Multimodal Emotion-Cause Pair Extraction in Conversations Fanfan Wang , Zixiang Ding , Rui Xia , Zhaoyu Li , and Jianfei Yu

（Transactions on Affective Computing ，ccf-b，2023）

Motivation：

对话中包含大量情感，本文希望基于多模态（文本，音频，视频）的对话中提取情绪，并在此基础上分析原因。构建了一个多模态对话情绪原因数据集 Emotion-Cause-in-Friends，其中包含 Friends 情景喜剧中 13,619 条话语中的 9,794 个多模态情绪原因对。我们通过建立两个基线系统来对任务进行基准测试，包括考虑原因和情绪位置的固有模式的启发式方法，以及结合多模态特征进行情绪-原因对提取的深度学习方法，并进行人类性能测试以进行比较。

情绪原因分析这一旨在提取潜在情绪原因的新任务因其重要的应用价值而受到广泛关注。它包含两个代表性任务：情绪原因提取 （ECE）旨在确定哪些子句包含给定情绪的原因 和情绪原因对提取 （ECPE）旨在提取情绪及其相应的原因，而无需提前进行情绪注释。

本文的数据集共有六个话语级别的情绪-原因对，并给出每个场景的话语以及情感，以及每个场景话语对其他场景情感的影响。一种方法是使用情绪-原因对之间的固有模式的启发式搜索，另一种是深度学习方法称为 MECPE-2steps，改编自新闻文章中 ECPE 的代表性方法 [9]。我们结合多模态特征进行话语表示，分别提取情绪话语和原因话语，最后构建情绪-原因对。

在对话中，情绪通常在话语层面进行注释。原因是指与相应的情绪高度相关的明确表达的事件或论点。在这项工作中，我们使用话语来描述情绪原因，即使用具体的话语，或是语言描述动作或是时间来描述产生对应情绪的原因。

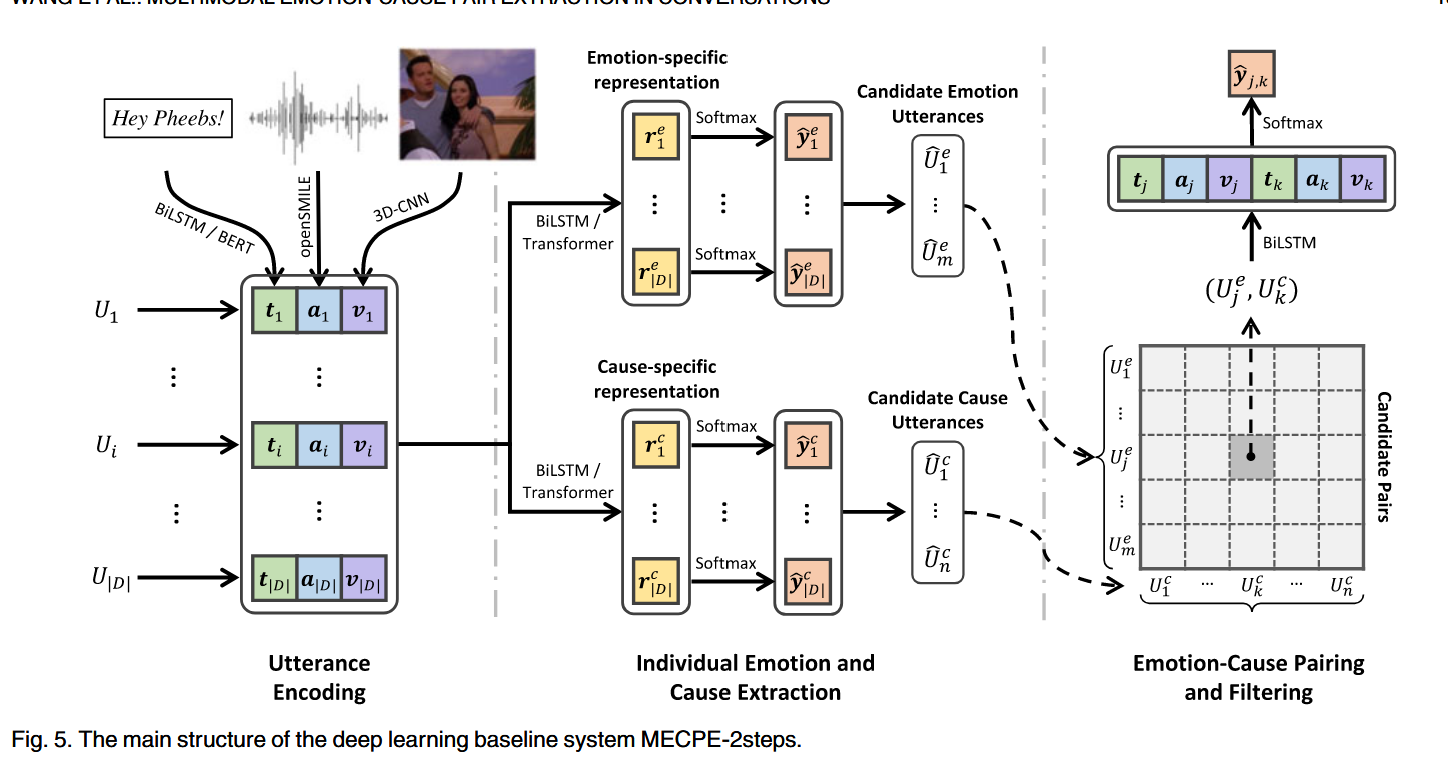
定义一个场景，包含文本，视频和音频，任务一是找出情绪-原因对。任务2，在任务1的基础上，确定情绪的类别。所以整体流程是先确定是什么事件引起了人物对应的情绪变化，再确定情绪是什么类型。

情绪产生的原因也可以分成客观原因，比如情侣之间的接吻之间是愉悦的，但是不小心看到的人会觉得厌恶，这就是没有直接影响其他人的情绪，和主观原因，比如某人对另一个人的训斥，就是一人的愤怒情绪导致了另一个人的悲伤情绪。

值得注意的是，以上两项任务可以在两种设置下执行：静态设置和实时设置。前者可以利用整个对话的上下文，而后者仅依靠历史话语来检测情绪和原因。也就是对应是不是Mask了后续数据。

Methods：

第一种寻找情绪-原因对的方式是启发式搜索，其内在原因是，原因通常出现在情绪之前几个话语中。首先使用情绪分析的方法，先找到属于情绪类型的话语，然后使用先验分布的方法，1是通过多变量伯努利分布假设。2.是利用多项式分布假设，确定原因是在情绪较近的句子中。



第二种寻找情绪-原因对的方式是深度学习，主要基于ECPE方法

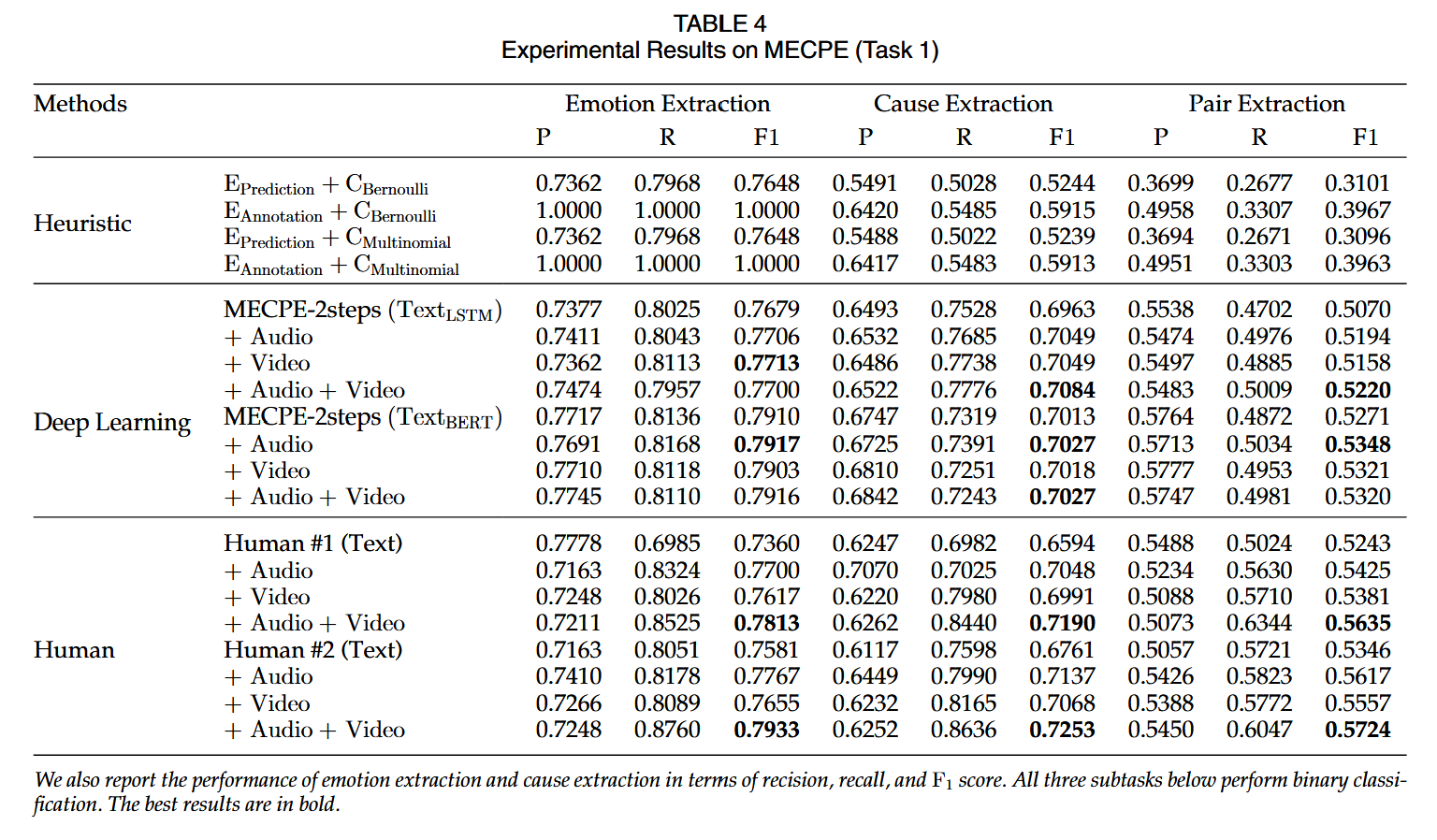
首先需要对单模态进行特征提取，然后进行连接得到多模态特征。对于文本信息来说，需要使用Glove算法（[10.1. 词嵌入（word2vec） — 《动手学深度学习》 文档 (gluon.ai)](https://zh.gluon.ai/chapter_natural-language-processing/word2vec.html)）以向量维度为300来学习得到词向量，之后输入具有标准注意力机制的BiLSTM双向LSTM的编码器中，或是将文本直接输入到预训练的BERT作为编码器，通过学习整个文本的内容，让文本中的每个词都能融合全文语义信息，进而获得每一个场景下的文本内容的文本特征，记为t[i]。对于声音模态信息，使用openSmile提取（多数文章均使用本方法）。记为a[i]。对于视频模态信息，使用C3D的3D-CNN网络（简单来说就是，将卷积核从二维的宽高，升维成三维的宽高深度。不过他与多通道卷积不同，多通道卷积的卷积核仍然是二维，但是只是同时对多个通道使用同一个卷积核进行卷积操作。而三维会的深度大小会比通道数小，所以可以做到在整体上移动，总之视频图像作为连续的图片，使用3维卷积核是优于二维的），使用深层3D卷积网络从每个场景中，先将视频改成171 \* 128的分辨率的16帧图像，送入C3D网络，隐藏层的维度维4096，最后使用全连接层降维到128维的视觉特征，记为v[i]。三者内容结合（如何结合）后得到多模态特征信息，记为U[i]

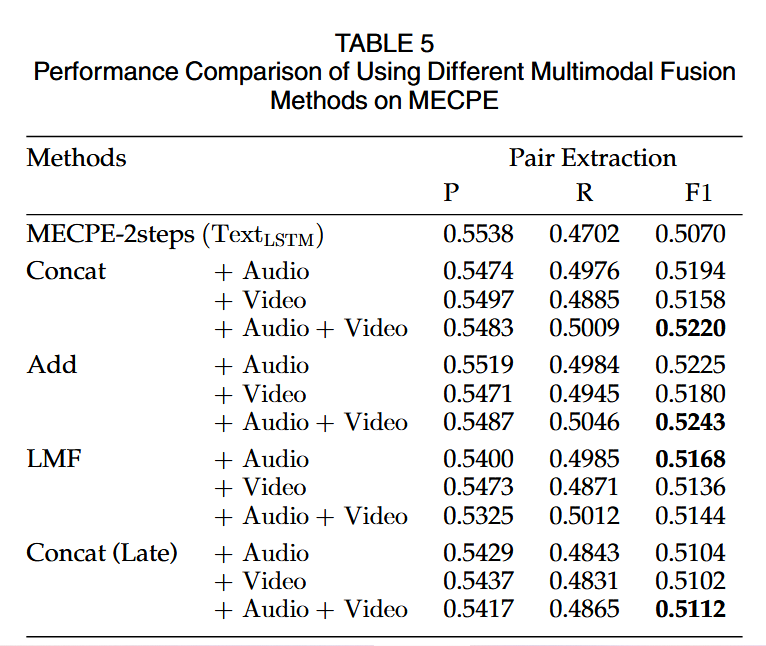
寻找完特征后，通过多任务学习提取情感和原因。需要将从对话中提取出来的文本，音频，视频信息合并为U[i]（内容应该和文本提取出来的batch\_size\*seq\_len\*embedding\_dim大小一样），分别送入基于BiLSTM或是Transformer所建立的编码器中进行情绪提取和原因提取得到两个编码，也可以说是隐藏层状态。然后将各自编码器中的隐藏状态经过一个softmax层 （做一个二分类，即当前的对话内容是不是情绪话语或者是一个原因话语，不太确定）。

当寻找完可能的情绪话语内容E={U[i],U[j]}和原因话语内容C={U[a],U[b]}后，需要进行两组配对和筛选以提取一组具有因果关系的情绪-原因对。第一步需要使用笛卡尔积获取所有可能的配对情况。第二步，使用一个特征向量X（U[i],U[j]）来代表每一个配对情况，它连接了情感话语和原因话语的独立多模态表示形式，以及两个话语之间的距离向量。最后，将对表示馈送到 softmax 层中，检测这个配对是否合理。(第二步没太看懂干了什么)

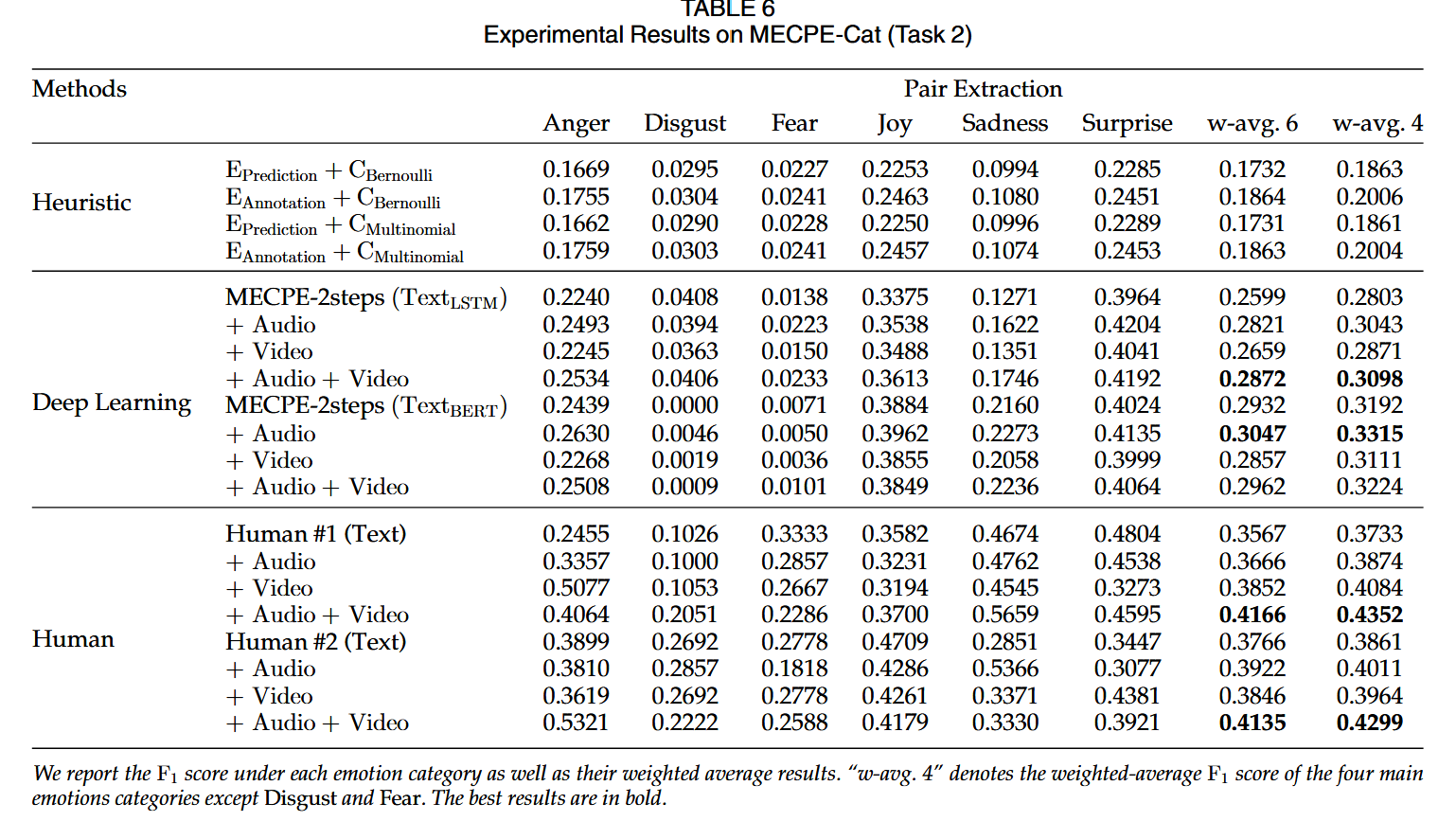
此外，不同的多模态融合方法也会影响情绪-原因对检测的精度，分别有Concat（直接串联）（并且区分了融合多模态是在进入BILSTM提取情绪或是原因隐藏层状态之前还是之后），Add（加法），LMF（低秩多模态融合方法）。

