Named Entity Recognition

Using BiLSTM-CNN with Word and Character Embeddings

Abhijeet Vaibhav
Department of Electrical Engineering, Indian Institute of Technology Bombay, Mumbai, India
Email: 21D070004@iitb.ac.in

Abstract *- **Named Entity Recognition (NER) is a critical task in Natural Language Processing (NLP) for identifying and classifying named entities such as persons, locations, organizations, and other domain-specific entities in text. This paper presents a hybrid deep learning approach combining Bidirectional Long Short-Term Memory (BiLSTM) networks with Convolutional Neural Network (CNN)-based character embeddings to capture both word-level context and subword information. The proposed model is trained on the CoNLL-2003 English dataset and demonstrates effective performance in recognizing named entities in unseen text.***

Keywords—Named Entity Recognition, BiLSTM, CNN, Word Embeddings, Character Embeddings, NLP.

I. Introduction

Named Entity Recognition (NER) is a foundational task in NLP, with applications in information extraction, question answering, and text summarization. Traditional machine learning approaches require extensive feature engineering and often fail to generalize to unseen or rare words. Deep learning models, particularly those leveraging both word-level and character-level embeddings, can automatically learn hierarchical features from raw text.

This paper proposes a **BiLSTM-CNN model** that processes input at both word and character levels, enabling robust recognition of named entities including out-of-vocabulary words and morphological variations. The model is evaluated using the CoNLL-2003 English NER dataset.

II. Dataset

The model is trained and evaluated on an extended version of the **CoNLL-2003 English dataset**. In addition to the original training file (eng.train), additional text files were added to increase the training corpus. The dataset now includes:

- **Training sentences:** 15,020 (original + additional files)
- **Test sentences:** 7,150 (eng.testa + eng.testb)

Each sentence contains one word per line, with the last column representing the NER tag. Common entity tags include B-PER (beginning of a person name), I-LOC (inside a location), B-ORG (organization), and O (non-entity).

Input shapes for the model:

DATA	***SHAPE***
X_TRAIN	(15020, 113)
Y_TRAIN	(15020, 113, 10)
X_TRAIN_CHAR	(15020, 113, 12)
X_TEST_CHAR	(7150, 113, 12)

Note: The increase in training data helps the model generalize better, especially for rare or unseen words in the test set.

III. Methodology

A. Preprocessing

1. *Tokenization:* Sentences are split into words; each word is split into characters for character-level embeddings.
2. *Vocabulary construction:*
    - Word vocabulary with special tokens PAD and UNK.
    - Character vocabulary with PAD and UNK.
3. *Sequence padding:*
    - Sentences are padded to a fixed maximum length.
    - Words are padded to a fixed maximum character length.
4. *Tag encoding:* Tags are converted to indices and one-hot encoded for softmax classification.

B. Model Architecture

The proposed model consists of:

1. *Word Embeddings:* 80-dimensional embeddings for each word.
2. *Character Embeddings:* 16-dimensional embeddings processed through 1D Convolution and GlobalMaxPooling to capture morphological features.
3. *Concatenation:* Word embeddings and character embeddings are concatenated per word.
4. *BiLSTM Layer:* 48 units in a bidirectional LSTM to capture context from both directions.
5. *TimeDistributed Dense Layer:* Softmax output for NER tag prediction per word.

Model summary highlights:

LAYER	OUTPUT SHAPE	PARAMS
WORD EMBEDDING	(None, 113, 80)	1,892,960
CHAR EMBEDDING + CONV1D + POOL	(None, 113, 20)	2,372
CONCATENATED	(None, 113, 100)	0
BILSTM	(None, 113, 96)	57,216
TIMEDISTRIBUTED DENSE	(None, 113, 10)	970

Hyperparameters:

- Optimizer: Adam
- Loss: Categorical Crossentropy
- Batch size: 32
- Epochs: 10

IV. Results

Sample Predictions:

Example 1:

Sentence: I eat apple every day.

Predictions: [('I', 'O'), ('eat', 'O'), ('apple', 'O'), ('every', 'O'), ('day', 'O')]

Example 2:

Sentence: I live in Mumbai.

Predictions: [('I', 'O'), ('live', 'O'), ('in', 'O'), ('Mumbai', 'B-LOC')]

Observations:

- Character embeddings improve recognition of rare and unseen words.
- BiLSTM effectively captures sequential context.
- The model performs well on CoNLL-2003 standard entity tags.

V. Conclusion

The BiLSTM-CNN model combining word and character embeddings achieves strong performance for NER tasks, especially in handling unseen words and capturing morphological features. Future work could integrate pre-trained embeddings (GloVe, BERT) and a CRF layer for improved structured prediction.

References

[1] T. CoNLL-2003, "Shared Task on Language-Independent Named Entity Recognition," 2003. [Online]. Available: https://www.clips.uantwerpen.be/conll2003/ner/