Named Entity Recognition and Relation Extraction for Linking Mathematical Symbols to their descriptions

Introduction- Motivation for your project with examples.

Mathematical symbols are often isolated within the text in the expansive field of scientific documents. Linking these symbols to their descriptions is not only an academic exercise but also an essential requirement for scientific document comprehensibility. The motivation for this project stems from the observation that mathematical symbols, while concise, usually lack the necessary contextual background for interpretation.

SemEval's 2022 task 12, "Linking mathematical symbols to their descriptions", can be divided into two broad parts. The first sub-task is **Named Entity Recognition**, which establishes the spans of the mathematical symbols and descriptions. The subsequent sub-task is **Relation Extraction**, which draws the relations between the mathematical symbols and descriptions.

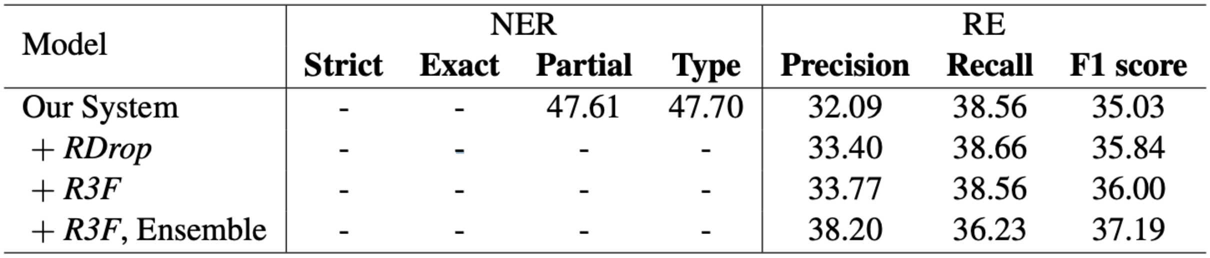
Related Work-

This paper (https://aclanthology.org/2022.semeval-1.230/) states the results and discusses the top teams of the competition. The Symlink track at SemEval-2022 received 4 system description paper submissions.

Popovic and Laurito (2022) (https://arxiv.org/abs/2203.05325) proposed an end-to-end joint entity and relation extraction approach based on transformers. They use a DL-MNAV model. It has 95.43% and 79.17% for physics and math content, respectively, while achieving macro F1 scores of only 19.23% and 13.84% for computer science and economics related content in relation extraction.
Ping and Chi (2022) participated in the Entity Extraction only. They finetuned a BERT model for each domain. F1 type for entity extraction is 16% and other scores have not been reported.

der Goot (2022) (https://aclanthology.org/2022.semeval-1.233/) proposed to pretrain a language model and re-finetune after multiple tasks.

Lee and Na (2022) (https://aclanthology.org/2022.semeval-1.231.pdf) used a symbol tokenizer and MRC based NER and span-pair-classification with solid markers is used for relation extraction.



Model	NER				RE		
	Strict	**Exact**	**Partial**	**Type**	**Precision**	**Recall**	**F1 score**
Our System	-	-	47.61	47.70	32.09	38.56	35.03
+ *RDrop*	-	-	-	-	33.40	38.66	35.84
+ *R3F*	-	-	-	-	33.77	38.56	36.00
+ *R3F*, Ensemble	-	-	-	-	38.20	36.23	37.19

Partial: This metric counts an entity mentioned as correct if there's partial overlap between the predicted entity boundaries and the true entity boundaries.
Type: This evaluates whether the predicted entity type matches the true entity type regardless of the boundary prediction. The scores are reported over all the subject files whereas we have only used CS.
(2nd team on the leaderboard)

Methodology

Our project implements a two-stage pipeline model that initially identifies mathematical symbols and their potential descriptions using a Named Entity Recognition (NER) approach and subsequently links these entities through a Relation Extraction (RE) model.

