Coreference Resolution: to What Extent Does it Help NLP Applications?

Ruslan Mitkov, Richard Evans, Constantin Orăsan, Iustin Dornescu and Miguel Rios

Research Institute in Information and Language Processing
University of Wolverhampton, United Kingdom
Email: {R.Mitkov, R.J.Evans, C.Orasan, I.Dornescu2, M.Rios}@wlv.ac.uk

Abstract This paper describes a study of the impact of coreference resolution on NLP applications. Further to our previous study [1], in which we investigated whether anaphora resolution could be beneficial to NLP applications, we now seek to establish whether a different, but related task — that of coreference resolution, could improve the performance of three NLP applications: text summarisation, recognising textual entailment and text classification. The study discusses experiments in which the aforementioned applications were implemented in two versions, one in which the BART coreference resolution system was integrated and one in which it was not, and then tested in processing input text. The paper discusses the results obtained.

Key words: coreference resolution, text summarisation, recognising textual entailment, text classification, extrinsic evaluation

1 Introduction

In [1], we conducted the first extensive study into whether NLP applications could benefit from anaphora resolution. In this work we conducted extrinsic evaluation of our anaphora resolution system MARS [2] by seeking to establish whether and to what extent anaphora resolution can improve the performance of three NLP applications: text summarisation, term extraction and text categorisation. On the basis of the results we concluded that the deployment of anaphora resolution has a positive albeit limited impact. More specifically, the deployment of anaphora resolution increased the performance rates of these applications but the difference was not statistically significant.

In this study we revisit this topic but this time we have opted for seeking to establish the impact that coreference resolution could have on NLP applications. While some authors use the terms coreference (resolution) and anaphora (resolution) interchangeably, it is worth noting that they are completely distinct terms or tasks [3]. Anaphora is cohesion which points back to some previous item, with the 'pointing back' word or phrase called an anaphor, and the entity to which it refers, or for which it stands, its antecedent. Coreference is the act of picking out the same referent in the real world. A specific anaphor and more than one of the preceding (or following) noun phrases may be coreferential, thus forming a coreferential chain of entities which have the same referent.

Coreference is typical of anaphora realised by pronouns and non-pronominal definite noun phrases, but does not apply to varieties of anaphora that are not based on referring

Petr Sojka, Aleš Horák, Ivan Kopeček and Karel Pala (Eds.): TSD 2012, LNAI ????, pp. 1–12, 2012.

expressions, such as verb anaphora. However, not every noun phrase triggers coreference. Bound anaphors which have as their antecedent quantifying noun phrases such as *every man*, *most computational linguistics*, *nobody*, etc. are another example where the anaphor and the antecedent do not corefer. As an illustration, the relation in '*Every man* has *his* own agenda' is only anaphoric, whereas in '*John* has *his* own agenda' is both anaphoric and coreferential. In addition, while identity-of-reference nominal anaphora involves coreference by virtue of the anaphor and its antecedent having the same real-world referent, identity-of-sense anaphora (e.g. 'The man who gave his paycheck to his wife was wiser than the man who gave it to his mistress') does not. Finally, there may be cases where two items are coreferential without being anaphoric. Cross-document coreference is an obvious example: two mentions of the same person in two different documents will be coreferential, but will not stand in anaphoric relation.

Having explained the difference between the terms/phenomena *anaphora* and *coreference*, we should point out that the tasks *anaphora resolution* and *coreference resolution* are not identical either. Whereas the task of anaphora resolution has to do with tracking down an antecedent of an anaphor, coreference resolution seeks to identify all coreference classes (chains).

