University of Edinburgh

Topics in Natural Language Processing

Essay

Natural Language Processing Of Morphology With Linguistically Motivated Applications To German Linking Elements

Author:
Daniel Vollmer
s1467926

March 20, 2015

Abstract

A survey of the history of the learning of morphological rules is presented. Further investigation is made into the current state of NLP techniques with regards to supervised and unsupervised learning morphology. An analysis of the outstanding problem of "German linking elements" is presented and reviewed. Finally, a proposal is made with the goal of applying current morphological analysis and NLP strategies to the problem of linking elements.

1 Introduction

The study of morphology has traditionally concerned itself with the examination of the smallest individual unit of linguistic meaning, the morpheme. Within the realm of natural language processing, analysis of morphology is largely concerned with the correct parsing of words into stems and affixes i.e. stems and morphemes. In morphologically simple languages, such as English, the majority of analysis has favored a syntagmatic or concatenative approach; morphemes are added one after another onto a stem in order to arrive at a whole word. Furthermore, in a strict syntagmatic approach, the view is usually taken that a morpheme refers to a bi-unique form-meaning mapping i.e. one form of a morpheme maps to one and only one meaning. Unfortunately, for more complex morphological problems, this approach has proved lacking in explanatory power. One such instance is the case of *Fugenelemente* or *Linking Elements* in German.

In the context of compound noun formation, linking elements are the linguistic "glue" that sometimes appears between two compound stems. For example, *Zeitungsindustrie* or *newspaper industry* is composed of three parts, *Zeitung* "newspaper," *industrie* "industry," and *-s-*. This "-s-" is an example of a linking element. In German, linguists commonly list nine linking elements:*-e, -er, -s, -es, -n, -en, -ns, -ens,* and *-∅*. The last one *−∅* is referred to as a zero morpheme. Other languages possess linking elements; for example, English possesses unproductive linking elements as seen in the words *huntsman, marksman,* and *spokesman*; in each instance, *-s-* is the linking element. However, the particular focus of this paper is on German, and its highly productive linking elements; notably, the distribution of German linking elements remains an open problem in linguistics. Simply put, there has been no convincing morphological analysis put forward to explain their presence or purpose.

The motivation of this essay is to provide a groundwork for future analysis of German linking elements within the context of natural language processing and current morphological theory. To this end, a literature review of several key works in the field of morphological NLP is presented. Particular attention is given to papers concerned with German or compound nouns in general. Prior drafts of this work reviewed Brown (2002)[**?**] and Kudo (2004).[**?**] However, the research they presented, while valuable to the field, had little bearing to the task at hand, and are thus excluded. For example, Kudo's application of conditional random fields in order to disambiguate word boundaries is not applicable on the whole to German, as whitespace serves as the word boundary marker when processing German text. Additionally, Hammarström (2011)[**?**] is referenced, but only in relation to his review of other works; his article is itself a literature review of over 200 prominent papers on NLP and morphology, and is thus useful primarily for its attempt to summarize the field's history, rather than any new contributions.

The final half of the paper reviews Neef (2015),[**?**] an article currently undergoing peer review. It provides a novel linguistic analysis of linking elements in light of previous failed efforts. The plausibility of this hypothesis is examined in light of recent developments in morphology e.g. Ackerman (2009).[**?**] Furthermore, proposals for future implementation and analysis of Neef's hypothesis are put forward. One approach considers linguistic analysis in terms of low conditional entropy between German noun forms, based on Ackerman (2013).[**?**] Additionally, before any linguistically motivated analysis can occur, a corpus of suitable candidates must be collected and processed. A potential approach as to how to effect this is proposed, largely based upon the work of Goldsmith and Reutter (1998).[**?**] Finally, the practical benefits of such an investigation are considered. Within NLP, compound nouns have proven particularly troubling in machine translation. Morphologically rich languages with extensive compounding have proven difficult to translate adequately due to data sparsity issues. Effective, linguistically motivated, analysis and splitting of compounds is a potentially effective method to mitigate data sparsity and thus improve BLEU scores. The hope is that deep linguistic analysis can lead to practical improvements in existing systems, or

at least not worsen them; this would be encouraging, since, at times, linguistics and natural language processing have been viewed as in competition with each other:

> *Every time I fire a linguist, my performance goes up.*
>
> —Fred Jelinek

2 Literature Review

2.1 Wothke (1986)

As a condensation of his 1985 PhD thesis "Maschinelle Erlernung und Simulation morphologischer Ableitungsregeln," Wothke's 1986 paper presents a program he developed during his research: PRISM. This program makes use of semi-supervised learning in order to learn rules for inflectional and derivational morphology. Though not mentioned by Wothke, Hammarström (2011) makes note that a primary motivation in developing PRISM was to minimize the requirements on RAM due to limited computing resources at the time; large lexicons were too big to fit into working memory.[1] However, this concern over limited working memory is now largely irrelevant.

To accomplish this task, PRISM is fed annotated training data. For example, if the goal is to learn pluralization rules for English, the data takes the form of singular-plural pairs e.g. *field-fields, anchovy-anchovies, fox-foxes.* PRISM also contains a set of abstract symbolic rules, of the form:

$$X \rightarrow Y / Z(1), Z(2), ..., Z(n)\_\_\#$$

This rule form is to be read as, X goes to Y when it is word final and is preceded by some context Z. The semi-supervised part of PRISM lies in that given the annotated training data, it can adapt the symbolic rules to the contexts that it learns from the training data. Furthermore, it automatically learns the proper ordering for the rules. This method is language-independent; Wothke describes having trained it to learn how to inflect male

[1] Hammarström, 2011 pg.315

nouns to produce female nouns in French, and how to derive nominal actions from verbs in German.[2] After training, the system was run on test data, and correctly produced the target form i.e. female nouns or nominal actions with 100% accuracy. Notably, and as a precursor to many later NLP systems utilizing machine learning, the author notes that performance improves as the size of the training data set increases.[3] Most NLP systems today based on Bayesian statistical methods exploit the same insight.

