SimpleNLG-ZH: a Linguistic Realisation Engine for Mandarin

Guanyi Chen[1], Kees van Deemter[12], Chenghua Lin[2]
[1]Department of Information and Computing Sciences, Utrecht University
[2]Department of Computing Science, University of Aberdeen
{g.chen, c.j.vandeemter}@uu.nl, chenghua.lin@abdn.ac.uk

Abstract

We introduce SimpleNLG-ZH, a realisation engine for Mandarin that follows the software design paradigm of SimpleNLG (Gatt and Reiter, 2009). We explain the core grammar (morphology and syntax) and the lexicon of SimpleNLG-ZH, which is very different from English and other languages for which SimpleNLG engines have been built. The system was evaluated by regenerating expressions from a body of test sentences and a corpus of human-authored expressions. Human evaluation was conducted to estimate the quality of regenerated sentences.

1 Introduction

A classic natural language generation (NLG) system (Reiter and Dale, 2000) is a pipeline consisting of document planning, sentence planning and surface realisation (in that order). Surface realisation maps information produced by earlier components to well-formed output strings in the target language. A (surface) realiser employs language-specific morpho-syntactic constraints to achieve proper word ordering, inflection, and selection of function words. Different types of realisers exist (Gatt and Krahmer, 2018). Unlike approaches that aim primarily for linguistic depth and coverage (White et al., 2007), realisers in the SimpleNLG tradition aim primarily for ease of use and extendibility (Gatt and Reiter, 2009), and have become the realisation method of choice in many practical NLG applications, such as BabyTalk (Portet et al., 2009) and Absum (Lapalme, 2013).

SimpleNLG, as a human-crafted grammar-based realisation engine, performs linearisation and morphological inflection. Another realisation strategy uses statistical methods for acquiring probabilistic grammar from large corpora. For example, OpenCCG (White et al., 2007) built a grammar bank based on Combinatorial Categorial Grammar, extracted from the Penn Treebank (Marcus et al., 1993). When realising, OpenCCG applies a chart-based algorithm to generate all possible surface forms, which are then re-ranked by language models. Such an approach tends to have broader coverage, but less controllability and extendibility, which may explain why SimpleNLG is more popular in practical applications.

To date, the original English SimpleNLG has been adapted to German (Bollmann, 2011), French (Vaudry and Lapalme, 2013), Portuguese (De Oliveira and Sripada, 2014), Italian (Mazzei et al., 2016), Spanish (Soto et al., 2017), Filipino (Ong et al., 2011) and Telugu (Dokkara et al., 2015). There is no such adaptation work yet for Sino-Tibetan languages, whose morpho-syntactic structure is very different from the above languages. Mandarin, a Sino-Tibetan language with nearly 1 billion first-language speakers, offers huge opportunities for natural language generation, yet only a limited amount of work has focused on Mandarin realisation. KPML, a large-scale multilingual generation and development, supports limited sentence structures in Mandarin (Yang and Bateman, 2009). He et al. (2009) introduced a data-driven generator, with dependency trees as input. They used divide-and-conquer to break the dependency tree into sub-trees, realising each sub-tree using a log-linear model recursively. However, their system needs a large amount of fully inflected dependency trees as training data.

This paper describes a realisation engine fol-

Proceedings of The 11th International Natural Language Generation Conference, pages 57–66,
Tilburg, The Netherlands, November 5-8, 2018.

lowing the design principles of SimpleNLG, i.e., keeping a clear separation between morphological and syntactic operations (Gatt and Reiter, 2009). Although we took existing SimpleNLG systems as a source of inspiration, the system is, in many ways, a re-design[1]. For example, Mandarin, as a highly *analytical* language, needs far fewer morphological operations but many more syntactic constraints than English (Huang et al., 2009). SimpleNLG-ZH[2] ("Zhongwen" is Mandarin for "Chinese") was firstly built as a realiser for generating referring expressions in Mandarin (van Deemter et al., 2017; Van Deemter, 2016) which are mostly noun phrases together with simple verb phrases, and then extended to coverage other constructions and phenomena in Mandarin. It was developed as an adaptation from V4.4.8 of the original SimpleNLG[3] (SimpleNLG-EN). We show that SimpleNLG-ZH has wide coverage on test-sentences, and on the human authored corpus MTuna (van Deemter et al., 2017) as well.

2 The idea of SimpleNLG

SimpleNLG is a realisation engine designed for practical use. The input format of SimpleNLG is similar to a simplified dependency tree where the user should determine the specifiers, modifiers and complements of each input phrase using a set of features. SimpleNLG encodes different constraints, regarding lexicon, morphology, syntax and orthography, as a feature set (combining the features from the input) and passes the resulting structure onto the next stage. Figure 1 shows examples of an input for SimpleNLG-EN and SimpleNLG-ZH, respectively. To construct a sentence using SimpleNLG, we need to establish a verb phrase object and set its object(s) and subject.

SimpleNLG follows good software engineering design principles, clearly separating the modules for lexical and syntactic operations. The lexical component provides interfaces that handle the lexical features and apply morphological rules. Vital features such as `person`, `number` and `tense` are appended to target constituents or words for further realisation processes. The syntactic component takes over at the phrase and clause level, and provides `Java` classes for each phrasal subtype (`PhraseSpecs`), where `SPhraseSpec` stands for the class that model clauses.

