因有二：其一全局指针网络仅通过判断的实体头和尾位置的概率，可以有效避免实体丢字、错字、多字的情况，其二，对于判断出的实体都存在一个置信分数，根据情况不同可以用置信分数去判断实体优先级顺序，解决了多实体问题.



具体来讲，经过句子编码器得到的语义表示向量ｘｅｎｅ作为该模块的输入，经过模型学习头实体和尾实体位置分布从而根据指针标注规则得到该句子在该潜在关系类型下的头实体和尾实体。具体来说，如图所示。假定待确定的文本序列的长度为ｎ，由于只需要一个实体类型来确定，且各待识别实体均为序列中连续的片段，其长度没有限制，并可互相嵌套（这两实体是相交的），然后，这个序列有ｎ＊（ｎ＋ｌ）／２个个不一样的连续子序列，这些子序列中包括全部可能存在的实体，而且我们只需要从这ｎ＊（ｎ＋ｌ）／２个候选实体中挑选出真实的实体即可，它实际上是多标签分类的问题。而只是空间复杂度的提高，在并行性能良好的情况下，甚至可将时间复杂度降至０（ｎ）以下，由此该任务的输出为其中ｎ表示输入句子ｘ的编码长度，Ｋ时包含０和１组成的二维矩阵，横轴位置坐标代表实体起始位置，纵轴位置坐标代表实体结束位置，并且横轴位置坐标数值小于等于纵轴位置坐标数值。值为１时代表该坐标下为一个实体，为０则代表无意义。设对话文本经过句子编码器的编码后得到向量序列Ｘ＝{x0,x1………xn }，通过公式



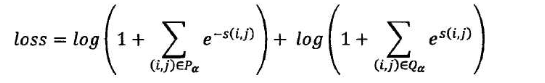
得到序列向量序列和Ｘｋ，Ｘｑ，其中Ｗｑ、ｂｑ、Ｗｋ和ｂｋ是模型可学习的参数，此时我们可以定义每个实体分数，如公式所示



同时加入旋转式位置编码（ＲｏＰＥ）来提升对实体的长度和跨度的敏感度，因此能更好地分辨出真正的实体出来，通过设置一个变换矩阵Ｒｉ，满足关系Ri\*Rj＝Ｒｊ－ｉ，这样一来我们分别应用到Ｘｋ，Ｘｑ中，如公式所示：



如公式，在该实验情境中，它的形式并不繁杂。实例说明了这种方法的有效性和优越性个具体一通过。针对目标类别数少的多标签分类、尤其适用于总类别数较多的情况，展是交叉熵在单目多分类中的一种扩，采用多标签分类损失函数，当计算目标损失时



其中４是该样本的所有实体的首尾集合，是该样本的所有非实体首尾集合，并且只需要考虑ｉ<=ｊ的组合