Given the success of Wothke's approach, it is worth noting its drawbacks. Wothke does not provide a description of the learning algorithm.[4] However, Goldsmith notes that Wothke's algorithms were usually complex, to the point that it was prohibitively time consuming to implement them to serve as a baseline comparison for other systems.[5] The time consuming implementation of the algorithms mean that although Wothke's system is language independent in theory, it may be practically difficult. Doing so requires hand annotated training data for every morphological rule in every language that one wishes to analyze. Furthermore, Wothke notes that PRISM cannot deal with infixes.[6] Though this form of affixation may be less common in European languages, it is relatively common amongst other languages of the world. Thus, though Wothke claims to have a language independent system, it is only within the domain of a relatively small handful of languages.

Regarding the applicability of this work to the problem of German linking elements, their is likely to be no influence. Besides being prohibitively complex to implement (supposedly), German linking elements are in the middle of compound nouns i.e. they are infixed, even if they do not function as infixes. Thus, the symbolic rules that PRISM uses are blind to this morphological variation and would have to be completely reformulated in order to provide an analysis of compound nouns. Rather, Wothke's early work provides an insight into some of the earliest attempts at semi-supervised learning of natural language morphology.[7]

---

2 Wothke, 1985 pg.292

3 Wothke, 1985 pg.290

4 A description is probably provided in his PhD thesis. However, it is unavailable as an e-resource.

5 Goldsmith, 2001 pg.157

6 Wothke, 1985 pg.292

7 Earlier work in the 1960s by Soviet linguist Nikolai Andreev can be seen as perhaps the first efforts at applying computational methods to morphology. However, his influence on the present state of research is minimal to say the least.

2.2 Goldsmith and Reutter (1998)

Though Goldsmith's seminal 2001 paper on unsupervised learning of morphology will be discussed in length further on, a more obscure paper produced while he was a researcher at Microsoft has direct relevance to our task of analyzing German compounds. Goldsmith and Reutter's "Automatic Collection and Analysis of German Compounds" presents efforts to automatically collect and split German compound nouns. Specifically, and of primary interest to the task at hand, their analysis focused on the distribution of the linking element amongst each word in the lexicon.

The author's collection of compounds begins with the observation that in German writing, nouns are always capitalized. Thus, their search space is limited to the approximately 300,000 words in Microsoft's *Encarta* that begin with capital letters. Though not described, this is probably accomplished via a regular expression search. From this collection, 8,426 noun stems were identified. Further processing was then done to filter the candidates. The next step was to identify productive German suffixes, 74 being identified.[8]. If their algorithm matched two words composed of an identical stem with two different suffixes, it was accepted as a legitimate stem and stored. This included the possibility of no suffix i.e. a stem that appears once with an suffix and once with no suffix is stored.

The authors, now having collected a list of stems, return to the original corpus to search for possible parses of compounds. This procedure is relatively straightforward as a search problem. With a list of stems, 74 suffixes, and 9 linking elements determined beforehand, the procedure is a matter of pattern matching i.e. find all items that match the pattern `stem+linking_element+stem+suffix`. The search returned "5522 compounds, based on 3866 distinct First Element stems.[9]" These results are stored in the form:

$$(Left\ Stem, Linking\ Element\{Examplar_1, Exemplar_2, ..., Exemplar_n\})$$

In this notation, the authors use "Exemplar" to refer to the right element of a compound, which has the form `stem+suffix`. No comment is made as to the data structures for storing the items; however, the bracketing above, which mirrors the authors' provides some guidance. Presumably, each item

[8] Goldsmith (1998) These suffixes include the common *-en, -e, -er,* and *-ung.*

[9] Goldsmith (1998)

is composed of a tuple, each tuple containing a left stem followed by a dictionary; this dictionary would have a single linking element as a key, and a list of exemplars as values e.g. (a,{b:[x,y,z]}). These individual entries could then be stored in a larger data structure such as a list or dictionary, depending upon other experimental considerations.

Another round of filtering is conducted on the records in order to reduce potentially ambiguous nouns. At this point, the data is restructured, such that in each entry, the left stem is paired with the linking element it appeared with to form a unit the authors call a "candidate." Thus, each compound noun is of the form *Candidate+Exemplar*, and the corresponding data is stored in the form $(candidate, [exemplar_1, exemplar_2, ..., exemplar_n])$ i.e. a two element tuple with the second being a list. The structure is not described as such, but the formatting is fairly transparent.

