Computational Linguistics and Intellectual Technologies:
Proceedings of the International Conference "Dialogue 2017"

Moscow, May 31—June 3, 2017

COREFERENCE RESOLUTION FOR RUSSIAN: THE IMPACT OF SEMANTIC FEATURES[1]

Toldova S. (toldova@yandex.ru)
National Research University Higher School of Economics, Moscow, Russia

Ionov M. (max.ionov@gmail.com)
Lomonosov Moscow State University, Moscow, Russia

This paper presents the results of our experiments on building a general coreference resolution system for Russian. The main aim of those experiments was to set a baseline for this task for Russian using the standard set of features developed and tested for coreference resolution systems created for other languages. We propose several baseline systems, both rule-based and ML-based. We show that adding some semantic information is crucial for the task and even the small amount of data can improve the overall result. We show that different types of semantic resources affect the performance differently and sometimes more does not imply better.

Key words: coreference resolution, semantic features, ontologies, mention-pair coreference resolution, Russian NLP

РАЗРЕШЕНИЕ КОРЕФЕРЕНТНЫХ СВЯЗЕЙ ДЛЯ РУССКОГО ЯЗЫКА: ВКЛАД СЕМАНТИЧЕСКИХ ПРИЗНАКОВ

Толдова С. (toldova@yandex.ru)
НИУ Высшая Школа Экономики, Москва, Россия

Ионов М. (max.ionov@gmail.com)
МГУ им. Ломоносова, Москва, Россия

[1] This research was supported by a grant from Russian Foundation for Basic Research Fund (15-07-09306).

1. Introduction

An important task for a number of high-level NLP applications, such as machine translation, summarization and storyline detection is the task of coreference resolution grouping the noun phrases that are the mentions of the same referent into one cluster.

All the noun phrases in one cluster form a coreference chain. A noun phrase that is a part of a coreference chain is called mention.

Even though there was a lot of research connected to the task of coreference resolution in the last 3 decades, there is still much work to do. One of important research directions in this field in the last decade is applying this task to less-resourced languages (e.g. Polish ([14]), Basque ([17]), and Czech ([11])). After RuCor, the first open Russian corpus with coreference annotation, was made available to the public ([18]), it became possible to create a coreference resolution system for Russian.

2. Background

Although the research on coreference resolution started more than 40 years ago (some of the classical papers include [5]–[2]), the machine learning approach to this task is relatively recent. One of the first papers on applying the ML approach to the task of coreference resolution was the seminal paper of Soon et al. ([16]). It introduced the mention-pair model of coreference resolution which was widely used since then.

This model works as follows: for every noun phrase that could be a mention generates a number of candidate antecedents from the preceding noun phrases. For each pair the classifier is invoked. The first (or the best, depending on the algorithm) positive pair is chosen. A set of pairs for training the classifier is created in a similar way.

Even though this model has flaws, e.g. its locality (it allows incompatible pairs in the chain) it is still widely used, especially as a baseline model.

3. Experiments

3.1. Data

Our experiments were conducted on RuCor, a Russian coreference corpus initially created as a dataset for the RU-EVAL campaign[2]. The collection contains 180 texts or text fragments (3,638 coreferential chains with 16,557 noun phrases in total) taken from different genres, such as news, scientific articles, blog posts and fiction. The corpus is already preprocessed: each text is tokenized, split into sentences, morphologically tagged and syntactically parsed using the tools developed by Serge Sharoff ([15]). The morphological tags were checked and fixed manually, since it was previously shown that errors on this level affects significantly the quality of a related task ([6]).

The annotation scheme is based on MUC-6 scheme. It includes only the annotation of the expressions referring to the real-world entities (e.g. there are no coreference

[2] The corpus may be downloaded from http://rucoref.maimbava.net.

links for abstract notions or generic expressions). The other constraint is that only identity relation is annotated, bridging or near-identity relations were not taken into consideration. The Gold Standard corpus contains annotations only for those NPs that are mentions (i.e. parts of a coreference chains), so singletons are not annotated.

