Alexander Geyken and **Jordan Boyd-Graber**. **Automatic classification of multi-word expressions in print dictionaries**. *Linguisticae Investigationes*, 2003, 16 pages.

Automatic classification of multi-word expressions in print dictionaries[*]

Alexander Geyken and Jordan Boyd-Graber

Berlin-Brandenburg Academy of Sciences / California Institute of Technology

Introduction

The dictionary of German idioms (H. Schemann 1993) contains more than 32,000 entries and is thus the largest printed collection of German idioms. The notion of idioms is used in a very broad sense by the author; Schemann considers all expressions as idioms where a word is constrained by a larger context.[1] A context can be either linguistic (syntagmatic) or related to the pragmatic use of an entry. Particular emphasis is put on the presentation of the syntagmatic contexts for all entries. Hence, an entry is not a linear multi-word expression but rather a set of different syntagmatic contexts around the target word. The selections range from completely frozen expressions to multi-word expressions where all components are substitutable. The complexity of form varies from two-word expressions (see Example 1 below) to complex PP-VP expressions (see Example 2).

In this paper we do not intend to evaluate the linguistic pros and cons of this dictionary. Rather, our goal is to explore the usefulness of this paper dictionary for natural language processing (NLP). Previous work in the field of knowledge extraction and the creation of machine readable dictionaries (MRDs) has shown the difficulties of making use of paper dictionaries for NLP purposes (e.g. B. Boguraev (1989) and J. Klavans (1996)). Even though obviously paper dictionaries are intended for use by humans and not for NLP, idiom dictionaries contain interesting information for NLP, which can be extracted by automatic means.

The following examples illustrate the variety of entry structures found in Schemann's dictionary.

Ex 1   *einen zischen*
one   drink
to knock back a drink

Ex 2   *ein absolutes / das absolute Gehör haben*
an absolute / the absolute ear   have
to have perfect pitch

Ex 3   *(voll) auf jdn. / … abfahren*
(fully) on sb. / … go off
to really fancy somebody

These entries demonstrate the lexicographic shorthand used by Schemann: / to indicate a mutually exclusive choice between elements to the left and right of the symbol, ( ) to indicate an optional phrase that may be omitted, […] to designate an arbitrary clause, and placeholders such as **jdn.** to designate flexible components that admit substitution, in this case an accusative noun phrase. This variety of entry structures means that Schemann's dictionary is not immediately usable for automatic processing.

Therefore, some simple preprocessing steps are applied in order to transform this format into a more suitable machine-readable format. Entries with optional elements are rendered simultaneously as multiple phrases: a phrase with the optional element and a phrase without. Likewise, […] can be thought of as an unconstrained substitution. Hence, it is possible to remove the token altogether without loss of significant information.

Somewhat more difficult is the disambiguation of the lexical elements within the entries. The word *einen* may be interpreted as a determiner, a verb, or a cardinal number. Preprocessing can readily identify such placeholders, whose strict usage within Schemann limits their ambiguity. Significantly more difficult, however, is the resolution of the scope of the / operator. It is not clear in Example 2 whether the resolution should be [*ein absolutes*]/[*das absolute*] *Gehör haben* or *ein* [*absolutes/das*] *absolute Gehör haben.*

These examples illustrate that multi-word entries of a paper dictionary cannot be extracted as a simple string if one intends to exploit them for NLP purposes. Given the 32,000 entries and an average length of 4.3 tokens for each entry, manual association of each token to its part-of-speech tag (POS-tag) would not be feasible. On the other hand, parsing the entries of idiom dictionaries is not equivalent to the parsing of naturally occurring text, since the entries follow a formalized structure that is quite different from that of ordinary language. The tokens are generally lemmatized, and the verb is placed at the end

of the entry. Given this formal structure, the idea of using a Part-of-Speech tagger such as the Brill Tagger (E. Brill (1994)) for these entries seems quite promising. Parameter files of other taggers are not suitable for the task, but an appropriate way forward may be to encode a certain number of entries by hand, qualify them as a training corpus, and then train an appropriate tagger on these rules in order to obtain appropriate parameter files.

In the remaining sections the following questions are addressed:

–   To what extent in terms of completeness and correctness is it possible to associate dictionary entries with recurrent POS sequences?
–   How does the size of the training set influence the results?
–   How are different kinds of German multi-word expressions statistically distributed in a dictionary of idiomatic German?

1.   Method

As explained above, machine learning seems to be an appropriate approach for the task at hand. Hence, a training set was encoded on the basis of a very simple tagset consisting of 10 tags (see below). Each dictionary entry was then processed, creating a sequence of simpler strings such that all optional elements are separated into subcases and all relative clauses and unneeded information are removed. The Brill Tagger then generates a POS sequence corresponding to each of the entries. If these sequences in turn correspond to a construction class (see below), the entry is assigned to that construction class.