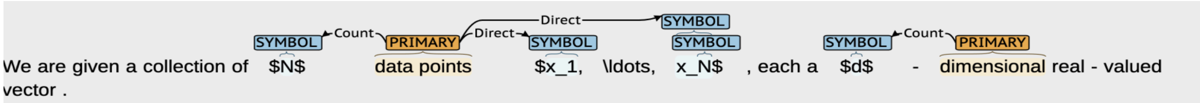
Model Architecture:

1) NER- We tried using multiple BERT based models for NER. The best performing model is SciDeBERTa. We trained it for 10 epochs on our training data.

2) RE- We used LUKE (Language Understanding with Knowledge-based Embeddings) (https://huggingface.co/docs/transformers/model_doc/luke), which is a transformer-based model. It is pre-trained on a large corpus with a self-attention mechanism. It is especially effective for relation extraction as it leverages both contextual word embeddings and discrete entity information. The fact that LUKE is a pretrained model allows it to learn richer representations of the texts and entities, which reduces our need for labelling data while fine-tuning for a task like relation extraction.

Dataset/Experimental Setup

Link to dataset: https://competitions.codalab.org/competitions/34011#participate-get_data
We used the CS related data files


We are given a collection of N data points $x_1, \ldots, x_N$, each a d - dimensional real - valued vector .

Example:

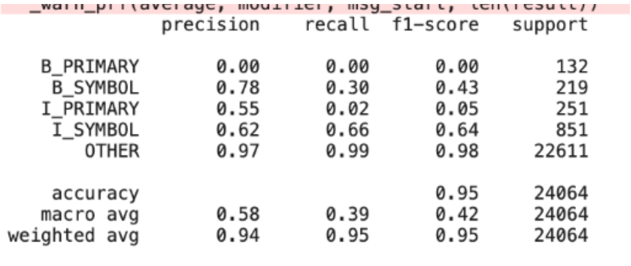
```
{
  ”id”: ”1503.01158v2...”,
  ”phase”: ”test”,
  ”topic”: ”cs.ai”,
  ”document”: ”1503.01158v2...”,
  ”paragraph”: ”paragraph_48”,
  ”text”: ”... with a covariance
  matrix of $I$ ; that is , ...”,
  ”entity”: {
   ”T1”: {
    ”eid”: ”T1”,
    ”label”: ”SYMBOL”,
    ”start”: 325,
    ”end”: 326,
    ”text”: ”I”
   },
   ”T2”: {
    ”eid”: ”T2”,
    ”label”: ”PRIMARY”,
    ”start”: 303,
    ”end”: 320,
    ”text”: ”covariance matrix”
   }
  },
  ”relation”: {
   ”R1”: {
    ”rid”: ”R1”,
    ”label”: ”Direct”,
    ”arg0”: ”T2”,
    ”arg1”: ”T1”
   }
  }
}
```

Results/Findings

In NER, we tried multiple BERT based models. We used open source models pretrained/ fine tuned on scientific data.
Models not pretrained on scientific data were classifying all words as OTHER.
Eg. roberta and distillbert

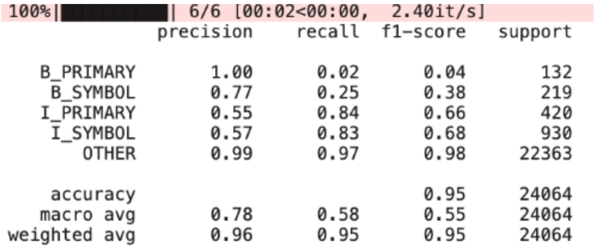
Here are the reports of some of them on the validation set.
Trained on 3 epochs.

Matscibert (https://www.nature.com/articles/s41524-022-00784-w)
By IIT Delhi, trained on a large corpus of peer-reviewed materials science publications



