Fine-tuning BERT with Kolmogorov-Arnold Network for Hierarchical Text Classification*

First Author[1][0000−1111−2222−3333], Second Author[2,3][1111−2222−3333−4444], and Third Author[3][2222−−3333−4444−5555]

[1] Princeton University, Princeton NJ 08544, USA
[2] Springer Heidelberg, Tiergartenstr. 17, 69121 Heidelberg, Germany
lncs@springer.com
http://www.springer.com/gp/computer-science/lncs
[3] ABC Institute, Rupert-Karls-University Heidelberg, Heidelberg, Germany
{abc,lncs}@uni-heidelberg.de

Abstract. Hierarchical text classification (HTC) remains a challenging task in natural language processing (NLP) due to the complexity of multi-level label structures. Traditional deep learning models, such as BERT-BiLSTM, often struggle to effectively capture hierarchical dependencies. In this paper, we propose a novel approach that fine-tunes BERT with Kolmogorov-Arnold Network (KAN) to improve HTC performance. BERT is leveraged to generate contextual text embeddings, while KAN, a neural network variant with learnable activation functions on edges instead of fixed activation functions on neurons, enhances classification by capturing hierarchical relationships more effectively. We evaluate the BERT-KAN model on a large-scale dataset of Amazon product reviews, structured into three hierarchical levels. Our results demonstrate that BERT-KAN outperforms BERT-BiLSTM across multiple evaluation metrics, particularly in Category Accuracy and Hierarchical F1 Score, confirming its superior ability to model hierarchical structures.

Keywords: Hierarchical text classification (HTC) · Deep learning · BERT · Kolmogorov-Arnold Network (KAN) · Neural networks (NN) · Text representation· Classification model.

1 Introduction

Text classification is a fundamental task in Natural Language Processing (NLP) that involves assigning predefined categories to textual data [1]. It has a wide range of applications, including document organization, sentiment analysis, and spam detection. Traditionally, text classification challenges have been addressed using machine learning methods, where models are trained on labeled datasets to learn patterns and make predictions [2]. However, the advent of deep learning has significantly improved the performance of text classification models. Deep learning, a subset of machine learning, leverages neural networks (NNs) with

* Supported by organization x.

multiple layers (hence the term "deep") to capture complex patterns in textual data [3].

The ability of deep learning models to process large-scale data and extract intricate linguistic patterns has led to the development of advanced architectures, such as Bidirectional Encoder Representations from Transformers (BERT), Multi-Layer Perceptron (MLP), Kolmogorov-Arnold Network (KAN), Long Short-Term Memory (LSTM), and Bidirectional LSTM (BiLSTM). These models have revolutionized NLP by enabling more accurate and efficient text classification.

Hierarchical Text Classification (HTC) introduces additional complexity compared to conventional (flat) text classification [4]. Unlike flat classification, where each document is assigned a single label, HTC requires assigning labels that follow a hierarchical structure. This hierarchical organization demands models that not only understand textual content but also capture the relationships between different levels of classification.

In this paper, we propose fine-tuning BERT with KAN for HTC. Fine-tuning aligns BERT with KAN to meet the specific demands of hierarchical classification. BERT, a deep learning-based model, transforms textual data into a 768-dimensional vector representation. KAN, an alternative to MLP, introduces learnable activation functions on edges (weights) and eliminates traditional linear weights. Instead, each weight is replaced by a univariate function parameterized as a spline. The proposed BERT-KAN model is evaluated against the BERT-BiLSTM model [5]. Experimental results demonstrate the superiority of the KAN-based approach in terms of Category Accuracy, Category F1 Score, Super-Category Accuracy, Super-Category F1 Score, and Hierarchical F1 Score.

The remainder of this paper is structured as follows: Section 2 reviews related work, with a focus on text classification approaches. Section 3 introduces the proposed BERT-KAN model, detailing how BERT is integrated with KAN for HTC. Section 4 presents a comparative study between BERT-KAN and BERT-BiLSTM, discussing performance metrics and experimental results. Finally, Section 5 concludes the paper by summarizing key findings and outlining future research directions in hierarchical text classification.

