Concentric: Studies in Linguistics
42.1 (May 2016): 45-72 DOI: 10.6241/concentric.ling.42.1.03

Word Dependency Sketch for Chinese Language Learning*

Meng-Hsien Shih & Shu-Kai Hsieh
National Taiwan University

This article describes an approach to constructing a language resource through automatically sketching grammatical relations of words in an untagged corpus based on dependency parses. Compared to the handcrafted, rule-based Word Sketch Engine (Kilgarriff et al. 2004), this approach provides more details about the different syntagmatic usages of each word such as various types of modification a given word can undergo and other grammatical functions it can fulfill. As a way to properly evaluate the approach, we attempt to evaluate the auto-generated result in terms of the distributional thesaurus function, and compare this with items in an existing thesaurus. Our results have been tailored for the purpose of Chinese learning and, to the best of our knowledge, the resulting resource is the first of its kind in Chinese. We believe it will have a great impact on both Chinese corpus linguistics and Teaching Chinese as a Second Language (TCSL).

Key words: dependency relation, computational lexicography, thesaurus, corpus linguistics

1. Introduction

Syntagmatic relational information has been the focus of interface studies in syntax and semantics over the past few years. With the rapid development of corpora in recent years, various corpus query tools, profiling tools and visualization tools have emerged quickly. Among these tools, Word Sketch Engine (WSE), originally developed in the United Kingdom for the English language (Kilgarriff et al. 2004, Huang et al. 2005), has provided an effective approach to quantitatively summarize grammatical and collocation behavior.[1] Its functions include Concordance, Word List, Word Sketch, Sketch Difference, Thesaurus, other web corpus crawling tools and processing tools. Recently, an implementation of the WSE-based interface for language learners, Sketch Engine for Language Learning (SkELL), has been introduced (Baisa & Suchomel 2014).[2]

Despite being proprietary, the Chinese version of the WSE system developed by Academia Sinica[3] has gained popularity among Chinese corpus linguists and language teachers because of its featured functions for grammatical collocational analysis (Huang et al. 2005, Hong & Huang 2006). In spite of the clear advantages of this

* We would like to thank the editors and two anonymous reviewers and copy editors for their valuable comments, which helped us considerably improve the quality of the paper. An early version of this research has been presented in the 26th Conference on Computational Linguistics and Speech Processing (ROCLING 2014), Jhongli, Taiwan, on Sep. 25-26, 2014.
[1] http://www.sketchengine.co.uk
[2] http://skell.sketchengine.co.uk
[3] http://wordsketch.ling.sinica.edu.tw

approach, the construction of WSE is time-consuming because its approach is top-down, requiring manually created sketch grammars. As an alternative to the top-down manner, the statistical dependency parser, as implemented in the Stanford Parser, works in a corpus-driven way. It not only provides more fine-grained grammatical relations compared to WSE, but it can also capture probabilistic information of linguistic constraints via a dependency structure of words and their collocates. Therefore, in this paper we propose an alternative approach of automatically sketching the grammar profile of words from a text corpus. By replacing the sketch grammar in the WSE system with a dependency parser (cf. Section 3), we would no longer require a POS (Part-Of-Speech)-tagged corpus to perform a word sketch. Rather, with the help of a parser, we could sketch word behavior in an untagged corpus. We could even exploit developments in the field of computational linguistics, such as deep learning, by updating the parser to a more accurate or faster system.

This paper is organized as follows: Section 2 reviews the current design of the WSE system. Section 3 introduces the dependency grammar framework and its application in WSE. Section 4 proposes a dependency-based approach to sketching words in a parsed Chinese corpus. Section 5 presents the results from the proposed approach and an evaluation. Section 6 analyzes the errors in the results. The final section concludes this paper and proposes the possible direction of future work.

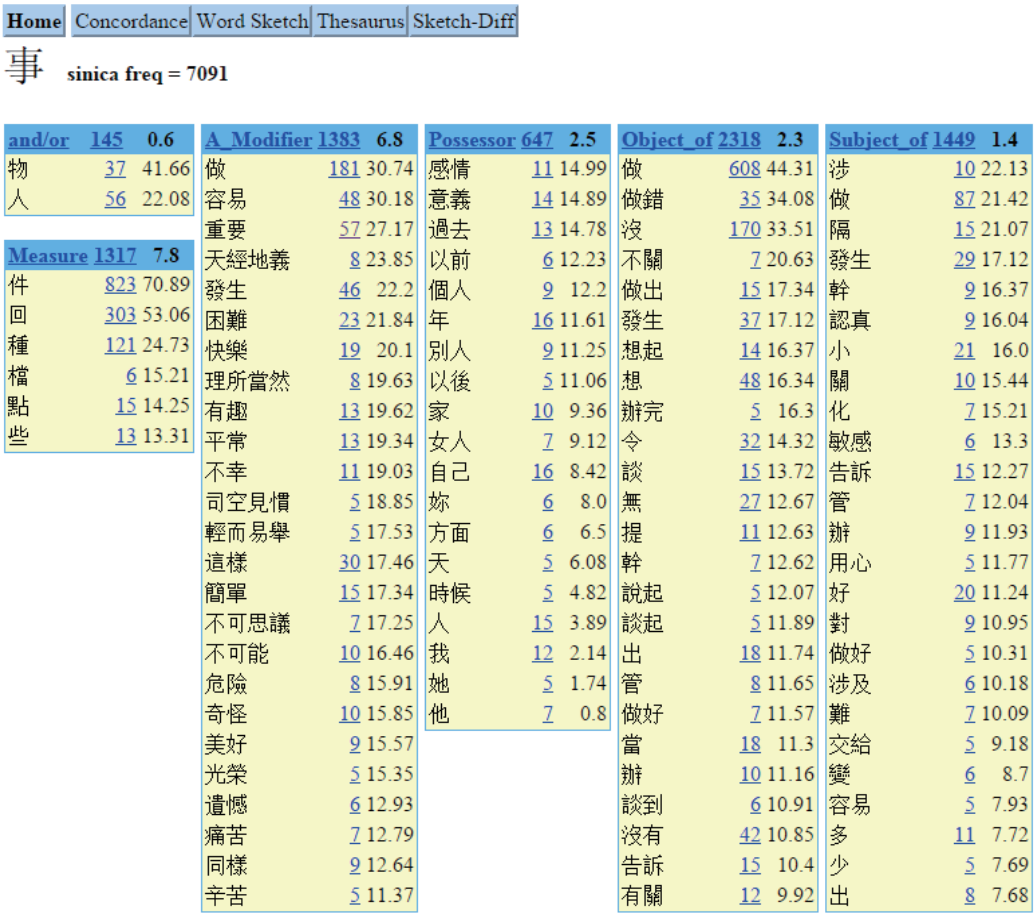
2. Current design of the word sketch engine

Given diverse needs and technical advances, the number of corpus query tools has grown over the past decade. Among them WSE provides a set of corpus query tools, such as a concordance, word grammar sketch and difference sketch, and a thesaurus, that aim to help users reveal linguistic patterns in language use. The grammatical collocation of a word (i.e., word sketch) is probably the most popular function, and it has been widely applied in studies of corpus linguistics and language pedagogy (Kilgarriff 2007). Recently, a light-weight version called SkELL targeted at language learners was also released (Baisa & Suchomel 2014).[4]

Collocation is an interesting linguistic phenomenon concerning the fact that certain words are more likely to co-occur. A collocate is defined as a word that occurs within the neighbouring context of another word. The strength of the co-occurrence can be estimated by various statistical measures, such as Mutual Information (MI) and log-likelihood. However, these measures are grammatically blind because they reveal only syntagmatic proximity. Collocates, though, are bound to the node word through a

[4] http://skell.sketchengine.co.uk

particular grammatical relation. This aspect of collocates has not been capitalized on in previous corpus tools (Kilgarriff & Kosem 2012). WSE, therefore, proposes to combine collocations and grammar, implemented as a function that produces "one-page automatic, corpus-based summaries of a word's grammatical and collocational behavior" (Kilgarriff et al. 2004:105). For instance, Figure 1 illustrates the word sketch of the noun *shi* [事] 'thing' as used in the Sinica Corpus. The salient collocates of *shi* are organized by their grammatical relations in terms of *subject, object, modifier* or in the *and/or* coordinate relation.



