9 - Reference String Parsing in Scholarly Documents

Liu Xuan, Hon Hao Chen, Chang Lei, Yap Dian Hao
{e0253718,e0313661,e0253725,e0313668}@u.nus.edu

Abstract & Approach

Scholarly documents cite other resources to acknowledge their contribution through reference strings. Accurate parsing and tagging of reference strings to identify field tags, such as Author, Title, Date is important to allow effective navigation.

In NLP, several kinds of engineering work has been introduced to process the **labelling task**. We aim to analysis the performance of different models in order to gain further insights on their relevance.

Our approach:

- **Study the dataset** from Neural-Parscit to understand each reference string and tags
- Explore different combinations of **pre-processing** on reference strings
- Implement and test **different models**
- **Tune hyper-parameters** of each model
- Select the **most accurate model** and **perform ablation study**
- **Conduct analyses** on the selected model, pre-processed steps, and its parameters

SciWing

SciWing is a framework to process Scholarly documents references. It is built on *PyTorch* and equipped with many pre-trained models. In our project, we focus mainly on Neural-Parscit, one of the reference string parsing model in SciWing. We treat Neural-Parscit as our baseline model to perform compare and contrast with the models that we came up with. Neural-Parscit is implemented with Bi-LSTM-CRF model, and this is the envision we have gained. We have also ran our datasets using Neural-Parscit and we get a mixed result of 99%.

Datasets preparation

We selected our datasets from the Neural-Parscit repository with 1110 references, 7209 tags from 6 different fields of citation sources. We had several observations:

- Different **citations format** (APA, Harvard)
- Author first names are abbreviated with **different symbols**
- Different **date formats**
- Different **number of tags** in different reference strings

Below is an example of a reference string with tags to indicate the type of tokens.

M. Kitsuregawa, H. Tanaka, and T. Moto-oka. Application of hash to data base machine
Author Title
and its architecture. New Generation Computing, 1(1), 1983.
Journal Volume Date

Pre-processing

- **Regular expression** with group capturing
- **Tokenization**
- Testing out with Penn treebank **tagging**
- Converting words to vectors for **Word Embeddings**
- **Lowercasing**

Models

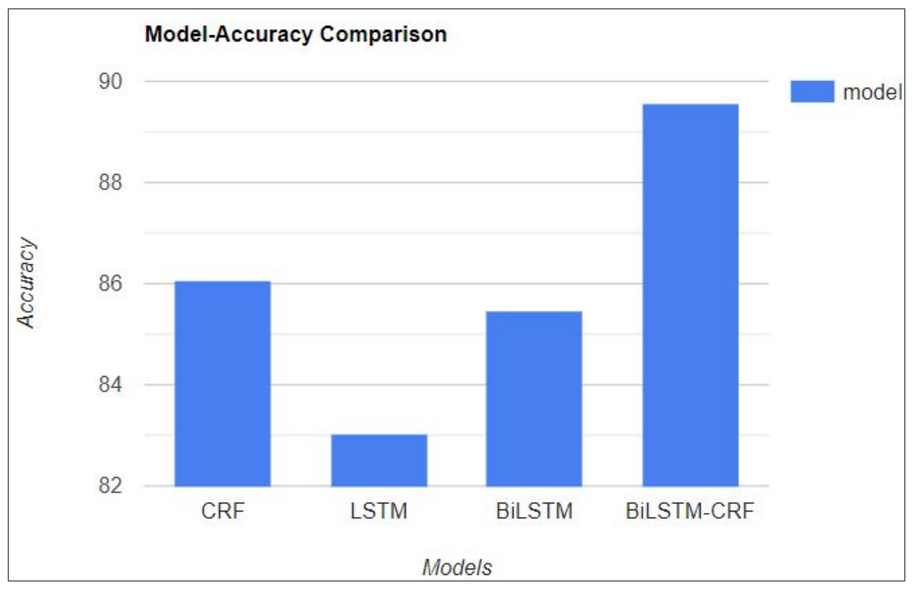
- **CRF** (Conditional Random Fields)
- **LSTM** (Long Short Term Memory)
- **BiLSTM** (Bi-directional Long Short Term Memory)
- **BiLSTM-CRF** (Combining both for performance test)

Hyperparameters Tuning

- **Activation functions** (softmax, tanh, relu, linear)
- Coefficients for **L1** and **L2** regularization
- **Loss function** (cross-entropy)

Initial Findings

Accuracies were obtained after 60 epochs of training. We also tested with 40 and 80 epochs and we discovered that 60 yield the best consistency of results without overfitting the data. BiLSTM-CRF is found to be the best model that produced the highest accuracy in identifying the tag in references. This is consistent with the architecture which Neural-Parscit from SciWing is built upon.