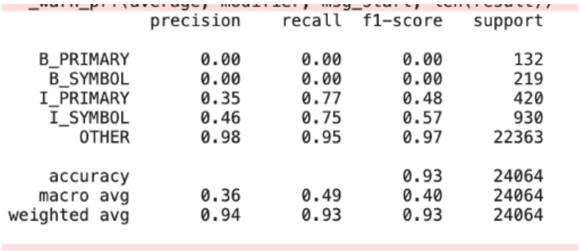
```
  _warn_prf(average, modifier, msg_start, len(result))
              precision    recall  f1-score   support

   B_PRIMARY       0.00      0.00      0.00       132
    B_SYMBOL       0.78      0.30      0.43       219
   I_PRIMARY       0.55      0.02      0.05       251
    I_SYMBOL       0.62      0.66      0.64       851
       OTHER       0.97      0.99      0.98     22611

    accuracy                           0.95     24064
   macro avg       0.58      0.39      0.42     24064
weighted avg       0.94      0.95      0.95     24064
```

SciDeBERTa (https://paperswithcode.com/paper/SciDeBERTa-learning-deberta-for-science)
Pretrained DeBERTa, which was trained with a general corpus, with the science technology domain corpus



```
100%|██████████| 6/6 [00:02<00:00,  2.40it/s]
              precision    recall  f1-score   support

   B_PRIMARY       1.00      0.02      0.04       132
    B_SYMBOL       0.77      0.25      0.38       219
   I_PRIMARY       0.55      0.84      0.66       420
    I_SYMBOL       0.57      0.83      0.68       930
       OTHER       0.99      0.97      0.98     22363

    accuracy                           0.95     24064
   macro avg       0.78      0.58      0.55     24064
weighted avg       0.96      0.95      0.95     24064
```

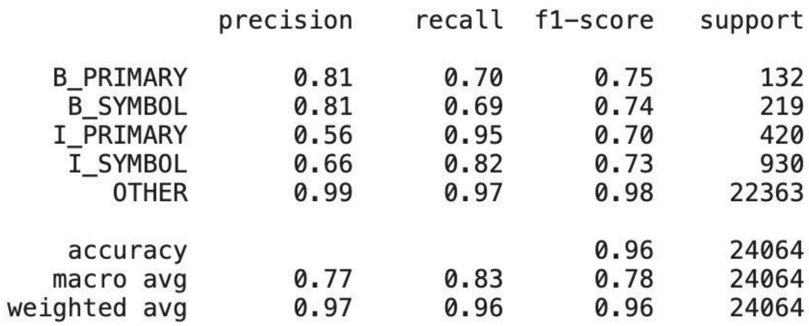
Robeta (https://arxiv.org/abs/1907.11692)



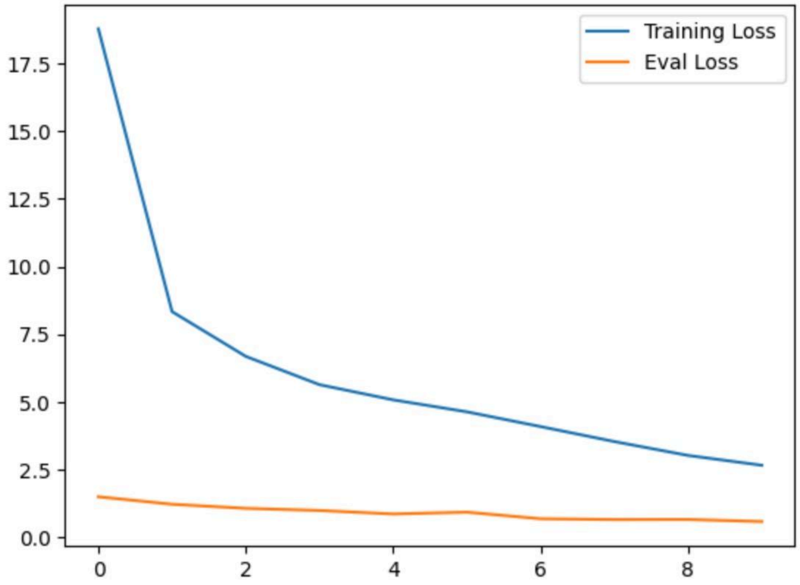
```
  _warn_prf(average, modifier, msg_start, len(result))
              precision    recall  f1-score   support

   B_PRIMARY       0.00      0.00      0.00       132
    B_SYMBOL       0.00      0.00      0.00       219
   I_PRIMARY       0.35      0.77      0.48       420
    I_SYMBOL       0.46      0.75      0.57       930
       OTHER       0.98      0.95      0.97     22363

    accuracy                           0.93     24064
   macro avg       0.36      0.49      0.40     24064
weighted avg       0.94      0.93      0.93     24064
```

Discussion/Analysis

After training for 10 epochs, we have reached this result. This is better than training for 3 epochs as there is a steady loss decrease.



