university of groningen / faculty of science and engineering

A Comparison of Data-Driven Morphological Segmenters for Low-Resource Polysynthetic Languages: A Case Study of Greenlandic

Bachelor's Project Thesis

Barbera de Mol (b.c.de.mol@student.rug.nl)
Supervisor: dr. J.K. Spenader

Abstract: Morphological segmentation is vital in many areas of natural language processing, including machine translation. However, very little research in this field has been performed on low-resource polysynthetic languages. Rather, most research in has focused on languages with existing resources and moderate morphological inflections. Greenlandic is such a polysynthetic language, and due to its relatively few native speakers, few resources have been developed. For this paper, the author manually crafted the largest publicly accessible annotated dataset of Greenlandic morphological segmentations. With this dataset, intrinsic experiments are conducted where seven different methods for morphological segmentations including one rule-based system and six (supervised) machine learning systems are compared through calculating precision, recall, F1-score and accuracy using tenfold cross-validation. The fully-supervised (F1-score = 0.633, accuracy = 0.542) and semi-supervised (F1-score = 0.631, accuracy = 0.553) Conditional Random Fields perform best. Extrinsically, a baseline with no segmentation and the six most promising models from the intrinsic evaluation are implemented in a neural machine translation model and their BLEU scores are compared. The results for the extrinsic evaluation were however not reliable because the neural machine translation models performed below par.

1 Introduction

A morpheme is the smallest unit of language that can convey meaning, and morphological segmentation is the language technological task of automatically identifying morphemes. This process is extremely vital in, amongst others, the domain of machine translation because all these separate units of meaning need to be translated in order to achieve an accurate translation. In analytic languages, such as English, word order and helper words are used to signify the relationships between words. Such languages can generally be accurately translated word to word, since their usage of inflections is minimal. On the opposite side of the spectrum however, we have polysynthetic languages like Greenlandic. These languages can be characterised by the fact that words are built up out of multiple concatenated morphemes, which enables a single word in a polysynthetic language to express what would be an entire sentence in English. Example (1) illustrates this for Greenlandic, where the entire word is split up into its morphemes as follows.

(1) ullaakkorsioreerusussaanngilanga
ullaakkor-sio-ree-rusu-ssaa-nngi-la-nga
breakfast-eat-finish-want-FUT-NEG-IND-1Sg
I would not want to finish eating breakfast

What increases the challenge of segmenting polysynthetic languages even more is that many of these languages are among the world's most endangered languages (Klavans, 2018). Examples of two highly polysynthetic and low-resource languages are Greenlandic and Inuktitut. These languages are the two most widely spoken Inuit languages as part of the Eskimo-Aleut family with 57k (Ethnologue, 2015) and 39k (O'Donnell & Anderson, 2017) speakers respectively. To sketch a contrast: whereas English takes up 59.6% of the web, Greenlandic takes up less than $3 \times 10^{-4}\%$.[1]

Originally, the research described in this paper was motivated by the news translation task from the Workshop on Machine Translation 2020.[2] Specifically, the research into Greenlandic started as a possible aid for the submissions of the neural machine translation (NMT) systems that were created by the University of Groningen in the English ↔ Inuktitut (EN ↔ IU) language pair. Last year, Toral et al. (2019) found that NMT was greatly improved for low-resource languages when synthetically generated backtranslated data from a similar language was also used to train the NMT system. This year, Greenlandic was therefore examined to investigate if the same improvement in NMT occurred for Inuktitut when Greenlandic was added.

The need for a large amount of data for many machine learning systems combined with the fact that few resources have been developed for Greenlandic makes data collection a significant challenge. The lack of large text corpora makes morphological segmentation even more vital because it can reduce data sparsity and the size of the vocabulary. This is because words that are built up out of many morphemes generally occur very infrequently due to the specific meaning they convey; splitting them up into morphemes provides smaller segments that do occur more frequently. The reduction of the vocabulary size then occurs because words do not have to be added with their many inflectional forms, but rather only the root is added together with all the morphemes in the language. Combinations between roots and morphemes can then be made, making for a much more compact dictionary. An additional benefit is also that unknown roots can be identified by removing the known morphemes that are attached to them, which in turn can help expand the Greenlandic lexicon.

The experiments are based on the hypothesis that some of the segmentations methods that have been used for higher-resource non-polysynthetic languages in previous research can also be applied to low-resource polysynthetic languages. Especially methods that have been used for agglutinative synthetic languages (e.g. Finnish and Turkish) are of interest. This is why we will compare several different morphological segmenters, ranging from rule-based to various levels of supervised machine learning. Which segmentation system works best with Greenlandic is determined by examining their outputs intrinsically as well as extrinsically. Intrinsically, their outputs will be examined by comparing the model's F1-scores and their token-accuracy. Extrinsically, we will also investigate which system best aids NMT for Greenlandic → English (KL → EN).

2 Theoretical Framework

There are many different methods to develop a morphological segmenter. The traditional way was to create them manually using a dictionary and a rule-based database. This unfortunately is an especially time-consuming project to create, and it is also very hard to keep up to date as languages are ever evolving. Although inflections often do not change, their databases cannot always generate analyses for unknown words. Because these rule-based segmenters are not very robust, are generally quite slow and are often not readily-available for lower-resource languages, there arose the need for automated morphological segmentation systems. More recent approaches therefore use machine learning, where the degree in supervision ranges from completely unsupervised (Creutz & Lagus, 2005a; Sennrich et al., 2015), to semi-supervised (Ataman et al., 2017; Grönroos et al., 2014; Lafferty et al., 2001; Virpioja et al., 2013) and completely supervised (Vaswani et al., 2017).

Most research into computational morphological segmentation has focused on higher-resource languages with only a moderate number of inflections in mind, whereas the actual challenge in segmentation lies with the highly inflectional languages. For Greenlandic, no research into data-driven morphological segmentation known by the author has been conducted, but other state of the art papers into the segmentation of the low-resource polysynthetic languages Mexicanero, Nahuatl, Yorem Nokki and Wixarika report F1-scores ranging from 0.75 to 0.88 (Eskander et al., 2019; Kann et al., 2018).