In this study we seek to establish whether the employment of coreference resolution to NLP applications is beneficial. The investigation has been undertaken by means of experiments involving three applications: text summarisation, textual entailment and text classification. It differs from our 2007 study, not only in the employment of a specific NLP task (coreference resolution as opposed to anaphora resolution) and in the applications covered (recognising textual entailment is a new NLP application), but also in the data selected for the current experiments. Since 2007 there have been significant developments in the construction and sharing of large-scale resources and this has been an ongoing trend in Natural Language Processing. By way of example, research in Textual Entailment is supported by the availability of several annotated datasets. These resources typically consist of sets of T-H pairs manually annotated with a Boolean value to indicate whether or not H is entailed by T. In the current paper, datasets RTE1 [4], RTE-2 [5], and RTE-3 [6] are used to evaluate the impact of coreference resolution on automatic RTE. We have opted to use such publicly available resources in spite of the fact that, as a result, we had to resort to the exploitation of different data for every evaluation/application in contrast to our previous experiments where we benefited from a common corpus.

The development of automatic coreference resolution systems began in earnest in 1996 in response to the MUC-6 competition organised by NRAD with the support of DARPA [7]. Since then, numerous coreference resolution systems have been developed, typically using machine learning, and exploiting supervised [8]; [9]; [10] and unsupervised (clustering) methods [11]. Current approaches continue to exploit machine learning, seeking improved models for the resolution process, based on various linguistic and contextual features [12].

In the research described in the current paper, the publicly available BART toolkit was exploited [13]. The coreference resolution system distributed with BART is reported to offer state of the art performance in coreference resolution, particularly with regard to the resolution of pronominal mentions. With reported recall in pronoun resolution at the

level of 73.4%, BART's performance is close to that of specialised pronoun resolution systems.

BART works by first preprocessing input documents in order to detect potential mentions such as pronouns, noun chunks, base noun phrases, and named entities. For each detected anaphor, the system extracts each pair consisting of the anaphor and a potential antecedent for that anaphor. Pairs are represented using a feature set that combines features exploited in the system developed by Soon et al. [8] with features encoding the syntactic relation between anaphors and their potential antecedents [14], and features based on knowledge extracted from Wikipedia. The coreferentiality of each pair of mentions is then determined using machine learning.

The automatic identification of coreference chains enables practical NLP applications to substitute semantically ambiguous references to entities (such as pronouns) with more informative phrases, before subsequent processing.

The rest of the paper is organised as follows. In Section 2 we review related research. In sections 3, 4 and 5 we present the experimental settings related to the application of BART to text summarisation, recognising textual entailment and text classification respectively. In section 6 we discuss the evaluation results and finally in the concluding section 7 we summarise the results of this study.

2 Related Research

The research described in the current paper is motivated by previous research in the fields of automatic summarisation, recognition of textual entailment, and text classification. This work highlights various challenges to be addressed in each area, and describes different attempts to ameliorate them.

2.1 Automatic summarisation

One of the drawbacks of most implementations of keyword-based summarisation is that they consider words in isolation. This means that most implementations will fail to recognise when two words are in an anaphoric relation or they are part of the same coreferential chain. For this reason, it was argued that it is possible to improve the results of an automatic summarisation system that relies only on keywords by obtaining better frequency counts using the information from an anaphora or coreference resolver.

Previous experiments using a pronominal anaphora resolver showed limited impact. Orăsan [15] shows that an automatic pronoun resolver does not really improve the results of an automatic summarisation method for scientific documents. However, he uses the annotated data to simulate a pronoun resolver and shows that a high accuracy pronoun resolution is useful in the summarisation process. Experiments on newswire texts show similar results [1] and lead to the conclusion that it may be possible to improve the results of the automatic summariser by using a coreference resolver instead of just a pronoun resolver. This conclusion is also supported by the research presented in [16] where the results on an LSA-based summariser are improved when a coreference resolver is used. In his thesis [17] and similar to our previous findings [1], Kabadjov establishes that the employment of his anaphora resolution system GUITAR to summarisation through

substitution leads to limited (statistically insignificant) improvement. On the other hand, anaphora resolution leads to statistically significant improvements, when lexical and anaphoric knowledge is integrated into an LSA-based summariser.