For our experiments[3] the corpus was randomly split into a training and a test set (70% and 30% respectively). We performed experiments on both gold mentions NPs that are annotated in the corpus as parts of some coreference chain and predicted mentions all NPs extracted from the corpus automatically based on the dependency parses.

For testing we used CoNLL reference coreference scorers ([13]) a set of tools used for scoring in the CoNLL evaluation campaign as a reliable implementation of scoring algorithms that can produce comparable results. Particularly, we used two metrics to evaluate our experiments: the MUC score ([19]) and the B3 score ([1]). The former is a baseline score used in nearly every paper on coreference resolution. Even though it has some flaws (e.g. a baseline system that treats all mentions as one coreference chain achieves around 80% precision and 100% recall on a MUC-5 corpus), it is crucial to provide this score when establishing a baseline. The latter provides the quality of constructing coreference chains on average hence gives a good approximation of how well the method works in general.

The scores are calculated based on the full noun phrases, so the error in the NP extraction leads to decreasing the coreference score. In order to evaluate the coreference resolution step itself, without penalties for the incorrect NP extraction, we used the so-called *Gold boundaries* evaluation strategy: the edges of the noun phrases from the coreferent chains were corrected using the GS data.

3.2. Features for the baseline models

Initially we created a set of mention-pair classifiers using simple shallow features proposed in a seminal paper by Soon et al. ([16]), a system that is often used as a baseline for machine learning models for coreference resolution.

Some features used in the paper are inapplicable to Russian in a straightforward way, for example, the feature *Definite Noun Phrase*, which should be 1 if the noun phrase starts with a definite article. Given that Russian is an article-less language, detecting definite NPs is a separate, complicated task.

Some other features are hard to implement, for example *Semantic class agreement*, which should be set to 1 if the two candidate NPs has the same semantic class. Another feature like this is *Alias*, which should be set to 1 if one NP is an alias of another. Due to the small amount of available NLP tools and resources that work with Russian, there is no straightforward way to obtain values for those features. Available tools and possible ways to extract this knowledge are discussed in section 3.5.

To compensate this, in the baseline system we replaced those features with the heuristics which use other shallow features that should correlate with original missing ones:

---

[3] The Jupyter notebooks which reproduce the experiments may be downloaded from https://github.com/max-ionov/rucoref/tree/master/notebooks/coreference-dialog-2017

1. *Animacy agreement*: True if both NP are animate or both NP are inanimate. This feature is used as a poor-man replacement for a *Semantic Class agreement* feature. The class hierarchy in this case consists of two classes on one level: *object / living thing*.
2. *Head match*: True if both NPs are not pronouns and an antecedent candidate head matches an anaphoric NP head. This feature is a simple analogue for an *Alias* feature.

3.3. Baseline experiment results, Rule-based

The first four systems that we created were rule-based. They were designed to yield very high precision sacrificing the recall:

- StrMatch: two NPs corefer if their lemmas are the same (only for nouns and deictic pronouns).
- StrMatchPro: like the previous one, only non-deictic pronouns are paired with the nearest NP that agrees in gender and number.
- HeadMatch: two NPs corefer if their heads are the same (only for nouns and deictic pronouns).
- HeadMatchPro: like the previous one, only non-deictic pronouns are paired with the nearest NP that agrees in gender and number.

The results for the baseline systems are given in the tables 1 and 2.

Table 1: Rule-based coreference systems, gold mentions

	MUC			B^3		
	P	**R**	**F_1**	**P**	**R**	**F_1**
StrMatch	94.29	37.36	53.52	97.09	38.19	54.82
StrMatchPro	84.90	52.42	64.82	89.34	43.35	58.37
HeadMatch	87.78	47.06	61.27	92.11	43.64	59.22
HeadMatchPro	84.89	52.50	64.87	89.29	43.38	58.40

While the HeadMatch and the StrMatch baselines resolvers show very high precision, two other algorithms increase the recall by adding resolving personal pronouns. Even though the precision is much lower in the latter two cases, the overall quality is still better.

Nevertheless, the application of these algorithms are obviously very limited.

