

# **CLARIN Annual Conference 2019**

## **PROCEEDINGS**

Edited by

Kiril Simov, Maria Eskevich

30 September – 2 October 2019  
Leipzig, Germany

Please cite as:

Proceedings of CLARIN Annual Conference 2019. Eds. K. Simov and M. Eskevich.  
Leipzig, Germany: CLARIN, 2019.

## A Parsing Pipeline for Icelandic Based on the IcePaHC Corpus

**Tinna Frímann Jökulsdóttir**

University of Iceland  
Reykjavík, Iceland  
tinnafrj@hi.is

**Anton Karl Ingason**

University of Iceland  
Reykjavík, Iceland  
antoni@hi.is

**Einar Freyr Sigurðsson**

The Árni Magnússon Institute for Icelandic Studies  
Reykjavík, Iceland  
einar.freyr.sigurdsson@arnastofnun.is

### Abstract

We describe a novel machine parsing pipeline that makes it straightforward to use the Berkeley parser to apply the annotation scheme of the IcePaHC corpus to any Icelandic plain text data. We crucially provide all the necessary scripts to convert the text into an appropriate input format for the Berkeley parser and clean up the output. The goal of this paper is thus not to dive into the theory of machine parsing but rather to provide convenient infrastructure that facilitates future work that requires the parsing of Icelandic text.

### 1 Introduction

Icelandic is a less-resourced language in the context of the CLARIN goals of fostering language resources and technology infrastructure; thus it is crucial to create further Icelandic resources that facilitate the development and use of Icelandic Language Technology for research as well as practical applications. Some efforts have been made in recent years to develop such resources and a notable example is the Icelandic Parsed Historical Corpus, IcePaHC (Wallenberg et al., 2011; Rögnvaldsson et al., 2012). The IcePaHC corpus contains one million running words of manually corrected phrase structure annotation. In this paper, we describe a novel parsing pipeline for Icelandic that makes crucial use of IcePaHC, thus making it straightforward for any future projects to take advantage of machine-annotation according to the IcePaHC annotation scheme. The pipeline is available on Github (<https://github.com/antonkarl/iceParsingPipeline>). This is ongoing work and we aim to package our solutions as CLARIN tools when they have reached a more mature state. To consider a concrete example from the corpus, the sentence below is taken from the 13th century manuscript called *Morkinskinna*.

- (1) *Par kom að þeim Danaher.*  
there came to them Danish army.  
'There, the Danish army came toward them.'

The corpus makes use of labeled bracketing similar to the Penn Treebank. The annotated version of the sentence is shown below.

```
( (IP-MAT (ADVP-LOC (ADV Par-par))
  (VBDI kom-koma)
  (PP (P að-að)
    (NP (PRO-D þeim-það)))
  (NP-SBJ (NPR-N Danaher-danaher))) (ID 1275.MORKIN.NAR-HIS, .522))
```

One of the main reasons for constructing a parsed corpus at all is the ability to train an automatic parser in order to get access to fast machine annotation of the same type. While there are various interesting deep technical problems involved in machine parsing, the practical challenges of setting up convenient infrastructure for parsing are sometimes overlooked when discussing, say, the theoretical side of training parsers. Even if a parsed corpus is freely available, a researcher in the humanities or social sciences may

This work is licensed under a Creative Commons Attribution 4.0 International Licence. Licence details: <http://creativecommons.org/licenses/by/4.0/>

not have access to the technical background that is required to train a parser and set up a parsing pipeline. The current project aims to ameliorate this situation by developing a pipeline that takes Icelandic in plain text format, converts it to an appropriate input format for the Berkeley parser, parses the data using a pre-trained model that we provide, and cleans up the output. This, for example, allows anyone who has learned to use a treebank search program to apply this knowledge to their own data.

The paper is organized as follows: In Section 2, we introduce IcePaHC and related tools. In Section 3, we describe our matrix clause boundary detection system, and in Section 4 we discuss our use of the Berkeley parser. Section 5 offers information about some post-processing steps and Section 6 concludes.

## 2 The IcePaHC corpus and related tools

The Icelandic Parsed Historical Corpus (IcePaHC) is a dual purpose project. It is designed to serve both as a language technology tool and a syntactic research tool. It contains about one million words, fairly evenly distributed throughout the written history of the Icelandic language, from the 12th century to modern times. IcePaHC is released under a LGPL-license and it can be freely downloaded from [http://www.linguist.is/icelandic\\_treebank/Download](http://www.linguist.is/icelandic_treebank/Download).

The annotation scheme, briefly demonstrated above, is based on the Penn Parsed Corpora of Historical English (Kroch and Taylor, 2000; Kroch et al., 2004). For most purposes, the English annotation guidelines are applied without modification to Icelandic and, in fact, the same search query can often be used for studying the same phenomenon in both languages. Some minor Icelandic-specific adjustments to the annotation scheme include a somewhat larger tagset and lemmatization; these reflect the morphological richness of Icelandic compared to English. The Icelandic guidelines also include a way to annotate non-nominative subjects, a well-known peculiarity of Icelandic syntax (Thráinsson, 2007).