2.2 Textual entailment

The exploitation of coreference information in tasks related to RTE is motivated by previous work which has demonstrated encouraging results. In the context of RTE competitions exploiting the RTE-3 dataset, systems exploiting coreference information have not been the highest ranking ones in terms of performance, but results have been encouraging in paraphrase recognition [18] and in the identification of sentences entailed by input queries [19]. Research reported in [20] suggests that discourse information can improve RTE in the context of the Search Task, but that such information should be integrated into the inference engine as opposed to serving as a preprocessing step or a feature exploited by an ML algorithm.

2.3 Text classification

One challenge for TC systems is caused by synonymy, e.g. when new terms are synonyms of observed terms but are ignored by the classifiers, and by polysemy, e.g. when a new sense of a known term is used. Previous work studied the use of WordNet in addressing the first problem, or WSD systems in addressing the second. Coreference is another natural language phenomenon which affects TC in a similar way to synonymy and polysemy: on the one hand the same entities or concepts are mentioned multiple times but using different words; on the other hand the same words can be used to refer to different concepts, as is usually the case with pronominal anaphoric expressions.

Coreference resolution provides discourse level information which could help classifiers alleviate some of these issues, but its usefulness for TC has received little attention in the literature. Incorporating coreference information in TC is usually achieved by changing the weights of those terms which occur in coreference chains [21].

2.4 Other related work

Hendrickx et al. [22] investigate the effect resulting from the deployment of a coreference resolution system for Dutch [23] in relation to information extraction and question answering. Their findings point to some increase in performance of information extraction after incorporating coreference resolution. However, this increase is not statistically significant. When incorporating their coreference system for Question Answering, while the number of extracted facts increases by 50%, overall performance decreases significantly. Overall, due to the number of additional facts retrieved, this leads to a 5% improvement in performance on the QA@CLEF 2005 test set.

3 Automatic Summarisation: Experimental Settings

This section presents the settings for an experiment where a keyword-based summariser is enhanced with information from a coreference resolver. Section 6.1 describes the evaluation results and discussion.

For the experiments presented in this section, we reimplemented the keyword-based summariser described in [24] and used only the best performing setting identified there. Therefore, for our experiment, words are scored on the basis of their frequency in the document and stopwords are filtered out.[1] On the basis of previous research, we decided to count words as they appear in the text and not to do any morphological processing. The final score of a sentence is calculated by adding up the scores of the words contained in the sentence. The summary is produced by extracting the sentences with the highest scores until the desired length is reached.

In order to obtain better frequency counts, we used the information from a coreference resolver to boost the scores of the sentences which contain coreferential chains by the scores of the chains. For this purpose we used two settings: In the first, we increased the score of a chain by the score of the longest mention which occurs in the chain. When several mentions of the same length are found in the text, the first one is used. This setting is used in order to have a setting similar to the other experiments presented in this paper. In the second setting, the score of the chain is given by the greatest score of the mentions it contains. This is to reflect the fact that the importance of a mention is given by its content, not by its length. In both experiments, the chains containing only one element are discarded.

The evaluation was carried out using 89 randomly selected texts from the CAST corpus [25]. The CAST corpus is a corpus of newswire texts annotated with information about the importance of the sentences with regards to the topic of the document. Annotators were asked to manually annotate 15% of the most important sentences as *ESSENTIAL* and a further 15% as *IMPORTANT*. In this way, it is possible to evaluate summarisation methods which produce summaries of 15% and 30% compression rates. All the texts selected for this experiment were annotated with coreference information using BART [13].

For the evaluation (Section 6.1), we compared the sets of sentences selected by the program with the set of sentences annotated by humans. On the basis of this comparison, we calculated precision and recall and we report the results using F-measure.