The first step in this round of filtering removes nouns of the form *Abbildung*, since the first element, *Ab* is not a noun. Problematically, how such stems entered into the potential list of candidates is not addressed at any step of the filtering. Presumably, this resulted from errors made during the first pass through the corpus that identified the 8,426 noun stems. Some of these candidate stems were not nouns themselves. Again, without further insight from the authors, we are left to conjecture. Second, left stems with multiple parts of speech are excluded, *gut*, which can mean "good" or "property" is used as an example. Third, candidates that have ambiguous divisions between the stem and linking element are removed; for instance *Name* can be parsed as *Name+∅* or *Nam+e*. Fourth, candidates wherein the final character of the left stem and the first character of the linking element are identical are excluded e.g. *industrie+er*. This step is essential for avoiding spurious parses such as *industrie+er+zeugnis+se*, which is properly parsed as *industrie+∅+erzeugnis+se*. Fifth, unlexicalized exemplars i.e. those not seen individually, are excluded. This is a potentially rare case, and the authors use the exemplar *bella+∅* as an example, which only occurs in conjunction with the candidate *Ara* "parrot+∅." Sixth, exemplars with ambiguous stem and suffix divisions are excluded. The most salient example proffered being *kammer*, which can be analyzed as *kamm+er* "comb+er" or *kammer+∅* "chamber." Seventh, exemplars which generate ambiguity in the division between the linking element and the exemplar stem are excluded, with *Abfallerfassung* having the possible parses of *Abfall+er+fassung+∅* or *Abfall+∅+erfassung+∅*. Finally, cases wherein the candidate plus exemplar i.e. the whole compound, are lexicalized, are removed. For instance

the candidate plus exemplar pair *Ara+∅+rat+∅* is lexicalized as *Ararat* (as in the Turkish mountain). This compound pair is removed. The terminology used in describing this final step is challenging. Though the example provided is obviously a correct example of a spruorious compound, the plain words of the report suggest that should one have the compound pair *Zeitung+s+industrie+∅* "newspaper industry," and actually encounter it in the corpus i.e. find *Zeitungsindustrie*, then the hypothesized candidate plus exemplar pair is excluded. This interpretation is obviously wrong, however, the report does not elaborate on the step in order to remove this ambiguity.[10]

Only 1341 nouns survive the stringent filtering process. However, with this new list, it is possible to calculate distributions for the linking elements. This process breaks down to simply calculating frequencies. Let T be the total number of exemplars associated with a given left stem, and L be the total number of exemplars associated with the corresponding candidate (left stem+linking element). The frequency of a given linking element for a left stem noun LF is:

$$LF = \frac{L}{T}$$

No table of results is presented by Goldsmith and Reutter, however, as an example of the above formula, they give the linking element distribution for the noun *Staat*. The distribution, with two associated linking elements, is $-en = 0.11$ and $-s = 0.89$.

The primary contribution of Goldsmith and Reutter's work is the scalability of their system. Though they only made use of a relatively small corpus of 300,000 words from *Encarta*, the automation of their method at all steps results in no theoretical limit on the source corpus size.[11] This automation provides great utility for potential future work. However, the author's are careless when using the phrase "automatic analysis." It is certainly true that their system accurately parses German compounds into

[10] To view this problem for oneself, see **Step 8** in Goldsmith (1998).

[11] Goldsmith and Reutter properly conclude that the limit is rather a practical one. Computational resources are the primary limiting factor in this method of collection and analysis. For the size of corpora in question e.g. EUROPARL, this issue is even more trivial now than it was in 1998.

`stem-linking_element-stem-suffix` form. Hammarström (2011) categorizes this segmentation as one of the lowest levels of analysis.[12] Traditionally, linguistic analysis is viewed as connecting empirical observations to an overarching justification[13] i.e. an empirically motivated theory is created, informed by the results of the segmentation and further analyses. In this regard, Goldsmith and Reutter do very little; their linguistic motivation is even, arguably, in error. Furthermore, clues as to appropriate data structures for storing hypothesized compound nouns benefit future work. The nested format chosen is indeed logical, but also bears traces of influence from Goldsmith's later work on Minimum Description Length, wherein morphemes are handled in a similar series of structures.

As an example, the authors dismiss the existence of "stem-dropping" such as when *Schule* is first reduced to *Schul-* before forming *Schulkind*. Rather, they argue that *Schul* is the stem, since it illustrates the relation between *Schule* and *schulen*.[14] This analysis is unconvincing. As a counterexample, it is improbable that a native speaker of English, when looking at the words *happy, happily, happier, happiness,* and *happiest*, would deduce the stem form to be *happ-*. Such an analysis is analogous to Goldsmith and Reutter's one of German compounds. Averred, such an approach does make sense from a text processing standpoint in that it requires less of it, which is certainly a valid motivation; however, this is not synonymous with linguistic motivation, and possibly in conflict with it.

2.3 Goldsmith (2001)

Goldsmith's seminal work "Unsupervised Learning of the Morphology of a Natural Language" immeasurably impacted this particular field of NLP, having over 670 citations on Google Scholar. While earlier work e.g. Wothke (1986) provided initial guidance in applying machine learning techniques to a subset of morphological problems, Goldsmith's work is arguably the first attempt at a comprehensive and general analysis of natural language morphology.[15]

Goldsmith describes the paper's approach to morphology as "top-down," and more specifically as a, "Globally optimal analysis of the corpus." The

[12] See Hammarström, 2011 pg. 312

[13] Hammarström, 2011 pg. 312

[14] See Goldsmith and Reutter's footnotes. They discuss this analysis at length.