Home | Concordance | Word Sketch | Thesaurus | Sketch-Diff

事 sinica freq = 7091

and/or	145	0.6
物	37	41.66
人	56	22.08

Measure	1317	7.8
件	823	70.89
回	303	53.06
種	121	24.73
檔	6	15.21
點	15	14.25
些	13	13.31

A_Modifier	1383	6.8
做	181	30.74
容易	48	30.18
重要	57	27.17
天經地義	8	23.85
發生	46	22.2
困難	23	21.84
快樂	19	20.1
理所當然	8	19.63
有趣	13	19.62
平常	13	19.34
不幸	11	19.03
司空見慣	5	18.85
輕而易舉	5	17.53
這樣	30	17.46
簡單	15	17.34
不可思議	7	17.25
不可能	10	16.46
危險	8	15.91
奇怪	10	15.85
美好	9	15.57
光榮	5	15.35
遺憾	6	12.93
痛苦	7	12.79
同樣	9	12.64
辛苦	5	11.37

Possessor	647	2.5
感情	11	14.99
意義	14	14.89
過去	13	14.78
以前	6	12.23
個人	9	12.2
年	16	11.61
別人	9	11.25
以後	5	11.06
家	10	9.36
女人	7	9.12
自己	16	8.42
妳	6	8.0
方面	6	6.5
天	5	6.08
時候	5	4.82
人	15	3.89
我	12	2.14
她	5	1.74
他	7	0.8

Object_of	2318	2.3
做	608	44.31
做錯	35	34.08
沒	170	33.51
不關	7	20.63
做出	15	17.34
發生	37	17.12
想起	14	16.37
想	48	16.34
辦完	5	16.3
令	32	14.32
談	15	13.72
無	27	12.67
提	11	12.63
幹	7	12.62
說起	5	12.07
談起	5	11.89
出	18	11.74
管	8	11.65
做好	7	11.57
當	18	11.3
辦	10	11.16
談到	6	10.91
沒有	42	10.85
告訴	15	10.4
有關	12	9.92

Subject_of	1449	1.4
涉	10	22.13
做	87	21.42
隔	15	21.07
發生	29	17.12
幹	9	16.37
認真	9	16.04
小	21	16.0
關	10	15.44
化	7	15.21
敏感	6	13.3
告訴	15	12.27
管	7	12.04
辦	9	11.93
用心	5	11.77
好	20	11.24
對	9	10.95
做好	5	10.31
涉及	6	10.18
難	7	10.09
交給	5	9.18
變	6	8.7
容易	5	7.93
多	11	7.72
少	5	7.69
出	8	7.68

Figure 1. Word sketch of *shi* [事] 'thing'

To extract the word sketch information, the WSE system assumes no available syntactically parsed corpus and adopts a top-down grammar writing approach. Given tokenized and POS-tagged corpus data, the WSE system makes use, in most languages (Ambati, Reddy & Kilgarriff 2012, Kilgarriff et al. 2014), of an extended Corpus Query Processor (CQP) syntax, to define the grammatical relations throughout the POS-tagged corpus data. The CQP-syntax[5] employs *regular expressions,* i.e., a

[5] The CQP was developed at the IMS, University of Stuttgart in the early 1990s.

sequence of characters that, in computer science, define a search pattern at the levels of character strings and token sequences, which has gained popularity in corpus encoding and indexing technology.

In WSE-related papers, CQP-syntax is usually referred to as CQL (Corpus Query Language). This can be flexibly applied to a sequence of token specifications in order to search for complex lexico-grammatical patterns (Evert & Hardie 2011). The core component in the WSE system is the so-called sketch grammar, which is mostly manually crafted by linguists. With the CQL extension, the sketch grammar defines linear patterns to automatically identify possible grammatical relations to a node word, as constrained by the surrounding context. This sketch grammar is used by a finite-state shallow parser to extract various grammatical relations.[6] Typical grammatical relations as defined in the English WSE include: [OBJECT_OF], [ADJ_MODIFIER], [NOUN_MODIFIER], [MODIFIES], [AND/OR], and [PP_INTO]. For instance, one of the sketch grammar rules defined in the huge Chinese corpus provided by WSE[7] concerns modification. Using this, we can identify cases of modification relation where the node word (indicated by the prefix "1:") can be any noun followed by non-nouns. The collocate, i.e., the word you want to capture (marked with the prefix "2:"), is any verb followed by the word *de* [的] 'DE' as shown in rule (1):

(1) =A_Modifier/Modifies
2:"V.*" [word="的"] [tag="N.*"]{0,2} 1:[tag="N.*"] [tag!="N.*"]

Consider the following example of "…*kuaile* [快樂](V) *de* [的](DE) *shi* [事](N)…" 'the happy things'. Given the node word *shi* [事] 'thing' to sketch, we may find sentences in the corpus with *shi* preceded by the verb *kuaile* [快樂] 'happy' and *de* [的] 'DE', which happens to match the sketch grammar rule (1) for Modifier. In this case, the Sketch Engine would be able to capture the relation that *kuaile* modifies *shi*, or the two words have a *modifier/modifies* relation.

Sketch grammar can be even more complicated with the increasing granularity of POS information. Grammar rule (2) shows the *Classification/Measure* relation developed by Huang et al. (2005) and implemented in the Chinese Word Sketch system. The node word can be a noun preceded by a measure word (tagged by Nf):

[6] http://www.sketchengine.co.uk/documentation/wiki/SkE/Help/CreateCorpus
[7] zhTenTen, with 2.1 billion tokens, is the huge Chinese corpus provided by WSE (Jakubíček et al. 2013).

(2) =Measure
2:“Nf.*” (“A”|“VH11”|“VH13” “VH21”|“V.*” “DE") [tag=“N[abcd].*” & tag!= “Ncd”] 1:[tag=“N[abcdhf].*” & tag!= “Nbc.*” & tag!= “Ncd.*” & word!= “者” & word!= “們”] [tag!= “N[abcdhef].*” |tag=“Nbc.*” |tag=“Ncd.*”]

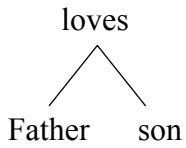
3. Word sketch and dependency grammar

The sketch grammar approach to grammatical collocation extraction can achieve a reasonably high rate of precision, that is, of all the extracted candidate collocates, a great many of them are indeed collocates. However, this approach often runs the risk of having a low recall rate, which means many true collocates cannot be identified. A recent comparative evaluation of sketch grammar and dependency-based approaches conducted on the Slovene Lexical Database has also attested to this (with precision: 84.9% vs. 88.4% and recall: 56.5% vs. 88.3%, respectively) (Krek & Dobrovoljc 2014).

In addition, the writing of such grammar is time-consuming and labor-intensitive, so we would like to exploit the latest parser techniques to capture word relations without any POS-tagged corpus. So it is in this research that we exploit the dependency parser to enrich the relational information. We also plan to develop a parser trained on traditional Chinese in future work. Unlike phrase-structure grammar, dependency grammar concentrates on the *typed dependency* between words rather than constituent information. It is highly advantageous to our study, for it is linguistically rich — capturing not only syntactic information such as *nsubj* (nominal subject) but also abstract semantic information such as *loc* (localizer) — and can be further applied to other syntactic-semantic interface tasks (Chang et al. 2009, de Marneffe et al. 2014).

Modern dependency grammar may be dated to the influential French linguist Lucien Tesnière’s *Éléments de syntaxe structural* (Elements of Structural Syntax 1959, 2015). Tesnière focused on the connections between words in a sentence, while we now label directed connections as dependencies. Every word of a sentence is either directly or indirectly connected to the verb in the sentence. He illustrated this with the sentence ‘A father loves a son’ (translated from Latin):

(3)



In this diagram, the verb ‘loves’ is superior in the sentence, while both ‘Father’ and ‘son’ are subordinate to the verb.

