Evaluating BERT Variants and Classification Approaches for Chemical Named Entity Recognition

J. Spencer Morris
University of California, Berkeley
jspencermorris@berkeley.edu

Abstract

Addressing the problem of chemical Named Entity Recognition (NER) in scientific texts, this study evaluates the performance of BERT, SciBERT, and SpanBERT models with various classification head configurations. Our results indicate that SciBERT, especially when fine-tuned using the last four hidden layers, achieves the highest accuracy. These findings highlight the importance of domain-specific pre-training and suggest future research directions, including alternative classification head approaches.

1 Introduction

The frontier of scientific knowledge, especially that of practical industrial applications, is typically communicated via patents and scholarly articles. However, the unstructured nature of these texts poses significant challenges for data extraction and aggregation. In the chemical domain, accurately identifying and extracting chemical entities is crucial for building knowledge graphs, enabling physical modeling, and conducting patent infringement analysis. Named entity recognition (NER) in this context is particularly difficult due to the specialized and complex nature of chemical terminology. For example, organic chemicals are often described in the IUPAC nomenclature using long string sequences to capture hierarchical molecular structure. Furthermore, subtle differences in chemical functionalities and combinations thereof can lead to differences in participation roles in a given reaction type.

Recent advances in transformer-based models, such as BERT, offer promising avenues for enhancing NER performance in scientific texts. While there exist transformer models pre-trained on corpora within the chemistry domain such as ChemBERTa (Chithrananda et al.), such models rely on representing chemical entities as SMILES strings. To our knowledge, there are no pre-trained models specialized in IUPAC representation for chemicals, which is more common in the literature. However, SciBERT (Beltagy et al.), developed by further pretraining BERT on a variety of scientific literature, including a fraction of natural-language chemistry texts, has achieved superior results on chemical NER tasks compared to BERT. In this paper, we additionally explore SpanBERT (Joshi et al.), which masks spans of tokens during training instead of single tokens. Since chemical entities often span many tokens, we hypothesized that SpanBERT may offer superior performance to BERT for chemical NER.

A key aspect of our research is the comparison between feature-based models and fine-tuned models. The fine-tuned approach involves adapting all parameters of a BERT model to the NER task, which can lead to better performance but is computationally expensive. In contrast, the feature-based approach uses fixed BERT embeddings as features for a separate model, and was found by Devlin et al. to be marginally less performant than the fine-tuned approach but also less computationally costly. We aimed to reproduce this result within the chemical domain.

In the original BERT paper, the authors' feature-based approach involved concatenation of the last four hidden layers, in contrast to the fine-tuned approach which utilized the final hidden layer. Subsequent research by Tenney et al. has shown that, while semantic knowledge in BERT is distributed throughout all layers, there is evidence that it is largely found in the mid-upper layers. As such, we further investigated if concatenation of the last four hidden layers further improved performance, as compared to relying solely on the last hidden layer, in fine-tuned models.

In this paper, we present our investigation of utilizing the last hidden layer as compared to concatenating the last four hidden layers in various BERT-based models, both with and without fine-tuning.

2 Background

Previous work in chemical NER has advanced the field by leveraging specialized corpora such as CHEMDNER (Krallinger et al.), which focused on identifying classifying chemical representations (eg. abbreviations, families, formulae, etc.), and ChEMU (He et al.), which focused on identifying the participation roles and workup steps in chemical reactions. However, the scope of these corpora is limited to specific types of chemical texts, which may not encompass the full range of terminologies and contexts encountered in the broader chemistry domain.

The Chemical Language Understanding Benchmark (CLUB) dataset, created by Kim et al., extends the challenge of chemical NER by incorporating data subsets from both journal articles as well as patents. In introducing CLUB, the authors developed RoBERTa-lit-pat, which was further pretrained on both journal articles and patents in the chemistry domain, and achieved a state-of-the-art (SOTA) combined F1 score of 0.7818. To our knowledge, RoBERTa-lit-pat is not publicly accessible.

Various pre-trained models have been developed for specialized relevant corpora and tasks. SciBERT (Beltagy et al.) was developed by further pre-training BERT on a large corpus of scientific literature. The model relies on a specialized scientific vocabulary, SciVocab. Although it was trained on biomedical and computer science texts, it has been demonstrated as superior to BERT for chemical NER.

