The 2018 Conference on Computational Linguistics and Speech Processing
ROCLING 2018, pp. 246-255

The Platform providing NLP System Deep Comparative Evaluation and Auxiliary Information for Hybrid NLP System Building ： Trial on Dependency Parser Evaluation

輔助建立混合性系統之自然語言處理系統深度評估平台 — 以評估依存關係分析器為例

王億祥 Yi-siang Wang
國立政治大學資訊科學系
Department of Computer Science
National Chengchi University
103305079@nccu.edu.tw

Abstract

This platform includes two main concepts. One, provide multifaceted and deep evaluation for users of Natural Language Processing (NLP) System (such as Syntaxnet or CoreNLP), to help them choose the best one for their needs. Two, independently evaluate single NLP stage of a NLP system (such as pos tagging or dependency parsing), to provide needed auxiliary information for building up a hybrid NLP system (NLP system which composes multiple NLP system for different NLP stages) which is better than a normal NLP system. This paper will be explanation and demonstration centered on these two goals.

摘要

本平台包含兩大概念。 一， 提供自然語言處理(NLP)系統(如Syntaxnet或CoreNLP)的多面向且深度的衡量，以期協助系統使用者挑選適合其需求的最佳系統。 二，獨立地評量單一NLP階段(如詞性標注或依存關係分析)，以提供在組建比一般的系統表現更好的混合型NLP系統(不同的階段使用不同的NLP系統)時，所需的輔助資訊。 本論文將圍繞此兩個目標展開說明和展示。

Keywords: Evaluation of NLP systems, Evaluation of NLP tools, compare NLP systems, compare NLP tools

關鍵詞： 評估自然語言處理系統，評估自然語言處理工具，自然語言處理系統比較，自然語言處理工具比較

1. Introduction

Most NLP projects are based on NLP systems, so the performance of them will deeply affects the result of the projects. But there are so many choices and NLP technology is constantly evolving, which makes normal user hard to get the best NLP system fits their needs.

In the previous researches, some didn't reflect sentence segmentation on their evaluation[1], some evaluate on only one language[2], some evaluate only under special cases[3], and some only focus on LAS and its extension but ignore evaluation on other metrics and factors[4], [5], and some aren't properly updated to the latest annotation scheme[6].

Generally, most of these are lack of a multifaceted and deep evaluation, which means rich choices of evaluation measure, that fit different needs of users or provide evaluations in different perspectives of viewing. Furthermore, we need to be able to evaluate single NLP stage of NLP system, thus we can pick up the best NLP system in the scope of every NLP stage respectively, and concatenate them into hybrid NLP system, which is expected to have better performance than normal NLP system.

In this paper, we will first explain our research scope, then briefly introduce evaluation measures for deep and multifaceted evaluation, then comes setting and usage of experiment, finally we will show the evaluation results of the experiment to demonstrate deep evaluation and single NLP stage evaluation for hybrid NLP system building.

2. Scope of this Paper and the Research

This paper is about a tool and core concepts behind it, so trivial or detailed things such as implementation of metrics, execution process, will be omitted, because of the page limit. And other thing, like means of evaluating single stage, are omitted because it is largely related to API of the NLP system or it is as the same as other related works do.

Additionally, there are many things to discuss about metrics and factors of evaluation, such as suited situations and not suited situations of them, but these are not in our research scope, because the tool is positioned to provide rich choices for users but not deeply research on them and we also expect it to be able to help the discussion on these measures. So in chapter 3 we will just briefly introduce metrics and factors.

3. Evaluation Measures

Though we won't deeply discuss about metrics and factors below, as mentioned above, we will briefly introduce every metric and factor and "one of" its significance for every unfamiliar metric and factor, to make readers have preliminary understanding about metrics and factors adopted.

3.1. Relevance Metrics

These three metrics matter when number of system prediction and gold differ.

