# 自然语言处理课程大作业报告

## 背景与方法调研

**1.课题目标：**

临床术语标准化方法是医学统计中不可或缺的一项方法。临床上，关于同一种诊断、手术、药品、检查、化验、症状等往往会有成百上千种不同的写法。标准化(归一)要解决的问题就是为临床上各种不同说法找到对应的标准说法。有了术语标准化的基础，研究人员才可对电子病历进行后续的统计分析。本质上，临床术语标准化方法也是语义相似度匹配任务的一种。但是由于原词表述方式过于多样，单一的匹配模型很难获得很好的效果。本次评测任务主要目标是针对中文申子病历中挖掘出的真实诊断实体，并以《国际疾病分类ICD-10北京临床版v601》词表为标准进行语义标准化。在给定一诊断原词，要求给出其对应的诊断标准词。

**2.课题重要性：**

（1）临床术语标准化的重要性在于确保医疗专业人员之间的有效沟通和信息交流。通过使用统一的标准术语，医生、护士和其他医疗工作者可以准确地描述和传达患者的病情、诊断和治疗计划。这有助于避免误解和混淆，提高医疗团队的协作效率。

（2）临床术语标准化还有助于促进医疗信息的互操作性：不同的医疗系统和机构可以使用相同的标准术语来记录和共享患者数据，从而实现数据的无缝集成和交换。这样，医疗专业人员可以更全面地了解患者的医疗历史和治疗情况，为患者提供更好的医疗护理。

（3）临床术语标准化对于提高医疗质量、减少错误和提高效率至关重要。它为医疗专业人员提供了一个共同的语言和框架，以确保患者得到准确、一致和高质量的医疗护理。

总之，把形式多样的医疗概念映射到标准的医疗术语编码，即临床术语标准化，对于疾病辅助诊疗、科研检索、疾病分组以及智能医保控费等研究具有重要的现实意义。以人工的方式进行标准化，不仅学习成本高，且效率低下。所以，研究一种自动的临床术语标准化方法对于推进医疗信息化建设以及减少人员的工作量、提高工作效率具有重要的现实意义。

**3.研究发展历程：**

现代临床术语标准化的算法和方法的发展历程涵盖了多个领域和技术。以下是一些重要的里程碑式的发展历程：

（1）机器学习：在临床术语标准化中，机器学习可以用于构建模型来自动识别和标注医学文本中的术语。通过训练算法使用已标注的数据集，机器学习可以学习术语的特征和上下文信息，并在新的文本中进行准确的术语标注。

（2）知识图谱：知识图谱是一种用于表示和组织知识的图形结构。在临床术语标准化中，知识图谱可以用于建立术语之间的关系和语义连接。通过将不同的术语和概念组织成图形结构，知识图谱可以提供更丰富的语义信息，帮助理解和解释医学文本中的术语。

（3）深度学习：深度学习是一种机器学习的分支，专注于使用多层神经网络来进行模式识别和特征提取。在临床术语标准化中，深度学习可以用于构建更复杂的模型来处理医学文本中的术语。通过使用深度神经网络，可以更好地捕捉术语之间的复杂关系和语义信息，提高术语标准化的准确性和效果。

随着技术的不断进步和医学知识的不断发展，临床术语标准化的算法和方法也在不断演进和改进。现代的算法和方法结合了多个技术和领域的成果，以提高术语的自动化标注和语义理解能力，为医疗信息的管理和应用提供更好的支持。

**4.可行算法：**

* word2vec，TF-IDF等无监督模型
* Siamese LSTM孪生模型
* SimCSE基于对比学习的模型
* BERT预训练模型fine-tuning
* 开源的大语言模型fine-tuning

## 方法或算法间比较与分析

1. **word2vec，TF-IDF等无监督模型:**

**Word2Vec：**

**优点：**Word2Vec 能够捕捉到临床术语的语义信息，将其映射到连续向量空间，从而实现语义相似度的计算。可以通过计算向量之间的相似度来度量术语之间的语义距离，从而进行标准化匹配。

**缺点：**Word2Vec 是基于无监督训练的，可能无法准确地捕捉到临床术语的特定语义，尤其是在特定领域和专业术语方面。Word2Vec 主要关注词级相似性，忽略了上下文和短语级别的语义信息，可能无法完全满足临床术语标准化任务的需求。

**实现难度：**Word2Vec 的实现相对较为简单，有现成的开源库和工具可供使用（如gensim）。

难点在于获取临床领域的大规模语料库进行训练，并调整模型参数以获得最佳性能。

**TF-IDF：**

**优点：**TF-IDF 可以评估临床术语的重要性，基于它们在文本中的频率和逆文档频率，从而进行语义相似度的计算。可以用于词级别的匹配和相似度计算，适用于简单的语义任务。

**缺点：**TF-IDF 忽略了单词之间的语义关系，无法捕捉到更高级别的语义相似性，尤其是在复杂的临床术语标准化任务中。对于具有不同长度和结构的术语，TF-IDF 可能无法提供准确的相似度度量。

**实现难度：**TF-IDF 的实现相对较简单，可以使用现有的库和工具进行计算（如scikit-learn）。

难点在于准备适当的语料库和调整权重参数，以便在临床术语标准化任务中获得良好的性能。

**可行性分析：**

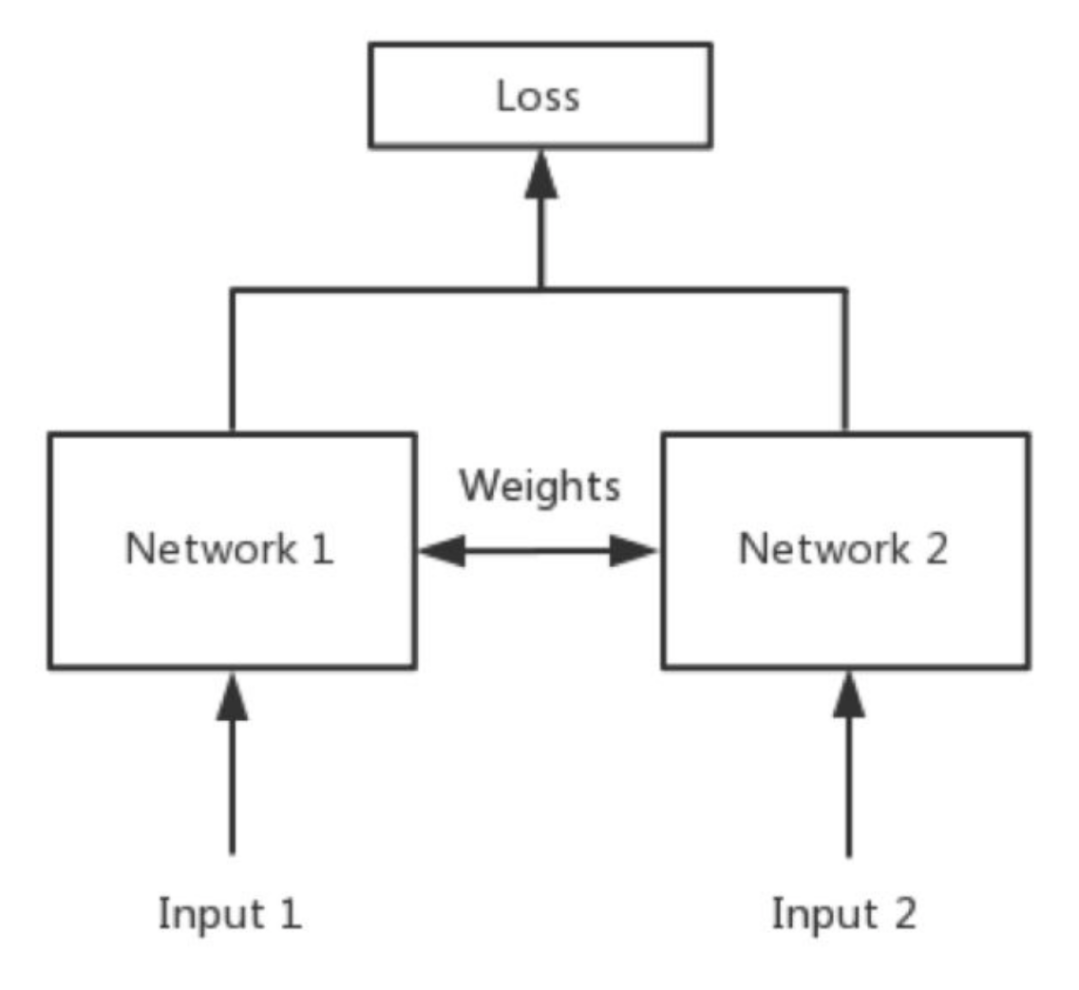
在临床术语标准化基于语义相似度匹配的任务中，Word2Vec 和 TF-IDF 可以作为一种可行的选择，具有一定的优势和局限性。