2 Related work

Text classification (TC) is one of the most extensively researched tasks in Natural Language Processing (NLP) [6]. TC techniques primarily involve supervised learning methods that map textual data to predefined labels. Over the years, various surveys and studies have explored a range of methodologies, from traditional machine learning approaches to contemporary deep learning models [1, 6–12].

Hierarchical Text Classification (HTC) is a specialized subset of TC, distinguished by its multi-level label structure, which introduces additional complexity in classification. Unlike standard TC, where labels exist independently, HTC requires capturing hierarchical relationships between labels to improve classification accuracy. Given the diverse range of HTC methodologies, we build upon the study by Zangari et al. [13] and provide a synthesized overview of recent approaches, consolidating them into a summarizing table (Table 1).

Table 1. Summary of recent HTC approaches.

Approach	**Description**	**References**
Sequence-to-sequence (Seq2Seq)	Treats HTC as a sequence generation task, where the input document is mapped to a sequence of hierarchical labels. Techniques such as breadth-first and depth-first ordering are applied to generate structured label sequences. Improves local label dependency modeling.	[14–21]
Graph-based methods	Utilizes graph neural networks (GNNs) and graph convolutional networks (GCNs) to represent label hierarchies. Enables hierarchy-aware and label-balanced classification. Some approaches incorporate knowledge graphs (KGs) for enhanced representation.	[18, 22–32]
Large model tuning	Techniques such as prompt-tuning and prefix-tuning are applied to large pre-trained language models, embedding classification tasks within a natural language modeling framework. Reduces fine-tuning costs while maintaining high performance.	[33–36]

In this study, we introduce the BERT-KAN model, a transformer-based deep learning approach tailored for HTC. This model leverages the representational power of Bidirectional Encoder Representations from Transformers (BERT) and the adaptability of Kolmogorov-Arnold Networks (KAN) to improve classification performance. The integration of KAN enables a more flexible and structured classification process by replacing traditional linear weights with learnable activation functions. Our proposed approach aligns closely with graph-based methodologies while leveraging transformer-based hierarchical learning for enhanced performance.

The next section provides a detailed discussion of the BERT-KAN model and its components, highlighting how it addresses key challenges in HTC.

3 BERT-KAN model

The proposed BERT-KAN model is composed of three key modules: the Data Loading module, the BERT module, and the KAN module (see Fig. 1). Each module plays a crucial role in hierarchical text classification.

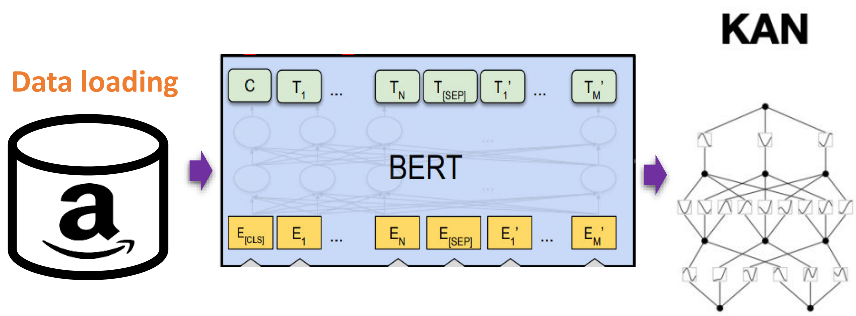

Fig. 1. BERT-KAN model architecture.

3.1 Data loading module

This module is responsible for preparing the dataset by loading data, encoding labels, and splitting the corpus into training and test sets. The dataset used in this study is derived from Amazon product reviews[4] and consists of 40000 textual entries. The hierarchical structure of this dataset is organized into three levels: 6 broad categories at the first level, 64 subcategories at the second level, and 510 fine-grained classes at the third level. The primary classes at the highest level include *grocery and gourmet food*, *beauty*, *pet supplies*, *health and personal care*, *baby products*, and *toys and games*.