The IcePaHC project has from the beginning emphasized a Free and Open Source (FOSS) policy. The development pipeline of the corpus itself consisted entirely of FOSS tools, including IceNLP for tagging and shallow parsing (Loftsson and Rögnvaldsson, 2007), Lemmald for lemmatization (Ingason, et al., 2008), CorpusSearch for structure building queries and quality checks (Randall, 2005), and Annotald for manual annotation (Beck et al., 2015). The project also gave rise to the Parsed Corpus Query Language (PaCQL), currently under development and available as a preview on [www.treebankstudio.org](http://www.treebankstudio.org). PaCQL has made it easier for scholars of the digital humanities to use IcePaHC. It should also be noted that another ongoing project aims to implement a conversion from IcePaHC annotation to Universal Dependencies, further expanding the technical ecosystem of the present development efforts.

## 3 Matrix clause boundary detection

A crucial step in parsing involves boundary detection of the segments that have a privileged status in the sense that they correspond to one tree in the annotation scheme. In the IcePaHC annotation scheme, this privileged unit is in most cases the matrix clause, meaning that we must have some means of detecting the boundaries of matrix clauses. In the original construction of the corpus, this step was carried out manually by research assistants, and the current project therefore needed to implement a software solution for this task. We divided the task into two subtasks, (i) punctuation-based sentence boundary detection and (ii) conjunction-based matrix clause boundary detection. For both steps, we configured a feature extractor for potential boundaries and used IcePaHC to train an implementation of the Averaged Perceptron classifier (Freund and Schapire, 1999) by Kyle Gorman (Gorman, 2019) to detect actual boundaries.

For the punctuation-based sentence boundary detection we used the Python package Detector Morse (Gorman, 2019) and the feature extractor that comes with this package. The machine-learning classification algorithm, the Averaged Perceptron, is loaded from the nlup Python package, also authored by Gorman. Detector Morse was originally developed for English but its design carries over straightforwardly to languages that use a similar alphabet and similar conventions for spelling and punctuation. As expected, training Detector Morse in its default configuration by using the sentence splits in IcePaHC as a gold standard gave good results without the need for any Icelandic-specific adjustments.

The output from Detector Morse is subsequently fed into a conjunction-based matrix clause boundary detection system. The design of our matrix clause splitter is based on Detector Morse, and it uses the

same implementation of the Averaged Perceptron, but in this case it is not possible to achieve good results without the ability to detect language-specific morphosyntactic properties of the context. Therefore, we developed our own feature extractor for this task.

The conjunctions *og* ‘and’, *en* ‘but’ and *eða* ‘or’ are considered potential matrix clause boundaries whenever they are attested. In each case, the classifier must determine if it is an actual matrix clause boundary, like *and* in (2) or a different use of the relevant conjunction, like *and* in (3).

(2) John walked to the store **and** Mary smiled when she saw him.

(3) John **and** Mary walked to the store.

The feature extractor considers two words preceding the conjunction and two words following the conjunction and the pattern it returns is sensitive to whether any of these words, in this relative position to the conjunction, is potentially a morphosyntactic indicator. The indicators in question are: comma, finite verb, non-finite verb, a word in the nominative case, a word that has any non-nominative (oblique) case value, i.e. accusative, dative or genitive case.

While this setup works relatively well, the focus of the current phase of our project is simply to get something working that can be used for practical applications, rather than optimizing individual steps. Our morphosyntactic indicators are chosen based on our experience of Icelandic syntax, but we are confident that the feature extraction can be improved and this will be done in future work along with proper evaluation. For now, the matrix clause splitter works well enough for most detected matrix clauses to be a suitable input to the Berkeley Parser as trained on IcePaHC.

#### 4 Training the Berkeley Parser

We chose the Berkeley Parser (Petrov et al., 2006) for our work because its split-merge algorithm is known to yield accurate results, it is relatively simple to use for data that are already in a labeled bracketing file format, and there exists a version of it that runs fast on massively multi-core GPU cards, i.e., its Puck implementation (Canny et al., 2013; Hall et al., 2014). The ability to run the parser on GPU’s is particularly important for large data sets and High-Performance Computing Clusters (HPCC) that emphasize GPU-computing are increasingly becoming a part of the technology infrastructure that research universities and organizations are continually expanding. We use the Berkeley Parser for Part-of-Speech tagging as well as for parsing phrase structure.

The training data we used was the full IcePaHC corpus. Although the parser assumes labeled bracketing, we did need to make a few minor adjustments to the file format to get the training phase to run smoothly, and this is important in the context of the present paper because it exemplifies how practice is more complicated than theory for a language technology task like machine parsing. In theory, having a parsed corpus and a freely available implementation of some trainable parser is enough to yield machine parsing for the annotation scheme in question, but in practice it takes quite a bit of technical work to set up the process. For researchers in the humanities and social sciences who may just want access to the annotation without having to put unreasonable effort into getting the system to work, having access to a pre-configured parsing pipeline can make an important difference in getting the results they want.