The difficulty in segmentation is not only due to polysynthetic languages often being sparse in data, but also because building consistent corpora of annotated data is extremely challenging due to their morphological complexity (Klavans, 2018).

[1] https://w3techs.com/technologies/overview/content_language/

[2] http://www.statmt.org/wmt20/translation-task.html

Furthermore, a challenge specific to NMT is that a lot of information is often lost between translation from an information-rich to an information-poor language and vice versa not enough information is present to go the other way around (Mager et al., 2018). Again, morphological segmentation could be an important step into properly translating from an information-rich to an information-poor language. Using segmentation, individual parts of meaning from words in information-rich languages can be filtered out more accurately and therefore also translated better.

Additionally, three more challenges arise specifically for Greenlandic because of its predominantly fusional character and inflectional system (Mahieu & Tersis, 2009). First of all, a fusional character means that a single morpheme can express multiple meanings;[3] Greenlandic for example does not distinguish inflectionally between the present and past tense and has no distinction between male and female third person. See Example (2).

(2) sinippoq
sinip-poq
sleep-IND.PRS/PST.M/F.3Sg
?he sleeps ?she slept

Secondly, a characteristic of languages with a fusional inflection system is that morphemes can merge together,[4] making it more difficult to extract them. This is illustrated in Example (3), where the morpheme *-qa(r)-* show the original suffix, and the version below without the letters in parenthesis shows the actual segmentation. Due to morphophonemic constraints, a *-rq-* sequence is forbidden in Greenlandic, so the former letter has been removed. *-qa-* here is the surface form of its morpheme *-qar-*, and the surface forms of morphemes will hereafter be referred to as morphs.

(3) meeraqanngilatit
meera-qa(r)-nngi-la-tit
meera-qa-nngi-la-tit
child-have-NEG-IND-INT.2Sg
don't you have any children

Lastly, in Greenlandic it also often occurs that the last letter of the morpheme is altered because of the first letter in the subsequent morpheme, see Example (4). The last part of the word *-fimmi* is actually a combination of the morphemes *-fik-* and *-mi*. Again, due to the morphophonemic constraints, *-km-* is an illegal sequence. In this case, the *k* has been turned into an *m*. Why the *m* is added to *-mi* instead of *-fi-* will be discussed in the next section with annotation choices.

(4) illoqarfimmi
illu-qar-fik-mi
illo-qar-fi-mmi
house-have-place.where-LOC
in town

To conclude these challenges, it is already a difficult task on its own to extract the proper morphemes in terms of underspecificity and fuzzy morpheme boundaries. Even when the morphemes have then been extracted properly, it still remains difficult to translate due to problems in building consistent bilingual corpora and problems with translating from an information-rich to an information-poor language and the other way around.

3 Datasets and Evaluation

3.1 Raw monolingual data

As discussed, Greenlandic is a low-resource language. Unfortunately, no large corpus is available for this language, making data collection a challenge on its own. For the purpose of the experiments on Greenlandic segmentation, a dataset was created using two Danish-Greelandic dictionaries together with a wikidump and websites crawled using Bitextor[5] (see Appendix A for the specific sources). The data from Bitextor provided aligned parallel data which will be helpful for translation, but for the morphological segmentation systems only the monolingual Greenlandic data was used. The same goes for the dictionaries, where only the Greenlandic half was added to the dataset. See Table 3.1 for the amount of data per set.

All data was then filtered using a tailor-made Python script, which makes all words lowercase, filters out non-Latin script and then continues to remove noise. This noisy data is lots of linking data from sites (e.g. words like 'http', 'www') as well as English and Danish text. These foreign languages

[3] https://glossary.sil.org/term/fusional-language
[4] https://dictionary.apa.org/fusional-language
[5] https://github.com/bitextor/bitextor

Datasets	Words
Wikidump	1044298
-unique	10091
Dictionary	7106
-unique	5443
Bitextor	1821170
-unique	111809
Total	2872574
-unique	121639

Table 3.1: The total and unique number of words gathered per dataset.

were removed by creating a list of the most commonly occurring words in the category and deleting the entire line containing any such word. Although Greenlandic has lots of loanwords from Danish, we opted to exclude these words in order to create a more uniform dataset. The author hypothesizes that this allows the morphological segmenters based on machine learning to detect patterns (e.g. morphemes) more easily by avoiding confusion with words from an analytic language with largely different patterns.

Afterwards, lines containing only a single word were also removed in all datasets apart from the dictionary. This is because individual words are hard to filter, as it is hard to categorize them. Not taking them out would therefore likely lead to a lot of noise. Additionally, words shorter than three letters or longer than 30 were also removed from the data.

Lastly, simple phonotactic constraints were applied to further erase illegal words. Phonotactics concern the allowed combinations of phonemes in a language, and for Greenlandic these constraints include removing words that do not start with ['a', 'o', 'u', 'i', 'e', 'p', 't', 'k', 'q', 's', 'm', 'n'] or end with this same set minus 's' and 'm' (Fortescue, 1984), as well as removing any word in which the same letter is repeated three or more times. This filter also helps with excluding more Danish loanwords that might have been missed by the language filter. The number of words remaining after the addition of all the filters on top of each other can be seen in Table 3.2. The remaining data is then saved as a list of unique words preceded by an integer which illustrates how often the word occurred in the dataset, so for example *paasilertoruminaatsoq* occurred 31 times and *akileraarutit* was found 64 times in the data, formatted as can be seen in Example (5).

