（命名）实体识别 NER 已有古文预训练模型测评实验报告

数据集

数据集 A 取自《新五代史》《北史》等古籍，训练集，验证集和测试集分别为 21128，2831，2148，平均句长为 26，采用 BIO，共有 5 种实体类型，[O，B-NOUN_BOOKNAME，I-NOUN_BOOKNAME，B-NOUN-OTHER ，I-NOUN-OTHER]

数据集 B 取自《春秋谷梁传》等古籍，训练集，验证集和测试集分别为 7528，865，877，平均字长为 85，采用 BIO，共有 13 种实体类型，[O ,B-PER ,I-PER, B-ORG ,I-ORG ,B-LOC ,I-LOC, B-JOB ,I-JOB ,B-WAR, I-WAR, B-BOO, I-BOO]

实验数据下载：
https://github.com/jizijing/C-CLUE
https://github.com/Ethan-yt/CCLUE

评价指标

准确率(Accuracy)

Accuracy 是从整体上衡量模型的性能，即模型预测正确的样本占全部样本的比例。它的计算公式如下：

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}$$

其中，TP（True Positive）表示真正例，即模型预测为正例且实际上也为正例的样本数量；FP（False Positive）表示假正例，即模型预测为正例但实际上为负例的样本数量；TN（True Negative）表示真负例，即模型预测为负例且实际上也为负例的样本数量；FN（False Negative）表示假负例，即模型预测为负例但实际上为正例的样本数量。

精确率(Precision)

Precision 关注的是模型预测为正例的样本中有多少是真正的正例。换句话说，它衡量的是模型预测的正例中有多少是正确的，计算公式为：

$$Precision = \frac{TP}{TP + FP}$$

其中，TP（True Positive）表示真正例，即模型预测为正例且实际上也为正例的样本数量；FP（False Positive）表示假正例，即模型预测为正例但实际上为负例的样本数量。

F1 值(F1 Score)

F1 分数是精确率和召回率的调和平均值，用于衡量模型的准确性。它的计算公式如下：

$$F1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

其中，精确率（Precision）是指模型预测为正例的样本中真正为正例的比例，召回率（Recall）是指所有真正的正例样本中被模型预测为正例的比例。

召回率(Recall)

召回率是指在所有真正的正例样本中，被模型正确预测为正例的比例。它的计算公式如下：

$$Recall = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

其中，TP（True Positive）表示真正例，即模型预测为正例且实际上也为正例的样本数量；FN（False Negative）表示假负例，即模型预测为负例但实际上为正例的样本数量。

Micro-Average

Micro-Average 把所有类别的结果汇总起来计算平均值。它将所有类别的贡献视为等同，因此对于样本量大的类别，Micro-Average 更加敏感。

$$\text{Micro-Average Precision} = \frac{\sum TP}{\sum TP + \sum FP}$$

$$\text{Micro-Average Recall} = \frac{\sum TP}{\sum TP + \sum FN}$$

$$\text{Micro-Average F1} = 2 \times \frac{(\text{Micro Precision} \times \text{Micro Recall})}{(\text{Micro Precision} + \text{Micro Recall})}$$

Macro-Average

Macro-Average 分别计算每个类别的性能指标，然后计算这些指标的算术平均值。它给予每个类别同等的权重，无论类别的样本量大小，因此对于样本量小的类别更加敏感。

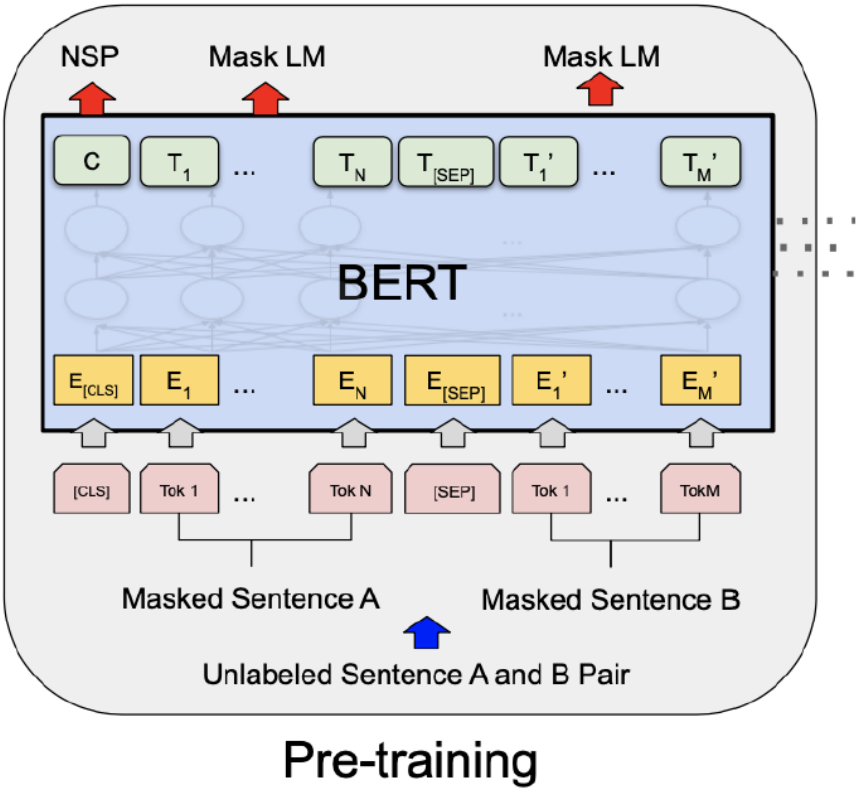
$$\text{Macro-Average Precision} = \frac{1}{N} \sum \frac{TP_i}{TP_i + FP_i}$$

$$\text{Macro-Average Recall} = \frac{1}{N} \sum \frac{TP_i}{TP_i + FN_i}$$

$$\text{Macro-Average F1} = \frac{1}{N} \sum 2 \times \frac{(Precision_i \times Recall_i)}{(Precision_i + Recall_i)}$$

