SimpleNLG-DE: Adapting SimpleNLG 4 to German

Daniel Braun
Department of Informatics
Technical University of Munich
daniel.braun@tum.de

Kira Klimt
Department of Informatics
Technical University of Munich
kira.klimt@tum.de

Daniela Schneider
Allianz SE
daniela.schneider1@allianz.com

Florian Matthes
Department of Informatics
Technical University of Munich
matthes@tum.de

Abstract

SimpleNLG is a popular open source surface realiser for the English language. For German, however, the availability of open source and non-domain specific realisers is sparse, partly due to the complexity of the German language. In this paper, we present SimpleNLG-DE, an adaption of SimpleNLG to German. We discuss which parts of the German language have been implemented and how we evaluated our implementation using the TIGER Corpus and newly created data-sets.

1 Introduction

More than 20 years after it was first published, the three-stage architecture for Natural Language Generation (NLG) systems described by Reiter (1994) is still frequently cited. According to his architecture, most NLG systems of the time consisted of a pipeline with three steps: Content Determination, Sentence Planning, and Surface Realisation (or Surface Generation). Today, stochastic approaches to NLG are very popular, which often use a black box approach instead of a modular pipeline (cf. e.g. Dušek et al. (2018)).

Nevertheless, rule-based systems still play a crucial role, especially in application contexts, because they provide advantages like higher controllability. SimpleNLG, developed by Gatt and Reiter (2009) is arguably the most popular open source realisation engine. It is implemented in Java and its current Version (4.4.8) is available under the Mozilla Public License (MPL).[1]

Since it was published in 2009, SimpleNLG was adapted to seven other languages, these are (in chronological order): German (Bollmann, 2011), French (Vaudry and Lapalme, 2013), Italian (Mazzei et al., 2016), Spanish (Ramos-Soto et al., 2017), Dutch (de Jong and Theune, 2018), Mandarin (Chen et al., 2018), and Galician (Cascallar-Fuentes et al., 2018).

Unfortunately, the German version of SimpleNLG[2] is not maintained anymore and is based on the outdated third version of SimpleNLG, which used a more restrictive license that prohibited commercial use. (Bollmann, 2019) Moreover, the existing German version also comes with a very limited lexicon, consisting of just around 100 lemmata. It also does not automatically recognise and handle separable verbs like "abfahren" ("to depart") or "einkaufen" ("to purchase"). The only openly available alternative is a German grammar for OpenCCG (Vancoppenolle et al., 2011), which is even more limited with regard to both grammatical coverage and its lexicon.

Therefore, we decided to develop SimpleNLG-DE, a new German version of SimpleNLG, implemented from scratch, based on SimpleNLG 4.4.8 and the MPL. SimpleNLG-DE comes with a standard lexicon containing more than 100,000 lemmata and is available from https://github.com/sebischair/SimpleNLG-DE.

2 German Language

As also acknowledged by Bollmann (2011), the German language has its specificities, which pose special challenges for the task of surface realisation.

2.1 Word Order

Many different possible word orders can exist for the same sentence in German. "Ohne Pause in den Hof tragen konnten die Kiste nur zwei kräftige Männer." ("Only two strong men could carry the

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

[2] https://marcel.bollmann.me/software/simplenlg.html

Proceedings of The 12th International Conference on Natural Language Generation, pages 415–420,
Tokyo, Japan, 28 Oct - 1 Nov, 2019.

box into the yard without a break.") can also be expressed by shuffling the sentence constituents, resulting in "Die Kiste in den Hof tragen konnten, ohne Pause, nur zwei kräftige Männer.", "In den Hof tragen konnten die Kiste, ohne Pause, nur zwei kräftige Männer.", or "Nur zwei kräftige Männer konnten die Kiste ohne Pause in den Hof tragen.". The German language is thus seen as a "partially free constituent order language" (Vancoppenolle et al., 2011), whereas the shuffling of constituents is called "scrambling" (Eisenberg et al., 2016, p. 881). Moreover, depending on the sentence type, the verb of a sentence must be placed at different positions. The finite verb has to be positioned either in second place, in the first place or in the last place (Eisenberg et al., 2016, pp. 875-878).

