Creating a Digital Bokmål-German Dictionary

Term Paper for
Computational Lexicography on the Web 2018-19

Verena Blaschke

September 16, 2019

Contents

1 Introduction

There are few existing digital tools for Norwegian, including dictionaries from Norwegian to German. Those that do exist are designed for a very straightforward look-up of lemmas. The dictionary built in the context of this course project however provides additional functionalities:

- Looking up words in a text without breaking the reading flow of the user,
- Lemmatization, also of irregularly inflected entries (which, due to umlaut mutations, is not necessarily trivial for a learner of the language),
- Compound splitting,
- Providing sample sentences to present entries in additional context,
- Pronunciation information.

This online dictionary is intended as a 'passive' dictionary that helps German speakers understand text written in Norwegian (Bokmål, see the following section) (cf. Svensén, 1992, p. 16). It is built using the Google Web Toolkit (GWT) framework.

In the following, I first present which input data my application is based on and how the data are preprocessed and combined into a single type of dictionary entry. Then, I describe how these entries are queried and displayed to the user. Lastly, I discuss this application.

1.1 A Note on Norwegian Written Standards and Pronunciation

Norwegian does not have a spoken standard language and speakers usually use their native dialects in all situations, including formal situations that involve interacting with speakers of other dialects (cf. Mæhlum and Røyneland, 2012, p. 135). The dialect with the largest number of speakers is the language variety spoken in and around Oslo, Standard East Norwegian (Mæhlum and Røyneland, 2012, pp. 60–61, 136). While the correspondence of the spelling and pronunciation in a given dialect is generally straight-forward in Norwegian, there are some non-trivial cases for learners, e.g. the Norwegian pitch accent or the pronunciation of syllable-final alveolar consonants.

Norwegian has two official written standards, Bokmål and Nynorsk, although non-official standards (such as riksmål) exist as well and many people write in dialect-based non-standard versions in informal or poetic contexts. Bokmål developed out of written Danish and while it does not directly correspond to any Norwegian dialect, it is closest to Standard East Norwegian (Mæhlum and Røyneland, 2012, pp. 60–62, 136). Nynorsk on the other hand is mostly based on rural Norwegian dialects spoken in the south-west and the middle of the country (cf. Mæhlum and Røyneland, 2012, p. 15). It is the preferred written standard of a minority of Norwegians, around 10% of the population (Hellevik, 2001). Both written standards have been revised several times and both of them offer a huge freedom of choice in terms of phonology, morphology and word choice (cf. Fjeld, 2015; Helset, 2016). This yields a continuum of written forms of Norwegian: conservative/moderate Nynorsk – radical Nynorsk (closer to Bokmål) – radical Bokmål (closer to Nynorsk) – moderate/conservative Bokmål (closer to written Danish) (cf. Helset, 2016; Fjeld, 2015).

2 Dictionary Entries

2.1 Data

The data I used to built this dictionary comes from several different sources. The Norwegian-to-German entries are largely based on entries from Langenscheidts Universalwörterbuch Norwegisch (Bokmål) and from the open online dictionary deno.dict.cc.[1] Using multiple Bokmål-German sources means that the resulting application can provide information on more lemmas and word senses than either source dictionary, and that a huger range of radical and moderate/conservative Bokmål forms are covered. Inflection information is sourced from Norsk Ordbank,[2] a lexical database compiled by the Norwegian National Library. The sample sentences are taken from the Tatoeba parallel corpus.[3] I used another parallel corpus, OpenSubtitles[4] (Lison and Tiedemann, 2016), as training data for automatically inferred entries for lemmas outside the two curated dictionaries' scopes. All of these data are stored in subfolders within `src/main/webapp/WEB-INF`.

The different data sources are parsed separately and then combined afterwards. The Java classes mentioned in this section are located in the `de.ws1819.colewe.server` and `shared` packages. In `Listener.contextInitialized`, the results from the steps detailed in this section are added to the servlet context, such that they can be accessed once the application is compiled.