1.1   Resources

The following resources were used:

–   a list of 32,000 dictionary entries (H. Schemann 1993);
–   a small tagset: *Adj, Adv, Conj, Det, N, NA, Prep, Pron, Ptk, V*;
–   a training set: approximately 6,000 dictionary entries, tagged by hand;
–   a list of 230 different POS sequences that were identified in the training set.

1.2   Association with construction classes

Even after the multi-word entries in the dictionary have been normalized to a large extent, numerous cases still cannot be processed straightforwardly. The

expression after the right arrow illustrate the way Schemann's entries are preprocessed.

Ex 4   *steif und fest einbilden, daß …*
        stiff and fast imagine,   that …
        to have it stuck into one's head that …
   → *steif und fest einbilden*

Ex 5   *das / (etw.) ist ein dicker Hund*
        that / (sth.) is a fat   dog
        that's outrageous
   → *das ist ein dicker Hund*

Ex 6   *Mit jdm. Tacheles   reden*
        With sb.   Tacheles (yidd.) speak
        To talk straight with somebody
   → *mit NA Tacheles reden*

Ex 7   *Von A bis Z Unsinn   / erlogen / erfunden … sein*
        From A to Z nonsense   lied   invented   be
        Sth. is a pack of lies from A to Z
   → *von A bis Z Unsinn sein → von A bis Z erlogen sein → von A bis Z erfunden sein*

In Examples 4, 5, and 7, subordinate clauses, placeholders for noun phrases, and the unconstrained substitution […] were deleted. Only when the place-holders stand for a prepositional phrase were they preserved and replaced by the tag NA, as in Example 6. In example (7), the mutually exclusive choices / makes it necessary to decompose the initial Schemann entry into more than one entry.

Hereinafter, entries resulting from preprocessing are called elementary entries. The POS sequence associated with an elementary entry is called an elementary POS sequence. As stated above, the training set contained 230 different POS sequences, henceforth called construction classes.

After preprocessing, the way entries are associated with construction classes has to be characterized. We call an association of a dictionary entry with a construction class successful iff:

i.   there exists an isomorphism between the construction class of the training set and the computed (elementary) POS sequence, or

ii. a complex entry can be decomposed into several elementary entries and more than 50% of the computed POS sequences are isomorphic to a single elementary construction class.

There are two failure modes for this process. Either the computed POS sequence cannot be associated with any construction class or a complex entry can be decomposed in more than one elementary entry, but their corresponding elementary POS sequences or at least 50% of them do not belong to a single construction class (cf. examples below, in particular Example 9). The latter failure mode is quite conservative and the conditions could be weakened.

The following examples illustrate the association of a dictionary entry with a construction class:

Ex 8   *blinder Passager*
blind   passenger
stowaway

Ex 9   *das/(die Hoffnung) kann ich mir/kann Peter sich/…   abschminken*
that/(the hope)   can I me/can   Peter himself/… forget about
I/Peter/… can forget the idea/my/his/… hopes/… (of doing sth.)

Ex 10   *so richtig / anständig / … einen abrocken (gehen)*
so really / decently   / … one   rock up   (to go)
to really rock (as in Rock and Roll)

successful association (condition i) Examples 3, 4, 5, 6, and 8 are success cases: they can all be successfully associated with a single construction class:

– *Example 1:* $\rightarrow$ *det.V*[2]
– *Example 4:* $\rightarrow$ *adj conj adj V*
– *Example 5:* $\rightarrow$ *det V adj N V*
– *Example 6:* $\rightarrow$ *prep NA N V*
– *Example 8:* $\rightarrow$ *adj N*

successful association (condition i and ii) Example 7 is decomposed into the following three normalized entries. The sequence "prep N prep N V V" belongs to the construction classes in the training set, therefore condition **i** above is met. Furthermore, two out of three are in the same construction class (condition **ii**).

Ex 7.1   *von A bis Z Unsinn sein*
     $\rightarrow$ *prep N prep N N V*

Ex 7.2   *von A bis Z erlogen sein*
     $\rightarrow$ *prep N prep N V V*

Ex 7.3   *von A bis Z erfunden sein*
     $\rightarrow$ *prep N prep N V V*

Association failures Example 9 is decomposed into the following four different elementary entries.