(5) `31 paasilertoruminaatsoq`
`64 akileraarutit`

3.2 Annotated monolingual data

Apart from the raw monolingual data, some morphological segmenters require additional annotated data. Some segmenters do not even use the raw data at all and only use the annotated data. For Greenlandic, no such publicly accessible dataset known by the author existed until the publication this dataset. The annotated data contains 640 words and was in large part gathered by hand using two courses on learning Greenlandic with an emphasis on individual morpheme meaning in words. The data was in part revised by a native Greenlandic speaker. The words were formatted as portrayed in Example (6), where the full word is followed by its morphological parts.[6]

(6) `ungasinngisaani ungasi nngi saa ni`
`illoqarputit illo qar pu tit`

[6] https://biturl.top/n2qema

Filters	Wikidump Words	Bitextor Words	Example
- tokens	1094965	1879626	“”
- links and internet jargon	892326	1837850	www nuuk gl
- Danish	844292	1444065	på grønlandsk
- English	810031	1395241	This is reserved
- loose words and titles	807078	1225479	br
- words with a unusual length	753284	1191322	se
- words with forbidden phonotactics	687014	995440	regeringen

Table 3.2: The number of words remaining from the wikidump each time a new filter is applied on top of the older ones.

```
illut illu t
tassa tassa
```

As briefly touched upon in Section 2, many morphemes cannot be split perfectly between two phonemes because of Greenlandic's fusional character. Reiterating Example (4) below, it is visible that the morphs *-fik-* and *-mi* together turn into *-fimmi* because *-km-* is an forbidden sequence in Greenlandic. Recalling the statement in Section 2 that building a consistent corpus for polysynthetic languages is a challenge, the author opted to make the design choice that for all such cases the changed letter should be added to the latter morph. For the example, *-fimmi* is split up into *-fi-mmi* instead of *-fim-mi*. Although the latter might be more correct linguistically, we hypothesize that the former leads to a better model for NMT. The reasoning behind this is that this leads to a smaller lexicon of morphs because instead of having to include *-fik-*, *-fim-*, and many more with all the possible endings, this allows you to only add *-ffik-*, *-fik-*, *-fi*, *-mi* and *-mmi*. A smaller lexicon is preferable because this results in less infrequent morphs leading to better translations of not commonly occurring words. Additionally, the computational complexity is reduced.

(4) illoqarfimmi
illu-qar-fik-mi
illo-qar-fi-mmi
house-have-place.where-LOC
in town

3.3 Evaluation

For evaluation, two methods will be used. First of all, the segmenters will be evaluated intrinsically based on their ability to segment individual words. For this, precision, recall, the F1-score and token-accuracy are calculated. For segmentation systems that need annotated data, tenfold cross-validation will be used to ensure that the models are trained with as much data as possible as well as that the testing data, that originates from the same set, stays valid. For systems that will not be using the annotated data, the entire annotated dataset can simply be used as gold-standard testing data. In the end, all models will then have been tested on the same testing set.

The second manner of evaluation will be extrinsic. This will be comparing the effect of the different segmentation systems on the overall BLEU score for a KL → EN NMT system. This method will examine to what extent segmentation actually aids or harms translation, which provides us with a good sense of what segmentation method helps in the big picture.

4 Intrinsic Experiments

The experiments section will compare seven different systems for segmenting Greenlandic morphology. Firstly, GroenOrd,[7] a rule-based segmentation system will be shown. Next, Byte Pair Encoding (Gage, 1994) and Morfessor Categories-MAP (Creutz & Lagus, 2005a) are discussed as unsupervised segmenters. After, four semi-supervised segmenters will be expanded on, namely Morfessor 2.0, Morfessor FlatCat, Linguistically-Motivated Vocabulary Reduction and Conditional Random Fields (Ataman et al., 2017; Grönroos et al., 2014; Lafferty et al., 2001; Virpioja et al., 2013). Lastly, this section will consider the completely supervised Transformer models (Vaswani et al., 2017).

4.1 Rule-based segmenters

Rule-based systems are handcrafted and generally provide quite accurate segmentations. Often however, they are too slow to be used as a morphological segmentation system in a real-time NMT environment and they do not allow for the processing of large chunks of texts needed for training a NMT system. Additionally, rule-based segmentation systems often only segment the words that are in their dictionary, leading to unknown words not being parsed at all. Also, rule-based systems are generally based on one language only, whereas this research aims to investigate morphological segmentation systems that would also work for other low-resource polysynthetic languages. Still, because of the fact that they are built upon a lot of knowledge about a language, having a rule-based segmenter for comparison can give a lot of insight how more automated systems perform in comparison to these handcrafted systems.

[7] https://www.groenord.dk

4.1.1 GroenOrd

GroenOrd[8] is an "electronic version of five Greenlandic dictionaries (1871-1997+)" created by Henrik Vagn Aagesen. Using these dictionaries, this rule-based segmenter is able to extract "the meaningful parts (i.e. morphemes) of a given Greenlandic and/or Danish word."

Evaluation for GroenOrd is only done intrinsically by segmenting all words in the golden-standard annotated dataset. The largest encountered problem in the results is, as was expected, robustness. This can be seen in Example (7) and (8), where the two words in Example (7) have a very similar buildup in which you can recognize the morphs *-kkor-* and *-sior-*, but they are segmented in a completely different fashion because the morphs are not recognized in the first word. The same applies for the two words in Example (8), where the root *arna-* is not recognized in the first word but it is in the second.

(7) `ullaakkorsior poq`
`unnu kkor sio rusup put`

(8) `arnaviaq`
`arna tut`

Another reason why the F1-score and the accuracy (see Table 4.7) of GroenOrd is lower than one might reasonably expect is because of the design choices in the golden-standard data. Many of the design choices of GroenOrd are the same as in the annotated data, as in Example (9), where in the segmentation of *kaffisorusuppunga* GroenOrd also illustrates that it prefers the beginning of a morph over the end; it is the case that the *r* in *-sorusup-* belongs to both the morph *-sor-* and *-rusup-* and GroenOrd allocates this *r* to the latter morph.

Other choices made with NMT in mind however, do not correspond. This can be seen in Example (10), where the *-mm-* in the annotated data is also attached to the latter morph, but in GroenOrd's segmentations it is split into two separate morphs. It should be noted however that in both of these examples, the rule-based segmenter also failed to recognize other morphs properly that were not based on design choices because the roots or morphs were not recognized in the dictionary.