2.2 Entry Format

The dictionary entry format used in the application (`...shared.Entry`) contains the following information:

- Lemma,
- Part-of-speech tag,
- A set of all known inflected forms, used when looking up inflected words,
- Irregularly inflected forms that should be displayed to the user,
- Translational equivalents (polysemes or homophones),
- Additional grammatical information,
- Usage and domain labels,
- A list of known abbreviations of the lemma,
- Ordbank's ID for the lemma,
- Collocations that include the lemma,
- Associated sample sentences.

Some of this information is included in several or all data sources (e.g. lemma, translational equivalents), whereas other information comes only from a single source. How the different data sources contribute to the dictionary entries in the final application is described in the subsequent subsections.

[1] `https://deno.dict.cc`

[2] `https://www.nb.no/sprakbanken/show?serial=oai%3Anb.no%3Asbr-5`

[3] `https://tatoeba.org/downloads`

[4] `http://opus.nlpl.eu/download.php?f=OpenSubtitles/v2018/moses/de-no.txt.zip`

Translational equivalents (`...shared.TranslationalEquivalent`) can also include usage or domain labels and abbreviated forms. Each translational equivalent can store multiple (synonymous) German translations.

The input dictionaries are not uniform in whether or how they discern between polysemy and homography (or how they encode cases where there exist multiple German synonyms for one sense of a Norwegian word). Therefore, I opted for a format following the formal-grammatical approach (cf. Svensén, 2009, p. 100) in that I group together all translational equivalents for one word form that share a part-of-speech tag under a single lemma. This also means that the resulting dictionary is microstructure-oriented (cf. Svensén, 2009, p. 365).

2.3 Langenscheidt: Entries

The digitalized version of Langenscheidts Universalwörterbuch includes about 15,000 Norwegian headwords. Each headword is accompanied by one or more German translational equivalents, as well as Norwegian pronunciation information (given in X-SAMPA) and part-of-speech information. Entries also include give information on irregularly inflected forms (with or without pronunciation information) and grammatical details(such as gender, number, case or tense).

The front matter of the dictionary does not give any details about which Bokmål forms were chosen or why, and there are no notes on the pronunciation information either. The pronunciation is given on a phonemic level, but it appears to be modelled on Standard East Norwegian. The morphology follows mostly moderate Bokmål (feminine nouns are often given masculine articles and are inflected like masculine nouns, and preterite or perfective forms of a specific verb class end in "-et" rather than "-a"). In terms of phonology and word choice, the dictionary is mixed although there seems to be a tendency towards moderate Bokmål (e.g., moderate "selv" *self* is preferred over radical "sjøl", moderate "kurv" *basket* is preferred over radical "korg", only the moderate version of the word "hensyn" *regard* (radical: "omsyn"), but however the radical "stein" *stone* instead of moderate "sten").

The Langenscheidt dictionary is processed in `DictionaryReader.readLangenscheid`. Each entry in the input file consists of three components: the lemma plus pronunciation and optionally inflected forms, the part-of-speech tag plus optional additional grammatical and usage information, and the translational equivalent(s). I extract the lemma and then transform the pronunciation information from X-SAMPA to IPA. If any inflected forms are given, I include them as irregularly inflected forms that should be shown to the user. The way I parse part-of-speech information is described in section 2.6.1. The additional grammatical information is added to the new dictionary entry if it does not indicate that the lemma itself is inflected. I skip inflected entries since they are indirectly included in entries for uninflected headwords (see section 2.5). When a German translation is divided into multiple senses, multiple translational equivalents are added to the new entry. When there are multiple German translational equivalents that are indicated as synonyms of one another describing the same meaning of the Norwegian lemma, they are considered parts of a single `TranslationalEquivalent`.

2.4 Dict.cc: Entries

The Bokmål-German dictionary deno.dict.cc is an example for bottom-up lexicography (cf. Svensén, 2009, p. 449), where users contribute and edit entries. It is presumably intended as both a reception and production dictionary for native speakers of either language.

Each source entry contains only one German translational equivalent per entry; polysemous, homographic and synonymous entries are all separate. There are more than 30,000 such entries. This is consequently also the case for the entries resulting from the first processing step (`DictionaryReader.readDictCc`), in contrast to the entries generated from the Langenscheidt dictionary.