2.2 Inflection

Manifold inflection rules are another major reason why the German language is, from a surface realisation perspective, more complex than e.g. English. For the English language, table look-ups for inflected forms can be performed reasonably. This is not feasible for German.

Whereas in the English language "the" as definite article and "a" / "an" as indefinite articles suffice, in German, "der", "die", and "das" as definite articles and "ein" and "eine" as indefinite articles exist. In the German language, additionally, all articles and pronouns must be inflected according to gender, number, person and grammatical case (nominative, genitive, dative, accusative). This results in more article forms, for instance for indefinite articles in "einen", "einem", "einer", "eines". (Eisenberg et al., 2016, p. 341)

Inflection of nouns is dependent on the noun's gender, the grammatical case the noun is in, and the number (singular or plural). (Eisenberg et al., 2016, pp. 146-228) Adjectives can be attributive, predicative, adverbial or nominalized. (Eisenberg et al., 2016, pp. 345-372) In most cases, attributive adjectives are inflected and change with the grammatical case, the number and the gender of the corresponding noun. The following examples illustrate the inflection:

- Inflection according to the case: "das große Haus" ("the big house"), in dative "dem großen Haus"
- Inflection according to the number: "das große Haus", in plural "die großen Häuser"
- Inflection according to the gender: "ein großes Haus" ("a big house", neutral), "die große Frau" ("a tall women", feminine)

Finally, verb conjugation reflects person, number, tense, voice, and mood. (Eisenberg et al., 2016, p. 395) German verbs can be grouped into strong and weak verbs, depending on their inflection pattern in past tense and participle II. (Eisenberg et al., 2016, pp. 440-466) Weak verbs build their past tense forms with a syllable introducing t-suffix, e.g. "lachte" ("laughed"), "redete" ("talked") and their participle II form with "-t"/"-et", e.g. "gelacht", "geredet". Normally, the stem vocal of weak verbs does not change. Strong verbs, in contrast, do not build their past tense forms with a suffix, but with an alteration of the stem vocal (ablaut), e.g. "rufen - rief" ("to call - called") or "finden - fand" ("to find - found"). Participle II forms are built with the suffix "-en" and, in some cases, with an ablaut: "singen - sang" ("to sing - sang"). Furthermore, there are some verbs with strong-weak mixed conjugation, or other irregularities, for example some modal verbs, auxiliary verbs, or the verb "wissen" ("to know").

2.3 Separable Verbs

Separable verbs (e.g. "losfahren" / "moving off"), also referred to as particle verbs, contain a prefix which can be separated. The order of the prefix ("los") and the verb ("fahren") can be reversed in some conjugated forms (Eisenberg et al., 2016, pp. 705-714). The verb "hinausgehen" ("to step out", "to leave"), for instance, consists of the adverb "hinaus" ("out"), and the verb "gehen" ("to go"). When "hinausgehen" is conjugated, the first person in the present tense is "ich gehe hinaus" ("I step out"), where the prefix is separated from the verb. Prefix types range from prepositional, to adverbial, adjective or substantive particles. "Preis" ("price") in "preisgeben" ("to reveal") contains a substantive particle, whereas "widerspiegeln" ("to reflect"), contains a preposition as prefix. Separable verbs also exist in other languages like Dutch. (de Jong and Theune, 2018)

2.4 Compound Words

Compounds are complex linguistic constructs consisting of several words. Subjective and adjective compounds can be built by combining two or more words into a compound, for example, "Wunderkind" ("prodigy child"), or "rubinrot"

("ruby red"). The last component of a compound word dominates the word, i.e., a "Wunderkind" is a "Kind" ("child") rather than a "Wunder" ("prodigy"). Internally, compound words can e.g. be inflected with a genitive ending, like in "Kapitän*s*mütze" ("captain's head"). The grammatical characteristics of the whole word, like the gender or the inflection type, are determined by the last component. Compound words are the most important way of building words in the German language. (Elsen, 2009)

So-called word group lexemes are similar to compound words. They are fixed phrases of at least two separately written words. "Erste Hilfe" ("first aid"), "Europäische Union" ("European Union"), or "Vereinigte Arabische Emirate" ("United Arab Emirates") are examples for word group lexemes. (Elsen, 2009) In contrast to compound words, word group lexemes are internally inflected. "Vereinigte Arabische Emirate" ("United Arab Emirates") is inflected in the dative case to "[aus den] Vereinigten Arabischen Emiraten" ("from the United Arab Emirates").