Table 2: Rule-based coreference systems, gold boundaries, mention detection f-score 51.38

	MUC			B^3		
	P	**R**	**F_1**	**P**	**R**	**F_1**
StrMatch	52.86	32.29	40.09	33.54	34.04	33.79
StrMatchPro	34.40	45.46	39.16	26.89	39.58	32.02
HeadMatch	35.26	41.38	38.07	29.57	38.88	33.59
HeadMatchPro	34.40	45.49	39.18	26.89	39.58	32.02

3.4. Baseline experiment results, ML approach

To incorporate more features without a need of combining them and handcrafting a set of rules we created a coreference resolution system based on a classic system by Soon et al. ([16]).

We used a decision tree classifier implemented in the scikit-learn Python module ([12]). Training and test instances were generated in the following way: for each anaphor-antecedent pair one positive example was generated. Also, a negative example was created for each candidate antecedent between the true pair of an anaphor and an antecedent.

For a baseline ML classifier we used a set of 11 features for the classifier:

1. The distance between an anaphoric NP and a candidate antecedent is 1 sentence.
2. Both NPs are not pronouns and after removing any demonstratives they match.
3. NPs agree in animacy and if they are not pronouns their syntactic heads match.
4. Anaphoric NP is a pronoun.
5. Candidate antecedent is a pronoun.
6. Both NPs are pronouns.
7. NPs agree in gender.
8. NPs agree in number.
9. Both NPs are proper.
10. An anaphoric NP is a demonstrative.
11. NPs are in the appositive relation.

Most of the features were taken from the original paper, some other were adapted to use with Russian (see 3.2 for details).

In order to decrease the noise in the data, we tweak the classifier setting the minimum number of samples required to be at a leaf node to 1% of the training samples. This simplifies the tree and makes the classifier less prone to overfitting.

The results are presented in the MLMentionPair row in the table 3 for the gold mentions case and 4 for the predicted mentions case. The classifiers show slightly better results than the rule-based ones: lowering the precision, it increases the recall.

Interestingly, training the classifier with a feature *Head Match* without any restrictions on part of speech yields better results which asymptotically approach the results of *HeadMatch* baseline classifier. A further analysis of feature importances for the classifier shows that this feature is the only one that takes part in classification decisions in this case.

To improve the baseline results, we added more features to the classifiers that can be grouped into 4 classes: *distance* features, *morphological*, *lexical* and *syntactical*.

The *Distance* group we contains the original distance feature and the binary feature if there are more than 3 nouns between the NPs. Other distance features that were tested (either in terms of nouns, NPs or words did not lead to an increase in quality).

The *Morphological* group consists of binary features checking if NPs are the pronouns of a specific type: deictic, relative, reflexive or possessive. This group increased the quality of noun-pronoun coreference resolution drastically.

Lexical features are two heuristics: a feature showing if one of the NPs equals to a noun modifier in another NP. This feature allows to resolve the cases like *prezident Obama* 'president Obama'—*prezident* 'the president' even if the head of the first NP is *Obama*. The second feature is a simple heuristic for acronym detection.

Syntactic features checks if either of NPs is a subject, an object, whether they are situated in the beginning of a sentence and whether they are both subjects (syntactic parallelism).

Table 3: ML-based coreference systems, gold mentions

	MUC			B^3		
	P	**R**	**F_1**	**P**	**R**	**F_1**
HeadMatchPro	84.89	52.50	64.87	89.29	43.38	58.40
MLMentionPair	73.98	62.24	67.61	71.40	49.34	58.36
MLUpdated	79.29	63.01	70.25	79.42	48.39	60.14

Table 4: ML-based coreference systems, gold boundaries, mention detection f-score 51.21

	MUC			B^3		
	P	**R**	**F_1**	**P**	**R**	**F_1**
HeadMatchPro	34.40	45.49	39.18	26.89	39.58	32.02
MLMentionPair	37.91	55.85	45.16	21.88	43.98	29.22
MLUpdated	37.94	53.87	44.52	25.00	42.61	31.51

The classifier with a full feature set outperforms both the naive baseline and the baseline ML system on gold mentions but does not performs so well on predicted mentions. The B^3 metric shows that the MLUpdated classifier is more precise than the basic ML classifier but less precise than the rule-based baseline, whereas it shows more recall than the rule-based classifier and slightly more than the basic ML one. The reason behind this is that it handles more cases than both classifiers and it can correctly distinguish more cases than the basic ML system.