实验过程：

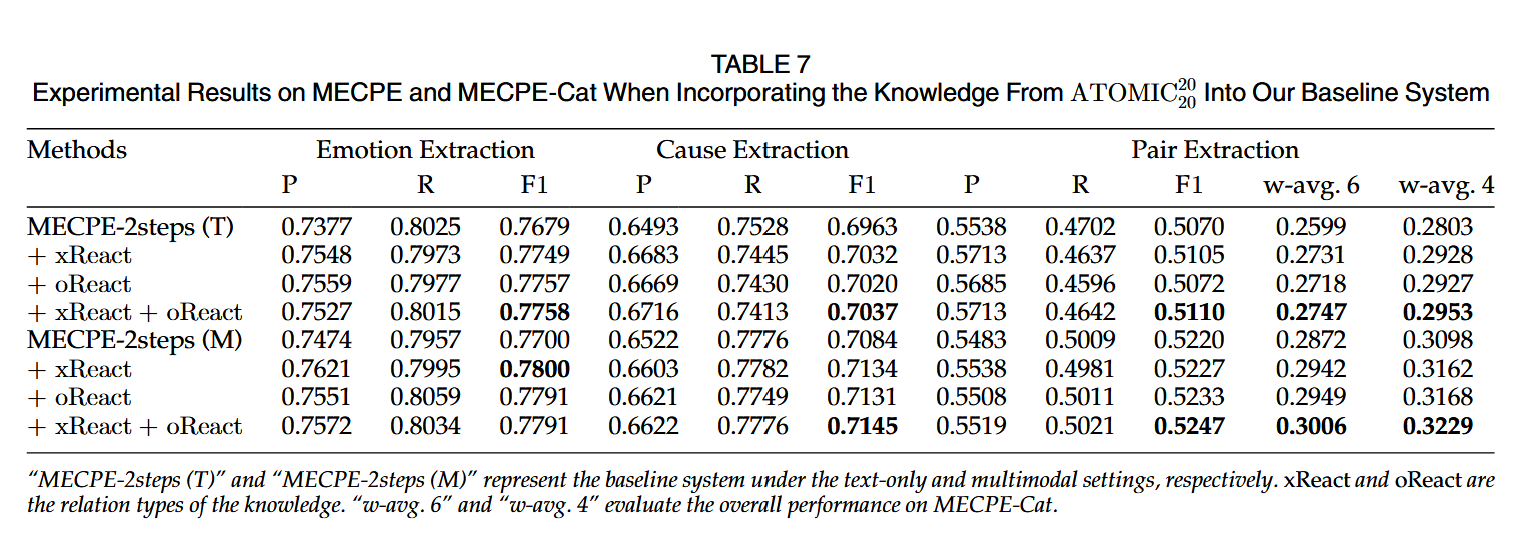
  
 先是依据准确率（预测的正确率），召回率（选中了多少注释好的句子），F1值评估了使用启发式搜索，深度学习，和人类直接识别三者的测评结果。可以发现，1、人类将文本信息结合音频和视频信息后，能更好的理解对话中包含的情感和原因，证明了多模态的结合能提升检测性能.2、但在深度学习的实验中也发现，在单纯的理解文本内容上，BERT和人类的预测F1值已经非常接近，说明BERT较强的表示能力，能很好的理解情绪和原因。但在结合音频和视频之后，机器并不会响人类一样，对识别精度有了明显提升。说明在音频和视频信息的采集上仍有较大提升空间。



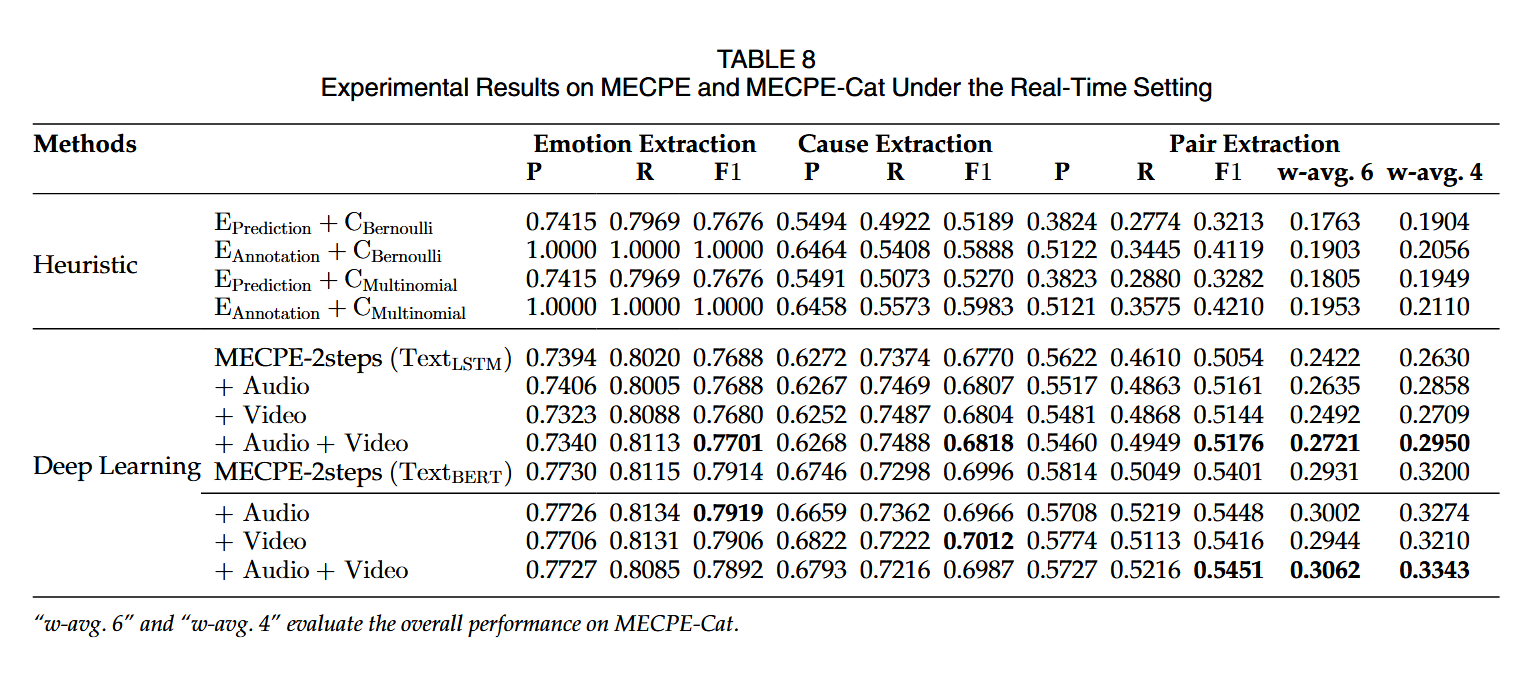
此外比较了多种多模态融合方法，发现在提取完话语的文本，音频，视频特征后融合好再去经过BILSTM或是Transformer的编码器得到隐藏层内容，进而判断是否是情绪话语，或是原因话语。多模态融合方法也是一个可以优化的方案。



下一个实验是为每个情绪-原因对预测一个额外的情绪类别。具体来说，第一步判断原因子句的方法不变，判断情绪子句的方法变为情绪分类，第二步的情绪-原因对判断不做变化。从实验中发现的好的部分是经过多模态融合之后，情绪分类的效果有所提升，但是发现的不好的方面是由于情绪类别失衡的存在，标注为惊恐和厌恶的情绪难以被识别，但基于MECPE的方法仍然较难预测具体的情绪-原因对（如果按照下一篇论文的解释，就是第一步判断情绪子句的错误被带到了第二步的情绪-原因对提取中。第一步的判断只有70~80左右正确率，自然会影响下一步的提取）



此外可以利用comic-atomic（一种常识知识库）的xReact和oReact（基于社会间关系构建的数据，比如oReact的数据，代表了人们在感谢他人，或是自豪等情绪下所会说出的语言的常识知识。）给定一个话语和一个关系类型，将生成五个尾部短语作为话语的常识知识，以达到扩大数据集的效果。再将生成的话语，经过BiLSTM编码器，得到每个话语的隐藏层状态，并依据数据库生成时所带的连接关系，确定好情绪-原因对，并且指定产生的具体情绪。可以提高模型性能。



最后一个实验，进行了实时测试，即不能使用整段对话，只能依据历史上下文进行情绪-原因对和情绪的判断，具体修改是，只能使用单向LSTM，以及带掩码Mask的Transformer，并将数据集中丢弃原因位于情绪之后的候选情绪-原因对。结果显示启发式方法和基于 BERT 的 MECPE-2steps 实现了更好的性能。改进主要体现在 Recall 分数上，可能是因为限制原因的位置简化了任务，并且更容易找到正确的情绪原因。但是，在 Real-time Setting 下，基于 LSTM 的 MECPE-2steps 的性能会变差，尤其是在 MECPE-Cat 任务上。我们推测，它对未来信息的缺乏更敏感，这导致原因提取的 F1 分数下降，从而抵消了位置限制的好处，而更强的基于 BERT 的系统基本不受影响。那么对于实际的应用来说，使用BERT会更好的满足实际需要。

历史方法回顾：

多模态情绪-原因分析仍然是难点，但是多模态的情绪识别已经有了很多方法，比如使用cNN，rNN，Transformer相关方法。常用数据集有MELD，和MEISD（最新的，一段话语有多种情绪，并由打分决定情绪强弱）。总之，我们认为 MECPE 任务的以下挑战值得进一步研究：

如何建立一个多模态对话表示框架，以有效地对齐、交互和融合来自三种模态的信息？

如何有效地对对话中情绪识别和原因提取的说话人相关性进行建模？

如何感知、理解和利用视觉场景来更好地协助对话中的情绪原因推理？ （在图像描述生成任务中，需要将图像特征和文本特征进行融合，共享特征学习可以找到图像和文本之间的对应关系，从而生成更准确的描述。）

如何利用外部常识性知识来弥合对话中未明确反映的情感和原因之间的鸿沟？（下文有使用常规知识扩充数据集的做法。）

UniMEEC: Towards Unified Multimodal Emotion Recognition and Emotion Cause（2024，还未收录）

UniMEEC：迈向统一的多模态情绪识别和情绪原因

本文主要研究问题，如何将情绪识别和情绪-原因对配对两个问题结合起来，因为二者都使用了相同的技术，多模态融合、上下文建和外部知识。但是，单独训练 MERC 和 MECPE 可能会导致在实际应用场景中无缝集成这两项任务的潜在挑战。一个自然而然的问题出现了：我们能否协调它们的输出并确保它们在实际应用中保持一致？

提出的UniMEEC框架，为了统一 MERC 和 MECPE，我们将 MERC 和 MECPE 重新表述为两个掩码预测任务，并根据模态特异性提示训练 UniMEEC。特定于模态的提示为每种模态构建了一个提示模板，旨在通过单模态和预训练模型之间的交互来编码未填充槽的表示。此外，现有的 MERC 和 MECPE 工作主要集中在多模态融合、模态对齐和外部知识。一个长期被忽视的事实是，对话中的话语间依赖关系和上下文。因此，我们提出了一个特定于任务的分层上下文聚合模块来捕获面向特定任务的上下文。该模块以三级图注意力网络为支柱，控制情感特定、原因特定和话语特定节点之间的信息流转，以聚合上下文信息。

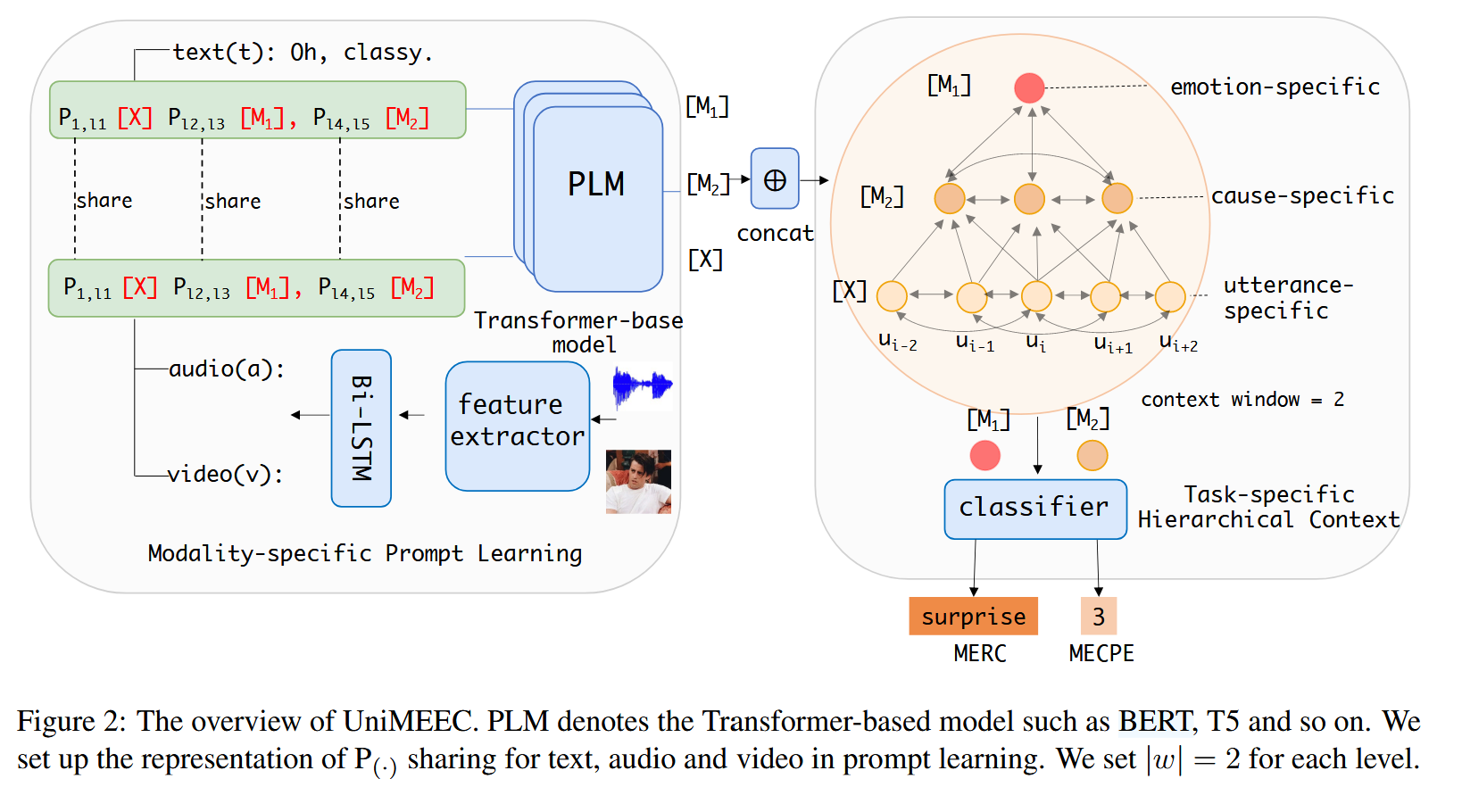
UniMEEC 包含特定于模态（即文本、视频和音频模态）提示学习 （MPL） 和特定于任务的上下文聚合 （THC），其中 MPL 从预训练语言模型 （PLM） 中探测特定于模态的知识并在模态之间共享提示学习（即使用相同的提示词，但各个模态提取各自的话语信息作为输入X提示），而 THC 有序捕获面向特定任务的上下文。

本模型包含三个部分，1、如何进行多模态的融合，可以使用图神经网络，对话语信息的内部依赖关系进行建模，或是使用Transformer模拟跨模态交互。2、如何解决上下文整合问题，即构建图形结构，或是引入情绪动力学的方法。3、如何利用外部知识。高级 MERC 研究整合了外部知识，（比如上一篇论文使用的comic-atomic数据集）采用迁移学习等技术、常识性知识、多任务学习和外部信息引入更多辅助信息，帮助模型理解对话。

MECPE 的目标是在给定情绪话语的情况下识别相应的原因话语，从而产生成对的话语。本文提出的模型基于提示调整Prompt-tuning（一种解决少数样本场景的问题）所构建。