Joshi et al. introduced SpanBERT, which relies on masking spans of tokens, as an approach for improved performance for token-level classification for tasks involving long spans of text. The authors demonstrated superior performance for coreference resolution and relation extraction. Jianfu et. al found SpanBERT to be slightly more performant than BERT for medical NER in medical texts, suggesting its utility for specialized scientific text.

In the seminal BERT paper (Devlin et al.), the normal fine-tuned approach was compared to a feature-based approach for NER. In the feature-based approach, the model's layers were frozen and the final four layers were used as contextual embeddings by using their concatenation as input into an LSTM model for token-level classification. Although the feature-based approach yielded inferior performance (with an F1 96.1 compared to 96.4), it was only marginally worse and required substantially less training time.

By exploring a variety of probing techniques, Tenney et al. found that different depths of BERT layers encode different kinds of linguistic knowledge, with semantic information primarily captured in the mid-to-upper layers. We found this an interesting proposition for explaining the original BERT paper's feature-based results. Furthermore, given that semantic understanding is most critical for chemical NER, we were motivated to explore if leveraging multiple upper-layers of fine-tuned BERT models yielded any improvement as compared to the last final layer alone.

Our study builds on these insights by exploring a variety of BERT-based models and configurations for chemical NER, including SciBERT and SpanBERT. We examine both feature-based and fine-tuned approaches, with a particular focus on the performance implications of using the last hidden layer versus concatenation of the last 4 layers. Our findings indicate that feature-based approaches are significantly less performant than fine-tuned approaches, and that there is some evidence that concatenation of multiple upper layers can lead modest improvements. We also find that SpanBERT may not be suitable for our task.

3 Methods

3.1 Data and preprocessing

In this study, we used the CLUB dataset for chemical NER. The 'Battery' subset was derived from journal articles with annotations for the chemical components in solid-state batteries, whereas the 'Catalyst' subset was derived from patents with annotations for the chemical constituents in polymerization processes. The IOB2 scheme was used for token-level classification. The CLUB datasets were accessed from HuggingFace[1]. A summary of the data subsets is included in the Appendix.

To prepare the data for model training and evaluation, we followed several preprocessing steps. Since only 'train' and 'evaluation' splits were provided, we further partitioned the train set at 85%/15% to generate new 'train' and 'validation' splits, to be used to monitor overfitting. We tokenized the text using the tokenizer specific to each pre-trained model and specified a maximum input length of 256. Special tokens were used to denote the beginning and end of sequences, padding, and continuing tokenized subwords.

3.2 Model architectures

Three pre-trained models were explored in the present study: BERT-base-cased, SciBERT-scivocab-cased, and SpanBERT-cased. All pre-trained models were accessed from HuggingFace.

We furthermore explored three distinct model configurations for each pre-trained model:

Feature-based with last 4 layers into an LSTM: In this configuration, we replicated the most performant feature-based method identified by Devlin et al., in which the last four hidden layers are concatenated and used as input features into a bi-directional LSTM layer. The biLSTM layer is then fed into a final classification head for token-level predictions.

Fine-tuned with last layer: This is the prototypical BERT configuration, which involves fine-tuning all parameters of the pre-trained model, then feeding the final hidden layer into a classification head to make token-level predictions.

Fine-tuned with last 4 layers: Similar to the fine-tuned model with the last layer, this configuration also involves fine-tuning, but it concatenates the outputs of the last 4 hidden layers before passing them to the classification head.

3.3 Grid-search and training procedures

We conducted a grid-search to identify optimal hyperparameters for each model configuration. The hyperparameters explored were the learning rates (2e-5, 5e-5, 8e-5) and the batch sizes (16, 32, and 64), representing a total of 9 combinations. Each combination of hyperparameters was randomly initialized 3 times with pre-defined seeds to measure performance variability and monitor result reliability.

An initial exploration of the train and validation splits during model-fitting consistently revealed overfitting onset around the third epoch, and so all models were fine-tuned for 3 epochs.

Training involved optimizing a custom cross-entropy loss function that considered only the labels for a given word's first subword (as illustrated in the Appendix).

We used AdamW as the optimizer with a warm-up ratio of 0.1 and a weight decay of 0.01.

Approach	Battery	Catalyst	Combined
feature based	3.7	9.1	6.4
fine tuned (1L)	4.5	11.3	7.9
fine tuned (4L)	5	13	9

Table 1: The mean fitting times, in minutes, for the train split. The feature-based approach requires substantially less time to train.

