

Least-effort noun gender induction for Low Saxon

Jan Strunk

MA student, Linguistics Department
Stanford University, California

jstrunk@stanford.edu

ABSTRACT

In this paper, I describe and evaluate a least-effort approach for automatically learning the gender of Low Saxon nouns. I propose two different methods: One is a corpus-based method which relies on counting clue words that occur in the immediate context of the nouns to be classified and uses very simple statistics to assign gender classes. A second method exploits the right-headedness of Low Saxon compound nouns to classify compound nouns for which the first method found insufficient evidence. The methods described here are an initial attempt at bootstrapping a Low Saxon lexicon which can be used as a basis for an electronic dictionary as well as a starting point for the development of natural language processing applications for Low Saxon.

Keywords

Noun Gender, Word Learning, Bootstrapping, Morphosyntactic Classes

1. INTRODUCTION

During the past months, I have been working on an information retrieval project¹ for a language called Low Saxon (a.k.a. Low German, Plattdeutsch, etc.). Low Saxon is a West Germanic language spoken in Germany, the Netherlands, and in emigrant communities throughout the world. It can be considered a “major” minor language in that estimates of the number of speakers are sometimes as high as 10,000,000, cf. [1]. However, it is a lesser-used language in that it lacked any official status or support until very recently and accordingly is not standardized in any way, i.e. there is no single accepted written standard. Instead, people use their own dialects’ grammar and loosely base their own spelling on either the German or the Dutch orthographic system.

In the course of the above-mentioned project, I have collected a fair amount of Low Saxon electronic text for use in training and evaluation. Now that such a corpus is available for the first time, it offers new opportunities for linguistic research and the development of computational linguistic tools and resources for this language.

As a future project, I would like to build an electronic dictionary which is based on this corpus and which could be a very useful tool for the Low Saxon language community (as it would be the first larger interdialectal dictionary). As an initial step in this direction, I chose to implement and test two very simple algorithms for learning the grammatical gender of Low Saxon nouns. This paper describes the development and evaluation of this least-effort approach.

2. RELATED WORK

I have not been able to locate a large number of previous studies on the problem of learning noun gender. Cucerzan and Yarowsky [3] is the only recent study I know of that directly deals with this challenge. This may be due to the fact that a lot of work about word learning tasks is done for the English language which lacks real gender distinctions (cf. [3]) or maybe researchers assume that learning gender is such an easy task that it is not worth publishing articles about. Unsurprisingly, there do not seem to be any papers at all on natural language processing for Low Saxon. However, I will refer to two recent papers: the one by Cucerzan and Yarowsky [3] about induction of grammatical gender and another one by Nakov et al. [4] about the classification of unknown German nouns. Furthermore, I will base my system on Michael Brent’s general model of learning lexical syntax described in [2].

Cucerzan and Yarowsky [3] use quite similar algorithms as the ones I describe in this paper. They employ a list of seed words with known gender to

¹ Web demo: <http://strunk.no-ip.org>

automatically find words occurring in the local context of a these nouns which represent good clues for a gender classification. They thus exploit the fact that syntactic dependents of head nouns often exhibit gender agreement. In addition, they try to extract morphological cues from the seed nouns which help to predict the gender of new nouns.

Nakov et al. [4] try to guess the morphological classes of unknown German nouns by largely relying on their morphological structure. They try to identify compound nouns and also use cues from derivational and inflectional morphology to guess a noun's morphological class which among other things includes its gender information. However, since they use substantial lexical information and try to tackle the much more difficult task of finding the noun's stem and inflectional paradigm, their study is not directly comparable to this one.

Brent [2] is a general study of how one should best deal with tasks of learning lexical syntax. Although the sample tasks that he evaluates concern verbal argument structure and are thus only distantly related to the present problem, I will refer to his paper for general considerations and methodology.