On the other hand, scholars coming from another position might favor the binary division, as illustrated in the French sentence 'Alfred speaks slowly':

(4) Alfred — speaks
|
slowly

This position hypothesizes that every sentence is divided into two parts: subject and predicate, as with this example, which is divided into the subject 'Alfred' and the predicate 'speaks slowly'. These two positions were later developed into phrase structure grammar and dependency grammar.

Unlabeled dependency parses actually can be derived from phrase structures (Xia & Palmer 2001). The algorithm used to automatically derive an unlabeled dependency parse from a phrase structure is shown below:

(a) Mark the head child of each node in a phrase structure using the head percolation table.
(b) In the dependency structure, make the head of each nonhead child depend on the head of the head child.

To illustrate, let us take an example of the phrase structure from Penn Treebank for 'Vinken will join the board as a nonexecutive director Nov 29'. The phrase structure is shown below:

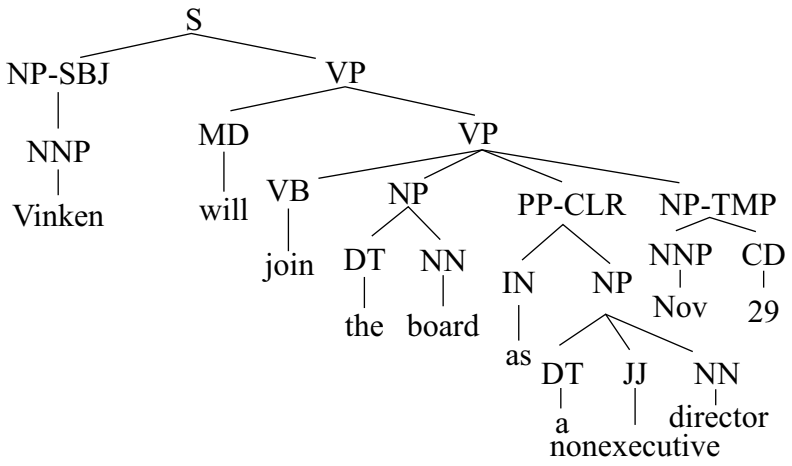


Figure 2. Phrase structure from Penn Treebank for 'Vinken will join the board as a non-evecutive director Nov 29'

Its context-free grammar should consist of the following rules:
S → NP [VP]
VP → [VB] NP
NP → DT [NN]
NP → NNP
NNP → Vinken
VB → join
DT → the
NN → board
(The *head child* is in square brackets)

To derive dependency parses from this phrase structure, you can check the head percolation table for the first grammar rule, finding that the *head child* of sentence node S is VP. So, in this sentence the head of VP is superior to the head of the other node NP, and eventually it is discovered that VB 'join' is the head of the VP after traversing the phrase structure tree, with NNP 'Vinken' as the head of the NP. Therefore, a dependency relation with 'join' superior to 'Vinken' is established. This operation is repeated to derive other dependency relations between the words in the sentence until there are no more grammar rules to check.

Take a Chinese sentence *da xiaohai de muqin* [打小孩的母親] 'hit the kid's mother' as another example. We can infer a *direct object* (*dobj*) relation between a verb and noun wherever the verb of the VP and the head noun of the NP descend from the same VP, and another *modifier* relation from any NP where the noun in an adjective phrase and the head noun descend from the same NP. However, it should be noted that the dependency parser used in this research employs a slightly different approach of derivation from phrase structure parses, which will be elaborated upon later.

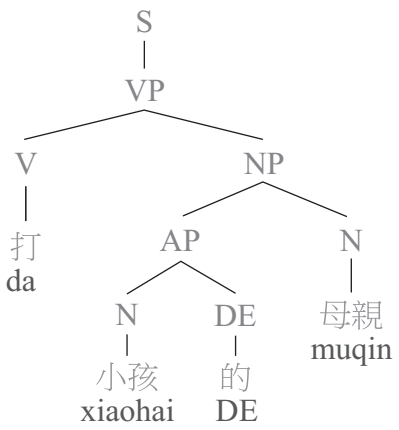


Figure 3. A phrase structure to derive dependency relations

The parser used in this paper, the Stanford lexicalized probabilistic parser version 3.4 (Levy & Manning 2003), works out the grammatical structure of sentences with a factored product model which efficiently combines preferences of PCFG phrase structure and lexical dependency experts. In addition to the phrase structure tree, the parser also provides *Stanford Dependencies* (SD)[8] that are known as representations of grammatical relations between words in a sentence. Take the following Chinese sentence for example: *Wo hen xihuan liang ze xifu yu xiyuan de gushi.* [我很喜歡兩則惜福與惜緣的故事。] 'I quite like the two stories of cherishing luck and cherishing affinity.' The head *xihuan* [喜歡] 'like' has the dependent*, wo* [我] 'I', as its nominal subject, and another dependent, *gushi* [故事] 'story ', as its direct object (Figure 4).

(ROOT (IP (NP (PN 我)) (VP (ADVP (AD 很)) (VP (VV 喜歡) (NP (DNP (NP (NP (NR 兩)) (NP (NN 則) (NN 惜福) (NN 與) (NN 惜緣))) (DEG 的)) (NP (NN 故事))))) (PU。)))	nsubj(喜歡-3, 我-1) advmod(喜歡-3, 很-2) root(ROOT-0, 喜歡-3) nn(惜緣-8, 兩-4) nn(惜緣-8, 則-5) nn(惜緣-8, 惜福-6) nn(惜緣-8, 與-7) assmod(故事-10, 惜緣-8) assm(惜緣-8, 的-9) dobj(喜歡-3, 故事-10)

Figure 4. Dependencies in the Chinese sentence *Wo hen xihuan liang ze xifu yu xiyuan de gushi* [我很喜歡兩則惜福與惜緣的故事] 'I quite like the two stories of cherishing luck and cherishing affinity.' with phrase structures

The Stanford dependency parser extracts dependency relations from the structures generated by another phrase structure parser (Klein & Manning 2003). The dependency parser first extracts the semantic head of a phrase based on rules similar to the Collins syntactic head rules (Collins 1999); then the extracted head and corresponding dependents are labeled with dependency relations by pre-defined patterns. For example, the Chinese sentence in Figure 4 has as its VP *xihuan liang ze*

[8] http://nlp.stanford.edu/software/stanford-dependencies.shtml

xifu yu xiyuan de gushi [喜歡兩則惜福與惜緣的故事] in a phrase structure. The dependency parser first extracts the head *xihuan* [喜歡] 'like' and the corresponding dependent *gushi* [故事] 'story' based on semantic head rules; then it labels the extracted pair with the *dobj* relation based on one dependency pattern of *dobj* defined over the phrase structure parse tree. Here we can also observe the fine-grained relations between words in this parser, such as the adverbial modification (*hen* [很] 'much' modifies *xihuan* [喜歡] 'like') and associative modification (*xiyuan* [惜緣] 'affinity valued' modifies *gushi* [故事] 'story').

The SD has been widely used in NLP-related fields such as sentiment analysis (Meena & Prabhakar 2007), textual entailment (Androutsopoulos & Malakasiotis 2010). The Chinese version of SD (Chang et al. 2009) trained on the Penn treebank is available on the Stanford Dependencies page.[9] The SD can distinguish 45 typed dependencies among Chinese words, as shown in Table 1.

