Syntactical Graph Neural Network for Authorship Attribution

Jingze Zhang, Gordon Erlebacher
Department of Scientific Computing, Florida State University

Abstract

Authorship Attribution is the process of identifying the author of an anonymous document. It involves using statistical and computational methods to analyze extensive collections of text. Traditional studies focus on finding useful patterns in linguistic and stylistic features to tell them apart. They also rely extensively on feature engineering. Many studies also use neural networks, which led to some improvements. However, they only consider local syntactic features and fail to deal with long-range dependencies. Our research presents a novel architecture based on word connections within sentences via dependency trees and connections between sentences via the self-attention mechanism. The sentence's sequential order and syntactical structure enable our model to outperform the state-of-the-art. We also conduct a comprehensive ablation study to analyze the effect of different linguistic components, including word order, word length, word frequency, dependency type, and sentence order.

Methods

Pre-trained language models

Language models have general knowledge about the language and are considered the fundamental basis of NLP neural network approaches. As language models grow in complexity, their parameter count increases, reaching upwards of hundreds of billions of parameters. Training from scratch is impractical for most researchers. Substantial work has shown that pre-trained models on large corpora can learn general language features beneficial for downstream tasks. Finetuning pretrained models on downstream tasks is now the mainstream approach in NLP. In this work, we initialize our model with pretrained Bert[1], providing a warm starting point for faster convergence.

Pretraining POS language model

Inheriting the idea of language modeling, a POS language model is a probability distribution over POS sequences that posses a general understanding of the syntactic properties of the sequence. Before pretraining, a POS tagger transforms text into sequences of POS tokens. During the training process, as directed by previous empirical analysis, the data loader dynamically and randomly masks 15% of the tokens, and the pretraining objective is to predict the masked token. The pretrained model is suitable for any downstream tasks involving a deep syntax understanding.

Feature representation learning

Besides POS tokens, there are other content-free linguistic components that reflect a person’s writing style. Therefore, our model includes the following components as additional linguistic features:

- A syntax tree is a tree representation of the hierarchical relationship between words in a sentence. Figure 1 shows an example of a dependency tree, one of the commonly used syntax trees.
- Word length is a content-free superficial feature that reflects the author’s choice of words.
- Word difficulty reflects the vocabulary richness and education level. In this work, we use the word frequency as a proxy feature. Specifically, for each word, we take the base-10 logarithm of the number of appearances per billion words and transform it as an integer.

We attach an edge list to each sentence for the dependency structure. In addition, we adopt an embedding layer for the dependency type on edge, the word length, and the word frequency to let the neural network learn the representation.

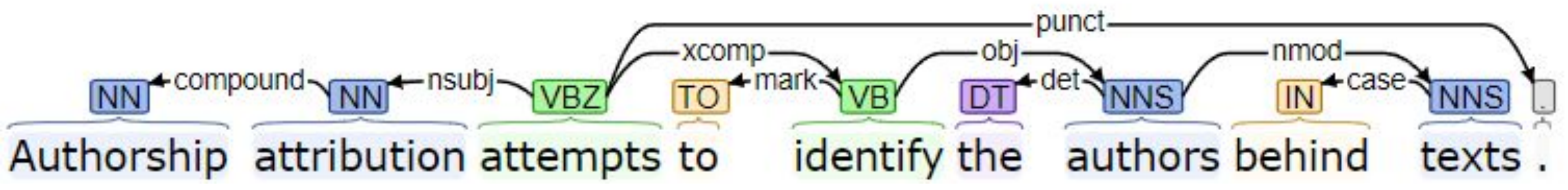

Figure 1 An example of dependency tree generated by Stanford Corenlp pipeline[2]. The tree consists of directed edges with the dependency relation on the edge.

Intra-sentence graph neural network

A graph neural network is a machine learning framework that works on graph-structured data. The core idea addresses message-passing between nodes in a graph, which empowers the network to transductive-ly propagate information to nearby nodes and capture the overall structure of the graph. In our case, we let the pretrained POS embedding exchange information along the dependency structure. Finally, we read out a global graph embedding with mean/max pooling to encode the syntactic information of a sentence.

Inter-sentence self-attention graph

Sentences in a document are more than just placed sequentially. There are logical and conceptual relations in semantics. However, the relations are only clear with expert analysis. Therefore, we adopt the self-attention mechanism on a fully connected graph and let the network figure out the relation.

Experiments

Data

The CCAT50 dataset is a collection of news articles widely used as a benchmark dataset for authorship attribution[4]. The 50 refers to the number of authors. Each author has 1,000 articles, evenly divided into train and test sets.