4 Recognising Textual Entailment: Experimental Settings

RTE can be regarded as a binary classification task in which each pair of text and hypothesis is classified according to whether or not the text is entailed by the hypothesis. In this context, RTE benchmark datasets are used to train a classifier [26]. We followed the methodology used by Castillo [19] to process coreference chains in which each mention in a chain is substituted by the longest (most informative) mention. In contrast to that approach, we used the two-way benchmark datasets (i.e. text and hypothesis pairs that have been manually classified as true/false) for training and testing. We appended each T-H pair as one piece of text, and processed each pair using the BART[2] coreference

[1] One of the differences between the current implementation and the implementation reported in [24,15] comes from the fact that the current implementation is in Python and uses NLTK for the stoplist and processing.

[2] `http://www.bart-coref.org/`

resolver. Then, for each coreference chain we selected the mention with the greatest word length and replace all other mentions in the chain with this most informative mention.

The RTE system is based on a supervised Machine Learning algorithm. The algorithm is trained to classify T-H pairs by means of metrics that assess the similarity between T and H. These include *lexical metrics* (precision, recall, and F-score), used with a bag-of-words representation of the T-H pairs; and metrics such as *BLEU* [27]; *METEOR* [28]; and *TINE* [29].

With these metrics we built a vector of similarity scores used as features to train a Machine Learning algorithm. We used the development datasets from the RTE 1 to 3 benchmark to train a Support Vector Machine algorithm distributed with Weka[3] with no parameter optimisation. Then, we tested the models using 10-fold-cross-validation over the development datasets and we compared them against the test datasets.

5 Text Classification: Experimental Settings

A TC system has three processing stages: document processing, classifier learning and evaluation. During document processing, or document indexing, textual documents are analysed and represented in a compact form as a weighted term vector, where each term corresponds to a feature and the weight quantifies its importance for a particular document. Terms often correspond to words mentioned in the document, but usually stop-words are removed and stemming can be applied. The weight of each term is usually computed using either statistical or probabilistic techniques, with tf · idf being one of the most popular methods. As unseen documents are likely to use vocabulary terms which did not occur in training, classifiers tend to perform better on training data than on test data. To reduce overfitting, a dimensionality reduction step can be employed which also reduces the computational complexity for building classifier models. Dimensionality reduction in TC usually involves a term selection method in which only the most relevant terms are used to represent documents.

A standard BOW approach was used in this study: punctuation and stop-words have been removed, all words have been converted to lower-case characters and Porter's stemmer was applied. Both single words and bigrams were used as terms [30].

Several studies [31,32] found that feature selection methods based on χ^2 statistics consistently outperformed those based on other criteria (including information gain) for the most popular classifiers used in TC. The terms with a document frequency less than 5 were also removed, as χ^2 is known to be less reliable for rare words [31]. Both methods were applied and 10% of the terms were selected for the vector space representation.

Length-normalised feature vectors were built using the standard $tf \cdot$idf function using log smoothing: $\text{tfidf}(t_k, d_j) = \text{tf}(t_k, d_j) \cdot \log \frac{|D|}{|D_k|}$, where $\text{tf}(t_k, d_j) = 1 + \log(\text{occ}(t_k, d_j))$ for terms t_k with at least one occurrence in document d_j and 0 otherwise, $|D|$ is the collection size and $|D_k|$ is the document frequency of term t_k.

The BART [13] coreference resolution system was run on the original plain text version of the documents in the R(10) corpus to identify coreference chains. This information was used to boost the weights of terms included in the chains, by using

[3] http://www.cs.waikato.ac.nz/ml/weka/

a modified term frequency function: $\text{tf}^{\text{coref}}(t_k, d_j) = \sum_{c \in C_{k,j}} \text{len}(c)$, where len(c) is the length of chain c, $C_{k,j}$ is the set of chains in document d_j containing at least one mention of term t_k. Essentially this function acts as if a term occurs in all mentions of a chain, as long as it occurs in at least one of them.

The SVM classifier was used in a binary mode: a different model was built for each of the 10 classes, including the term selection step, also known as local-selection [32]. The average precision of the individual classifiers is used for evaluation (Section 6.3).