**需求分析：**如果任务要求较为精确的语义匹配和标准化，Word2Vec 可能更合适，因为它可以更好地捕捉到上下文和短语级别的语义信息。Word2Vec模型在中文文本中的应用已经得到广泛验证，可以用于词语的向量表示。然而,它无法直接处理语义标准化任务，**本项目需要得到句向量而非词向量，我们可以考虑将词向量叠加取平均作为句向量**，所以需要进一步的处理和匹配步骤来将词语映射到ICD-10词表。

**数据可用性：**对于 Word2Vec 和 TF-IDF，都需要大规模的临床领域语料库进行训练和计算，因此数据的可用性是一个重要考虑因素。

**领域专业性：**如果任务涉及特定的临床领域和专业术语，Word2Vec 可能更适合，因为它可以从大规模语料库中学习到更具体的语义信息。

1. **Siamese LSTM孪生模型:**



**优点：**

* 考虑上下文信息：Siamese LSTM能够捕捉临床术语的上下文信息，通过学习长短期依赖关系，可以更好地理解和表示术语的语义。
* 非线性建模：LSTM作为一种递归神经网络，可以建模复杂的语义关系，通过记忆单元和门控机制，有效处理长距离依赖和序列信息。
* 孪生结构：通过使用孪生结构，即共享参数的两个LSTM网络，可以将两个术语进行编码，并计算它们之间的相似度，从而实现语义相似度匹配。

**缺点：**

* 数据需求量大：Siamese LSTM模型通常需要大规模的标注数据进行训练，以获得较好的性能。在临床术语标准化任务中，获取大规模标注数据可能是一项挑战。
* 超参数调整：Siamese LSTM模型有多个超参数需要调整，如网络结构、学习率、正则化等，需要进行实验和调优，以获得最佳的性能。

**实现难度**：

* 模型搭建：实现Siamese LSTM需要熟悉深度学习框架（如TensorFlow、PyTorch等），并了解LSTM的原理和用法。
* 数据准备：需要准备一定规模的临床术语标注数据集，用于训练和评估Siamese LSTM模型。
* 超参数调优：需要进行适当的超参数调整和模型训练，以获得最佳的性能。

**可行性分析：**

* 数据可用性：要获取**足够规模的标注数据集，在本项目中数据集较少，所以训练效果不一定很好，可以通过爬虫等获取充足的语料信息，**如果有足够多的已标注的临床术语数据集可用，我们就可以用于训练和评估Siamese LSTM模型。
* 任务复杂性：Siamese LSTM适用于较为复杂的语义匹配任务，可以考虑使用该模型来解决临床术语标准化中的语义相似度匹配问题。但是**诊断原词和预测标准词都有多个的可能，所以在数据预处理的部分需要考虑好标签的方案以及预测的方案还有F1得分的计算方式**。

**3.SimCSE基于对比学习的模型：**

**优点：**

* 对比学习：SimCSE通过对比学习的方式进行训练，可以学习到临床术语的语义表示，使得相似的术语在嵌入空间中更加接近，从而实现语义相似度匹配。
* 无监督训练：SimCSE是一种无监督学习方法，不需要标注的匹配对数据，可以利用大规模的未标注数据进行训练，降低了数据标注的成本。
* 多任务学习：SimCSE可以同时学习多个任务，如语义相似度、语义匹配等，提高了模型的泛化能力。

**缺点：**

* 训练时间：**对比学习方法通常需要较长的训练时间才能收敛**，尤其是在大规模数据集上训练时。
* 预训练阶段：SimCSE在无/有监督预训练阶段需要使用大规模未标注数据，这可能对于临床术语标准化任务来说，获取合适的领域特定数据集可能比较困难。

**实现难度：**

* 模型搭建：实现SimCSE需要熟悉深度学习框架（如PyTorch、TensorFlow等）和对比学习的原理，了解模型的结构和训练方法。
* 数据准备：需要准备大规模的未标注数据集用于模型的无/有监督预训练阶段，以及标注数据集用于有监督微调阶段。
* 超参数调优：SimCSE有一些超参数需要调整，如学习率、负样本采样策略等，需要进行实验和调优。

**可行性分析：**

* 数据可用性：**需要获取大规模的未标注数据集**，同时**还需要标注数据集用于有监督微调阶段**。如果可用的数据集较小，可能会影响模型的性能。
* 训练时间和资源：对比学习方法通常需要较长的训练时间和大量的计算资源，需要根据可用的计算资源和时间进行评估。
* 任务复杂性：SimCSE适用于较为复杂的语义匹配任务，可以考虑使用该模型来解决临床术语标准化中的语义相似度匹配问题。

**4.BERT预训练模型fine-tuning：（本次选用sentence-bert）**

**优点：**

* 上下文理解：BERT模型通过预训练阶段学习到了丰富的语义信息和上下文理解能力，能够更好地捕捉临床术语的语义特征。
* 迁移学习：通过fine-tuning，可以在相对较小的标注数据集上进行训练，利用BERT预训练模型的泛化能力，从而提高模型性能。
* 多任务学习：BERT模型支持多任务学习，在临床术语标准化任务中，可以同时学习语义相似度匹配和其他相关任务，提高模型的效果。

**缺点：**

* 计算资源要求高：**BERT模型相对较大，包含大量参数**，需要较大的计算资源和内存来进行训练和推理。
* 预训练阶段：BERT模型的预训练需要大规模的未标注数据集和较长的训练时间，这可能对于临床术语标准化任务来说，获取合适的领域特定数据集和进行大规模预训练可能比较困难。
* 超参数调优：BERT模型有多个超参数需要调整，如学习率、批大小等，需要进行实验和调优。

**实现难度：**

* 模型搭建：实现BERT预训练模型fine-tuning需要熟悉深度学习框架（如PyTorch、TensorFlow等）和BERT模型的结构、预训练和微调方法。
* 数据准备：需要准备标注数据集用于模型的微调阶段，同时还可以考虑使用领域特定的数据集进行预训练。
* 超参数调优：BERT模型有多个超参数需要调整，需要进行实验和调优，以获得最佳的性能。