Precision: number of correct prediction / number of system prediction

Recall : number of correct prediction / number of gold

F1-measure : (precision + recall) / 2

3.2. Metrics

Label Accuracy (LA) : The accuracy in assigning the correct dependency label.

Unlabelled Attachment Score (UAS) : The accuracy in assigning the correct dependency head.

Labelled Attachment Score (LAS) : The accuracy in assigning the correct head.

Morphology-Aware Labeled Attachment Score (MLAS) : Extend LAS with POS and morphological features. aims at comparability across typologically different languages, see CoNLL 2018 shared task for details[5].

Bi-lexical Dependency Score (BLEX) : Extend LAS with lemmatization, aims at evaluation closer to semantic content, see CoNLL 2018 shared task for details[5].

Dependency Branch Precision : The accuracy of path from root to a word in the dependency tree. To be a correct path, every word on the path should have correct label and parent. This metric aims at correctness of the main structure of a sentence.

Speed : Processed tokens or sentences per second, which is necessary to NLP projects that need immediate reaction.

3.3. Bases

Take “sentence-based” for example, instead of calculate one score in the scope of the whole document, first calculate scores in the scope of the same sentence respectively, and then take the average of these scores as the final score. This metric relieves the effect of extremely high or low performance in some sentences, see Table 1. Similarly we can get “POS-based”, and “dependency-label-based” for the sake of the same purpose.

Table 1. Example of no base and sentence-based when calculating precision.

	A sentence	B sentnece			Equation
tokens correctly predicted	10	25		nobase	(10+25) / (50+35) = 0.41
tokens predicted by system	50	35		sentence-based	(10/50 + 25/35) / 2 = 0.46

3.4. Factors

Dependency Distance : Absolute distance of parent and child of dependency in the sentence. This factor affects usage of memory and efficiency[7].

Dependency Direction : Direction from parent to child in the sentence. Relation to genre is one of the reasons that it is important[8].

Dependency Children Amount : How many words that depends on this word. Being the feature of classifying is one of the reasons that it matters[9].

Sentence Length: Number of non-space characters in a sentence. Performance under different sentence lengths might be meaningful for dataset with certain range of sentence length.

POS Tag: Performance under different POS tag matters when dataset has some POS tags frequent, or when being interested in some kinds of POS tag.

Dependency Label: Similar importance as mentioned in POS tag.

3.5. Other

Full / Main dependency label: For example, “nmod” is the main label of “nmod:tmod”. Some people might only care for the function of main label.

POS / UPOS : There is also universal POS(UPOS) in universal dependency, aims at POS annotation acrosses languages.

4. Experiment Setting

4.1. Dataset

Table 2. Dataset and distribution of data of our experiments.

UD_Chinese-GSD[1]	Tokens	Sentences
Training	98,608	3997
Developement	12,663	500
Test	12,012	500

4.2. NLP Systems

4.2.1. Syntaxnet

Based on transition-based neural networks[10], and have a major update in 2017[11]. But pre-trained models didn’t updated to the latest universal dependency[12], [13], so we will train our model with recommended setting according to the Syntaxnet’s document[2].

4.2.2. UDPipe

UDPipe is trainable pipeline specialized for universal dependency[14], good at others also. We will use its 2017 CoNLL shared task model[15].

5. Usage of Platform

First activate environment using docker, “docker-compose up -d”, then execute main program, “docker-compose exec app python experiment.py /path/to/dataset/directory/ <name of NLP system> [optional:NLP stage]”, and the evaluation results will be stored in the database. For your dataset or NLP systems, user should write code to implement abstract classes, which we won’t explain here because of the page limit.

[1] https://github.com/UniversalDependencies/UD_Chinese-GSD
[2]https://github.com/tensorflow/models/blob/master/research/syntaxnet/g3doc/syntaxnet-tutorial.md#part-of-speech-tagging

6. Experiment Results

Next, we will evaluate whole NLP task (will reflect result of preppended NLP works) for demonstration of deep evaluation, and independently evaluate single NLP stage for single NLP stage evaluation.