【传统的使用方法是fine-tuning，微调法，在预训练阶段，使用大规模无监督预料上预先训练好一个大模型，比如BERT，根据下游任务，我们只需要准备好数据集，直接调用BERT模型，获取最后一层的隐藏层状态，之后利用状态经过后续变换后，进行损失函数，等计算，只需要对自已定义的参数和损失函数进行具体的微调。而prompt-tuning方法，简单来说需要两个东西，模板和标签词映射。对于模板，初始输入 X 通过模板进行修改，以形成一个带有未填充槽的文本字符串提示 X′。随后，使用语言模型来概率地填充缺失的信息，从而产生一个最终的字符串 Xˆ，模型从中输出 y，然后依据下游任务，以情感分类识别来说，需要将大模型给出的y，对应到具体的结果中，比如分两类，积极和消极，Mask中给出了fun的单词，归类为积极。总的来说，prompt-tuning将整个训练过程都交给预训练好的大模型了操作，只需要构建合适的模板内容，合并到需要操作的文本中即可。其主要的功效就是，免去了自已再次利用预训练模型，训练一个自已模型的过程，转为直接利用提示，只使用预训练模型进行。】

模型组成：



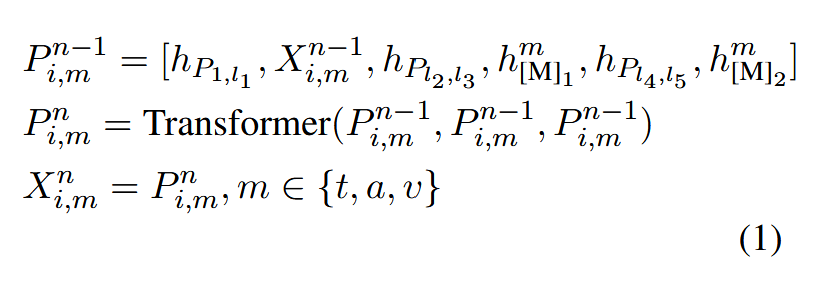
由模态特定提示学习 （MPL） 和任务特定上下文聚合 （THC） 组成。特定于模态的提示模板包含模态信息 [X]、辅助提示词 P（·） 以及掩码词元 [M]1 和 [M]2，上下文窗口为2。模态信息 [X]是从各个模态中提取到的信息。提示词为了指定出文本，音频，视频中体现的情绪，掩码词元，表示下一步送入PLM（预训练的大语言模型中需要预测的词语）每一个模态共享提示词p，每一个模态各自需要提取模态信息，各自预测两个掩码词元。

任务特定上下文聚合 （THC）中，先将之前单模态的表示融合连接起来，其中的X表示了当前话语的三个模态，文本，音频，视频的整体的隐藏层状态。THC 将特定于情感、特定原因和特定于话语的表示作为节点，并在上下文窗口中对它们的依赖关系进行建模，我们分别根据 [M]1 和 [M]2 的表示来预测情绪类别和原因话语在对话中的位置。

与上一篇论文，相同的是，同样使用了一个话语中的文本，音频，视频的三个模态。但与分离情绪分析和情绪-原因对寻找将两项任务分别进行不同。本任务直接利用模板，将两个任务形式化为掩码预测问题一起执行。本模型的提示模板手动设计，它包含文本输入[X]、情感类别槽[M]1、原因索引槽[M]2和辅助提示部分。[X]是用原始文本序列填充的槽，以获取提示模板。[M]1表示一个句子的情感类别，例如“快乐”或“悲伤”。[M]2表示对话中原因句子的位置，例如“1”或“2”。这两个都是未填充的答案槽，由我们提出的UniMEEC预测。而对于辅助提示部分一般为“the emotion category of”、“is”和“and its corresponding cause utterance index is”等指示掩码中需要填充的类型词语。对于音频信息，和视频信息的提示，需要将模板中的[X]提示替换成其他形式。具体来说，文本模态的信息从词嵌入层得到，（是指直接将本次话语中的文本，经过词嵌入后直接当作X吗）音频模态信息需要通过librosa处理为数字序列向量后，提取梅尔频谱图作为音频模态信息（这个内容在MeiJU2025挑战赛的代码中也使用到了，可以学习这里的），而视频模态信息使用effecient-Net的预训练模型提取信息。

获得了提示Prompt之后，需要经过三个不同的Transformer层，分别处理各个模态信息。（也就是说这个使用了后融合的方法）对于文本信息来说，直接将文本提示部分输入编码器，通过监督学习检查两个插入部分M1，M2的真实答案。得到文本的提示序列。而对于音频和视频信息。在前面我们已经通过librosa或是efficient-Net将其转换为向量，那么就和文本预先转为word-embedding的效果一样了，得到这个音频，或是图像的向量表示（可以找一下源码看看大小是什么样的，毕竟音频视频长度肯定会不一致。也可以学习一下视频信息的提取效果）再把他们传递到BILSTM中得到当前话语中的对应的提示输入X，此外，由于多模态融合的需要，需要将两者的向量，进行扩充，1是长度上的扩充，2是维度上的扩充。均用填0的方法实现。经过多层Transformer不断更新提示模板中每个部分的隐藏层状态。最后将三个模态上的向量进行拼接，得到本个话语的编码后的表示。（每个模态信息的提取方法和提取结果，以及最后的融合过程，看看其他论文有没有详细阐述的。）

此外每一层的Transformer需要重复计算N次，每一轮的输入文本信息X，都是当前话语的上一轮结合了mask1，mask2，输入X，提示词P的提示句子的隐藏层状态P再放入一个Transformer计算得到下一轮。



为了解决mask1，mask2无法捕捉上下文信息，是不同话语之间的上下文。所以此处还需要限定上下文窗口。为此构建了一个三层的图注意力网络，分为mask1，mask2，话语层。

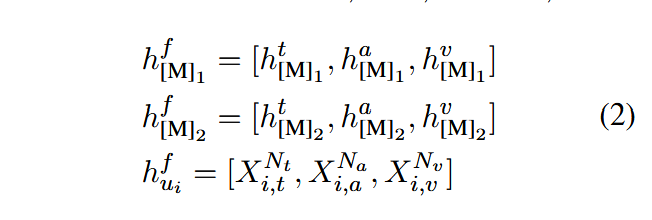
图注意力网络的介绍：

图神经网络中的图结构，用于表示节点之间的关系重要性的强弱。点只会和与他相连的边有关系，如果是无向边，代表两者互相影响，如果有向边就是单边影响。

计算的公式如下：



H[i],h[j]表示每个点所代表的隐藏层状态，就n\*1的向量表示。W[i],W[j]表示权重矩阵，（可以理解为注意力机制的计算QKV矩阵的意思，将信息降维）。然后拼接两个相邻节点的向量。变成2F\*1的向量，最后经过a矩阵，转换为一个数字，最后经过softmax，计算结点ij之间的权重，占i的所有相邻节点总权重的比值。在此基础上还可以增加多层注意力机制，通过初始化不同的W权重矩阵，分析不同的特征。

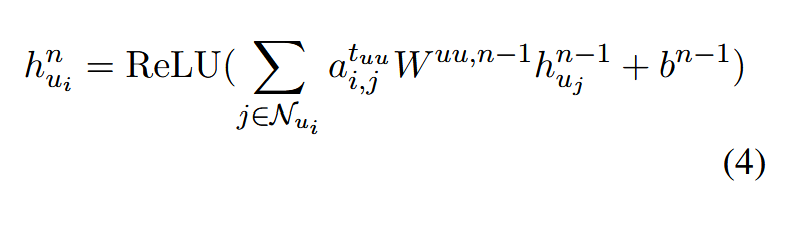


本文的图注意力网络节点共设置了三层，每层的初始状态，由上一轮得到的三个模态信息concat得到。每一层之间是否相连用一个矩阵表示，这个和第五篇论文Neutral Utterances are Also Causes: Enhancing Conversational Causal Emotion Entailment with Social Commonsense Knowledge的有向无环图中的Ac矩阵应该是一个意思，只不过两者如何连边有所不同。

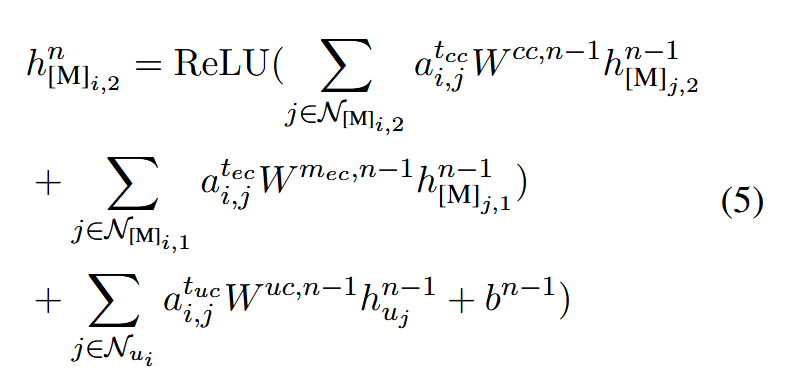
最底层的话语层，只有有向边连接第二层的M2原因层，代表了话语只能影响原因和情绪，反过来的计算是无用的。第二层原因层与第一层情绪层是无向边，表示可以互相影响。此外，每一层之间也可以互相影响，至此本模型一共5种类型边，并且堆叠了N层的网络（不是有多个注意力机制，只是重复利用相同的一个初始化权重矩阵，重复计算了多次）。计算使用的初值为第一个任务提取多模态信息，通过使用提示调参，经过多层Transformer得到的。此外，本模型没有计算各个节点的重要性，而是都考虑计算了所有相关联的节点信息进来，所以公式变为如下，

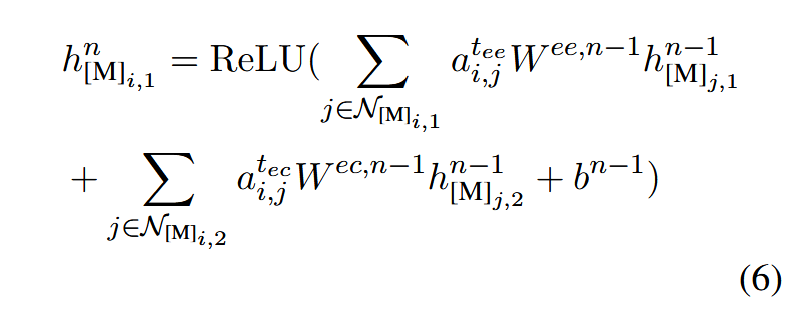
各个节点的贡献通过线性函数aWh\*b， W\*h先转换形状，a\*WH计算出两个节点是否有边相连，是否可以提供贡献值，最后加上偏置项B。其余计算同理。其中h代表了经过了Transformer后融合了多模态的隐藏层状态。（这里应该和第一篇论文一样吧，只使用了编码器部分，且经过多个块，or多次编码。得到每个输入的隐藏层状态重复多层，最后得到第二个任务需要的输入。）

对于最底层的话语层，只有相互窗口为2的上下文话语影响。下面三个公式的Nui都代表了当前位置所包括的上下文部分是那些点



对于第二层，原因层，多了来自话语层，和情绪层的影响，（不过按照画的图来说，情绪层和原因层的影响窗口是否需要对应的变小呢。。还不太了解，得看看源码是怎么写的）



第三层，情绪层  


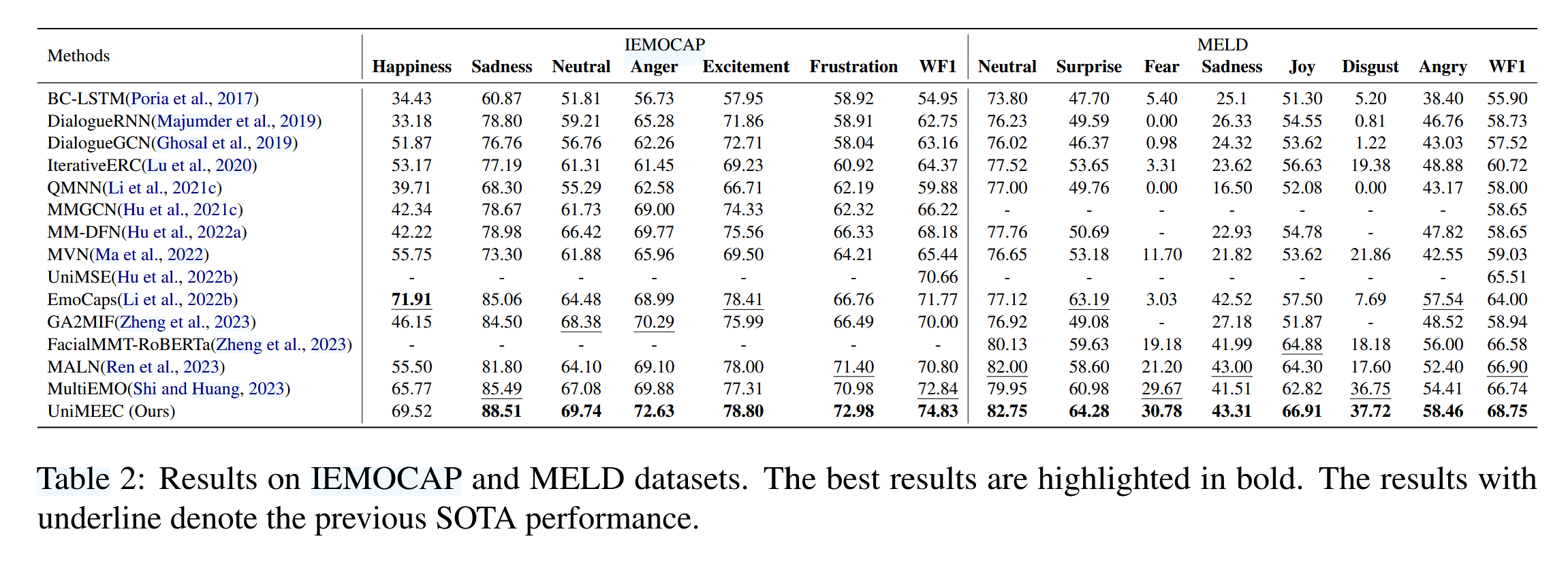
重复计算N次后得到当前话语Mask1，Mask2的最终表示。最后利用一个全连接层，

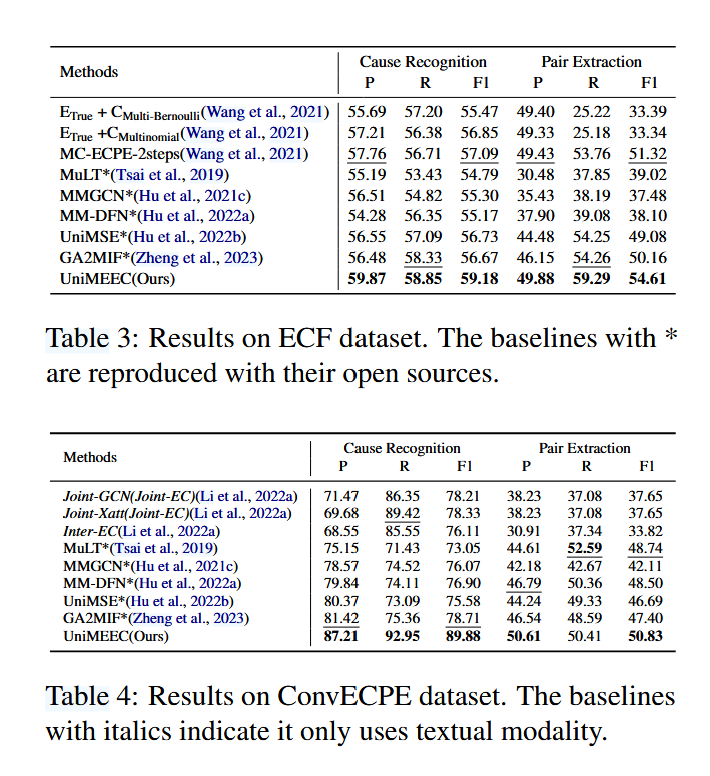


进行结果的分类，Mask1，代表了MERC任务，即当前话语的情绪是什么，Mask2，代表了MECPE任务，即当前话语情绪产生的原因是那一句话。相应的，从提示学习中获得的分类的损失函数也使用交叉熵，只不过需要将损失之和合并起来。

实验过程：

选用MELD，IEMOCAP作为MERC任务数据集，验证提取情绪是否正确，验证指标为准确性和加权F1值，对于MECPC任务，选用ECF数据集，验证指标为精度，召回率和F1值。





基于比对发现，提出的UniMEEC模型在整体的WF1分数均得到了提高，此外还提高了各个情绪的识别准确率，改善的原因可能是MERC和MECPE的统一框架提供了更多的辅助信息，增强了情感和情感原因之间的互动，从而缓解了IEMOCAP和MELD数据集的标签不平衡，而且提示的调参语句中也给出了反应情绪和事件的因果关系，说明此方法的有效性。同样的在情绪-原因对的提取也获得了提升，并且在消融实验中和对比实验中也再一次证实了多模态信息进行融合是有效的，并且音频信息相较于视频信息来说，提升更大，（但是基于第一篇论文文末的提出的问题，也有可能是当前的方法还不足以更好的感知或是说利用视觉信息导致的。第一篇论文使用3dcnn提取的视频信息，本文是efficient-Net预训练模型，）

参数设置情况：  
所有实验均在 NVIDIA RTX A100 中进行。我们以 BERT 作为基于 Transformer 的模型，它有 110M 参数，包括 12 层、768 个隐藏维度和 12 个头（这里可以看看具体代码）。我们使用以前的 Nt = 9 Transformer 层作为特定于文本的编码器，使用以下 Na = 2 和 Nv = 1 Transformer 层分别作为特定于音频的编码器和特定于视频的编码器。Nt、Na 和 Nv 的值由模型在有效测试集中的性能决定。此外，我们采用线性衰减学习率时间表和热身策略。