Ex 9.1   *das kann ich mir abschminken*
        → *det V Pron Pron V*

Ex 9.2   *das kann ich mir Peter sich abschminken*
        → *det V Pron Pron N Pron V*

Ex 9.3   *die Hoffnung kann ich mir abschminken*
        → *det N V Pron Pron V*

Ex 9.4   *die Hoffnung kann ich mir Peter sich abschminken*
        → *det N V Pron Pron N Pron V*

Apart from the fact that the scope of the second (/) operator is underspecified, thus yielding wrong elementary entries, it is also a failure case according to condition **ii** since all the elementary POS sequences are different. Cases like Example 9 are the main motivation to introduce the second condition since it helps to sort out incorrect preprocessing results. In particular, decompositions such as 9.2 and 9.4 are generated by the preprocessor but, as any speaker of German knows, they are ungrammatical.

Example 10 is decomposed into the following four different elementary entries:

Ex 10.1 so richtig einen abrocken
        → *adv adv det V*

Ex 10.2 so richtig einen abrocken gehen
        → *adv adv det V V*

Ex 10.3 so anständig einen abrocken
        → *adv adv det V*

Ex 10.4 so anständig einen abrocken gehen
        → *adv adv det V V*

Examples 10.1 and 10.3 are in the construction class "adv adv det V" whereas 10.2 and 10.4 are in the construction class "adv adv det V V". Again, this example violates condition **ii** since there is no majority of POS sequences for a single construction class. As stated above, it would have been possible to weaken condition **ii**, but even better than that seems the possibility to define the POS sequences in a different way. Indeed, if we defined VP := V | V.V all these decompositions would be in the same construction class (see below, experiment 2).

2.   Experiment 1

In order to measure the impact the POS tagger has on the quality of the results, we conducted the experiment described below with a simple lookup in the Morphy lexicon (W. Lezius 1996), a fairly large German lexicon containing approximately 350,000 full forms. After conducting the preprocessing steps described above, we could associate only 35% of all Schemann entries success-fully according to the above mentioned conditions. 65% of all the cases were failure cases, i.e. either the computed POS sequences were not found in the training set (condition **i** above) or the computed POS sequences were too heterogeneous (thus violating condition **ii** above).

Then, we conducted the experiment with the Brill Tagger, which was trained on the basis of a training set of 6,000 manually tagged multi-word entries totalling 25,800 tokens. As a result, a tagger of 91 context rules, 96 lexical rules, and a lexicon of 5,341 different entries was generated. The following figure shows the accuracy of the trained tagger as a function of the size of the training set. With a training set of 1,000 entries we obtain an accuracy of 94%, which increases to 97.8% with a training set of 6,000 entries (J. Boyd-Graber 2002).
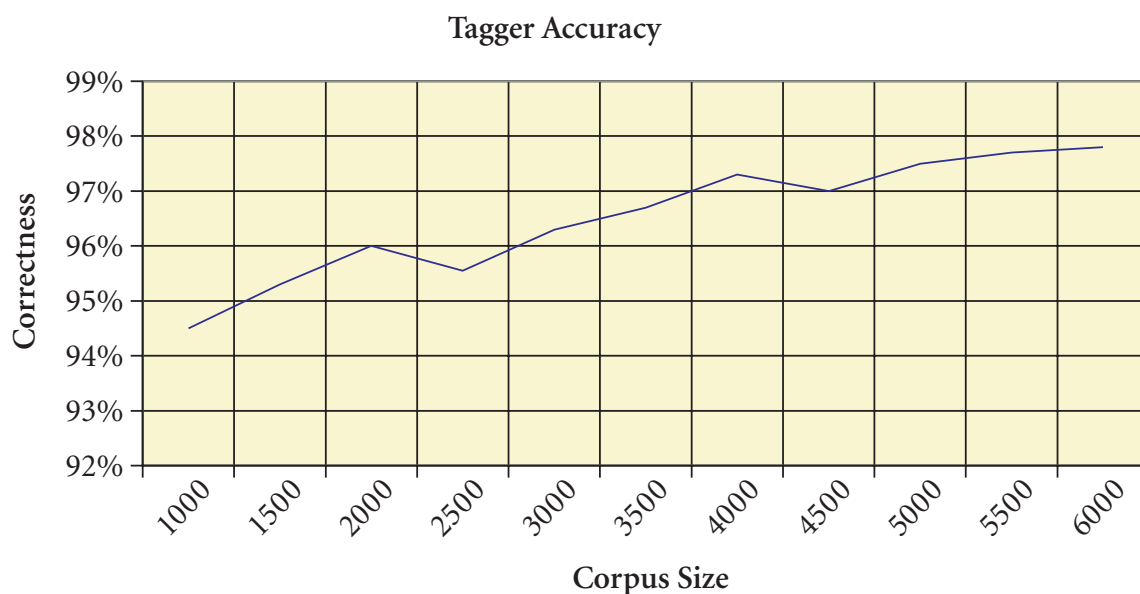

Figure 1. Tagger accuracy as a function of corpus size.