**可行性分析：**

* 计算资源和时间：BERT模型相对较大，需要较大的计算资源和较长的训练时间，需要根据可用的计算资源和时间进行评估。
* 任务复杂性：BERT模型适用于各种语义匹配任务，包括临床术语标准化中的语义相似度匹配问题。

**5. 开源的大语言模型fine-tuning：（如chatglm2b、llama2等）**

**优点：**

* 强大的语言表示能力：大语言模型经过大规模预训练，具备了丰富的语言表示能力，可以更好地捕捉临床术语的语义特征。
* 迁移学习：通过对大语言模型进行fine-tuning，可以在相对较小的标注数据集上进行训练，利用预训练模型的泛化能力，从而提高模型性能。
* 上下文理解：大语言模型能够理解上下文信息，对于临床术语标准化任务中的语义相似度匹配问题，可以考虑使用上下文信息来进行匹配。

**缺点：**

* 计算资源要求高：大语言模型通常具有巨大的模型规模和参数量，需要大量的计算资源和内存来进行训练和推理。
* 数据需求量大：大语言模型的训练通常需要大规模的标注数据集，以及合适的领域特定数据集，这可能会对临床术语标准化任务来说，获取足够的数据集可能具有一定难度。
* 超参数调优：大语言模型具有多个超参数需要调整，如学习率、批大小等，需要进行实验和调优。

**实现难度：**

* 模型搭建：实现大语言模型的fine-tuning需要熟悉深度学习框架和模型的结构，以及fine-tuning的方法和技巧。
* 数据准备：需要准备标注数据集用于模型的fine-tuning阶段，同时还可以考虑使用领域特定的数据集进行预训练和微调。
* 计算资源：大语言模型的训练和推理需要大量的计算资源，包括高性能的GPU或TPU，并且可能需要长时间运行。

**可行性分析：**

* 计算资源和时间：大语言模型需要大量的计算资源和较长的训练时间，需要根据可用的计算资源和时间进行评估。
* 任务复杂性：大语言模型适用于各种语义匹配任务，包括临床术语标准化中的语义相似度匹配问题。

总上所述，在**实现难度**方面，从小到大，**依次是机器学习，深度神经网络，人工和规则**。从**可行性角度**来讲，为了达到一个良好的准确度，其准确率从大到小，依次是，**人工和规则，深度神经网络，机器学习。**

## 拟使用算法

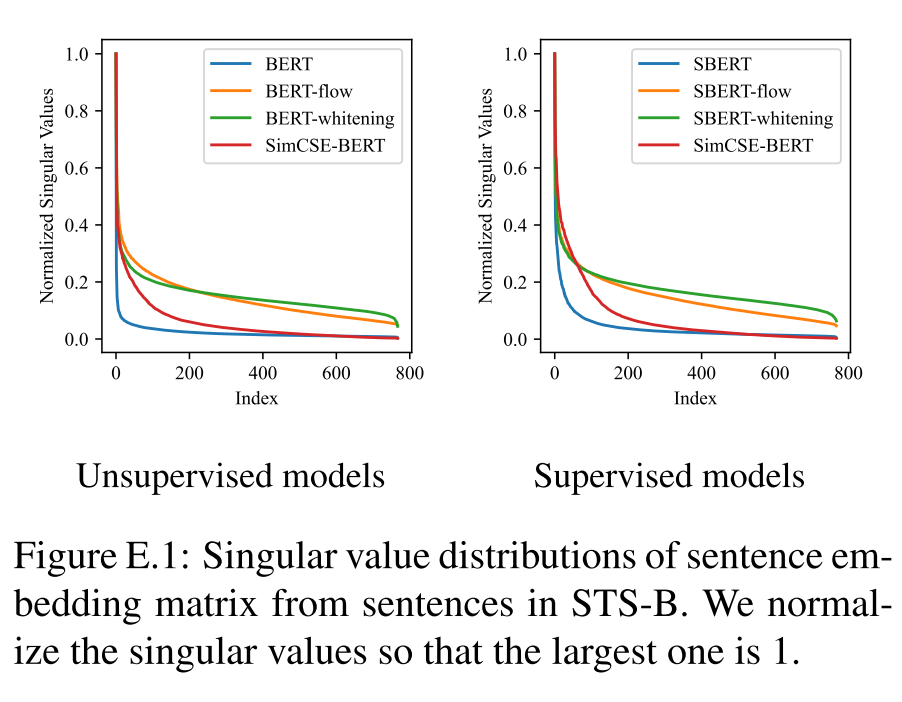
我们组使用了两种方法，一种是Sentence-BERT模型，另一种是SimCSE模型。

Sentence-BERT（SBERT）是一种基于BERT模型的句子级别嵌入算法，用于将句子映射到向量空间中。与传统的BERT模型只关注单词级别的表示不同，SBERT通过对输入句子进行特殊的处理，使得相似的句子在向量空间中更加接近。

SBERT的核心思想是使用两个特殊的嵌入操作：句子嵌入（Sentence Embedding）和余弦相似度损失（Cosine Similarity Loss）。在句子嵌入操作中，SBERT将输入句子分别通过BERT模型的最后一层输出，并取句子的CLS标记的隐藏状态作为句子的嵌入表示。然后，通过余弦相似度损失来训练模型，使得相似的句子在向量空间中的余弦相似度更高。

SBERT的实现难度相对较低，因为它是基于BERT模型的简单扩展。已经有一些开源的实现可供使用，如sentence-transformers库。这些实现提供了预训练的SBERT模型和用于训练和微调的工具。

SBERT本身并不复杂，仅仅是一个基于BERT的孪生网络而已，想要在SBERT的基础上改进指标，只能从训练目标下手，而我们经过调研，发现有论文中，将对比学习的思想引入了SBERT，大幅刷新了有监督和无监督语义匹配SOTA，更让人惊叹的是，无监督SimCSE的表现在STS基准任务上甚至超越了包括SBERT在内的所有有监督模型。同时论文中的作者借助SentEval在多个数据集上评估了SimCSE的性能，可以发现，在STS系列任务上，SimCSE在无监督和有监督的条件下均大幅超越了之前的SOTA模型，而在迁移任务上，SimCSE也有着可观的性能提升。因此我们准备使用有监督的SimCSE也进行了实验。