Still, it is impossible to resolve correctly coreferent NPs which are, for example, synonyms. To do so, we need to incorporate semantic information into the classifier feature set.

3.5. Incorporating semantic information

Incorporation of some semantic information has shown to be very useful for coreference resolution. Named entity detection can improve the quality by giving the possibility to compare semantic classes of two mentions. Named entity linking can resolve the coreference, stating that two NE should be linked to the same object. Measuring semantic relatedness between two NPs, we can get the probability of two mentions to be coreferent.

As it was already mentioned, there is a limited amount of NLP tools and resources that work with Russian that are available for research purposes. For example, there is no publicly available NER detector. There is a freely available corpus with annotated named entities that can be used for train the NER detector[4], but creating a named entity resolution system is out of the scope of this paper. Overall situation with resources that can be used to extract semantic information is becoming much better over the last few years: A subset of the RuThes ontology was made available as RuThes-Lite ([9]), YARN, a crowdsourcing project to build a WordNet for Russian is developing rapidly ([3]), word2vec models trained on Russian texts are available as a part of the project "RusVectōrēs" ([8]).

In this paper we compare 3 different instruments to integrate the semantic information in the coreference resolution system:

1. A list of named entities with their types and possible synonyms.
2. A word2vec model to check if the two NPs are synonyms.
3. A thesaurus to check if two NPs are synonyms or one of them is a more general term for another.

With the aid of those instruments we can improve both an *Alias* and a *Semantic agreement* features in our classifier:

1. One NP is an alias of another.
2. NPs agree in animacy and if they are not pronouns they are semantically compatible if there is information about the semantic class of the mentions. Otherwise their heads should match.

For a first experiment, we constructed two lists of named entities. First, we compiled a small list which contained 5 frequent NEs from the training set. In the second iteration we used the GeoNames database[5] to create a list of geographical names as named entities. In total 32 934 names were used. Both experiments showed an improvement over a baseline system. Even using the small list improved the recall for the coreference resolution, improving the F-measure as a result. Further extension of the list improves the results further.

For the second experiment we employed the "RusVectōrēs" word2vec model. As a preliminary experiment we used it only to enrich the *Alias* feature. If the semantic similarity between the heads of the two NPs were more than the threshold, the NPs were considered aliases. This approach gave a slight improvement over the initial results, improving the recall and decreasing the precision. The reason behind the small impact is that there were very few cases when the similarity between NPs were big enough. Still, as described below in 4, this method allows to join the NPs that cannot be joined without the semantic information. There are other possible ways to employ word2vec models, but they are not covered in this paper and are for future research.

The third source of semantic information was RuThes-Lite, a thesaurus with several relations between concepts and a set of string representations for each concept. We used it to implement an *Alias* feature and to replace the *Semantic agreement* feature: if the heads of two NPs are the synonyms in RuThes-Lite, or there is a path

[4] https://github.com/dialogue-evaluation/factRuEval-2016

[5] http://geonames.org/

between the heads of two NPs using the parent concept relation ('ВЫШЕ') they are considered aliases. If the domains of the heads of two NPs are the same they are considered semantically related. As in the previous case, the approach shows a slight improvement, increasing the recall of the system.

The results for all the experiments are given in Table 3 for gold mentions and Table 4 for predicted mentions.