SimpleNLG-EN offers significant coverage of English morphology and syntax, and provides easy-to-use APIs with which the realisation process is programmatically controllable. It provides a well established lexicon, the repository of the relevant items and their properties. The lexicon was constructed from the NIH specialist lexicon[4], which contains more than 300,000 entries. Each lexical entry was tagged with detailed lexical features as initial features of words. Simple shallow semantic features, like `COLOUR` and `QUANTITATIVE`, are appended for deciding word order.

3 Morphology

Morphology in Mandarin is usually thought to be extremely simple (Jensen, 1990). Packard (2000) has challenged this view, arguing that more morphological operations are involved in the construction of Chinese words than is usually thought. However, key mechanisms such as subject-verb agreement (which SimpleNLG-EN treated as part of morphology operations) are absent from Mandarin. We have therefore sided with mainstream linguistic opinion and kept our morphology component relatively simple. We use only two main rules for morphology: mapping pronouns to their surface forms and appending the collective marker "们" (mén).

3.1 Pronoun

Realising the surface forms of pronouns in SimpleNLG-ZH is similar to SimpleNLG-EN in its use of the features `gender` (masculine, feminine or neuter), `number` (singular or plural), and `person` (first, second or third). However, written Mandarin has different *third person plural* forms for all three different genders, i.e., "他们" (masculine), "她们" (feminine) and "它们" (neuter) (all of them have the same pronunciation: *tāmén*) rather than the one plural form *they* in English.

3.2 Collective Marker

In Mandarin, to say how many entities there are in a set, *classifiers* must be used. This is typically done in a *number phrase* of the form [number + classifier + noun], for instance "一把椅子" (yì bǎ

[1] The German, Portuguese, and Spanish SimpleNLG systems copied many features from the one for English (in the case of German) or French (in the other two cases).

[2] The software is available at: https://github.com/a-quei/simplenlg-zh.

[3] https://github.com/simplenlg/

[4] https://github.com/simplenlg/simplenlg/blob/master/src/main/java/simplenlg/lexicon/default-lexicon.xml

```
Phrase s1 = new SPhraseSpec('leave');
s1.setTense(PAST);
s1.setObject(new NPPhraseSpec('the', 'house'));
Phrase s2 = new StringPhraseSpec('the boys');
s1.setSubject(s2);
```

```
Phrase s1 = new SPhraseSpec('离开');
s1.setParticle('了');
s1.setObject(new NPPhraseSpec('房子'));
Phrase s2 = new NPPhraseSpec('男孩');
s1.setSubject(s2);
```

Figure 1: Input code for generating the sentence "男孩离开了房子" (nánhái líkāile fángzi; *The boys left the house*) using SimpleNLG-EN (left) and SimpleNLG-ZH (right).

yǐzi; *a chair*), "两张桌子" (liǎng zhāng zhuōzi; *two tables*). Since number phrases are typically used referentially (not as quantifiers), they have generally been regarded as indefinite expressions, and these cannot be placed in subject or topic position in Mandarin (Huang et al., 2009).

Unlike English, Mandarin bare nouns and number phrases with numbers larger than 1 can express plural meaning without the help of inflected plural markers. The morpheme "们" in plural nouns serves as a "collective" marker rather than a traditionally plural marker (Li, 2006); here a "plurality" is a number of individuals, whereas a "collective" is a group (of individuals) as a whole. Under that definition, adding a morpheme "们" makes a nominal phrase definite, which results the morpheme "们" incompatible with a number phrases, so "们" cannot co-occur with number phrases. For example, the phrase "三个人们" (sān gè rénmén; *three people*) is not acceptable in Mandarin. Note that the rules discussed above do not apply to pronouns which follow the rules defined in §3.1.

It is hard to determine automatically whether a user wants to talk about a number of individuals or about a group as a whole. Moreover, "们" is always only optional. Therefore, in SimpleNLG-ZH, "们" is only added if the feature `MEN` is set to true. In addition, the system will refuse to add a "们" to a number phrase. The way of constructing number phrases is discussed in §4.

4 Syntax

The syntax module inherits the basic structure of SimpleNLG-EN, dividing the syntactic operations into processors that handle noun phrases, adjective phrases, verb phrases, verb phrases, and clauses. Each processor is enriched based on the grammar of Mandarin.

4.1 Noun Phrase

The Noun Phrase (NP) module is the most complex phrase module in SimpleNLG-ZH. Each noun phrase in SimpleNLG-ZH contains multiple specifiers, pre-modifiers, post-modifiers, complements, and a head noun.

4.1.1 Number Phrase

Each number phrase is constructed by a number, a classifier and a head noun; both the numeral and the classifier function as *specifiers* of the NP (for more about specifiers, please see §4.1.2).