3. THE CORPUS

I developed and tested the algorithms that I describe in this paper on a corpus that I had built manually for an earlier project by harvesting the internet for Low Saxon texts. The Low Saxon web community is quite large and luckily well interlinked, so that it was relatively easy to find a large number of websites wholly or partly in Low Saxon. I downloaded about 2700 documents of which about 1700 contain Low Saxon only text while the rest is only partly Low Saxon. Downloading these documents to my local file system and saving them in utf-16 format resulted in about 74 MB of html files of which about 40 MB are Low Saxon only. I collected a large diversity of texts ranging from Wikipedia² articles to poetry in a large number of different dialects and orthographic systems (some quite idiosyncratic and experimental). Although this document collection is relatively small, I estimate that it contains a sizeable portion of all Low Saxon texts on the internet. The resulting corpus built from it comprises about 93,700 distinct word types³

² <http://nds.wikipedia.org/wiki.cgi>

³ This large number of distinct types is due to the high number of different orthographic and dialectal variants of the same word.

and about 1,200,000 tokens of running text (including punctuation, etc.).

4. THE CURRENT SYSTEM

4.1 Outline of the system

According to Brent [2], a system that tries to learn the lexical syntax or morphosyntax of words should “*rely on local morpho-syntactic cues to structure rather than trying to parse entire sentences.*” Moreover, it should “*treat these cues as probabilistic rather than absolute indicators of syntactic structure.*” He suggests the following two-step model for such learning systems: First, go through the training corpus and extract the necessary clues by counting observations. Second, classify the words you are interested in using a probabilistic model that can make sense of the collected observations.

The first approach which I propose in the present article does exactly this. Given a list of nouns, it searches a corpus for certain clue words that occur within a specified context of the nouns on the list. For each noun, it counts how often clue words that are good evidence for a certain gender class have occurred in its context. The second stage consists of a very simple classification algorithm that solely uses absolute frequencies to classify the nouns into gender classes. This general architecture is also used by Cucerzan and Yarowsky [3].

The second algorithm I describe in this paper is inspired by Nakov et al. [4]. It exploits the fact that Low Saxon like German or Dutch is a language which makes heavy use of noun compounding and that these compound nouns are also usually written as one word (in contrast to English orthographic practice). Nakov et al. obtain a high coverage and good precision of about 94 % with an algorithm that uses morphological decomposition of German compounds to classify them into noun classes.

Last but not least, I will test whether a combination of the two approaches yields better results than each of them does alone.

4.2 First approach: Clue words

The first algorithm searches through a corpus and extracts a certain width of context tokens from around all the occurrences of the nouns it is supposed to

Consider: *säkje, sööken, söken, seuken, zuiken, zoeken*, etc. all meaning „to search“.

classify. It then scans through these contexts in order to find clue words which suggest a certain gender classification for the noun in question. Instead of counting the clue words themselves, it simply notes whether a clue word favors a certain gender and then adds 1 to all gender classes that are supported by this particular clue word. The clue word information from all contexts of a certain noun is then simply summed to give the final counts for each noun type.

But where do the clue words come from? Cucerzan and Yarowsky [3] use different simple bootstrapping mechanism to get a number of seed words and then automatically learn effective context clue words from their contexts. One could use a similar approach to get clue words for the present approach, too, but as I have sufficient knowledge of Low Saxon and the high orthographic variation in the corpus makes it more difficult to automatically learn clue words, I preferred to come up with classes of clue words myself. This is not a lot of work, since one can simply choose closed class clue words.

The following is a quick overview over the relevant morphosyntactic facts of the Low Saxon language. Low Saxon like German has a three gender system: masculine, feminine, and neuter. But like Dutch, it has lost many case and gender distinctions marked on the dependents of the head noun in a noun phrase. Table 1 gives the paradigms for the definite and indefinite articles and the demonstrative.

As can easily be seen, except for the indefinite article, the plural behaves like a feminine singular. Only few forms of these function words show a clear gender marking. Only the neuter can be easily distinguished from masculine and feminine. In fact, for many agreement facts masculine and feminine behave the same way. I will therefore choose the term **common gender** for both of them.