In addition to a Norwegian lemma and a German translational equivalent, dict.cc entries can optionally include a part-of-speech tag, usage information, abbreviated forms, information on noun gender and domain labels. I extract all of this information except for the grammatical gender of German nouns, since this is not listed by the Langenscheidt entries either and either unnecessary for the intended user of my application or (in cases where noun gender makes a meaning distinction in German) apparent because of the other translational equivalents or the domain information.

If the lemma is a term for a person that can have a gendered suffix in German but not Norwegian, I remove information intended for Norwegian speakers to help them differentiate between the gendered German forms. The goal is to merge, e.g., *student [kvinnelig] - Studentin* and *student [mannlig] - Student* into a single entry *student - Studentin, Student* without any gender-based labels, since these are unnecessary in a Norwegian reception dictionary for speakers of German.

2.5 Ordbank: Inflection Information

The lexical database Norsk Ordbank contains information on almost 150,000 lemmas. For the purposes of this project, two files from this database are relevant: `lemma.txt` and `fullformsliste.txt`. The first file contains a list of Norwegian lemmas and corresponding numerical IDs. The second file contains inflected word forms, which are accompanied by the associated lemma's ID and details on the inflection. Each lemma has a unique set of inflected word forms. These two files are processed in `DictionaryReader.readOrdbankLemmas` and `readOrdbankFullformsliste`. The result is one dictionary entry per lemma that includes a set of inflected forms.

If a preterite, perfective, comparative, superlative or plural form of a word is irregular, it is stored as a form that should be displayed to the user. Ordbank does not provide any details on whether a given word form is regular or not, therefore I determine this in the methods `Tools.isRegularPret`, `isRegularPerf`, `isRegularComparative`, `isRegularSuperlative` and `isRegularPlural`. The rules for determining the different (regular) verb classes and their corresponding past/perfective forms are based on Faarlund et al. (2002, pp. 482–485).

The ordbank database does not include genitive forms of nouns. These are therefore added explicitly in `readOrdbankFullformsliste`.

2.6 Merging Entries

At this stage, there exists one map from lemmas to dictionary entries per input source. The three maps are merged in `ProcessResources.generateEntries`. Two entries can be merged if their lemma is identical and, in case both entries have part-of-speech tags, if their part-of-speech tags are identical. After merging, the dictionary contains about 29,000 distinct entries. The merged entries are stored in a map from inflected word forms to entries. These inflected forms are normalized by removing or replacing non-standard characters and removing ellipses and placeholders like "noen" *somebody*.

While merging the entries, I also prepare a collection of prefixes and a collection of suffixes that can be used for compound splitting. These (bound) affixes are not added to the general collection of dictionary entries used for direct look-up. More details on compound splitting are in section 3.3.

A small number of high-frequency words cannot be merged adequately because their part-of-speech assignment does not match across source dictionaries. Because of this, I only keep one source dictionary's entry for the infinitive marker "å" (deno.dict.cc), pronouns (Langenscheidt) and the indefinite article "en" (Langenscheidt).

2.6.1 Part-of-speech Tags

All three main input sources use different tag sets. The part-of-speech tags I decided to use are based on the parts of speech in the Norwegian reference grammar by Faarlund et al. (2002, pp. 27–30), except that I use a single category for both coordinating and subordinating conjunctions since deno.dict.cc does not distinguish between them, and that I introduced categories for sentences, prefixes and suffixes. Table 1 contains an overview of the tag set used in this project as well as the ones used in the input data.

2.6.2 Collocations

While merging the entries, I also extract collocation information. If a lemma consists of several words, it is added as collocation to all of these individual words' entries, so long as the words do not belong to a functional part of speech (conjunctions, determiners, prepositions or pronouns). A collocation is therefore entered under the base and all collocators, regardless of whether its meaning is transparent or not (cf. Svensén, 2009, pp. 162, 168–170). What I call *collocation* here also includes idioms and any other kind of fixed word combination (cf. Svensén, 2009, pp. 189–192) as well as particle verbs.

For example, the entry for "å gå fra sans og samling" *to lose consciousness, to be out of one's mind* is added as collocation for the lemmas "å gå" *to go*, "sans" *sense* and "samling" *collection*.