As Number Phrases are very common in Mandarin, we designed a new constructor specifically for them. For instance, the number phrase "一本书" (yì běn shū; *a book*) can be constructed using this input:

```
NPPhraseSpec book = this.
   phraseFactory.createNounPhrase
   ("一", "本", "书");
```

The choice of classifiers depends mainly on the head noun. Additionally, for a given noun, the choice of classifiers may depend on its meaning. For example, the classifier of "房子" (fángzi; *house*) can be "座", "幢", "间", and many other possible classifiers based on the size or the shape of the house. The current SimpleNLG-ZH requires classifiers to be specified "by hand". By introducing a language model in the future, this process might be automated.

4.1.2 Specifier

SimpleNLG-ZH allows multiple specifiers (compared to a single specifier in SimpleNLG-EN) within one NP. For example, a number phrase needs two specifiers: a numeral and a classifier. All the following categories can be placed in specifier position: pronouns (with or without the collective marker "men"), proper names, classifiers, numerals and demonstratives. These specifiers appear in the following order (the $A > B$ means A should appear before B): `proper name` > `pronoun` > `demonstrative` > `numeral` > `classifier`. The decision of whether or not to realize each of these specifiers is subject to a number of constraints (Huang et al., 2009).

1. Suppose the input specification asks for a pronoun in the specifier position. This pronoun must have a collective marker except in a structure that includes [demonstrative/numeral + classifier] For instance, “他们学生” (tāmén xuéshēng; *them students*) contains the collective marker, but “他一个学生” (tā yígè xuéshēng; *them students*) does not;
2. Proper names in specifier position can only be realised if the structure includes [pronoun + numeral + classifier], [demonstrative + classifier] or [demonstrative + numeral + classifier]: “张三那个学生” (zhāngsān nàgè xuéshēng; *the student called Zhangsan*);
3. A demonstrative or a numeral will only be realised if there is a classifier in the same NP and vise versa: “(那/一)个学生” (nà/yí gè xuéshēng; *that/a student*).

As discussed in §3.2, number phrases are often seen as indefinite phrase but not always. When they are for quantification they can be placed in the subject/topic position. Therefore, SimpleNLG-ZH permits a number phrase in the subject/topic position, e.g., the sentence “三个人吃两块蛋糕” (sān gè rén chī liǎng kuài dàngāo; *three people eat two cakes*)

For nouns (including bare nouns, pronouns and proper nouns), the feature `possessive` is also realised in the specifier position: SimpleNLG-ZH adds a particle “的” (de) as an associative marker after the noun.

4.1.3 Localiser

Localisers (corresponding to English words such as “on”, “above”, etc.) form a special syntactic category. They are used in *location phrases*, which is a particular type of preposition phrases. The location information in a location phrase is expressed in the localiser rather than the head preposition, for example: [$_{PP}$ 在 [$_{NP}$ 桌子 上]] (zài zhuōzi shàng; *on the table*). The localiser “上” (*on*) works as a supplement of the noun phrase in the proposition phrase (i.e., location phrase).

In SimpleNLG-ZH, the localiser itself is defined as a normal noun with a lexical feature `LOCATIVE` in the lexicon. When constructing a location phrase, if the localiser is a disyllabic word, such as “上面” (shàngmiàn), then a particle “的” is inserted before the localiser to construct the phrase: “在桌子的上面” (zài zhuōzi shàngmiàn; *on the table*). However, if such a prepositional phrase works as a pre-modifier of another noun, then that inserted particle will be disregarded, for example: “在桌子上面的书” (zài zhuōzi shàngmiàn de shū; *the book on the table*).

4.1.4 Pre-modifier

SimpleNLG-EN handles the orders of multiple pre-modifiers based on their meanings, where the meanings are acquired from a huge lexicon that contains a series of tags (e.g., `COLOUR`, `QUANTITATIVE`) indicating the meaning of words. It adds pre-modifiers in the order of quantitative adjectives, colour adjectives, classifying adjectives and nouns. For SimpleNLG-ZH, more categories of words can be placed in the pre-modifier position, other than just adjectives and nouns. It performs re-ordering based on pre-modifiers’ part-of-speech and lexical features set by the users.

Our system handles two different types of adjectives, namely, normal adjectives and non-predicate adjectives. For normal adjectives, the system will automatically add a “的” (de) between the adjectives and the head noun, such as “绿的椅子” (lǜ de yǐzi; *green chair*). “的” can be omitted by setting the feature `NO_DE` to `TRUE`, which results in the phrase “绿椅子“ (*green chair*). Non-predicate adjectives, in contrast to normal adjectives, are a special type of adjectives that cannot function as predicate on their own (e.g., “男” (ná; *male*) and “女” (nǚ; *female*)), in which the particle “的” (de) is always omitted. Thus, the particle “的” will not be appended if the adjective is non-predicate, such as “男人” (nánrén; *man*). The feature is set based on the information of the lexicon loaded into SimpleNLG-ZH (details see §5).

Nouns and noun phrases, as pre-modifiers, can play two different roles: they can be concatenated with the head noun to construct a compound noun: for example, “大 学 教 育” (dàxuē jiàoyù; *university education*); or, they can be connected by means of a particle “的”, which works as an associative marker: for example, “黑头发的人” (hēitóufà de rén; *the man with black hair*). To construct the latter, the feature `ASSOCIATIVE` should be set to `TRUE`. The order of the pre-modifiers is `localisers > verbs/clauses > adjectives with de > nouns with associative marker > adjectives without de > non-predicate adjectives > nouns`.