3 Grammatical Coverage

In this section, we describe which parts of the grammar of the German language are implemented in the first version of SimpleNLG-DE. In Section 5, we will discuss which important parts are not yet covered.

3.1 Syntax

The handling of the word order in SimpleNLG-DE is implemented according to the topological model of the "Duden" (Eisenberg et al., 2016, pp. 874 - 880). The library currently supports two types of clauses, declarative clauses and questions. The word order for declarative clauses without a front-modifier is *subject - finite verb - objects - other verb forms*. Unlike the previous implementation of SimpleNLG for German, SimpleNLG-DE detects separable verbs automatically and changes the word order to *subject - finite verb - objects - separable particle* (e.g. "Alice räumt das Auto ein." / "Alice is loading the car."). The handling of separable words is similar to the implementation in the Dutch version of SimpleNLG (de Jong and Theune, 2018). Separable verbs are marked as separable in the lexicon and the lexicon entry includes their prefix as a separate entry. Initiated subordinate clauses which contain a separable verb have to be treated with care. As an example, the complex sentence "Florian geht einkaufen, Alex räumt sein Zimmer auf." ("Florian goes shopping, Alex cleans his room.") can be changed to "Florian geht einkaufen, während Alex sein Zimmer aufräumt." ("Florian goes shopping, while Alex cleans his room."), with the second sentence added as initiated subordinate clause to the first. In the second sentence, besides the changed word oder, the verb conjugation changes too. The separable verb is separated in the first clause ("räumt auf" / "tidies up"), but stays together in the second clause ("aufräumt"). (Agbaria, 2009) For all initiated subordinate clauses, SimpleNLG-De does not split separable verbs.

Moreover, SimpleNLG-DE can produce five different kinds of questions: yes/no questions and questions about the subject and object of a sentence for both people ("wer" / "who") and things ("was" / "what").

Beyond the main clauses, SimpleNLG-DE can handle compound sentences connected with "und" ("and") or comma ("Der Hund bellt und die Katze miaut" / "The dog barks and the cat mews."), temporal, causal, conditional, consecutive, concessive, modal, comparative, final, and adversative subordinate clauses ("Die Sonne scheint, während es regnet." / "The sun shines while it is raining."), appositions ("SAP, eine deutsche Firma, ..." / "SAP, a German company, ..."), and enumerations ("SAP, Bayer und EON" / "SAP, Bayern, and EON").

3.2 Morphology

Morphology in SimpleNLG-DE is based on a combination of rules for regular inflections, extracted from Eisenberg et al. (2016) and Agbaria (2009), and a lexicon covering 100,000 unique lemmata (∼ 78,000 nouns, 10,000 verbs, 11,000 adjectives, 1,000 adverbs), which was extracted from Wiktionary[3]. Like Wiktionary itself, the lexicon is licensed under the Creative Commons Attribution-ShareAlike 3.0 Unported (CC BY-SA 3.0)[4] license. The lexicon does not contain all conjugated forms for all persons in all tenses but covers a set of forms big enough to create all inflected forms with additional rules. If a verb is not in the lexicon, it is conjugated regularly.

[3] https://de.wiktionary.org/

[4] https://creativecommons.org/licenses/by-sa/3.0/deed.en

Verb conjugation currently covers present, past, perfect, and future tense, passive in present and past, modal verbs in present, as well as the handling of separable verbs. Adjectives are declined according to the case, number, and article. Moreover, comparative and superlative for adjectives and adverbs can be generated. Nouns can be inflected based on the case and number and their articles according to the case, number, and gender, for both, definite and indefinite articles. Word group lexemes can also be inflected according to the case and number. Additionally, SimpleNLG-DE is able to automatically detect the contraction of prepositions and inflect adjectives correctly in cases like "in dem großen Haus" which can be contracted to "im großen Haus" ("in the big house").