Table 1. Low Saxon determiners

Definite article				
	Subj. SG	Subj. PL	Obj. SG	Obj. PL
Masc	de	de	den / dem / 'n	de
Fem	de	de	de	de
Neut	dat	de	dat / 't	de
Indefinite article				
	Subj. SG	Subj. PL	Obj. SG	Obj. PL
Masc	een	∅	een / nen / 'n	∅
Fem	een / ne	∅	een / ne / ner / e	∅
Neut	een	∅	een / 'n	∅
Demonstrative				
	Subj. SG	Subj. PL	Obj. SG	Obj. PL
Masc	düsse	düsse	düssen	düsse
Fem	düsse	düsse	düsse	düsse
Neut	düt / düssset	düsse	düt	düsse

I chose the following words as clue words for the different genders⁴:

1. *nen, den, düssen* ⇒ MASC
2. *ne, ner, 'e, tor* ⇒ FEM
3. *dat, 't, düt, düssset* ⇒ NEUT
4. *de, düsse* ⇒ MASC / FEM
5. *dem, 'm, im, tom, vom*⁵ ⇒ MASC / NEUT

Every time the observation gathering algorithm encounters one of these forms or an orthographic variant thereof, it adds 1 to the count of positive clues for all genders that are supported by the specific clue word. For example, if the algorithm encounters *de*, it

⁴ Table 1 and the clue-word sets I list are highly idealized as they do not contain any orthographic or dialectal variants. Of course, I included different variants of the same form in the real clue-word sets.

⁵ *tor, im, tom*, and *vom* are combinations of different prepositions with an enclitic definite article.

will add 1 to the counts for masculine and feminine. After all observations have been counted, the classification stage can compare the counts for the three different genders. It simply chooses the gender for a certain noun which has the highest absolute number of occurrences in the contexts of that noun.⁶ If, however, there is a tie between the highest counts, the current system can make two different decisions: First, if all three genders have the same count (which can be zero) or if either masculine or feminine and neuter are tied for the highest count, no decision at all is made and the noun is assigned to the class *UNKNOWN*. If however masculine and feminine are tied for the best count, we can already assume that the noun in question is not a neuter. This is already valuable information. In that case, the noun is assigned to the class *COMMON*. Otherwise, if there are no ties, the noun is assigned to one of *MASC*, *FEM*, or *NEUT*. For example, if the noun *kind* (“child”) had a count of MASC: 2, FEM: 4, and NEUT: 10, it would be assigned to the neuter class, which in fact would be correct. However, if the noun *goorn* (“garden”) showed the following observations: MASC: 5, FEM: 5, NEUT: 1, it would neither be assigned to *MASC* nor to *FEM*, but to the safer option *COMMON*. Cucerzan and Yarowsky [3] in fact use a similar approach stating that “*In this manner, one captures not only the evidence but also the lack of evidence.*” If all three genders are tied, if for example the algorithm encountered no good or only contradictory clues for the word *speel* (“play”), MASC: 0, FEM: 0, NEUT: 0, the noun would be assigned to the class *UNKNOWN*.

4.3 The second approach: Compound nouns

The second approach uses the simple fact that almost all compound nouns in Low Saxon have the same gender as their last component. For example, the element *huus* (“house”) which as a simplex is neuter also makes the compound noun *sükenhuus* (“hospital”) neuter.

The second algorithm can therefore use a list of nouns for which the gender is known and check for each noun with an unknown gender whether it contains one of the nouns with known gender as last element using

⁶ This is largely similar to Cucerzan and Yarowsky’s algorithm, although the present approach is still simpler because it does not calculate a level of confidence.

regular expressions. It would thus find *-huus* in *sükenhuus* and classify it as neuter.

5. EVALUATION

5.1 Test nouns

I evaluated my system by applying the two approaches from the preceding section individually and in combination to two different lists of nouns. I extracted these lists by exploiting the convenient fact that the dialects of Low Saxon from the German side of the border are often written using the famous / infamous German noun capitalization rule, i.e. all nouns are always capitalized.⁷ I extracted one list of frequent nouns by collecting all types from the corpus which occurred more than 20 times and at least three times as often with an uppercase first letter than with a lowercase first letter. This produced a list of 1200 candidates. I also extracted a list of longer, rare nouns in order to see how well the first algorithm would be able to cope with data sparseness and to evaluate the second method. This second list comprised all types at least 10 characters long which occurred between 5 and 20 times in the corpus and more than three times as often with an uppercase first letter than with a lowercase first letter. This list of rare nouns initially contained 519 types. For the present study, I decided to avoid extra difficulties by manually scanning through these candidate lists and excluding all proper names (which do not show gender features in the same way as common nouns), all plural nouns (because I currently have no way to combine the evidence for singular and plural forms), and all non Low Saxon words. This resulted in final candidate lists of 670 frequent nouns and 290 rare nouns⁸. I classified all these nouns by hand to serve as gold standard during the evaluation.