Table 1. Chinese dependency relations (excerpted and converted into Traditional Chinese characters from Chang et al. 2009)

abbre-viation	short description	Chinese example	typed dependency	counts	percent-age
nn	noun compound modifier	服務 中心	nn(中心,服務)	13278	15.48%
punct	punctuation	海關 統計 表明 ，	punct(表明,，)	10896	12.71%
nsubj	nominal subject	梅花 盛開	nsubj(盛開,梅花)	5893	6.87%
conj	conjunct	設備 和 原材料	conj(原材料,設備)	5438	6.34%
dobj	direct object	浦東 頒布 了 七十一 件 文件	dobj(頒布,文件)	5221	6.09%
		⋮			
nsubj -pass	nominal passive subject	鎳 被 稱作 現代 工業 的 維生素	nsubjpass(稱作,鎳)	14	0.02%

[9] http://nlp.stanford.edu/software/stanford-dependencies.shtml#Chinese; an on-going project for universal dependencies across different languages can be referred to http://universaldependencies.github.io/docs/

4. Proposed dependency-based sketch system

In this study, we chose the Sinica Balanced Corpus (Chen et al. 1996) as our data source since this corpus has already had its word segmentation and POS tagging manually checked, though the parser only requires segmented texts. After removing the POS information, untagged texts of 567,702 short sentences (less than 30 words, in order to save computing time) from Sinica Corpus 3.0[10] were parsed with dependency relations by the Stanford Parser 3.4 (Chang et al. 2009) running by Java SE 1.6 on four Amazon EC2 servers (Xeon E5-2670 3.1Ghz, 20M Cache). This took approximately 16 hours to process (10 sentences/sec), and we obtained 574,552 dependency relations (of 23 types) between 44,257 words.

To sketch a word, we made use of the dependency tuples from the parsed corpus (see the right panel of Figure 4) to extract the relations of the word with its dependents. The sample sketch obtained in this manner is exemplified in *da* [打] 'hit', as shown below:

Table 2. Dependency sketch of *da* [打] 'hit'
(Matches with Chinese Sketch Engine are marked in bold-faced red)

PREP	DOBJ	ADVMOD/MMOD	NSUBJ	ASP	CONJ
在	**電話**	**去**	**武松**	了	重建
到	**折**	**要**	**棍子**	著	是
自	**籃球**	**就**	**球**		鬧
	高爾夫球	**先**	**我**		
	硬仗	**不會**	**你**		
	招呼	**該**	**他**		
	折扣	**一起**	**爸爸**		
	哈欠	**會**	**兩**		
	太極拳	連續	**人**		
	麻藥針	**一**	**老師**		
	盹兒	**能**	**他們**		
	虎	**可以**	**她**		
	羽毛球	還要	**學生**		
	排球	**都**	**自己**		
	蛇	雖然	湖人		
	起來	仍然	來		
	秋千	而	政		

[10] http://www.sinica.edu.tw/SinicaCorpus

Besides typical relations in the Word Sketch Engine such as OBJECT, MODIFIER and SUBJECT, the Dependency Sketch further provides relations such as CONJUNCT and ASPECT markers, which might be important for language pedagogy and linguistic research.

5. Evaluation

Since the Stanford Parser still suffers from parsing difficulty in Chinese, the grammatical relations automatically acquired, though impressive, may contain heterogeneous errors originating from mistagging errors,[11] syntactic ambiguities and other dependency parsing issues. Therefore we observed some minor sketch errors in the results. However, it is hard to evaluate the results in an automatic way as conventionalized in the field of NLP. The main reasons are:

1. Currently, there is no gold standard (in Chinese). It is particularly hard to measure recall as the set of 'correct answer' is not available.
2. An overall evaluation of the sketch performance necessarily relies on the separate assessment of each module (word segmentation, POS tagging, sketch grammar and/or dependency parsing, etc.). A comparative table is shown in Table 3.

Table 3. Comparison of different word sketch systems

Word Sketch System	Word segmentation	Pos tagging/tagset	Sketch grammar	Dependency parser
CWSE.sinica	CKIP	CKIP/ASBC	Hand-crafted rules	*
zhTenTen.11	Stanford Chinese Word Segmenter	Stanford Log-linear Part-Of-Speech Tagger/Chinese Penn Treebank standard	Hand-crafted rules	*
Proposed	Stanford Chinese Word Segmenter	*	*	Stanford dependencies

In addition, from the perspective of language resource construction as well as applied lexicography, as the system aims to identify highly salient candidate patterns,

[11] In this study, since the Stanford Parser takes manully-tokenized input from the Sinica Corpus, the number of segmentation errors may be fewer than the results come from an automatic segmenter had and are, therefore, omitted here.

the noisy data should not constitute a serious problem. This position is also well-articulated and proposed by Kilgarriff et al. (2010), where a variant of an evaluation paradigm (user/developer-oriented paradigm) is required.

Unlike Ambati, Reddy & Kilgarriff (2012) and Reddy et al. (2011) where external evaluation tasks such as *topic coherence* or *semantic composition* were adopted, and due to time constraints, the proposed resource was not manually evaluated by language teachers or learners as Word Sketch Engine was (Kilgarriff et al. 2010). Instead, we evaluated the proposed method on its performance on the task of automatically constructing a thesaurus, for our main concern is the construction of a language resource rather than NLP system performance.

The thesaurus in the WSE is called a **distributional thesaurus**, and can be built for any language if the word sketch data of the language is available. This thesaurus is constructed by computing the similarity between words based upon the overlapping rate of their word sketches. We maintained the same thesaurus function found in WSE and anchored it to a manually constructed thesaurus, Chilin[12] (Chao & Chung, 2013).

We adopted the measures of word association and distance from WSE to generate a thesaurus. In WSE, the association of two words in relation R is calculated as their logDice coefficient:[13]

$$(5)\quad AScore(w_1, R, w_2) = 14 + \log \frac{2 \cdot \|w_1, R, w_2\|}{\|w_1, R, *\| + \|*, *, w_2\|}$$

Here $\|w_1, R, w_2\|$ refers to the number of occurrences of w_1 and w_2 in relation *R*, $\|w_1, R, *\|$ the number of occurrences of w_1 in relation *R*, and $\|*, *, w_2\|$ the number of occurrences of w_2.

In fact, the *logDice* coefficient attempts to measure the association of two words in terms of the *Harmonic Mean* of two conditional probabilities, the probability of a w_1 occurrence given w_2, and the probability of a w_2 occurrence given w_1:

$$(6)\quad H(P(w_1|w_2), P(w_2|w_1)) = \frac{2}{\frac{1}{P(w_1|w_2)} + \frac{1}{P(w_2|w_1)}} = \frac{2}{\frac{P(w_2)}{P(w_1, w_2)} + \frac{P(w_1)}{P(w_1, w_2)}}$$
$$= \frac{2}{\frac{f(w_2)}{f(w_1, w_2)} + \frac{f(w_1)}{f(w_1, w_2)}} = \frac{2f(w_1, w_2)}{f(w_1) + f(w_2)}$$

[12] http://code.google.com/p/tw-synonyms-chilin
[13] http://www.sketchengine.co.uk/documentation/raw-attachment/wiki/SkE/DocsIndex/ske-stat.pdf

Although this coefficient looks similar to the formula for Mutual Information (7), a major distinction is that addition has been replaced by the multiplication in the denominator, which returns a significantly different result mathematically.

$$(7)\ MI = \log \frac{N \cdot \|w_1, w_2\|}{\|w_1\| \|w_2\|}$$

On the other hand, similarity between two words is measured as the association score of the overlapping of the two words across relations:

$$(8)\ Sim(w_1, w_2) = \frac{\sum_{(r,c) \in ctx(w_1) \cap ctx(w_2)} logDice(w_1, r, c) + logDice(w_2, r, c)}{\|w_1, R, *\| + \|*, *, w_2\|}$$

In principle, the synonyms of a target word are regarded here as those words having the highest similarity scores with the target words. In practice, we calculated the similarity scores between target words and all other words, ranked these words according to their similarity scores. We then extracted the top ten words as synonyms of the target words.