结果表明，统一两者能提高情绪识别的准确度。消融实验也表明，联合训练能提高整体模型性能，并验证了融合声学信息和视频信息的重要性

Improving Representation with Hierarchical Contrastive Learning for Emotion-cause Pair Extraction（TAC，ccf-b，2024）

通过分层对比学习改善情绪-原因对提取的表征

背景：  
1.现有的大多数工作主要集中在情感从句和原因从句之间的语义关系上，而忽略了它们在表征空间中的内在统计关系（表征空间指的是单词或是句子的高维向量，内在统计关系可以说是表达同一个情绪的情绪话语和原因话语的表达，可能存在这于word-embedding这样的相同属性的名词之间的余弦相似度大的特点。）

现有工作对情感子句和原因子句之间的相对位置敏感，损害了模型的鲁棒性，这意味着当情感从句和原因从句在文本中的位置发生变化时，这些现有的方法可能会受到较大的影响。例如，如果一个模型习惯了在文本中先出现情感从句，然后紧接着是原因从句的这种结构，那么当遇到原因从句在前、情感从句在后的情况时，该模型可能就不能很好地处理或者分析它们之间的关系。

为了解决这两个问题，我们提出了一个分层对比学习框架 （HCL-ECPE），它从两个层次对表示进行分层对比学习。第一级是子句间对比学习 （ICCL），它通过互信息最大化在情感子句和原因子句之间执行。第二个级别是配对内对比学习 （IPCL），它通过对比预测编码 （CPC） 在子句表示和对表示之间执行。

先前工作：

ECPE早期工作集中在情感子句提取、原因子句提取和情绪-原因对提取之间的联合学习，最早提出的两步法，1、找情绪子句，原因子句。2、从所有可能配对找出真正的e-c pair。而最早提出的方法属于流水线工作法，如果选取了错误的子句，也会干扰到e-cpair的提取。

为此有的解决方案是有将情感子句提取、原因子句提取和情绪-原因对整合到统一的架构中，如第二篇论文UniMEEC提到的，通过提示调参的方法，给出一段模板话语，根据相应的输入需要他预测出情绪，原因，以及他们的关系。

本文根据它们的情绪子句和原因子句之间的互信息提取情绪-原因对的可行性。同时，情绪-原因对的互信息不太容易受到相对距离变化的影响（互信息（Mutual Information）是信息论中的一个概念，用于衡量两个随机变量之间的相互依赖程度。从本质上讲，它表示一个随机变量包含另一个随机变量的信息量，使用在NLP中能在不被距离干扰的情况下衡量情感 - 原因对中情感部分和原因部分的依赖程度。）受这项工作的启发，我们提出了一个分层对比学习框架 （HCL-ECPE） 来解决 ECPE 任务。分层对比学习被广泛应用于多模态情感分析社区，这些工作旨在通过引入对比学习损失作为模型的正则化器来增强模态之间的交互。

HCL-ECPE 从子句间和对内的角度分层正则化表示学习与对比损失。子句间关系表示同一 emotion-cause 对中 emotion 和 cause 子句之间的关系，旨在通过考虑它们在嵌入空间中的关系来促进子句表示的学习。与从句间关系相反，对内关系表示情感从句/原因从句与由它们组成的情感-原因对之间的关系，旨在通过调整从情感子句到情感-原因对或从原因子句到情感-原因对的配对路径来促进情感-原因对的学习。

方法动机：  
 1、情绪 - 原因对的互信息显著大于非情绪 - 原因对的互信息。

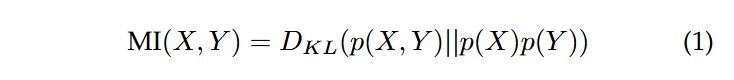
1. 情绪 - 原因对的互信息受其情绪从句和原因从句相对距离的影响较小。

提出了一个分层对比学习框架 （HCL-ECPE） 来正则化子句和配对表示学习。一方面，具有互信息最大化的模型可以捕捉到情感子句和原因子句之间深刻的统计相关性。另一方面，具有互信息最大化的模型减轻了模型对情感子句和原因子句之间相对位置变化的脆弱性。HCL-ECPE 使用对比损失作为正则化项，以限制情感子句和原因子句在特征空间中的表示。HCL-ECPE 强调从互信息的角度来表达子句间和对内统计关系，而不是以前的工作广泛使用的相对位置。（比如上文中UniMEEC，利用具有上下文窗口的图注意力网络，还是限制了原因从句和情绪从句之间存在一定的位置关系）

对比学习可以提高表征的学习能力，遵循正样本对拉近，负样本对拉远，对比学习是一种自监督学习方法，它的核心思想是通过学习对相似和不相似的数据样本进行区分，从而学习到数据的有效表示。为此本文提出的正样本是已标注好的情绪-原因对，负样本则是从已标注好的情绪-原因对中错误排序得到。

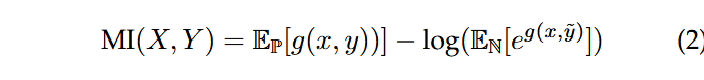
而对比学习需要学习到样本的表示，转移到NLP的情绪-原因对分析任务中来，就分为了两个任务1、子句间对比学习，表示好的句子中的正确配对和错误配对中学习

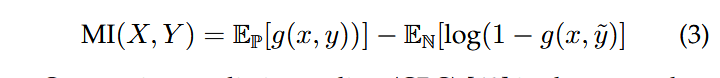
与之对应的，本文使用互信息作为对比学习的损失函数，因为能减少位置信息对结果的干扰。

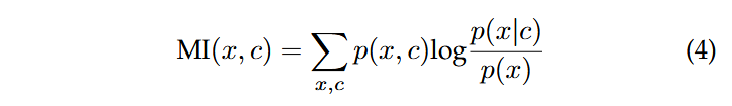
其定义为

其中D KL是联合分布p(X,Y)与X和Y的边缘分布乘积之间的 Kullback - Leibler（KL）散度。

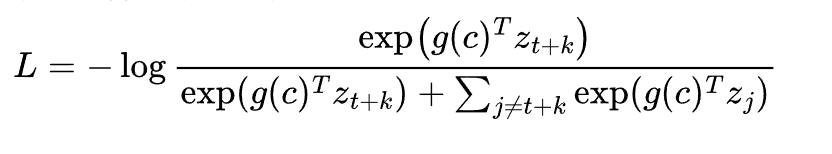
但是精确计算互信息是很困难的，为此需要使用一些近似估计的方法如神经近似方法 MINE（Mutual Information Neural Estimation）。

MINE 通过训练一个分类器来区分样本是来自和的联合分布还是来自它们边缘分布的乘积来估计互信息。MINE 使用基于 KL - 散度的 Donsker - Varadhan（DV）表示的 MI 下界作为互信息的近似估计，公式如下：

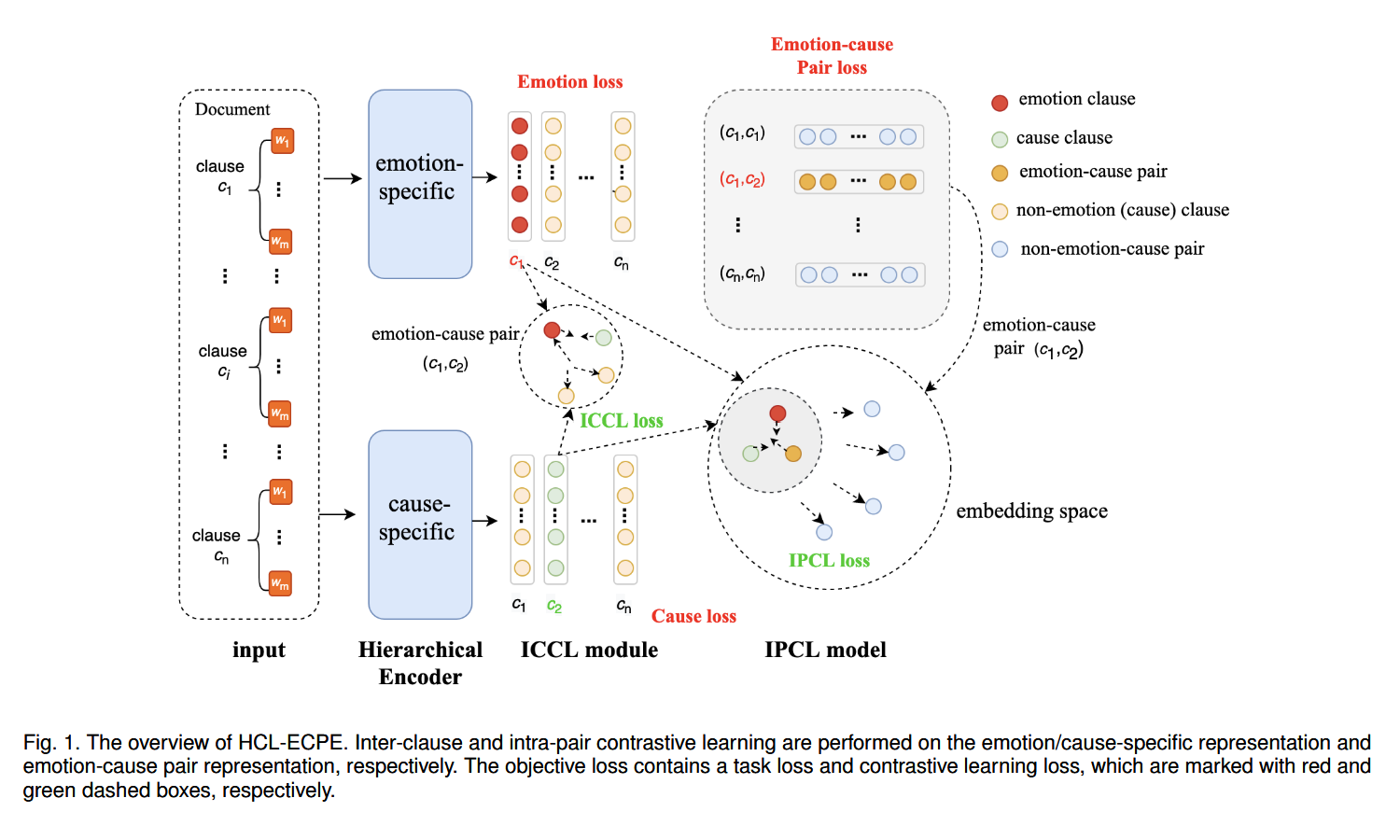
其中Ep和En分别表示对正样本和负样本的期望，G(x,y)是一个判别函数,比如本文就是使用一个带可学习矩阵神经网络，经过sigmoid激活函数得到。或者也可以使用JS散度代替KL散度，公式如下： 至此为子句间对比学习所用公式。

对于情绪-原因对间对比学习，使用对比预测编码，公式如下 

在实际使用中，操作流程如下，假设有一个序列数据x1,x2,x3,...。先使用一个编码器将其变为特征表示，对应于本文的分层编码器部分。对于序列中的每个位置t，有一个上下文Ct（过去几个时间步的编码表示），以及一个目标x[t+k]表示要预测的编码表示。CPC就是需要通过一个神经网络学习得到预测编码表示，其损失函数为



整体架构：



1、首先是输入部分，由文本中的从句构成，每个从句有任意个单词。

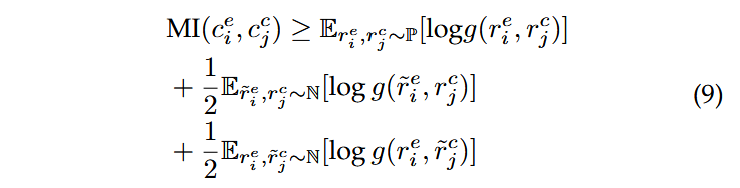
2、接着进入两个分层编码器。每个分层编码器包含词级编码和从句级编码。对于词级编码来说需要添加注意力机制，输出的结果就是获取一个句子的状态向量（这个操作与MeiJU2025代码中的LSTM很类似，第一步需要经过word-embedding比如BERT的模型和分词器，得到每一个单词的词嵌入向量，之后经过BILSTM，得到每一个从句中词语的隐藏层向量，进而得到整个从句的隐藏层状态输出，之后需要经过一个注意力矩阵，（它是一个不修改隐藏层维度r\_out的矩阵）计算后得到[batch\_size,seq\_len,2\*hidden\_state]的大小，并用tanh的激活函数激活，然后将其与一个类似与查询矩阵的[batch\_size,2\*hidden\_state，1]的大小的向量相乘并用softmax优化后，得到每个词语对整个从句的贡献度[batch\_size,seq\_len,1]。最后将贡献度乘上隐藏层状态r\_out得到[batch\_size,1,2\*hidden\_state],并在第二个维度上求和，得到一个二维向量【batch\_size,hidden\_size】，这就是词语编码器的结果） 具体描述如下lstm\_output \* softmax(u \* tanh(W \* lstm\_output + b))

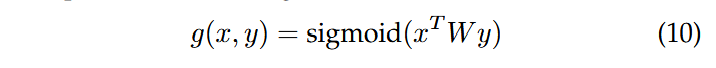
子句级的编码器需要将上一层得到的子句的隐藏层状态输入BI-LSTM或是BERT - base -chinese 来模拟文档中子句的顺序关系，（这里想看一下输入的维度大小是什么样的，一般都是直接句子[batch\_size,seq\_len,hidden\_state]进去，而现在少一维再进去如何呢，扩充维度吗）生成表示为r，然后使用两个不同的前馈网络，（全连接层）将每个从句转换为特定于情感的表示和特定于原因的表示并设置dropout值，就是找出一段话语中包含的情感从句和原因从句。从图中来看就是先经过第一步将一句话得到一个hidden\_state再经过一个前馈网络得到情绪或是原因的表示，大小的话没有说明。

疑惑点1.是不是每个句子都要经过情绪前馈神经网络F1和原因神经网络F2得到一个表示。（这里如果按照最后的来看的话这里还只是得到句子表示，并没有判断一个句子是情绪还是原因）

2.接下来的对比学习是否是用于更新这个神经网络F1，F2的参数，如果前一轮的经过注意力提取得到的隐藏层状态经过BiLSTM变成句子表示的网络要不要更新，再往前推，一开始进行词语编码的BiLSTM要不要更新。

3、子句间的对比学习，目标是最大化情感子句和原因子句表示之间的互信息从而增强它们的相关性，所以他的数据来源是在第二步中得到的关于每个子句的表示中标注为情绪和原因的句子。因为正样本是正确配对，负样本为错误配对。每个训练批次只有M个情绪-原因对{Ce,Cc}（从训练集的话语中标注得到，就是不知道需不需要再同一轮对话中）。然后使用基于上一轮中得到的每个子句的情绪表示和原因表示{Re,Rc}作为正样本。然后通过打乱同一批次中的情感 - 原因对来生成负例，使得一个情感从句随机与批次中的一个原因从句相关联。每个样本有两种负例情况，1、情绪-错误原因，2、错误情绪-原因。之后通过公式估计互信息的值下界



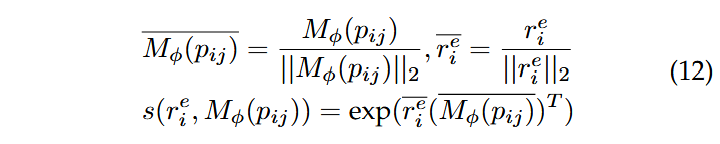


其中的g表示一个二分类器，W是一个可学习的矩阵，用于判断从句对（Ci，Cj）是否为正例，之后对三者情况，一个正样本，两个随机打乱得到的负样本，进行加权求和算期望。得到从句对（Ci，Cj）互信息的下界，计算批次中的M对后得到的平均值作为ICCL的损失，进而优化情绪表示和原因表示的提取效果。想要达成的效果就是情绪子句和原因子句的表示应该尽可能接近，与非情绪子句和原因子句的表示应该尽可能远。

4、情绪-原因对内对比学习，学习从情感子句或原因子句到它们组成的情感-原因对的路径，通过利用它们之间的信息特征，进而调整情感子句或原因子句与情感 - 原因对在嵌入空间中的相对位置来改进表征学习。所以其抽取的数据是所有标注出来的情绪-原因对数据。目的是将情绪-原因对的表示与其包含的情绪句子，原因句子表示尽可能靠近，与其他的情绪-原因对表示尽可能远离。

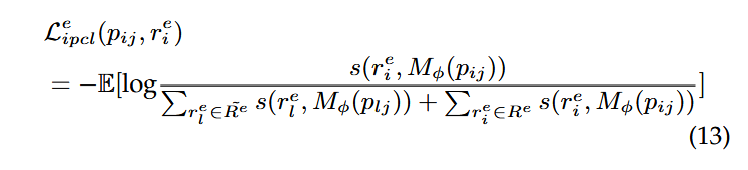
具体实现方式的第一步是生成一个子句对编码器函数F，也就是输入内容为我们标注好的对应的情绪-原因对中的情绪子句表示rie，和原因子句表示ric，经过一个参数化网络，（计算一个可训练的网络）得到输出为情绪-原因对的表示，从而得到一个情绪子句rie和原因子句ric到情绪-原因对的生成路径F，现在期望构建一个反向路径，给出一个情绪-原因对表示，能得到情绪子句表示rie，和原因子句表示ric。为此需要使用CPC分数函数测量情感子句和情感 - 原因对的相关性，或是原因子句和情感 - 原因对的相关性。