5.2 Evaluation measures

In the present study, all the evaluation measures I provide are calculated for the type level and not weighted by the frequency of the different nouns. I will use two different measures of coverage and two different error rates for evaluating the two algorithms.

⁷ Nakov et al. [4] also use this as an important clue for detecting nouns.

⁸ Most excluded words were proper nouns or plural nouns. Almost all words on the list were in fact nouns. This demonstrates the effectiveness of a noun extraction algorithm based on capitalization for Low Saxon.

They are defined as follows:

$$\begin{aligned} \text{Coverage1:} & \quad (\text{ALL} - \text{UNKNOWN}) / \text{ALL} \\ \text{Coverage2:} & \quad (\text{ALL} - \text{UNKNOWN} - \text{COMMON}) / \\ & \quad \text{ALL} \\ \text{Error1:} & \quad (\text{FP_COMMON} + \text{FP_NEUT}) / (\text{ALL} \\ & \quad - \text{UNKNOWN}) \\ \text{Error2:} & \quad (\text{FP_MASC} + \text{FP_FEM} + \text{FP_NEUT}) \\ & \quad / (\text{ALL} - \text{UNKNOWN} - \text{COMMON}) \end{aligned}$$

ALL is the number of all candidate types; FP represents the number of false positives for the respective class. As the different classes on the same level of coverage are mutually exclusive, i.e. no noun has more than one gender⁹; the sum of false positives of all relevant classes is equal to the overall number of errors.

I thus give information about how many nouns of all candidate nouns were classified into *COMMON* vs. *NEUT*, this is represented by Coverage1. And I provide Coverage2 as the portion of types for which a more precise classification into *MASC* vs. *FEM* vs. *NEUT* was made.

The same distinction applies to the error measures. Error1 is the error rate for the laxer decision *COMMON* vs. *NEUT*; Error2 represents the quality of the more precise classification as *MASC*, *FEM*, or *NEUT*. As *COMMON* subsumes *MASC* and *FEM*, I counted the correctly assigned classes *MASC* and *FEM*, as true positives for the class *COMMON* for the calculation of the Error1 measure.

5.3 Experiments

As a first experiment, I tested different context widths for the observation gathering stage of the clue word approach. This initial experiment shows that larger contexts introduce too much noise and that the optimal solution is to consider only the one token immediately left of the noun. Including more tokens on the left or right of the noun increased the error rate as much as the coverage. Applying the first method to the list of frequent nouns resulted in a Coverage1 of 89.1 % with an Error1 of 3.69 % for the classification

NEUT vs. *COMMON*, and a Coverage2 of 59.4 % with an Error2 of 4.52 %. Including the first two tokens to the left of the noun during data collection increased Coverage1 to 92.99 %, Error1 to 7.38 %, Coverage2 to 67.76 %, and the Error2 measure to 9.69 %. All other wider contexts left and right from the noun led to an even greater deterioration of the classification results. Testing the clue-word method on the list of rare nouns yielded a much lower coverage but surprisingly good error rates. For these results and the results of the following experiments, please see table 2.

In a second experiment, I took the classification of the frequent nouns produced by the first approach as the basis for classifying possible compound nouns in the set of longer, rare nouns. Using only this second approach on the rare nouns achieved a Coverage1 of 25.86 % and a Coverage2 of 20 %. The error rates were only slightly higher than those achieved with method one. However, the best result for the rare nouns were achieved by classifying them first with the clue-words approach and then looking for yet unclassified compounds using the gender information from the frequent noun list, cf. *clue words + compounds (II)* in table 2.