3.4 Evaluation

Our primary metric for evaluation was the overall macro-average F1 score (across all classes), for both datasets, which were then averaged into a combined F1 score. All scoring was performed using the standard scoring scripts for token-level seqeval metrics[2].

We use as our baseline BERT$_{\text{Base-Cased}}$ with the standard fine-tuning approach.

4 Results and discussion

4.1 Hyperparameter selection

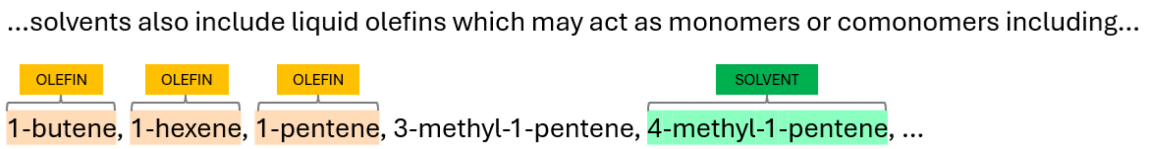
In our experiments, we found that the combination of a batch size of 16 and an initial learning rate of 8e-5 consistently yielded the best performance across all model architectures. An example of the grid-search results can be found in the Appendix.

4.2 Training time analysis

All training was performed using an A100 GPU hosted in Google Colab. The feature-based approach demonstrated a significant advantage in terms of training speed compared to both fine-tuned approaches. The mean fitting times for the train split are summarized in Table 1. The feature-based method was faster to train by approximately

[1] https://huggingface.co/datasets/bluesky333/chemical_language_understanding_benchmark

[2] https://pypi.org/project/seqeval/


...solvents also include liquid olefins which may act as monomers or comonomers including...

1-butene, 1-hexene, 1-pentene, 3-methyl-1-pentene, 4-methyl-1-pentene, ...

Figure 1: Misclassification pattern seen in SpanBERT but not the other two pre-trained models. All entities on the second line are solvents, but only SpanBERT misclassified some as olefins.

Base Model	Approach	Battery	Catalyst	Combined
$\text{BERT}_{\text{Base-Cased}}$	feature based	0.693 +/- 0.002	0.603 +/- 0.014	0.648 +/- 0.003
	fine tuned (1L)	0.767 +/- 0.003	0.674 +/- 0.008	0.721 +/- 0.003
	fine tuned (4L)	0.777 +/- 0.007	0.681 +/- 0.010	0.729 +/- 0.003
$\text{SciBERT}_{\text{Scivocab-Cased}}$	feature based	0.654 +/- 0.009	0.620 +/- 0.011	0.637 +/- 0.001
	fine tuned (1L)	0.757 +/- 0.003	0.687 +/- 0.015	0.722 +/- 0.002
	fine tuned (4L)	**0.788** +/- 0.006	**0.707** +/- 0.009	**0.747** +/- 0.002
$\text{SpanBERT}_{\text{Cased}}$	feature based	0.641 +/- 0.005	0.540 +/- 0.019	0.591 +/- 0.003
	fine tuned (1L)	0.771 +/- 0.009	0.647 +/- 0.009	0.709 +/- 0.005
	fine tuned (4L)	0.768 +/- 0.005	0.656 +/- 0.001	0.712 +/- 0.004

Table 2: Summary of the overall F1 scores, for the evaluation data split, across different models and configurations. Our baseline model, shown in gray, involved the traditional fine-tuning approach on the base BERT model. SciBERT achieved superior performance compared to the other pre-trained models. The feature-based approach produced markedly lower F1 scores than the traditional fine-tuning approach. Concatenation of the final 4 layers showed marginal improvement compared to using the last hidden layer.

22% compared to the fine-tuned (1-layer) approach. This finding aligns with the anticipated reduction in computational complexity when using a feature-based approach, where the pre-trained model weights are used directly as inputs into a biLSTM prior to classification.

4.3 Performance on specific labels

All models encountered difficulty with COATING_METHOD entities in the battery subset and SOLVENT entities in the catalyst subset. These labels were, by far, the least common in their respective datasets, which likely contributed to the high misclassification rates, often being labeled incorrectly as ‘O.’ Confusion matrices for both datasets from our best performing model are shown in the Appendix.