The results of the ranked synonyms were evaluated with the semi-manual thesaurus, Chilin. The accuracy rate was calculated by dividing the number of correctly extracted synonyms (with accuracy based on Chilin) with the number of synonyms from Chilin:

$$(9)\ Accuracy\ Rate = \frac{count(extracted\ synonyms \cap Chilin)}{count(Chilin)}$$

6. Results and analysis

We compared our extracted synonyms with synonyms in the Chilin thesaurus. Of 17,817 Chilin entries (類), 9,995 are target synonym entries labeled with " = " for our evaluation, as shown in the following table:

Table 4. Some Statistics on Chilin

Synonym	9,995
Near-Synonym	3,445
Closed	4,377
Total	17,817

Of 9,995 synonym entries, there were only 7,258 headwords in the entries for which dependency data were available for thesaurus generation. We extracted 657 synonyms for 503 entries of the Chilin thesaurus. Calculation of the accuracy rate is illustrated by the following 33 synonyms taken from the Chilin thesaurus. The synonyms 'identical' are *xiangtong* [相同]、*tong* [同]、*leitong* [雷同]、*tongyang* [同樣]、*tongyi* [同一]、*yiyang* [一樣]、*yilu* [一律]、*yise* [一色]、*yizhi* [一致]、*huayi* [劃一]、*dengtong* [等同]、*tongdeng* [同等]、*pingdeng* [平等]、*xiran* [翕然]、*yi* [一]、*ping* [平]、*yimuyiyang* [一模一樣]、*yirujiwang* [一如既往]、*qianpianyilu* [千篇一律]、*tianxiawuyayibanhei* [天下烏鴉一般黑]、*haowuerzhi* [毫無二致]、*ruchuyizhe* [如出一轍]、*wuyi* [無異]、*wuyiyu* [無異於]、*jundeng* [均等]、*dengtongyu* [等同於]、*yimashi* [一碼事]、*dengxiao* [等效]、*yiran* [亦然]、*cheping* [扯平]、*tongyi* [同義]、*banping* [扳平] and *dengwei* [等位]. Treating *xiangtong* [相同] 'identical' as the head word for synonym generation, we extracted a word list of those words with the ten highest similarity scores: *xiangtong* [相同] (1.0), *tongyang* [同樣] (0.55), *nanwang* [難忘] (0.45), *youli* [有力] (0.40), *chongfu* [重覆] (0.37), *jiang* [講] (0.36), *zhongxing* [中性] (0.36), *guanshang* [觀賞] (0.35), *jishen* [機身] (0.35), and *wanzheng* [完整] (0.34). In this case, only one synonym, *tongyang* [同樣], in accordance with the Chilin entries, was correctly extracted by our system. Thus, the accuracy rate was calculated as 1/33 = 3% in the results.

In the following two sections, we analyse the results according to semantic classes and relational richness respectively.

6.1 Performance analysis according to Chilin semantic classes

In the Chilin thesaurus, every synonym entry is tagged as one of 12 semantic classes: Person, Object, Time and Space, Abstract, Attribute, Action, Mentality, Activity, Phenomenon and State, Relation, Expletive, and Honorific. In this section, we first analyze the system performance (accuracy rate) of each semantic class. Table 5 shows the number of correctly extracted synonyms, the average number of dependency relations for all the entries in the semantic class, and the corresponding average accuracy rate:

Table 5. Accuracy rate for each semantic class

Class	# synonyms	Average # rel	Accuracy rate
Person	26	19	1.04%
Object	40	16	1.08%
Time and Space	28	19	2.15%
Abstract	124	18	2.55%

Attribute	89	17	1.05%
Action	28	20	2.00%
Mentality	46	22	2.05%
Activity	140	21	2.17%
Phenomenon and State	53	20	1.48%
Relation	52	23	3.78%
Expletive	31	21	1.97%
Honorific	0	-	0%

Note: The system extracted no synonyms for words from the Honorific class.

It was observed that the four groups having the lowest accuracy rate — Person, Object, Attribute and Honorific — were also the groups with fewer relations. It would be worth investigating why these words on average have fewer dependency relations with other words. Figure 5 shows the average number of relations of the 12 semantic classes and their corresponding accuracy rates. This correlation between accuracy and relations motivated us to analyze the synonym extraction performance according to the number of relations (relational richness), as discussed in the next section.

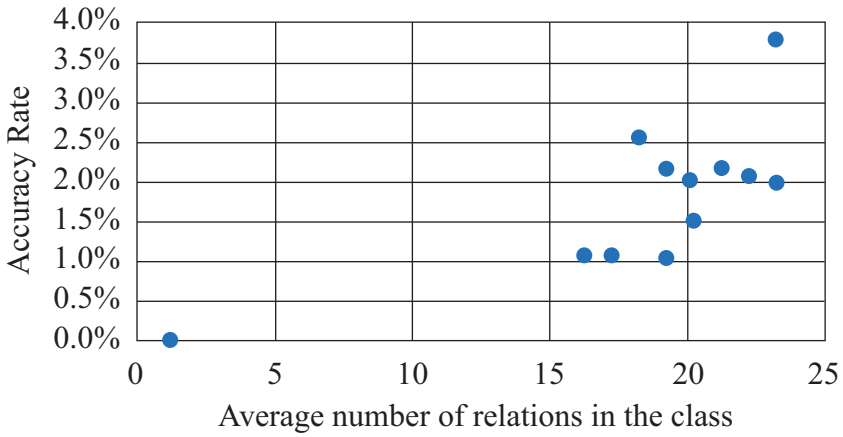


Figure 5. Accuracy rate and average number of relations for each semantic class

6.2 Performance analysis according to relational richness

For this second analysis, we divided the partially hit 5,650 Chilin synonym entries into seven groups according to the number of correctly extracted synonyms, and analyzed the number of relation types for each group. Table 6 shows the total number of relations, the total number of entries, and the average number of relations in each group:

Table 6. Average number of relation types grouped according to number of correct synonyms

# correct synonyms	total # rel	# entries	average # rel types
0	60,653	5,147	12
1	7,729	394	20
2	1,688	79	21
3	452	22	21
4	93	4	23
5	59	3	20
8	20	1	20
Total	10,041	503	20

From the previous figure, Figure 5, and Table 6, we observed that synonyms were extracted more correctly for target words having a larger variety of grammatical relations (at least 20) with other words.

From these two sections of analysis, we claim that this approach to extracting synonyms highly relies on the number of relations the target word has, as shown in Figure 6. In future, we will attempt to enlarge the corpus data or enrich dependency relations in order to improve the performance of this approach.

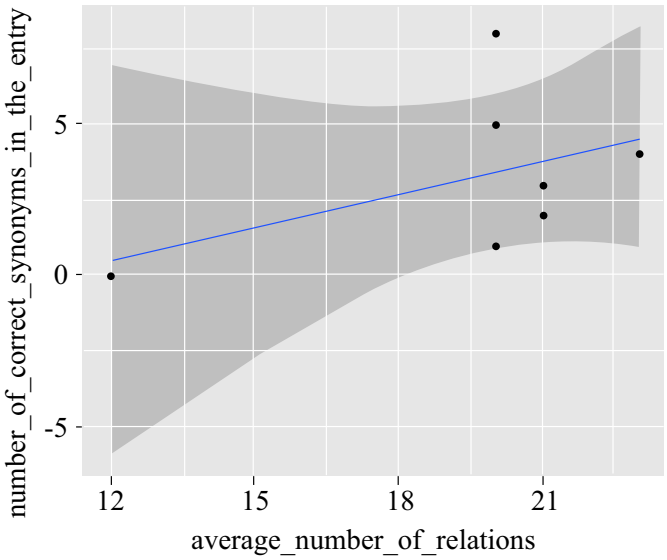


Figure 6. The correlation between the average number of relations in each group and the number of correct synonyms in the corresponding group

6.3 Evaluation of a case study

In addition to the two evaluations before, in terms of the automatic thesaurus generation function, we will attempt to compare the proposed resource with the manual results from a case study. We chose *gao* [搞] 'do' for our case study because it has already been studied using the same Sinica corpus (Cai 2014).