2.7 Tatoeba: Sample Sentences

Most of the sample sentences are sourced from the Tatoeba corpus. Since extracting the Bokmål–German entries can take several minutes, this is carried out in advance in a class that is not connected to the GWT application (`PreprocessTatoeba`). This leaves circa 5,400 sentence pairs that are read into the application in `DictionaryReader.readTatoeba`. In addition, full-sentence entries from the two source dictionaries are also treated as sample sentences.

I add sample sentences to the entries in the same way as I added the collocations: Each entry for a content word contained in the sentence gets a reference to the relevant `...shared.SampleSentence` object. In case a content word in a sentence is not one of the dictionary entries, the sample sentence is saved in a map from words to sample sentences. The use case of this map is explained in section 3.5.

2.8 OpenSubtitles: Fall-back Option

As an additional source for dictionary entries, I use translational equivalents for single-word lemmas inferred by a machine translation tool. The training data consist of the Bokmål-German entries in the OpenSubtitles corpus (Lison and Tiedemann, 2016). They are preprocessed in

POS	Langenscheidt	deno.dict.cc	Ordbank
adjective	A	adj past-p [past participle] pres-p [present participle]	adj
adposition	PRP [preposition] CCP [circumposition]	prep	prep
adverb	ADV FADV [question word adv.] RADV [response adverb]	adv	adv
conjunction	CNJ [coordinating conj.] SBJ [subordinating conj.]	conj	konj [coord. conj.] sbu [subord. conj.]
determiner	DET FDET [question word det.] FNUM [question word num.] NUM [numerical]	pron [no POS tag]	det
interjection	ITJ FITJ [question word interj.]	[no POS tag]	interj
noun	N NE [named entity] NM [typo for N]	noun	subst
prefix	PFX	prefix	pref i sms [word form used in compounds]
pronoun	PRN FPRN [question word pron.] RPRN [response pronoun]	pron	pron
sentence in database]	S FS [question]	[no POS tag]	[no sentences
suffix	SFX	suffix	[the resulting word's POS tag]
verb	V VI [intransitive verb] VR [typo for VT] VT [transitive verb] VTT [ditransitive verb]	verb	verb

Table 1: The tag sets used in this application (left column) and the data sources.

src/main/webapp/WEB-INF/opensubtitles/preprocess.py. This includes case normalization, tokenization and lemmatization. The Norwegian input is processed using the NLTK library (Bird et al., 2009).[5] I use the spaCy library (Honnibal and Montani, 2017) to tokenize and lemmatize the German input. The bilingual word alignment is then carried out by GIZA++ (Och and Ney, 2003), as implemented at https://github.com/moses-smt/giza-pp.

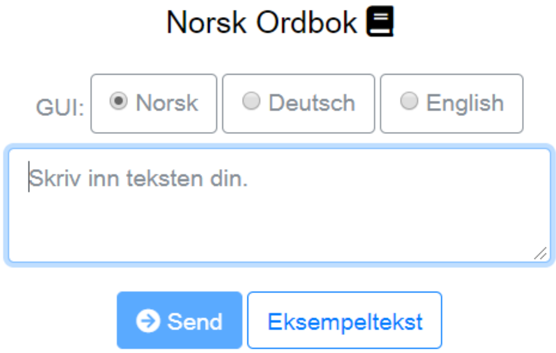
The result is a translation probability table containing at least one potential German translational equivalent per Norwegian entry, such that each potential translation is accompanied by a translation probability. This table is processed in PreprocessOpenSubtitles. I use all entries that have a translation probability of at least 0.5 to filter out unlikely translations. This leaves over 270,000 automatically inferred entries.

3 Processing Search Queries and Displaying the Results

3.1 Graphical User Interface

The graphical user interface consists mainly of GWT widgets that are styled using Bootstrap[6] and Font Awesome.[7] The widgets are contained in the de.ws1819.colewe.client package.

The input page lets the user choose between three display languages (Norwegian, German and English), enter a word or text and press a send button (see Figure 1). Furthermore, the user has the option of working with a sample text instead of entering their own input. This sample text is the first paragraph of the Norwegian Wikipedia entry on dictionaries.[8]



GUI: ● Norsk ○ Deutsch ○ English

Figure 1: The input page, consisting of a HeaderWidget ("Norsk Ordbok" plus book icon) and an InputWidget (everything else).