[15] Even if it is not the absolute first, it is certainly the most ambitious in its scope.

key insight for this approach is that the number of letters in a given list of words is greater than the number of letters in a list of stems and affixes generated from the parsing of the original list of words. This observation provides the motivation for Goldsmith's use of Minimum Description Length (MDL). More succinctly, MDL asks, what is the most concise way to store information about the given language's morphology? For example, consider a sample of three stems and two suffixes, which can be combined to form any $stem + suffix$ pattern. Listed exhaustively as individual instances, we would have six entries:

$$\begin{Bmatrix} \text{stem}_1 + suffix_1 \\ stem_1 + suffix_2 \\ stem_2 + suffix_1 \\ stem_2 + suffix_2 \\ stem_3 + suffix_1 \\ stem_3 + suffix2 \end{Bmatrix}$$

However, MDL exploits the observation that by splitting up the stems and suffixes, or any affix more generally, we can more compactly store the information:

$$\begin{Bmatrix} \text{stem}_1 \\ stem_2 \\ stem3 \end{Bmatrix} \begin{Bmatrix} suffix_1 \\ suffix_2 \end{Bmatrix}$$

In this admittedly small example, storing stems and suffixes separately reduced the number of entries stored from 6 to 5. As the size of the corpus analyzed increases, the advantage gained from storing morphological information using this approach increases.

The algorithm used in Goldsmith's system is not described, though he does thoroughly ground the approach in a mathematical description.[16] Unlike Wothke's semi-supervised approach, Goldsmith's method is truly unsupervised, general, and language independent. It requires no annotated training data and takes raw natural language text as input. When the system

[16] Goldsmith (2006) does actual detail the algorithm used in his unsupervised MDL approach.

is tested on English and French data, the correct morphemes are identified with 85% accuracy. With no annotated training data required, Goldsmith's system should be easily applicable to any language for which text corpora are available.[17]

Unfortunately, Goldsmith's article concludes with noting that unsupervised analysis of compounds remains an unsolved problem. As it stands, this particular approach will not be useful in analyzing German linking elements. Furthermore, when it comes to a deeper linguistic analysis of the MDL algorithm's results, several spurious decisions are made. For example, the related words *abet, abetted, abetting* are correctly analyzed as having the uniform stem *abet*; however, the corresponding suffixes are determined to be *∅, -ted, -ting*. A syntagmatic analysis would usually conclude that an additional *-t-* is inserted when suffixing occurs. Furthermore, this is arguably an orthographical issue, rather than a linguistic one, since the doubling of consonants in writing does not correspond to an increase in duration of the spoken utterances. These criticisms presume that morphology is properly analyzed as a concatenative process, a contentious claim which is examined later in this essay.

2.4 Koehn and Knight (2003)

Koehn and Knight's efforts at compound splitting bear direct relevance to issue of linking elements as the source language for their work is German. In particular, the authors even mention the presence and unpredictability of linking elements, though no deeper linguistic analysis is attempted. Rather, the process of compound splitting is treated as a means to an end. The hope of the authors is that by splitting compounds in training sets, the BLEU score of a machine translation system can be increased[18]; the primary reason for this hypothesized performance increase is that compound splitting can ameliorate the problem of data sparsity i.e. splitting compound nouns provides more examples of individual nouns during the alignment phase.

Multiple options are considered as to how to collect and split the compounds from a given corpus. One such possibility is using dynamic programming.[19] However, as the authors note, computational complexity is not an

[17] As with most unsupervised learning techniques, Goldsmith's system's performance increases as the amount of training data increases.

[18] Koehn, 2003 pg.1

[19] Koehn, 2003 pg.2

issue in this case, so they elect to perform an exhaustive recursive search. Since the goal of the paper is to improve the training of a machine translation system, the authors must decide upon a method for deciding when to split a compound into its components and when to leave it whole. They elect to use the geometric mean of the word frequencies of a compounds parts.[20] If compound occurs more frequently as a whole when compared to the geometric mean of its parts, then the word is not split for the purposes of improving alignment. For example, the word *Aktionsplan* as a compound, occurs 852 times. Meanwhile, the potential split *Aktions(5)-plan(710)* has a geometric mean of 59.6.[21] In this instance, the compound will not be split for the purposes of alignment.

The authors note that one of the initial flaws in the baseline approach is that prefixes and suffixes are likely to be split off in error. To illustrate, the problem is analogous to a compound splitter taking the word *there* and splitting off *the* due to its high frequency as an individual word in English. To avoid this spurious analysis, the authors tag the German corpus with POS tags.[22] Potential splits can then be excluded based upon parts-of-speech. For example, splits are limited to compound elements that have the tag of NN, NE, ADJA, ADV, etc. This prevents the error of splitting frequently occurring determiners for instance. When incorporating a few other heuristics (such as the use of a parallel corpus), this method of compound splitting achieved 99.1% accuracy against a gold standard reference set. Furthermore, once the split compounds are incorporated into the alignment phase of a word based and phrase based machine translation system, the resulting BLEU scores are increased by up to 0.039.[23]

Unfortunately, Koehn and Knight provide no linguistic analysis of linking elements, merely mentioning their existence. In fact, they treat them as largely an annoyance, stating that, "There are no simple rules for when such letters may be inserted.[24]" Their solution is to simply allow for linking elements to be inserted between any two words. While pragmatic, this decision provides no guidance in actually analyzing linking elements. Though

[20] Koehn, 2003 pg.3

[21] The parentheses indicate the frequency of the individual components. Other splitting options are given as well, however, this one conforms to the analysis proposed by Neef (2015), which is the driving motivation for this paper.

[22] Koehn, 2003 pg.4. The authors use the TnT tagger developed in Brants (2000).

[23] Koehn, 2003 pg.6. Table 2 and Table 3 confirm these results.