Some of the most popular collections used to compare different approaches to text categorisation are 20-newsgroups, [4] Reuters-21578 [5] and Reuters Corpus Volume 1 [33]. A study of the impact of class distribution on the performance of automatic TC systems [32] showed that the relative ranking of several approaches depends on which subset of the Reuters-21578 corpus is used. The study also revealed that the SVM classifier usually outranks other learners and that χ^2 usually achieves better results than other selection methods such as information gain, information ratio and mutual information.

In this paper, TC performance was assessed using a subset of the ApteMod dataset. ApteMod[6] is a collection of 10,788 documents from the Reuters-21578 corpus, partitioned into a training set with 7,769 documents and a test set with 3,019 documents. The subset exploited in the current work consists of the 10 categories with the highest number of positive training examples, also known as R(10) in the literature. This subset has also been exploited in research presented in [34,35,36].

6 Results and Discussion

This section presents an evaluation of the impact that automatically obtained coreference information [13] has on the three NLP applications described in Sections 3-5. In each case, a comparison is made between the efficacy of systems exploiting such information and those that do not.

6.1 Automatic summarisation

Table 1 presents the results of the evaluation. Column *Without BART* shows the results of the system which does not use any coreference information. Columns *With BART len* and *With BART weight* show the results when information from BART is used and correspond to the two settings presented described in Section 3. To our surprise, the results of the summarisation process decrease when coreference information is added. For both experiments, the decrease is statistically significant at 15% compression rate, but not at 30% compression rate.

The results presented in the table were obtained by giving a weight of 1 to the contribution from the coreference resolver. In order to find out whether it is possible to obtain better results using a different weight for this contribution, we run an experiment

[4] http://people.csail.mit.edu/jrennie/20Newsgroups/

[5] http://www.daviddlewis.com/resources/testcollections/reuters21578

[6] `http://www.cpan.org/authors/Ken_Williams/data/reuters-21578.readme`

Table 1. Evaluation results of the automatic summarisation method

Compression rate	Without BART	With BART len	With BART weight
15%	32.88%	28.62%	27.14%
30%	46.34%	45.88%	45.19%

where the contribution increased from 0 to 10 in 0.25 increments. Figure 1 and 2 show that as the contribution of coreference resolver increases, the results of the summariser decrease. This is the case for both experiments.

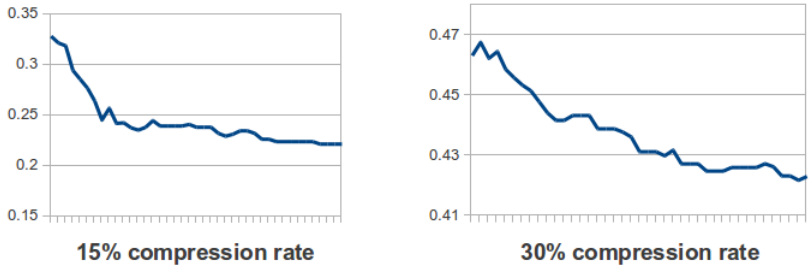

Figure 1. The results for the first setting of the summarisation experiment when the contribution of the coreference resolver increases

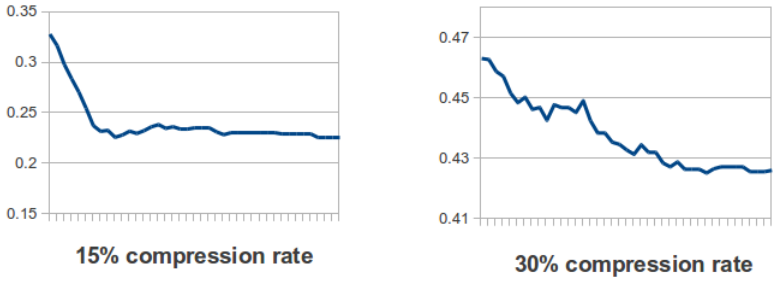

Figure 2. The results for the second setting of the summarisation experiment when the contribution of the coreference resolver increases

On the basis of the experiment presented in this section, it can be concluded that using information from a coreference resolver in such a simple way is not beneficial for automatic summarisation. The main reason for this is the errors introduced by the coreference resolver. As future research, we plan to employ the approach proposed

by [15] and use a gold standard to simulate a coreference resolver to find out what level of accuracy is necessary in order to improve the results of an automatic summariser.