模型

BERT



Pre-training

BertModel 部分包含了 BERT 的所有主要组件，包括词嵌入层、位置嵌入层、令牌类型嵌入层、层归一化层和 Dropout 层。
BertEncoder 部分负责处理输入序列，通过多个 BERT 层来进行特征提取。每个 BERT 层都包含注意力机制和前馈神经网络。
BertForTokenClassification 模型的最后一部分是一个线性分类器，它将 BERT 编码器的输出映射到目标类别的概率分布上。

1.Transformer 架构： BERT 建立在 Transformer 模型的基础上，这是一种使用自注意力机制（Self-Attention Mechanism）的深度神经网络。Transformer 允许模型在处理序列数据时同时关注序列中的所有位置，而不是像传统的循环神经网络（RNN）或卷积神经网络（CNN）那样逐步处理。
2.预训练策略： BERT 采用了无监督的预训练策略，通过大规模的语言模型预训练来学习丰富的语义表示。该模型通过对大量文本数据进行“遮蔽语言模型”（Masked Language Model，MLM）任务的预训练，使得模型能够理解词汇和语法结构，并捕捉单词之间的关系。
3.双向性： BERT 在预训练时考虑了双向信息，即使用上下文信息来理解每个词的语义。这种双向性有助于模型更好地理解文本中的语境和关联，提高了对上下文相关性的捕捉。
4.Fine-tuning： 预训练后，BERT 模型可以通过微调（fine-tuning）来适应特定的下游任务，如命名实体识别、情感分析等。这种能力使得 BERT 在各种 NLP 任务中都表现出色，无需从零开始训练新的模型。
5.Contextual Embeddings： BERT 生成的词向量是上下文相关的，每个词的表示取决于整个输入句子的上下文，而不是简单地从固定的嵌入中获取。这种上下文敏感的嵌入有助于更准确地捕捉语义信息。

实验结果

Bert-ancient-Chinese

训练最终结果

数据集	Eval Precision	Eval Recall	Eval F1	Eval Accuracy	Eval Loss	Eval Runtime
数据集 A	0.973022168545 7736	0.9734486209964412	0.973205034052774 1	0.9734486209964412	0.15202167630195618	4.6471
数据集 B	0.989200309989 5879	0.9911334325396826	0.990085767321637 7	0.9911334325396826	0.02559696696698665 6	10.0205

数据集	Micro-Average Precision	Micro-Average Recall	Micro-Average F1	Macro-Average Precision	Macro-Average Recall	Macro-Average F1
数据集 A	0.9734486209964412	0.9734486209964412	0.9734486209964412	0.8710822709848423	0.8494438276449081	0.8599516197788132

数据集B	0.9911334325396826	0.9911334325396826	0.9911334325396826	0.24364633330184382	0.21699121039347588	0.21620752290443754

模型介绍和分析

这是一个基于 BERT 的模型，专门训练用于古汉语处理。它适用于古文的语义理解和文本生成，旨在提高对古汉语的处理能力。

训练过程分析

数据集 A

Epoch	Eval Precision	Eval Recall	Eval F1	Eval Accuracy	Eval Loss	Eval Runtime
2	0.9651031456399267	0.9666370106761566	0.9651928655745571	0.9666370106761566	0.1015414595603429	4.6365
4	0.9678706559288105	0.9669984430604982	0.9673165804611824	0.9669984430604982	0.1039665490388702	4.5978
6	0.9714789822369363	0.9719472864768683	0.9715817412063057	0.9719472864768683	0.0998795256018638	4.6208
8	0.9718535415569596	0.9717804715302492	0.9717633388042668	0.9717804715302492	0.1338398754596710	4.6113
10	0.9732551487821443	0.973810053380783	0.9734712820803293	0.973810053380783	0.1370591372251506	4.6348

数据集 B

Epoch	Eval Precision	Eval Recall	Eval F1	Eval Accuracy	Eval Loss	Eval Runtime
2	0.9844360351562 5	0.9921875	0.9882965686274511	0.9921875	0.0293049532920122 15	7.6138
4	0.9874243224761378	0.9916648951247166	0.9894932891321785	0.9916648951247166	0.0257867444306612	7.5513
6	0.9880146702085819	0.9918774801587301	0.9891842041092 48	0.9918774801587301	0.0242147631943225 86	7.4568
8	0.9886410337332172	0.9913283021541 95	0.9898839745068818	0.9913283021541 95	0.0242569167166948 32	7.5093
10	0.9895343898980461	0.9911777210884354	0.9902227660139274	0.9911777210884354	0.0251425318419933 32	7.4757

SikuRoberta

训练最终结果

数据集	Eval	Eval Recall	Eval F1	Eval	Eval Loss	Eval

	Precision			Accuracy		Runtime
数据集 A	0.971049537254968	0.9716136565836299	0.9712887620106267	0.9716136565836299	0.1508568525314331	3.5724
数据集 B	0.989639677599 6501	0.9911068594104309	0.9903303967883023	0.9911068594104309	0.02541288547217846	7.8184