[24] Koehn, 2003 pg.2

Goldsmith and Reutter's approach to collecting compounds from a corpus is likely to suffice, the use of a POS tagger to improve accuracy is another possibility, which clearly benefited Koehn and Knight's work. However, the author's driving motivation i.e. improving BLEU scores provides a useful experimental metric following a collection and analysis of compounds containing linking elements. After correctly splitting compounds based upon Neef's analysis of linking elements as belonging to the left element, we can examine if such an analysis is beneficial to the training of an machine translation system. The likely answer would appear to be no. The authors note that an "eager" splitting heuristic i.e. one which split as much as possible, performed best in terms of BLEU score when using a phrase based system. However, this should not deter an attempt at using Neef's analysis; statistical machine translation is a highly pragmatic field and the approach which improves BLEU scores the most is not necessarily the linguistically motivated one.

2.5 Botha, et al. (2012)

Botha et al. provide a novel approach to the task of modelling and splitting German compounds. Whereas previous approaches e.g. Goldsmith and Reutter, employed an exhaustive search that matched a predetermined pattern to words in a corpus, Botha uses n-grams. Essentially, German compounds are viewed as individual n-grams that do not have white-space separating them in the text. Furthermore, their approach contains a great deal of linguistic motivation, presciently conforming to some of Neef's (2015) analysis of compounds. In particular, the authors correctly conclude that German compounds are dependent upon the right element i.e. in *Eisenbahn*, *bahn* is the primary component. Meanwhile, the right element is dependent upon the preceding context e.g. *mit der* in the phrase *mit der Eisenbahn*. Thus, the authors propose a reverse n-gram model:

$$p(eisenbahn|mit\ der) \equiv p(bahn|mit\ der) \times p(eisen|bahn) \times p(\#|eisen)$$

In the model proposed above "#" indicates the word boundary.[25] More importantly, Botha mentions the problem of analyzing linking elements in

[25] Botha et al. uses $, however # is more common in linguistic writing.

this model. Though the linguistic motivation is not explored, they arrive at, according to Neef (2015), the correct analysis. Analyzing linking elements as individual elements is likely to disturb the conditional probabilities e.g. for *Küchentisch*, they wish to avoid $P(küche|n)$. To counter this problem, linking elements are merged onto the left element.[26] This produces a conditional of the form $P(küchen|tisch)$.

The primary motivation of Botha et al. is to improve existing machine translation systems, so the primary thrust of their experiments and discussion centers around BLEU scores and other metrics following the training of an MT system. They report little impact on BLEU.[27] However, when the resulting output is examined for precision in correctly analyzing compounds, the improved model results in a 12% increase over the baseline i.e. greater accuracy in producing the correct English translation of a German compound noun. Furthermore, this is done at minimal cost to recall. Since the authors do not discuss the actual implementation of the n-gram model at length, their methodology is not of immediate use to analyzing linking elements. However, the encouraging element of their results is that the correct linguistic analysis i.e. attaching linking elements to the left noun in a compound was not detrimental to the performance of the MT system. This provides hope that a linguistically compelling analysis of German compounding rules will not conflict with the pragmatic goals of machine translation and other NLP tasks.

3 German Linking Elements

3.1 Neef (2015)

In a paper currently undergoing peer review, Martin Neef, a professor of linguistics at Technische Universität Braunschweig, examines the problems with previous attempts at analyzing German linking elements. To reiterate, these include nine potential candidates: *-e, -er, -s, -es, -n, -en, -ns, -ens,* and *-∅*.[28] Prior attempts at linguistically motivating the occurrence of linking elements are explored comprehensively, this includes previous pho-

[26] Botha et al., 2012 pg.246

[27] The authors note that the negligible increase is probably due to compounds making up only a relatively small amount of the overall translations.

[28] Neef does not include the zero morpheme in his initial analysis, but many others do.

netic, phonological, and semantic attempts. These analyses are classed as *functional* analyses i.e. they implicitly assume that linking elements have a specific function within the German language. However, in light of previous failed efforts at functional analysis, Neef proposes a *non-functional* analysis.

By non-functional, Neef does not imply that linking elements are useless; they are an essential component of well formed German words. Rather, non-functional means that they do not have a specific linguistic purpose; they simply exist. To this end, Neef posits that linking elements are properly place as constituents of the left hand member of a compound noun e.g. in *Tagebuch*, the linking element *-e* is properly understood as part of the preceding noun i.e. *Tage*. This movement away from syntagmatic morphology has made great strides recently in providing analysis of seemingly complex morphological problems.[29] Neef points out that despite the wide variety of compound stem forms in German, most lexemes are only associated with one specific linking element i.e. it is predictable and non-random. Furthermore, if a lexeme has multiple compound stem forms i.e. can take on different linking elements, one usually predominates.[30] When multiple compound stem forms occur, one form usually has a frequency of over 90%. Furthermore, Neef posits that only one compound stem form is productive i.e. only one form can be used derivationally to form novel compounds.[31] Though Neef notes that it is impossible to prove conclusively which linking element of a given lexeme is the regular productive one, the frequency can be measured empirically. For instance, noted that in Goldsmith (1998), they measured the frequency of the linking elements for *Staat* *-en* and *-s* at 0.11 and 0.89 respectively. This conforms to Neef's prediction about frequency of multiple compound stem forms. Furthermore, the more frequent element *-s* is the productive one in this case.

Moving forward, Neef's analysis, while plausible, could benefit from empirically confirmed justifications.[32] There are two elements to this analysis. The first is additional linguistic motivation. The non-compositionality of linking elements is discussed below in light of Ackerman's (2009) work.