Cai manually annotated 418 instances of the pro-verb *gao* [搞] in the Sinica Corpus 3.0, and found 213 instances followed by an object noun. We searched the proposed resource for the same word, and found 211 instances of nouns as the direct object of *gao* [搞] in the parsed Sinica Corpus. By juxtaposing our results with those from Cai's study and WSE, we can see in Table 7 that our resources have provided a larger number of collocations in the case of *gao* [搞]'s object nouns.

Table 7. Comparison of the object noun counts for the verb *gao* [搞] 'do' in Cai (2014), WSE, and our proposed resource.

Object Noun	Cai	WSE	Proposed
社會主義	5	8	7
政治	5	12	9
共產主義	4	-	2
(封建)迷信	3	-	2
(大)躍進	3	-	2
花樣	2	-	2
(新)項目	2	-	2
(小)圈圈	2	-	3
臺獨	2	-	2
(十年)文革	2	-	2
這個領域	2	-	-
多媒體	2	-	-
運動	-	11	6
個	-	11	6
鬼	-	-	5
什麼	-	-	4
改革	-	-	4
工程	-	-	3

By further examining the data of *gao* [搞] + *yundong* [運動] as in (10), a collocation which is absent from other research, we found that most of the six cases in (10) are long-distance dependency relations such as *gao (yixie minzhu yundong)* [搞(一些民主)運動] 'do (some democratic) movement' which would take a lot more effort to notice when using human annotation.

(10) i.

suoyi tamen jingchang gao yixie minzhu yundong
所以 他們 經常 搞 一些 民主 運動
'so they frequently do some democratic movement'

ii.

gao suku yundong
搞 訴苦 運動
'do complaint movement'

iii.

xihuan gao yundong
喜歡 搞 運動
'like to do movement'

iv.

daxue li henduo youming youming de laoshi ye gao shehui yundong qu le
大學 裡 很多 有名 有名 的老師 也 搞 社會 運動 去了
'in the university many very famous teachers also went doing social movement'

v.

gedi de dang zuzhi anzhao gao yundong de guanli
各地 的 黨 組織 按照 搞 運動 的 慣例
'party organizations around the country according to the convention of doing movement'

vi.

haiyou yixie tongzhi bu zhuyi zai gongye fangmian gao da guimo de qunzhong yundong
還有 一些 同志 不 注意 在 工業 方面 搞 大 規模 的 群眾 運動
'there are still some comrades ignoring the doing large-scale mass movement in industry'

Figure 7. Snapshot of the dependency sketch function

From the above case study, it seems that the performance of our proposed resource is competitive with human annotation and WSE regarding relations with object nouns. However, performance for other relations still needs to be examined.

6.4 Web interface of dependency sketch

In anticipation of potential users such as TCSL (Teaching Chinese as a Second Language) teachers and linguists, a web interface was built for user-friendly access.[14]

[14] http://lopen.linguistics.ntu.edu.tw/near_synonym/sketch

Figure 7 shows a snapshot of the prototype. Like the classical WSE, our one-page dependency sketch shows the roles of collocates of the query word. Concerning dependency grammar, the relation between two words in a sentence can be labeled with the form of *relation* (*governor, dependent*). Thus, in Chinese language teaching, we identify the noun phrase in the *nsubj* relation (主語) as a dependent, the predicate (謂語) as governor, and the noun phrase in the *dobj* relation (賓語) as a dependent. For instance, the user can see that *gaoxing* [高興] 'happy' occurs in a given sentence as a compliment (補語) of *you* [有] 'have' and as an adverb (狀語) of *maigei* [賣給] 'sell to', as shown in Figure 7. Further provided are the syntactic functions of collocates, as shown at the bottom of Figure 7: *feichang* [非常] 'very' occurs as a compliment (補語) of *gaoxing* [高興] 'happy', and *yuangong* [員工] 'employee' as a subject (主語). The source code has been put on GitHub[15] for open access and further collaboration.

In language teaching, the difference between two near-synonyms is also important (Tsai 2011, Wang, Chen & Pan 2013). For instance, in Tsai's study she compared *bianli* [便利] and *fangbian* [方便] 'convenient' in terms of their syntactic functions and frequencies in Sinica Corpus 3.0, as Table 8 shows:

Table 8. Comparison of the syntactic functions and frequency of *bianli* [便利] and *fangbian* [方便] 'convenient'

Syntactic Function	Predicate	Compliment	Adverb	Attributive	Nominalization
便利	66	1	3	63	40
173	38.15%	0.58%	1.73%	36.42%	23.12%
方便	470	2	14	36	69
591	79.53%	0.34%	2.37%	6.09%	11.67%

Concerning this point, we have also designed a function to sketch differences between near-synonyms.[16] It compares two near-synonyms such as *gaoxing* [高興] and *quaile* [快樂] 'happy' from several aspects. First of all, this function provides basic definitions of the two near-synonyms from the gloss in Chinese Wordnet. It lists example sentences for the two synonyms in a usage having the same dependency roles. Five collocates having the same dependency relations with the two synonyms are shown in corresponding columns if available. It then extracts one example sentence for each synonym to illustrate how the synonyms are used in different dependency

[15] http://github.com/mhshih/near_synonym
[16] http://lopen.linguistics.ntu.edu.tw/near_synonym/near_synonym

relations. It also shows five collocates with dependency relations to one synonym that do not occur with the other synonym. Finally, it offers a cloze exercise — a sentence with a slot for students to fill with the correct synonym.

Take the two near-synonyms *gaoxing* [高興] and *kuaile* [快樂] 'happy' for example. When the user queries these two words in the web interface, this function compares the two synonyms from seven aspects. In Figure 8, it first shows the Wordnet definitions for *gaoxing* [高興] and *kuaile* [快樂]: *xingrong yinwei teding shijian er gandao lianghao de qingxu* [形容因為特定事件而感到良好的情緒] 'to describe the feeling of a good mood because of a specific event' and *xingrong gandao xiangshou he gaoxing de* [形容感到享受和高興的] 'to describe the feeling of enjoyment and pleasue' respectively. Then two example sentences are given where both *gaoxing* [高興] and *kuaile* [快樂] serve as the subject (主語) in one of the sentences, and the other sentence where *gaoxing* [高興] is used exclusively as an adverbial modifier (補語) in this near-synonym pair of 高興–快樂. Finally, an exercise sentence *mi laoda hen* ____ *de yao mai gei ta* [米老大很____地要賣給他] 'Boss Mi is very ____ to sell it to him' is provided for students to complete with either or both of the near-synonyms.

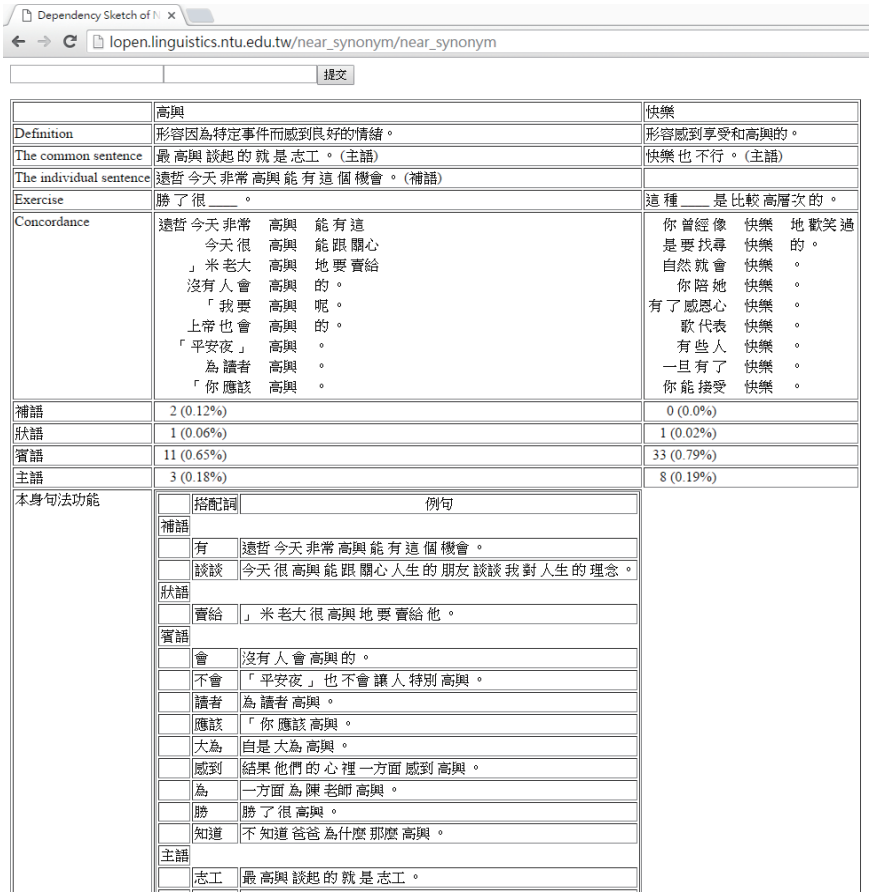


Figure 8. Snapshot of the function for the sketch of differences between two near-synonyms *gaoxing* [高興] and *kuaile* [快樂] 'happy'