Clicking on the send button triggers a switch to the OutputWidget (see Figure 2). This widget contains the user's text input. Each word (whitespace-separated string) is contained in a WordWidget. When the user clicks on one or more WordWidgets, the dictionary is queried for the selected word(s). More details on this are given in the following subsection.

The results are displayed at the bottom of the page, in one EntryWidget per dictionary entry. At the core of each EntryWidget lies a SimpleEntryWidget containing the lemma, its transitional

[5] Due to a lack of openly accessible NLP tools for Norwegian, it is actually stemmed, not lemmatized, using NLTK's snowball stemmer for Norwegian: http://snowball.tartarus.org/algorithms/norwegian/stemmer.html.

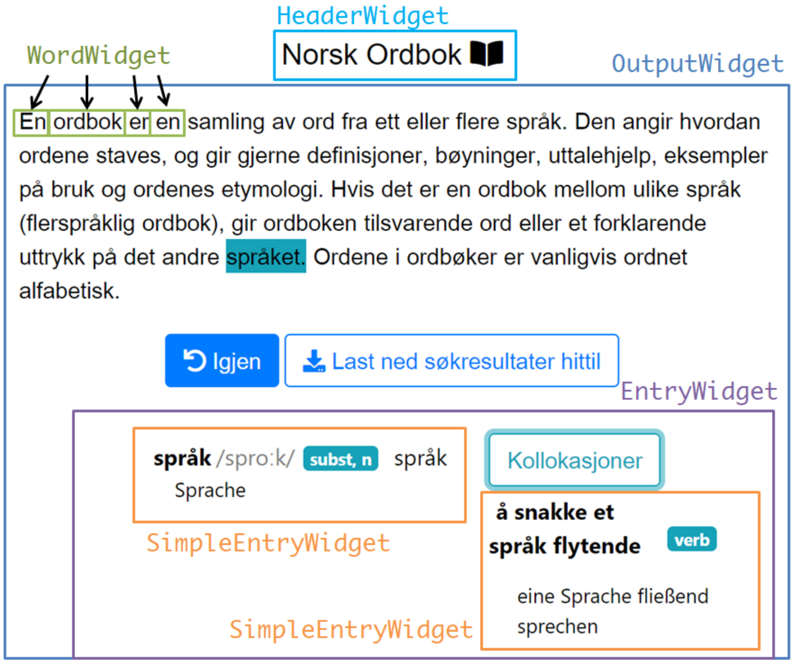
[6] https://getbootstrap.com/

[7] https://fontawesome.com

[8] https://no.wikipedia.org/wiki/Ordbok, CC BY-SA 3.0, last accessed Aug 25th, 2019.

equivalents and (when available) its part-of-speech tag and other grammatical details, a phonemic transcription, inflected forms, abbreviated forms and domain or usage information. The structure of a `SimpleEntryWidget` is shown in Figure 3. If there are collocations and/or sample sentences associated with the lemma, they are contained as `SimpleEntryWidget`s within collapsed lists that are also part of the `EntryWidget`.

The microstructure is thus similar to that of a typical print dictionary, albeit less condensed (cf. Svensén, 2009, pp. 345–347). The lemma, set off in bold type, introduces the entry and is followed by a formal section introducing grammatical and pronunciation information. Visually separate are the translational equivalents in the semantic-pragmatic section and the collocations and sample sentences in the contextual section. Having a separate contextual section means that this is an example of an unintegrated microstructure in polysemous entries (cf. Svensén, 2009, pp. 354).

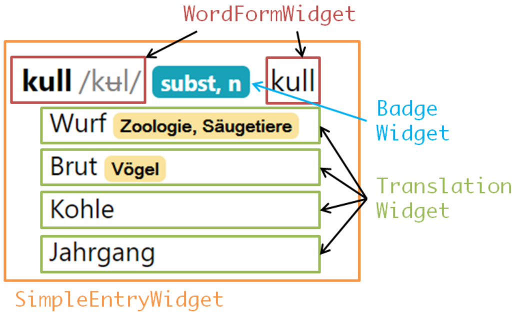


En ordbok er en samling av ord fra ett eller flere språk. Den angir hvordan ordene staves, og gir gjerne definisjoner, bøyninger, uttalehjelp, eksempler på bruk og ordenes etymologi. Hvis det er en ordbok mellom ulike språk (flerspråklig ordbok), gir ordboken tilsvarende ord eller et forklarende uttrykk på det andre språket. Ordene i ordbøker er vanligvis ordnet alfabetisk.