4.4 Comparative performance of pre-trained models

Surprisingly, SpanBERT performed worse than BERT, while SciBERT outperformed both. This may be due to the nature of the training data for each model. It is not surprising that SciBERT, which was pre-trained on scientific literature, has some advantages as compared to the other models, given its pre-training corpus. However, some of the misclassifications unique to SpanBERT may arise from that model’s approach to language masking. In Figure 1, for example, an excerpt is shown that mentions solvents as well as olefins, and many entities in the subsequent list of solvents are misclassified as olefins. This pattern was only observed for the SpanBERT models, suggesting cases where masking spans is actually detrimental to overall performance.

4.5 Feature-Based vs. Fine-Tuned Approaches

Our experimental results are summarized in Table 2. This table shows the 3 tested classification head approaches for each of the 3 pre-trained models we evaluated. All of them incorporated the same hyperparameter combination.

On average, the feature-based approach yielded a combined overall F1 score approximately 0.092 less than the fine-tuned approach using 1 layer. This difference is significantly larger than the marginal difference reported in the original BERT paper, which may be attributed to the complexity and specificity of chemical NER tasks. Our classification head incorporated a BiLSTM like the original BERT paper, but it is possible that a different head could be more suitable for chemical NER.

The fine-tuned models using the last 4 layers demonstrated modest improvements in F1 score compared to those using only the last hidden layer, with an average gain of 0.012. This improvement suggests that leveraging information from multiple upper layers can enhance the model's contextual understanding, which is critical for accurately identifying chemical entities.

Overall, the results indicate that SciBERT, with all parameters fine-tuned and with the classification head relying on the concatenation of the last 4 layers, is the most effective model for chemical NER tasks in this study. This finding underscores the importance of domain-specific pre-training, the strength of conventional masked language modeling, and the benefit of utilizing multiple upper layers for enhanced semantic understanding. The differences in performance among the models and configurations we explored provide valuable insights into the strengths and limitations of each approach, guiding future efforts in optimizing NER systems for specialized scientific domains.

Conclusion

The primary problem addressed in our study was the optimization of Named Entity Recognition (NER) models for identifying chemical entities within both journal articles and patents. Our goal was to compare the performance of different pre-trained models (BERT, SpanBERT, SciBERT) and classification head configurations (feature-based and fine-tuned) to determine the most effective approach for chemical NER tasks.

We found that the traditional fine-tuning approach outperformed the feature-based approach for our task much more significantly than was reported by Devlin et al. for the CoNLL dataset. Our classification head for feature-based models incorporated a BiLSTM like the original BERT paper, but it is possible that a different head could be more suitable for chemical NER, and exploring alternative structures could be an area of fruitful research.

We furthermore found that SciBERT outperformed both BERT and SpanBERT across all configurations, highlighting the importance of domain-specific pre-training.

The fine-tuning approach, particularly when using the last 4 layers, yielded the best performance. Given the importance of semantic relationships for components of chemical systems, we are interested in pursuing additional experiments based on other combinations of mid-upper hidden layers to our task.

References

Atilla Kaan Alkan, Cyril Grouin, Fabian Schussler, and Pierre Zweigenbaum. 2022. A Majority Voting Strategy of a SciBERT-based Ensemble Models for Detecting Entities in the Astrophysics Literature (Shared Task). In *Proceedings of the first Workshop on Information Extraction from Scientific Publications*, pages 145–150, Online. Association for Computational Linguistics.

Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. SciBERT: A Pretrained Language Model for Scientific Text. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3615–3620, Hong Kong, China. Association for Computational Linguistics.

Seyone Chithrananda, Gabriel Grand, and Bharath Ramsundar. 2020. ChemBERTa: Large-Scale Self-Supervised Pretraining for Molecular Property Prediction. Preprint at arXiv:2010.09885.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Jiayuan He, Dat Quoc Nguyen, Saber A Akhondi, Christian Druckenbrodt, Camilo Thorne, Ralph Hoessel, Zubair Afzal, Zenan Zhai, Biaoyan Fang, Hiyori Yoshikawa, Ameer Albahem, Lawrence Cavedon, Trevor Cohn, Timothy Baldwin, and Karin Verspoor. 2020. ChEMU 2020: Natural Language Processing Methods Are Effective for Information Extraction From Chemical Patents. In *Multimodality, and Interaction: 11th International Conference of the CLEF Association*, pages 237-254, Thessaloniki, Greece. Frontiers in Research Metrics and Analytics.

Mandar Joshi, Danqi Chen, Yinhan Liu, Daniel S. Weld, Luke Zettlemoyer, and Omer Levy.