Aiming to provide a comprehensive view of lexical behaviors, the above two functions, word sketch and near-synonym differences, are embedded into a project called Chinese WordMap[17], as shown in Figure 9.

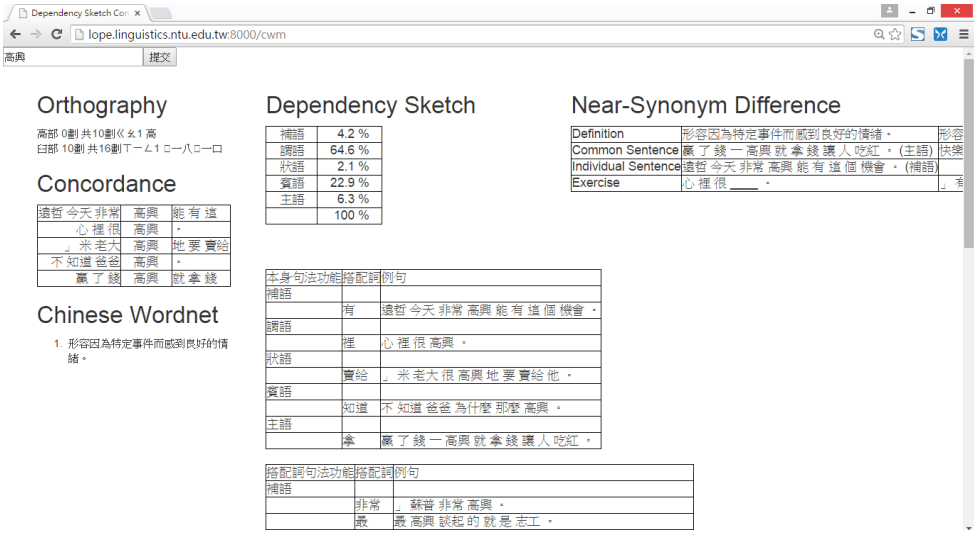


Figure 9. Snapshot of Chinese WordMap

6.5 Chinese dependency data API

We also released the processed dependency sketch data of the balanced texts as the Chinese dependency data API.[18] Although Universal Dependency data from the Stanford NLP group (de Marneffe et al. 2014) has been available for various languages including English, French, German, Italian and Spanish, Chinese data is still being developed.[19] Although the Chinese Dependency Treebank 1.0 (Che, Li & Liu 2012) comprised of about 50,000 newswire sentences is available on the Linguistic Data Consortium[20] website, the data are all provided in the format of CoNLL-X, a shared task of multi-lingual dependency parsings which intends to represent every parsed sentence in ten fields (Buchholz & Marsi 2006), but this is not so appropriate for other NLP training and applications. The ten fields used in the format are ID, FORM, LEMMA, CPOSTAG, POSTAG, FEATS, HEAD, DEPREL, PHEAD, and PDEPREL, as illustrated in the following representation of the parsed sentence "*Aerjiliya quanguo guodu weiyuanhui zhuxi bensalahe 7 ri xuanbu* [阿爾及

[17] http://lope.linguistics.ntu.edu.tw:8000/cwm
[18] http://lopen.linguistics.ntu.edu.tw/near_synonym/sketch/高興
[19] http://universaldependencies.github.io/docs
[20] http://catalog.ldc.upenn.edu/LDC2012T05

利亞全國過渡委員會主席本薩拉赫７日宣布] ‘Chairman of the National Transitional Council of Algeria Bensalah announced on 7th’ ”, here converted from the simplified Chinese characters used in the Chinese Dependency Treebank. This illustrated representation means that the FORM 宣布 in TOKEN ID 9 is the HEAD of the FORM in TOKEN ID 0 (the sentence root), the FORM本薩拉赫 is the SUBJECT of the FORM in TOKEN ID 9 (宣布), and so on.

ID	FROM	LEMMA	CPOSTAG	POSTAG	FEATS	HEAD	DEPREL	PHEAD	PDERREL
1	阿爾及利亞	–	Ns	–	–	4	ATT	–	–
2	全國	–	n	–	–	4	ATT	–	–
3	過渡	–	n	–	–	4	ATT	–	–
4	委員會	–	n	–	–	5	ATT	–	–
5	主席	–	n	–	–	6	ATT	–	–
6	本薩拉赫	–	nh	–	–	9	SBV	–	–
7	７	–	m	–	–	8	ATT	–	–
8	日	–	q	–	–	9	ADV	–	–
9	宣布	–	v	–	–	0	HED	–	–

Figure 10. Representation of a parsed sentence in CoNLL-X format

While the above texts mostly come from news or other web media and focus on the representation of parsed sentences, our dependency data API focuses on the frequency of dependency relations for each word, as illustrated in the following case for *gaoxing* [高興] ‘happy’:

(11) {“賓語”: [11, {“知道”: [“不 知道 爸爸 為什麼 那麼 高興 。”]}], “狀語”: [1, {“賣給”: [“」 米 老大 很 高興 地 要 賣給 他 。”]}], “補語”: [2, {“有”: [“遠哲 今天 非常 高興 能 有 這 個 機會 。”]}], “主語”: [34, {“裡”: [“心 裡 很 高興 。”]}]}

Our data API provides the frequencies of the dependency roles, such as 賓語 (*dobj*) for *gaoxing* [高興] ‘happy’, along with the collocates such as *zhidao* [知道] ‘know’, and the corresponding context sentence shown in (12), which is different from the above two resources and more suitable for NLP training and applications.

(12) *Bu zhidao baba weisheme name gaoxing*
不 知道 爸爸 為什麼 那麼 高興 。
Not know father why so happy .
‘Do not know why the father was so happy.’

7. Conclusion

A word sketch is a corpus-based automatic summary of a word's grammatical and collocational behavior. Based on the handcrafted finite-state sketch grammar over a POS-tagged corpus, the WSE system can identify collocates in grammatical relations with a target word. However, the grammar engineering is time-consuming and requires the involvement of many experts. In this paper, we propose an alternative by leveraging an existing dependency parser upon a tag-removed balanced corpus. The results were evaluated based on comparison with the semi-manually constructed thesaurus, Chilin.

From the viewpoint of application, this paper serves as the first attempt to create an open-sourced word sketch-like corpus profiling system for Chinese linguistics and TCSL. The proposed method is pipelined and can be applied to user-created corpora. The extracted relation triples $<w_1,R,w_2>$ can be used to enrich our on-going Chinese DeepLEX database. Future works including the exploration of other dependency parsing algorithms, the incorporation of advanced statistics to single out salient collocations, and the development of an open evaluation platform for further improvement of the resource are in progress.

References

Ambati, Bharat, Siva Reddy, and Adam Kilgarriff. 2012. Word sketches for Turkish. Paper presented at the 8th International Conference on Language Resources and Evaluation (LREC-2012), Istanbul Convention and Exhibition Centre, Istanbul, Turkey.