6.1. Deep Evaluation

There are 3 relevance metrics x 7 metrics x 4 base (include no-base) x 7 factors = roughly up to 588 scores, which proves multifaceted and deep evaluation we said. But due to the page limit, here only demonstrates several representative results and take Syntaxnet for example.

Table 3. Some scores with different relevance metrics on tokenization or POS tagging. we can also see that UPOS is hard to predicted than POS for Syntaxnet.

NLP Task	Method	Metric	Base	Other	Score
Tokenization	Precision	-	Token	-	98
POS Tagging	Pecall	-	Token	POS	92
POS Tagging	Recall	-	POS	POS	89
POS Tagging	Precision	-	POS	POS	93
POS Tagging	F1	-	POS	POS	91
POS Tagging	F1	-	Token	UPOS	79

Table 4. Scores on the same task but different bases. Scores differ obviously when different bases, even under the same other metrics.

NLP Task	method	metric	Base	Other	Score
Dependency Parsing	F1	LAS	Token	Full label	74
Dependency Parsing	F1	LAS	Sentence	Full label	75
Dependency Parsing	F1	LAS	POS	Full label	77
Dependency Parsing	Precision	LAS	UPOS	Full label	76
Dependency Parsing	Recall	LAS	UPOS	Full label	92
Dependency Parsing	Precision	LAS	Dependency Label	Full label	76
Dependency Parsing	Recall	LAS	Dependency Label	Full label	70

Table 5. Results on metrics that are not LAS. Since there is no morphology feature and lemma in Chinese, MLAS and BLEX are omitted.

NLP Task	method	metric	Base	Other	Score
Dependency Parsing	F1	LS	Token	Full label	82
Dependency Parsing	F1	LS	Token	Main label	83
Dependency Parsing	Precision	UAS	Token	-	82

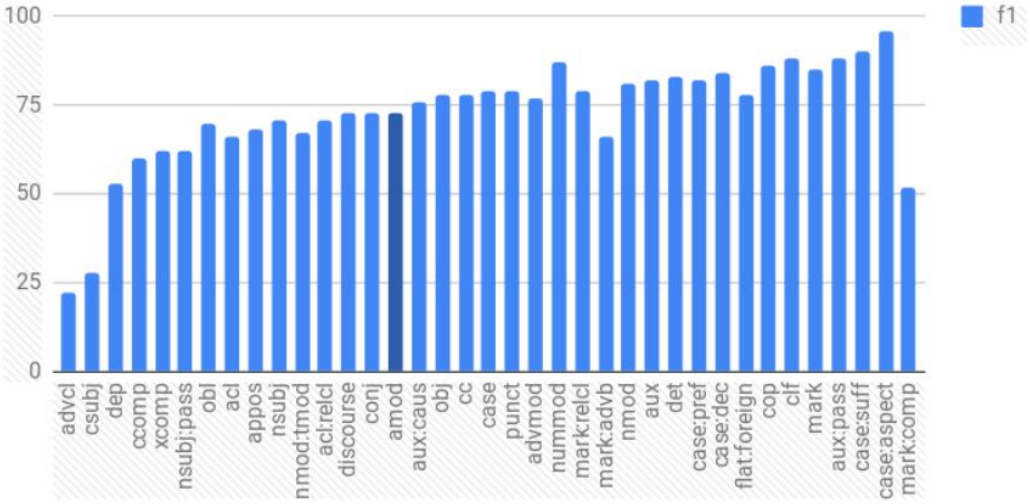

Figure 1. LAS under different dependency labels. Scores is low under some dependency, which may be a opportunity to research the reason of it.