språk /spro:k/ subst, n språk
Sprache

å snakke et språk flytende verb
eine Sprache fließend sprechen

Figure 2: The widget structure of the output page given the sample text as input, after selecting the word “språket” and expanding the collocations list (cut off in this screenshot).



kull /kʉl/ subst, n kull
Wurf Zoologie, Säugetiere
Brut Vögel
Kohle
Jahrgang

Figure 3: The widget structure of a `SimpleEntryWidget`.

3.2 Querying the Dictionary

Clicking on a `WordWidget` fires a remote procedure call. A normalized version of the selected word (lower case, no special characters) is transmitted to the server-side `DictionaryServiceImpl`. From there, a list of dictionary entries is returned. In the simplest case, the word can simply be looked up in the map from inflected word forms to `Entry` objects (`querySingleWord`). Other cases are listed in the following subsections.

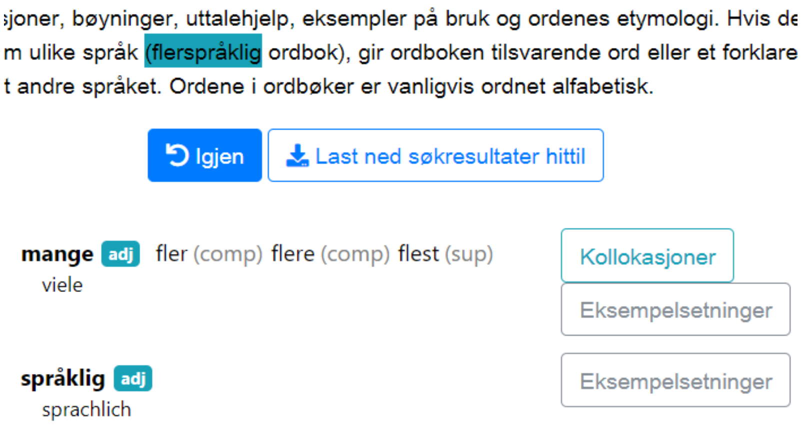
3.3 Compound Splitting

If the queried word is not in the dictionary, it might be a compound word whose stems however might be contained in the dictionary. Combinations of stems and derivational affixes are processed the same way, although derivational affixes are treated as entries that do not necessarily need to have translational equivalents.

The last element in a Norwegian compound word can be inflected. If an earlier element is an adjective, it can be inflected by gender, number and degree (although this is not always the case) (Johannessen, 2001, pp. 62–63). If it is a verb or noun, it is always uninflected (Johannessen, 2001, pp. 65–66), but a noun can be followed by a linking "-s-" or "-e-". Generally, it is not possible to predict whether a noun is followed by "-s-", "-e-" or no infix at all, unless the noun ends in one of a small numbers of specific suffixes (Mac Donald, 2014, pp. 31–32). Combinations of nouns and linking morphemes are generated while merging the entries in `...server.ProcessResources` and stored together with prefixes listed in the input dictionaries.

In `queryWithPossibleSplit`, the search term is split into two strings until the first string matches an uninflected word form in the dictionary or a prefix. If the second term does not match a suffix or a word form in the dictionary, it is similarly (recursively) split. If there are still no matches, this procedure is repeated with a different initial split.

All potential stems (or derivational morphemes) contained in the search term are displayed to the user. Figure 4 shows an example.



sjoner, bøyninger, uttalehjelp, eksempler på bruk og ordenes etymologi. Hvis de
m ulike språk (flerspråklig ordbok), gir ordboken tilsvarende ord eller et forklare
t andre språket. Ordene i ordbøker er vanligvis ordnet alfabetisk.

mange adj fler (comp) flere (comp) flest (sup)
viele

språklig adj
sprachlich

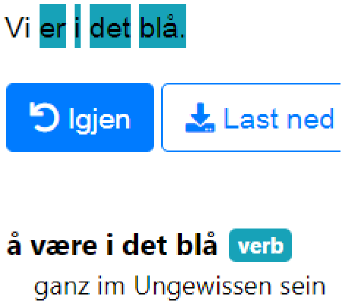
Figure 4: Searching for an out-of-vocabulary word, "flerspråklig" *multilingual*, returns entries for the stems.

