Multi-ERSS and ERSS 2004

René Witte, Sabine Bergler, Zhuoyan Li, Michelle Khalifé,
Yunyu Chen, Monia Doandes, and Alina Andreevskaia
The CLaC Laboratory
Department of Computer Science
Concordia University, Montréal, Canada

1 Introduction

We present Multi-ERSS, a further development of last year's entry ERSS [1]. ERSS produced 10-word summaries of newspaper texts based on a knowledge-poor way of computing coreference chains built using fuzzy set theory. That system ranked slightly below average and was run only on one DUC task. ERSS-2004 has been enhanced by a more rigorous use of the fuzzy-theory based reasoning component. This otherwise unchanged system participated in Task 1.

Multi-ERSS is the evolution of ERSS-2004 to produce extraction summaries of multiple documents in a single summary of roughly 100 words. The documents were preclustered according to some topic, which was not known beforehand. Multi-ERSS participated in Task 2.

We used the same analysis techniques and largely the same summarization strategy for all five tasks. Because our technique is based on a knowledge-poor determination of noun phrase coreference, we felt that our system should be impervious to the fact that the text was machine translated for tasks 3 and 4. This was only partially true, some translation particularities did decrease performance (Task 3: average decrease of 0.075, Task 4: 0.03, based on ROUGE-1). We have a hunch that the performance decrease could be compensated in part by adjustments but have not tested to what degree it would be possible.

Task 5 was the most specific task, in that it was known that the topic of the summary was the description of a person. We chose not to build a special system, but instead to rely on our cross-document coreference chains and only adjust the weights of our regular summarization parameters.

2 System Architecture

Our summarization system is a slighty enhanced and improved version of the system we ran last year [1]. It is again implemented within the GATE framework [4, 5]. The GATE framework has been designed for component-based application development: rather than developing a single monolithic NLP application or a set of loosely-coupled subsystems, each GATE component must implement a precisely defined interface. Components, also called *processing resources*, exchange information based on document *annotations*. Their execution is controlled through a *processing pipeline*, which defines the run-time configuration of each component and the order of their execution.

In the following subsections, we describe the main components of our system as they are executed.

2.1 Named Entity Recognition

Before the named entity (NE) recognition, we perform some basic pre-processing steps, including tokenization, gazetteering, sentence splitting, and part-of-speech (POS) tagging. The *Gazetteer* step adds lookup information to individual tokens based on a number of lists: first and last names, company names, dates, countries, titles, abbreviations, and so on.

These annotations, together with their POS tags, are run through a named entity grammar cascade in order to detect a number of different entities: date expression, person and company names, locations, etc. Each of the individual NE transducers is based on a number of JAPE grammars, describing regular expressions over annotations, from which a non-deterministic final state transducer is created by one of the standard GATE components.

Our NE recognition is based heavily on the ANNIE system that comes as an example application with the GATE framework; we simply extended the existing NE grammars and added new JAPE transducers based on our experiments and the results of last year's DUC.

2.2 Noun Phrase Extractor (NPE)

NPE uses a context-free NP grammar and an Earley-type chart parser to extract minimal noun phrases, i.e., NPs without any attachments. It relies heavily on the various named entities (names, dates, and so on) and only falls back to part-of-speech tags if the input tokens have not been marked by any of the NE transducer grammars. This pre-processing of NPs boosts recall and precision compared to chunking all tokens, mainly by removing ambiguities. When compared to manually annotated NPs, we can retrive up to 99% of the marked NPs when scored leniently, that is when marked NPs that overlap with retrieved NPs score as a hit.

In a final step, another JAPE grammar cascade joins these minimal NPs into long NPs by attaching certain grammatical features, like conjunctions, prepositions, appositions, or relative clauses.

2.3 Fuzzy Coreferencer

Fuzzy-ERS groups the NPs extracted by NPE into *coreference chains*, ordered sets of NPs that refer to the same entity. Details on our fuzzy coreferencer and its algorithms can be found in [11] and [10]. Here, we only describe the core idea of the fuzzy resolution algorithm and the enhancements we added compared to last year's system.

The core idea for using a fuzzy-theory based resolution algorithm is the realization that coreference between noun phrases can neither be established nor excluded with absolute certainty. While statistical methods employed in natural language processing already model this *uncertainty* through probabilities, non-statistical methods that have been used so far had no systematic, formal representation for such imperfections. Instead, weights or biases are derived experimentally or through learning algorithms [2]. Here, uncertainty is implicitly and opaquely dealt with in the system and changing it requires rebuilding the system or training set.