数据集	Micro-Average Precision	Micro-Average Recall	Micro-Average F1	Macro-Average Precision	Macro-Average Recall	Macro-Average F1
数据集 A	0.971613656583623699	0.971613656583623699	0.971613656583623699	0.864802897459217676	0.842991834751582323	0.853425216439090303
数据集 B	0.991106859410430909	0.991106859410430909	0.991106859410430909	0.231851156607637 94	0.230711271903512 04	0.221999412876054 57

模型介绍和分析

这个模型是 Siku Quanshu（四库全书）的基础上训练的 RoBERTa 变体。它对古典文献中的文本理解有很强的能力，适合处理古籍和文献分析任务。

训练过程分析

数据集 A

Epoch	Eval Precision	Eval Recall	Eval F1	Eval Accuracy	Eval Loss	Eval Runtime
2	0.960833118499789	0.9616047597864769	0.9601480741181836	0.9616047597864769	0.11830540746450424	3.5616
4	0.967570569968203	0.9680549377224199	0.9674608679832772	0.9680549377224199	0.1031668409705162	3.559
6	0.969178426 8414622	0.9699455071174378	0.9694620149663 79	0.9699455071174378	0.1095397397 8757858	3.5763
8	0.971794223 3651111	0.9722809163701067	0.9720013951812252	0.9722809163701067	0.1303454190 49263	3.5741
10	0.971084619 4442453	0.9715024466 19217	0.9712697469725421	0.9715024466 19217	0.1374275982 3799133	3.5555

数据集 B

Epoch	Eval Precision	Eval Recall	Eval F1	Eval Accuracy	Eval Loss	Eval Runtime
2	0.984436035 15625	0.9921875	0.9882965686274511	0.9921875	0.0290909465 40236473	5.4347
4	0.98735217	0.99180661	0.98954351	0.99180661	0.025082616	5.27

	26932424	84807256	36377729	84807256	13547802	25
6	0.98752439 2029138	0.99191291 09977324	0.98926084 41941434	0.99191291 09977324	0.023758890 107274055	5.23 86
8	0.98878774 15659296	0.99158517 57369615	0.99002233 63677141	0.99158517 57369615	0.024264896 288514137	5.46 21
10	0.98936904 56522269	0.99123086 73469388	0.99020889 12819328	0.99123086 73469388	0.024774728 34289074	5.23 45

AnchiBERT

训练最终结果

数据集	Eval Precision	Eval Recall	Eval F1	Eval Accuracy	Eval Loss	Eval Runtime
数据集 A	0.96772 6471010 9994	0.9684719 75088968	0.9680270 67726180 2	0.96847197 5088968	0.16520337 760448456	3.0608
数据集 B	0.98981 3124692 8284	0.9911068 594104309	0.9903980 65962140 8	0.99110685 94104309	0.02563157 305121421 8	7.6369

数据集	Micro-Average Precision	Micro-Average Recall	Micro-Average F1	Macro-Average Precision	Macro-Average Recall	Macro-Average F1
数据集 A	0.9684719 75088968	0.9684719 75088968	0.9684719 75088968	0.8426885 820191009	0.82060433 51382434	0.83101526 89098269
数据集 B	0.9911068 594104309	0.9911068 594104309	0.9911068 594104309	0.2512589 632357055	0.24764968 296968584	0.23697334 205426301

模型介绍和分析

AnchiBERT 是针对古代汉语的 BERT 变体，提供了对古汉语语料的更深入的理解。它优化了 BERT 架构以适应古汉语的特殊需求，适合古文翻译和解析。

训练过程分析

数据集 A

Epoch	Eval Precision	Eval Recall	Eval F1	Eval Accuracy	Eval Loss	Eval Runtime
2	0.96003681 25761395	0.96202179 71530249	0.96050850 30338348	0.96202179 71530249	0.116751648 48566055	3.04 96
4	0.96362474 02560569	0.96421819 39501779	0.96360568 10176072	0.96421819 39501779	0.116235248 74448776	3.07 19

6	0.9670201712841807	0.9678325177935944	0.9673408421844694	0.9678325177935944	0.1162397712469101	3.0606
8	0.9668893423942839	0.9684441725978647	0.9672642557197699	0.9684441725978647	0.1434069722890854	3.0503
10	0.9682593609860043	0.9689724199288257	0.9685297974658393	0.9689724199288257	0.1514194011688324	3.0645

数据集 B

Epoch	Eval Precision	Eval Recall	Eval F1	Eval Accuracy	Eval Loss	Eval Runtime
2	0.9861255086400601	0.9920989229024944	0.9889600864359019	0.9920989229024944	0.0277487654238393	5.0794
4	0.9864952449481542	0.9919483418367347	0.9890672028096879	0.9919483418367347	0.0247181635349988 94	4.8241
6	0.9884934437504811	0.9917003259637188	0.9899320485624267	0.9917003259637188	0.0244986005127429 96	4.9891
8	0.9892648799954938	0.9912840136054422	0.9902119089568483	0.9912840136054422	0.0243615210056304 93	5.386
10	0.9892282763694249	0.9911954365079365	0.9901136042309379	0.9911954365079365	0.0251300539821386 34	4.8651

guwenbert-base

训练最终结果

数据集	Eval Precision	Eval Recall	Eval F1	Eval Accuracy	Eval Loss	Eval Runtime
数据集A	0.9758289107635478	0.9762010676156584	0.9759814042908822	0.9762010676156584	0.11708889156579971	2.7589
数据集B	0.9959199707856833	0.9959254535147393	0.9958638948914785	0.9959254535147393	0.013080289587378502	6.8721