图源自论文：**SimCSE: Simple Contrastive Learning of Sentence Embeddings**

在该项目背景下，使用SBERT模型和有监督的SimCSE模型需要对现有的BERT模型进行微调和适应，以满足特定的任务需求。

## 主要步骤介绍

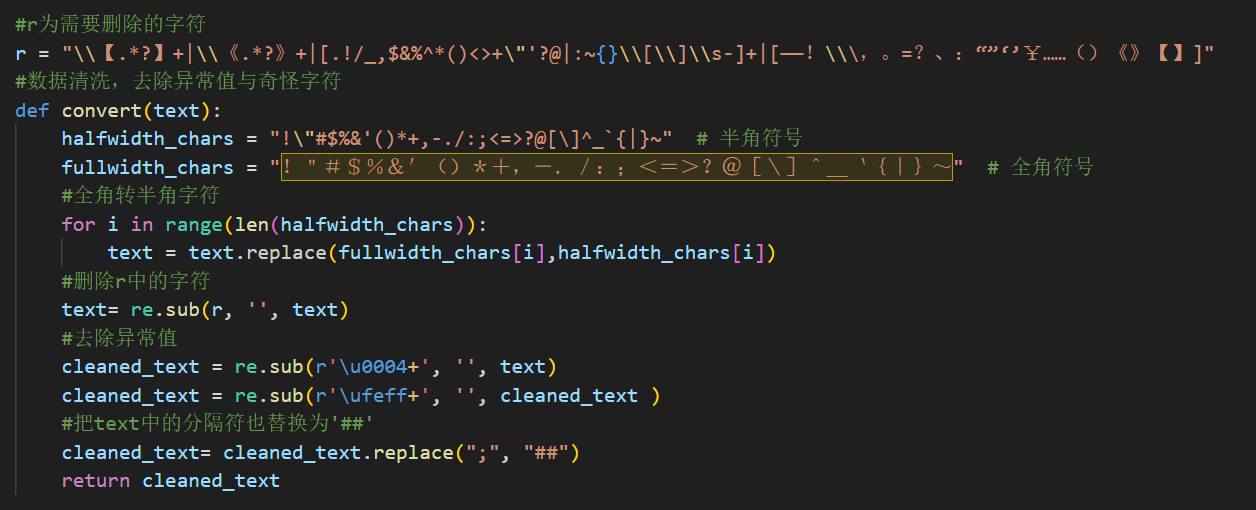
主要内容：

1. 列举完成项目所需的主要步骤。

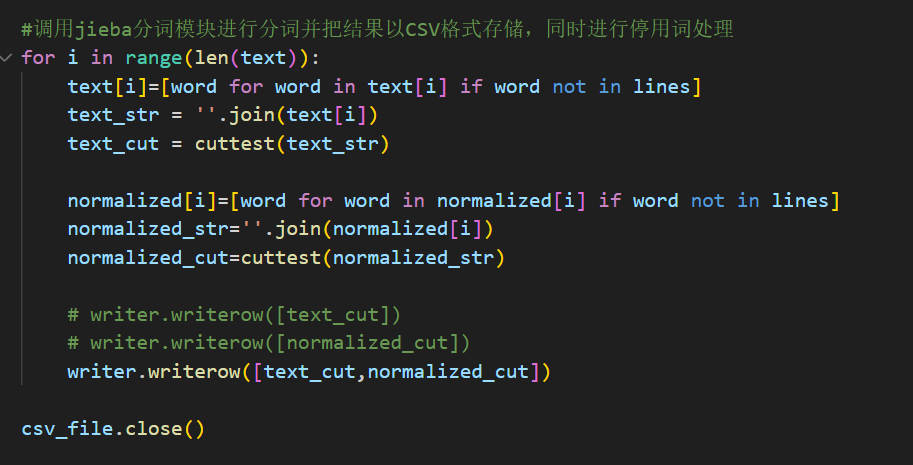
* 数据预处理，进行数据清洗和分词等
* 商讨负样例的方案，为清洗后的数据打上label
* 寻找可用的模型（sentence-bert,SimCSE）
* 针对本数据集，进行模型的微调和训练
* 得到对应的评价指标得分，分析原因，进行进一步调整和优化
* 得到训练效果最好的模型，进行test测试集的预测，并把结果保存为csv文件
* 进行拓展，实现STT和TTS功能模块，方便进一步使用

1. 对每一个步骤进行详细说明。

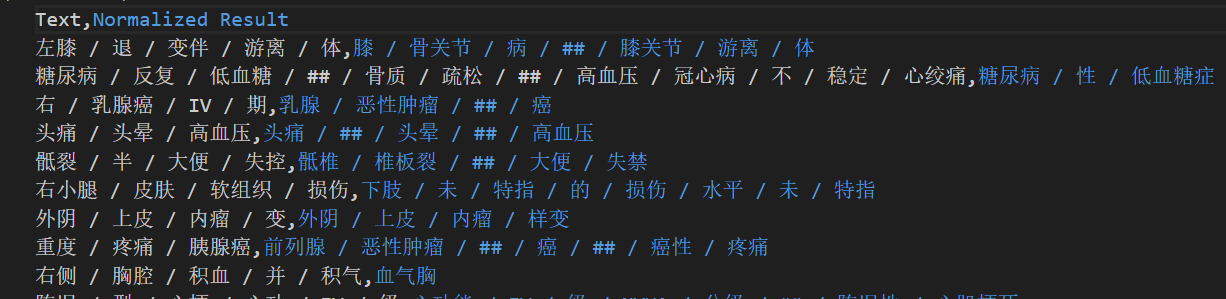
* 数据清洗部分，我们结合了ark-nlp在CHIP2021临床属于标准化方法中的数据集处理文件，将部分字符进行了替换，同时将字母大小写进行了统一，然后根据数据集部分，我们进行了中文的全角字符转英文的半角字符，使用正则表达式删除了数据集中出现的奇怪字符，根据字符编码去除了其中的异常值，同时把输入文本中的分隔符从分号替换为井号，和标准值连接的井号保持一致，做了统一处理。同时为了方便后续任务进行，我们也进行了jieba分词的数据预处理。本来我们想要进行停用词处理，但是发现这样处理会丢失很多语义信息，于是也没有采用这个方案。