6.2 Recognising textual entailment

Two different models for RTE were trained and tested, one of which exploits coreference information and one of which does not. The models use the same features, but with different preprocessed input data, where model *coref* denotes data processed with coreference information and model *token* denotes data processed without coreference information. Table 2 shows the comparison of both models' accuracy via 10-fold cross-validation over the development datasets.

Table 2. Results of 10-cross-fold-validation for Model *coref*: with coreference information and Model *token*: without coreference information

Dataset	Model coref	Model token
RTE-1	54.14	56.61
RTE-2	58.50	60
RTE-3	60.25	67.25

For 10-fold cross-validation, the model *token* (without coreference information) outperforms the model *coref*. In order to measure the differences between models we compared them over the test datasets and computed McNemar's test.

Table 3. Results over the test datasets for Model *coref*: with coreference information and Model *token*: without coreference information

Dataset	Model coref	Model token
RTE-1	56.87	56.87
RTE-2	57.12	59.12
RTE-3	60.25	61.75

Table 3 shows the results of both models over the test datasets. The models in which coreferential mentions are substituted show worse performance than those which do not make such substitutions, but the differences in performance are not statistically significant. Furthermore, when assessed over the RTE-1 dataset, this RTE system outperforms the system exploiting coreference information for paraphrase detection described in [18], but the models show similar performance regardless of whether or not coreference information is exploited.

We analysed the datasets in order to investigate cases in which the quality of the coreference resolver is decisive in affecting the performance of the method. For example, in the RTE-1 test dataset with 800 T-H pairs the average number of coreference chains per document is 2.1, the average number of words is 38.16 and the number of pairs with no chain is 60. In the RTE-3 dataset, over which the model obtains the best result,

with 800 T-H pairs and a similar average number of chains (1.80), the number of pairs without chains increases to 104. Thus, the method reduces the amount of errors with fewer coreference-enhanced T-H pairs. However, the appended T-H pairs do not differ from one dataset to another in terms of the number of words. Therefore the number of entities is insufficient to make a significant difference to the result. More conclusive results may be achieved in the context of the Search Task.

6.3 Text categorisation

In text categorisation, the two term weighting functions yield two experimental settings: *run-bow* using the standard pipeline, and *run-bart* which boosts the weight of terms occurring in coreference chains, in proportion to the chain length. The results of the experiments show that the difference between the two settings is small: the macro-averaged precision for *run-bow* is 95.6% and for *run-bart* it is 95.7%. The performance difference between the corresponding binary classifiers is also small, suggesting that the state-of-the-art approach using the bag-of-words representation does not take advantage of coreference information. This result confirms that of [21] who used a different coreference system and a slightly different weighting function.

This result can be partially explained by errors in the coreference chains produced by the resolver, but also suggests that a more explicit way of employing this information is necessary. The intuition is that the presence or absence of a particular entity or term better indicates the topic of the document than the actual number of times it is mentioned. Future investigations should consider ways in which coreference information can be used to enhance TC systems using semantic features to represent documents, which can make use of what entities represent, instead of just using entity names. A TC system employing a semantic representation using external knowledge could exploit coreference information directly, e.g. it could know that a document is about sport based on the number of mentions of sportspeople instead of their actual names, which could be very sparse and occur in too few documents.

