Report

Task Descrpition

Given a sentence, our objective is to identify all the named entities present in that sentence and predict their correct type (tag) from a predefined tag set.

Model Description

Model Type

- The model used is Bidirectional LSTM
- The model is called Named Entity Recognition

Hyperparameters

- Batch Size : Number of samples processed before the model is updated. Value = 30
- Number of Epochs : Training the neural network with all the training data for one cycle. Value = 10
- Hidden Dimensions : The number of Hidden dimensions in the model. Value = 400
- Embedding Dimensions : Low-dimensional, learned continuous vector representations of tokens. Value = 200
- Number of Layers : The number of LSTM units for each RNN unit. Value = 1
- Sequence Length: The length of each sequence. If the length of the sequence is more than this value, then it is chopped off else it is given required padding. Value = 25
- Patience: Number of epochs with no improvement after which training will be stopped. Value = 2

Experiment Setup

Dataset Description

MultiCoNER 1 is a large multilingual dataset (11 languages) for Named Entity Recognition. It is designed to represent some of the contemporary challenges in NER, including low-context scenarios (short and uncased text), syntactically complex entities such as movie titles, and long-tail entity distributions. MultiCoNER 2 is a large multilingual dataset (12 languages) for fine grained Named Entity Recognition. Its fine-grained taxonomy contains 36 NE classes, representing real-world challenges for NER, where named entities, apart from the surface form, context represents a critical role in distinguishing between the different fine-grained types (e.g. Scientist vs. Athlete). Furthermore, the test data of MultiCoNER 2 contains noisy instances, where the noise has been applied to both context tokens as well as the entity tokens. The noise includes typing errors at character level based on

keyboard layouts in the the different languages. Here we are applying our model only on 3 languages English, Hindi and Bangla