数据集	Micro-Average Precision	Micro-Average Recall	Micro-Average F1	Macro-Average Precision	Macro-Average Recall	Macro-Average F1
数据集A	0.9762010676156584	0.9762010676156584	0.9762010676156584	0.8841642431251063	0.8680398426337851	0.8758174096034843

数据集B	0.9959254535147393	0.9959254535147393	0.9959254535147393	0.40479139253470026	0.4110982524602927	0.3914383970521158

模型介绍和分析

guwenbert-base 是一个专门为古文设计的 BERT 模型。它利用大量古文语料进行训练，目标是提高古文文本的理解和处理能力。

训练过程分析

数据集 A

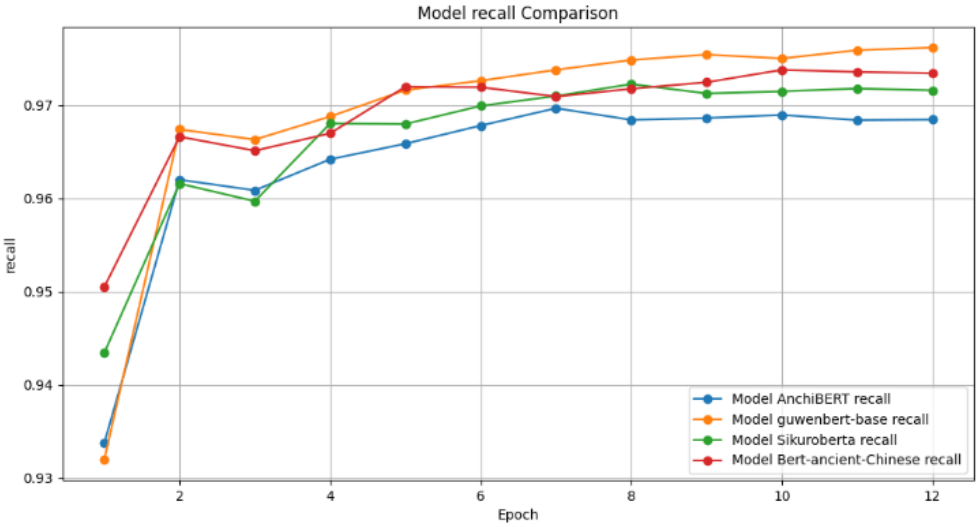
Epoch	Eval Precision	Eval Recall	Eval F1	Eval Accuracy	Eval Loss	Eval Runtime
2	0.9661654325093485	0.9674154804270463	0.9666295824622826	0.9674154804270463	0.1100587323307991	2.6951
4	0.967617962053998	0.9688056049822064	0.9676484892367837	0.9688056049822064	0.1030073761940024	2.6698
6	0.9727868540697165	0.9726423487544484	0.9726361929489298	0.9726423487544484	0.0946088582277298	2.6626
8	0.9748909685689006	0.9748665480427047	0.9747762081255394	0.9748665480427047	0.1125110611319542	2.6597
10	0.9749028335985271	0.9750333629893239	0.9749133871389762	0.9750333629893239	0.1154715940356254 6	2.7261