To align with our model's input requirements, the dataset structure was adapted from its original format (*Text* / *Category_level1* / *Category_level2* / *Category_level3*) to a more suitable form: (*Text* / *Category* / *Super_category*). This transformation ensures efficient label encoding and hierarchical classification.

Determining the optimal train-test split is a well-known challenge in text classification tasks. Bichri et al. [37] demonstrated that using at least 70% of the dataset for training leads to superior performance. Vrigazova [38] reached a similar conclusion after multiple experiments, recommending a 70/30 train-test split for classification problems. Following these findings, our dataset was randomly split 70% for training and 30% for testing, ensuring a balanced distribution of samples across hierarchical levels.

[4] https://www.kaggle.com/datasets/kashnitsky/hierarchical-text-classification

3.2 BERT module

This module is responsible for generating contextual embeddings for input text using BERT (Bidirectional Encoder Representations from Transformers). BERT, introduced by Devlin et al. [39], is a transformer-based deep learning model designed to produce high-quality contextualized word representations. Unlike traditional word embeddings, BERT considers both preceding and succeeding words in a sentence, making it bidirectional and significantly improving its language understanding capabilities.

Before text can be processed by BERT, it undergoes the following preprocessing steps:

1. *Tokenization:* The input text is converted into subword tokens using WordPiece tokenization.
2. *Special token insertion:* Each input sequence is prefixed with a [CLS] token and suffixed with a [SEP] token.
3. *Padding and truncation:* Sequences are adjusted to a fixed length by padding shorter sequences and truncating longer ones.
4. *Attention masking:* Tokens are masked appropriately to distinguish real words from padding tokens during processing.

Once preprocessed, the tokenized text is passed through BERT's transformer layers, generating contextual embeddings that capture syntactic and semantic relationships within the text.

3.3 KAN module

The Kolmogorov-Arnold Network (KAN) module transforms the BERT-generated embeddings into a hierarchical representation, enhancing classification performance. KAN, introduced by Liu et al. [40], is a novel neural network architecture inspired by the Kolmogorov-Arnold representation theorem.

Unlike traditional Multi-Layer Perceptrons (MLPs), which rely on fixed activation functions at the neuron level, KAN represents a fundamental shift:

– *Learnable activation functions:* Instead of fixed activations, KANs assign learnable univariate functions to edges (weights), parameterized as splines.
– *No linear weights:* Unlike MLPs, KANs entirely replace linear weights with learnable functions, improving expressiveness.
– *Efficient representation:* KANs require fewer parameters than MLPs while maintaining or surpassing accuracy, making them more computationally efficient.
– *Interpretability:* The function-based transformations in KANs can be visualized and understood intuitively, enabling better interpretability and human interaction.

In the BERT-KAN model, KAN layers are employed to map BERT embeddings to hierarchical label representations, effectively capturing:

– Interdependencies between hierarchical classes (e.g., level 1 influences level 2, which in turn affects level 3 predictions).
– Non-linear relationships in text features for improved classification accuracy.
– Flexible transformations that adapt dynamically to variations in input text structure.

By integrating KAN with BERT, our model achieves an optimal balance between contextual representation learning (via BERT) and hierarchical label modeling (via KAN), resulting in improved performance for hierarchical text classification.

4 Comparative study

This section presents a comparative analysis of the proposed BERT-KAN model against an existing deep learning approach, BERT-BiLSTM, introduced by Hamzaoui et al. [5]. BiLSTM, or Bidirectional Long Short-Term Memory, extends the traditional LSTM model to address the limitations of standard recurrent neural networks (RNNs) in capturing long-term dependencies. While LSTM networks regulate the flow of information through specialized gating mechanisms to retain or discard information as needed [41], BiLSTM enhances this mechanism by processing input sequences in both forward and backward directions. This bidirectional structure ensures that contextual dependencies from both ends of a sequence are considered, leading to a more comprehensive understanding of textual data [42].

