Human Language Technologies – The Baltic Perspective
A. Utka et al. (Eds.)

doi:10.3233/FAIA200595

A Study in Estonian Pronominal Coreference Resolution

Eduard BARBU [a,1], Kadri MUISCHNEK [a,b] and Linda FREIENTHAL [b]
[a] *Institute of Computer Science, University of Tartu, Estonia*
[b] *Institute of Estonian and General Linguistics, University of Tartu, Estonia*

Abstract. The first study for Estonian pronominal coreference resolution using machine learning is presented. Appropriate machine learning algorithms and techniques for balancing the data are tested on a human-annotated corpus. The results are encouraging, showing an F-score comparable with the results obtained for English before the advent of deep neural networks.

Keywords. Pronominal coreference resolution, machine learning, low resource language

1. Introduction

The pronominal coreference resolution [1] is the task of automatically finding the correct reference for a pronoun. The task is hard because the syntactic and semantic information is not enough to solve it. It constitutes the backbone of the Winograd Schema Challenge [2], a machine test intelligence that improves on the Turing Test. Given a text, for example: ”The trophy would not fit in the brown suitcase because it was too big.” a machine should answer a question like: What was too big: the trophy or the suitcase? The answer to this question amounts to solving the coreference between the pronoun ”*it*” and the noun phrase. This example shows that a pronominal coreference resolution system needs world knowledge: it needs to know that the suitcases are containers and therefore the pronoun *it* should be resolved to the noun phrase ”*the suitcase*”.

In this paper, the first machine learning study in Estonian automatic pronominal coreference resolution is presented. Appropriate machine learning algorithms and techniques intended to solve the imbalanced data problems are tested for a manually annotated pronominal coreference corpus.

Automatically resolving the coreference in Estonian is more complicated than in English. Unlike English, Estonian has no gender. Gender is a crucial feature that helps pronominal coreference resolution systems discriminate against the coreference pairs with gender agreement. For example, in English (John, **he**) could be a coreference pair, but (Ana, **he**) is not. The amount of annotated data for Estonian is much less than in English. Finally, the external knowledge than can be incorporated in a coreference system, shown to increase the performance significantly [3], is limited. The Estonian language

[1] Corresponding Author: Eduard Barbu; E-mail: eduard.barbu@ut.ee.

can only rely on the Estonian WordNet while English has a vast pool of ontologies and lexical resources.

The paper has the following structure. The next section places this study in the context of global research about automatic coreference resolution. Section 3 describes the manually annotated coreference corpus. Section 4 shows the features of the system and the machine learning algorithms tested. Section 5 presents and discusses the results. The paper ends with the conclusions.

2. Related Work

Nowadays, the best automatic pronominal coreference resolution systems are based on deep neural networks and incorporated world knowledge. Clark and Manning [4] proposed a coreference resolution algorithm that uses features defined over clusters of mentions. A two-layer model for pronoun coreference resolution leveraging the context and external knowledge is presented in a state of the art pronominal coreference system for English [3]. In particular, the authors use English Wikipedia to learn the distribution of the selectional preference of the verbs appearing in their corpus. For other languages than English a relevant study is one for Polish coreference resolution [5] that explores several deep learning architectures. For German [6], Tuggener proposes an incremental discourse processing algorithm that can address issues caused by the underspecification of mentions.

As for Baltic languages, Žitkus et al. [7] present a rule-based method for anaphora resolution in Lithuanian in the context of processing e-health records. In the same paper they provide a thorough overview of coreference/anaphora resolution in Balto-Slavic languages. Znotiņš and Paikens [8] developed a rule-based coreference resolution system for Latvian. It relies on morpho-syntactic information as well as Named Entities identification.

In Estonian, the pronominal coreference resolution was studied by Mutso [9], who adapted Mitkov's knowledge low rule-based approach [10], and Puolakainen [11] who employed Constraint Grammar [12] rules for solving the referents to pronouns. Unfortunately, neither of these experiments can be reproduced.