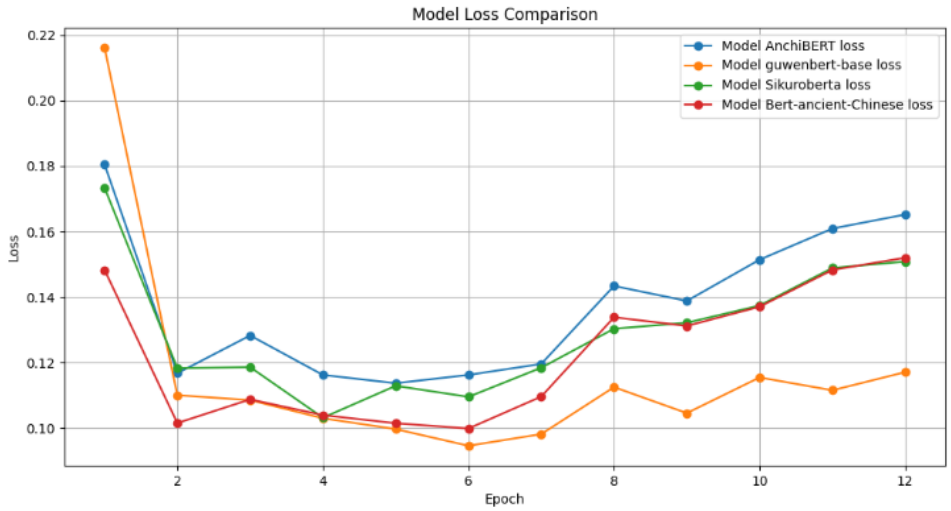
数据集 B

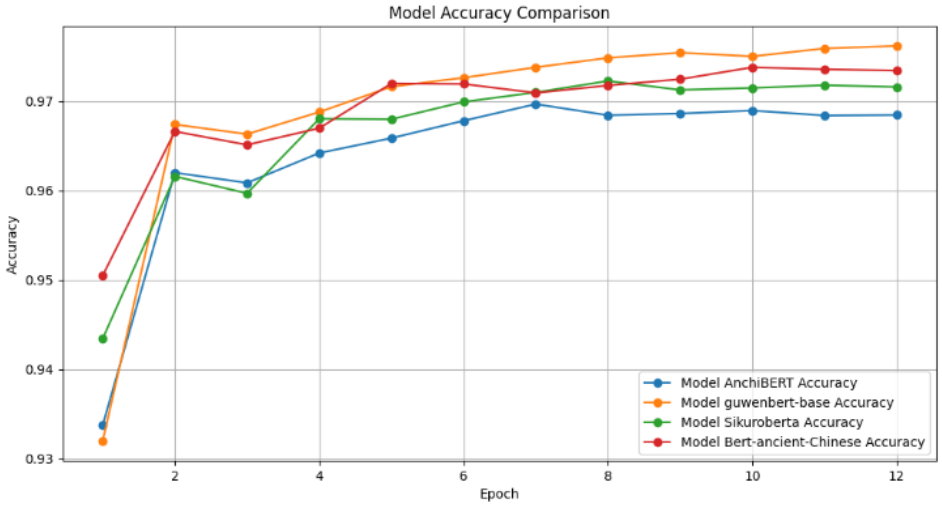
Epoch	Eval Precision	Eval Recall	Eval F1	Eval Accuracy	Eval Loss	Eval Runtime
2	0.9931340195989583	0.9949068168934241	0.9935917561585891	0.9949068168934241	0.0200618654489517 2	4.0195
4	0.9941246281408016	0.9954648526077098	0.9944911669147691	0.9954648526077098	0.0167252141982316 97	4.0648
6	0.9954770617321981	0.9956154336734694	0.9952556196658517	0.9956154336734694	0.0141125638037 92	4.2764
8	0.9958814885623937	0.9962531887755102	0.9958650430433779	0.9962531887755102	0.0128818154335021 97	4.1083
10	0.9960199291173533	0.9959431689342404	0.9959180774967511	0.9959431689342404	0.0131414178758859 63	4.6196