Our approach is to examine *explicit* representation and processing models for uncertainty based on fuzzy set theory [12, 7, 3]. There are several advantages in explicitly modelling uncertainty: we do not have to choose arbitrary cut-off points when deciding between "corefering" and "not corefering", like for the semantic distance between words. Instead of such an a priori decision to be lenient or restrictive, we can dynamically decide on certainty thresholds to suit different processing contexts and this value itself can become part of the system deliberations.

As a consequence, we have more information available when building coreference chains, improving overall performance. Moreover, it is now possible to use the same result in different contexts by requesting a specific coreference certainty: a summarizer, for example, can decide to select only coreferences with a high certainty, while a full-text search engine might allow a user to retrieve information based on a more lenient certainty degree.

The output of our coreference algorithm is a set of fuzzy coreference chains, similar to classical resolution systems. Each chain holds all noun phrases that refer to the same conceptual entity. However, unlike for classical, crisp chains, we do not have to reject inconsistent information out of hand, so we can admit a noun phrase as a member of more than one chain, with a varying degree of certainty for each.

Example (Fuzzy Coreference Chain) Figure 1 shows an example for a fuzzy coreference chain. Here, the noun phrases np_3 and np_6 have a very high certainty for belonging to the chain, np_1 only a medium certainty, and the remaining NPs are most likely not chain members.

Fuzzy Coreference Chain $\mathscr{C}$

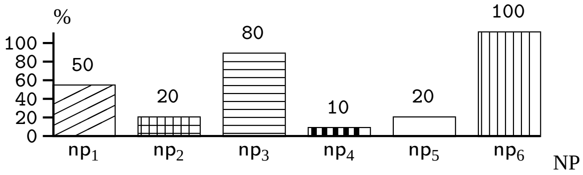

Figure 1: Example for a fuzzy chain showing the membership grades for each noun phrase

Fuzzy Heuristics and Anti-Heuristics: The fuzzy coreference resolution is based on a number of heuristics for establishing coreference, each focusing on a particular linguistic phenomenon. Examples for fuzzy heuristics are pronominal coreference, synonym/hypernym-coreference, or substring coreference. Unlike crisp heuristics' binary decisions, fuzzy heuristics compute a degree of certainty varying between 0 (impossible) and 1 (certain) for a given noun phrase pair. Formally, a fuzzy heuristic $\mathscr{H}_i$ takes as input a noun phrase pair $(\mathrm{np}_j, \mathrm{np}_k)$ and returns a fuzzy set $\mu^{\mathscr{H}_i}_{(\mathrm{np}_j,\mathrm{np}_k)}$ that indicates the certainty of coreference for the noun phrase arguments.

Such a certainty degree can be intuitively determined for almost all heuristics: an example is the synonym/hypernym heuristic, which has been implemented with WordNet [6]. Here, we assume two NPs that are synonyms corefer with the degree *certain*, hence they are assigned a value of 1.0. For hypernyms, our certainty decreases linearly with increasing semantic distance. Heuristics currently in use include:

Synonym/Hypernym the WordNet-based semantic distance heuristic mentioned above;

Substring a simple string comparison, assigning a 1.0 certainty for identical NP strings and a linearly decreasing coreference for substrings depending on their overlap;

Acronym a heuristic comparing NPs with their acronyms and abbreviations;

Apposition a heuristic that checks if two NPs appear inside an apposition;

Pronoun a pronoun resolution algorithm, assigning lower coreference degrees for certain types of gender mismatches without degrading to the *impossible* certainty of 0.0; and

Common Head a comparison of the head noun of two NPs. We currently assign a coreference degree of *likely* (0.8) if two NPs match in their head noun.

Similar to the positive heuristics, *anti-heuristics* compute a degree of certainty between two NPs, but here the degree indicates how certain the two NPs do *not* corefer. The concept of anti-heuristics allows us to encapsulate exceptions to the general heuristics described above, without overloading each of them individually. We currently implemented the following anti-heuristics:

Anti-Synonym/Hypernym also based on WordNet, this anti-heuristic checks if two NPs both appear under certain inner nodes within the hierarchy (including their word sense). This way, we avoid corefering entities that have a small semantic distance within WordNet, but cannot corefer, for example, persons, measurement, or locations.