3. Corpus

The annotated coreference corpus used in the experiments is called EstAnaphora[2]. It contains texts from Estonian newspapers, magazines and a scientific journal spanning the years 1998 to 2007. The size of the corpus is ca 253,000 words. The following pronouns are annotated for coreference information:

- personal pronouns;
- demonstrative pronoun *see* 'it, that';
- relative pronouns *kes* 'who' and *mis* 'what'.

[2] https://github.com/EstSyntax/EstAnaphora

Each corpus file was annotated manually by two annotators, using the brat annotation tool[3]. A judge, helped by two linguists for the problematic cases, compared the annotations and provided the definitive version. For our experiments, the corpus annotations were converted to the CONLL-U format[4] where the coreference information is presented on the 10th (miscellaneous) field.

EstAnaphora contains $7,250$ nominal coreference pairs, that is pairs which contain one of the above mentioned pronouns and a referent, which is a common noun, a proper noun or another pronoun, with the following distribution:

- $4,268$ pairs in which a pronoun refers to a common noun;
- $2,721$ pairs in which the pronoun refers to a proper noun;
- 261 pairs in which the pronoun refers to another pronoun.

Moreover, in $6,577$ cases, the pronoun refers to a single referent, and in 289 cases, the pronoun refers to more than one antecedent. A case when the personal pronoun refers to more than one referent is illustrated in the following example: "**John** and **Mary** claimed that **they** are not guilty".

Figure 1 shows the percent of the pronoun referents found in a context window around the sentence containing the pronoun. Approximately 90 percent of the referent occurrences are found in a two sentences window to the left of the pronoun sentence.

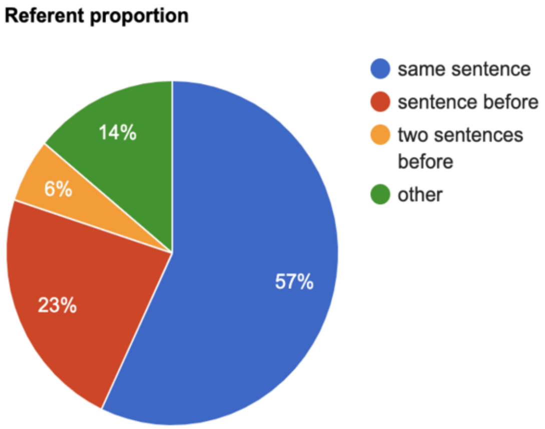

Figure 1. The percent of the referents found in the sentences in the immediate context of the sentence containing the pronoun

These figures help to set the appropriate context window for searching for candidate referents. There is a trade-off between the context window length and the algorithm performance: the wider the context window is, the more candidate pairs are generated and the less accurate the algorithm is. Vice versa, if the context is too narrow, several correct referents will be missed.

[3] https://brat.nlplab.org

[4] https://universaldependencies.org/format.html

4. Machine Learning

The coreference resolution uses the mention-pair model (for other models employed in coreference resolution see for example [13]), which is formulated as a binary classification problem. A machine learning algorithm trained on negative and positive coreference pairs learns to classify unseen coreference pairs. The coreference resolution, like fraud detection, is an imbalanced classification problem meaning that the number of positive samples is much lower than the number of negative samples. In our corpus, the proportion of positive to negative examples is roughly 1 to 24. When applied to a test set that has the same proportion of positive to negative classes, a classifier might yield an optimistic accuracy estimate. The classifier might assign every single test case to the majority class, thereby achieving an accuracy equal to the proportion of test cases belonging to the majority class.

To mitigate this known problem, techniques for dealing with the imbalanced data have been explored. As the positive class appears infrequently, extra weight has been added to it. Moreover, the standard techniques for balancing the data set (the negative class has been undersampled, and the positive class oversampled) have also been tested. The well known SMOTE (Synthetic Minority Over-sampling Technique) algorithm generates new training data for the positive class considering the k-nearest neighbors of the positive example [14]. An advanced balancing technique called Adaptive Synthetic Sampling Method for Imbalanced Data, known as ADASYN, was also tested [15]. ADASYN weighs the positive class examples based on the level of difficulty in learning. Hence more synthetic data is generated for harder to learn positive class examples. The last technique tried is One-Class SVM [16]. This algorithm is trained only on negative examples to learn the boundaries of the negative points. Any points that lie outside the boundaries are considered outliers (e.g., they correspond to the positive data examples).