7 Conclusions

This study sought to establish whether or not coreference resolution could have a positive impact on NLP applications, in particular on text summarisation, recognising textual entailment, and text categorisation. The evaluation results presented in Section 6 are in line with previous experiments conducted both by the present authors and other researchers: there is no statistically significant benefit brought by automatic coreference resolution to these applications. In this specific study, the employment of the coreference resolution system distributed in the BART toolkit generally evokes slight but not significant increases in performance and in some cases it even evokes a slight deterioration in the performance results of these applications. We conjecture that the lack of a positive impact is due to the success rate of the BART coreference resolution system which appears to be insufficient to boost performance of the aforementioned applications.

References

1. Mitkov, R., Evans, R., Orasan, C., Ha, L., Pekar, V.: Anaphora resolution: To what extent does it help nlp applications? In Branco, A., ed.: Anaphora: Analysis, Algorithms and Applications. Lecture Notes in Artificial Intelligence (LNAI 4410). Springer-Verlag (2007) 179—190
2. Mitkov, R., R., E., Orasan, C.: A new, fully automatic version of mitkov's knowledge-poor pronoun resolution method. In Gelbukh, A., ed.: Computational Linguistics and Intelligent Text Processing. Springer (2002) 169–187
3. Mitkov, R.: Anaphora Resolution. Longman, Cambridge, MA; London (2002)
4. Sekine, S., Inui, K., Dagan, I., Dolan, B., Giampiccolo, D., Magnini, B., eds.: Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing. Association for Computational Linguistics, Prague (2007)
5. Ido, R.B.H., Dagan, I., Dolan, B., Ferro, L., Giampiccolo, D., Magnini, B., Szpektor, I.: The second pascal recognising textual entailment challenge (2006)
6. Dagan, I., Glickman, O.: The PASCAL recognising textual entailment challenge. In: In Proceedings of the PASCAL Challenges Workshop on Recognising Textual Entailment. (2005)
7. Grishman, R., Sundheim, B.: Message understanding conference-6: A brief history. In: Proceedings of the 16th International Conference on Computational Linguistics, COLING-96, Copenhagen, Denmark (1996)
8. Soon, W.M., Ng, H.T., Lim, D.C.Y.: A machine learning approach to coreference resolution of noun phrases. Computational Linguistics **27** (2001) 521–544
9. Ng, V., Cardie, C.: Improving machine learning approaches to coreference resolution. In: Proceedings of ACL-2002. Association for Computational Linguistics (2002)
10. Uryupina, O.: Coreference resolution with and without linguistic knowledge. In: In Proceedings of LREC-2006, Genoa, Italy (2006) 893–898
11. Cardie, C., Wagstaff, K.: Noun phrase coreference as clustering. In: Proceedings of the 1999 Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora, College Park, MD, Association for Computational Linguistics (1999) 82–89
12. Ng, V.: Supervised noun phrase coreference research: The first fifteen years. In: Proceedings of ACL-2010. Association for Computational Linguistics (2010)
13. Versley, Y., Ponzetto, S.P., Poesio, M., Eidelman, V., Jern, A., Smith, J., Yang, X., Moschitti, A.: Bart: A modular toolkit for coreference resolution. In: In Proceedings of LREC-2008. (2008)
14. Yang, X., Su, J., Tan, C.L.: Kernel-based pronoun resolution with structured syntactic knowledge. In: Proceedings of CoLing/ACL-2006, Association for Computational Linguistics (2006)
15. Orăsan, C.: The Influence of Pronominal Anaphora Resolution on Term-based Summarisation. In Nicolov, N., Angelova, G., Mitkov, R., eds.: Recent Advances in Natural Language Processing V. Volume 309 of Current Issues in Linguistic Theory. John Benjamins, Amsterdam & Philadelphia (2009) 291–300
16. Steinberger, J., Kabadjov, M.A., Poesio, M., Sanchez-Graillet, O.: Improving LSA-based summarization with anaphora resolution. In: Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing (HLT/EMNLP), Vancouver, Canada (2005) 1–8
17. Kabadjov, M.: A Comprehensive Evaluation of Anaphora Resolution and Discourse-new Classification. Ph.D. thesis, Department of Computer Science, University of Essex (2007)
18. Andreevskaia, A., Li, Z., Bergler, S.: Can shallow predicate argument structures determine entailment? In: In Proceedings of the PASCAL Challenges Workshop on Recognising Textual Entailment. (2005)