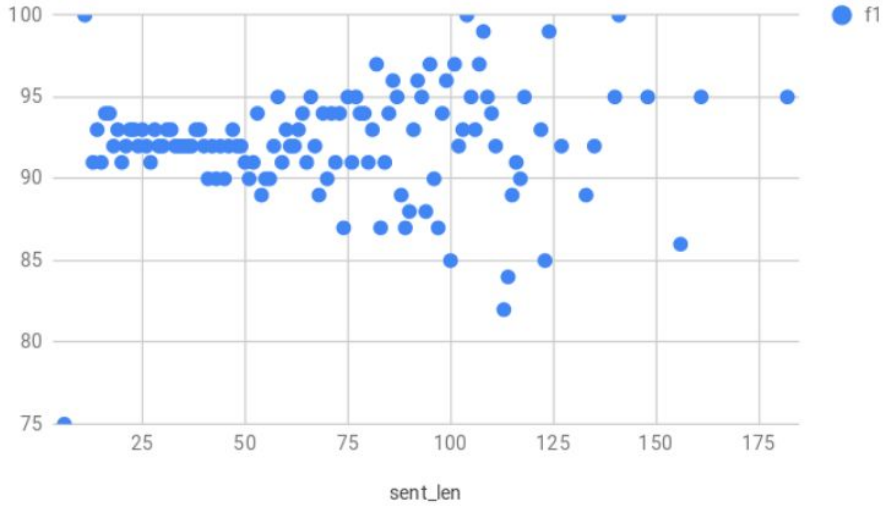

Figure 2. LAS under different sentence lengths. Performance is unstable in long sentences.

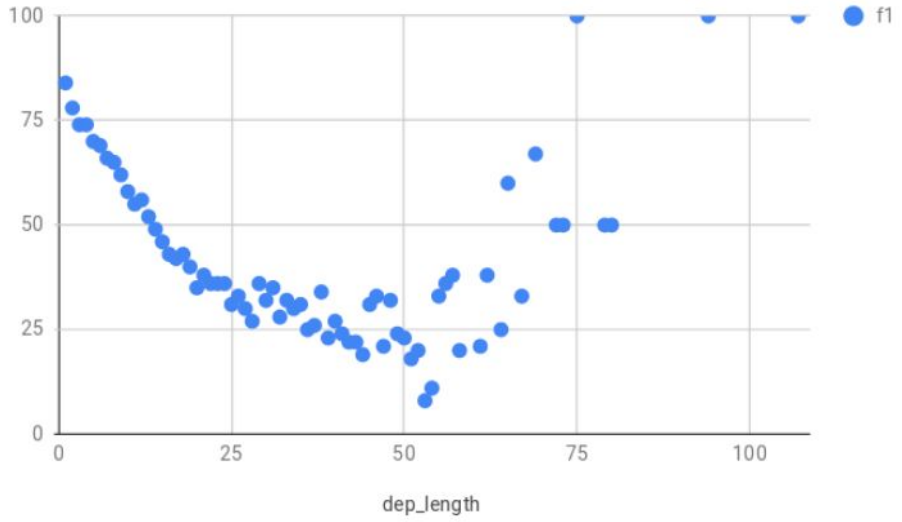

Figure 3. LAS (F1) under different dependency distances. The reason that score is low under dependency distance 50 is also worth researching.

6.2. Single NLP Stage Evaluation

Table 6. Independent evaluation on POS tagging for Syntaxnet and UDPipe.

NLP Stage	POS type	Score of Syntaxnet	Score of UDPipe
POS tagging	POS	93	98
POS tagging	UPOS	80	98

Table 7. Independent evaluation on dependency parsing for Syntaxnet and UDPipe

NLP Stage	Metric	Score of Syntaxnet	Score of UDPipe
dependency parsing	LA	88	89
dependency parsing	UAS	86	85
dependency Parsing	LAS	81	81

We can see that Syntaxnet performs better than UDPipe on head attachment, and UDpipe performs better than Syntaxnet on POS tagging. So if we can tag POS using UDPipe then parse using Syntaxnet, we should get better result on head attachment, than using just one of them to do it. Though slight difference in this case, still we showed the potential to provide auxiliary information for building hybrid NLP system (but not doing it).