4.1 BERT-BiLSTM model

The BERT-BiLSTM model architecture (see Fig. 2) consists of several key components:

– *Input layer:* The dataset is loaded and divided into two subsets, with 70% used for training and 30% for testing.
– *BERT embedding layer:* A pre-trained BERT model generates contextual embeddings for input text sequences. These embeddings capture complex linguistic patterns, and provide dense vector representations influenced by the entire input sequence.
– *Bidirectional LSTM layer:* Two LSTM layers process BERT embeddings. One layer analyzes the text in a forward direction. The other layer processes it in reverse. This bidirectional approach enhances the model's ability to detect dependencies within the hierarchical structure.
– *Dense layer:* A fully connected layer applies a softmax activation function to classify text at multiple hierarchical levels, producing a probability distribution over predefined categories and super-categories.

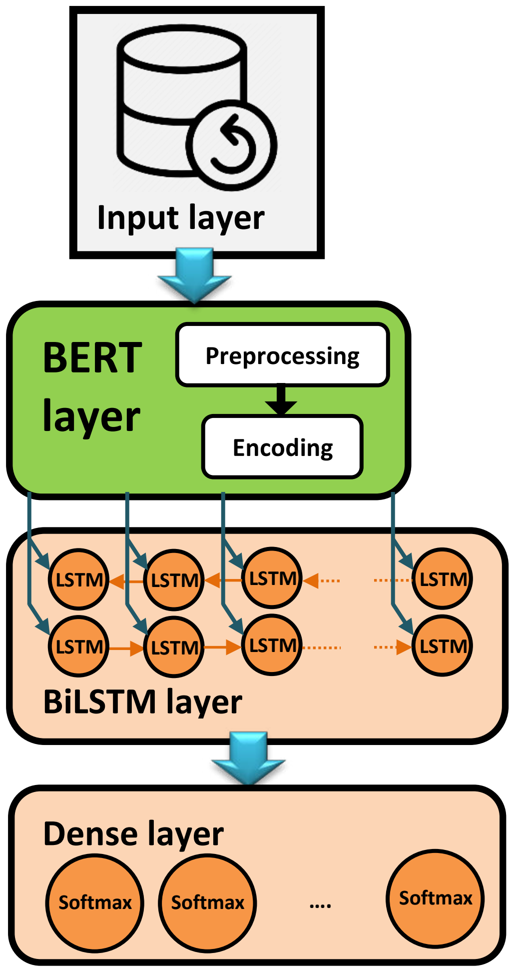

Fig. 2. BERT-BiLSTM system architecture [5].

4.2 Evaluation metrics

To evaluate our model performance, the following metrics are used:

- *Category accuracy:* Measures the proportion of correct predictions at the category level [43].

$$Category\ Accuracy = \frac{Number\ of\ Correct\ Category\ Predictions}{Total\ Number\ of\ Predictions} \quad (1)$$

- *Category F1 Score:* Evaluates classification performance by balancing precision and recall [44].
  - *Precision:* The ratio of correctly predicted positive instances to all predicted positives.

$$Precision = \frac{True\ Positives}{True\ Positives + False\ Positives} \quad (2)$$

  - *Recall:* The ratio of correctly predicted positive instances to all actual positives.

$$Recall = \frac{True\ Positives}{True\ Positives + False\ Negatives} \quad (3)$$

  - *F1 Score:* The harmonic mean of precision and recall.

$$Category\ F1\ Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (4)$$

- *Super-category accuracy (S_ C accuracy):* Assesses the correctness of predictions for higher-level super-categories [43].

$$S\_C\ Accuracy = \frac{Number\ of\ Correct\ Super\_Category\ Predictions}{Total\ Number\ of\ Predictions} \quad (5)$$

- *Super-Category F1 Score:* Similar to the category-level F1 score but applied at the super-category level [44].