4.1. Features

Four kinds of features are computed for the generated coreference pairs: distance features, morphological, syntactic, and semantic features. The distance features encode the distance between the pronoun and the referent, as well as the position of the referent in the sentence. Examples of distance features are :

- **Distance pronoun-referent**. The feature encodes the distance between the sentence of the pronoun and the sentence of the referent. If the pronoun and the referent are in the same sentence, the distance is 0.
- **Distance in nouns**. The feature counts the number of nouns separating the pronoun and the referent.
- **Referent position** gives the position of the referent in the sentence. The position can be one of the values: beginning, middle, or end.

The morphological features encode the morphological information found in a context window around the referent and the pronoun. Examples of morphological features are:

- **POS referent/pronoun**. These features encode the part of speech (POS) tag of the referent and the pronoun.
- **POS before referent/pronoun**. Theses features give the POS tag of the word found 1, 2 or 3 positions before the referent or the pronoun.

- **POS after referent/pronoun**. Theses features give the POS tag of the word found 1, 2 or 3 positions after the referent or the pronoun.

The syntactic features encode syntactic information about the coreference pairs. Examples of syntactic features are:

- **Syntactic function referent/pronoun**. The features encode the syntactic functions of the referent and the pronoun.
- **POS head referent/pronoun**. The features encode the POS tag of the syntactic heads of the referent and the pronoun.

The semantic features encode the cosine similarity scores between the embeddings corresponding to the pronouns and referents. The embeddings are trained with word2vec on the Estonian Reference Corpus [17]. For this study, 29 features have been implemented.

4.2. Algorithms

The machine learning algorithms were selected based on three criteria: resistance to data unbalancing, boundary type (linearly separable or not), and performance.

1. **Decision trees** (DT). The advantage of the decision tree algorithms is that humans can interpret their output. It is also known that they are resistant to imbalanced data because they have an inductive bias towards axis-aligned bounding boxes.
2. **Logistic regression** (LR). The Logistic Regression works particularly well when the features are linearly separable. The classifier is robust to noise, avoids overfitting, and its output can be interpreted as probability scores.
3. **K-Nearest Neighbors** (knn). This algorithm classifies a new instance based on the distance it has to k instances in the training set. The prediction output is the label that classifies the majority. Because it is a non-parametric method, it gives good results in classification problems where the decision boundary is irregular.
4. **XGBoost** is a widely used, high-performance machine learning algorithm from the tree boosting family [3]. It has won numerous Kaggle competitions, thus showing a state of the art performance in many tasks.

4.3. Experiment

The automatic coreference resolution experiment follows three steps.

1. **Candidate coreference-pair generation**. The coreference pairs between nominals and pronouns are generated. The generation algorithm allows the specification of several parameters, like the window context for each pronoun. In order to choose the best configuration, runs with different parameter values have been performed.
2. **Training**. The coreference pairs labeled in the corpus are assigned to the positive class. The rest of the coreference pairs generated based on the parameters above are assigned to the negative class. The features are computed for the training set, the machine learning algorithms are trained, and the trained model is stored.
3. **Testing**. The test set coreference pairs and their features are generated. The trained models are loaded and the test coreference pairs are assigned to the positive and negative classes by the machine learning algorithms.