Anti-Modifier this anti-heuristic examines the *modifier* slot: if an NP is modified by a location or some kind of number (measurement, percentage, value) and the modifier differs, the two NPs cannot corefer.

Currently, the results of the positive and negative heuristics are joined with a simple fuzzy-and operation of the positive and the fuzzy-set complement of the negative result.

Single-document and cluster coreferences: We use the same coreference algorithm and the same heuristics for both single-document and cross-document coreference resolution. For intra-document coreference, only NPs from the same document are compared by the available heuristics. Similarly, to determine inter-document coreference, we only examine NPs from different documents (i.e., never two NPs from the same document). The only difference between the two modes is that for cross-document coreference we do not apply the pronoun, apposition, and synonym/hypernym heuristics.

2.4 Summarizer

Our summarization component works purely extraction-based. A summarization framework allows the development of individual summarization strategies. For each strategy, features are extracted from the document's annotations (for example, the length of a coreference chain), the features are weighted, resulting in a rank for an annotation. Based on this rank, we then extract the selected annotation(s), for example a list of NPs or sentences.

The detailed strategies for each task are described in the following sections.

3 Very Short Summaries of Single Documents

Very short summaries (75 characters) of single documents are required in Task 1 (summarization of single English newspaper articles) as well as Task 3 (summarization of manual and automatic translations from Arabic newspaper articles).

We participated in Task 1 for calibration purposes: how did our changes to the fuzzy set reasoner affect the 10-word summary performance and where does our system stand with respect to this year's texts and participants?

As mentioned, we expected the same system to score similarly in Task 3.

3.1 Summarization Strategy

For this kind of summary, we rank all NPs of a single document by two features: (1) the length of the coreference chain they appear in (NPs appearing in longer chains receive a higher rank) and (2) whether the NP appears within the first two sentences. Both features are equally weighted (1.0, 1.0). For the summary, we then extract the highest-ranking NPs until the length limit has been reached.

Some simple post-processing is performed on the resulting set of NPs to remove determiners and other fill-words, and to remove redundant (overlapping) NPs.

3.2 Performance and Evaluation

ROUGE-1 scores for our very short summaries are summarized in Figure 2. We did not evaluate ROUGE-2 scores in detail, as they are usually too low to give a meaningful indication of a summary's quality (see Figure 3).

	NIST-baseline	**ERSS-2004**
Task 1	0.22	0.2
Task 3 (manual)	0.14	0.255
Task 3 (automatic IBM)	0.14	0.184
Task 3 (automatic ISI)	0.14	0.2

Figure 2: ROUGE-1 scores for 10-word summaries

	NIST-baseline	**ERSS-2004**
Task 1, Priority-1	0.06	0.04
Task 2, Priority-1	0.06	0.06
Task 2, Priority-2	0.06	0.07

Figure 3: ROUGE-2 Scores for Tasks 1 and 2

On Task 1, ERSS-2004 scored just slightly below the baseline (the headlines), whereas in Task 3 it was above the baseline on all data sets: manual, IBM, and ISI translations. Note that the summaries on the ISI data set were not submitted to competition, but obtained post-DUC. The best relative performance was achieved on the manual

IBM: An official source confirmed that the American Secretary of Defense William Cohen arrived in Amman this evening on a visit to continue until tomorrow in the framework of the tour in the region.

ISI: Stressed official source announced that the Minister of Defence American William Cohen arrived in Amman Thursday evening to visit continue until tomorrow under the Uruguay Round by in the region.

Figure 4: Two different machine translations of the same Arabic sentence by IBM and ISI

translations (second place), because the automatic translations produce more adjacent NP patterns, due to Arabic's V S O structure. The IBM translation is chunk to chunk and shows this more strongly, whereas the ISI translation performs a syntax tree transformation that lessens the resulting ambiguity perceived by our chunker (see Figure 4 for an example). On average the number of parsed units dropped by ca. 15%.

4 Short Cross-Document Summaries

This section describes Multi-ERSS, an extension of ERSS-2004 to compute NP coreference chains across documents and to select the most important NPs in the most important chains to extract the sentences for the summary.