3.3 Orthography

The orthography processor of SimpleNLG-DE handles terminating declarative clauses with ".", questions with "?", capitalising the first character in a sentence, and comma rules. If a sentence is set as a complement to another sentence, and both of them do not add a complementiser, or the complementiser is in a list of conjunctions which requires a comma, the complement is added with a preceding comma. For sentences added with the complementiser "und", no comma is added. Appositions have a comma added before and after them, no matter if "und" is contained in the apposition or not. Enumerations are connected by adding a comma between the first constituents, and separating the last one with "und", for instance in "A, B und C" ("A, B, and C").

4 Evaluation

Evaluating a surface realiser is in many aspects a difficult task. There are two facets which we tried to evaluate: First, how robust and correct is the implementation of the grammatical features described in Section 3 and second, how much of everyday language can be covered with the current implementation of SimpleNLG-DE. The first aspect can be evaluated relatively easy by manually creating special test cases for the different grammatical features that have been implemented. Evaluating the coverage of a language is far more complex. The best, yet flawed, approach is choosing an existent corpus which is believed to be somewhat representative of the language as a whole. This approach was also chosen by the authors of other versions of SimpleNLG. Bollmann (2011), for example, used five Wikipedia articles with 152 sentences in total to evaluate the coverage and achieved 75.66%. Unfortunately, the test data was not published, therefore, we can not compare the new implementation directly with the previous version on this dataset. Many other versions, like the Spanish and Mandarin versions, used translations of the 144 test sentences from the original SimpleNLG version. However, it should be noted that these sentences were merely meant to be an "indication of efficiency" test (Gatt and Reiter, 2009) and not an evaluation of the coverage.

We used more than 3,800 test sentences to evaluate the correct implementation of the grammatical features described in Section 3. These tests cover e.g. the inflection of verbs (2,436 sentences), the inflection of adjectives (1,002 sentences), and the inflection of nouns (390 sentences), but also other features like question generation. SimpleNLG-DE was able to generate all of these sentences correctly. The sentences were implemented manually and partially based on sentences from documents from the financial domain and partially written by the authors for testing purposes.

In order to get an estimate how much of the German language is covered, we used the TIGER Corpus (Brants et al., 2004). It contains 50,000 sentences of German newspaper articles taken from the "Frankfurter Rundschau". As newspaper text can contain rather complex phrase structures, it is considered suitable test data for a German language realiser. Annotations in TIGER corpus include semi-automatically generated POS-tags as well as syntactic structure, morphological and lemma information. The TIGER corpus is freely available for research and evaluation purposes.

Since writing the code to generate a sentence is a very time consuming task, we could not test our implementation on the whole corpus. Instead, we randomly chose 100 declarative sentences from the TIGER corpus (i.e. interrogative, imperative, and exclamatory sentences were excluded) and implemented them using SimpleNLG-DE. We used the annotations from the TIGER corpus to semi-automatically create the code for the tests, however all sentences were manually checked and adapted before they were added to the test set.

84% of all sentences could be generated correctly using the library. Counted as correct are only sentences which are equal to their corresponding sentence in the corpus. The main reasons for wrongly realised sentences include problems with the pluralisation of irregular compound nouns which are not part of the lexicon and of verbs in cases where the corresponding noun is a number (e.g. "Im Schnitt waren es seit 1980 jedoch nur 4 208." / "On average, however, since 1980 it has been only 4 208.").

Since the code for the tests is only compatible with SimpleNLG version 4, we were not able to directly compare the performance of the old and new version of SimpleNLG on the TIGER corpus. For license reasons, the tests generated from the TIGER corpus are not published alongside the code of SimpleNLG-DE, however, all other tests are part of the repository.