The first run of the model assignment program assigned 55% of the Schemann entries to POS-sequence models, which is better than mere lexicon lookup, but was deemed to be insufficient for use within the project. A rapid

inspection of the remaining unassigned sequences revealed the following assignable causes:

1. The entry is too complex. Most of these cases are due to the fact that the scope of the slash (/) is ambiguous: e.g. Example 7's scope is one word, whereas Example 2's is two words.
2. The computed POS sequences do not correspond to any of the construction classes identified in the training set.
3. Transcription errors due to typewriting or incorrect segmentation of the entries.

3.   Extension of the method

3.1   Correction and disambiguation of the slash

In addition to the manual correction of some entries because of transcription errors (see Figure 1 above) the scope of the slash (/) was disambiguated manually. This task required native knowledge of German but almost no linguistic knowledge. It took about 2 days to annotate the ambiguous cases using brackets.

3.2   Extension of the models

Experiment 1 showed that the 6,000 entries of the training set evidently did not contain the complete list of POS sequences of 32,000 multi-word entries in the dictionary. In a second experiment we wanted to verify if an *a priori* definition of construction classes would lead to a higher recall. In particular, the idea was to draw up a list of all *plausible* construction classes. Two remarks are necessary before describing the language of construction classes in more detail. First, even though the basis for this description is the dictionary of idiomatic German (H. Schemann 1993), the major part of this description should hold for other languages as well. Evidence for this can be found in C. Tschichold (1998), who describes a subset of the construction classes defined below in her description of English multi-word lexemes. Second, it should be kept in mind that phrase notations such as NP or AP or VP below do not correspond to real syntactic categories since they operate only on POS sequences, not on dependency trees.

In the following, the language of these plausible construction classes, called **K**, is described. Let **T** be the set of POS tags, $T := \{Adj, Adv, Conj, Det, N, Pron, Ptk, Rel, V\}$. The set **K** of all plausible construction classes is defined inductively as the union of the following POS sequences: $K := K_1 \cup K_2 \ldots \cup K_7$.

3.3 Overview

ID	Construction class	Description
K_0	Prep.N.Prep etc.	idiom specific sequences
K_1	$NP{:=}(N|Det.N|\ldots);PP{:=}..$	atomic phrases
$K_{1'}$	$P{:=}(NP|PP|AdjP)$	atomic phrases
K_2	$P_n{:=}P.(P)+$	atomic combinations
K_3	$P.V$	atoms and verbs
K_4	Conj.P := $Conj.P_n|Konj.P$	conjunction phrases
K_5	AdvP_n	adverbial phrases
K_6	AdvP.KonjP.P_n	combinations with $K_{4,5}$
K_7	$K_{1,2,4\ldots6}.VP$	complex verb combinations

3.4 Detailed description of the construction classes

1. K_0 idiom specific sequences (not iterating):
   a. $PP_0{:=}(Det|\varepsilon).(Prep|\varepsilon).(Adj|\varepsilon).N.Prep$
   b. $AP_0 := Adj.Prep$
   c. Pron.Det.V
   d. Prep.Conj.Prep.Pron
   e. Prep.N.Det
   f. Det.N.Prep.V
   g. $Det.N.Prep.(Pron|V)$
   h. Conj.Conj
   i. Det.N.Det.N.Prep
   j. Prep.Adj.N.Prep.V
   k. $Det.N.(V_{inf}|Rel)$
2. K_1: atomic phrases: AdvP, AP, NP und PP
   a. $AP{:=}(Adj|\varepsilon).Adj \cup Ptk.Adj$
   b. $NP{:=}(Det|\varepsilon).(Adj|\varepsilon).(Adj|\varepsilon).N \cup Pron$
   c. $PP{:=}(Prep|\varepsilon).Prep.(Det|\varepsilon).((Adv).Adj|\varepsilon).N \cup$ Prep.Adj $\cup$ Ptk.Prep.N
   d. $AdvP{:=}(Prep|\varepsilon).(Adv).(Prep|Adv|\varepsilon) \cup$ Adv.Prep.Adv $\cup$ Prep.Pron.(Adv|Prep)
3. K_2: simple combinations of atomic phrases
   a. $P_2{:=}(AP|NP|PP).AP|NP|PP)$
   b. $P_3{:=}(AP|NP|PP).(AP|NP|PP).(AP|NP|PP)$
4. K_3: simple combinations of atoms and verbs:
   a. $PV{:=}(AP|NP|PP).V$