3.4 Multi-word Phrases

The user can also look up multi-word phrases by keeping the `CTRL` key pressed while selecting multiple words. To keep track of the individual words, they are temporarily saved in a hidden `queryContainer` whose contents are reset every time a `WordWidget` is selected without the `CTRL` key being pressed at the same time. If the queried phrase is a lemma in the dictionary, the corresponding entry is shown to the user. Otherwise, the mapping from inflected forms to lemmas is used to lemmatize the words contained in the search term (`queryMultiWordPhrase`) and, if available, the entry for the lemmatized query is displayed. An example can be found in Figure 5 below.

3.5 Fall-back Options for Missing Entries

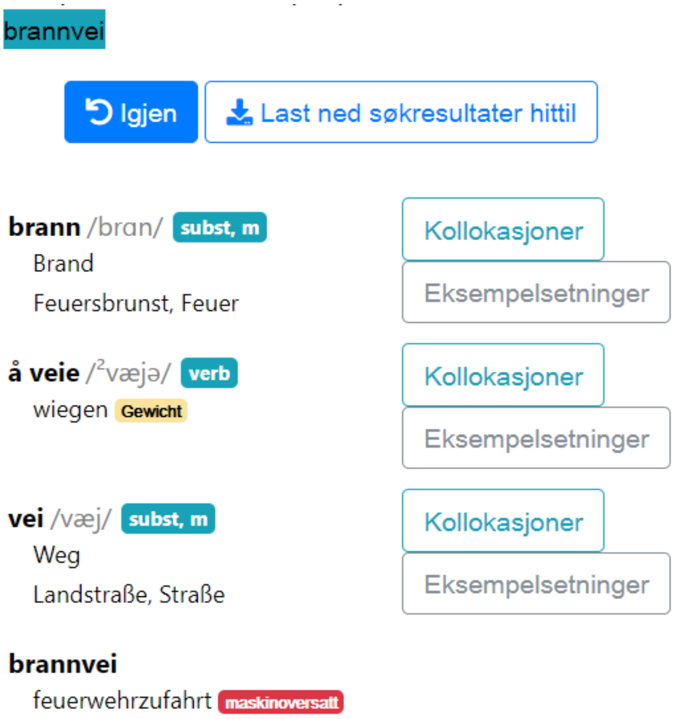
If a (single-word) search term cannot be looked up in the entries based on the Langenscheidt and deno.dict.cc data, I try to look it up in the collection of the automatically inferred translations. If there is a result, it is explicitly marked as machine-translated (see Figure 6). Additionally, if the search term is part of a sample sentence, the sample sentence is also displayed as an entry.



Vi er i det blå.

å være i det blå verb
ganz im Ungewissen sein

Figure 5: Looking up "er i det blå" (lit. *is in the blue*) yields the entry for "å være i det blå" (lit. *to be in the blue*).



brannvei

brann /bran/ subst, m
Brand
Feuersbrunst, Feuer

å veie /²væjə/ verb
wiegen Gewicht

vei /væj/ subst, m
Weg
Landstraße, Straße

brannvei
feuerwehrzufahrt maskinoversatt

Figure 6: If an out-of-vocabulary word is a lemma of one of the machine translation-based entries, this entry is added to the results.

3.6 Download Option

It is also possible to query the dictionary via a GET request. The relevant HTTP servlet is implemented in `..server.DownloadServiceImpl` and can be accessed like this: `http://127.0.0.1:8888/Ordbok/downloadService?query=vrang%26maske` (`%26` encodes `&`). In this example, *vrang* and *maske* are the lemmas that should be looked up. The results are saved in a text file that is downloaded on the client side. The contents of the file for the sample GET request look like this:

```
 QUERY: vrang

vrang /vrɑŋ/ (adj)
> verkehrt
> falsch
> unwillig
> links [beim Stricken]
> querköpfig
Collocations:
    å strikke to rette og to vrange (verb)
    > zwei rechts, zwei links stricken


vrang (noun) {m}
> linke Seite

##########################

QUERY: maske

maske /$^2$mɑskə/ (noun) {m, f}
> Masche
> Maske

##########################
```

The download option can also be accessed via the download button in the `OutputWidget`. Clicking on it triggers a GET request for all previously queried lemmas. To keep track of the query history, each queried lemma is saved in a hidden `historyContainer` on the page.

