IWNLP: Inverse Wiktionary for Natural Language Processing

Matthias Liebeck and Stefan Conrad
Institute of Computer Science
Heinrich-Heine-University Düsseldorf
D-40225 Düsseldorf, Germany
{liebeck,conrad}@cs.uni-duesseldorf.de

Abstract

Nowadays, there are a lot of natural language processing pipelines that are based on training data created by a few experts. This paper examines how the proliferation of the internet and its collaborative application possibilities can be practically used for NLP. For that purpose, we examine how the German version of Wiktionary can be used for a lemmatization task. We introduce IWNLP, an open-source parser for Wiktionary, that reimplments several MediaWiki markup language templates for conjugated verbs and declined adjectives. The lemmatization task is evaluated on three German corpora on which we compare our results with existing software for lemmatization. With Wiktionary as a resource, we obtain a high accuracy for the lemmatization of nouns and can even improve on the results of existing software for the lemmatization of nouns.

1 Introduction

Wiktionary is an internet-based dictionary and thesaurus that lists words, inflected forms and relations (e.g. synonyms) between words. Just as Wikipedia, Wiktionary uses MediaWiki as a platform but focuses on word definitions and their meaning, rather than explaining each word in detail, as Wikipedia does. The dictionary contains articles, which can each list multiple entries for different languages and multiple parts of speech. For instance, the English word home has entries as a noun, verb, adjective and as an adverb.

Each article is rendered by the MediaWiki engine from a text-based input, which uses the MediaWiki syntax and relies heavily on the use of templates. The articles are editable by everyone, even by unregistered users. Although vandalism is possible, most of the vandalized entries are identified by other users who watch a list of the latest changes and subsequently revert these entries to previously correct versions. All text content is licensed under the Creative Commons License, which makes it attractive for academic use.

There are currently 111 localized versions of Wiktionary, which contain more than 1000 articles. A localized version can establish own rules via majority votes and public opinion. For example, the German version of Wiktionary currently enforces a 5-source-rule, which requires that each entry that is not listed in a common dictionary is documented by at least 5 different sources. The German version of Wiktionary has grown over the last years and currently contains almost 400000 articles. Each word is listed with its part-of-speech tag, among other information. If a word is inflectable (nouns, verbs, adjectives, pronouns and articles are inflectable in the German language), all inflected forms are also enumerated. Table 1 shows the declension of the noun Turm (tower). Wiktionary provides information that can be used as a resource for Natural Language Processing (NLP), for instance for part-of-speech tagging, for lemmatization and as a thesaurus.

<table>
<thead>
<tr>
<th>Case</th>
<th>Singular</th>
<th>Plural</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nominative</td>
<td>der Turm</td>
<td>die Türme</td>
</tr>
<tr>
<td>Genitive</td>
<td>des Turmes</td>
<td>der Türme</td>
</tr>
<tr>
<td>Dative</td>
<td>dem Turm</td>
<td>den Türmen</td>
</tr>
<tr>
<td>Accusative</td>
<td>den Turm</td>
<td>die Türme</td>
</tr>
</tbody>
</table>

1https://meta.wikimedia.org/wiki/Wiktionary
2https://de.wiktionary.org
3https://de.wiktionary.org/wiki/Wiktionary:Meilensteine
The rest of the paper is structured as follows: Section 2 gives an overview of previous applications of Wiktionary for natural language processing purposes. Section 3 outlines the basic steps of parsing Wiktionary. The use of Wiktionary as a lemmatizer is evaluated in section 4 and compared with existing software for lemmatization. Finally, we conclude in chapter 5 and outline future work.

2 Related Work

The closest work to ours is JWKTL (Zesch et al., 2008). JWKTL is a Wiktionary parser that was originally developed for the English and the German version of Wiktionary, but it now also supports Russian. Our work differs from JWKTL, because we currently focus more on inflections in the German version than JWKTL. Therefore, we have a larger coverage of inflections, because we additionally reimplemented several templates from the namespace Flexion. Also, we have an improved handling of special syntactic cases, as compared to JWKTL.