7. Conclusions

In this paper, we first defined scope of this paper as introduction of the platform and concepts behind it, then brief introduce metrics and factors used, then came usage of the platform, and then experiment settings for technical details, finally got experiment results demonstrate the insighted multifaceted evaluation and potential to support building hybrid NLP system. So here, why we need the platform and what can the platform do is clear.

The platform is still under development, we haven't implemented speed metric, included more NLP systems and evaluated on larger corpus. But still, we have shown the main functions and potential of the platform.

Nowadays, NLP systems are more and more widely used, choosing a good NLP system contributes a lot to our projects. The platform provides rich measure for users can do some research on those then find the best evaluation for their needs. The platform also provides auxiliary information, make you possible to get better NLP system than the general one.

References

[1] J. D. Choi, J. Tetreault, and A. Stent, "It Depends: Dependency Parser Comparison Using A Web-based Evaluation Tool," in Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), 2015, pp. 387–396.

[2] N. N. Alam, "The Comparative Evaluation of Dependency Parsers in Parsing Estonian," University of Tartu, 2017.

[3] M. Alabbas and A. Ramsay, "Evaluation of dependency parsers for long Arabic sentences," in 2011 International Conference on Semantic Technology and Information Retrieval, 2011.

[4] D. Z. Jan Hajič, Ed., "CoNLL-2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies," in Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies, 2017.

[5] D. Z. Jan Hajič, Ed., "CoNLL-2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies," in Proceedings of the CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies, 2018.

[6] J. N. Nilsson Jens, "MaltEval: An Evaluation and Visualization Tool for Dependency Parsing," in Proceedings of the Sixth International Conference on Language Resources

and Evaluation (LREC'08), 2008.

[7] K. Gulordava, P. Merlo, and B. Crabbé, "Dependency length minimisation effects in short spans: a large-scale analysis of adjective placement in complex noun phrases," in Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers), 2015, pp. 477–482.

[8] Y. Wang and H. Liu, "The effects of genre on dependency distance and dependency direction," Lang. Sci., vol. 59, pp. 135–147, 2017.

[9] Prudhvi Kosaraju, Sruthilaya Reddy Kesidi, Vinay Bhargav Reddy Ainavolu and Puneeth Kukkadapu, "Experiments on indian language dependency parsing," in Proceedings of the ICON10 NLP Tools Contest: Indian Language Dependency Parsing, 2010, pp. 40–45.

[10] D. Andor et al., "Globally Normalized Transition-Based Neural Networks," in Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), 2016.

[11] Lingpeng Kong, Chris Alberti, Daniel Andor, Ivan Bogatyy, David Weiss, "DRAGNN: A Transition-based Framework for Dynamically Connected Neural Networks." 13-Mar-2017.

[12] Chris Alberti, Daniel Andor, Ivan Bogatyy, Michael Collins, Dan Gillick, Lingpeng Kong, Terry Koo, Ji Ma, Mark Omernick, Slav Petrov, Chayut Thanapirom, Zora Tung, David Weiss, "SyntaxNet Models for the CoNLL 2017 Shared Task," in Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies, 2017.

[13] Ryan McDonald , Joakim Nivre, Yvonne Quirmbach-Brundage, Yoav Goldberg, Dipanjan Das , Kuzman Ganchev, Keith Hall, Slav Petrov, Hao Zhang, Oscar Täckström, Claudia Bedini, Núria Bertomeu Castelló, Jungmee Lee, "Universal dependency annotation for multilingual parsing," in Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics, 2013, pp. 92–97.

[14] H. J. Straka Milan, "UDPipe: Trainable Pipeline for Processing CoNLL-U Files Performing Tokenization, Morphological Analysis, POS Tagging and Parsing," in the Tenth International Conference on Language Resources and Evaluation (LREC 2016), 2016.

[15] M. Straka and J. Straková, "Tokenizing, POS Tagging, Lemmatizing and Parsing UD 2.0 with UDPipe," in Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies, 2017, pp. 88–99.