4.2 Adjective Phrase

Adjective phrases in Mandarin differ from those in the languages for which previous SimpleNLG engines were built. Most adjectives in Mandarin can act as the predicate of a clause without the help of a copula verb (see below). Such adjectives are called predicate adjectives.

4.2.1 Predicate Adjective

Although adjectives can act as predicates, it is necessary to distinguish them from verbs (Huang et al., 2009). We implemented realisation of a clause like "他很高" (tā hěngāo; *he is very tall*) by specifying an empty copula. This is achieved by creating a new constructor which accepts a subject noun and a predicate adjective.

Predicate adjectives in SimpleNLG-ZH also accept negative words and modal words. For example, the sentence "他应该不高" (tā yīnggāi bùgāo; *he couldn't be tall*) has both a negative word "不", and a modal word "应该".

4.2.2 Non-predicate Adjective

As discussed in §4.1.4, non-predicate adjectives always omit the particle "的" between the adjective and the head noun. However, when a non-predicate adjective functions as a predicate (with the help of a copula), such as "他是男的" (tā shì nánde; *he is a man*), the copula "是" (shì) and the particle "的" (de) are obligatory (Paul, 2010).

4.2.3 "比" construction

In English, degree adjectives have comparative and superlative degrees, whose realisation is implemented in the morphology processor. In Mandarin, realisation is performed by modifying the syntax. The superlative degree is realised by adding an adverb pre-modifier "最" (zuì; *most*); the comparative through the "比" construction.

SimpleNLG-ZH implements the "比" (bǐ) construction as a prepositional phrase. For example, for the sentence "他比小明高" (tā bǐ xiǎomíng gāo; *he is taller than xiaoming*), the word "比" (bǐ) itself is seen as the head of a preposition phrase, which is a pre-modifier of a adjective phrase. Such a construction (i.e., as an adjective phrase), can act as the pre-modifier of a noun phrase, for example, "他们班没有比他更高的人" (tāmén bān méiyǒu bǐ tā gènggāode rén; *none of his classmates is taller than he*). Note that the head of this noun phrase can be omitted, but the particle "的" (de) should be maintained as a sentence-final marker, i.e. "他们班没有比他更高的" (tāmén bān méiyǒu bǐ tā gènggāode).

4.3 Verb Phrase

4.3.1 Pre-modifier and Post-modifier

Verb phrases can contain the associative markers "得" and "地". The latter is appended to the pre-modifier if it is disyllabic, for example, "快速地跑" (kuàisù de pǎo; *fast run*). If the pre-modifier is monosyllabic, "快跑" (kuàipǎo) is constructed instead, with the particle "地" (de) disregarded. The particle "得" (de) connects head verbs with their complements: "跑得快" (pǎodekuài; *running fast*).

4.3.2 Aspect

KPML (Yang and Bateman, 2009) used templates with particles like "过", "了" or "着" (zhe)to model aspect. However, KPML's coverage of language variation is limited because it uses a limited number of templates. Since aspect in Mandarin is realised using post-verbal or post-clause particles, we took a more flexible strategy that enables users to add particles based on their need.

Particles can be in two positions: post-verbal and post-clausal. In "他吃着饭" (tā chīzhe fàn; *he is eating*), the particle "着" (zhe), which expresses the present continuous tense, is appended to a `VPPhraseSpec` object. Similarly, the class `SPhraseSpec`, which represents a clause, has the capability to append a particle to its end. For example, in "他吃饭了" (tā chī fànle; *he has eaten*), the particle "了" is appended to the clause "他吃饭" (tāchīfàn; *he eats*).

4.4 Clause

At the Clause level, apart from the issues related to negative and interrogative sentences inherited from SimpleNLG-EN, we considered "把" (bǎ) and "被" (bèi) constructions which are two common constructions in Mandarin. We also discuss how topicalised sentences are realised using SimpleNLG-ZH.

4.4.1 Negative Sentence

Negative sentences in SimpleNLG-ZH are realised by inserting negative words before the predicate verb (or the predicate) and after a modal word. For example, the negation of "他应该去上学" (tā yīnggāi qù shàngxué; *he should go to school*) is the sentence with an inserted negative word "不" (bù; *not*) before "去" (qù; *go*) and after the modal

word “应该” (yīngāi; *should*): “他应该不去上学” (tā yīnggāi bū qù shàngxué; *he should haven't gone to school*). SimpleNLG-ZH can also realise negative modal by viewing the negative modal as a merged word, much like *haven't* or *shouldn't* in English (Xu, 1997). For example,“他不应该去上学” (tā bū yīnggāi qù shàngxué; *he should not go to school*).

In addition, Mandarin has a number of different negative words, selected based on the head verb. For example, applied to the sentence “他有椅子” (tā yǒu yǐzi; *he has chairs*), instead of using “不” (bù), the word “没” (méi) should be used: “他没有椅子” (tā mēiyǒu yǐzi; *he doesn't have a chair*). SimpleNLG-ZH allows users to specify by hand what negation word should be chosen in a specific case by using the feature `negative_word`, thus overruling the system's default choice.