(9) GroenOrd: `kaffiso rusup pu nga`
Annotated: `kaffi so rusup pu nga`

(10) GroenOrd: `illoqarfim m i`
Annotated: `illo qar fi mmi`

So, although the rule-based system is filled with a lot of highly relevant data, the data is not reliably accessible in the way needed for morphological segmentation. On top of that, the system in place for Greenlandic is not capable of handling texts of more than one word, making it very impractical to use for training morphological segmentation and NMT systems. This, together with previously mentioned arguments such as language compatibility, calls for a more data-driven approach into morphological segmentation.

4.2 Unsupervised segmenters

Generally speaking, the idea behind the data-driven segmenters is to find a balance between the size of the lexicon and the cost of the model. It is good for the model to have a small lexicon to reduce infrequent sub-words, as discussed in Section 3.2. The precision of the model however should also be as good as can be, meaning that the smallest possible lexicon of only the individual letters of the language's alphabet obviously does not perform well. A balance between these two therefore needs to be found in order to create an optimal segmentation model.

Unsupervised morphological segmenters are especially attractive to low-resource languages as they do not require any annotated data at all, and annotated data can be difficult to find for these languages. Unsupervised segmenters can be frequency-based, such as Byte Pair Encoding, but they can also rely on probabilities with more linguistic features in mind, such as Morfessor Categories-MAP. The intrinsic evaluation for these systems is performed by segmenting all words in the gold-standard dataset and comparing them to their golden-standard goal segmentations.

4.2.1 Byte Pair Encoding

BPE (Gage, 1994) is a simple data compression algorithm which identifies bytes that commonly occur together and replaces them with a new and unique byte. This technique was first applied to

[8] https://www.groenord.dk/

Merges	Precision	Recall	F1-score	Accuracy
4000	0.151	0.208	0.175	0.042
10000	0.172	0.198	0.184	0.064
20000	**0.191**	**0.205**	**0.198**	0.081
30000	0.181	0.182	0.181	0.084
40000	0.173	0.169	0.171	0.085
50000	0.177	0.167	0.172	**0.087**
60000	0.173	0.163	0.168	0.085

Table 4.1: The effect of the number of merges in the BPE algorithm on precision, recall, f1-score and accuracy.

natural language processing by Sennrich et al. (2015) to split up words into frequently occurring characters, which inherently are often similar to morphs. At the start of the process, a word occurs as a sequence of its characters and a token to mark the end of the word. Then, characters that often occur together are merged together into a new unique symbol. The way in which BPE segments words is therefore also largely dependent on how many merges the program is allowed to make, and different number of merges work better for different languages and applications.

Because of its simplicity and robustness, BPE is currently the most widely used segmentation technique in natural language processing. This is what makes it an excellent baseline to compare other more linguistically refined machine learning segmentation systems with. Important to note is that BPE was not designed to extract morphs in particular, but rather to identify commonly occurring characters. It can therefore be expected that BPE will perform relatively poorly on the intrinsic evaluation.

Using Sennrich's original program,[9] a model was trained on the full monolingual dataset using different numbers of merges, see Table 4.1. No dropout was used, meaning that the programs splits all the words consistently. With dropout, there exists the chance at every merge step that this merge is randomly cancelled out. Depending on how you value F1-score and accuracy, there are two different best models; the one with 20k merges has the best F1-score (`0.198`) and the model with 50k merges has the best accuracy (`0.087`).

The two amounts of merges fit more or less in the order of degree with 30k–40k, which was most often used for BPE during WMT17 and WMT18. The difference between the two is characterised by the fact that the model with 50k merges has consistently fewer splits in the words than the model with 20k does, as can be see in Example (11). For extrinsic testing, the model with the highest F1-score, so with 20k merges, will be created. Both because the F1-score is a more robust measurement than the accuracy and because the system with 20k merges has less unique tokens, making it more suitable for NMT.

(11) 20.000: `mi ki voq`
50.000: `miki voq`

The extrinsic testing, however, will make use of a transformer model type, and, based on the recommendations from Ding et al. (2019) for this type of NMT, the range of merges should be around 0-4k. Therefore, a NMT model with only 4k merges was also trained for extrinsic evaluation.

4.2.2 Morfessor Categories-MAP

This first program from the Morfessor family we will discuss is completely unsupervised. As the name suggests, Morfessor Categories-MAP (Creutz & Lagus, 2005a) is based on a probabilistic Maximum A Posteriori (MAP) estimation. It is an extension of the (now outdated) Morfessor Baseline model, which was originally developed by Creutz and Lagus (2005b). In Morfessor Categories-MAP, morphological categories are attached to morphs based on a first-order hidden Markov model (HMM). Morfessor Baseline assumed independence between morphs, meaning that *learn + s* had the same probability to be segmented as *s + learn*, and the advantage of the use of a HMM is that context

[9] https://github.com/rsennrich/subword-nmt

Perplexity Threshold	Precision	Recall	F1-score	Accuracy
1	0.397	0.353	0.374	0.204
10	0.330	**0.454**	0.382	0.178
50	**0.428**	0.409	**0.418**	**0.233**
100	0.380	0.359	0.369	0.213
200	0.367	0.323	0.334	0.187
400	0.359	0.286	0.318	0.176

Table 4.2: The effect of the perplexity threshold in the Morfessor Categories-MAP algorithm on precision, recall, f1-score and accuracy.

is now taken into account when calculating probabilities. Furthermore, Morfessor Categories-MAP uses a hierarchical lexicon, which reuses already existing segments when finding new ones. This means that the segmentation of a word is not encoded by its letters, but rather by the references to its individual segments.

An important parameter for this system is the perplexity threshold. This perplexity is based on the morphological complexity of a language and the size of the training set, where more training data and larger morphological complexity leads to a higher perplexity threshold. The difference the threshold makes while segmenting is best illustrated with its extremities: the threshold of one and the threshold of 400. As can be seen in Example (12) and (13), a higher perplexity threshold expects morphs to occur relatively less frequently as it assumes higher morphological complexity. This is shown by having larger morphs when the threshold is set to 400 compared to one. Although the resulting difference might look similar to the difference in amount of merges in BPE, Example (13) shows that the divisions are not as clean. Whereas BPE without dropout consistently merges more morphs together, the perplexity threshold also influences the placement of the morph boundaries.