2020. SpanBERT: Improving Pre-training by Representing and Predicting Spans. *Transactions of the Association for Computational Linguistics*, 8:64–77.

Yunsoo Kim, Hyuk Ko, Jane Lee, Hyun Young Heo, Jinyoung Yang, Sungsoo Lee, and Kyu-hwang Lee. 2023. Chemical Language Understanding Benchmark. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 5: Industry Track)*, pages 404–411, Toronto, Canada. Association for Computational Linguistics.

Martin Krallinger, Obdulia Rabal, Florian Leitner, Miguel Vazquez, David Salgado, Zhiyong Lu, Robert Leaman, Yanan Lu, Donghong Ji, Daniel M Lowe, Roger A Sayle, Riza Theresa Batista-Navarro, Rafal Rak, Torsten Huber, Tim Rocktäschel, Sérgio Matos, David Campos, Buzhou Tang, Hua Xu, Tsendsuren Munkhdalai, Keun Ho Ryu, SV Ramanan, Senthil Nathan, Slavko Žitnik, Marko Bajec, Lutz Weber, Matthias Irmer, Saber A Akhondi, Jan A Kors, Shuo Xu, Xin An, Utpal Kumar Sikdar, Asif Ekbal, Masaharu Yoshioka, Thaer M Dieb, Miji Choi, Karin Verspoor, Madian Khabsa, C Lee Giles, Hongfang Liu, Komandur Elayavilli Ravikumar, Andre Lamurias, Francisco M Couto, Hong-Jie Dai, Richard Tzong-Han Tsai, Caglar Ata, Tolga Can, Anabel Usié, Rui Alves, Isabel Segura-Bedmar, Paloma Martínez, Julen Oyarzabal, and Alfonso Valencia. 2015. The chemdner corpus of chemicals and drugs and its annotation principles. Journal of cheminformatics, 7(S1):S2.

Jianfu Li, Qiang Wei, Omid Ghiasvand, Miao Chen, Victor Lobanov, Chunhua Weng, and Hua Xu. 2022. A comparative study of pre-trained language models for named entity recognition in clinical trial eligibility criteria from multiple corpora. *BMC Medical Informatics and Decision Making*, 22 (Suppl 3), 235

Ian Tenney, Dipanjan Das, and Ellie Pavlick. 2019. BERT Rediscovers the Classical NLP Pipeline. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4593–4601, Florence, Italy. Association for Computational Linguistics.

A Appendices

			TRAIN SPLIT			VALIDATION SPLIT		
Model	Approach	(Batch Size, Init LR)	Battery	Catalyst	Combined	Battery	Catalyst	Combined
BERT$_{\text{Base-Cased}}$	fine tuned (1L)	(16, 2e-05)	0.761 +/- 0.016	0.734 +/- 0.013	0.748 +/- 0.000	0.699 +/- 0.013	0.613 +/- 0.023	0.656 +/- 0.003
		(16, 5e-05)	0.859 +/- 0.012	0.805 +/- 0.008	0.832 +/- 0.001	0.771 +/- 0.015	0.672 +/- 0.010	0.722 +/- 0.003
		(16, 8e-05)	0.894 +/- 0.006	0.814 +/- 0.009	**0.854** +/- 0.002	0.790 +/- 0.004	0.686 +/- 0.010	**0.738** +/- 0.003
		(32, 2e-05)	0.680 +/- 0.017	0.598 +/- 0.023	0.639 +/- 0.002	0.666 +/- 0.005	0.561 +/- 0.020	0.613 +/- 0.004
		(32, 5e-05)	0.753 +/- 0.012	0.695 +/- 0.016	0.724 +/- 0.001	0.704 +/- 0.015	0.609 +/- 0.013	0.656 +/- 0.003
		(32, 8e-05)	0.787 +/- 0.018	0.709 +/- 0.022	0.748 +/- 0.002	0.732 +/- 0.015	0.628 +/- 0.019	0.680 +/- 0.004
		(64, 2e-05)	0.549 +/- 0.070	0.343 +/- 0.054	0.446 +/- 0.016	0.565 +/- 0.061	0.337 +/- 0.039	0.451 +/- 0.018
		(64, 5e-05)	0.661 +/- 0.020	0.486 +/- 0.015	0.574 +/- 0.010	0.649 +/- 0.004	0.445 +/- 0.012	0.547 +/- 0.013
		(64, 8e-05)	0.687 +/- 0.027	0.508 +/- 0.027	0.597 +/- 0.010	0.664 +/- 0.023	0.440 +/- 0.024	0.552 +/- 0.016

Table 3: Grid-search results for the our baseline model. All models saw similar trends in performance across the tested hyperparameter combinations.