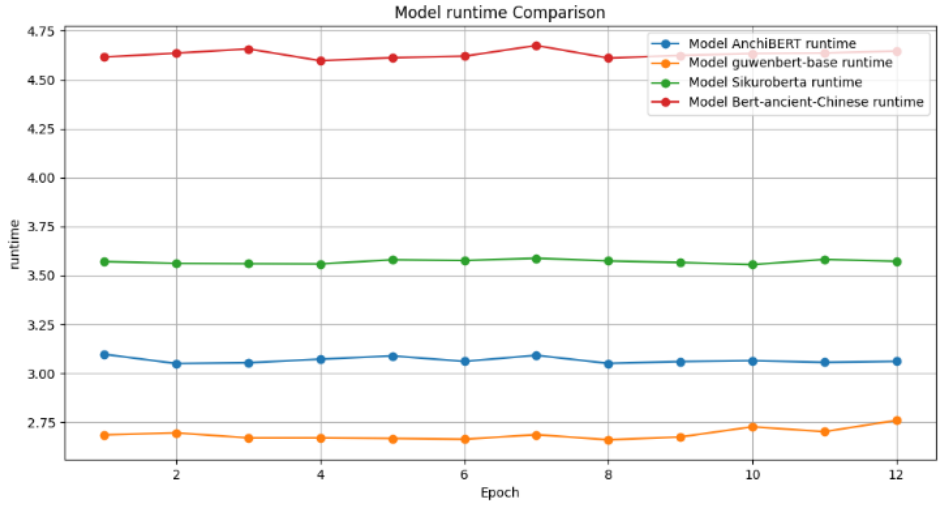
比较分析

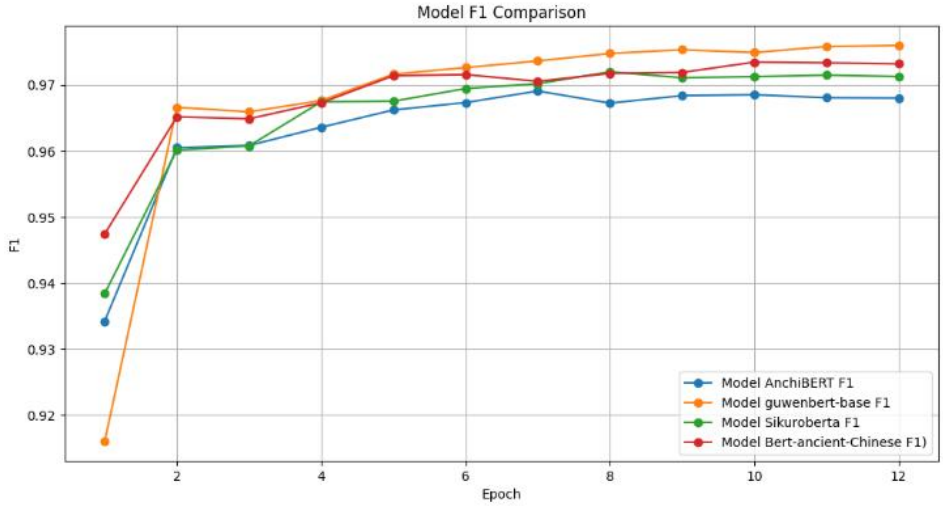
数据集 A

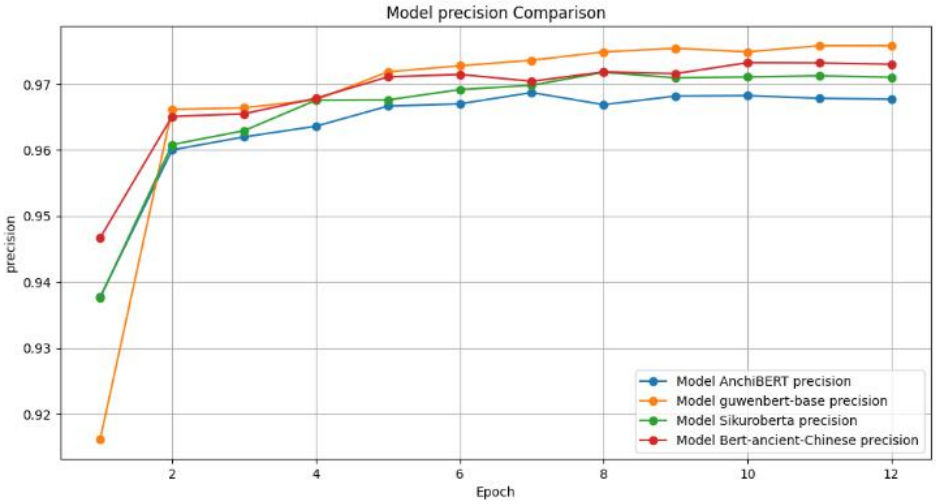

Model recall Comparison
recall
Epoch
Model AnchiBERT recall
Model guwenbert-base recall
Model Sikuroberta recall
Model Bert-ancient-Chinese recall


Model Loss Comparison
Loss
Epoch
Model AnchiBERT loss
Model guwenbert-base loss
Model Sikuroberta loss
Model Bert-ancient-Chinese loss


Model Accuracy Comparison
Accuracy
Epoch
Model AnchiBERT Accuracy
Model guwenbert-base Accuracy
Model Sikuroberta Accuracy
Model Bert-ancient-Chinese Accuracy







guwenbert-base

guwenbert-base 在各方面表现优越。f1、precision、accuracy、recall 在 1-2 个 epoch 之间

有显著提升，在后续 epoch 中表现优秀，处于最高位。loss 在 1-2 个 epoch 之间有显著下降，在后续 epoch 中表现优秀，处于最低位。训练时间在四个模型中最短。

Bert-ancient-Chinese

Bert-ancient-Chinese 在整个训练中相对优秀。f1、precision、accuracy、recall 在 1-2 个 epoch 之间变化最小，在后续 epoch 中表现稳定，处于第二高位。loss 在 1-2 个 epoch 之间也变化最小，在后续 epoch 中表现平稳，处于第二低位。但训练时间在四个模型中最长。

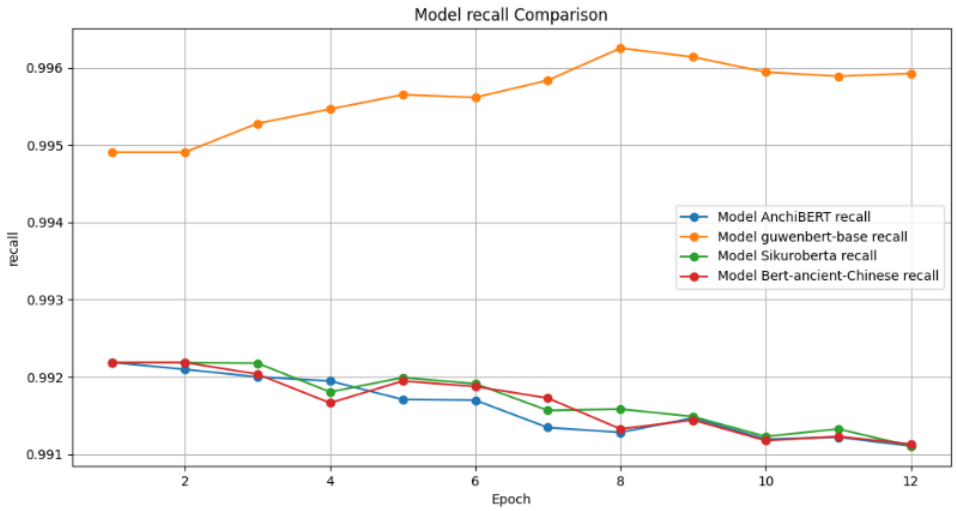
AnchiBERT

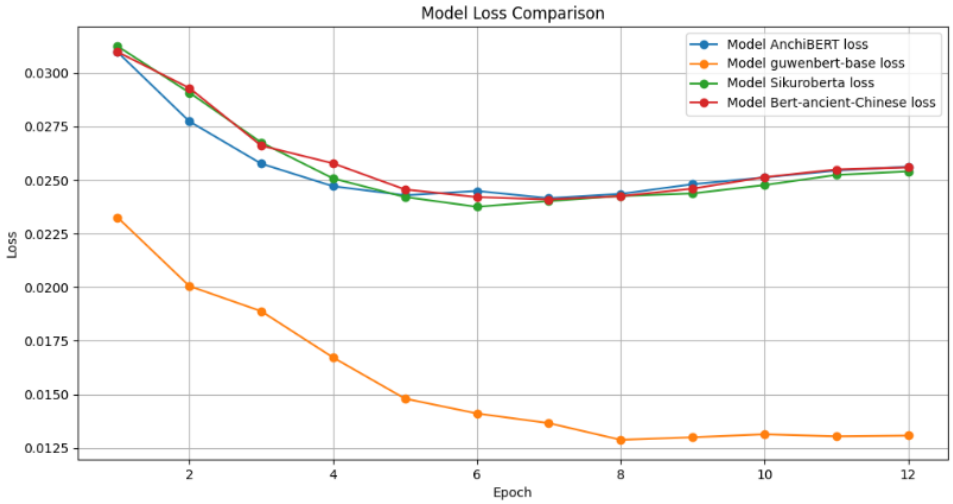
AnchiBERT 在整个训练中相对差。f1、precision、accuracy、recall 在 1-2 个 epoch 之间变化较小，在后续 epoch 中表现稳定，处于最低位。loss 在 1-2 个 epoch 之间也变化较小，在后续 epoch 中表现平稳，处于最高位。训练时间在四个模型中是第二短。