Feature Engineering: Subword Embedding

By subword embedding, we can analyse the internal structure of names and technical terminologies to improve the accuracy. Byte Pair Encoding performs a statistical analysis on the frequent consecutive characters of any length. We made use of *Sentencepiece* which is a good subword processor to handle subword embeddings.

Feature Engineering: Hierarchical Labelling

By identifying the structure of names, we can further improve the accuracy. We used *re* library to do further processing for author tag. The class "name" in reference strings are further classified into a tree structure of subclasses, "first name", "middle name", and "last name".

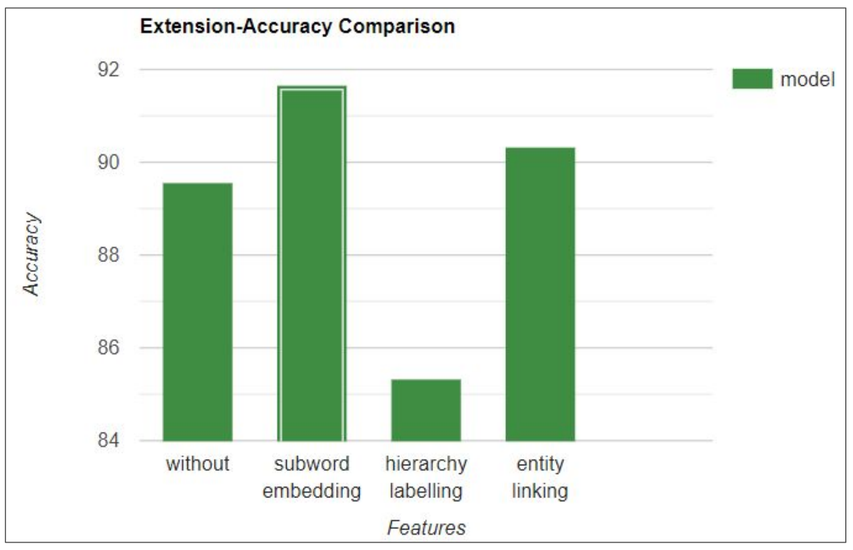
Feature Engineering: Entity Linking

The entity linking refers to online resources to determine how probable certain words and numbers are when grouped together in phrases. We used *Spacy's* pipeline to perform classification tasks of reference strings' classes such as authors and dates. We explored partial entity linking, which we classified author and date tags with entity linking, and total entity linking, which all labels are classified with entity linking.

Results and Insights

Based on our engineering work, we are able to obtain better performance for BiLSTM-CRF model with subword embedding and entity linking features.

BiLSTM-CRF Model with extension	Accuracy
Hierarchy labelling	85.3225827
Subword Embeddings	91.6585385
Entity Linking	90.4377281



Conclusion

Consistent experiments have shown that the extensions our team came up could increase the performance of reference string labelling task. We remain optimistic for hierarchy labelling as it might be due to error on our processing end. Regardless, we believe that if these extensions are built into SciWing, the SciWing model can improve its labelling accuracy further.

References:

1. Natural Language Processing for Hackers. (2018, September 8). *Build a POS tagger with an LSTM using Keras*. Retrieved from https://nlpforhackers.io/category/deep-natural-language-processing/
2. Kashyap, A. R. & Kan, M. (2020) SciWing -- A Software Toolkit for Scientific Document Parsing. Retrieved from https://arxiv.org/abs/2004.03807v2
3. Zhang, A., Lipton, Z. C., Li, M. & Smola, A. J. (2020). Dive into Deep Learning. Retrieved from https://d2l.ai
4. Prasad, A., Kaur, M. & Kan, M. (2018) Neural-Parscit: A Deep Learning Based Reference String Parser. Retrieved from https://link.springer.com/article/10.1007/s00799-018-0242-1