Androutsopoulos, Ion, and Prodromos Malakasiotis. 2010. A survey of paraphrasing and textual entailment methods. *Journal of Artificial Intelligence Research* 38.1: 135-187.

Baisa, Vit, and Vit Suchomel. 2014. SkELL: Web interface for English language learning. *Proceedings of Recent Advances in Slavonic Natural Language Processing*, ed. by Ales Horak and Pavel Rychly, 63-70. Brno: Tribun.

Buchholz, Sabine, and Erwin Marsi. 2006. CoNLL-X shared task on multilingual dependency parsing. Paper presented at the Tenth Conference on Computational Natural Language Learning (CoNLL-X), New York City, USA.

Cai, Chen. 2014. The semantic prosody of pro-verb *gao* "*do*" in cross-strait varieties between Modern Chinese. *Journal of Chinese Language Teaching* 11.3:91-110.

Chang, Pi-Chuan, Huihsin Tseng, Dan Jurafsky, and Christopher Manning. 2009.

Discriminative reordering with Chinese grammatical relations features. Paper presented at the Third Workshop on Syntax and Structure in Statistical Translation (SSST-3), Boulder, Colorado.

Chao, August, and Siaw-Fong Chung. 2013. A definition-based shared-concept extraction within groups of Chinese synonyms: A study utilizing the extended Chinese synonym forest. *Computational Linguistics and Chinese Language Processing* 18.2:35-56.

Che, Wanxiang, Zhenghua Li, and Ting Liu. 2012. *Chinese Dependency Treebank 1.0.* Philadelphia: Linguistic Data Consortium.

Chen, Keh-Jiann, Chu-Ren Huang, Li-Ping Chang, and Hui-Li Hsu. 1996. Sinica Corpus: Design methodology for balanced corpora. Paper presented at the The 11th Pacific Asia Conference on Language, Information and Computation (PACLIC-11), Kyung Hee University, Seoul.

Collins, Michael. 1999. Head-Driven Statistical Models for Natural Language Parsing. Doctoral dissertation, University of Pennsylvania, USA.

de Marneffe, Marie-Catherine, Timothy Dozat, Natalia Silveira, Katri Haverinen, Filip Ginter, Joakim Nivre, and Christopher D. Manning. 2014. Universal Stanford Dependencies: A cross-linguistic typology. Paper presented at the Ninth International Conference on Language Resources and Evaluation (LREC-2014), Reykjavik, Iceland.

Evert, Stefan, and Andrew Hardie. 2011. Twenty-first century Corpus Workbench: Updating a query architecture for the new millennium. Paper presented at the Corpus Linguistics 2011 Conference, Birmingham, UK.

Hong, Jia-Fei, and Chu-Ren Huang. 2006. Using Chinese Gigaword Corpus and Chinese Word Sketch in linguistic research. Paper presented at the The 20th Pacific Asia Conference on Language, Information and Computation (PACLIC-20), Wuhan, China.

Huang, Chu-Ren, Adam Kilgarriff, Yiching Wu, Chih-Ming Chiu, Simon Smith, Pavel Rychly, Ming-Hong Bai, and Keh-Jiann Chen. 2005. Chinese Sketch Engine and the extraction of grammatical collocations. Paper presented at the Fourth SIGHAN Workshop on Chinese Language Processing, Jeju Island, Korea.

Jakubíček, Miloš, Adam Kilgarriff, Vojtěch Kovář, Pavel Rychlý, and Vít Suchomel. 2013. The TenTen Corpus Family. Paper presented at the Seventh International Corpus Linguistics Conference (CL2013), UCREL Research Centre, Lancaster.

Kilgarriff, Adam. 2007. Using corpora in language learning: The Sketch Engine. *Optimizing the role of language in Technology-Enhanced Learning* 22:21-23.

Kilgarriff, Adam, and Iztok Kosem. 2012. Corpus tools for lexicographers. *Electronic Lexicography*, ed. by Sylviane Granger and Magali Paquot, 31-55. Oxford: Oxford

University Press.

Kilgarriff, Adam, Pavel Rychly, Pavel Smrz, and David Tugwell. 2004. The Sketch Engine. *Proceedings of EURALEX*, ed. by Geoffrey Willams and Sandra Vessier, 105-116. Lorient, France.

Kilgarriff, Adam, Vít Baisa, Jan Bušta, Miloš Jakubíček, Vojtěch Kovář, Jan Michelfeit, Pavel Rychlý, and Vít Suchomel. 2014. The Sketch Engine: Ten years on. *Lexicography: Journal of ASIALEX* 1.1:7-36.

Kilgarriff, Adam, Vojtěch Kovář, Simon Krek, Irena Srdanović, and Carole Tiberius. 2010. A quantitative evaluation of word sketches. Paper presented at the 14th EURALEX International Congress, Leeuwarden, The Netherlands.

Klein, Dan, and Christopher Manning. 2003. Accurate unlexicalized parsing. Paper presented at the 41st Annual Meeting of the Association for Computational Linguistics, Sapporo, Japan.

Krek, Simon and Kaja Dobrovoljc. 2014. Sketch Grammar: RegEx-over-POS or dependency parser? A comparison of two MWE extraction methods. Paper presented at the PARSEME 2nd general meeting Athens, Greece.

Levy, Roger, and Christopher Manning. 2003. Is it harder to parse Chinese, or the Chinese Treebank? Paper presented at the 41st Annual Meeting of the Association for Computational Linguistics, Sapporo, Japan.

Meena, Arun, and T. V. Prabhakar. 2007. Sentence level sentiment analysis in the presence of conjuncts using linguistic analysis. *Lecture Notes in Computer Science Volume 4425: Advances in Information Retrieval - 29th European Conference on IR Research* ed. by Amati Giambattista, Carpineto Claudio and Romano Giovanni 573-580. Berlin: Springer.

Reddy, Siva, Ioannis Klapaftis, Diana McCarthy, and Suresh Manandhar. 2011. Dynamic and static prototype vectors for semantic composition. Paper presented at the 5th International Joint Conference on Natural Language Processing (IJCNLP-2011), Chiang Mai, Thailand.

Tesnière, Lucien. And Sylvain Kahane. 2015. *Elements of Structural Syntax*. Amsterdam: John Benjamins.

Tsai, Mei-chih. 2011. "Convenient" during the process or as a result-event structure of synonymous stative verbs in TCSL. *Journal of Chinese Language Teaching* 8.3:1-22.

Wang, Yi-Ting, Hao-Jan Chen, and I-Ting Pan. 2013. Investigation and analysis of Chinese synonymous verbs based on the Chinese learner corpus: Example of *bang*, *bang-zhu*, *bang-mang* and *bian*, *bian-de*, *bian-cheng*. *Journal of Chinese Language Teaching* 10.3:41-64.

Xia, Fei, and Martha Palmer. 2001. Converting dependency structures to phrase

structures. Paper presented at the First International Conference on Human Language Technology Research, San Diego, California.

[Received October 12, 2015; revised February 5, 2016; accepted April 26, 2016]

Gradute Institute of Linguistics
National Taiwan University
Taipei, TAIWAN
Meng-Hsien Shih: simon.xian@gmail.com
Shu-Kai Hsieh: shukaihsieh@ntu.edu.tw

華語教學的字詞依存關係描繪

施孟賢　　謝舒凱
國立臺灣大學

本文描述自動建立語言學習資源的方法，藉由依存剖析器對文本的分析，我們可以描繪中文字詞間的語法關係。與先前研究相比，本資源可以提供更周延的字詞用法，例如各式各樣的修飾關係，這在語言教學上將有所應用。雖然其他語言的資源也試圖藉由剖析文本來描繪字詞關係，然而我們尚未在中文資源裡看到針對自訂文本來描繪字詞的語言資源，因此我們提出此方法並評估其產生同義詞的功能。我們並針對語言學習開放分析結果的介面，相信對中文語言學和教學有所助益。

關鍵詞：依存關係、計算詞彙學、同義詞典、語料庫、語言學習