Wiktionary has previously been used for several NLP tasks. The use of the German edition as a thesaurus has been investigated by Meyer and Gurevych (2010). The authors compared the semantic relations in Wiktionary with GermaNet (Hamp and Feldweg, 1997) and OpenThesaurus (Naber, 2005).

Smedt et al. (2014) developed a part-of-speech tagger based on entries in the Italian version of Wiktionary. They achieved an accuracy of 85.5 % with Wiktionary alone. By using morphological and contextual rules, they improve their tagging to an accuracy of 92.9 %. Li et al. (2012) also used Wiktionary to create a part-of-speech tagger, which is based on a hidden Markov model. Their evaluation of 9 different languages shows an average accuracy of 84.5 %, with English having the best result with an accuracy of 87.1 %.

3 Parsing Wiktionary

There are multiple ways to parse Wiktionary. It is possible to crawl all existing articles from the online servers. To reduce stress from the servers and to easily reproduce our parsing results, we parse the latest of the monthly XML dumps from Wiktionary. For this paper, we use the currently latest dump 20150407.

We iterate over every article in the XML dump and parse articles which contain German word entries. These articles can be separated into two groups: the ones in the main namespace (without any preceding namespace, like ‘namespace:’) and the ones in the namespace Flexion. First, we describe how we parse the articles in the main namespace. An article can contain entries for multiple languages. Therefore, we divide its text content into language blocks (== heading ==) and skip non-German language blocks. Afterward, we extract one or more entries (=== heading ===) from each German language block. If an article lists more than one entry with the same name, its word forms will be different from each other. For instance, the German word Mutter\(^5\), contains an entry for mother and for nut, which have different plural forms. We parse the part-of-speech tag for each entry. If a word is inflectable, we will also parse its inflections, which are listed in a key-value-pair template. Depending on the part-of-speech tag, different templates are used in Wiktionary for which we use different parsers. We provide parsers for nouns, verbs, adjective and pronouns. The key-value-template for the adjective gelb (yellow) is displayed in Figure 1.

\begin{verbatim}
== gelb ({{{Language|German}}}) ==
=== {{POS|Adjective|German}} ===
|German Adjective Overview
|Positive=gelb
|Comparative=gelber
|Superlative=am gelbsten
\}
\end{verbatim}

Figure 1: Adjective template for the word gelb (yellow), with keywords translated into English

At this point, we should point out that the inflections for verbs and adjectives in the main namespace are only a small portion of all possible inflections. For example, a verb in the main namespace only lists one inflection for the past tense (first person singular), while other possible past tense forms are not listed.

Fortunately, it is possible that a verb or an adjective has an additional article in the namespace Flexion, where all inflections are listed. However, the parsing of these inflections is more challenging, because the articles use complex templates.

\(^5\)https://de.wiktionary.org/wiki/Mutter
\(^4\)http://dumps.wikimedia.org/dewiktionary/
Although the parsing of the parameters for the templates remains the same, it is more difficult to retrieve the rendered output by the MediaWiki engine (and thus the inflections) from these templates, because it is very rare that inflections are listed as a key-value-pair. Instead, these templates require principal parts, which are combined with suffixes. The users of Wiktionary have created templates, that take care of special cases, for instance for a verb conjugation, where the suffix 'est' is added to a verb stem instead of 'st', if the last character of a verb stem is a 't'. Wiktionary uses a MediaWiki extension called ParserFunctions, which allows the use of control flows, like if-statements and switch statements. Special cases for the conjugation of verbs and the declension of adjectives are covered by a nested control flow. We have analyzed these templates and re-implemented the template of the adjectives and the most frequently used templates for verbs into IWNLP as C# code. In total, Wiktionary currently contains 3705 verb conjugations in the Flexion namespace, which use several templates. We have limited our implementation to the three most used verb conjugation templates (weak inseparable (51.4 %), irregular (27.2 %), regular (12.4 %)).