---

[29] See Ackerman (2009) for a complete discussion on non-compositional, paradigmatic morphology.

[30] Neef, 2015 pg.25

[31] Neef, 2015 pg.25

[32] I do not mean to insinuate that I find Neef unconvincing. To the contrary, I believe he is certainly correct. Rather, I mean that his theory could further benefit from experimental results.

Specifically, the hope is that an analysis of the distribution of linking elements within the German nominal paradigm will conform to Ackerman's hypothesis of the low conditional entropy of seemingly complex morphological systems. Before any linguistic analysis can be carried out, data must be collected. This forms the computational component of the proposed experiments. To this end, a brief discussion is given as to which concepts presented in the literature review are best suited to the task at hand. Additionally, further metrics for analysis are considered.

3.2 Proposal

3.2.1 Linguistic Motivations

Though external analysis of German linking elements results in an observation of byzantine complexity, our guiding principle of analysis must be that any natural language must be simple to learn and use for its speakers. If one vein of analysis results in baffling complexity, we must abandon it in favor of a simpler explanation. Neef's non-functional analysis of linking elements does precisely this. After reviewing prior attempts phonetic, phonological, semantic, and structural analysis, he concludes that none provide a satisfactory account.[33]

The key to a linguistically motivated analysis lies in the problem of the phrase *external* analysis. Our motivating hypothesis that a language must be simple to learn and use says nothing about observed *external complexity* in a language; the nine forms of German linking elements and their distributions may appear opaque to outside observers (and probably are), however, this pales in comparison to Finnish nouns, for example, which have 15 cases with numerous noun classes i.e. each class is case marked differently from another class, yielding thousands of possible combinations. However, neither the moderate external complexity of German or the high external complexity of Finnish prevents children from learning their native language. This ease of acquisition leads us to conclude that a language must have *internal simplicity* i.e. there must be linguistic clues that make an outwardly difficult language easy to learn for its speakers.

With the hypothesis of internal simplicity in hand, we are left with the possible conclusion that the syntagmatic approach to morphology is flawed;

[33] Neef exhaustively examines the current literature on linking elements, and provides numerous counter-examples for every prior attempt at functional analysis.

syntagmatic refers to the view of words as being compositional in meaning i.e. a word consists of a stem with morphemes concatenatively added in sequence, yielding a complete meaningful unit. This approach has dominated morphology in the post-Bloomfield era.[34] Though such approaches perform well on morphologically simple languages, such as English, the syntagmatic hypothesis has shown remarkably weak explanatory power when applied to more morphologically complex languages e.g. Finnish and other Turko-Ugric languages. Recent advances however have been made by taking a non-compositional approach to morphology.[35] This suspicion of the syntagmatic approach is not recent, with suspicions as to its explanatory power going back 50 years:

> I know of no compensating advantage for the modern descriptive reanalysis of traditional paradigmatic formulations in terms of morpheme sequences. This [morphemic analysis] seems, therefore, to be an ill-advised theoretical innovation ... It seems that in inflectional systems, the paradigmatic analysis has many advantages and is to be preferred ... It is difficult to say anything more definite, since there have been so few attempts to give precise and principled description of inflectional systems in a way that would have some bearing on the theoretical issues involved here.[36]

This non-compositional approach has been termed, "the Paradigm Cell Filling Problem" or (PCFP).[37] See **??** for an example of a paradigm. Each noun class is restricted to a possible set of case markers. Determining a noun declension becomes a question of which cell it fits into. This approach reduces the learning morphology to the question of what licenses a speaker to make inferences about the form i.e. morphological form, of a given novel word. This problem of inference is further reducible to a question of measuring conditional entropy.

---

[34] Hammarström pg. 313
[35] see Ackerman, 2009
[36] Chomsky, 1965 pg.174
[37] Ackerman, 2009 pg. 54

	SINGULAR				PLURAL			
CLASS	NOM	GEN	ACC	VOC	NOM	GEN	ACC	VOC
1	-os	-u	-on	-e	-i	-on	-us	-i
2	-s	-Ø	-Ø	-Ø	-es	-on	-es	-es
3	-Ø	-s	-Ø	-Ø	-es	-on	-es	-es
4	-Ø	-s	-Ø	-Ø	-is	-on	-is	-is
5	-o	-u	-o	-o	-a	-on	-a	-a
6	-Ø	-u	-Ø	-Ø	-a	-on	-a	-a
7	-os	-us	-os	-os	-i	-on	-i	-i
8	-Ø	-os	-Ø	-Ø	-a	-on	-a	-a

Figure 1: Greek noun inflections. Pay particular attention to the Genitive Singular and Accusative Plural, as they will serve as examples for our calculations.