图示为我们自己根据数据集特点进行处理



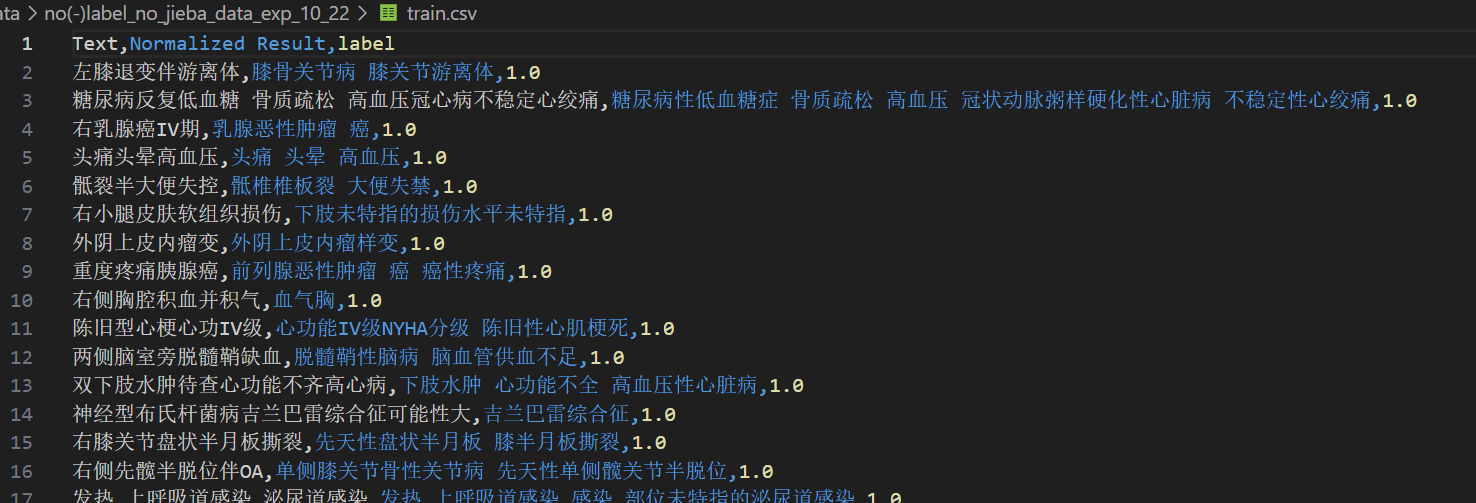
图示为jieba分词模块



图示为jieba分词后的结果

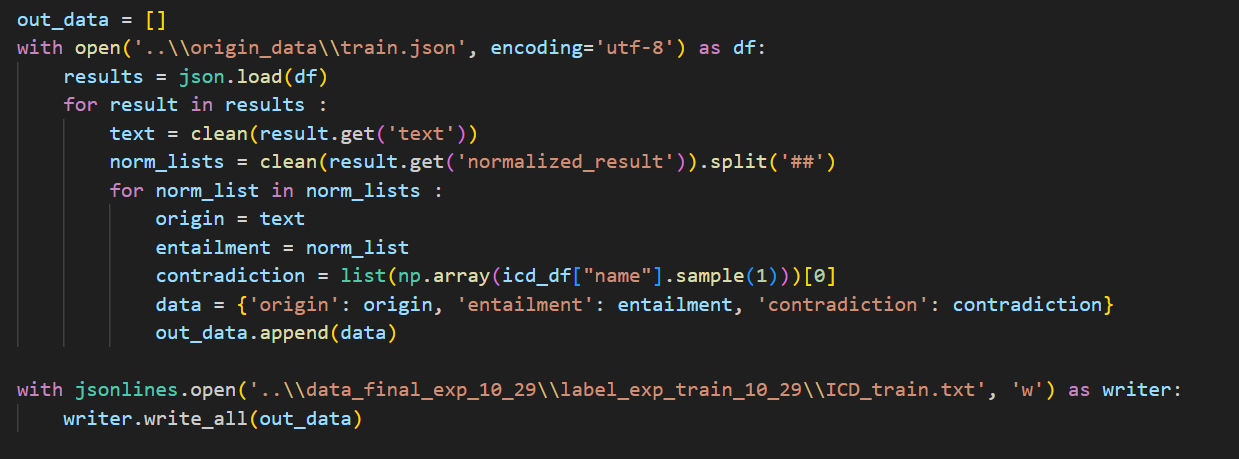


图示为去除停用词后进行jieba分词的结果



图示为没有经过jieba分词进行数据清洗后的结果

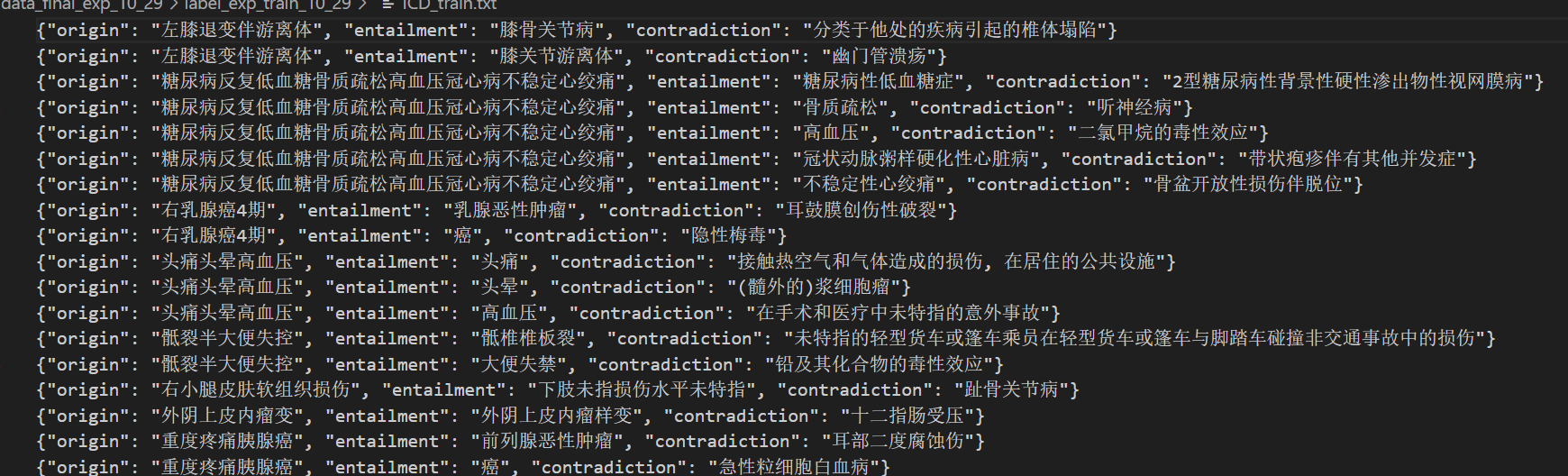
* 由于原数据集中没有负样例，我们经过讨论（不一定对），我们想要通过ICD-10标准数据集中随机挑选的文本作为负样例来进行label的设定。同时把标准值中的##进行切分，我们假定只要预测值在多个标准值中，就判定为预测正确，所以为每个标准值都打了正标签。



图示为我们进行label的标定，包括生成负标签，以及进行文本的切分



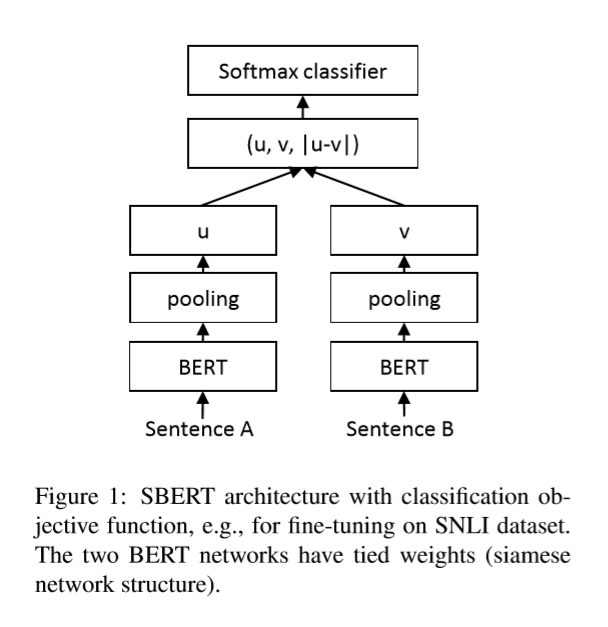
图示为我们标注后的结果



图示为我们将原文本、正样例、负样例组合为三元组的形式

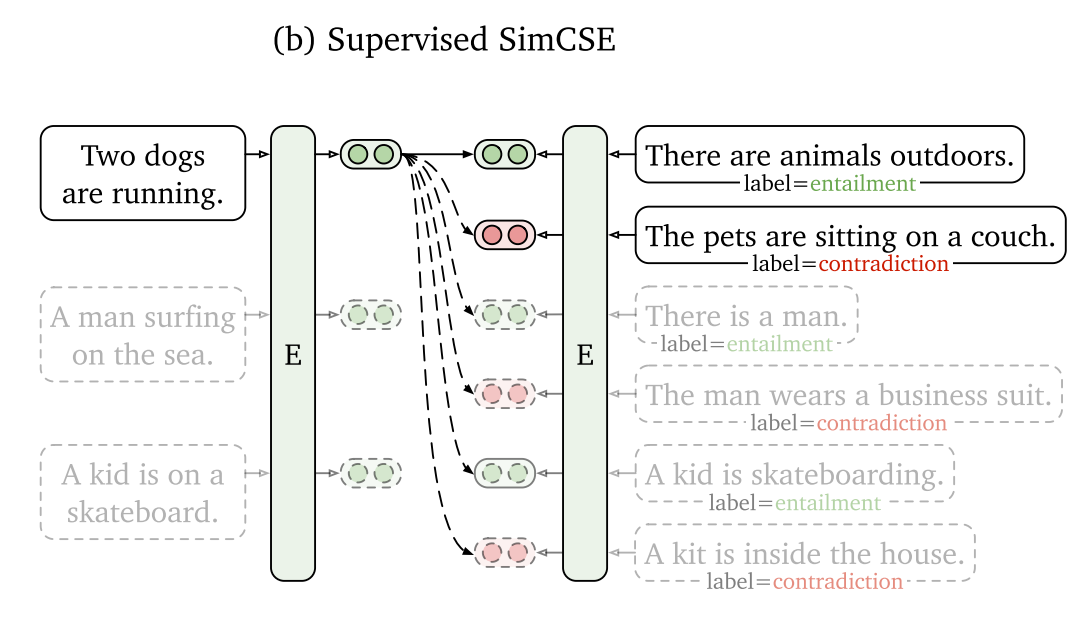
* 我们想要使用预训练模型进行微调，一开始的方案是有以下两种：
  + 使用最近开源的大语言模型如：llama2,chatglm2等，特别是llama2的中文模型进行大模型的微调。
  + 使用开源的bert模型，如bert-base-chinese，针对bert模型进行fine-tuning