(12) 1: `unnu kkor si or poq`
400: `unnukkor siorpoq`

(13) 1: `akunn attu uk uju llunga`
400: `akunnat tuukuju llunga`

Through empirical testing (see Table 4.2), the threshold for Morfessor Categories-MAP was set at 50 to achieve the best linguistic correctness in intrinsic testing with a resulting F1-score of `0.414` and an accuracy of `0.233`.

4.3 Semi-supervised segmenters

Semi-supervised, or minimally-supervised, segmenters have a head start over unsupervised segmenters when identifying morphs. By considering data from a gold-standard, the semi-supervised models can consider, amongst other things, how many characters an average morph should have, how many morphs there should approximately be in the words, and of course what some morphs already are.

In this section, four different methods that use both the raw and the annotated data will be compared. Because the created golden-standard dataset with segmented words is relatively small, ten-fold cross validation will be used to evaluate these systems intrinsically in order to assure all data is used while the research stays valid.

4.3.1 Morfessor 2.0

Morfessor 2.0 (Virpioja et al., 2013) is also a newer implementation of Morfessor Baseline and, amongst other improvements, Morfessor 2.0 introduces semi-supervised training. The algorithm uses the maximum a posteriori estimation as a cost function, which is used as base to again train a model resembling a hidden Markov model. The cost is based on the likelihood and priors, which are gathered respectively from model assumptions and based on applying the minimum description length principle (Rissanen, 1978) to the data.

Table 4.3 shows the result of the testing with a different number of morph types specified each time. When tuning the desired number of morph types, the α parameter is indirectly tuned as $\alpha = \frac{m1}{m2}$, where *m1* is the initial vocabulary size of the corpus and *m2* is the desired vocabulary size. Without any specification, Morfessor 2.0 will set the de-

# Morph Types	Precision	Recall	F1-score	Accuracy
N/S	**0.597** (SD: 0.074)	0.566 (SD: 0.073)	0.581 (SD: 0.071)	**0.449** (SD: 0.072)
2000	0.571 (SD: 0.054)	0.570 (SD: 0.051)	0.570 (SD: 0.049)	0.409 (SD: 0.039)
3000	0.584 (SD: 0.050)	0.573 (SD: 0.051)	0.578 (SD: 0.045)	0.424 (SD: 0.046)
4000	0.594 (SD: 0.058)	**0.575** (SD: 0.056)	**0.584** (SD: 0.054)	0.445 (SD: 0.049)
5000	0.589 (SD: 0.059)	0.559 (SD: 0.052)	0.574 (SD: 0.054)	0.425 (SD: 0.056)

Table 4.3: The effect of the number of morph types in the Morfessor 2.0 algorithm on precision, recall, f1-score and accuracy.

Perplexity	Precision	Recall	F1-score	Accuracy
1	0.594 (SD: 0.075)	0.534 (SD: 0.075)	0.562 (SD: 0.074)	0.408 (SD: 0.083)
10	0.582 (SD: 0.054)	0.554 (SD: 0.059)	0.568 (SD: 0.057)	0.423 (SD: 0.063)
50	0.595 (SD: 0.049)	**0.559** (SD: 0.055)	**0.576** (SD: 0.052)	**0.423** (SD: 0.068)
100	**0.602** (SD: 0.056)	0.547 (SD: 0.058)	0.573 (SD: 0.057)	0.414 (SD: 0.068)

Table 4.4: The effect of the perplexity threshold in the Morfessor FlatCat algorithm on precision, recall, f1-score and accuracy.

sired number of morph types with the data in this experiment at 12419. For translation however, this is likely too large of an amount, so experiments are performed with alternative number of morph types. We can see that similar results to the model without specification in terms of F1-score and accuracy is achieved by the model with 4000 morph types with a F1-score of `0.584` and an accuracy `0.445`.

Unfortunately, the model with 4000 morph types was created after extrinsic testing was already performed, so the model without any specification was used for this instead of the one with 4000 morph types.

4.3.2 Morfessor FlatCat

FlatCat (Grönroos et al., 2014) can be used as an extension of Morfessor, and the FlatCat models in this research are extensions of a semi-supervised Morfessor model. FlatCat again uses a hidden Markov model but it now also attaches categories to its segments. It uses a flat lexicon, meaning that the segments are encoded as their strings and that each letter is encoded by a certain amount of bits. This means that the longer the segment, the more bits needed to encode it and therefore the more expensive. This then also gives away the underlying mechanism of the program, namely a balancing of the cost whether a smaller segment is worth having its own segmentation, which is the case when it occurs often in the data.

Like with Morfessor Categories-MAP, FlatCat can be tuned by modifying the perplexity threshold. Likewise, the perplexity threshold of 50 is the best for both the F1-score (`0.576`) and accuracy (`0.423`), see Table 4.4. This model is therefore picked for extrinsic evaluation as well.

4.3.3 Linguistically-Motivated Vocabulary Reduction

Linguistically-Motivated Vocabulary Reduction (LMVR) (Ataman et al., 2017) can reduce the vocabulary size to a desired amount while keeping in mind the linguistic properties of the segmentation. It is desirable to have as few infrequent morphs as possible, and by considering the categories that all segments have been placed in, LMVR prevents words from being split up at random positions when a frequently occurring string is encountered. The big difference between LMVR and other vocabulary reduction techniques is therefore that the linguistic properties are kept consistently, meaning that a morph which has been tagged as a 'STEM' will not be changed in such a way that it is no longer a 'STEM'.

LMVR itself uses unsupervised learning, but it should be used on top of a program that provides categories to the segments. As in its original paper, the LMVR models in these experiments will be built upon semi-supervised FlatCat models.