$$Precision = \frac{True\ Positives}{True\ Positives + False\ Positives} \quad (6)$$

$$Recall = \frac{True\ Positives}{True\ Positives + False\ Negatives} \quad (7)$$

$$Super\_Category\ F1\ Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (8)$$

– *Hierarchical F1 Score (HF1 Score):* Accounts for classification correctness across both category and super-category levels [45]. It considers true positives, false positives, and false negatives in the hierarchical structure. It also ensures that errors are evaluated in relation to the hierarchical nature of the dataset.
Let TP_H be the number of true positives considering the hierarchy, FP_H be the false positives, and FN_H be the false negatives.

$$Hierarchical\ Precision = \frac{TP_H}{TP_H + FP_H} \tag{9}$$

$$Hierarchical\ Recall = \frac{TP_H}{TP_H + FN_H} \tag{10}$$

$$HF1\ Score = 2 \times \frac{Hierarchical\ Precision \times Hierarchical\ Recall}{Hierarchical\ Precision + Hierarchical\ Recall} \tag{11}$$

4.3 Results and discussion

For the comparative study, both BERT-KAN and BERT-BiLSTM models were implemented and made publicly available on GitHub[5] to support further research in AI and NLP. The evaluation was conducted on a large hierarchical dataset structured into three levels, utilizing a high-performance GPU-powered system. The experimental setup included an MSI GeForce RTX 3060 GAMING 12G graphics card with a boosted frequency of 1777 MHz, 12 GB of GDDR6 memory, and 3584 CUDA cores, ensuring efficient processing of deep learning computations.

The results of the classification tasks are presented in Fig. 3. The BERT-KAN model demonstrated superior performance across all evaluation metrics compared to BERT-BiLSTM. The category accuracy for BERT-KAN reached 50.95%, surpassing the 44.89% achieved by BERT-BiLSTM. Similarly, super-category accuracy was higher for BERT-KAN at 64.33%, compared to 62.65% for BERT-BiLSTM.

F1 scores further reinforced this advantage. The hierarchical F1 score, a crucial metric for hierarchical classification, showed a notable difference, with BERT-KAN achieving 0.2972, significantly outperforming BERT-BiLSTM's score of 0.2411. These findings suggest that KAN's capabilities contribute to a more effective capture of hierarchical relationships within the dataset.

Despite BERT-BiLSTM's strong performance as a baseline, its reliance on sequential processing within the LSTM network may limit its ability to model complex hierarchical dependencies as effectively as KAN. The results indicate that KAN's learnable activation functions and data transformation mechanisms provide a more robust classification approach.

[5] https://github.com/khouloud-1/BERT-KAN

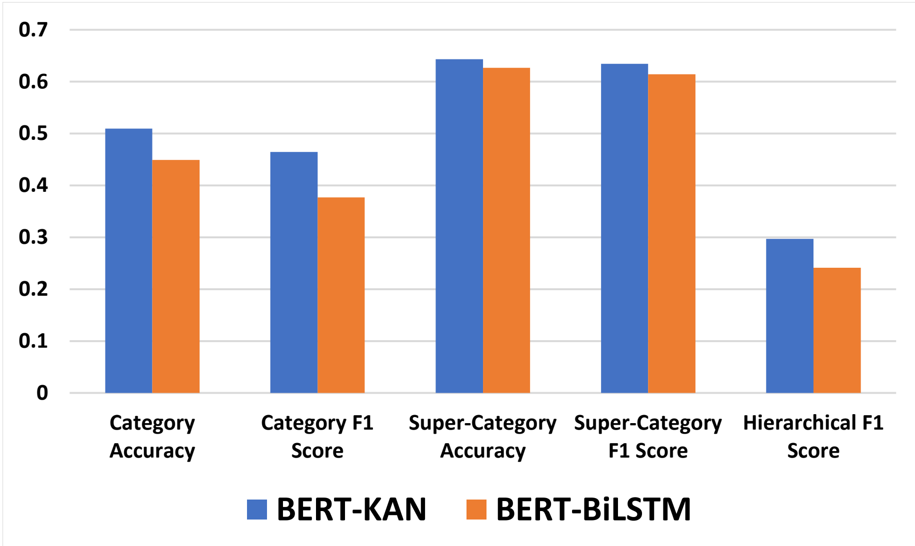

Fig. 3. Evaluation results of HTC with BERT-KAN and BERT-BiLSTM models.