但是经过调研，我们发现，大模型虽然最后可能效果会比较好，但是它参数量太大，对显卡要求比较高，我们电脑可能跑不动，而且耗费的时间会非常长。因此我们选择了较为好调的中文bert模型，也是一个baseline的模型。（虽然我们后面没来的及选择其他的bert模型进行对比分析）再经过调研，我们发现有一个sentence-bert模型可以很好的处理我们句子间语义相似度匹配的问题，我们决定通过fine-tuning这样一个模型来适应我们的数据集。



图示为sentence-bert模型的原理图

同时经过调研，我们发现有监督的SimCSE模型效果可能会更好，所以也进行了尝试。



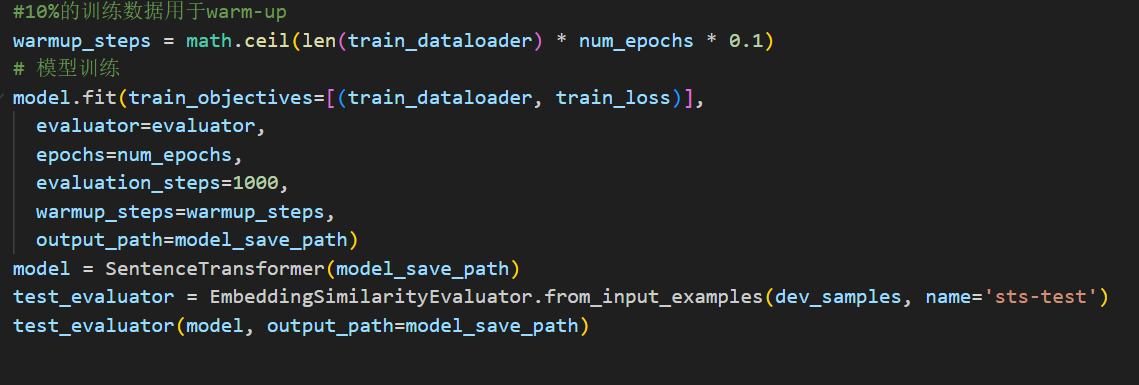
图示为有监督的SimCSE模型示意图

* 模型的fine-tuning

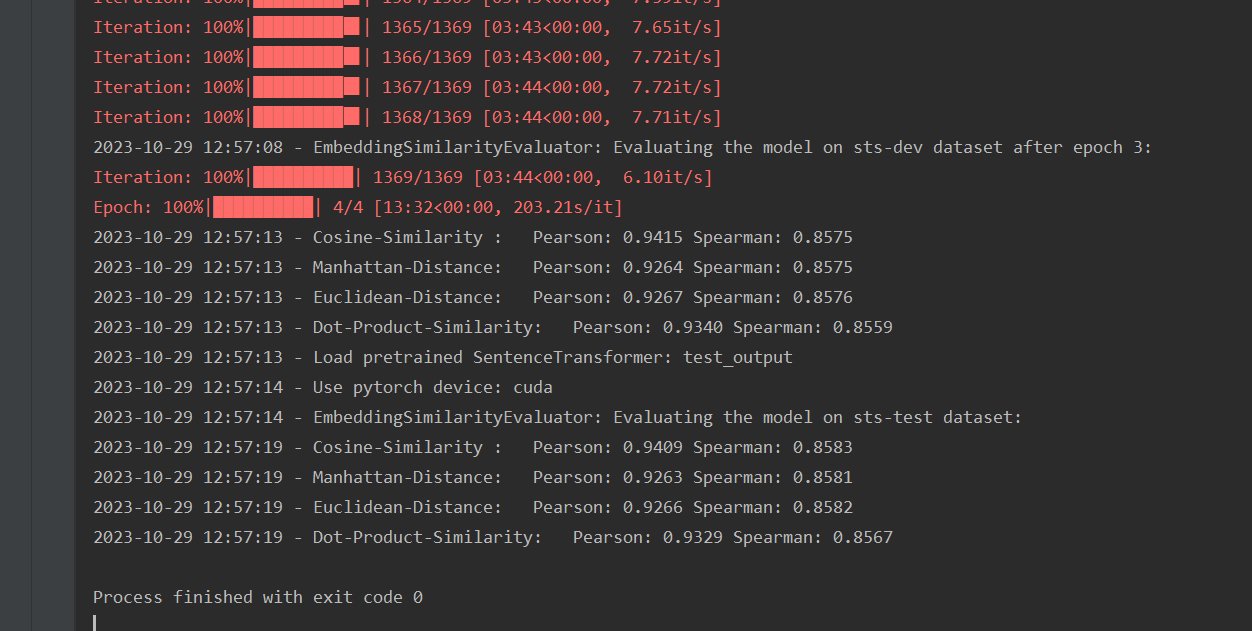
根据sentence-bert模型原理图，我们搭建了如下框架，并进行了模型的训练。