Altogether, we have extracted 74254 different words and 281457 different word forms. To reduce errors while parsing, we have written more than 150 unit tests to ensure that our parser operates as accurate as possible on various notations and special cases. During the development of IWNLP, we have manually corrected more than 200 erroneous Wiktionary articles, which contained wrong syntax or false content. To guarantee that we didn’t worsen the quality of these articles, we’ve consulted experienced Wiktionary users before performing these changes.

Our parser and its output will be made available under an open-source license.6

4 Lemmatization

Wiktionary can be used as a resource for multiple NLP tasks. Currently, we are interested in using Wiktionary as a resource for a lemmatization task, where we want to determine a lemma for a given inflected form. For each lemma, Wiktionary lists multiple inflected forms. As outlined in section 3, we have parsed the inflected forms for each lemma. For our lemmatization task, we inverse this mapping to retrieve a list of possible lemmas for a given inflection, hence our project name IWNLP. For example, we use the information presented in Table 1 to retrieve Türm → Turm. For each lemma l in Wiktionary, we have also added a mapping ℓ → l. Our mapping will also be available via download.

It is possible, that an inflected form maps to more than one lemma. For instance, the word Kohle maps to Kohle (coal) and Kohl (cabbage). In total, our mapping contains 2035 words, which map to more than one lemma.

With this paper, we want to evaluate how good Wiktionary performs in a lemmatization task. Additionally, we want to validate our assumption, that by first looking up word forms and their lemmas in Wiktionary, we should be able to improve the performance of existing software for lemmatization.

Therefore, we evaluate IWNLP and existing software on three German corpora, which list words and their lemmas: TIGER Corpus (Brants et al., 2004), Hamburg Dependency Treebank (HDT) (Foth et al., 2014) and TüBa-D/Z (Telljohann et al., 2012) release 9.1. The TIGER Corpus consists of 50472 sentences from the German newspaper Frankfurter Rundschau. The Hamburg Dependency Treebank (part A) contains 101981 sentences from the German IT news site Heise online. The TüBa-D/Z corpus comprises of 85358 sentences from the newspaper die tageszeitung (taz). Each word in these corpora is listed with its part-of-speech tag from the STTS tagset (Schiller et al., 1999). We evaluate the lemmatization for nouns (POS tag NN), verbs (POS tags V*) and adjectives (POS tags ADJA and ADJD). Due to the low amount of different articles and pronouns in the German language, we ignore them in our evaluation.

In our experiments, we look up the nouns, verbs and adjectives from each corpus in IWNLP. If we map a word form to more than one lemma in IWNLP, we treat this case as if there would be no entry for this particular word form in IWNLP. The same policy is applied in all of our experiments. We preserve case sensitivity, which worsens our results slightly. In a modification, that we name keep, we assume that a word w will be its own lemma, if w does not have an entry in the mapping. IWNLP is compared with a mapping7 excerpted from www.danielnaber.de/morphologie/index_en.html

6http://www.iwnlp.com

7http://www.danielnaber.de/morphologie/index_en.html
Table 2: Lemmatization accuracy for nouns, verbs and adjectives in all three corpora