以情绪子句rie为例子，公式第一部分如下：

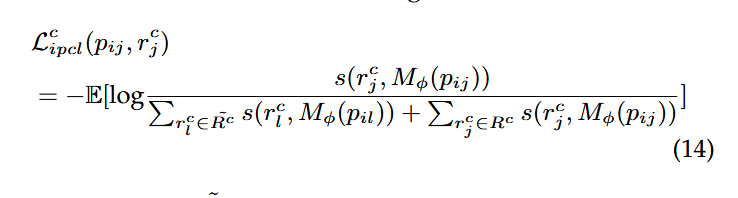


其中M是一个可学习的网络，输入为情感-原因对的表示，输出为我们需要的情感子句表示rie，然后使用一个欧几里得2范数进行归一化，rie则是当前选定的情感-原因对中情绪子句rie的真实表示。按照上文所属的话，rie的维度应该是[batch\_size,hidden\_state]，所以两者转置相乘得到一个数字，就是CPC分数函数的一部分。

第二部分：将同一批次（这个的意思应该是指在子句间对比学习提到的一批次下有M个情绪-原因对中所包含的各个情绪表示）下的所有其他情绪子句表示，记为~Re，Plj代表了子句间对比学习中每个情绪子句构造的错误匹配的原因句子。分子就是唯一的正样本，分母就是负样本和正样本之和



第三部分，同理计算关于原因子句的信息

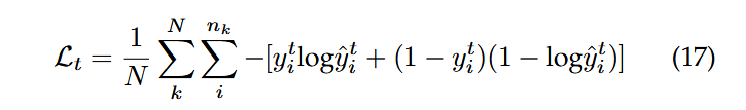


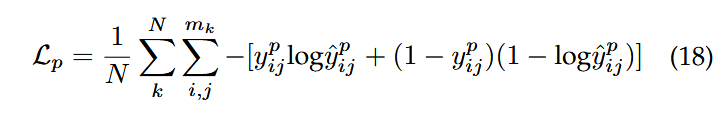
1. 通过超参数平衡两者关于损失函数的占比，进而优化输入为情绪-原因对pij，输出为情绪子句表示，或是原因子句表示的网络M，和输入为情绪子句表示和原因子句表示，输出为情绪-原因对表示的网络F

CPC 评分作用：对比预测编码（CPC）通过分数函数来衡量情感子句（或原因子句）与情感 - 原因对之间的相关性，从而确保情感或原因子句表示的重要信息被情感 - 原因对表示接收。并且反向优化情绪子句或是原因子句的表示。

最后一个部分是对于情绪，原因，情绪-原因对的预测（应该是指通过2步中的子句编码器得到的情绪表示rie，原因表示ric，和从第4步训练得到的情绪-原因对表示pij）将他们用一个前馈神经网络和softmax激活函数得到一个句子是不是情绪，原因，或是情绪-原因对。

情绪，原因子句预测的损失函数如下，是交叉熵函数，N表示输入的文本个数，nk表示一个文本里面包含的子句个数。



情绪-原因对的损失函数如下：

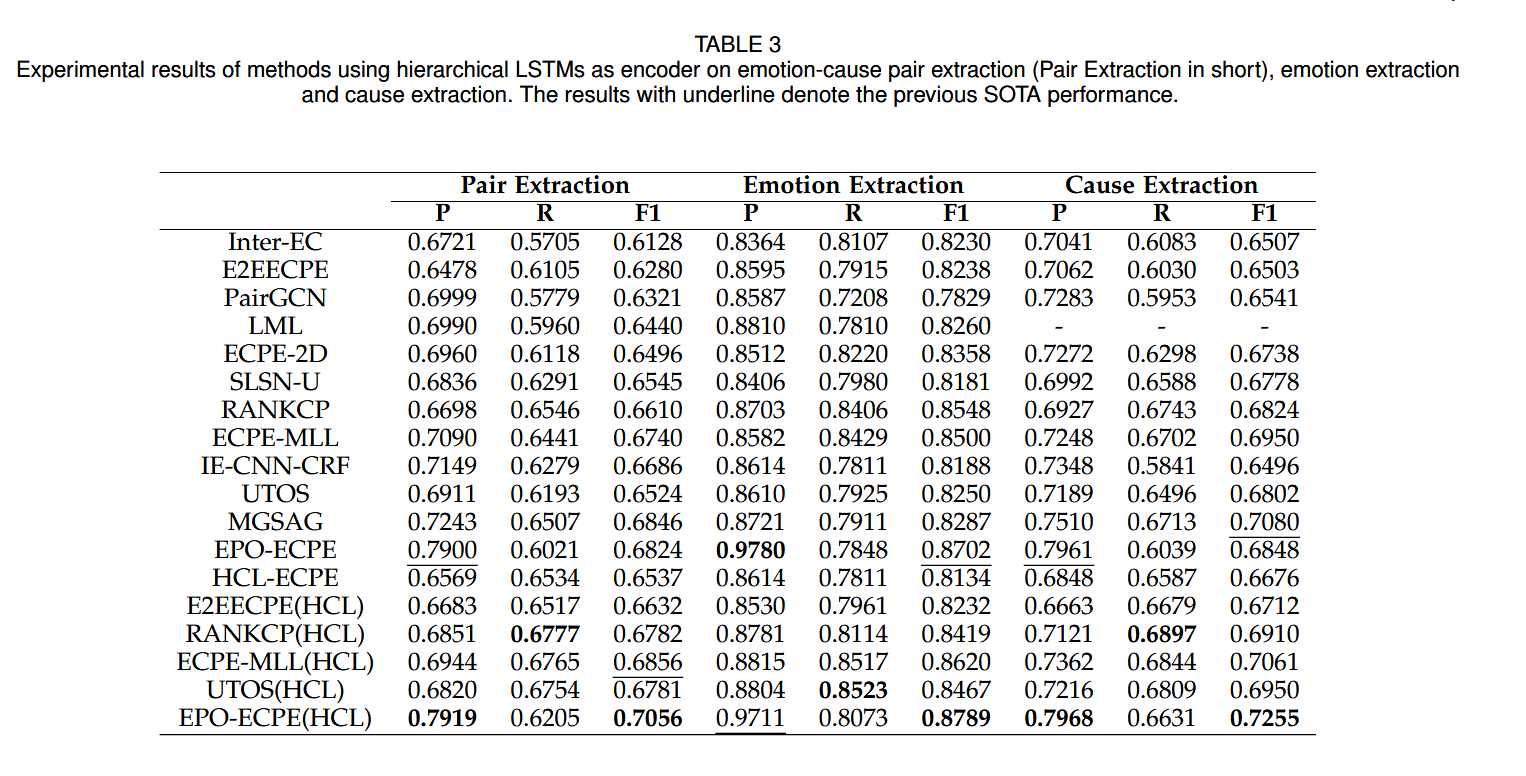
前面为止的内容都是常见方法，从表示经过分类得到预测结果。

但是本文还可以将上文提到的子句间对比学习损失和情绪-原因对的对比学习损失一起作为整体的损失。所以本篇提出的方法旨在通过对比学习的方法提升从文本这个模态学习得到的情绪子句表示，原因子句表示，情绪-原因对表示，

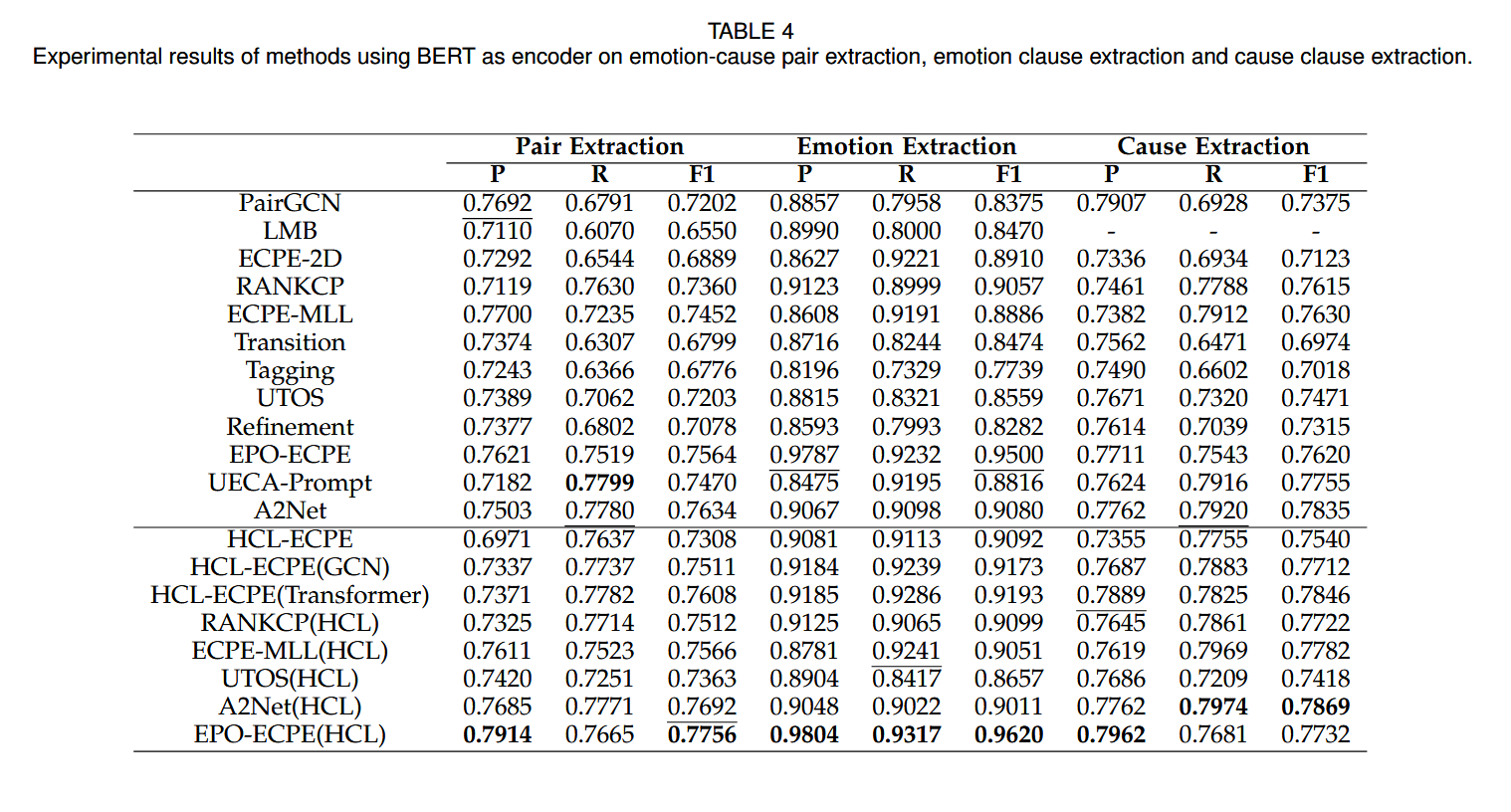
实验部分

预设值和参数部分，1、首先使用了维度200的预训练的Word2vec模型作为词嵌入，同时BERT-Base Chinese的预训练模型。2、对于字词编码，子词间Bi\_LSTM，从句级别编码（Bi\_LSTM）和最后的情绪从句预测，原因从句预测，情绪-原因对预测的前馈神经网络的dropout值进行了设定，以及分层学习中使用到的超参数。

数据集使用ECPED和RECCON，使用10折交叉验证，

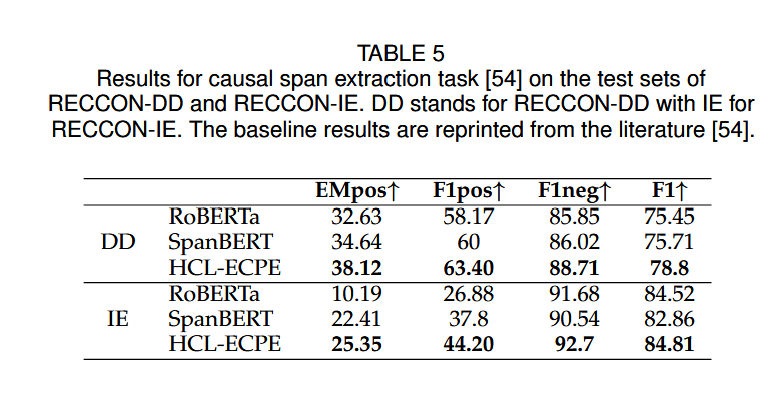


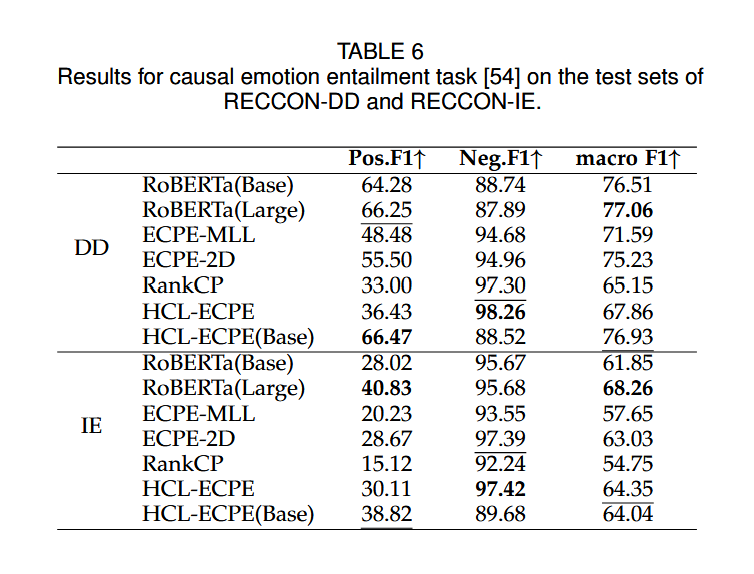
本结果是在没有使用BERT作为编码器的时候，（这个应该是指从句级编码器的部分），可以发现例如Inter-EC这类的两步设置架构都会比使用统一架构的性能低一点，此外如果将本论文提到的ICCL（子句间对比学习）模块，IPCL（情绪-原因对的对比学习）



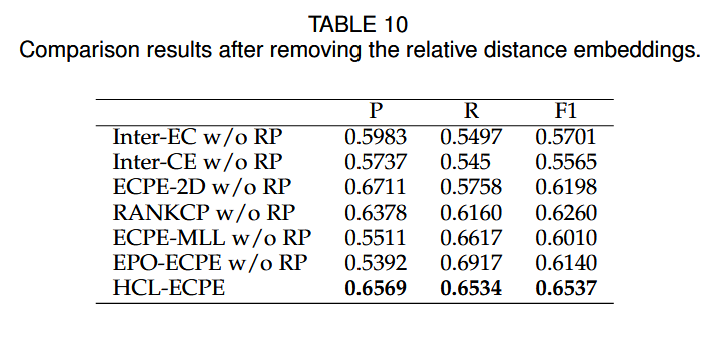
对于使用BERT预训练模型构建句子表示编码器的时候，效果会比使用BiLSTM更好，此外由于本文在句子表示编码时只用了一层，且用了正则化，效果相较于多层的Transformer的编码器，或者是使用图注意力网络的RANKCP编码器有所不足。所以再改进了这个部分后，本文的预测精度再一次得到了提升。

此外还检测了不同模型的因果跨度提取性能和因果情感蕴含的结果，HCL-ECPE从对话中提取带有上下文信息的因果跨度，与 RoBERTa 和 SpanBERT 相比，HCL-ECPE 获得了明显的边缘改善。此外还进行了因果情感蕴含任务的判断，此任务旨在预测对话历史记录中的哪些特定话语负责给定的目标非中性话语，或者是给定一个原因话语，判断它是否可以时某种情绪产生的原因。结果发现，由于编码器过于简单，会导致本模型精度下降，但提升编码器性能后，模型整体性能也得到提高





之后测试了各个模型对于相对位置的敏感性，将其他模型去除相对位置嵌入（定义：相对距离嵌入是将数据点之间的相对距离信息编码为特征向量的一种方法。它试图在特征空间中反映出原始数据中不同元素之间的相对距离关系。目的：增强模型对数据结构的理解和表达能力，尤其是在处理具有空间或顺序关系的数据时。比如Transformer中的位置编码）后，均导致了模型预测准确度的下降。与这些模型相比，我们提出的模型对相对距离的敏感度较低，因为它依赖于情感子句和原因子句之间的统计关系，而不是它们的位置关系，因此表现出更稳定的性能。这些结果表明，从互信息的角度提取情绪-原因对是有效的，可以提高模型的鲁棒性。



结论：  
 分层对比学习就是多出了使用互信息的类似感念，定义了新的损失函数来评测1.原因句子和情绪句子之间的表示接近，加强对他们的表示信息。2.情绪-原因对和情绪句子，原因句子之间的表达应该比其他情绪-原因对更接近，并且使用了双向的表示网络

Hierarchical Shared Encoder with Task-specific Transformer Layer Selection for Emotion-Cause Pair Extraction（TAC，ccf-b，2024）

第一段介绍了三种ECPE方法策略，他们都有不足

1、完全配对方法，完全配对方法将所有对视为一个对矩阵，其中大小与文档长度呈二次方关系。由于 ECPE 基准语料库中有效对的平均比例小于 1：200，因此全配对方法存在标签不平衡问题，随着可选择的情绪句子，原因句子的增多，可能会判断更多错误的配对，导致弱对提取。

2、滑动窗口方法，基于以下假设：大多数原因子句要么紧接在其相应的 emotion 子句之前/之后出现，要么是 emotion 子句本身。因此，这些方法忽略了遥远子句之间的有效因果关系，如shi'y图注意力机制分析窗口为2的上下文