4.4.2 “把” Construction

The “把” construction is a common seen and useful structure for focusing on the result or influence of an action, which is not exist in English. For example, considering the sentence, “他把小明重重地打” (tā bǎ xiǎomíng zhòngzhòng de dǎ; *he beat xiaoming heavily*), with the “把” construction, the influence of “打” (dǎ; beat) is highlighted. The natural phrase order of this example is: “他重重地打小明” (tā zhòngzhòng de dǎ xiǎomíng; *he beat xiaoming heavily*), which is the basic structure that SimpleNLG-ZH can handle. i.e., [`subject` + `predicate verb` + `object`]. In the “把” construction, however, the marker adverb “把” is added after the subject, and the object is moved to the position right before the predicate verb phrase: [`subject` + “把” + `object` + `predicate verb`].

Note that the positions of modal words and negative words do not follow the movement of the verb phrases (Liu et al., 2001). In other words, in the resulting “把” construction, the modal words and negative words are placed before the object in their own order, as in “他应该没把小明重重地打” (tā yīnggāi méi bǎ xiǎomīng zhòngzhòng de dǎ; *he should haven't beaten xiaoming heavily*). SimpleNLG-ZH realises a sentence with the “把” construction if the user set the feature `BA` to `TRUE`.

4.4.3 “被” Construction

The “被” construction in Mandarin is one of the ways to express the passive, using the basic syntactic structure: [`object` + “被” + `subject` + `predicate verb`]. Using the same example as before in §4.4.2, the transformed sentence would be “小明被他重重地打” (xiǎomīng bèitā zhòngzhòng de dǎ; *Xiaoming is beaten heavily by him*). SimpleNLG-ZH chooses between active and passive based on the value of the feature `PASSIVE`, which is inherited from SimpleNLG-EN.

4.4.4 Interrogative

SimpleNLG-ZH inherits and adapts all its interrogative patterns from SimpleNLG-EN, including “有没有” (yǒuméiyǒu; *Yes-or-no*) and wh-questions: “怎么” (zěnmè; *How*), “什么” (shénmè; *What*), “哪里” (nǎlǐ; *Where*), “谁” (shuí; *Who*), “为什么” (wèishénmè; *Why*), “多少” (duōshǎo; *How Many*). SimpleNLG-ZH adds two further types, namely “哪个” (nǎgè; *Which*) and “什么时候” (shénmèshíshòu; *When*). For Yes-or-no sentences, SimpleNLG-ZH appends the interrogative particle “吗” at the end of a sentence; for instance, “你去上学吗？” (nǐ qù shàngxué ma; *Will you go to school?*).

In SimpleNLG-EN, for wh-questions, only *What* and *Who* made a difference between whether to place the interrogative marker in subject or object position. In SimpleNLG-ZH, however, nearly all wh-question markers can be placed in both positions. Here we use a “什么” (*What*) sentence as an example: For “台风摧毁了他的房子” (táifēng cuīhuǐ le fángzi; *the typhoon destroyed his house*), if we set the feature `INTERROGATIVE_TYPE` to `what_object`, then the sentence is changed to “台风摧毁了什么？” (táifēng cuīhuǐ le shénme; *what did the typhoon destroy?*). Setting the feature to `what_subject` results in “什么摧毁了他的房子？” (shénme cuīhuǐ le tādefángzi; *what destroyed his house?*). In interrogated “把” constructions and ‘被” constructions, the wh-question markers are placed *in situ*, i.e., replacing the phrases in the original subject or object position, according to the value of `INTERROGATIVE_TYPE`.

4.4.5 Topicalisation

Topic structures, especially gapped topic structures, are a very common syntactic structure in Mandarin (Xu and Langendoen, 1985). For example, “绿色的椅子，那把大号的” (lǜsè de yǐzi, nà bǎ dàhào de; *(As for) the green chair, it is the large one*) is a gapped topicalised sentence, in which the constituent after the “的” in the phrase ”那把大号的” (nàbǎ dàhào de; *the large one*) moved into the

topic position and left a gap.

In the current version of SimpleNLG-ZH, we realise a gapped topicalised sentence by viewing it as two coordinated noun phrases, in which the second noun phrase has an empty head noun. For the sentence above, the two noun phrases are "绿色的椅子" (lǜsè de yǐzi; *the green chair*) and "那把大号的" (nàbǎ dàhào de; *the large one*). In the current version of our system, there is no guarantee that the empty head of the second clause is bounded by the first clause. We also consider orthography in topicalisation, i.e., a conjunction words between two phrases should be changed to a comma. In our system, the topicalised sentence, as a `CoordinatedPhraseElement` object, calls the `topicalise()` function to take care of the punctuation.