Table 2. Results of the experiments

Experiment	Cov. 1	Error 1	Cov. 2	Error 2
Frequent nouns (Baseline error all classified as <i>MASC</i> : 58.81 %)				
<i>Clue words alone</i>	89.1 %	3.69 %	59.40 %	4.52 %
<i>Clue words + Compounds (I)</i>	89.55 %	3.67 %	60.00 %	4.48 %
Rare nouns (Baseline error all classified as <i>FEM</i> : 57.24 %)				
<i>Clue words alone</i>	64.83 %	2.66 %	26.21 %	1.32 %
<i>Compounds alone</i>	25.86 %	4.00 %	20.00 %	5.17 %
<i>Clue words + Compounds (II)</i>	74.48 %	2.78 %	39.31 %	2.63 %

Also, if the classification obtained from the compound method was more precise than the initial classification by the clue-words method, the initial classification was revised, i.e. if method one had classified a noun as *COMMON* and it was found to be a compound by method two, the initial classification was overridden if the head component of the compound had been classified either as *MASC* or as *FEM*. This yielded a ten percent increase for the coverage measures

⁹ In very rare cases, nouns can indeed have more than one gender either because of homography or of dialectal differences, cf. also [3].

compared to using method one alone and very good error rates.

In a third experiment, I first classified both of the lists with method one and after that applied the compound guessing strategy within each list (i.e. I tried to find compounds within the list of frequent nouns using only data from this same list and the same for the list of rare nouns). This alternative resulted in a very small increase of coverage for the frequent nouns – see *clue words + compounds (I)* – and no increase at all for the rare nouns which is understandable because they all already are at least ten characters long.

As a last interesting result, consider table 3 which contains the percentage of all true *MASC*, *FEM*, and *NEUT* nouns in the list of frequent nouns which have been classified precisely into *MASC*, *FEM*, or *NEUT*.

Table 3. Precise coverage of true *MASC*, *FEM*, and *NEUT*

	true MASC	true FEM	true NEUT
Precise Coverage	67.03 %	31.00 %	92.12 %

FEM seem to be especially difficult to find, while *NEUT* is the easiest gender class. The reason is simply that there are consistent clues for neuter nouns in all dialects (e.g. the definite article *dat*), whereas only a minority of dialects has any real clues for feminine nouns. On the other hand, the classification as *NEUT* is the one with the lowest precision (92.76 % using the method one for the frequent nouns vs. 97.74 % for *MASC* and 95.65 % for *FEM*), probably because *dat* is also frequently used as a complementizer.

5.4 Discussion of results

The results of this study seem quite encouraging to me. I had not expected to get such good results with such simple methods. The task of learning noun gender does not seem to be a very difficult challenge, at least not for Low Saxon. However, one should not underestimate the difficulty because of the ongoing

loss of gender distinction in determiners in many dialects and the nature of the corpus used in this study which contains a great variety of different dialects and writing systems (thus worsening the problem of data sparseness). Comparing my results to those obtained in Cucerzan and Yarowsky [3], the present algorithm performs slightly worse. They achieve accuracies between 95 % and 99 % for Romanian, French, Spanish, Slovene, and Swedish. However, one has to keep in mind that these are standardized languages and that Cucerzan and Yarowsky also used additional morphological cues. However, I think that it could be advantageous to use their automatic method of learning context clue words because I just might have chosen the right clue words by chance this time. There are still a lot of problems and unsolved issues: one obvious next step would be to try to find a good way of combining singular and plural noun forms into paradigms.

6. CONCLUSION

The present study has shown that even very simple methods of learning morphosyntactic information can be quite effective and hopefully can be used in bootstrapping more and more useful information about the lexicon of Low Saxon.

7. REFERENCES

- [1] The Ethnologue. <http://www.ethnologue.com/>.
- [2] Brent, M.R. (1993): From Grammar to Lexicon: Unsupervised Learning of Lexical Syntax. Computational Linguistics Volume 19 Number 2, pp. 243 – 262.
- [3] Cucerzan, S., and Yarowsky, D. (2003): Minimally Supervised Induction of Grammatical Gender. In: Proceedings of HLT-NAACL 2003 Edmonton, Main Papers, pp. 40 – 47.
- [4] Nakov, P., Bonev, Y., Angelova, G., Cius, E., and von Hahn, W. (1996): Guessing Morphological Classes of Unknown German Nouns. In: Proceedings of RANLP 2003 Borovets, Bulgaria.