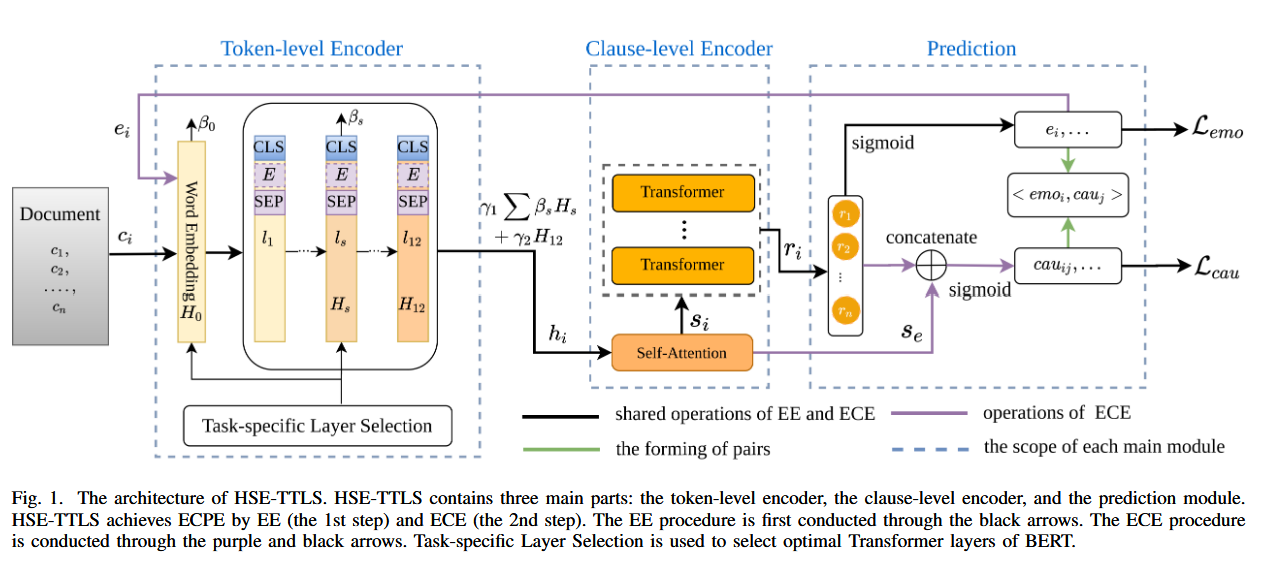
3、 设计独特标记方案的序列标记方法。

此外现有的ECPE工作中仍面临一些挑战：  
1、在文档中，许多子句可以包含具有情感倾向的事件，并且可以通过直接原因提取（从没有情感信息的文档中提取原因子句）作为原因子句来提取。但只有少数子句是与显式情感表达相关的 true cause 子句。简单的例子来说就是，今天下雨了，可能有人会因为下雨感到厌烦，但如果文本中根本没有人表现出具体的情绪，那么这个原因子句就是虚假的。所以原因和它引发的情绪是不可分割的，在不指定情绪的情况下提取原因是一种病态的。

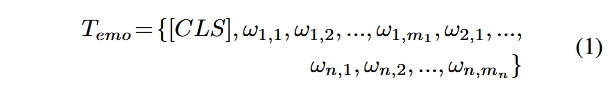
2、情绪和原因提取有不同的目标，这迫使他们专注于不同的特征。但是 emotion 子句和它各自的 cause 子句是相互指示的（就比如在上一篇论文中使用了互信息方法，让情绪句子和原因句子的表示信息尽可能靠近，远离其他句子的表示信息便体现了这一点）。统一编码器的共享特征编码学习混合特征，而不区分不同子任务的特征。相比之下，具有两个编码器来分别学习情绪和原因特定特征的方法忽略了子任务之间的交互。完全共享的编码器或完全并行的任务特定编码器都是次优的。 为了应对上述挑战，我们提出了一个名为 HSE-TTLS 的多任务学习框架。

其包含两个任务，1、情绪提取2、情绪 原因提取。第一步就是提取情绪句子2、然后，将每个 emotion 子句与整个文档进行编码，通过 ECE 任务找到其对应的原因。因为每次只考虑一个情绪句子，相较于完全配对方法更关注具体的信息。HSE-TTLS 采用分层共享的 Bidirectional Encoder Representation from Transformers （BERT）和特定于任务的 Transformer 层选择。与现有的完全共享编码器或完全并行任务特定的编码器不同，我们提出了一个任务特定的层选择，使共享编码器也能专注于与两个子任务中的每一个相关的重要信息。不进行原因子句的识别，就是为了保证原因一定是有着对于情绪，而非空穴来风。

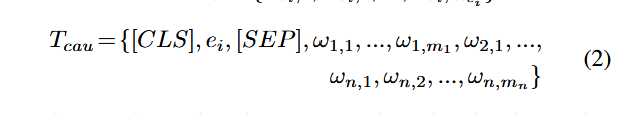
整体架构：包含三个部分 标记编码层，语句编码层，预测层。



第一部分是共享标记编码层，它的输入对于情感子句识别来说就是原始的文档D，而对于情感原因提取任务，输入就是情感句子（有个小疑问，第一轮的情绪是根据标注数据决定的吗，后面的轮数可能是依据情绪识别的结果作为输入的吧）以及文档D。同我们使用的BERT等模型一样，这里也包含CLS指定开头，SEP指示不同句子的特定标记。SEP只会使用在情感句子和文档的分割中。

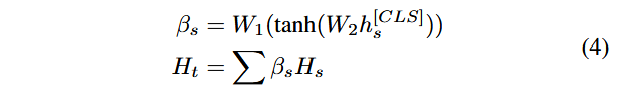
情感识别任务的输入 

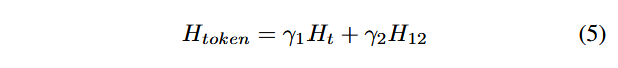
只在开头使用了特殊标记，不同子句之间不需要使用SEP特殊标记。

对于情感原因提取任务输入

获取到了各个特定任务需要的输入后，就可以把它都输入到BERT模型中了（好像需要拆成两部分，第一部分是预测情绪句子，第二步才把得到的情绪句子组合文本作为有一个输入），其包含12层BERT。BERT的中间层编码了丰富的语言信息层次，表面特征在底部，句法特征在中间，语义特征在顶部。而情绪识别和情感原因提取作为两个任务有着不同任务目标，所以需要获取不同的信息，但相应的两个任务又是密切相关的，将情绪-原因对的编码任务与情绪句子编码任务合并在一起，（不同于前文的基于情绪句子，原因句子，通过网络学习得到情绪-原因对表示方法）

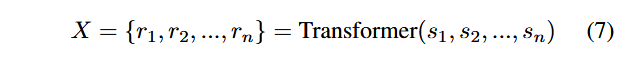
为了获得两个子任务的特定于任务的情感表示，我们选择不同的BERT层来生成两个子任务中每个子句的最终表示。对于情绪句子的识别来说，它应该多注重在寻找情绪关键词上，所以尽可能的在低层比如第0层寻找可用信息。除了情感关键词上，情感的表达也是可用的参考信息。一般情况下，情感关键词可能会与说话人（主语）相差较远，这会导致信息提取难度增加，所以本文使用的方法是判断主谓一致性的表现是否良好，并依据实验结果选择了BERT第八层。而在情绪-原因对的识别上，表示因果关系的连词是表明存在情绪-原因对的重要特征，所以选择第0层寻找字级信息。另一个特征是原因句子往往是一个事件，因此，事件的结构对于能否找到原因子句很重要，因此我们将 ECE 的层选择问题映射到结构信息学习，对应在第十层的信息有最深的结构信息。除此之外，也需要整合上最后的第十二层的信息，经过融合后得到句子中每个词的嵌入信息，公式如下：

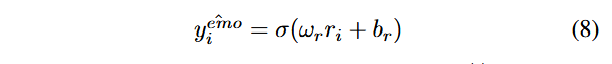


其中s表示当前是BERT的第s层，hs[CLS]表示第s层中CLS特殊符号的词嵌入表示，W1维度[d,1],W2维度[d,d]都是可学习矩阵，d表示的是词嵌入的维度大小，计算得到权重矩阵βs，维度[d,d]，接着对所有要求的BERT层计算后求和得到Ht，作为最终的表示信息。

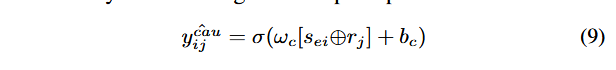
然后将得到的输出和最后一层的BERT输出，按照比例求和，得到每个句子中每个词语的嵌入表示。

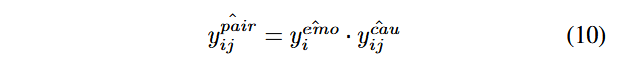
第二部分、对于句子级编码器，只需要将上文获取得到的当前句子的每个词嵌入送入自注意力机制，（情绪识别，和情绪-原因对识别都用同一个方法），然后将每个词的，乘上注意力机制的嵌入表示求和，得到这个句子的表示。（和上一篇论文中使用Bi-LSTM注意力的方法一样，或者说就是一个意思，这个方法很常见啊，感觉可用为基本方法了，只需要研究一下输入的问题）最后将得到的每个句子都输入进Transformer中得到每个句子的最终表示情况。



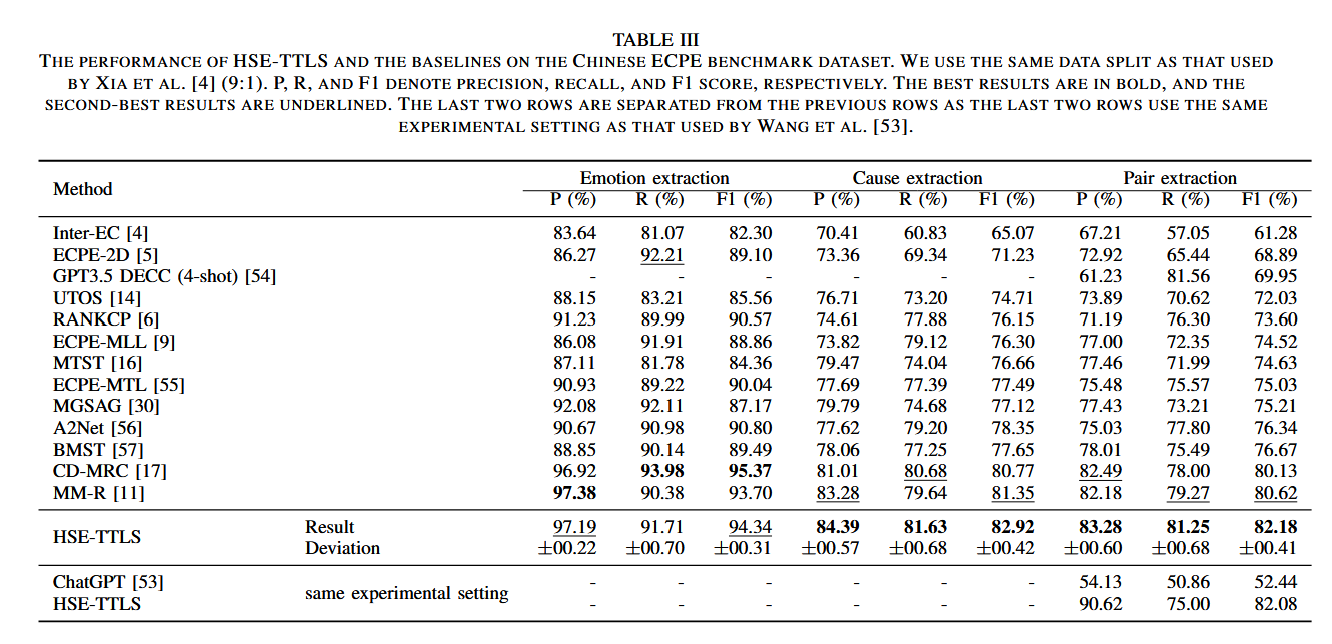
第三部分、预测和联合训练。对于情绪识别是一个二分类的前馈神经网络。并且为了提高检测准确率，引入情感词典 ANTUSD作为情感子句的过滤器，如果这个句子中没有包含任何一个情感词语，会被一个取值范围[0,1]之间的修正，最终的判断依据是，将概率大于0.5的视为真正的情感句子。结果记为Y[i,emo]。本部分的损失函数使用了FocalLoss是一个优化了交叉熵的损失函数，可以解决类别不平衡的问题。

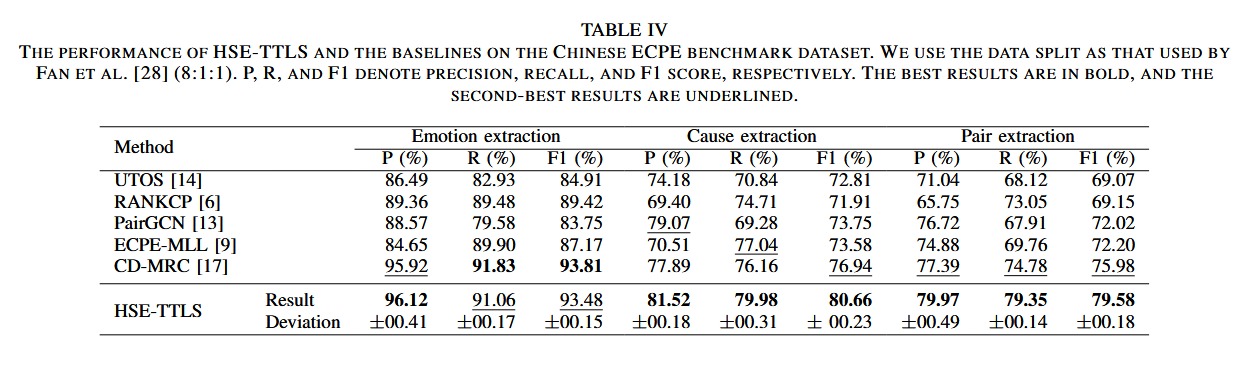
情绪-原因对的提取，首先需要找到每个情绪句子所对应的原因句子是那一句，为此将情绪句子Se与文档中的每一个句子进行concat，同样经过前馈神经网络，进行sigmoid，二分类计算得到记为Y[ij,cau]。



对于最终的对提取，以第 i 个子句为情感，第 j 个子句为相应原因的对的概率分布，Y[ij,Pair]是将Y[i,emo]\*Y[ij,cau]计算得到，如果结果大于0.5，则说明可以是一个情绪-原因对。损失函数使用交叉熵函数。

此外HSE-TTLS 采用联合学习策略来训练模型。联合学习是指同时学习多个任务或目标的机器学习范式。它训练模型同时执行多个相关任务，利用不同任务之间的共享信息来提高模型性能 。（类似的应该和那篇UniMEEC差不多吧，它是使用了提示微调的方法，让情绪类别和指示原因句子位置的任务合在一起。）本模型就是将情感句子识别和原因句子识别和情绪-原因对识别，使用同样的输入。

实验结果：  


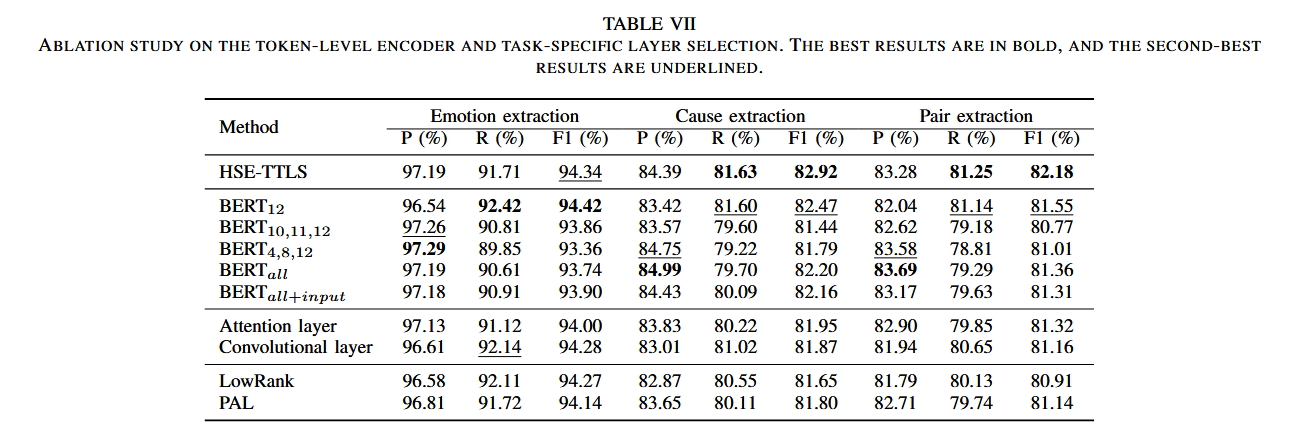


表中给出了给基准模型在中国基准数据集上不同划分方法的结果，发现本文提出的模型，在原因句子的提取的各个指数都高，



在英文数据集上的结果。

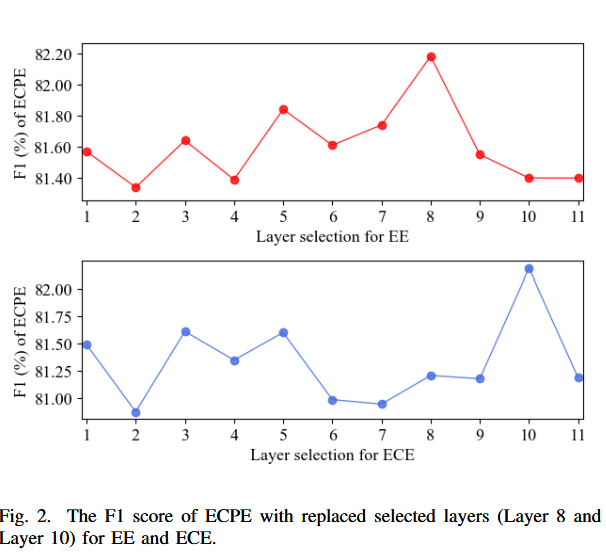
接着是实验不同在字词编码器中选择不同BERT层的结果。第一行是基准模型，第一部分是提取不同的BERT层的结果，第二部分是使用其他方法集成不同层的信息，原始方法是非12层方法使用了类似注意力机制的方法计算了比重，然后按比例添加其与12层的结果。第三部分是添加特定于任务的适配器以生成特定于任务的信息的变体。