5 Lexicon

Unlike SimpleNLG-EN, we did not have a ready-to-use elaborate lexicon for SimpleNLG-ZH. Instead, we extracted a primary lexicon from the Chinese as a Foreign Language (CFL) corpus[5] (Lee et al., 2017), which is a sub-corpus of the Universal Dependencies corpus. The CFL corpus has 451 human tagged dependency trees and 7,256 tokens in total. Each word in CFL was primarily mapped to one of the lexical categories in SimpleNLG-ZH based on the relations in Table 1 as well as the following rules:

1. The tag `<proper/>` is appended for `PROPN`s;
2. The tag `<nonpredicate/>` is appended for non-predicate adjectives manually, which is based on the non-predicate adjective list in Liu et al. (2001);
3. The tag `<locative/>` is appended for localisers manually;
4. The words that serve as a dependent of a `clf` (classifier) dependency relation are given the category `classifier`.

The constructed lexicon has 1,639 lexical entries at in total.

6 Evaluation

We decided to evaluate SimpleNLG-ZH in two ways. Firstly, following Soto et al. (2017) and Bollmann (2011), we applied a set of unit test to each module of the system, using the test cases from SimpleNLG-EN plus a set of newly constructed test cases that address some of the peculiarities of Mandarin (e.g., the "把" construct).

Lexical Category	Universal POS Tag
adverb	ADV, PART
noun	NOUN, PROPN
preposition	ADP
demonstrative	DET
conjunction	SCONJ, CCONJ
pronoun	PRONOUN
adjective	ADJ
modal	AUX
verb	VERB

Table 1: Relationship between Universal POS tags and lexical categories in SimpleNLG-ZH.

Secondly, we evaluated the system using a set of expressions from a corpus of actual language use; this was reminiscent of Mazzei et al. (2016) and Bollmann (2011), but using a larger set of expressions. In all cases, when faced with an input expression (i.e., from a test set or corpus), we used this expression to construct a formatted input that was then passed to SimpleNLG-ZH to produce an output expression which was then compared to the input expression.

Evaluation with tests cases. The test cases consist of 144 sentences manually translated and adapted from SimpleNLG V4.4.8 `JUnit Tests` and two reference grammar books (Huang et al., 2009; Liu et al., 2001). The test cases cover all the linguistic features discussed in previous sections and all possible syntactic structures of referring expressions in Mandarin introduced in van Deemter et al. (2017). All the tests were passed by SimpleNLG-ZH, that is, the generated sentences were all identical *verbatim* to the inputs.

Corpus-based evaluation. We picked 100 noun phrases at random from the MTuna corpus (van Deemter et al., 2017), which is the corpus that first version of SimpleNLG-ZH focus on as stated in §1. MTuna is a corpus that has totally 1,650 referring expressions. We then re-generated these expressions using SimpleNLG-ZH. Not all re-generated NPs were identical verbatim to the original MTuna NPs. 35 noun phrases did not match completely (i.e., *verbatim*) with the original noun phrases. Table 2 lists some typical examples, showing differences in word ordering, punctuation, and so on. We ran a human evaluation to find out whether the realised sentences were acceptable (i.e., are they fluent and do they have the same meaning as their inputs). Two native speak-

[5] https://github.com/UniversalDependencies/UD_Chinese-CFL/tree/master

Type	ID	Noun Phrases from MTuna	Realised Sentence	Acceptable
1	1	黑头发，络腮胡，黑西服，浅色衬衣 hēitóufà, luòsāihú, hēixīfú, qiǎnsèchènyī *a man with black hair, whiskers, black suit and light shirt*	黑 头发 络腮 胡 黑 西服 浅色 衬衣 hēitóufà luòsāihú hēixīfú qiǎnsèchènyī	Yes
2	2	一张大的红色的沙发 yìzhāng dà de hóngsè de shāfā the large red sofa	一 张 红色 的 大 的 沙发 yìzhāng hóngsè de dà de shāfā	Yes
	3	戴眼镜的两个人 dài yǎnjìng de liǎng gè rēn *the people who wear glasses*	两 个 戴 眼镜 的 人 liǎng gè dài yǎnjìng de rēn	Yes
	4	红色正面朝向屏幕小椅子或者绿色背向屏幕的大风扇 hóngsè zhèngmiàn cháoxiàng píngmù xiǎo yǐzi huòzhě lǜsè bèixiàng píngmù de dà fēngshàn *the fronting small red chair and the backing large green fan*	正面 朝向 屏幕 小 红色 椅子 或者 背 向 屏幕 的 绿色 大 风扇 zhèngmiàn cháoxiàng píngmù xiǎo hóngsè yǐzi huòzhě bèixiàng píngmù de lǜsè dà fēngshàn	No
	5	黑色头发戴眼镜的 hēsè tóufà dài yǎnjìng de *the person with black hair and glasses*	戴 眼镜 的 黑色 头发 dài yǎnjìng de hēsè tóufà	No
3	6	红色椅子，椅子背朝向右边，可以看到椅子背的正面 hóngsè yǐzi, yǐzìbèi cháo yòubiān, kěyǐ kàndào yǐzìbèi de zhèngmiàn *It is a red chair whose back is facing right and we could see the front of its back.*	*(failed)*	No
	7	正朝向我们的小的椅子和正朝向我们的大的风扇 zhèng cháoxiàng wǒmén de xiǎo de yǐzi hé zhèng cháoxiàng wǒmén de dà de fēngshàn *the fronting small chair and the fronting large fan*	正 朝向 我 的 小 的 椅子 和 正 朝向 我 的 大 的 风扇 zhèng cháoxiàng wǒ de xiǎo de yǐzi hé zhèng cháoxiàng wǒ de dà de fēngshàn	No

Table 2: Example sentences (with their Pinyin and translations) that were not identical to the inputs from MTuna *(unmatched sentences)*. The last column says whether the output was judged to be acceptable by our annotators.

ers annotated the outputs; they reached good inter-annotator agreement ($\kappa = 0.77$) and were asked to produce a consensus annotation, which was then used for our evaluation. It turned out that 90 out of 100 sentences were judged to be acceptable, which we consider a very encouraging result.