5 Limitations

SimpleNLG-DE covers a subset of the German language. Some grammar parts are left for future work, due to the complexity of German language, its manifold inflected words and rules, and its diverse word order possibilities with a large number of exceptions. Indications on how to extend the library in the future, according to its current limitations, are given in this section.

Tenses currently not covered by SimpleNLG-DE are future II ("Ich werde es gekauft haben." / "I will have bought it.") and plusquamperfect tense ("Sie hatte Fußball gespielt." / "She had played football."). Furthermore, passive currently only works for present and preterite tenses, and modal verbs only work for present active. Phrases such as "soll verursacht sein" ("shall be caused"), for instance, are not covered. Only indicative mood is integrated. Conjunctive and imperative are not yet implemented.

Compound words (cf. Section 2.4) are currently only correctly handled if they are part of the lexicon. While there are existing approaches on how to automatically split compound words into their respective parts (e.g. by Baroni et al. (2002), Koehn and Knight (2003), Daiber et al. (2015), Sugisaki and Tuggener (2018), and Weller-Di Marco (2017)), the problem is far from being trivial and is not yet addressed by the implementation.

For some German verbs, there are several correct ways to conjugate them. The verb "senden" ("to send"), for instance, in the third person past tense can either be conjugated to "sendete" or to "sandte", without changing the meaning. Such subtleties are currently not covered by the library. While in the previous example, this is merely a question of style, some verbs actually change their meaning. The verb "wachsen" in third person present tense in its irregular form is "er wächst" meaning "he is growing", whereas the regular form "er wachst" means "he waxes (sth.)". In future, an option for the user to set the desired meaning should be given.

6 Conclusion

In this paper, we presented SimpleNLG-DE, an adaption of the open source surface realiser SimpleNLG for the German language which is licensed under the MPL. The current implementation covers the most important basic features of the German language and comes with a lexicon covering more than 100,000 lemmata.

The implementation was validated by testing for grammatical functionality, e.g. verb conjugation, and language coverage on real-world newspaper articles from the TIGER corpus. SimpleNLG-DE was able to correctly reproduce 84% of the selected sentences from the TIGER corpus.

In the future, we would like to enhance the implementation by addressing the limitations mentioned in Section 5. Furthermore, we would like to test SimpleNLG-DE in different application domains with specific language, like the legal domain (cf. e.g. Braun et al. (2019)).

Acknowledgments

This work has been sponsored by the German Federal Ministry of Education and Research (BMBF) grant A-SUM 01IS17049 and Allianz SE.

References

Evelyn Agbaria. 2009. *PONS die deutsche Rechtschreibung*, volume 1. PONS GmbH, Stuttgart.

Marco Baroni, Johannes Matiasek, and Harald Trost. 2002. Predicting the components of german nominal compounds. In *ECAI 2002: 15th European Conference on Artificial Intelligence*, pages 470–474.

Marcel Bollmann. 2011. Adapting SimpleNLG to German. In *Proceedings of the 13th European Work-*

shop on Natural Language Generation, pages 133–138, Nancy, France. Association for Computational Linguistics.

Marcel Bollmann. 2019. Simplenlg for german. https://marcel.bollmann.me/software/simplenlg.html. Last accessed 2019-09-10.

Sabine Brants, Stefanie Dipper, Peter Eisenberg, Silvia Hansen-Schirra, Esther König, Wolfgang Lezius, Christian Rohrer, George Smith, and Hans Uszkoreit. 2004. Tiger: Linguistic interpretation of a german corpus. *Research on language and computation*, 2(4):597–620.

Daniel Braun, Elena Scepankova, Patrick Holl, and Florian Matthes. 2019. Consumer protection in the digital era: The potential of customer-centered legaltech. In *INFORMATIK 2019: 50 Jahre Gesellschaft für Informatik - Informatik für Gesellschaft*, pages 407–420, Bonn. Gesellschaft für Informatik e.V.

Andrea Cascallar-Fuentes, Alejandro Ramos-Soto, and Alberto Bugarín Diz. 2018. Adapting SimpleNLG to Galician language. In *Proceedings of the 11th International Conference on Natural Language Generation*, pages 67–72, Tilburg University, The Netherlands. Association for Computational Linguistics.