The results of the experiments are shown in Ta-

Perplexity	Precision	Recall	F1-score	Accuracy
1	**0.602** (SD: 0.062)	0.541 (SD: 0.072)	**0.570** (SD: 0.068)	0.411 (SD: 0.085)
10	0.567 (SD: 0.059)	0.538 (SD: 0.059)	0.552 (SD: 0.058)	**0.416** (SD: 0.057)
50	0.578 (SD: 0.053)	**0.548** (SD: 0.059)	0.563 (SD: 0.055)	0.411 (SD: 0.072)
100	0.586 (SD: 0.055)	0.534 (SD: 0.053)	0.559 (SD: 0.053)	0.398 (SD: 0.063)

Table 4.5: The effect of the perplexity threshold in the Linguistically Motivated Vocabulary Reduction algorithm on precision, recall, f1-score and accuracy.

Supervision	Precision	Recall	F1-score	Accuracy
fully-supervised	0.638 (SD: 0.050)	**0.629** (SD: 0.044)	**0.633** (SD: 0.047)	0.542 (SD: 0.056)
semi-supervised	**0.640** (SD: 0.061)	0.623 (SD: 0.056)	0.631 (SD: 0.058)	**0.553** (SD: 0.059)

Table 4.6: The performance of CRFs on precision, recall, f1-score and accuracy.

ble 4.5. The top scores are divided over perplexity thresholds of 1, 10 and 50, where the threshold of 1 has the highest precision and F1-score with `0.602` and `0.570` respectively. A perplexity of 10 has the highest accuracy with a score of `0.416`, and the 50-threshold has the highest recall with `0.548`.

After consideration of these different models, this research chose to use the model with a perplexity threshold of 50 for extrinsic testing. Mainly because the difference between the three best performing models was very minimal, and for Morfessor Categories-MAP and Morfessor FlatCat the perplexity threshold of 50 was also the best option. This choice would likely lead to the most consistency between the models and therefore allow for a better comparison between technique-specific differences.

4.3.4 Conditional Random Fields

Conditional Random Fields (CRFs) are discriminative models for segmenting and labeling sequential data originally developed by Lafferty et al. (2001). The Morfessor family is often based upon generative HMMs, and the large difference compared to CRFs is that HMMs assume independence and use directed graphs, whereas CRFs define conditional probability using observed sequences and the use of undirected graphs.

For natural language processing, linear-chain CRFs are often used. The difference between these and general CRFs is that for linear-chains dependencies are only imposed on the previous element instead of on all. Using the implementation on linear-chain CRFs from Ruokolainen et al. (2013), both fully and a semi-supervised models were trained. The fully-supervised models are solely based on CRFs, whereas the semi-supervised methods are based on the classic letter successor variety (LSV) scores, originally presented by Harris (1955). Using the idea that the variability of the sequence of letters should be low within morphs and high at the boundaries, LSV can extract likely morph boundaries from the unannotated data.

Both a semi-supervised and a fully-supervised CRF model are trained. Looking at the segmentations, it is hard to specify the exact difference between the types of supervision. At points, the semi-supervised model has more segments, at other points the fully-supervised model has more segments. The place of segmentation also differs, see (14). Here, Harris' LSV classified the sequence -*pp*- to more likely belong to two different morphs rather than the same. Worth mentioning is that the way that the semi-supervised segmenter split up the word can also be correct, as *allappat*'s stem is *allat*- and the *p* is attached to the second morph because of the design choices explained in Section 3.2. In other semi-supervised segmentations, such as *sini-ssa-pput* the *-pp-* sequence is kept together, but this is likely because the *-pput* morph often occurs in the annotated dataset.

(14) Semi: `*allap pat`
Fully: `alla ppat`

In Table 4.6 the results of both models are illustrated. Again, the results are not unanimous, as the fully-supervised outperforms the semi-supervised on recall (`0.629`) and F1-score (`0.633`) but it is the

Method	Specification	Precision	Recall	F1-score	Accuracy
CRF	semi-supervised	**0.640**	**0.623**	**0.631**	**0.553**
Morfessor 2.0	-	0.597	0.566	0.581	0.449
FlatCat	p=50	0.595	0.559	0.576	0.423
LMVR	p=50	0.578	0.548	0.563	0.411
GroenOrd	-	0.400	0.439	0.419	0.282
Morfessor-CatMAP	p=50	0.428	0.409	0.418	0.233
BPE-simple	20k merges	0.191	0.205	0.198	0.081
BPE-simple	4k merges	0.151	0.208	0.175	0.042
None	-	0.069	0.069	0.069	0.069

Table 4.7: The performance of all methods on precision, recall, f1-score and accuracy.

other way around for precision (0.640) and accuracy (0.553). The results between the models vary minimally on all aspects.

For extrinsic testing, the semi-supervised model is chosen because its ability to generalize on new data should in theory be bigger. By considering the many unannotated words, the author hypothesizes that the semi-supervised model has a better overview of the language and is therefore better at parsing unseen words.

4.4 Completely-supervised segmenters

Completely-supervised segmenters only use annotated data. As it can be difficult to merge patterns from both annotated and unannotated data, some models better fit in the completely-supervised format. Using this method, noise from the often way larger unannotated dataset is left out, but there is a higher chance that the model will be overfit to the annotated data. Problematic for completely-supervised segmenters is however that a larger chunk of annotated data is needed, which is often scarce for low-resource polysynthetic languages.

4.4.1 Transformer

Using the implementation from Junczys-Dowmunt et al. (2018) based on the paper by Vaswani et al. (2017), all and only the annotated data was used to train a Transformer model. This technique is solely based on attention mechanisms instead of (also) using recurrence and convolutions. With the annotated data, it is able to figure out global dependencies in the input and transform these to the output. Therefore, this model type can be used for a wide range of natural language processing tasks such as NMT but also morphological segmentation.