Table 5: The impact of semantic information, gold mentions

	MUC			B^3		
	P	R	F_1	P	R	F_1
MLMentionPair	73.98	62.24	67.61	71.40	**49.34**	58.36
MLUpdated	79.35	63.44	70.51	**79.37**	48.60	60.29
NamedEntities	**79.43**	63.72	70.71	**79.37**	48.86	**60.48**
Word2vec	79.29	63.49	70.52	79.25	48.64	60.28
RuThes	79.19	63.79	70.66	78.92	48.78	60.29
All	79.19	**63.97**	**70.77**	78.85	48.94	60.39

Table 6: The impact of semantic information, gold boundaries, mention detection f-score 51.21

	MUC			B^3		
	P	R	F_1	P	R	F_1
MLMentionPair	37.91	55.85	**45.16**	21.88	**43.98**	29.22
MLUpdated	37.94	53.87	44.52	**25.00**	42.61	31.51
NamedEntities	**38.01**	54.10	44.65	24.99	42.83	**31.56**
Word2vec	37.69	53.92	44.37	24.95	42.68	31.49
RuThes	36.27	54.20	43.46	24.63	42.83	31.28
All	36.08	**54.32**	43.36	24.60	42.94	31.28

4. Discussion

In this paper we described our experiments on building a coreference resolution system for Russian. We established a baseline for Russian by building an ML-based system using the features proposed in [16], and showed that by adding shallow non-semantic features we can improve its F-measure by 2–3%.

Our experiments with adding semantic information from various sources showed that even the tiniest bits of semantic information can improve the overall quality of the system. It helps coreference linking improving the overall recall, although it usually decreases the precision. At the same time we showed that using the ontology and distributional model had a very small impact on the results.

Named entity list showed the largest impact on the results, mainly because its decrease in precision was minimal due to its nature. The main limitation of this

approach is, obviously, a limited size of such list: with its growth the precision should drop due to inevitable cases of homonymy.

In the case of the distributional model, the main reason of its small impact was the small amount of cases where the heads of NPs had a similarity score higher than the threshold. With a decreased threshold there were more cases but more unwanted results (mainly co-hyponyms like *muzh* 'husband'—*zhena* 'wife'). Nevertheless, this model improved the results in some cases that were impossible without it, e.g. *muzh* 'husband'—*suprug* 'spouse'. Those cases can be easily solved also by ontologies like RuThes, as we will see below, but theoretically a distributional model trained on different kinds of texts should work well with non-standard vocabulary. Another space for an improvement in this area is to use a distributional model in a more elaborated way, not only as a filter with a threshold. This is a direction for a future research.

The impact of using the RuThes ontology was also low, again, mainly because of a small amount of cases in which it was used, but as with the distributional model, its use was crucial for some cases, e.g. *rabota* 'job'—*trud* 'labour'. The main problem of this approach was homonymy. In cases like *litso* 'face' / 'person'—*chelovek* 'person', NPs were erroneously considered as aliases. Since this is not the problem of an ontology but its usage, this can be improved in the future.

There are still cases which require semantic information which cannot be linked with the methods described in the paper. There are two important classes of them. The first one is when NPs are from the same base class but the connection between them is not universal but arises in the text. E.g. *tjotushka* 'aunt'—*pomesh'itsa* 'landlady'. In this example, there is a person who is an aunt and a landlady at the same time. This problem in principle can be solved with an ontology but this solution may increase the noise in the output.

Another problem arises when the equality between the NPs are derived from the world knowledge. Like *ministr* 'minister'—*pomosh'nik prezidenta* 'a person who helps the president'. This kind of information cannot be extracted from the ontology (at least, not all the cases, even if an ontology contains some specific relations to tackle this problem).

References

1. *Bagga A., Baldwin B.* (1998), Algorithms for scoring coreference chains, The First International Conference on Language Resources and Evaluation Workshop on Linguistics Coreference, pp. 563–566.
2. *Bobrow D. G.* (1964), A question-answering system for high school algebra word problems, Proceedings of the October 27–29, 1964, fall joint computer conference, part I, ACM, pp. 591–614.
3. *Braslavski P., Ustalov D., Mukhin M., and Kiselev Y.* (2016), Yarn: Spinning-in-progress, Proceedings of the Eight Global Word net Conference, pp. 58–65.
4. *De Marneffe M.-C., Recasens M., C. Potts* (2015), Modeling the lifespan of discourse entities with application to coreference resolution, Journal of Artificial Intelligence Research, vol. 52, no. 1, pp. 445–475.
5. *Hobbs J.* (1978), Resolving pronoun references, Lingua, Vol. 44, pp. 311–338.
6. *Ionov M., Kutuzov A.* (2014), Influence of morphology processing quality on automated anaphora resolution for Russian, Proceedings of the international conference "Dialogue-2014", Moscow.