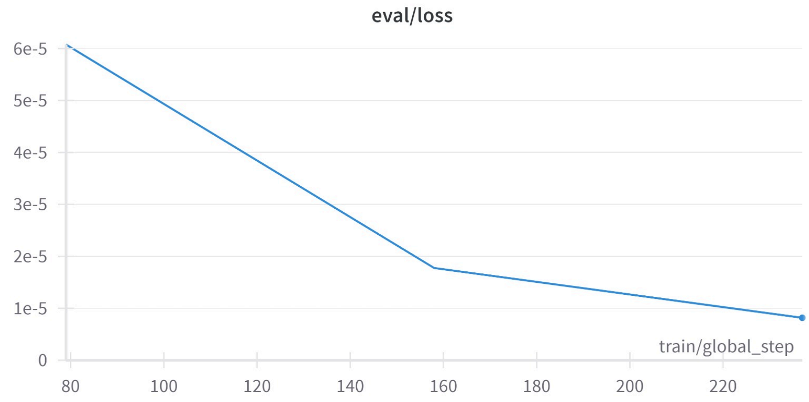
```
              precision    recall  f1-score   support

   B_PRIMARY       0.81      0.70      0.75       132
    B_SYMBOL       0.81      0.69      0.74       219
   I_PRIMARY       0.56      0.95      0.70       420
    I_SYMBOL       0.66      0.82      0.73       930
       OTHER       0.99      0.97      0.98     22363

    accuracy                           0.96     24064
   macro avg       0.77      0.83      0.78     24064
weighted avg       0.97      0.96      0.96     24064
```



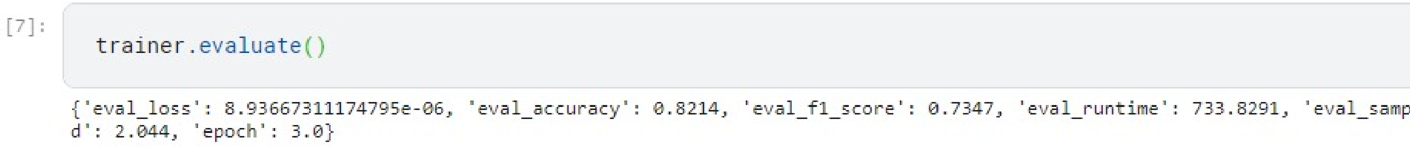
All the previous work used BERT based architectures and our best output is using SciDeBERTa which is pretrained on domain specific knowledge. Our F1 scores for NER are better than the existing metrics.

Among four submitted systems, MaChAmp (der Goot, 2022) and AN(L)P (Ping and Chi, 2022) teams used the default tokenizer from either BERT or mBERT, which are not designed for scientific documents so they achieved the lowest Named Entity Recognition performance. Whereas AIFB-WebScience (Popovic and Laurito, 2022) and JBNU-CCLab (Lee and Na, 2022) achieved much higher performances.

For RE, we trained the model for 4 epochs and got a steady decrease in losses across them



These were the accuracy and f1 scores:



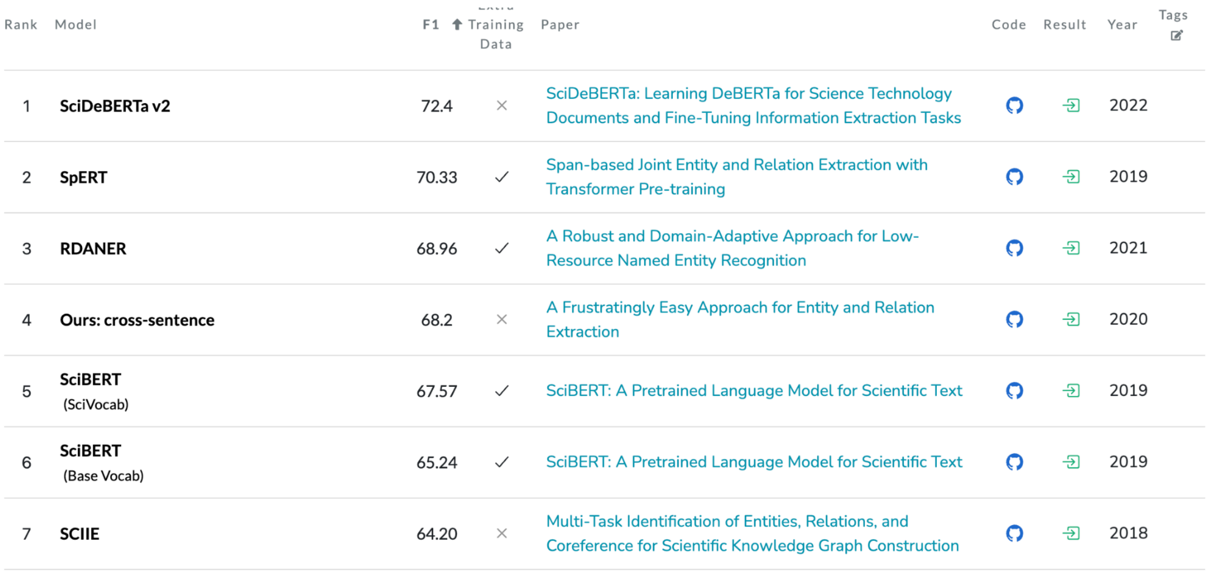
```
trainer.evaluate()
```

```
{'eval_loss': 8.93667311174795e-06, 'eval_accuracy': 0.8214, 'eval_f1_score': 0.7347, 'eval_runtime': 733.8291, 'eval_samp
d': 2.044, 'epoch': 3.0}
```

Accuracy: 0.8214
F1 Score: 0.7347

Conclusion and Future Work

SciDeBERTa(2022) works best for NER. It beats the existing metrics. This model came out in 2022, after this challenge, and is the state-of-the-art in NER on scientific texts.



Rank	Model	F1 ↑	Extra Training Data	Paper	Code	Result	Year	Tags
1	**SciDeBERTa v2**	72.4	×	SciDeBERTa: Learning DeBERTa for Science Technology Documents and Fine-Tuning Information Extraction Tasks			2022	
2	**SpERT**	70.33	✓	Span-based Joint Entity and Relation Extraction with Transformer Pre-training			2019	
3	**RDANER**	68.96	✓	A Robust and Domain-Adaptive Approach for Low-Resource Named Entity Recognition			2021	
4	**Ours: cross-sentence**	68.2	×	A Frustratingly Easy Approach for Entity and Relation Extraction			2020	
5	**SciBERT** (SciVocab)	67.57	✓	SciBERT: A Pretrained Language Model for Scientific Text			2019	
6	**SciBERT** (Base Vocab)	65.24	✓	SciBERT: A Pretrained Language Model for Scientific Text			2019	
7	**SCIIE**	64.20	×	Multi-Task Identification of Entities, Relations, and Coreference for Scientific Knowledge Graph Construction			2018	

We can finetune and train NER models for all the spheres such as bio, physics and economics data given.

For RE, LUKE was an excellent architecture as it significantly enhanced the model's capabilities in processing and understanding entities within the text. Future work can use an exhaustive dataset with symbols from physics, economics math and biology, rather than only computer science.
Furthermore, adapting the model for specific domains (eg medical texts), where the entities and relations have some unique characteristics can also be part of the future work, as standard models may not be able to handle that well. We can also incorporate external knowledge bases

The scores are reported over all the subject files whereas we have only used CS.

Models:
https://drive.google.com/drive/folders/1UHXPNZ92zuv0ydJaC1hHi8Nj3LN8ZU8n?usp=sharing