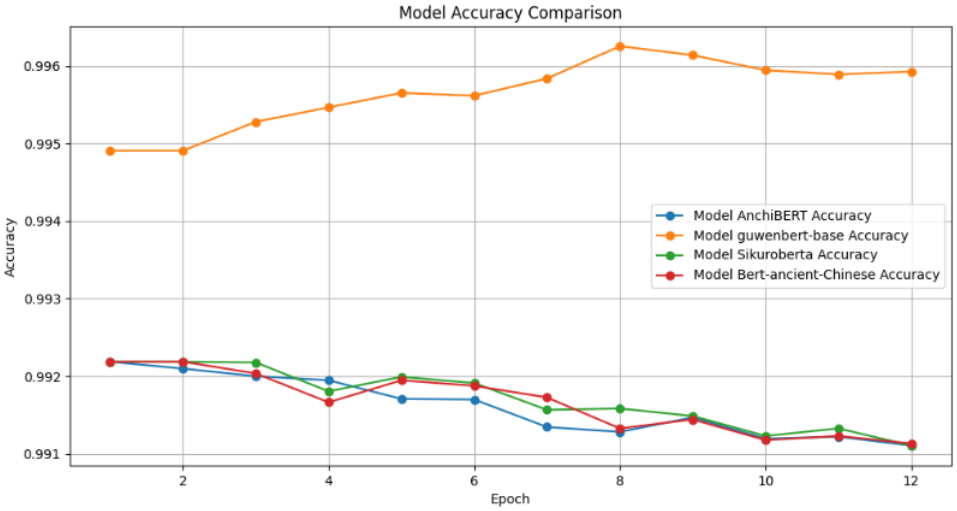
SikuRoberta

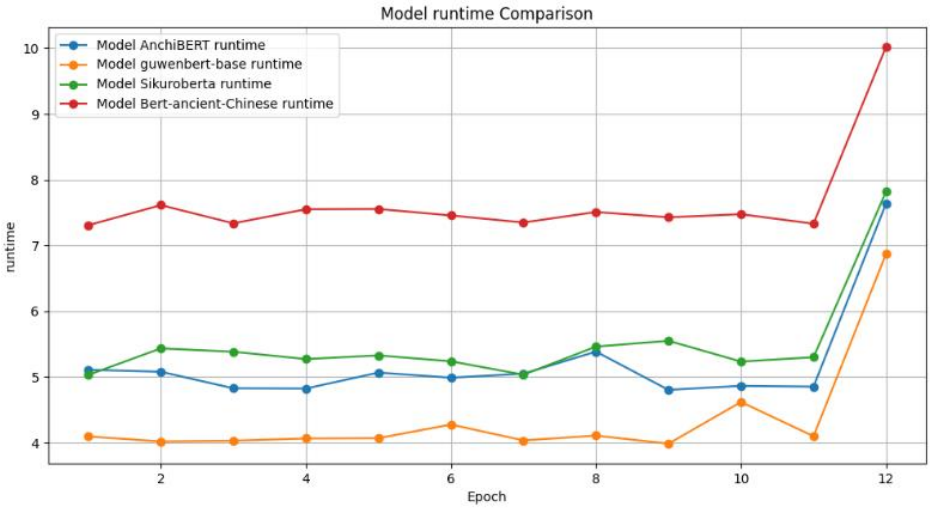
SikuRoberta 与 AnchiBERT 在整个训练中类似，各方面比 AnchiBERT 相对较好，但训练时间处在第二长。