图示为我们fine-tuning的超参数设定、词嵌入、平均池化等过程截图

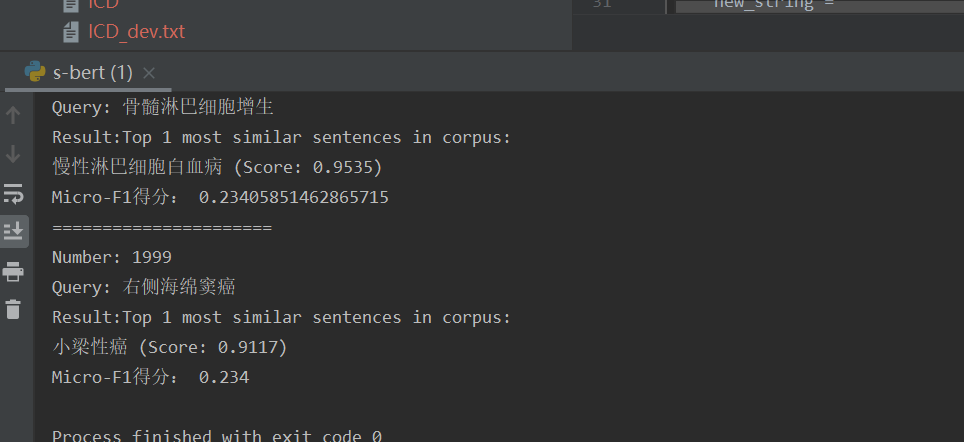


图示为训练过程，其中百分之10的训练数据进行warm-up来调整学习率

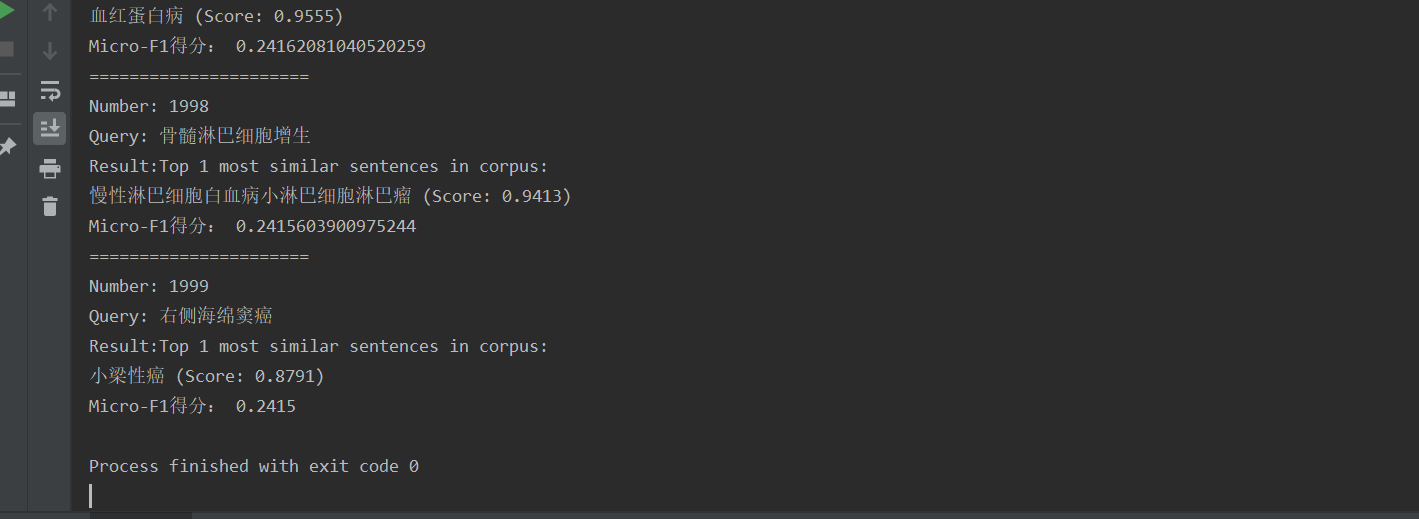


图示为训练四轮后的运行截图

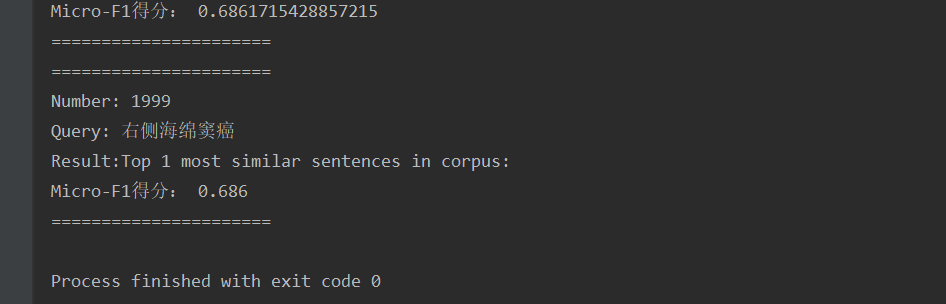
* 我们加载我们得到微调后的模型，进行dev，得到对应的评价指标得分如下



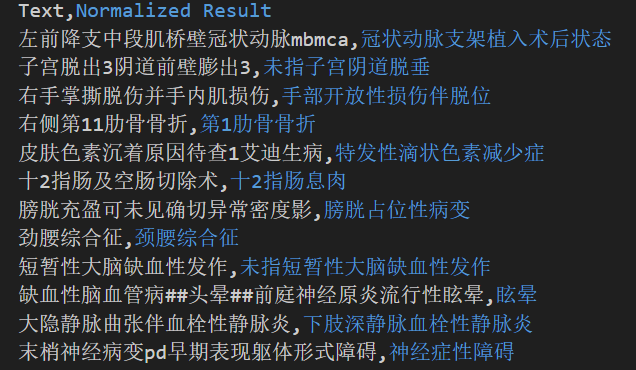
发现F1得分仅有0.234，我们认为可能是超参数的问题，然后我们进行调参，batch\_size设为48，训练了15epochs，但是得分依旧很低，为0.2415



我们经过讨论，应该是F1得分代码逻辑的问题，我们设定输出只能是一个，有多个预测值的结果被判定为预测失败，所以我们重新修改输出设置，现在F1得分为0.686，提升明显。



* 得到模型后进行test，并把结果以csv格式进行存储



* 进行STT、TTS模块的搭建，这里调用了百度的api提供语音识别和语音合成服务

1. 每一步拟使用时间。

* 数据预处理，数据清洗：一周
* 标签处理：3天
* 寻找可用模型fine-tuning : 4天
* Fine-tuning sentence-bert :一周
* 有监督的SimCSE实现：一周
* 模型测试，评估和调优：一周
* STT&TTS模块搭建：3天
* 整体联调：4天
* 继续调整模型结构，寻找优化方法：持续

1. 可能存在难点及潜在解决办法。

* 环境搭建：借鉴github的部分开源代码中环境配置存在问题，有的python版本不对，有的框架版本和GPU版本对不上等等。

解决办法：仔细阅读readme文档，查询了很多博客，理清了对应的版本，同时尽量用 conda去安装对应的包，pip安装容易导致不同包版本不兼容，注意pip安装时要指定 镜像源，否则容易因连接不稳定而下载不了，或者可以科学上网。

* 数据预处理：数据清洗的方式很多，要针对特定数据集进行清洗方式的调整

解决办法：首先参考一般nlp数据清洗部分的常用方法，如全角转半角字符、异常值删 去、英文大小写统一、特殊字符通过正则表达式进行去除等，然后我借鉴了ark-nlp在 CHIP2021临床属于标准化方法中的数据集处理文件，将该数据集中的特定字符集进行 了转换。

* Fine-tuning:模型搭建对于pytorch框架不熟悉

解决办法：先去补了一下基础（初入门，基础不太行），完成了手写数字识别项目，熟悉了整个网络流程的基本框架，包括dataloader、tensorboard使用、网络搭建、优化器、损失函数等，然后借鉴github中的开源sentence-bert、SimCSE代码，进行了部分修改，迁移到我们这个任务中。

* 评价指标：对Micro-F1得分不理解

解决办法：参考了学长发的链接，同时在b站找了对应的教程，实现了简单的F1得分 的代码，自己编写了F1函数，测试后发现和机器学习库中的函数结果一致，然后迁移 到自己的任务中，实现了F1得分评价指标。

1. 若合作项目，列出分工明细。

胡景斌：**进行数据预处理、数据清洗、sentence-bert模型的fine-tuning、实现了SimCSE有监督模型的本地部署、实现TTS&STT模块的搭建、根据评价指标进行调参和进一步模型的优化。**