Model

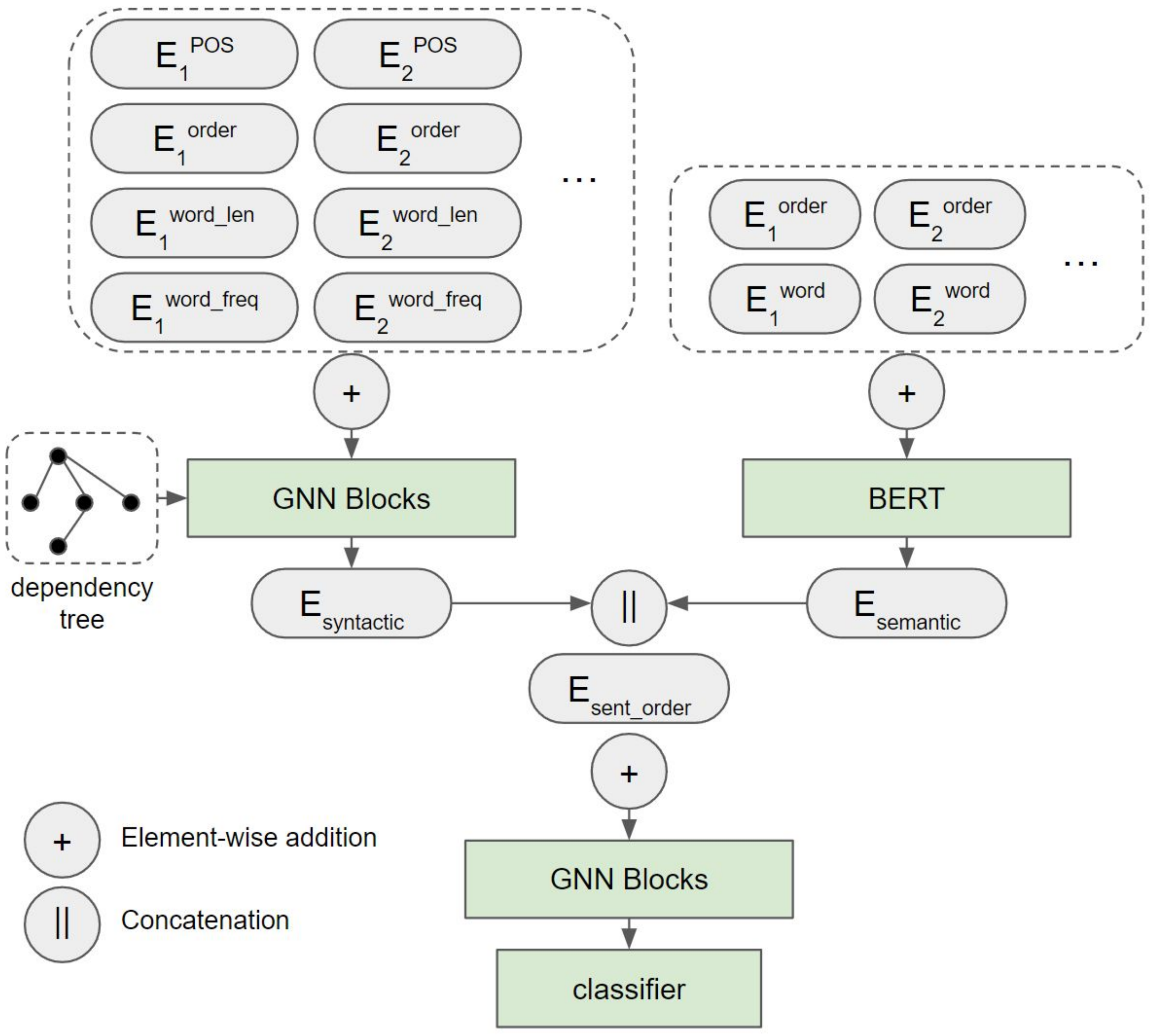

Figure 2 Architecture of the proposed syntactical graph neural network. The syntactic embedding from the intra-sentence GNN and the semantic embedding from the pretrained Bert are concatenated as the input for the inter-sentence self-attention GNN.

Results

Model	Accuracy
Syntax-CNN[3]	81.00
Style-HAN[4]	82.35
MCSAN[5]	83.42
Our model	**84.08**

Model	Accuracy
Our model	**84.08**
w/o inter-GNN	83.4
semantic	82.44
syntactic	71.28
w/o intra-GNN	45.88

Discussion and future work

Our proposed model outperforms the previous state-of-the-art on CCAT50. The ablation study shows the importance of each component. Experiments on other datasets are in progress. Other syntactic information, such as the constituency tree, might be helpful.

References

[1] Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805.

[2] Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J. R., Bethard, S., & McClosky, D. (2014, June). The Stanford CoreNLP natural language processing toolkit. In Proceedings of 52nd annual meeting of the association for computational linguistics: system demonstrations (pp. 55-60).

[3] Zhang, R., Hu, Z., Guo, H., & Mao, Y. (2018). Syntax encoding with application in authorship attribution. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (pp. 2742-2753).

[4] Jafariakinabad, F., & Hua, K. A. (2021). Unifying Lexical, Syntactic, and Structural Representations of Written Language for Authorship Attribution. SN Computer Science, 2(6), 481.

[5] Wu, H., Zhang, Z., & Wu, Q. (2021). Exploring syntactic and semantic features for authorship attribution. Applied Soft Computing, 111, 107815.