4 Conclusion and Discussion

I implemented a Bokmål-to-German dictionary that includes all of the features promised in the project proposal: the look-up of inflected forms, compound words and collocations, additional machine-translation-based entries and a GUI that displays diasystematic labels, grammatical information, irregular forms, pronunciation information, collocations and sample sentences. I also implemented to of the nice-to-have features: exporting search results via a file download and providing several GUI language choices.

There is one optional feature that I did not implement for lack of time: sorting entries by frequency in a reference corpus whenever several entries are displayed to the user. Similarly, while creating this application, I noticed several other features that would be nice to include or improve in a future version of this dictionary.

Firstly, a useful feature of many digital dictionaries are hyperlinks to other entries (for example, for words in multi-word phrases, collocations or sample sentences) (see also Svensén, 2009, pp. 443–445).

Furthermore, additional and/or more thorough preprocessing of some of the data might improve the dictionary. This includes part-of-speech-tagging and lemmatizing collocation candidates and sample sentences to be more accurate when linking them to lemmas. There is also room for improvement when it comes to the automatically inferred entries. Mainly, better preprocessing tools (lemmatization, normalization/spelling correction) for Bokmål would help here.

To make the application more comfortable to use, further improvements could include making it possible to update the input text without having to switch back to a (blank) input page, and including an about page with information on the application's features.

References

Langenscheidts Universal-Wörterbuch Norwegisch (Bokmål): Norwegisch–Deutsch, Deutsch–Norwegisch. 14th edition. Revised edition by Kjell Bjørnskau. Berlin: Langenscheidt, 1998.

Norsk Ordbank – Bokmål 2005. Last updated 2019-02-20. Oslo: Nasjonalbiblioteket, 2019. Available at `https://www.nb.no/sprakbanken/show?serial=oai%3Anb.no%3Asbr-5&lang=nb`.

Bird, S., E. Klein, and E. Loper (2009). *Natural language processing with Python: analyzing text with the natural language toolkit.* O'Reilly Media, Inc.

Faarlund, J. T., S. Lie, and K. I. Vannebo (2002). *Norsk referansegrammatikk* (3rd ed.). Universitetsforlaget.

Fjeld, R. V. (2015). Språklige varieteter eller språklige rariteter? Om bruk av valgfrie former i norsk bokmål. *LexicoNordica* (22), 35–55.

Hellevik, O. (2001). Nynorskbrukaren–kven er han? *Kampen for språket: Nynorsken mellom det lokale og det globale*, 117–139.

Helset, S. J. (2016). Tilhøvet mellom konservative, moderate og radikale former i nynorsk–ein studie av nynorskskrivaren sine språklege intuisjonar og val. *Maal og Minne 108*(1), 141–172.

Honnibal, M. and I. Montani (2017). spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing. To appear.

Johannessen, J. B. (2001). Sammensatte ord. *Norsk Lingvistisk Tidsskrift 19*(1), 59–92.

Lison, P. and J. Tiedemann (2016). Opensubtitles2016: Extracting Large Parallel Corpora from Movie and TV Subtitles.

Mac Donald, K. (2014). *Norsk grammatikk for læreren: Spørsmål og svar når norsk er andrespråk* (2nd ed.). Cappelen Damm.

Mæhlum, B. and U. Røyneland (2012). *Det norske dialektlandskapet: Innføring i studiet av dialekter.* Cappelen Damm akademisk.

Och, F. J. and H. Ney (2003). A systematic comparison of various statistical alignment models. *Computational Linguistics 29*(1), 19–51.

Svensén, B. (1992). Lexikografi och lexikografiska produkter: Några grundbegrepp. *Nordiske Studier i Leksikografi 1*, 9–27.

Svensén, B. (2009). *A Handbook of Lexicography: The Theory and Practice of Dictionary-Making*. Cambridge University Press.