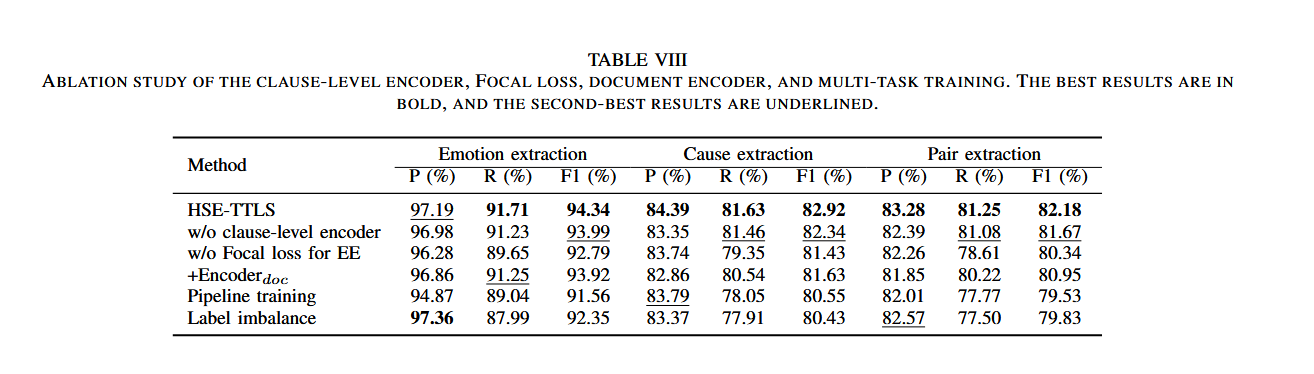
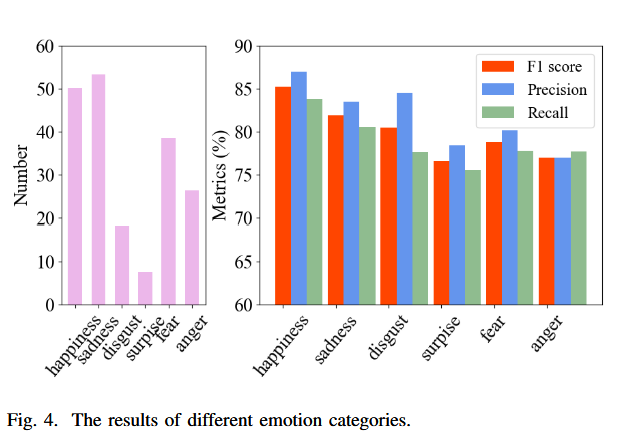


第一部分体现出了BERT中会有很多噪声数据，需要提取有用信息，

第二部分使用注意力机制，和3\*3卷积核处理最后一层和输入序列的隐藏层状态。

第三部分使用了相关处理BERT隐藏层状态的方法，但由于他们均是为了学习层与层之间的相关性，并没有很好的提取句子的内部信息。所以还是从BERT层提取特定层效果最好

 所以再次测试了不同层对结果的影响情况。

消融实验的结果，验证了FocalLoss对情绪句子的提取效果的提升，并进一步提高了情绪-原因对的提取准确度。也验证了文档编码器只汇总了文档信息，忽略了详细信息，反而降低了准确率。此外如果不进行情绪提取来判断情绪-原因对，也会使得准确率下降。、此外，即使使用不同的方法，在大多数论文中，不同的情绪的判别准确率都会有较大出入，初步判断是背景因素，或是常识的影响，比如说过度的夸赞可能不会让人开心。

最后再次给出了如何计算BERT每一层对特定任务的影响大小。为了进一步分析针对特定任务的层选择方法，引入了可学习参数a[i]作为每一层的权重，以确定在特定任务中哪些层对于模型性能的贡献更大。

初始设置与训练

首先将每一层的权重都设置为1，然后对模型进行训练。在训练过程中，将输入的令牌编码为具有动态变化权重的隐藏表示。

结论：在 ECPE 中，文件中情感的存在是清晰且毫无疑问的，具有明显的情感标记，但原因的存在是情感依赖性和不确定性的。因此，直接原因提取并不理想。此外，共享特征编码和两个并行编码器的特征编码都不能同时利用情感子句和原因子句的相互指示特征和任务特定特征。为了解决上述问题，编码器直接将整个文本作为一开始BERT的输入，进而得到词嵌入，在经过Transformer等操作得到句子的表示。我们将 ECPE 转化为两个子任务：EE 和 ECE，并提出了一个名为 HSE-TTLS 的框架。HSE-TTLS 避免了没有情感信息的直接原因提取，并根据 EE 和 ECE 的特点获得了不同级别的共享交互、情感相关和原因相关特征。HSE-TTLS 显著减少了 ECPE 中需要探索的对数，从而解决了标签不平衡问题。

Neutral Utterances are Also Causes: Enhancing Conversational Causal Emotion Entailment with Social Commonsense Knowledge

中性话语也是原因：用社会常识知识增强对话因果情感蕴涵

IJCAI ccf-a 2022

动机：

对话构建为图表，以克服原始蕴涵风格的隐式上下文建模并将情感信息融入图中，具体就是将一组对话信息构建成图表，对于情感信息，表示为情感嵌入并与这段话语中对应说话人的话语相连起来，情感信息可以显著促进对情感与目标话语相同的因果话语的检测。然而，即使有情绪信息，具有不同情绪的因果话语，尤其是中性的因果话语（中性因果话语占此类原因的 87%）仍然很难检测。可能的原因是模型在推理因果线索以及在话语之间传递线索方面能力有限。举例子来说就是难以检测与目标话语具有不同情绪的因果话语，论文中的例子是a指出了b的错误（情绪为中立），b因为给出了错误信息而感到悲伤或是其他的情绪。原因是模型在推理因果线索和在话语之间传递它们方面受到限制。

方案：  
 引入社会常识知识（CSK）并提出了一种基于图的结构来正确利用 CSK。具体来说，社会 CSK 可以推断一个人的社会互动，如心理状态和反应。为了恰当地利用知识，提出知识增强对话图（KEC）。KEC 通过边传递知识。由于知识应该与说话者的互动和情绪对齐，因此提出了一种说话者和情感实现的知识选择策略，可以为话语选择合适的知识。为了处理 KEC，扩展了用于 ERC 的有向无环图网络（DAG），构建了知识增强 DAG 网络。

此外，ECPE 和 ECE 都只是在文章中提取原因，没有扩展到对话场景。（但是后来的数据集也往多模态信息的数据集上实验了）对话由于流动的情绪动态和说话者的参与而更加复杂。RECCON（Recognizing Emotion Causes in Conversations）是一个用于识别对话中情绪原因的任务。

一、主要涉及的两个子任务

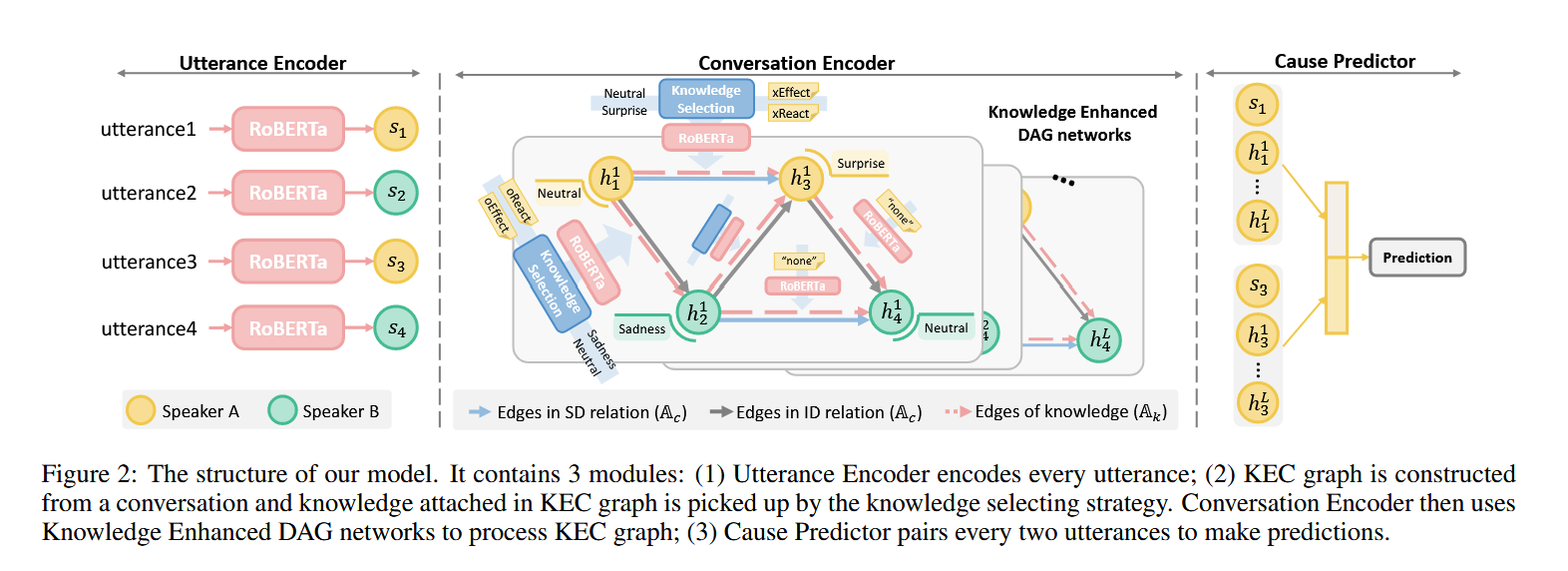
Conversational Causal Span Extraction (C2SE)：

任务重点在于从对话中提取因果跨度，被建模为机器阅读理解任务。它主要是找出对话中导致特定情绪产生的具体文本跨度。

Conversational Causal Emotion Entailment (C2E2)：

这是一个分类任务。具体来说，给定一段对话中的一个目标话语以及相应的情绪，任务是检测出对话中的哪些话语是该目标话语产生这种情绪的原因。例如，在一段对话中，已知第三个话语带有特定情绪，任务是从其他话语中找出导致第三个话语产生这种情绪的原因话语。

本文模型集中于解决C2E2问题。



模型组成：  
 先定义了任务类型，本文目的是从历史上下文中找到目标话语的因果话语。对话序列C=[U1,U2,....Un]，具有相应的情绪序列E = [e1,e2,....en],此外还有对应的说话人序列。，每个话语 ui 都与其上下文话语 uj （j ≤ i） 配对。（这里有点异议，原因也可以是在情绪之后的，这样判断是不是欠妥。比如一个人a正在悲伤，b询问为什么，a再回答原因。）如果 ui 是非中性话语，而 uj 是其因果话语，则对 （ui， uj） 标有 1。否则，（ui， uj） 标记为 0。

第一部分是编码器部分，这里使用了BERT的改进版，RoBERTa对对话中的每一句话语进行编码得到输出X。整体的话语表示需要将X进行最大池化并放入一个线性层，转换为指定的维度大小。得到句子的表示（上两篇文章都是将结果放入一个带注意力机制的LSTM中，那个用的好像比较多）。（目标话语指的是某一个情绪语句的原因语句）

第二部分是对话编码器部分。由于我们只需要判断一个话语的上下文，可以使用有向无环图来建模知识增强对话网络。（所以按照图中来看的话，他的上下文窗口只包含上文）它包含三个部分，V表示网络中话语的集合并且用rep属性记录由第一部分得到的每个句子的隐藏层状态，Ac 表示传统话语交互的邻接矩阵，Ak 表示话语之间传递的知识的邻接矩阵。

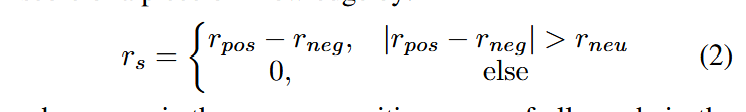
话语交互邻接矩阵：作用是将合适的其他话语的上下文信息丰富目标话语，（是指丰富从情绪句子找其对应原因句子中的 情绪句子信息还是其他待选的原因句子信息呢。）该邻接矩阵为三角形矩阵，包含两个属性，item，指示两个话语之间是否存在一条边，rel，存储这条边的关系类型。对于目标话语，更近的上下文，会比远的上下文更好，（这一点在UniMEEC模型中也验证了，但是又有一篇论文（分层对比学习）使用了互信息的概念，模糊掉具体的上下文选取范围。），所以使用大小为Wc的窗口大小在对话中滑动，只有处在窗口内的上下文话语才会有到目标话语的一条边。而具体的rel信息则分为自我依赖 （SD） 和说话人间依赖 （ID），如果是同一个人说的话语记为SD，否则就是ID。

知识传递邻接矩阵：同样也是一个三角形矩阵（所以遍历计算的时候只需要算一遍，就是只看后面的数据），有着item和klg两个属性，item同上，记录两个话语是否有联系如果上下文窗口中的i有情绪，item记为1，否则为0。Klg存储在边上附加的知识，使用的常识知识为COMET-ATOMIC（在第一篇论文MECPC中也提到了使用常识知识，是将生成的常识知识直接作为数据集的扩充）。本模型选择了四个关系：xEffect、xReact、oEffect 和 oReact。选择xEffect模型以5的束搜索生成知识。但是由于 COMET 不保证生成与情感相关的知识，因此有必要获取与目标话语的情感相关的知识。此外，知识传递应与演讲者的互动保持一致。因此，我们提出了一种知识选择策略，即情感实现并符合对话的条件。

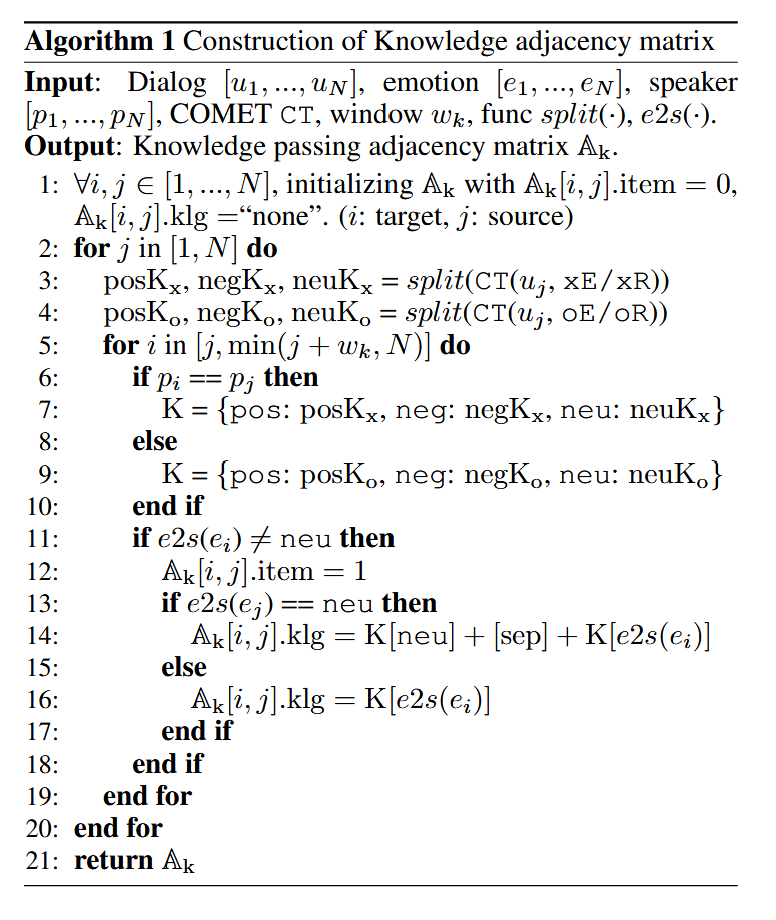
选取方案的制定策略：

说话人实现：如果目标 ui 的 pi 等于源 uj 的 pj，则选择 xEffect 和 xReact 的知识，否则选择 oEffect 和 oReact 的知识。（就是之前提到的话语交互邻接矩阵中记录了两个话语是不是同一个人说的）

情绪分析：为了实现生成的社会常识知识也带有情绪，就需要对生成的知识进行情绪识别，为话语Ui和Uj选择情感相关句子。识别方法采用了SentiWordNet，其通过用中性，积极，消极三种情感极性，对句子的每个单词进行评分。