Guanyi Chen, Kees van Deemter, and Chenghua Lin. 2018. SimpleNLG-ZH: a linguistic realisation engine for Mandarin. In *Proceedings of the 11th International Conference on Natural Language Generation*, pages 57–66, Tilburg University, The Netherlands. Association for Computational Linguistics.

Joachim Daiber, Lautaro Quiroz, Roger Wechsler, and Stella Frank. 2015. Splitting compounds by semantic analogy. In *Proceedings of the 1st Deep Machine Translation Workshop*, pages 20–28, Praha, Czechia. ÚFAL MFF UK.

Ondřej Dušek, Jekaterina Novikova, and Verena Rieser. 2018. Findings of the E2E NLG challenge. In *Proceedings of the 11th International Conference on Natural Language Generation*, pages 322–328, Tilburg University, The Netherlands. Association for Computational Linguistics.

Peter Eisenberg, Jörg Peters, Peter Gallmann, Cathrine Fabricius-Hansen, Damaris Nübling, Irmhild Barz, Thomas A Fritz, Reinhard Fiehler, and Mathilde Henning. 2016. *Duden - Die Grammatik. Unentbehrlich für richtiges Deutsch*. Dudenverlag, Mannheim.

Hilke Elsen. 2009. Komplexe komposita und verwandtes. *Germanistische Mitteilungen: Zeitschrift für Deutsche Sprache, Literatur und Kultur*, (69):57–71.

Albert Gatt and Ehud Reiter. 2009. Simplenlg: A realisation engine for practical applications. In *Proceedings of the 12th European Workshop on Natural Language Generation*, ENLG '09, pages 90–93, Stroudsburg, PA, USA. Association for Computational Linguistics.

Ruud de Jong and Mariët Theune. 2018. Going Dutch: Creating SimpleNLG-NL. In *Proceedings of the 11th International Conference on Natural Language Generation*, pages 73–78, Tilburg University, The Netherlands. Association for Computational Linguistics.

Philipp Koehn and Kevin Knight. 2003. Empirical methods for compound splitting. In *Proceedings of the Tenth Conference on European Chapter of the Association for Computational Linguistics - Volume 1*, EACL '03, pages 187–193, Stroudsburg, PA, USA. Association for Computational Linguistics.

Alessandro Mazzei, Cristina Battaglino, and Cristina Bosco. 2016. SimpleNLG-IT: adapting SimpleNLG to Italian. In *Proceedings of the 9th International Natural Language Generation conference*, pages 184–192, Edinburgh, UK. Association for Computational Linguistics.

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, Santiago de Compostela, Spain. Association for Computational Linguistics.

Ehud Reiter. 1994. Has a consensus nl generation architecture appeared, and is it psycholinguistically plausible? In *Proceedings of the Seventh International Workshop on Natural Language Generation*, pages 163–170. Association for Computational Linguistics.

Kyoko Sugisaki and Don Tuggener. 2018. German compound splitting using the compound productivity of morphemes. In *14th Conference on Natural Language Processing-KONVENS 2018*, pages 141–147. Austrian Academy of Sciences Press.

Jean Vancoppenolle, Eric Tabbert, Gerlof Bouma, and Manfred Stede. 2011. A german grammar for generation in open ccg. In *Multilingual resources and multilingual applications: Proceedings of the Conference of the German Society for Computational Linguistics and Language Technology (GSCL)*, pages 145–150. Citeseer.

Pierre-Luc Vaudry and Guy Lapalme. 2013. Adapting SimpleNLG for bilingual English-French realisation. In *Proceedings of the 14th European Workshop on Natural Language Generation*, pages 183–187, Sofia, Bulgaria. Association for Computational Linguistics.

Marion Weller-Di Marco. 2017. Simple compound splitting for German. In *Proceedings of the 13th Workshop on Multiword Expressions (MWE 2017)*, pages 161–166, Valencia, Spain. Association for Computational Linguistics.