19. Castillo, J.J.: Textual entailment search task: An initial approach based on coreference resolution. Intelligent Computing and Cognitive Informatics, International Conference on **0** (2010) 388–391
20. Mirkin, S., Dagan, I., Padó, S.: Assessing the role of discourse references in entailment inference. In: ACL. (2010) 1209–1219
21. Li, Z., Zhou, M.: Use semantic meaning of coreference to improve classification text representation. In: Information Management and Engineering (ICIME), 2010 The 2nd IEEE International Conference on. (2010) 416 –420
22. Hendrickx, I., Bouma, G., Coppens, F., Daelemans, W., Hoste, V., Kloosterman, G., Mineur, A.M., Vloet, J.V.D., Verschelde, J.L.: A coreference corpus and resolution system for Dutch. In: Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08). (2008) 144–149
23. Hendrickx, I., Hoste, V., Daelemans, W.: Semantic and syntactic features for anaphora resolution for dutch. In: Recent Advances in Natural Language Processing V. Volume 4919 of Lecture Notes in Computer Science. Springer (2008) 351–361
24. Orăsan, C.: Comparative evaluation of term-weighting methods for automatic summarization. Journal of Quantitative Linguistics **16** (2009) 67–95
25. Hasler, L., Orăsan, C., Mitkov, R.: Building better corpora for summarisation. In: Proceedings of Corpus Linguistics 2003, Lancaster, UK (2003) 309–319
26. Dagan, I., Dolan, B., Magnini, B., Roth, D.: Recognizing textual entailment: Rational, evaluation and approaches – erratum. Natural Language Engineering **16** (2010) 105
27. Papineni, K., Roukos, S., Ward, T., Zhu, W.J.: Bleu: a method for automatic evaluation of machine translation. In: Proceedings of the 40th Annual Meeting on Association for Computational Linguistics. ACL '02, Stroudsburg, PA, USA (2002) 311–318
28. Banerjee, S., Lavie, A.: METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In: Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization, Ann Arbor, Michigan (2005) 65–72
29. Rios, M., Aziz, W., Specia, L.: Tine: A metric to assess mt adequacy. In: Proceedings of the Sixth Workshop on Statistical Machine Translation, Edinburgh, Scotland, Association for Computational Linguistics (2011) 116–122
30. Tan, C.M., Wang, Y.F., Lee, C.D.: The use of bigrams to enhance text categorization. Inf. Process. Manage. **38** (2002) 529–546
31. Rogati, M., Yang, Y.: High-performing feature selection for text classification. In: Proceedings of the eleventh international conference on Information and knowledge management. CIKM '02, New York, NY, USA, ACM (2002) 659–661
32. Debole, F., Sebastiani, F.: An analysis of the relative hardness of Reuters-21578 subsets: Research articles. J. Am. Soc. Inf. Sci. Technol. **56** (2005) 584–596
33. Lewis, D.D., Yang, Y., Rose, T.G., Li, F.: Rcv1: A new benchmark collection for text categorization research. J. Mach. Learn. Res. **5** (2004) 361–397
34. Bennett, P.N.: Using asymmetric distributions to improve text classifier probability estimates. In: Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval. SIGIR '03, New York, NY, USA, ACM (2003) 111–118
35. Bennett, P.N., Dumais, S.T., Horvitz, E.: Probabilistic combination of text classifiers using reliability indicators: models and results. In: Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval. SIGIR '02, New York, NY, USA, ACM (2002) 207–214
36. Nigam, K., McCallum, A.K., Thrun, S., Mitchell, T.: Text classification from labeled and unlabeled documents using em. Mach. Learn. **39** (2000) 103–134