The use of information entropy as a metric for change in predictability[38] of a given word form is particularly useful for the notion of learning morphology by inference. It allows us to measure the uncertainty of a given word form being realized and therefore allows us to determine, on average, the number of guesses a speaker would require to choose the correct morphological form of a word. Drawing from the work of Ackerman (2009) we begin with the definition of entropy:

$$H(X) = -\sum_{x \in X} P(X) log_2 P(X)$$

Applying this definition to calculating the predictability of Greek genitive singular forms is quite straightforward. Observe in **??**[39] that the genitive singular has eight classes with five possible markers: (-u, -∅, -s, -u, -us, -os). If we assume that all declensions are equally likely, with a probability of $\frac{1}{8}$, then the entropy of the Greek genitive singular is:

$$H(GEN.SG) = -\left(\tfrac{3}{8} log_2 \tfrac{3}{8} + \tfrac{1}{8} log_2 \tfrac{1}{8} + \tfrac{2}{8} log_2 \tfrac{2}{8} + \tfrac{1}{8} log_2 \tfrac{1}{8} + \tfrac{1}{8} log_2 \tfrac{1}{8}\right)$$

$$H(GEN.SG) = 2.156\ bits$$

[38] Shannon, 1948

[39] Image borrowed from Ackerman (2013)

With this result, the number of guesses needed to blindly choose the correct declension for a given noun is:

$$2^{2.156} = 4.312 \; guesses$$

This observed value of more than four guesses being necessary to choose the proper Greek genitive form marks an upper bound on entropy for the class. However, Ackerman (2013) notes that inference allows humans to perform much better than "random guessing."[40] Observe the co-variation between the genitive singular and the accusative plural forms. Given knowledge of one, it should be possible to infer the other i.e. if we know the accusative plural declension of a noun, it gives us information about its potential genitive singular form. This ability to infer given prior information is known as *conditional entropy*, notated as $H(Y|X)$. For example, if one encounters a Greek noun in the accusative plural with the ending *-i*, one can infer that its genitive singular ending must be *-us* i.e. $H(GEN.SG|ACC.PL = -a) = 0$. This observation demonstrates that speakers of a language can use inferential knowledge to predict unseen word forms, sometimes with absolute certainty; refer to **??** if you wish to confirm this observation. Sometimes, prior knowledge does not reduce entropy to zero; however, it still results in an overall reduction in entropy. Ackerman (2013) defines the conditional entropy of a word form c_1 given knowledge of a word form c_2 as follows:

$$H(c_1|c_2) = \sum_{r_1} \sum_{r_2} P_{c_1}(r_1) P_{c_2}(r_2) log_2 P_{c_1}(r_1|c_2 = r_2)$$

Taking as a working example the Greek accusative plural ending *-a*, if this ending occurs, it leads to two possible genitive singular endings: in two cases, we observe *-o* and in one case *-∅*. Using our working definition of conditional entropy, this prior knowledge yields:

$$H(GEN.SG|ACC.PL = -a) = -\left(\tfrac{2}{3} log_2 \tfrac{2}{3} + \tfrac{1}{3} log_2 \tfrac{1}{3}\right)$$

$$H(GEN.SG|ACC.PL) = 0.918 \; bits$$

$$2^{0.918} = 1.889 \; guesses$$

As we can see, prior knowledge greatly decreases the amount of uncer-

[40] Ackerman, 2009 pg.441

tainty present in the language. Ackerman (2013) calculated the average conditional entropy for the genitive singular given all possible realizations of the accusative plural, and found a conditional entropy of 0.594 *bits*, equating to $2^{0.594} = 1.188$ *guesses*. Given that Neef views German linking elements in the same framework as Ackerman i.e. as non-compositional, it is worth investigating the applicability of Ackerman's work on conditional entropy. As a starting point, given nine German linking elements, and assuming equal likelihood, we can calculate the given entropy for correctly assigning a given noun its correct linking element:

$$H(Linking\ Element) = -(\tfrac{1}{9}log_2\tfrac{1}{9}+\tfrac{1}{9}log_2\tfrac{1}{9}+\tfrac{1}{9}log_2\tfrac{1}{9}+\tfrac{1}{9}log_2\tfrac{1}{9}+...+\tfrac{1}{9}log_2\tfrac{1}{9})$$

$$H(Linking\ Element) = 3.170\ bits$$

$$2^{3.170} = 9.000\ guesses$$

This calculation serves as a baseline, an upper bound, on the unpredictability of linking elements. If Ackerman's analysis is applicable in this instance, it is highly probable that we will uncover a lower conditional entropy, probably, considering Ackerman's other results, below 1 bit.

A linguistically motivated analytical account of German linking elements should attempt to examine their potential place within the Paradigm Cell Filling Problem i.e. given a noun N in a given case C, can its linking element form be inferred? Ackerman's work gives reason to believe so. Furthermore, it gives us a quantifiable measure in conditional entropy (something which evades many linguistic analyses). This question requires further investigation, and to do so requires data. Though some German compound noun corpora are potentially available for analysis, they are likely to be small in size.[41] Thus the first step to an empirical evaluation of Neef's analysis of linking elements is data collection. An experimental approach is proposed below.

[41] Goldsmith (1998) mentions that after pre-processing, they had a corpus of 1341 lexicalized nouns for analysis. Koehn (2003) provides some hope, as their approach started with the Europarl corpus (20 million words)

3.2.2 Experimental Approach

As discussed, the first challenge in empirically testing Neef's hypothesis is data sparsity. Though he provided several examples of compound nouns with their linking elements in his work, they are likely *ad hoc* i.e. as a native German speaker, he produced them himself. Even if they are drawn from a secondary source, there are still not enough to allow for any meaningful analysis using NLP techniques. Thus, collection of German compound nouns is the first necessary step.

Goldsmith (1998) mentions the scraping of Microsoft's *Encarta* for German words, with an initial 300,000 candidates (a candidate being a capitalized word).[42] Though an admirable effort in 1998, currently, the EUROPARL corpus seems a useful starting point for data collection, containing approximately 44 million German words.