5 Conclusions and perspectives

This study explored the integration of Kolmogorov-Arnold Network (KAN) with BERT to enhance hierarchical text classification (HTC). BERT was utilized to generate contextual embeddings, providing numerical representations of input texts, while KAN served as the classification model, leveraging its capacity for function approximation and hierarchical pattern recognition.

Through comparative experiments on a large dataset of Amazon product reviews, our findings indicate that BERT-KAN consistently outperforms BERT-BiLSTM, particularly in Category Accuracy and Hierarchical F1 Score. These results emphasize the potential of KAN in effectively capturing hierarchical structures in text data, making it a promising alternative for HTC tasks requiring contextual understanding at multiple levels.

To further enhance the BERT-KAN model, several avenues for improvement can be explored:

- *Hyperparameter optimization:* Fine-tuning key parameters, such as the learning rate, batch size, and activation functions, may yield improved performance.
- *Architectural refinements:* Investigating different KAN configurations or hybridizing it with other deep learning techniques could further enhance hierarchical classification.
- *Dataset expansion:* Incorporating more diverse texts and labels or extending the dataset to include additional hierarchical levels would provide a more rigorous evaluation of the model's robustness.

– *Cross-domain and multilingual applications:* Applying BERT-KAN to different languages and domains could broaden its applicability and contribute to the advancement of hierarchical text classification research.

By addressing these challenges and exploring these opportunities, future research can further solidify the role of BERT-KAN as a powerful model for hierarchical text classification in varied and complex linguistic settings.

References

1. Kowsari, K., Jafari Meimandi, K., Heidarysafa, M., Mendu, S., Barnes, L., Brown, D.: Text Classification Algorithms: A Survey. Information 10, 150 (2019)
2. Palanivinayagam, A., El-Bayeh, C.Z., Damaševičius, R.: Twenty Years of Machine-Learning-Based Text Classification: A Systematic Review. Algorithms 16, 236 (2023)
3. Patel, J.M.: Deep Learning: Concepts, Architectures, Workflow, Applications and Future Directions. International Journal For Multidisciplinary Research 5, (2023)
4. Zangari, A., Marcuzzo, M., Schiavinato, M., Rizzo, M., Gasparetto, A., Albarelli, A.: Hierarchical Text Classification: a review of current research. EXPERT SYSTEMS WITH APPLICATIONS 224, (2023)
5. Hamzaoui, B., Bouchiha, D., Bouziane, A., Doumi, N.: BERT-BiLSTM model for hierarchical Arabic text classification. Indonesian Journal of Electrical Engineering and Informatics (IJEEI) 12, 690-699 (2024)
6. Gasparetto, A., Marcuzzo, M., Zangari, A., Albarelli, A.: A Survey on Text Classification Algorithms: From Text to Predictions. Information 13, 83 (2022)
7. Li, Q., Peng, H., Li, J., Xia, C., Yang, R., Sun, L., Yu, P.S., He, L.: A Survey on Text Classification: From Traditional to Deep Learning. ACM Trans. Intell. Syst. Technol. 13, Article 31 (2022)
8. Minaee, S., Kalchbrenner, N., Cambria, E., Nikzad, N., Chenaghlu, M., Gao, J.: Deep Learning–based Text Classification: A Comprehensive Review. ACM Comput. Surv. 54, Article 62 (2021)
9. Gasparetto, A., Zangari, A., Marcuzzo, M., Albarelli, A.: A survey on text classification: Practical perspectives on the Italian language. PLOS ONE 17, e0270904 (2022)
10. Fields, J., Chovanec, K., Madiraju, P.: A Survey of Text Classification With Transformers: How Wide? How Large? How Long? How Accurate? How Expensive? How Safe? IEEE Access 12, 6518-6531 (2024)
11. Bhawsar, S., Dubey, S., Kushwaha, S., Sharma, S.: Text Classification Using Deep Learning: A Survey. In: Proceedings of International Conference on Computational Intelligence, pp. 205-216. Springer Nature Singapore, (Year)
12. Hamzaoui, B., Bouchiha, D., Bouziane, A.: A comprehensive survey on arabic text classification: progress, challenges, and techniques. Brazilian Journal of Technology 8, e77611 (2025)
13. Zangari, A., Marcuzzo, M., Rizzo, M., Giudice, L., Albarelli, A., Gasparetto, A.: Hierarchical Text Classification and Its Foundations: A Review of Current Research. Electronics 13, 1199 (2024)
14. Yang, Z., Liu, G.: Hierarchical Sequence-to-Sequence Model for Multi-Label Text Classification. IEEE Access 7, 153012-153020 (2019)
15. Zhao, W., Gao, H., Chen, S., Wang, N.: Generative Multi-Task Learning for Text Classification. IEEE Access 8, 86380-86387 (2020)