7. *Ionov M., Toldova S.* (2016), Identification of singleton mentions in Russian, CEUR Workshop Proceedings, in press.
8. *Kutuzov A., Andreev I.* (2015), Texts in, meaning out: neural language models in semantic similarity task for Russian, Proceedings of the International Conference on Computational Linguistics and Intellectual Technologies "Dialogue 2015", Moscow, Vol. 2, pp. 133–145.
9. *Loukachevitch N., Dobrov B., Chetviorkin I.* (2014), Ruthes-lite, a publicly available version of thesaurus of Russian language Ruthes, Computational Linguistics and Intellectual Technologies: Papers from the Annual International Conference Dialogue, Bekasovo, Russia, pp. 340–349.
10. *Ng V., Cardie C.* (2002), Identifying anaphoric and non-anaphoric noun phrases to improve coreference resolution, Proceedings of the 19th International Conference on Computational Linguistics, Volume 1, ser. COLING'02. Stroudsburg, PA, USA: Association for Computational Linguistics, pp. 1–7.
11. *Novák M., Žabokrtský Z.* (2011), Resolving noun phrase coreference in Czech, 8th Discourse Anaphora and Anaphor Resolution Colloquium, DAARC 2011, Springer Berlin Heidelberg, pp. 24–34.
12. *Pedregosa F., Varoquaux G., Gramfort A., Michel V., Thirion B., Grisel O., Blondel M., Prettenhofer P., Weiss R., Dubourg V., Vanderplas J., Passos A., Cournapeau D., Brucher M., Perrot M., Duchesnay E.* (2011), Scikit-learn: Machine learning in Python, Journal of Machine Learning Research, vol. 12, pp. 2825–2830.
13. *Pradhan S., Luo X., Recasens M., Hovy E., Ng V., Strube M.* (2014) Scoring coreference partitions of predicted mentions: A reference implementation, Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, Vol. 2: Short Papers, Baltimore, Maryland: Association for Computational Linguistics, June 2014, pp. 30–35. [Online]. Available at: http://www.aclweb.org/anthology/P14-2006
14. *Savary A., Ogrodniczuk M., Zawislawska M., Glowinska K., Kopec M.* (2015), Coreference: Annotation, Resolution and Evaluation in Polish, Walter de Gruyter GmbH, Berlin.
15. *Sharoff S., Nivre J.* (2011), The proper place of men and machines in language technology: Processing Russian without any linguistic knowledge, Proceedings of the International Conference on Computational Linguistics and Intellectual Technologies "Dialogue 2011", Bekasovo, pp. 591–605.
16. *Soon W. M., Ng H. T., Lim D. C. Y.* (2001), A machine learning approach to coreference resolution of noun phrases, Computational linguistics, vol. 27, no. 4, pp. 521–544.
17. *Soraluze A., Arregi O., Arregi X., Ceberio K., De Ilarraza A. D.* (2012), Mention detection: First steps in the development of a basque coreference resolution system, Proceedings of KONVENS 2012, pp. 128–136.
18. *Toldova S., Grishina Y., Ladygina A., Sim G., Kurzukov M., Azerkovich I., Vasilyeva M.* (2014), Coreference corpus in Russian, Program & Book of Abstracts. CILC 2014. Las Palmas de Gran Canaria, Aelinco, pp. 154–155.
19. *Vilain M., Burger J., Aberdeen J., Connolly D., Hirschman L.* (1995), A model-theoretic coreference scoring scheme, Proceedings of the 6th Conference on Message Understanding, ser. MUC6 '95. Stroudsburg, PA, USA: Association for Computational Linguistics, 1995, pp. 45–52. [Online], available at: http://dx.doi.org/10.3115/1072399.1072405