Though Koehn and Knight demonstrated an alternative method to collecting and splitting compound nouns, the prior methods used by Goldsmith and Reutter are likely to suffice. Carrying out an exhaustive recursive search is not a problem, since we are not concerned with computational resources. Furthermore, Goldsmith and Reutter provide an in-depth explanation of their methodology. This alone makes their approach worth adopting, as it will be easiest to reimplement, and provides us with a baseline to ensure we are performing compound collection and splitting properly. We would like our collection and splitting to be precise and accurate as well. Though Koehn and Knight were certainly successful, they incorrectly split compounds 15% of the time; Goldsmith and Reutter achieved better performance in this regard.

Once the collection task is completed, it seems likely that some level of sorting will need to be done to the data. The goal of these experiments would be to detect some underlying pattern in the occurrence of linking elements. As a baseline, recalculating the frequency of occurrence of the linking elements of a given lexeme seems an appropriate starting point. Goldsmith and Reutter only briefly discuss this effort and it would be helpful to have more information. Additionally, after splitting compounds according to Neef's proposed analysis, the split compounds could then be used in the alignment

[42] Goldsmith and Reutter do not seem to have put much emphasis on data collection. This is possibly due to the paper being written quickly while they were researchers at Microsoft, and since the paper is merely background work for Goldsmith's later work on unsupervised learning of morphology.

phase of training an MT system, and then checked to see if the resulting BLEU scores improved, etc. Again, as averred previously, BLEU scores in no way indicate correct linguistic analysis. However, as Botha et al. (2012) found, it would be encouraging to find that a supposedly correct linguistic analysis can lead to improvements in practical applications. Finally, Neef's hypothesis that only one linking element for a given compound stem is productive could be tested by using our data generatively. Specifically, the most frequently occurring linking element of a given lexeme would be used, and that left element could then be combined with other nouns in order to automatically coin novel German compounds. A list of these compounds could then be given to German native speakers and checked for grammaticality.

4 Conclusion

An attempt has been made to give a cursory treatment of important developments in natural language processing within the domain of morphology. This review was purposely limited in scope; unlike Hammarström (2011), who looks a over 200 publications on the topic, it was never intended to be comprehensive. Rather, publications were evaluated in light of their potential application to the specific problem of German linking elements. The notable work of Goldsmith and Reutter (1998) provides the most straightforward approach to the collection of compound noun data. Additionally, these articles were evaluated in light of Neef's recent non-functional and non-syntagmatic analysis of linking elements.

Proposed future experiments aim to evaluate Neef's claims in light of prior linguistic and NLP research; are they supported linguistically, and if so, can this analysis be used to improve existing NLP systems, such as BLEU scores in machine translation. The optimistic answer is yes, with the work of Botha et al. (2012) demonstrating that a linguistically viable analysis of German compounds is not necessarily detrimental to the results of NLP systems. Additionally, if German linking elements prove analyzable based upon Ackerman's theory of low conditional entropy, then, moving forward, attempts should be made at analyzing linking elements in other languages e.g. Dutch and Finnish, using the same approach. The possibility remains that after collecting the data, German linking elements will remain opaque to any attempt at linguistic or empirical analysis. However, these unanswered questions must be left to future research.

References

[1] Ackerman, Farrell, James P. Blevins, and Robert Malouf. "Parts and wholes: Implicative patterns in inflectional paradigms." *Analogy in grammar: form and acquisition* (2009): 54-82.

[2] Ackerman, Farrell, and Robert Malouf. "Morphological organization: The low conditional entropy conjecture." *Language* 89.3 (2013): 429-464.

[3] Botha, Jan A., Chris Dyer, and Phil Blunsom. "Bayesian Language Modelling of German Compounds" *COLING*. (2012).

[4] Plaehn, Oliver, and Thorsten Brants. "Annotate–an efficient interactive annotation tool." *Proceedings of the Sixth Conference on Applied Natural Language Processing ANLP-2000*. (2000).

[5] Brown, Ralf D. "Corpus-driven splitting of compound words." *Proceedings of the Ninth International Conference on Theoretical and Methodological Issues in Machine Translation*. TMI. (2002).

[6] Noam, Chomsky. "Aspects of the Theory of Syntax." Cambridge, Mass (1965).

[7] Goldsmith, John, and Tom Reutter. "Automatic collection and analysis of German compounds." *Workshop on Computational Treatment of Nominals (COLINGACL)*. (1998).

[8] Goldsmith, John. "Unsupervised learning of the morphology of a natural language." *Computational linguistics* 27.2 (2001): 153-198.

[9] Goldsmith, John. "An algorithm for the unsupervised learning of morphology." *Natural Language Engineering* 12.04 (2006): 353-371.

[10] Hammarström, Harald, and Lars Borin. "Unsupervised learning of morphology." *Computational Linguistics* 37.2 (2011): 309-350.

[11] Koehn, Philipp, and Kevin Knight. "Empirical methods for compound splitting." *Proceedings of the tenth conference on European chapter of the Association for Computational Linguistics-Volume 1*. Association for Computational Linguistics, (2003).

[12] Kudo, Taku, Kaoru Yamamoto, and Yuji Matsumoto. "Applying Conditional Random Fields to Japanese Morphological Analysis." *EMNLP*. Vol. 4. (2004).

[13] Neef, Martin. "The Status of So Called Linking Elements in German." (2015).

[14] Shannon, Claude Elwood. "A mathematical theory of communication." *ACM SIGMOBILE Mobile Computing and Communications Review* 5.1 (2001): 3-55.

[15] Wothke, Klaus."Machine learning of morphological rules by generalization and analogy." *Proceedings of the 11th Conference on Computational Linguistics*. Association for Computational Linguistics, (1986).