5. K_4: combinations of phrases with conjunctions
   a. $ConjP:=(Conj|\varepsilon).Conj.(AP|NP|PP)$
   b. $PConjP:=(AP|NP|PP|PP_0|\varepsilon).Conj.(AP|NP|PP)$
   c. $ConjPConjP:=Conj.(AP|NP|PP).Conj.(AP|NP|PP)$
   d. $ConjPP:=Conj.(AP|NP|PP).(AP|NP|PP)$
   e. $PConjPP:=(AP|NP|PP|\varepsilon).Conj.(AP|NP|PP).(AP|NP|PP)$
   f. $PPConjP:=(AP|NP|PP|\varepsilon).(AP|NP|PP).Conj.(AP|NP|PP)$

6. K_5: combinations of phrases with adverbs
   a. $AdvPP:=AdvP.(AP|NP|PP) \cup (AP|NP|PP).AdvP$
   b. $PAdvPP:=(AP|NP|PP).AdvP$
   c. $AdvPPP:=AdvP.(AP|NP|PP).(AP|NP|PP)$
   d. $AdvPP_3:=AdvP.(AP|NP|PP).(AP|NP|PP).(AP|NP|PP)$

7. K_6: combinations of phrases, conjunctions and adverbs
   a. $AdvPConjP:=AdvP.ConjP$
   b. $AdvPPConjP:=AdvP.PConjP$
   c. $AdvPPConjP:=AdvP.Conj.AdvP$

8. K_7: complex verb phrase combinations, i.e. $K_{1,2,4,5,6}$ with
   VP$:=(V|\varepsilon).(V|\varepsilon).(V|\varepsilon).V \cup (V|\varepsilon).(V|\varepsilon).Conj.V.(Conj.V|\varepsilon)$
   a. $K_{7,2}$ — combinations of phrases with VP
      i. $PV:=(AP|NP|PP_0|PP).V.VP$
      ii. $P_2V:=(Pron)?.(AP|NP|PP).(AP_0|AP|NP|PP).VP$
      iii. $P_3V:=(AP|NP|PP).(AP|NP|PP).(AP|NP|PP).VP$
      iv. $PVP:=(Pron|\varepsilon).(AP|NP|PP).VP.(AP|NP|PP)$
      v. $PVP_2:=(Pron|\varepsilon).(AP|NP|PP).VP.(AP|NP|PP).(AP|NP|PP)$
      vi. $P_2VP:=(Pron|\varepsilon).(AP|NP|PP).(AP|NP|PP).VP.(AP|NP|PP)$
   b. $K_{7,4}$: combinations of conjunctive phrases with VP
      i. $ConjPV:=(Pron|\varepsilon).ConjP.VP$
      ii. $PConjPV:=(Pron|\varepsilon).PConjP.VP$
      iii. $ConjPPV:=Conj.(AP|NP|PP).(AP|NP|PP).VP$
      iv. $ConjPConjPV:=Conj.(AP|NP|PP).Conj.(AP|NP|PP).VP$
      v. $PConjPPV:=(AP|NP|PP|\varepsilon).Conj.(AP|NP|PP).(AP|NP|PP).VP$
      vi. $ConjPPPV:=Conj.(AP|NP|PP).(AP|NP|PP).(AP|NP|PP).VP$
      vii. $PPConjPV:=(AP|NP|PP).(AP|NP|PP).Conj.VP$
   c. $K_{7,5}$: combinations of adverb phrases with VP
      i. $AdvPPV:=AdvP.(AP|NP|PP|\varepsilon).VP \cup (AP|NP|PP|\varepsilon).AdvP.VP$
      ii. $AdvPP_2V:=AdvP.(AP|NP|PP|\varepsilon).(AP|NP|PP).VP \cup (AP|NP|PP).AdvP.(AP|NP|PP).VP$

      iii.   *AdvPP3V:=AdvP.(AP|NP|PP).(AP|NP|PP).(AP|NP|PP).VP*
      iv.   *P2AdvPV:=(AP|NP|PP|ε).(AP|NP|PP).AdvP.(AP|NP|PP).VP*

  d. $K_{7,6}$: combinations of conjunctive and adverb phrases with VP VP
      i.   *AdvPConjVP:=AdvP.Conj.VP*
      ii.   *AdvPConjPVP:=AdvP.ConjP.VP*
      iii.   *AdvPConjPPVP:=AdvP.ConjPP.VP*
      iv.   *AdvPPConjPPVP:=AdvP.PConjPP.VP*

4.   Experiment 2

Experiment 2 was conducted in the same way as experiment 1 except for the changes described in Sections 3.2–3.4.