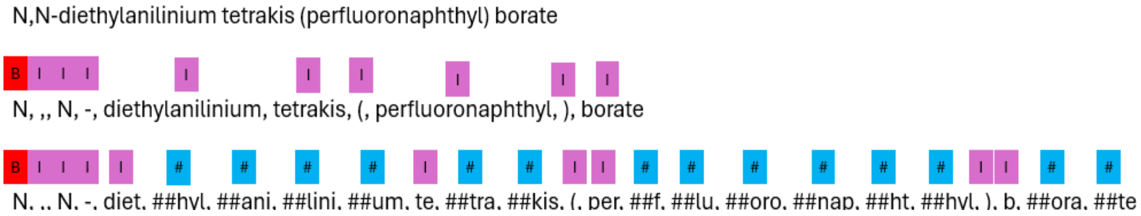

N,N-diethylanilinium tetrakis (perfluoronaphthyl) borate
B I I I I I I I I I
N, ,, N, -, diethylanilinium, tetrakis, (, perfluoronaphthyl,), borate
B I I I I # # # # I # # I I # # # # # # I I # #
N, ,, N, -, diet, ##hyl, ##ani, ##lini, ##um, te, ##tra, ##kis, (, per, ##f, ##lu, ##oro, ##nap, ##ht, ##hyl,), b, ##ora, ##te

Figure 4: Subword tokenization of a chemical entity. The first line indicates the chemical, the second shows the tokens with labels, and the third shows subword tokenization. Blue tokens were masked during loss

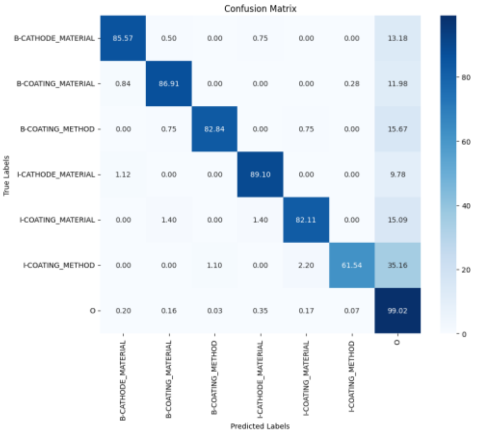


Figure 2: Confusion matrix of Battery labels for the best-performing model.

Data Subset	Label	Train	Validation	Evaluation
Battery	B-CATHODE_MATERIAL	1188	223	402
	B-COATING_MATERIAL	1304	206	359
	B-COATING_METHOD	352	57	134
	I-CATHODE_MATERIAL	1872	362	624
	I-COATING_MATERIAL	958	147	285
	I-COATING_METHOD	212	34	91
	O	98314	16791	28838
Catalyst	B-ADDITIVE	821	126	153
	B-OLEFIN	1137	150	356
	B-PRE_CATALYST	314	51	71
	B-SOLVENT	352	50	131
	B-SUPPORT	392	25	83
	I-ADDITIVE	3785	669	620
	I-OLEFIN	656	83	163
	I-PRE_CATALYST	6199	1041	1350
	I-SOLVENT	85	17	39
	I-SUPPORT	384	14	68
	O	340522	60333	100138

Table 4: Raw label counts of both chemical NER datasets.

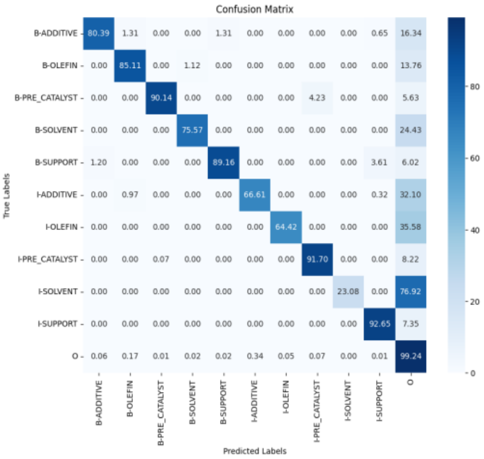


Figure 3: : Confusion matrix of Catalyst labels for the best-performing model.