16. Rivas Rojas, K., Bustamante, G., Oncevay, A., Sobrevilla Cabezudo, M.A.: Efficient Strategies for Hierarchical Text Classification: External Knowledge and Auxiliary Tasks. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pp. 2252-2257. Association for Computational Linguistics, (Year)
17. Risch, J., Garda, S., Krestel, R.: Hierarchical Document Classification as a Sequence Generation Task. Proceedings of the ACM/IEEE Joint Conference on Digital Libraries in 2020, pp. 147–155. Association for Computing Machinery, Virtual Event, China (2020)
18. Zhang, J., Li, Y., Shen, F., Xia, C., Tan, H., He, Y.: Hierarchy-Aware and Label Balanced Model for Hierarchical Text Classification. Knowledge-Based Systems 300, 112153 (2024)
19. Kwon, J., Kamigaito, H., Song, Y.-I., Okumura, M.: Hierarchical label generation for text classification. In: Proceedings of the Findings of the Association for Computational Linguistics: EACL 2023, pp. 625-632. (Year)
20. Torba, F., Gravier, C., Laclau, C., Kammoun, A., Subercaze, J.: A Study on Hierarchical Text Classification as a Seq2seq Task. In: Advances in Information Retrieval. ECIR 2024., pp. 287-296. Springer Nature Switzerland, (Year)
21. Zhang, J., Li, Y., Shen, F., He, Y., Tan, H., He, Y.: Hierarchical text classification with multi-label contrastive learning and KNN. Neurocomputing 577, 127323 (2024)
22. Zhou, J., Ma, C., Long, D., Xu, G., Ding, N., Zhang, H., Xie, P., Liu, G.: Hierarchy-Aware Global Model for Hierarchical Text Classification. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pp. 1106-1117. Association for Computational Linguistics, (Year)
23. Liu, H., Huang, X., Liu, X.: Improve label embedding quality through global sensitive GAT for hierarchical text classification. EXPERT SYSTEMS WITH APPLICATIONS 238, 122267 (2024)
24. Chen, H., Ma, Q., Lin, Z., Yan, J.: Hierarchy-aware Label Semantics Matching Network for Hierarchical Text Classification. In: Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, pp. 4370–4379. Association for Computational Linguistics: Kerrville, TX, USA, (Year)
25. Pal, A., Selvakumar, M., Sankarasubbu, M.: MAGNET: Multi-Label Text Classification using Attention-based Graph Neural Network. In: Proceedings of the 12th International Conference on Agents and Artificial Intelligence (ICAART 2020), pp. 494–505. SciTePress: Setubal, Portugal, (Year)
26. Zhao, R., Wei, X., Ding, C., Chen, Y.: Hierarchical Multi-label Text Classification: Self-adaption Semantic Awareness Network Integrating Text Topic and Label Level Information. In: Proceedings of the Knowledge Science, Engineering and Management, pp. 406-418. Springer International Publishing, (Year)
27. Chen, J., Zhao, S., Lu, F., Liu, F., Zhang, Y.: Research on patent classification based on hierarchical label semantics. In: 2022 3rd International Conference on Education, Knowledge and Information Management (ICEKIM), pp. 1025-1032. (Year)
28. Yao, Z., Chai, H., Cui, J., Tang, S., Liao, Q.: HITSZQ at SemEval-2023 Task 10: Category-aware Sexism Detection Model with Self-training Strategy. In: Proceedings of the 17th International Workshop on Semantic Evaluation (SemEval-2023), pp. 934–940. (Year)
29. Ning, B., Zhao, D., Zhang, X., Wang, C., Song, S.: UMP-MG: A Uni-directed Message-Passing Multi-label Generation Model for Hierarchical Text Classification. Data Science and Engineering 8, 112-123 (2023)