    As a result of the revised model assignment program, the recall rose to 86% (26,703 out of 32,000), which corresponds to an improved recall of more than 30% with respect to experiment 1. We estimate that 15% is due to the slash corrections, the new construction classes accounting for the other 15%.
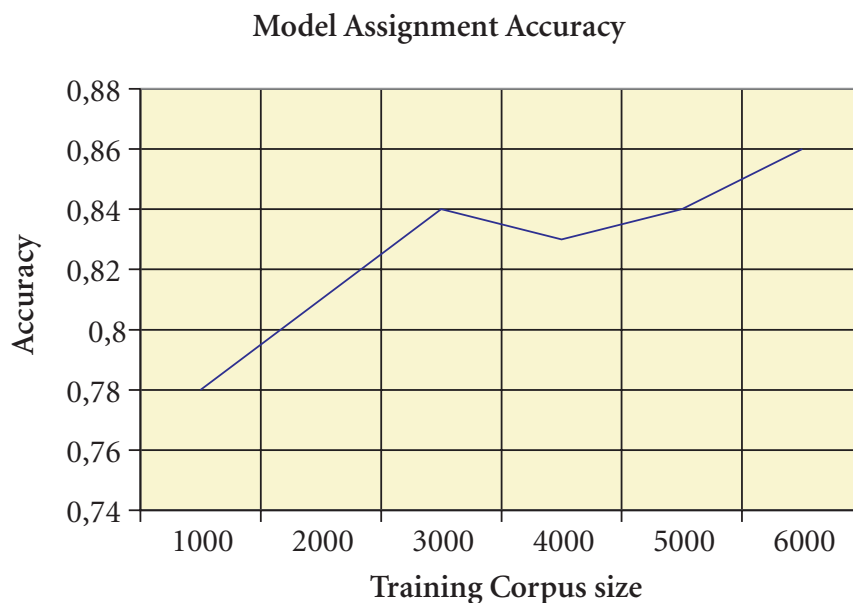
Model Assignment Accuracy

Figure 2. Association with construction classes

    An estimation of the statistical distribution of German multi-word expressions in the dictionary of idiomatic German is shown in Fig. 3.

    Figure 3 shows that the overwhelming majority of multi-word expressions in the dictionary are verb expressions, i.e. they contain a verb as the head. About one half of the verb units are simple combinations of atoms and verbs
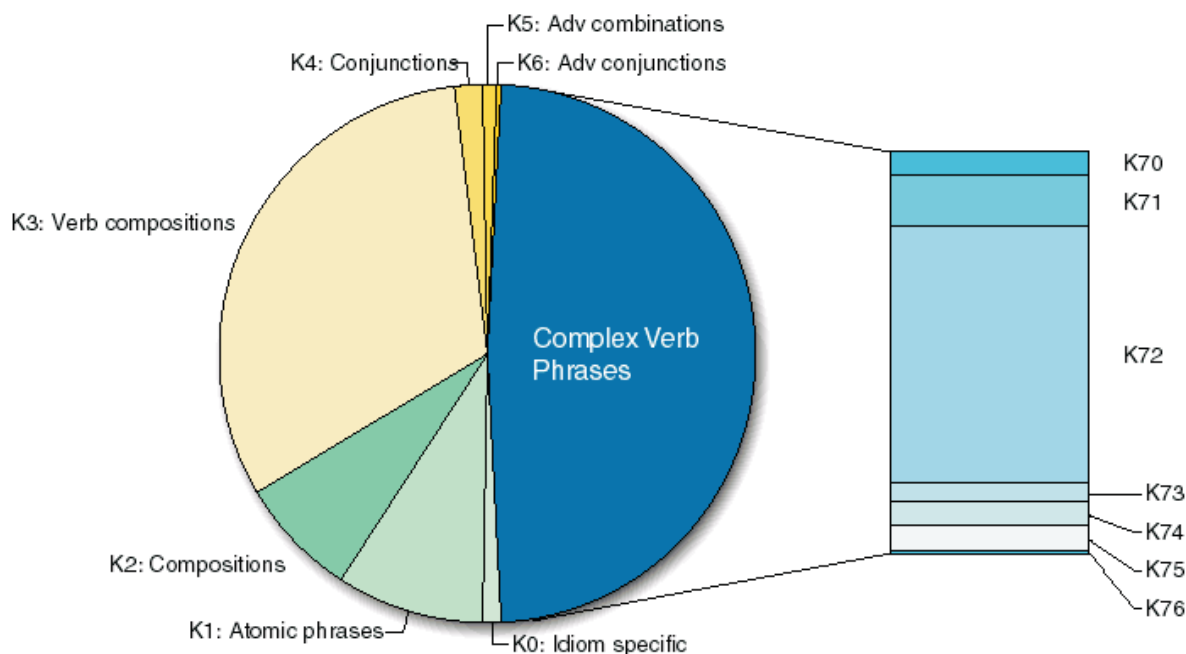
Figure 3. Distribution of construction classes

(class K$_3$), the remainder consists of complex combinations of phrases and verb phrases, i.e. multi-word units that contain either two verbs or a non-atomic phrase. Comparatively small but not negligible is the number of multi-word expressions with obligatory adverbial or conjunctive phrases. The same holds true for idiom-specific patterns.