Table 1. The results of the machine learning algorithms

Classifier	Parameters	Balanced	F1 score
DT	default	no	0.49
XGBoost	**default**	**no**	**0.60**
knn	neighbors =3	no	0.39

LR	solver='lbfgs',
	max_iter=4000
		no	0.40

LR 1	solver='lbfgs',
	max_iter=4000,
	class_weight={0: 1, 1: 5}
		no	0.46

LR 2		undersampling threshold 0.5	0.37
LR 3		ADASYN	0.31
XGBoost 1	**class weight={0: 1, 1: 5}**	**no**	**0.60**
XGBoost 2		undersampling threshold 0.5	0.44
XGBoost 3		ADASYN	0.51
DC		no	0.04
BC		no	0.32

5. Results and Discussions

The experiment is performed with the scikit-learn toolkit. There are two baselines. The first baseline is a weak one (abbreviated DC in the table), implemented by a dummy classifier that generates predictions according to the positive and negative class distribution in the training set. The second baseline (abbreviated BC) is a competitive baseline that resolves the mention to the closest pronoun.

Though all classifiers have been run in multiple configurations, only the best results are reported. The One Class SVM, for example, had a very low performance and we have excluded it from the analysis.

The results reported in Table 1 are for 4-fold stratified cross-validation on the annotated corpus. The parameters column gives the value of the hyperparameters for the classifiers. The Balanced column stipulates if the training set is balanced or not. There are three configurations of the Logistic Regression and XGBoost algorithms, each one with a different technique that treats the imbalanced data. LR 1 and XGBoost 1 is a configuration where the positive class receives five times more weight than the negative class. LR 2 and XGBoost 2 is a configuration where the dataset is balanced by random undersampling. LR 3 and XGBoost 3 is a configuration where the dataset is balanced by ADASYN.

The best results are obtained by the XGBoost algorithm in two configurations marked in bold in Table 1. The techniques to deal with imbalance data seem to be detrimental to the algorithm performance. However, more experiments should be performed to reach a definitive conclusion.

The Logistic Regression performance increases 6 points when we weigh the positive class 5 times more than the negative class, but undersampling imbalanced technique and ADASYN lower the algorithm performance.

It is known that the decision trees perform relatively well with imbalanced data, so the score attained slightly behind XGBoost 2 result is no surprise.

As expected, the weak baseline performs a little better than chance. The BC baseline, a competitive baseline, is soundly beaten on the test set by all machine learning algorithms, including the nonparametric lazy learning knn.

6. Conclusion and Future Work

In this paper, the first machine learning coreference study for the Estonian language has been presented. The results obtained are encouraging though they are not yet comparable to the state of the art results for the English language. The best results obtained by the best classifier XGBoost are in the same range as the results obtained for the English language by the knowledge poor systems before the advent of deep neural network revolution.

In the introduction of the paper, we have given three reasons why this might be the case. We believe that annotating more data based on the error analysis will substantially improve the results. Moreover, new, linguistically motivated features will be devised in the next version.

More importantly, we will explore advanced algorithms based on deep neural networks, which are state-of-the-art English language. There are some preliminary experiments performed by one of the authors of this paper [18]. However, for the features calculated by the system, it seems that the neural network architecture tested is not better than the XGBoost algorithm.

The incorporation of semantic information from the Estonian WordNet might improve the performance of the coreference system. However, the fact that Estonian is a low resource language and lacks the grammatical category of gender are severe limitations placed on any automatic coreference resolution system. The Estonian coreference system can be accessed from the following Github repository [5].

7. Acknowledgments

We would like to thank the annotators who have annotated the EstAnaphora coreference corpus.

This study was supported by the Estonian Language technology grant ”Estonian Universal Syntax: Resources and Applications” and by the Centre of Excellence in Estonian Studies (CEES, European Regional Development Fund).

References

[1] Hobbs JR. Resolving pronoun references. Lingua 44. 1978:311–338.

[2] Levesque HJ, Davis E, Morgenstern L. The Winograd Schema Challenge. In: Proceedings of the Thirteenth International Conference on Principles of Knowledge Representation and Reasoning. KR’12. Rome, Italy: AAAI Press; 2012. p. 552–561. Available from: https://cs.nyu.edu/faculty/davise/papers/WSKR2012.pdf.

[5] https://github.com/SoimulPatriei/EstonianCoreferenceSystem

[3] Zhang H, Song Y, Song Y. Incorporating Context and External Knowledge for Pronoun Coreference Resolution. In: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers). Minneapolis, Minnesota: Association for Computational Linguistics; 2019. p. 872–881. Available from: https://www.aclweb.org/anthology/N19-1093.