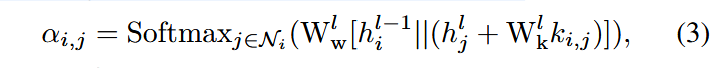
Rpos代表知识中所有单词的平均积极分数，Rneg就是消极，Rneu为中性，rs为计算结果，大于0认为是积极，0代表中性，小于0是消极的。然后根据极性评分将生成的所有知识分为三个集合，不同知识集合使用sep标记分割。如果有一个集合为空，则用none代表。对于话语中的情绪，幸福属于积极，中性属于中性，其余情绪归类消极。对于目标话语j，如果上下文窗口中的话语i有情绪，则A[k].item = 1如果目标话语j的情绪为中性，其所包含的社会常识知识为中性和上下文窗口中i的对应情绪的社会常识知识。如果目标话语j的情绪不是中性的，只需要上下文窗口中i的对应情绪的社会常识知识即可。



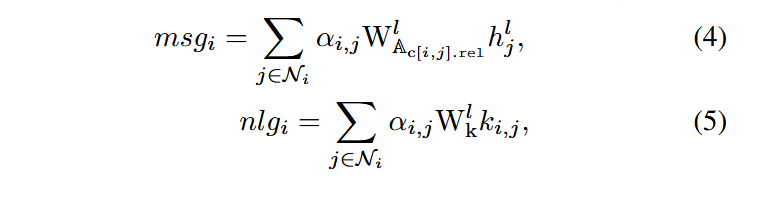
知识增强网络：包含l层，但每层信息共用。

首先对应每个上下文A[k].klg得到的附加知识同样经过RoBERTa的编码后得到k[i,j]，然后整体的话语表示需要将X进行最大池化并放入一个线性层，转换为指定的维度大小，得到一个句子的表示。

在l层的每一层中，对于一个目标话语i，需要计算它上下文窗口的相邻节点的权重信息。遍历顺序和计算Ak一样。

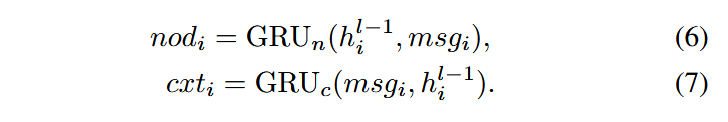


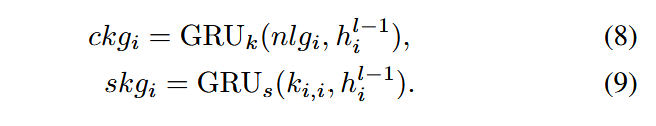
其中h[i,l-1]表示上一层i的节点表示，h[i,0]是通过公式初始化得到的，s[i]表示话语i的句子表示，维度大小为d[u]的隐藏层状态，eembe[i]表示话语i的情感 e[i] 的可训练嵌入，由 RoBERTa 的 MLM 头中的权重向量初始化。K[i,j]表示A[k].klg得到的附加知识同样经过RoBERTa的编码后得到的，其维度同样是d[u]。W[l,k]大小[du,du]，W[w,l]大小[1,2\*du]。所以结果为一个数。



之后通过计算得到的权重，在每一层l中再一次计算上下文信息和知识信息。Msg[i]的计算W[Ac[i,j].rel,l]表示的需要区分两个话语i,j是不是同一个人说的话语，需要分别计算。H[j,l]表示第l层第j节点表示。所有的维度都是[1,d[u]]。计算结果都是[1,d[u]].

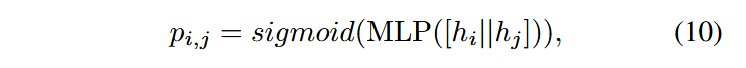
然后经过4个GRU（GRU（Gated Recurrent Unit，门控循环单元）是一种循环神经网络（RNN）的变体，就是那个LSTM改进版）





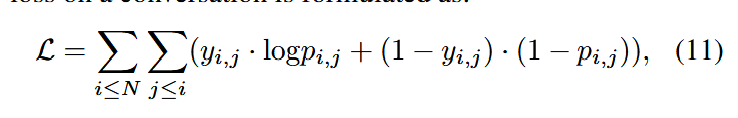
经过以上4个式子的计算后得到各类信息，首先输入的两个参数使用concat操作进行合并。Nodei节点信息，cxti上下文信息，ckgi上下文节点信息，skgi自循环信息。将4个信息加起来得到第l层第i个节点的节点表示。维度d[u]。

预测原因句子：  
最终的话语表示是将上下文编码器中l层的所有表示加在一起，维度仍然是d[u]。预测话语j是不是话语i的原因的公式如下：（不过这里没说需不需要判断话语i是不是情绪再计算。。）



这里首先将话语ui和uj的表示hi和hj进行拼接，然后通过一个多层感知机（MLP）进行映射，将拼接后的维度从2du映射到1。

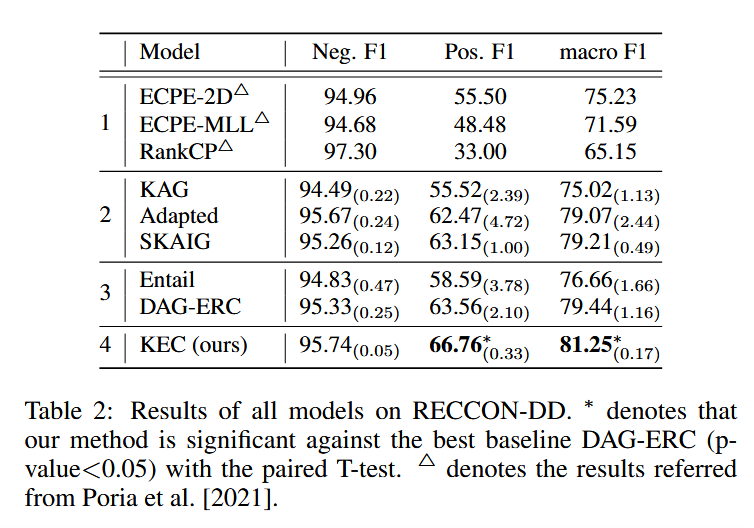
最后通过sigmoid函数将结果映射到[0, 1]区间，得到ui和uj之间存在因果关系的概率。损失函数使用交叉熵函数。



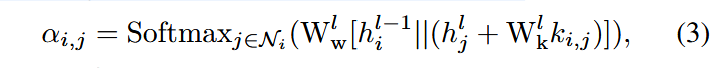
总结一下就是，社会常识知识能保证模型能理解带有与对应情绪带有不同情绪的原因句子是如何影响对应人物的情绪的，为了将合适的社会常识知识引入到合适的情绪-原因对中，设计了网络的连接方式，包括权重，信息表示，节点表示等的实现。一开始先把黑哥话语中的句子进行表示，维度d[u]。然后定义知识增强网络，定义其窗口大小Wc，然后计算好窗口内的话语之间是同一个人的对话，还是a对b说的话，记为Ac。之后遍历每一个话语的窗口，先依据是不是同一个人的话还是不同人对话生成社会常识知识，然后根据社会常识知识所表现的情绪极性，分出三个集合，之后判断原因句子是否包含情绪，以及情绪句子的情绪类型，赋值不同类型的社会常识知识。之后进入如何利用知识增强网络了，先是把每一条边上的社会常识知识进行嵌入表示，对于每一个节点根据它的话语和情绪借助RoBERTa同样进行嵌入表示。之后对每一层知识增强网络的每个Ac=1的节点对进行权重计算，信息计算，然后放入GRU网络计算各种表示，（它是怎么想到这样算的。。。）得到知识增强网络下一层的结果。最后将每层结果concat，放入神经网络计算话语j是否是话语i的原因。

实验过程：

（预训练模型使用RoBERTa-base）。我们使用 AdamW 优化器训练我们的模型，有 40 个 epoch，学习率为 3e-5，L2 正则化为 1e-4。批次设置为 4，我们以累加步长 2 对梯度进行步进。对于 Conversation Encoder，（就是编码一句话语的维度）我们将话语维度设置为 300，将 GRU 的隐藏大小设置为 300，（在计算CSK社会常识知识如何融入每个节点的表示中）将 DAG 网络的层数（就是中间的Conversation Encoder）设置为 5。对于情感嵌入，我们使用一个线性单元来映射 768 到 200 的维度（因为初始化是由RoBERTa的MLM头提供，需要一个转换）。对于 Cause Predictor，MLP (多层感知机)的维度设置为 [600， 300， 300， 1]，（输入是两个节点表示拼接而成）丢弃率设置为 0.1。对于 knowledge 和 context 的窗口大小，我们发现我们的模型在相同时获得了更好的性能，因此我们将它们设置为 2。

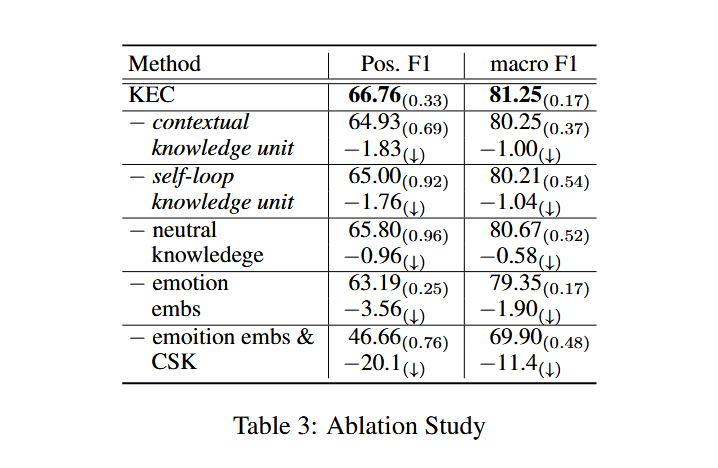


第 1 行是 ECPE 模型，第 2 行是使用 CSK 的模型，第 3 行是没有 CSK 的模型，第 4 行是我们的方法。发现ECPE模型任务和C2E2任务不同，而KAG基于新闻文章提出的知识增强网络，而对话相比文章更加简短，需要更复杂的模型从很短的句子中提取信息，并且KAG的知识仅用于计算边的权重，也就是只保留了



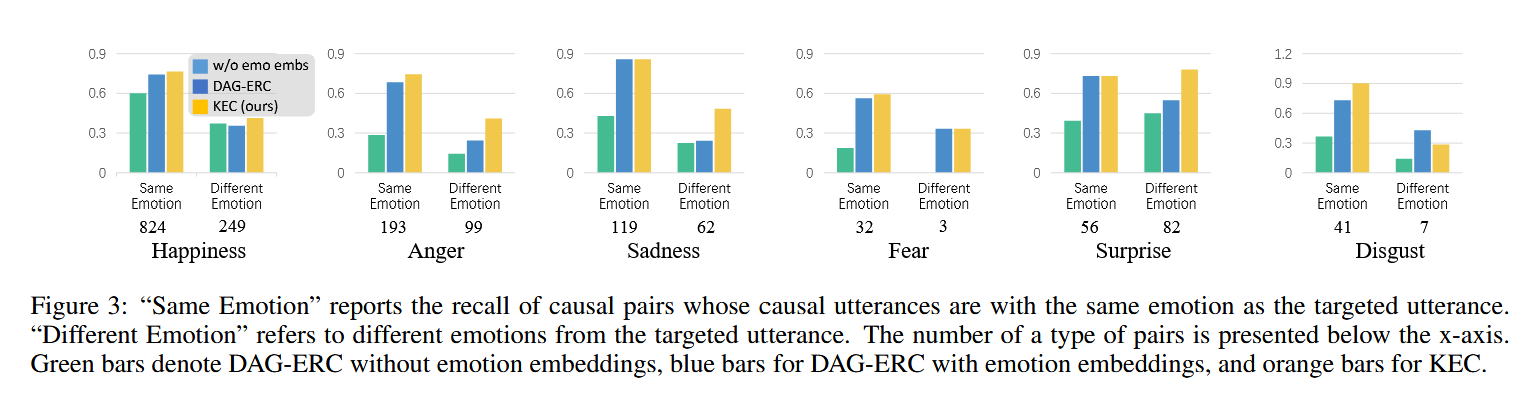
通过知识计算节点之间的权重，而缺失了后面信息通过边进行传播的过程。而对于Adapted和SKAIG模型，只是将知识直接放入话语中，缺少了选择合适信息的部分，也导致了模型精度不足。

并且消融测试了引入上下文窗口中的节点信息，和GRU计算的信息，模型精度大幅下降。而我们分别删除了上下文知识单元 GRUk、自循环知识单元 GRUs、在算法 1 的第 14 行中当源话语中是中性时添加的中性知识以及情感嵌入。



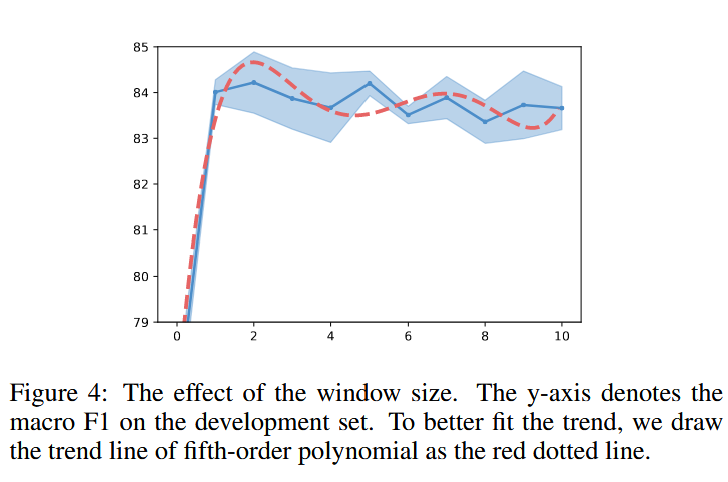
删除 GRUk 后性能下降表明从上下文中控制知识的必要性。至于删除 GRU 会影响性能，这表明目标话语的自我认知丰富至关重要，因为 37% 的因果话语是 RECCON 中的目标话语本身。通过去除中性知识，性能显示下降。这表明中立知识可以帮助模型更好地检测原因，因为它可以提供一些推理信息。去除情感嵌入时，性能仍然与配备情感嵌入的 DAG-ERC 相比具有竞争力。这可以归因于我们的知识选择策略可以为模型获取与情感相关的知识。由于情感嵌入和知识都被去除，模型变成了没有情感信息的 DAG-ERC。性能急剧下降，这验证了情感信息对 C2E2 很重要的说法。

此外还检测了一开始所述的模型是否可以促进检测情绪与目标话语不同的因果话语。结果如下：



可以发现如果将情绪嵌入融入每个节点所代表的句子表示中，再使用DAG图，会极大增加判断具有和情绪句子有着相同情绪的原因句子的准确度，使其配对，而使用了本文的模型，在此基础上引入社会常识知识，并且保留中性知识，则会极大增加判断判断具有和情绪句子有着不同情绪的原因句子的准确度，尤其是判断产生负面情绪的原因句子上。

上下文窗口大小的不同也会有影响



在本文中，我们研究了对话因果情感蕴涵。我们发现，由于模型的因果推理能力有限，在图中添加情感信息无法有效地检测与目标话语具有不同情绪的因果话语。为了缓解这种情况，我们引入了社会常识性知识，并提出了一种基于图的结构来正确利用知识。具体来说，我们提出了一个知识增强对话图 （KEC），它有一个设计的知识选择策略。为了处理 KEC，我们构建了知识增强的有向无环图网络。我们的方法优于基线，并从目标话语中检测到更多具有不同情绪的原因。

小结一下：  
第一篇论文MECPE中提出了总体的框架，多模态的信息识别情绪句子和原因句子，然后判断两者的所有组合而成的情绪-原因对是否识别正确。

在这个的基础上第二篇论文UniMEEC，将第一篇提到的两步任务合并为一步，不仅可以综合考虑内部信息，还避免了将第一步的错误识别信息传递到第二步的判断中来。并且提出了Prompt方法利用大模型来生成每个句子对应模态信息，情绪句子模态信息，原因句子模态信息。Concat他们的信息后变成图注意力网络中的节点信息，然后依据相关公式结合上下文节点（这个考虑上文和下文，第五篇就是只考虑上文）的信息得到新的节点信息。最后依据节点信息进行预测。

同样使用类似的方法的有第五篇论文Neutral Utterances are Also Causes: Enhancing Conversational Causal Emotion Entailment with Social Commonsense Knowledge，只使用了文本模态信息，并且转换任务为C2E2，简单说是给出情绪找原因。在编码得到每一个句子的表示之后，又加入了情绪嵌入作为信息补充，并且为了能判断出与情绪句子不具有相同情绪的原因子句，引入了社会常识知识（这个在MECPE中提到但没有用上）并基于此方法设计了KEC网络，每个句子依据是否为同一说话人，情绪类型，选择不同的社会常识知识，得到其表示。网络中的节点的初值则是每个句子的表示和其情绪嵌入（无论是不是情绪句子），经过多层的计算，融合由公式计算得到的上下文信息和自循环信息，得到最终的每个节点或者说句子的表示，然后依据情绪寻找它对应的原因是哪一句。

第四篇论文则提出了只提取情绪句子，再去找情绪原因对，也只是使用文本信息（感觉和第五篇论文的C2E2方法差不多，只不过情绪句子和类型需要自已找，而不是直接给出。）并且也是使用了一步出结果，情绪句子判别只需要给出文本，情绪-原因对只需要给出想判别的情绪句子和文本。将他们输入BERT中提取特定层信息按比例合并后得到每个句子的每个词的嵌入。然后利用得到的词嵌入经过自注意力机制和多层的Transformer后得到每个句子的表示。对于情绪句子判断，直接二分类得到，情绪-原因对的判断，需要将当前句子的表示与文本中的每一句话进行concat后，同样二分类判断，是不是原因句子。最后将两者概率相乘，如果大于0.5可以认为是情绪-原因对。

至于第三篇，我难以总结。