We could not associate 14% of the entries with a construction pattern. A manual examination of 873 of these entries (i.e. approximately 1/6 of the remaining 5,300 cases) yields the following classification:

1. Failures where the majority of the cases do not correspond to one construction class (268 cases), e.g.

    {in eins}/{Leib und Seele} verschmelzen → prep det V | N Conj N V

2. Incompleteness of the grammar of construction classes (277 cases). We did not list POS sequences corresponding to full sentences (first example) or to a sentence with an NP that is too complex (second example).

    ich glaub', mich knutscht ein Elch!
    j. den Ausdruck von Götz von Berlichingen an den Kopf werfen.

3. remaining Ambiguities of the slash (/) (132 cases)
4. remaining transcription errors (52 cases)
5. clitics (52 cases)

    für die Katz' reden

6.   problems arising from the Brill-Tagger's lexicon (49 cases)

>   *… und damit basta!*
>   *il dolce far niente*[3]

7.   other (17 cases): These are cases that deviate from the normal form of the dictionary, such as entries with discontinuous particle verbs, where the verb does not stand in the final position: *jm. geht ein Kronleuchter auf* (so. begins to see daylight).

5.   Discussion

It has been shown that the multi-word expressions of Schemann's dictionary of idiomatic German can be tagged using minimal resources with 97.5% accuracy on the basis of a training set of 6,000 entries. In order to obtain reasonable results a minimum of 2,000 hand-tagged multi-words entries or approximately 9,000 tokens seems necessary. With this training set, an accuracy rate of 96% can be obtained for the POS-tagger. In the same way, the categorization rate is 81% for the training set of 2,000 entries and rises to 86% with the full training set of 6,000 entries.

Of course the regularities of the normalized patterns of multi-word units in the dictionary contributes to this success. It is remarkable that this approach did not have to rely on any language-specific linguistic resources except for the training corpus. Also, the recall of the association of the computed POS sequences with predefined construction classes is satisfying: 86%[4] is a good recall rate for further processing (see below).

Compared to manual tagging we believe that this method provides a considerable gain in time. Given an average of 25 encoded entries per hour, it took about 80 hours for the manual tagging part. In order to tag all the entries of the dictionary of German idioms (32,000 entries), it would have taken more than 1,280 hours. Our method required training the Brill tagger (60 hours), correction of the wrong results (470 hours), plus the initial time of the manual tagging of the training set (80 hours). This totals 610 hours, which amounts to a gain of time of about 50%.

Before looking at possible applications of these results, we mention two limitations of the approach. Indeed, the reduced tag set and the absence of syntactic analysis result in some loss of quality.

An example of the drawbacks of a small tag set is the missing distinction

between indefinite pronouns (PIDAT) and reflexive pronouns (REFLPRON) tag names according to the STTS tag set (A. Schiller 1993).

Ex 11   *Sich ein paar Tränen abquetschen*
        Itself a   few   tears   squeeze
        To squeeze out a few tears

Here, *paar*: PIDAT and *sich*: REFLPRON should be distinguished. Otherwise the entry in example 11 would be mapped to the sequence 'pron.det.pron.N.V' which does not correspond to any meaningful construction class. On the other hand, a fairly simple extension of the tag set might correct this erroneous sequence: $NP_{new} := NP_{old} \cup Det.PIDAT.N$.

Complex sentence like expressions such as the expressions mentioned below would require syntax analysis and are thus too complex for a POS based approach:

> *ich glaub', mich knutscht ein Elch!*
> *j. den Ausdruck von Götz von Berlichingen an den Kopf werfen.*

Conclusion

The results of this approach can be applied in three ways.

The association of an idioms dictionary entry with its POS sequence is a good basis for a program that recognizes idioms in unrestricted texts. To implement this, one would have to generate a local grammar or finite state transducer from the association results, i.e. the POS/token sequence of each dictionary entry, and match them it with the corresponding text portion (which we is supposed to be previously POS tagged and lemmatized). Examples for such a finite state filter, supposed to extract those sequences of the corpus that correspond to the longest match of a multi-word expression, can be found in Karttunen (1996) or in Hanneforth (2002). A realization for the specific problem of multi-word expressions is given by Senellart (1998).