[4] Clark K, Manning CD. Improving Coreference Resolution by Learning Entity-Level Distributed Representations. In: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). Berlin, Germany: Association for Computational Linguistics; 2016. p. 643–653. Available from: https://www.aclweb.org/anthology/P16-1061.

[5] Nitoń B, Morawiecki P, Ogrodniczuk M. Deep Neural Networks for Coreference Resolution for Polish. In: Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018). Miyazaki, Japan: European Language Resources Association (ELRA); 2018. Available from: https://www.aclweb.org/anthology/L18-1060.

[6] Tuggener D. Incremental Coreference Resolution for German. University of Zurich, Faculty of Arts; 2016.

[7] Žitkus V, Butkienė R, Butleris R, Maskeliunas R, Damasevicius R, Woźniak M. Minimalistic Approach to Coreference Resolution in Lithuanian Medical Records. Computational and Mathematical Methods in Medicine. 2019 03;2019:1–14.

[8] Znotiņš A, Paikens P. Coreference Resolution for Latvian. In: Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14). Reykjavik, Iceland: European Language Resources Association (ELRA); 2014. p. 3209–3213. Available from: http://www.lrec-conf.org/proceedings/lrec2014/pdf/729$_paper.pdf$.

[9] Mutso P. Knowledge-poor anaphora Resolution System for Estonian. Tartu Ülikool; 2008.

[10] Mitkov R. Robust pronoun resolution with limited knowledge. In: Proceedings of the 18th International Conference on Computational Linguistics (COLING'98)/ACL'98 Conference; 1998. p. 869–875.

[11] Puolakainen T. Anaphora resolution experiment with CG rules. In: Proceedings of the Workshop on "Constraint Grammar - methods, tools and applications" at NODALIDA 2015, May 11-13, Vilnius, Lithuania; 2015. p. 35–38.

[12] Karlsson F, Voutilainen A, Heikkilä J, Anttila A. Constraint Grammar: A Language-Independent System for Parsing Unrestricted Text; 1995.

[13] Ng V. Machine Learning for Entity Coreference Resolution: A Retrospective Look at Two Decades of Research. In: Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence. AAAI'17. AAAI Press; 2017. p. 4877–4884.

[14] Chawla NV, Bowyer KW, Hall LO, Kegelmeyer WP. SMOTE: Synthetic Minority over-Sampling Technique. J Artif Int Res. 2002 Jun;16(1):321–357.

[15] He H, Bai Y, Garcia EA, Li S. ADASYN: Adaptive synthetic sampling approach for imbalanced learning. In: IN: IEEE INTERNATIONAL JOINT CONFERENCE ON NEURAL NETWORKS (IEEE WORLD CONGRESS ON COMPUTATIONAL INTELLIGENCE), IJCNN 2008; 2008. p. 1322–1328.

[16] Zhang R, Zhang S, Muthuraman S, Jiang J. One Class Support Vector Machine for Anomaly Detection in the Communication Network Performance Data. In: Proceedings of the 5th Conference on Applied Electromagnetics, Wireless and Optical Communications. ELECTROSCIENCE'07. Stevens Point, Wisconsin, USA: World Scientific and Engineering Academy and Society (WSEAS); 2007. p. 31–37.

[17] Kaalep HJ, Muischnek K, Uiboaed K, Veskis K. The Estonian Reference Corpus: Its Composition and Morphology-aware User Interface. In: Proceedings of the 2010 Conference on Human Language Technologies – The Baltic Perspective: Proceedings of the Fourth International Conference Baltic HLT 2010. Amsterdam, The Netherlands, The Netherlands: IOS Press; 2010. p. 143–146. Available from: http://dl.acm.org/citation.cfm?id=1860924.1860949.

[18] Freienthal L. Pronominaalsete viitesuhete automaatne lahendamine eesti keeles närvivõrkude abil. Tartu Ülikool; 2020.