韦浩洋：**进行了数据的标签处理，帮助进行整体联调和环境搭建。**

## 代码实现与解释性标注

主要内容：

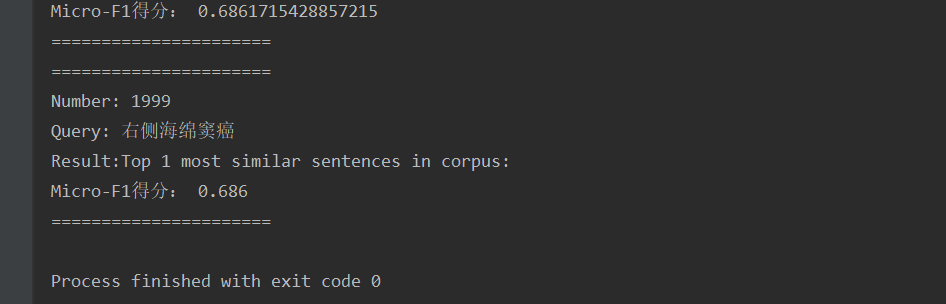
1. 上传代码zip文件，源代码关键步骤需进行标注工作。
2. 上传Readme文档，包括：配置环境，安装步骤，demo命令。
3. （可选）上传结果演示视频

## 项目总结评估及未来发展展望

主要内容：

1. 对项目代码进行评估，分析代码性能。

Sentence-bert：

模型的F1得分一开始很低，经过数据清洗和修改输出逻辑，最终提高到0.686，效果良好。

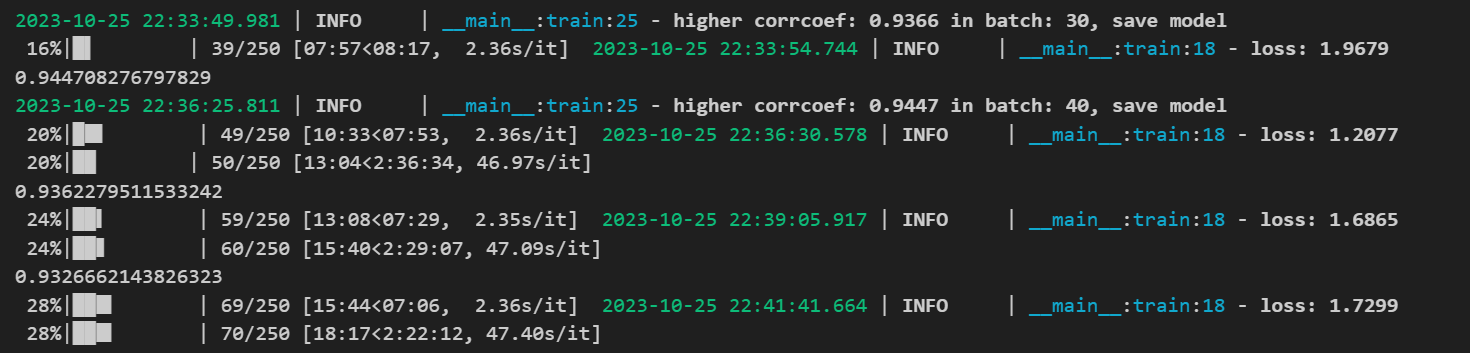


图示为微调过程中的评价指标情况，可以看到余弦相似度的皮尔逊相关系数和斯皮尔曼相关系数等均在不断升高，说明微调效果良好。

这次模型采用bert-base-chinese进行fine-tuning的方法，属于迁移学习的一种尝试。最终F1得分在0.686。但是仍然存在一些问题，数据集比较小，训练后的效果不太好，需要进一步扩充数据集再进行训练，可能效果会更好。

SimCSE：

本次模型我们采用BM25进行初步筛选，SimCSE进一步筛选的方法，结合传统方法和深度学习，最终在单个相似度检测中准确率达到了0.92以上。



但是同样存在着一些问题，经过SimCSE进行编码过后的向量，采用余弦相似度直接进行比较，缺少语义信息，尽管速度会快很多，但是其深层次的语义特征并没有被很好的挖掘出来，从而导致准确率降低。

1. 总结项目开发过程中的得失。

得：

* 充分调研了当下的语义相似度匹配模型，对该领域有了初步的了解。
* 学习了基础的pytorch框架，自己复现了手写数字识别等开源项目，看懂了开源代码并进行了一定的修改。
* 熟悉了shell脚本的编写方式。
* 熟悉了yaml文件的配置格式。
* 学会用python处理基础的文件、进行数据清洗，熟悉了numpy、pandas、csv、json等库的使用，学会用大语言模型辅助代码的编写与学习。
* 自己复现了sentence-bert、SimCSE开源代码并将其迁移到我们这个任务中。

失：

* 代码中缺少可视化模块，没来得及使用tensorboard记录训练过程loss的变化。
* 没有对比分析，bert模型只是选用了最基础的bert-base-chinese，没有来的及用其他的bert如：biobert/umlsbert等面向医疗领域的变种bert模型，缺乏比较分析。
* SimCSE没来的及计算F1得分，由于时间紧迫加上自己能力不足，只是分析了Sentence-bert模型的F1得分，SimCSE没来得及做，后面有时间会做完，对比分析。
* 相似度评价指标选择余弦相似度，没有考虑到多义信息，效果欠佳，后续考虑选择其他的评价指标。
* 对pytorch框架、机器学习库不熟悉，很多代码还是用的python的语法去写循环，F1其实可以直接从sklearn库调用，但是不太熟悉，还是自己用for循环搭建了一个F1函数。

1. 对项目所在领域的未来发展进行自己的分析：

未来对于临床术语标准化的发展，我觉得基于大模型的发展，最终会走向prompt learning的道路，nlp我也觉得最终会走向prompt learning。当下有很多预训练模型，针对于下游任务我们需要在预训练模型的基础上进行finetuning。随着llama2和chatglm2等大模型的开源，现在很多nlp任务丢给大模型微调效果会好很多，班里也有同学微调llama2，F1得分达到了70%以上，因此我们有理由相信，未来一定会是走向prompt learning的方向。

而基于现有的nlp方向，知识图谱和本体论是表示和组织知识的重要方法，将在临床术语标准化中发挥更大的作用。通过将临床术语与知识图谱和本体论结合起来，可以提供更丰富的语义信息和推理能力，帮助更好地理解和解释医学文本中的术语。

参考开源仓库：

<https://github.com/princeton-nlp/SimCSE.git>

<https://github.com/vdogmcgee/SimCSE-Chinese-Pytorch>

<https://github.com/DataArk/CHIP2021-Task3-Top3.git>

<https://github.com/Lisennlp/two_sentences_classifier.git>

参考文献：

1. 查汉. 基于文本匹配的临床医学术语标准化方法研究[D].中南财经政法大学,2022.DOI:10.27660/d.cnki.gzczu.2022.002782.
2. 艾丽娜. 多源异构健康医疗大数据治理平台设计与实现[D].山东大学,2020.DOI:10.27272/d.cnki.gshdu.2020.001766.
3. Lee Y,Krishnamoorthy S,Dinakarpandian D. A semantic framework for intelligent matchmaking for clinical trial eligibility criteria[J]. ACM Transactions on Intelligent Systems and Technology (TIST),2013,4(4).
4. 袁雯. 电子病历中临床术语的标准化研究[D].厦门大学,2022.DOI:10.27424/d.cnki.gxmdu.2020.002253.
5. 孙曰君,刘智强,杨志豪等.基于BERT的临床术语标准化[J].中文信息学报,2021,35(04):75-82.
6. 郭玉峰,尹爱宁,周霞继等.浅谈中医临床术语标准化工作现状及其深化推进[J].中国中医药信息杂志,2009,16(11):3-4.
7. 何清行. 语义相似度算法研究与医学术语服务平台的实现[D].湖南大学,2018.