30. Song, J., Wang, F., Yang, Y.: Peer-Label Assisted Hierarchical Text Classification. In: Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics, pp. 3747-3758. Association for Computational Linguistics, (Year)
31. Wang, Z., Wang, P., Huang, L., Sun, X., Wang, H.: Incorporating Hierarchy into Text Encoder: a Contrastive Learning Approach for Hierarchical Text Classification. In: Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics, pp. 7109–7119. (Year)
32. Liu, Y., Zhang, K., Huang, Z., Wang, K., Zhang, Y., Liu, Q., Chen, E.: Enhancing Hierarchical Text Classification through Knowledge Graph Integration. In: Proceedings of the Findings of the Association for Computational Linguistics: ACL 2023, pp. 5797–5810. Association for Computational Linguistics, (Year)
33. Wang, Z., Wang, P., Liu, T., Cao, Y., Sui, Z., Wang, H.: HPT: Hierarchy-aware Prompt Tuning for Hierarchical Text Classification. In: Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, pp. 3740–3751. (Year)
34. Ji, K., Lian, Y., Gao, J., Wang, B.: Hierarchical Verbalizer for Few-Shot Hierarchical Text Classification. In: Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics, pp. 2918–2933. (Year)
35. Zhang, Y., Yang, R., Xu, X., Li, R., Xiao, J., Shen, J., Han, J.: TELEClass: Taxonomy Enrichment and LLM-Enhanced Hierarchical Text Classification with Minimal Supervision. arXiv abs/2403.00165, (2024)
36. Chen, L., Chou, H., Zhu, X.: Developing Prefix-Tuning Models for Hierarchical Text Classification. In: Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing: Industry Track, pp. 390-397. Association for Computational Linguistics, (Year)
37. Bichri, H., Chergui, A., Hain, M.: Investigating the Impact of Train/Test Split Ratio on the Performance of Pre-Trained Models with Custom Datasets. International Journal of Advanced Computer Science & Applications 15, (2024)
38. Vrigazova, B.: The Proportion for Splitting Data into Training and Test Set for the Bootstrap in Classification Problems. Business Systems Research Journal 12, 228-242 (2021)
39. Devlin, J., Chang, M.-W., Lee, K., Toutanova, K.: BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In: the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pp. 4171-4186. Association for Computational Linguistics, (Year)
40. Liu, Z., Wang, Y., Vaidya, S., Ruehle, F., Halverson, J., Soljačić, M., Hou, T.Y., Tegmark, M.: Kan: Kolmogorov-arnold networks. arXiv preprint arXiv:2404.19756 (2024)
41. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural computation 9, 1735-1780 (1997)
42. Schuster, M., Paliwal, K.K.: Bidirectional recurrent neural networks. IEEE transactions on Signal Processing 45, 2673-2681 (1997)
43. Powers, D.M.W.: Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation. International Journal of Machine Learning Technology 2, 37-63 (2011)
44. Christen, P., Hand, D.J., Kirielle, N.: A Review of the F-Measure: Its History, Properties, Criticism, and Alternatives. ACM Comput. Surv. 56, Article 73 (2023)
45. Kosmopoulos, A., Partalas, I., Gaussier, E., Paliouras, G., Androutsopoulos, I.: Evaluation measures for hierarchical classification: a unified view and novel approaches. Data Mining and Knowledge Discovery 29, 820-865 (2015)