<table>
<thead>
<tr>
<th>Method</th>
<th>TIGER Corpus</th>
<th></th>
<th>TüBa-D/Z</th>
<th></th>
<th>HDT</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Noun</td>
<td>Verb</td>
<td>Adj</td>
<td>Noun</td>
<td>Verb</td>
<td>Adj</td>
</tr>
<tr>
<td>IWNLP</td>
<td>0,734</td>
<td>0,837</td>
<td>0,633</td>
<td>0,720</td>
<td>0,809</td>
<td>0,567</td>
</tr>
<tr>
<td>IWNLP + keep</td>
<td><strong>0,894</strong></td>
<td>0,854</td>
<td>0,692</td>
<td>0,897</td>
<td>0,827</td>
<td>0,650</td>
</tr>
<tr>
<td>Morphy</td>
<td>0,196</td>
<td>0,713</td>
<td>0,531</td>
<td>0,181</td>
<td>0,671</td>
<td>0,490</td>
</tr>
<tr>
<td>Morphy + keep</td>
<td>0,857</td>
<td>0,962</td>
<td>0,763</td>
<td>0,860</td>
<td>0,916</td>
<td>0,744</td>
</tr>
<tr>
<td>Mate Tools</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>0,926</td>
<td>0,927</td>
<td><strong>0,852</strong></td>
</tr>
<tr>
<td>TreeTagger</td>
<td>0,860</td>
<td><strong>0,974</strong></td>
<td>0,867</td>
<td>0,848</td>
<td>0,930</td>
<td>0,832</td>
</tr>
<tr>
<td>IWNLP + Mate Tools</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td><strong>0,943</strong></td>
<td>0,929</td>
<td>0,841</td>
</tr>
<tr>
<td>Morphy + Mate Tools</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>0,918</td>
<td><strong>0,932</strong></td>
<td>0,837</td>
</tr>
<tr>
<td>IWNLP + TreeTagger</td>
<td>0,888</td>
<td>0,969</td>
<td><strong>0,869</strong></td>
<td>0,879</td>
<td>0,927</td>
<td>0,795</td>
</tr>
<tr>
<td>Morphy + TreeTagger</td>
<td>0,859</td>
<td>0,970</td>
<td>0,810</td>
<td>0,843</td>
<td>0,926</td>
<td>0,787</td>
</tr>
</tbody>
</table>

For our comparison with existing software, that can be used for lemmatization, we have chosen Mate Tools (Björkelund et al., 2010) and TreeTagger (Schmid, 1994), which both accept token-based input.

The results of our experiments are shown in Table 2. In a direct comparison between IWNLP and Morphy, IWNLP outperforms Morphy in the basic variant in all POS tags across all corpora. With the modification keep, the results of IWNLP and Morphy improve. IWNLP + keep is still superior for nouns, but Morphy + keep achieves better results for verbs and adjectives. The results from Mate Tools on the TIGER Corpus are excluded from Table 2 because Mate Tools was trained on the TIGER Corpus and, therefore, cannot be evaluated on it. The direct comparison of Mate Tools and TreeTagger shows that Mate Tools achieves an accuracy that is at least 2% better in four of the six cases. In the other two cases, TreeTagger only performs slightly better.

For the lemmatization of nouns, IWNLP is able to improve on the results of Mate Tools and TreeTagger across all three corpora. In total, IWNLP enhances the results of Mate Tools in five of the six test cases. Surprisingly, the additional lookup of word forms in IWNLP and Morphy can impair the accuracy for verbs and adjectives. In our future work, we will systematically analyze which words are responsible for worsening the results, correct their Wiktionary articles and improve our lookup in IWNLP.

The overall bad performance for the lemmatization of nouns in the HDT corpus can be explained by the gold lemmas for compound nouns, which are often defined as the last word in the compound noun. For instance, HDT defines that Freiheit (freedom) is the gold lemma for Meinungsfreiheit (freedom of speech).

5 Conclusion

We have presented IWNLP, a parser for the German version of Wiktionary. The current focus of the parser lies in the extraction of inflected forms. They have been used to construct a mapping from inflected forms to lemmas, which can be utilized in a lemmatization task. We evaluated our IWNLP lemmatizer on three German corpora. The results for the lemmatization of nouns show that IWNLP outperforms existing software on the TIGER Corpus and can improve their results on the TüBa-D/Z and the HDT corpora. However, we have also discovered that we still need to improve IWNLP to get better results for the lemmatization of verbs and adjectives. We will try to resolve the correct lemma for an inflected form if multiple lemmas are possible.

Additionally, IWNLP will be extended to parse hyponyms and hypernyms for nouns. We plan to compare the use of Wiktionary as thesaurus with GermaNet (Hamp and Feldweg, 1997).

We expect that the presented results for the lemmatization task will improve with every new monthly dump if Wiktionary continues to grow and improve through a community effort.

Acknowledgments

This work is part of the graduate school *NRW Fortschriftskolleg Online-Partizipation*. We thank...
the Wiktionary user Yoursmile for his help.

References


