Anna Sophia Stein
anste145 2394420

Modeling MorphologyWiSe 22/23
Prof. Ingo Plag

Implementation Task

1 Introduction

Reading the Arndt-Lappe (2014) paper in class made me think about whether a Naive Discriminative learning (NDL) model would be just as successful as an Analogical model in predicting *-ness* and *-ity* assignment, using the same data. In order to act on my curiosity, I chose to conduct a small-scale replication study of Arndt-Lappe (2014) using NDL. Therefore, the goal of this study is to run an NDL model on the data used in Arndt-Lappe (2014) and to compare the performance of AM and NDL.

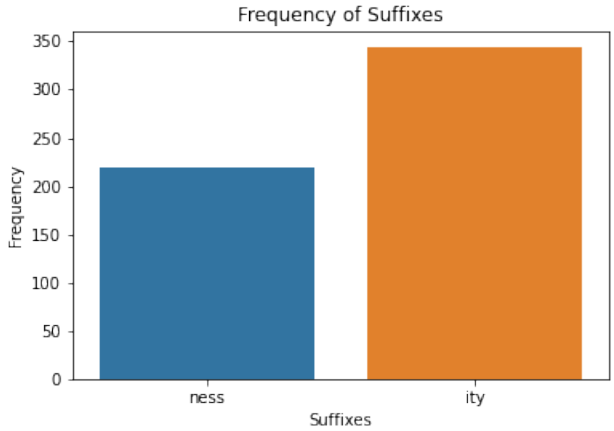
The original study by Arndt Lappe uses three datasets from the Oxford English Dictionary, each from one century, to predict the nominalisation suffix of English neologisms. Based on evidence from previous studies, certain criteria were selected to be represented in the model as input for the predictions. This information consists of the last two syllables of a word, split into onset, nucleus and coda, as well as the syntactic function of the word, and of course the affix. I derived three NDL models from this to be able to test the performance of different cue-outcome structures. The structures of each model can be seen in table 1 below.

	Cues	Outcomes
M1 syllable	sh=n sh=n word additionality	ity
M2 segments	sh = n sh = n word additionality	ity
M3 filtered	sh n sh n word additionality	ity

Table 1: Cue-outcome structures of the three models.

2 Data

The scope of this analysis only allows for the use of one of the three datasets originally used in the study by Arndt Lappe. The 20th-century dataset consists of 564 words, of which 220 include -ness and 344 include -ity.



3 Implementation

In the first step, the data was preprocessed. A header was added to the file, and extra commas were removed from the suffix and word column. Additionally, a column containing numbers was removed from the data frame since it would not be used to form cues or outcomes.

The Python implementation of NDL, *pyndl* Sering *et al.* (2017), uses specific formatting for input files, so-called event files. Cues and outcomes need to be in separate columns in a tab-delimited file. Each line of the files represents one learning event. If there are multiple cues or outcomes in the same line, they must be separated by an underscore.

The *pyndl* package has a function to preprocess texts to that format, but this does not work for more specific, nonconsecutive cues and outcomes. The cue and outcome constellations were therefore formed manually by joining the relevant columns of the table. For model one, all syllable info was joined in order to create a syllable1 and a syllable2 cue. For model two, each segment represented an individual cue. This was copied for model three so that only the equal signs needed to be filtered out of the cues. Finally, the files were compressed into a gzip format and used to train the model.

In order to be able to evaluate predictions of the NDL model, I computed the activation matrix for each model using the corresponding function from the pyndl package. The suffix with the higher prediction was interpreted as the prediction of the model. I then wrote a script to calculate the f1 score, the macro-averaged f1 score, and the overall accuracy percentage from the model.

4 Results

2 show the results of the evaluation of the three models. Model one performed considerably worse than Models two and three, especially concerning the *-ness predictions*. The performance of models two and three is almost identical. Overall, the performance of the models is worse on the *-ness* classification, the f1 score of in model one .26 is especially low. This is most likely due to the skewed training data as can be seen in **??**.

	F1 -ity	**F1 -ness**	**F1 macro-averaged**	**Correct Predictions (%)**
M1 syllable	.78	.26	.52	66
M2 segments	.87	.76	.82	83
M3 filtered	.87	.76	.82	83

Table 2: Evaluation of the three models.

Compared to the scores of the AM model in Arndt-Lappe (2014), NDL performs worse. A more detailed study could focus a more fine-grained analysis of the results, in particular regarding the f1 score and the C-score that is missing in this anaylsis. For unbalanced data sets one may want to consider precision and recall with a different weight than what is done in the calculation of the f1 score here. Alternatively, reproducing this study with a balanced data set may also be interesting.

References

Arndt-Lappe, Sabine. 2014. Analogy in suffix rivalry: The case of english-ity and-ness. *English language & linguistics*, **18**(3), 497–548.

Sering, Konstantin, Weitz, Marc, Künstle, David-Elias, Schneider, Lennart, & Shafaei-Bajestan, Elnaz. 2017. *Pyndl: Naive discriminative learning in python.*

5 Appendix

The full code as well as the data used to conduct this experiment can be found on my [github page].