Another possibility would be to use the association of multi-word entries with their POS sequences as a pre-classification for further manual linguistic encoding. Rather than describing the multi-word entries in alphabetic order, it is thus possible to proceed with the encoding according to their POS sequence. It seems clear that the encoding of multi-word expressions ordered by POS sequences rather than in alphabetic order is less error prone and faster. More-

over, ordering by POS sequence helps to organize encoding projects: the encoding of simple patterns like adj→N or prep→det→N can be done by novices, whereas more complex patterns, such as those containing verbs or predicative complements are left to more experienced linguists.

Finally, this approach could be extended to other print dictionaries containing multi-word expressions. Since our approach requires a minimum basis of about 2,000 entries as a training set, the dictionary would have to contain substantially more entries than this training set for the method to become efficient. Due to the minimal linguistic requirements, the major adaptation of the approach would reside in the description of the entry format of different dictionaries. Hence, we believe that this approach could be useful for print dictionaries in languages other than German. In particular, this approach could be beneficial for examining and pre-classifying print dictionaries in languages that have historically been underdeveloped with respect to NLP or for dictionaries that have not been exploited yet for the production of machine-readable dictionaries.

Notes

1. "Der Begriff der 'Idiomatik' wurde bewußt weit gefaßt: als 'idiomatisch' gelten alle Einheiten, die kontextgebunden sind." (Schemann 1993: XII)

2. In this example the tagger disambiguates the entry. The other possible sequence V.V with 'einen' as the infinitive 'to unite' would be meaningless.

3. POS sequence of four Italian words

4. This estimate was obtained through a random sampling of assigned collocations. These random samples were manually assigned to a collocation pattern; the population proportion was estimated through the binomial distribution as 88% ± 4%.

References

Boguraev, Bran and Ted Briscoe eds. 1989. Computational Lexicography for Natural Language Processing. London: Longman.

Brill, Eric. 1994. *Some Advances in Rule-Based Tagging.* AAAI. http://www.cs.jhu.edu/~brill/TAGGING_ADVANCES.ps

Boyd-Graber, Jordan. 2002. Collocation Tagging. *Caltech Summer Undergraduate Fellowship — Final Report.* http://jbg.caltech.edu/surf02/final.pdf.

Hanneforth, Thomas. 2002. *Eigennamenerkennung mit lokalen {G}rammatiken.* Vortrag Universität Heidelberg.

Karttunen, Lauri, J-P. Chanod, G.Grefenstette, A. Schiller. 1996. *Regular Expressions for Language Engineering.* Natural Language Engineering 2 (4) 305–328.

Klavans, Judith. 1996. *Computing Corpus and Machine-Readable Dictionary Data for Building Bilingual Lexicons.* Machine Translation 10: 3–4, 1–34.

Lezius, Wolfgang; Rapp, Reinhard and Manfred Wettler. 1996. *A Morphology-System and Part-of-Speech Tagger for German.* In D. Gibbon, ed., *Natural Language Processing and Speech Technology.* Results of the 3rd KONVENS Conference, pp. 369–378. Mouton de Gruyter.

Schemann, Hans. 1993. *Deutsche Idiomatik. Die deutschen Redewendungen im Kontext.* Stuttgart: Pons.

Schiller, Anne, Teufel, Simone, Stöckert, Christiane and Christine Thielen. *Vorläufige Guidlines für das Taggen deutscher Textcorpora mit STTS.* Technical report, IMS, Univ. Stuttgart and SfS, Univ. Tübingen, 1995.

Senellart, Jean. 1998. *Reconnaissance automatique des entrées du lexique-grammaire des phrases figées.* In Lamiroy, B. (éd.) *Le lexique-grammaire.* Travaux de Linguistique, 37, 109–127.

Tschichold, Cornelia. 2000. *Multi-word units in natural language processing.* Zürich: Olms.

Summary

This work demonstrates the assignment of multi-word expressions in print dictionaries to POS classes with minimal linguistic resources. In this application, 32,000 entries from the *Wörterbuch der deutschen Idiomatik* (H. Schemann 1993) were classified using an inductive description of POS sequences in conjunction with a Brill Tagger trained on manually tagged idiomatic entries. This process assigned categories to 86% of entries with 88% accuracy. This classification supplies a meaningful preprocessing step for further applications: the resulting POS-sequences for all idiomatic entries might be used for the automatic recognition of multi-word lexemes in unrestricted text.

Authors' addresses

Alexander Geyken
Berlin-Brandenburgische Akademie der Wissenschaften
Project DWDS
Jaegerstr. 22/23
D-10117 Berlin
geyken@bbaw.de

Jordan Boyd-Graber
MSC 388
Caltech
Pasadena, CA 91126
jbg@caltech.edu