#### 5 Post-processing and cleanup

Our pipeline also includes scripts that take the output of the Berkeley parser and make some minor adjustments to it. While these steps do not change the information that is included in the output from the parser, they make its format more similar to what scholars who study historical syntax are used to.

The community of researchers who study historical syntax using treebanks and quantitative methods has by now become very familiar with the files that are used for the raw data of the Penn Parsed Corpora of Historical English because these corpora are frequently used and other corpora that have been developed for the same group of users have adopted this format. This format is, of course, machine-readable, but as the conventions used for indenting etc. have become somewhat standard for this subset of treebanks, it is, for an experienced user, also conveniently human-readable. While such formatting

nuances might be considered an unimportant detail, we believe that it will make our parsing pipeline more pleasing to use for the users that are most likely to make it a part of their workflow.

## 6 Summary and future work

We have described our parsing pipeline for Icelandic that takes plain text input and yields output that is machine-annotated according to the annotation scheme of the IcePaHC treebank. This includes pre-processing, parsing using a pre-trained model of the Berkeley-parser, and formatting and cleanup of the output. We make these steps easily executable by any future project that may benefit from parsing Icelandic. While our parsing pipeline is already available and useful, much remains to be done. The configuration that we use for individual steps such as training the matrix clause splitter and the parser will be improved in future work in order to yield even better results and proper evaluation will be an essential ingredient of any iterative improvements. At this point, we focus on a pipeline that can be used for further development, hence evaluation of different parser configurations remains a future task. With our pipeline in place, we have also started work on a Machine-parsed IcePaHC (MicePaHC), a corpus that does not have the manual corrections of IcePaHC, but can grow much faster in size because all of the annotation is carried out by computers. We aim to release the first version of MicePaHC in the near future, both in terms of freely available raw data and as a search option on treebankstudio.org.

## References

- Beck, Jana, Aaron Eay, and Anton Karl Ingason. 2015. Annotald. Version 1.3.7.
- Canny, John, David Hall, and Dan Klein. 2013. A multi-Teraflop Constituency Parser using GPUs. *Proceedings of Empirical Methods in Natural Language Processing*, pp. 1898–1907.
- Freund, Yoav, and Robert E. Schapire. 1999. Large margin classification using the perceptron algorithm. *Machine Learning* 37.3 (1999): 277–296.
- Gorman, Kyle. 2019. Detector Morse. A Python Package. Version 0.4.1.
- Hall, David, Taylor Berg-Kirkpatrick, and Dan Klein. 2014. Sparser, Better, Faster GPU Parsing. *Proceedings of ACL 2014*, pp. 208–217.
- Ingason, Anton Karl, Sigrún Helgadóttir, Hrafn Loftsson, and Eiríkur Rögnvaldsson. 2008. A mixed method lemmatization algorithm using a hierarchy of linguistic identities (HOLI). *Proceedings of GoTAL*, pp. 205–216. Springer, Berlin, Heidelberg.
- Kroch, Anthony, Beatrice Santorini, and Lauren Delfs. 2004. Penn-Helsinki Parsed Corpus of Early Modern English (PPCEME). Department of Linguistics, University of Pennsylvania. CD-ROM, first edition, release 3 (<http://www.ling.upenn.edu/ppche-release-2016/PPCEME-RELEASE-3>).
- Kroch, Anthony, and Ann Taylor. 2000. The Penn-Helsinki Parsed Corpus of Middle English (PPCME2). Department of Linguistics, University of Pennsylvania. CD-ROM, second edition, release 4 (<http://www.ling.upenn.edu/ppche-release-2016/PPCME2-RELEASE-4>).
- Loftsson, Hrafn, and Eiríkur Rögnvaldsson. 2007. IceNLP: A natural language processing toolkit for Icelandic. *Proceedings of Eighth Annual Conference of the International Speech Communication Association*.
- Petrov, Slav, Leon Barrett, Romain Thibaux and Dan Klein. 2006. Learning Accurate, Compact, and Interpretable Tree Annotation. *Proceedings of COLING-ACL 2006*, pp. 433–440.
- Randall, Beth. 2005. CorpusSearch 2. User's manual.
- Rögnvaldsson, Eiríkur, Anton Karl Ingason, Einar Freyr Sigurðsson, and Joel Wallenberg. 2012. The Icelandic Parsed Historical Corpus (IcePaHC). *Proceedings of LREC'12*, pp. 1977–1984.
- Thráinsson, Höskuldur. 2007. *The Syntax of Icelandic*. Cambridge University Press, Cambridge.
- Wallenberg, Joel C., Anton Karl Ingason, Einar Freyr Sigurðsson, and Eiríkur Rögnvaldsson. 2011. The Icelandic Parsed Historical Corpus, version 0.9. 1 million words.