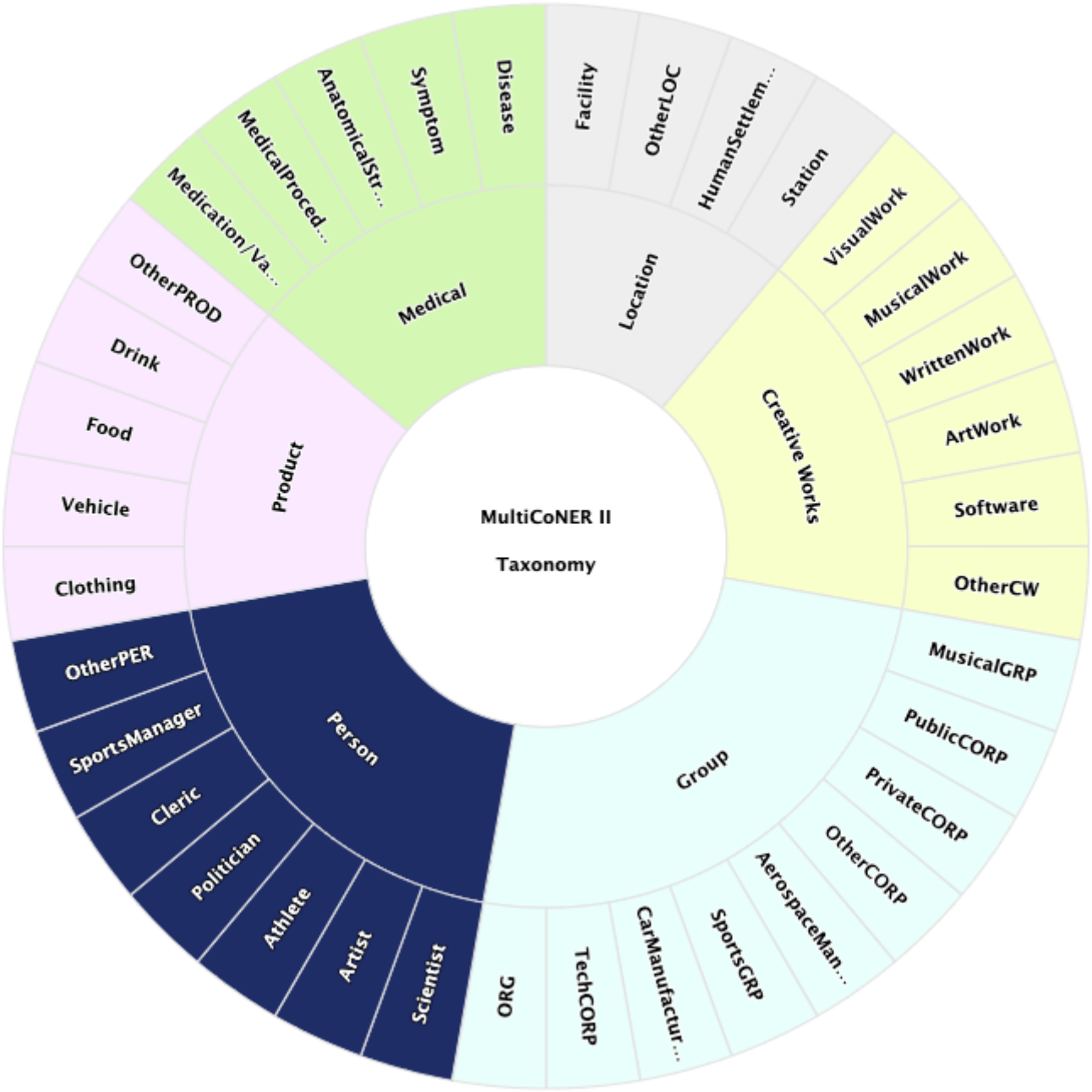
Tagset

The tagset of MultiCoNER is a fine-grained tagset.

The fine to coarse level mapping of the tags are as follows:

Location (LOC) : Facility, OtherLOC, HumanSettlement, Station
Creative Work (CW) : VisualWork, MusicalWork, WrittenWork, ArtWork, Software
Group (GRP) : MusicalGRP, PublicCORP, PrivateCORP, AerospaceManufacturer, SportsGRP, CarManufacturer, ORG
Person (PER) : Scientist, Artist, Athlete, Politician, Cleric, SportsManager, OtherPER
Product (PROD) : Clothing, Vehicle, Food, Drink, OtherPROD
Medical (MED) : Medication/Vaccine, MedicalProcedure, AnatomicalStructure, Symptom, Disease

The following figure shows the fine-grained taxonomy of the dataset.



Statistics

English

- Training Set : 16,778
- Validation Set : 871
- Test Set : 249, 890

Hindi

- Training Set : 9,632
- Validation Set : 514
- Test Set : 18,399

Bangla

- Training Set : 9,708
- Validation Set : 507
- Test Set : 19,859

Results

Performance

Model for English

Fine Grained

• accuracy: f1 score → 95%
• macro avg: precision: → 70%, recall → 60%, f1 score → 62%
• weighted avg: precision: → 95%, recall → 95%, f1 score → 95%

Coarse Grained

• accuracy: f1 score → 97%
• macro avg: precision: → 73%, recall → 64%, f1 score → 66%
• weighted avg: precision: → 96%, recall → 96%, f1 score → 97%

Model for Hindi

Fine Grained

• accuracy: f1 score → 92%
• macro avg: precision: → 52%, recall → 35%, f1 score → 40%
• weighted avg: precision: → 90%, recall → 92%, f1 score → 91%

Coarse Grained

• accuracy: f1 score → 90%
• macro avg: precision: → 69%, recall → 53%, f1 score → 59%
• weighted avg: precision: → 90%, recall → 93%, f1 score → 93%

Model for Bangla

Fine Grained

• accuracy: f1 score → 95%
• macro avg: precision: → 64% recall → 56% f1 score → 59%
• weighted avg: precision: → 94%, recall → 95% f1 score → 94%

Coarse Grained

• accuracy: f1 score → 90%
• macro avg: precision: → 39%, recall → 36%, f1 score → 34%
• weighted avg: precision: → 91%, recall → 89%, f1 score → 88%

Analysis

- Increasing the value of Embedding Dimension by keeping others constant increased the Macro Average.
- Increasing the value of Hidden Dimension by keeping others constant increased the Macro Average
- Increasing the Batch Size increased the Macro Average upto a limit then it started decreasing it.
- Increasing the number of epochs increased the Macro Average.
- Increasing the number of layers decreased the Macro Average
- Increasing the Patience increased the Macro average

Link to the Models

https://drive.google.com/drive/folders/1SMa52M7fYtKRe4Y6dwKRTreDrl_muj9_?usp=share_link