The Transformer model showed great promise for the closely related language Inuktitut (Roest, unpublished MsC thesis, 2020), but unfortunately it appears that there is not enough annotated data for Greenlandic. Instead of segmenting the words for testing, it too often occurred that completely new spurious strings were generated. This caused some words to be segmented in a way that made them no longer recognizable, which is a known problem with character-based methods (Lee et al., 2016). The results of the Transformer models are therefore omitted from this report. When the annotated database is extended however, the Transformer models might be promising to revisit.

5 Extrinsic Experiments

All systems in Table 4.7 except for Morfessor Categories-MAP and GroenOrd will be used to create a KL→EN NMT system. GroenOrd could not process the large amount of data and Morfessor Categories-MAP was not included in the extrinsic evaluation as it would most likely not have an advantage over the other models in the Morfessor-family that were included.

The NMT systems are created with Marian 1.7.6 (Junczys-Dowmunt et al., 2018) using the Transformer model type (Vaswani et al., 2017). The models have an embedding dimension of 512 and use encoder and decoder layers of six.

All NMT models are trained with exactly the same specifications and the exact same dataset for training, testing and validation, where only the

Segmentation Model	BLEU
None	2.05
BPE-4k	1.60
FlatCat	1.53
BPE-20k	1.44
LMVR	1.11
CRF	1.01
Morfessor	0.95

Table 5.1: The performance of all methods in the extrinsic evaluation on BLEU score.

Greenlandic data is segmented differently by the various segmenters. Because the semi-supervised segmenters were first trained using tenfold cross-validation, a new model is trained with all available data using the exact conditions that were best for tenfold cross-validation.

For training, 70k sentences that were crawled and aligned using Bitextor were used and 2k sentences from manually crawled Greenlandic magazines.[10] The Bitextor data was very noisy, so the magazine data was over-sampled ten times to increase its weight. For testing and validation, respectively 500 and 200 separate lines from the magazines were used.

The results of the testing can be found in Table 5.1. As can be seen, the BLEU scores are disappointing, with the system without morphological segmentation having the highest score of 2.05 BLEU. The most likely reason behind this is that the translating of names and numbers worked best without any segmentation, whereas they got mixed up in the segmentation programs. Although it was not the aim of this research to provide a workable NMT system for KL→EN, the results of the extrinsic evaluation seem too poor to draw any significant conclusions from. Interesting to note however, is that the two intrinsically best performing programs perform poorly. I first suspected this could be caused by them having a too large vocabulary to fit the linguistic structures, but this does not seem to be the full problem. Considering the magazine data used for training, Morfessor 2.0 and CRF have 5448 and 6573 unique tokens respectively, whereas the better performing FlatCat has 7429 unique tokens.

There are many possible explanations for the poor BLEU scores. The most obvious is the fact that the dataset is not very large, and that the largest part of the data comes from Bitextor and has a lot of noisy lines and poor translations. On top of that, the magazine data dates back to 1999 whereas the Bitextor data is more recent, possibly leading to a slightly different writing style as well. The third factor is that the data extracted from the magazines was Greenlandic and Danish instead of Greenlandic and English. The Danish text was therefore translated into English, which caused the highest-quality dataset to still have synthetic data.

Another issue is that not all data was segmented evenly. The words that were in all capital letters were not segmented well by the models in the Morfessor family and by the CRF. These models, apart from Morfessor 2.0, did not segment these words at all. Morfessor 2.0 performed opposite, as it segmented each letter individually. This is due to the fact that the filtering for the training data in the models was much more strict than for segmenting the data to be translated.

It should also be noted that it is common practice in morphological segmentation for NMT to combine several morphological segmentation techniques. This allows for the exploitation of the benefits of various techniques while masking their weaknesses with other techniques. A prime example of this could be to first let a rule-based segmenter segment the words and applying BPE afterwards to segment the words that could not be segmented by the rule-based segmenter due to the fact that the stem was for example not available in the dictionary. To compare the individual morphological segmentation techniques however this could not be done, further explaining the low BLEU scores.

Lastly, it might have been a possibility to use the previously mentioned idea of Toral et al. (2019) to also use the data from the very closely related language Inuktitut to increase its pool of bilingually aligned sentences. This was not implemented however because this would also provide for noisy results when considering specific Greenlandic parsing.

6 Conclusions

To conclude, low-resource polysynthetic languages pose many challenges for morphological segmen-

[10] Atuagagdliutit, year 1999, issues 1,2,3,5,6&7

tation and NMT. Little research has been done into most of these languages, and the research that is done is limited by the absence of (annotated) databases. The author has manually crafted an annotated dataset for Greenlandic containing 640 unique segmented words. A database of words without annotation was created from a wikidump, a dictionary and Greenlandic websites and contains 2.9M words of which 122k are unique.

The intrinsic experiments (see Table 4.7) showed that Conditional Random Field greatly outperformed the semi-supervised segmenters in the Morfessor-family, which in turn outperformed the systems without any supervision. This is in line with expectations however because the unsupervised techniques have a disadvantage in comparison to the supervised programs because they cannot know possible design choices. Regardless, when aiming for linguistic correctness in morphological segmentation, having annotated data is very helpful.

For NMT, linguistically accurate morphological segmentations does not necessarily lead to more accurate translations. Byte Pair Encoding is a prime example of this, which was not specifically designed to extract morphs but does often increase the BLEU scores in translation (Sennrich et al., 2015). From all segmentation systems apart from no segmentation, BPE also worked best in the extrinsic evaluation (see Table 5.1), but due to suboptimal circumstances, it is highly questionable how trustworthy these results are.

Further development of mostly the annotated databases would be the next step in creating better morphological segmentation systems for low-resource languages. As seen in the intrinsic results, even a small annotated database already greatly improves the performance. For NMT systems, a reliable bilingual corpus is desired for Greenlandic and other low-resource polysynthetic languages before further investigation into the effects of segmentation on NMT can be performed.

Acknowledgements

I would like to thank my supervisor dr. J.K. Spenader for your all-round guidance. My sincere thanks also goes out to dr. A. Toral Ruiz and Christian Roest for the inspiration and the help with various segmentation programs. I also want to thank Christian Roest for creating NMT systems and his ideas in the external evaluation. Lastly, I am grateful to Kevin Kelly, especially for all the aligned parallel data.