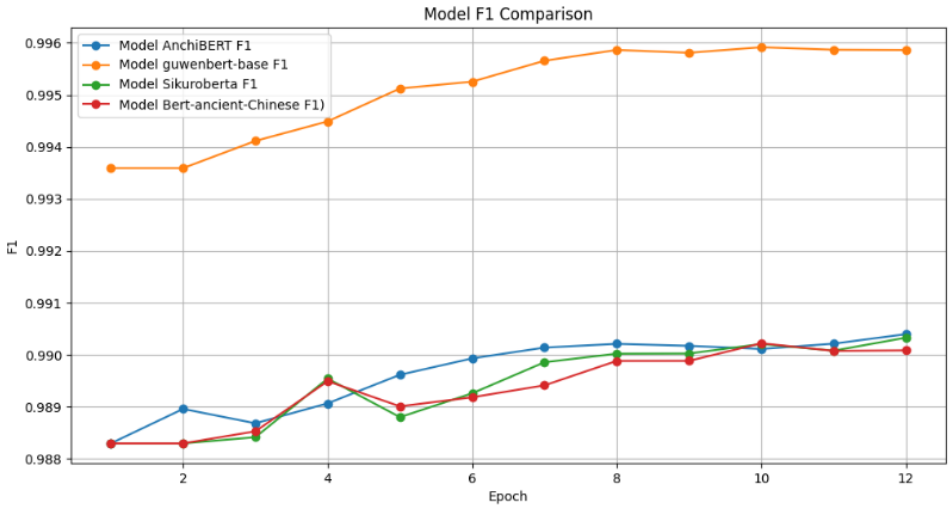
数据集 B

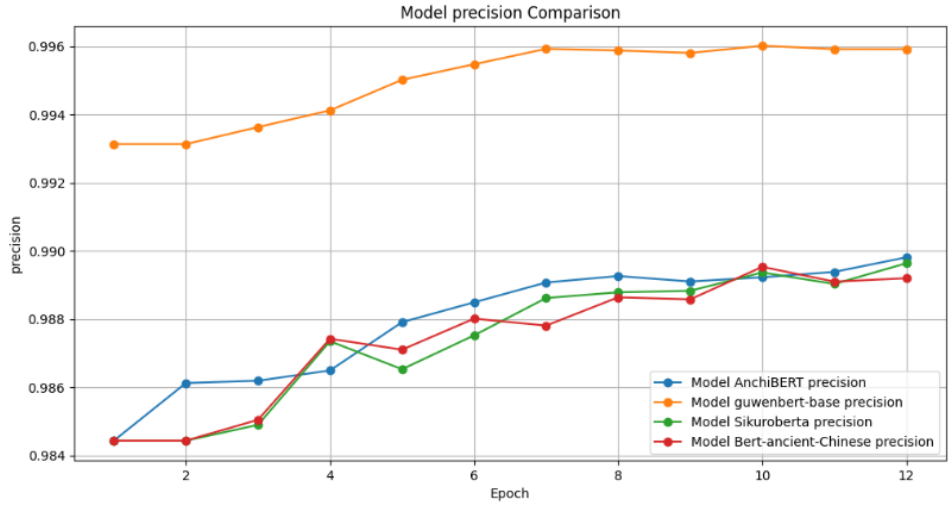



Model Loss Comparison
Model AnchiBERT loss
Model guwenbert-base loss
Model Sikuroberta loss
Model Bert-ancient-Chinese loss
Loss
Epoch


Model Accuracy Comparison
Model AnchiBERT Accuracy
Model guwenbert-base Accuracy
Model Sikuroberta Accuracy
Model Bert-ancient-Chinese Accuracy
Accuracy
Epoch


Model runtime Comparison
Model AnchiBERT runtime
Model guwenbert-base runtime
Model Sikuroberta runtime
Model Bert-ancient-Chinese runtime
runtime
Epoch





guwenbert-base

guwenbert-base 在各方面表现优越。Recall、accuracy、f1 在训练中逐步提升，precision 在 1-6 个 epoch 之间有显著上升，在后续 epoch 中较为平稳，处于最高位。loss 在 1-6 个 epoch 之间有显著下降，在后续 epoch 中较为平稳，处于最低位。训练时间在四个模型中最短。

Bert-ancient-Chinese

Bert-ancient-Chinese 在各方面都明显不如 guwenbert-base。训练时间在四个模型中最长。

AnchiBERT

AnchiBERT 在各方面都明显不如 guwenbert-base，Recall、accuracy、f1、precision、loss 同 Bert-ancient-Chinese 的相似。训练时间在四个模型中处在第二短。

SikuRoberta

SikuRoberta 在各方面都明显不如 guwenbert-base，Recall、accuracy、f1、precision、loss 同 Bert-ancient-Chinese 的相似。训练时间在四个模型中处在第二长。

综合比较

guwenbert-base 表现相对出色，且在整个训练过程中稳定增长。

实验整体分析

数据集 A

模型	Micro-Average Precision	Micro-Average Recall	Micro-Average F1	Macro-Average Precision	Macro-Average Recall	Macro-Average F1
Bert-ancient-Chinese	0.973448620996441 2	0.973448620996441 2	0.973448620996441 2	0.871082270984842 3	0.849443827644908 1	0.859951619778813 2
SikuRoberta	0.971613656583629 9	0.971613656583629 9	0.971613656583629 9	0.864802897459217 6	0.842991834751582 3	0.853425216439090 3
AnchiBERT	0.968471975088968	0.968471975088968	0.968471975088968	0.842688582019100 9	0.820604335138243 4	0.831015268909826 9
guwenbert-base	0.976201067615658 4	0.976201067615658 4	0.976201067615658 4	0.884164243125106 3	0.868039842633785 1	0.875817409603484 3

数据集 B

模型	Micro-Average Precision	Micro-Average Recall	Micro-Average F1	Macro-Average Precision	Macro-Average Recall	Macro-Average F1
Bert-ancient-Chinese	0.991133432539682 6	0.991133432539682 6	0.991133432539682 6	0.243646333301843 82	0.216991210393475 88	0.216207522904437 54
SikuRoberta	0.991106859410430 9	0.991106859410430 9	0.991106859410430 9	0.231851156607637 94	0.230711271903512 04	0.221999412876054 57
AnchiBERT	0.991106859410430 9	0.991106859410430 9	0.991106859410430 9	0.251258963235705 5	0.247649682969685 84	0.236973342054263 01
guwenbert-base	0.995925453514739 3	0.995925453514739 3	0.995925453514739 3	0.404791392534700 26	0.411098252460292 7	0.391438397052115 8

Micro-Average 指标反映了所有类别的总体性能，这意味着在考虑样本不平衡的情况下，guwenbert-base 在整体上取得了最好的平衡性能。
Macro-Average 指标展示了模型对所有类别同等对待的性能，guwenbert-base 表现最佳，表明其对于少数类别的识别能力较强。