We classified the unmatched sentences into three types. The first one is where punctuation was different, as in Example 1 in Table 2. The reason is that some sentences used commas to separate modifiers but SimpleNLG-ZH does not. These cases were generally judged to be acceptable.

The second type is where the word order of the realised sentences was different from the input. There are three sub-types: a) The order of adjective pre-modifiers was different, as in Examples 2 and 4. Most of these deviations were judged to be acceptable, but sentence 4 shows an unacceptable example, where the word "红色" (hóngsè; *red*) before "小" (xiǎo; *little*) accidentally produced a new word, "小红色" (*light red*), which has different meaning; b) SimpleNLG-ZH enforces the pre-modifiers to appear following the specifiers. However, in the MTuna corpus, there are expressions, like Example 3, that switch the place of specifiers and pre-modifiers. All such re-orderings were judged to be acceptable; c) There is a special syntactic pattern of noun phrases in Mandarin, where a Noun is omitted that is recoverable from the context. For example, in Example 5, the head is omitted in the original sentence to construct a free relative (Teng, 1979) where the particle "的" works as sentence-final marker. However, SimpleNLG-ZH cannot recognise the functionality of the particle, thus it switches two pre-modifiers according to the orders defined in §4.1.4, which results in a noun phrase with different meaning. We found 6 unacceptable cases of the second type.

SimpleNLG-ZH failed to reproduce some types of language use that are highly colloquial and not strictly grammatical. We found 4 such cases, as in Example 6 in Table 2, and in Example 7, where

the pronoun "我们" (*us*) in the sentence actually refers to the subject himself (but using the plural form); SimpleNLG-ZH realises this as a singular pronoun.

Comparing these results with earlier evaluations of SimpleNLG-like systems, our results on the tests sets were perfect (with system input constructed by hand from the input expressions), which was also the cases for most earlier studies (Soto et al., 2017; Bollmann, 2011). Only three of the previous evaluations involved a corpus. Bollmann (2011) and Dokkara et al. (2015) evaluated their system on 152 sentences from five Wikipedia articles and 738 sentences randomly picked from a book, respectively. The linguistic variation of their test set is greater than ours (which focussed on referring expressions), but the quality of their output may have been lower: Dokkara et al. (2015) reported 57% of exact matches, lower than our 65%. Bollmann (2011) reported 76% of the sentences "could be generated", though what this meant is not entirely clear. Mazzei et al. (2016) tested the coverage and scalability of their system by automatically mapping 20 dependency trees from the Universal Dependency corpus. They reported only 10% exact matching sentences (2/20) and their discussion suggests that their results for declarative and interrogative sentences may have been disappointing.

7 Conclusion and Future Work

We have introduced and evaluated a realisation engine for Mandarin in the tradition of SimpleNLG. We hope SimpleNLG-ZH can be a good starting point for work on other Sino-Tibetan languages, such as Tibetan and Cantonese.

Realisation has turned out to be non-trivial in all the languages addressed in the SimpleNLG tradition so far, but *where* the most challenging problems are (i.e., in which components of the system), and what the optimal balance between handcrafting and Machine learning should lie, is something that differs per language.

As for the former issue, we have seen that Mandarin appears to require only a small set of morphological operators, but a much enhanced set of syntactic processing rules.

As for the latter issue, our study of errors in SimpleNLG-ZH offers support for the idea that some issues in realisation are best handled using Machine Learning (Langkilde, 2000; White et al., 2007). As it stands, SimpleNLG-ZH makes all its decisions based on a combination of handcrafted rules and explicit stipulation. It would be preferable if the role of the developer in making these decisions could be reduced. This is true for the choice of classifiers (see §4.1.1), for the use of particles (such as "的" and "了"), for the choice between different negation words ("不" or "没"), and for ordering the modifiers and specifiers (as mentioned in §6). In all these cases, SimpleNLG-ZH assumes that the choice is made outside the system (i.e., by a person or by another component of the NLG system). It would be useful if these choices were made by SimpleNLG-ZH itself, but it is difficult to see how a rule-based approach could accomplish this. We therefore aim to experiment with statistical models (e.g., language models) to make these decisions. The result would be a hybrid realisation system that combines rules and Machine Learning.

Acknowledgements

As well as the anonymous reviewers, we thank Rint Sybesma, Xiwu Han, Ehud Reiter, Yaji Sripada, and others in the Aberdeen CLAN group for their comments on SimpleNLG-ZH and this paper.

References

Marcel Bollmann. 2011. Adapting SimpleNLG to German. In *Proceedings of the 13th European Workshop on Natural Language Generation*, pages 133–138. Association for Computational Linguistics.