References

Ataman, D., Negri, M., Turchi, M., & Federico, M. (2017). Linguistically motivated vocabulary reduction for neural machine translation from turkish to english. *The Prague Bulletin of Mathematical Linguistics*, *108*(1), 331–342.

Creutz, M., & Lagus, K. (2005a). Inducing the morphological lexicon of a natural language from unannotated text, In *Proceedings of the international and interdisciplinary conference on adaptive knowledge representation and reasoning (akrr'05).*

Creutz, M., & Lagus, K. (2005b). Unsupervised morpheme segmentation and morphology induction from text corpora using morfessor 1.0.

Ding, S., Renduchintala, A., & Duh, K. (2019). A call for prudent choice of subword merge operations. *CoRR*, *abs/1905.10453* arXiv 1905.10453. http://arxiv.org/abs/1905.10453

Eskander, R., Klavans, J., & Muresan, S. (2019). Unsupervised morphological segmentation for low-resource polysynthetic languages, In *Proceedings of the 16th workshop on computational research in phonetics, phonology, and morphology*, Florence, Italy, Association for Computational Linguistics. https://doi.org/10.18653/v1/W19-4222

Ethnologue. (2015). *Greenlandic language* (Vol. 18th ed). SIL International.

Fortescue, M. D. (1984). *West greenlandic*. Croom Helm London.

Gage, P. (1994). A new algorithm for data compression. *C Users Journal*, *12*(2), 23–38.

Grönroos, S.-A., Virpioja, S., Smit, P., & Kurimo, M. (2014). Morfessor flatcat: An hmm-based method for unsupervised and semi-supervised learning of morphology, In *Proceedings of coling 2014, the 25th interna-*

tional conference on computational linguistics: Technical papers.

Harris, Z. S. (1955). From phoneme to morpheme. *Language*, *31*(2), 190. https://doi.org/10.2307/411036

Junczys-Dowmunt, M., Grundkiewicz, R., Dwojak, T., Hoang, H., Heafield, K., Neckermann, T., Seide, F., Germann, U., Fikri Aji, A., Bogoychev, N., Martins, A. F. T., & Birch, A. (2018). Marian: Fast neural machine translation in C++, In *Proceedings of acl 2018, system demonstrations*, Melbourne, Australia, Association for Computational Linguistics. http://www.aclweb.org/anthology/P18-4020

Kann, K., Mager, M., Meza-Ruız, I. V., & Schütze, H. (2018). Fortification of neural morphological segmentation models for polysynthetic minimal-resource languages. *CoRR*, *abs/1804.06024*arXiv 1804.06024. http://arxiv.org/abs/1804.06024

Klavans, J. L. (2018). Computational challenges for polysynthetic languages, In *Proceedings of the workshop on computational modeling of polysynthetic languages*, Santa Fe, New Mexico, USA, Association for Computational Linguistics. https://www.aclweb.org/anthology/W18-4801

Lafferty, J. D., McCallum, A., & Pereira, F. C. N. (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data, In *Proceedings of the eighteenth international conference on machine learning*, San Francisco, CA, USA, Morgan Kaufmann Publishers Inc.

Lee, J., Cho, K., & Hofmann, T. (2016). Fully character-level neural machine translation without explicit segmentation. *CoRR*, *abs/1610.03017*arXiv 1610.03017. http://arxiv.org/abs/1610.03017

Mager, M., Mager, E., Medina-Urrea, A., Meza, I., & Kann, K. (2018). Lost in translation: Analysis of information loss during machine translation between polysynthetic and fusional languages. *arXiv preprint arXiv:1807.00286*.

Mahieu, M.-A., & Tersis, N. (2009). *Variations on polysynthesis: The eskaleut languages* (Vol. 86). John Benjamins Publishing.

O'Donnell, V., & Anderson, T. (2017). *The aboriginal languages of first nations people, métis and inuit.* Canadian Minister of Industry.

Rissanen, J. (1978). Modeling by shortest data description. *Automatica*, *14*(5), 465–471.

Roest, C. (2020). Comparison of morphological segmentation approaches for polysynthetic languages for neural machine translation, University of Groningen.

Ruokolainen, T., Kohonen, O., Virpioja, S., & Kurimo, M. (2013). Supervised morphological segmentation in a low-resource learning setting using conditional random fields, In *Proceedings of the seventeenth conference on computational natural language learning*, Sofia, Bulgaria, Association for Computational Linguistics. https://www.aclweb.org/anthology/W13-3504

Sennrich, R., Haddow, B., & Birch, A. (2015). Neural machine translation of rare words with subword units. *arXiv preprint arXiv:1508.07909*.

Toral, A., Edman, L., Yeshmagambetova, G., & Spenader, J. (2019). Neural machine translation for english–kazakh with morphological segmentation and synthetic data, In *Proceedings of the fourth conference on machine translation (volume 2: Shared task papers, day 1)*.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need, In *Advances in neural information processing systems*.

Virpioja, S., Smit, P., Grönroos, S.-A., Kurimo, M., Et al. (2013). Morfessor 2.0: Python implementation and extensions for morfessor baseline.

A Appendix: Data Sources

A.1 Raw monolingual data

A.1.1 Greenlandic-Danish Dictionaries

https://oqaasileriffik.gl/approved-words/

http://www.ilinniusiorfik.gl/oqaatsit/daka

A.1.2 Wikipedia dump

https://dumps.wikimedia.your.org/klwiki/20200301/

A.1.3 Crawled websites

www.nis.gl

www.sermitsiaq.ag

www.sermersooq.gl

www.peqqik.gl

www.naalakkersuisut.gl

www.knr.gl

www.qeqqata.gl

www.kak.gl

www.kujalleq.gl

www.banken.gl

www.banknordik.gl

www.qaasuitsup.gl

A.2 Annotated monolingual data

A.2.1 Greenlandic learning sites

https://learngreenlandic.tumblr.com/lessons

https://tulunnguaq.tumblr.com/