Rodrigo De Oliveira and Somayajulu Sripada. 2014. Adapting SimpleNLG for Brazilian Portuguese realisation. In *INLG*, pages 93–94.

Kees van Deemter, Le Sun, Rint Sybesma, Xiao Li, Chen Bo, and Muyun Yang. 2017. Investigating the content and form of referring expressions in Mandarin: introducing the Mtuna corpus. In *Proceedings of the 10th International Conference on Natural Language Generation*, pages 213–217.

Sasi Raja Sekhar Dokkara, Suresh Verma Penumathsa, and Somayajulu Gowri Sripada. 2015. A simple surface realization engine for Telugu. In *ENLG*, pages 1–8.

Albert Gatt and Emiel Krahmer. 2018. Survey of the state of the art in Natural Language Generation: Core tasks, applications and evaluation. *Journal of Artificial Intelligence Research (JAIR)*, 61:65–170.

Albert Gatt and Ehud Reiter. 2009. SimpleNLG: A realisation engine for practical applications. In *Proceedings of the 12th European Workshop on Natural Language Generation*, pages 90–93. Association for Computational Linguistics.

Wei He, Haifeng Wang, Yuqing Guo, and Ting Liu. 2009. Dependency based Chinese sentence realization. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2-Volume 2*, pages 809–816. Association for Computational Linguistics.

Cheng-Teh James Huang, Yen-hui Audrey Li, and Yafei Li. 2009. *The syntax of Chinese*, volume 8. Cambridge University Press Cambridge.

John T Jensen. 1990. *Morphology: Word structure in generative grammar*, volume 70. John Benjamins Publishing.

Irene Langkilde. 2000. Forest-based statistical sentence generation. In *Proceedings of the 1st North American chapter of the Association for Computational Linguistics conference*, pages 170–177. Association for Computational Linguistics.

Guy Lapalme. 2013. Natural language generation and summarization at RALI. In *Proceedings of the 14th European Workshop on Natural Language Generation*, pages 92–93.

John Lee, Herman Leung, and Keying Li. 2017. Towards universal dependencies for learner Chinese. In *Proceedings of the NoDaLiDa 2017 Workshop on Universal Dependencies (UDW 2017)*, pages 67–71.

Yen-hui Audrey Li. 2006. Argument determiner phrases and number phrases. *Argument*, 29(4).

Yuehua Liu, Wei Gu, and Wenyu Pan. 2001. *Chinese Grammar*. The Commercial Press.

Mitchell P Marcus, Mary Ann Marcinkiewicz, and Beatrice Santorini. 1993. Building a large annotated corpus of English: The Penn Treebank. *Computational linguistics*, 19(2):313–330.

Alessandro Mazzei, Cristina Battaglino, and Cristina Bosco. 2016. SimpleNLG-IT: adapting SimpleNLG to Italian. In *INLG*, pages 184–192.

Ethel Ong, Stephanie Abella, Lawrence Santos, and Dennis Tiu. 2011. A simple surface realizer for Filipino. In *PACLIC*, pages 51–59.

Jerome L Packard. 2000. *The morphology of Chinese: A linguistic and cognitive approach*. Cambridge University Press.

Waltraud Paul. 2010. Adjectives in Mandarin Chinese: The rehabilitation of a much ostracized category. *Adjectives: Formal analyses in syntax and semantics, ed. Patricia Cabredo Hofherr and Ora Matushansky*, 1:15–151.

François Portet, Ehud Reiter, Albert Gatt, Jim Hunter, Somayajulu Sripada, Yvonne Freer, and Cindy Sykes. 2009. Automatic generation of textual summaries from neonatal intensive care data. *Artificial Intelligence*, 173(7-8):789–816.

Ehud Reiter and Robert Dale. 2000. *Building natural language generation systems*. Cambridge university press.

Alejandro Ramos Soto, Julio Janeiro Gallardo, and Alberto Bugarín Diz. 2017. Adapting SimpleNLG to Spanish. In *Proceedings of the 10th International Conference on Natural Language Generation*, pages 144–148.

Shou-hsin Teng. 1979. Remarks on cleft sentences in Chinese. *Journal of Chinese Linguistics*, 7(1):101–14.

Kees Van Deemter. 2016. *Computational models of referring: a study in cognitive science*. MIT Press.

Pierre-Luc Vaudry and Guy Lapalme. 2013. Adapting SimpleNLG for bilingual English-French realisation. In *ENLG*, pages 183–187.

Michael White, Rajakrishnan Rajkumar, and Scott Martin. 2007. Towards broad coverage surface realization with CCG. In *Proc. of the Workshop on Using Corpora for NLG: Language Generation and Machine Translation (UCNLG+ MT)*.

Ding Xu. 1997. *Functional Categories in Mandarin Chinese*, volume 26. Holland Academic Graphics.

Liejiong Xu and D. Terence Langendoen. 1985. Topic structures in Chinese. *Language*, pages 1–27.

Guowen Yang and John A Bateman. 2009. The Chinese aspect generation based on aspect selection functions. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2-Volume 2*, pages 629–637. Association for Computational Linguistics.