4.1 Summarization Strategy

For short summaries (665 characters) we rank the NPs from all documents based on the length of the cross-document coreference chains. We then extract the sentences with the highest-ranking NPs until the length limit has been reached and sort the sentences first by document, then by their order within the documents.

No post-processing was performed on these summaries due to time constraints, although we do have resources in place to remove unreferenced entities, for instance.

4.2 Performance and Evaluation

Multi-ERSS produces summaries of the form given in Figure 5.

We ran Multi-ERSS on both, Task 2 and Task 4. The performance in the DUC competition is reported in Figure 6.

One interesting question is how performance is affected by including or omitting text that is between (any kind of) quotation marks. In Task 2, we calculated the following figures: on average, over all documents, the ROUGE-1 score decreases by 0.012 when including material from sentences that include quotation marks. On average, over documents whose ROUGE-1 score changes, the score decreases by 0.028, which is significant, considering the scale and range of ROUGE-1 scores.

> President Yoweri Museveni insists they will remain there until Ugandan security is guaranteed, despite Congolese President Laurent Kabila's protests that Uganda is backing Congolese rebels attempting to topple him. After a day of fighting, Congolese rebels said Sunday they had entered Kindu, the strategic town and airbase in eastern Congo used by the government to halt their advances. The rebels accuse Kabila of betraying the eight-month rebellion that brought him to power in May 1997 through mismanagement and creating divisions among Congo's 400 tribes. A day after shooting down a jetliner carrying 40 people, rebels clashed with government troops near a strategic airstrip in eastern Congo on Sunday.

Figure 5: Sample summary of cluster d30007 for Multi-ERSS (ROUGE score:0.36, average)

	baseline	**Multi-ERSS**
Task 2	0.32	0.36
Task 4 (manual)	0.33	0.39
Task 4 (autom. IBM)	0.33	0.36
Task 4 (autom. ISI)	0.33	0.36

Figure 6: ROUGE-1 scores for 100-word summaries

In particular, we find:

Cluster	**Ignore Quotes**	**Include Quotes**
d30044t	0.423	0.368
d30001t	0.388	0.416

Cluster d30044t is the more frequent case (3:1), material from sentences with quotation marks is too detailed or specific to be much use for our general summaries. But Cluster d30001t illustrates a case where quoted material was, to the contrary, of importance; in this case single significant words (i.e. words that occur in the model summary) were put in quotation marks (e.g. "internationalize"). On the DUC 2004 corpus, in 23 of 50 cases, however, Multi-ERSS displayed no difference in ROUGE-1 scores under either analysis.

Multi-ERSS does well on clusters 31043, 31009, 30036 and 30040, where most documents within the cluster are in the same style, namely the reporter summarizes in one or two paragraphs at the very beginning. This is textbook newspaper style which we exploit to our advantage by putting more weight on NPs that occur in the first two sentences.

Multi-ERSS performs badly on clusters where the individual articles present different aspects of a common topic. Natural disasters are a case in point (e.g. cluster 30002 on Hurricane Mitch), as are summary topics like the United States Midterm Election (cluster 30050), where we find nine articles reporting on different elections in several different states. These clusters demonstrate the styles of articles for which this summarization strategy is not suited.

Task 4, like Task 3, is based on Arabic to English translations. Again, Multi-ERSS performs better on the better input data from the manual translations, partly due to better NP chunking (see Figure 7 for the results of one cluster).

	ROUGE-1	Parsed Units
IBM	0.363	78.14%
ISI	0.282	88.66%
Manual	0.43	89.51%

Figure 7: Comparison of ROUGE score and parser performance for d1043t

5 Question-Based Cross-Document Summaries

Task 5 asks for a summary from a cluster of texts focused by a question in form of a single named entity.

5.1 Summarization Strategy

We use Multi-ERSS almost unchanged by adding the question NP as another single document to the cluster. The summary is produced by extracting sentences that contain the most important NPs of the chains that corefer with NPs[1] from the question only.

We adopt two strategies to solve Task 5: *simple sentence selection* and *fuzzy coreference clustering*. We also submitted different settings of the IgnoreQuotes[2] parameter for a total of three runs for Task 5:

Run	Strategy	IgnoreQuotes
Priority-1	simple sentence	True
Priority-2	clustering	True
Priority-3	clustering	False

In the Simple Sentence Selection strategy, the chains that include the question NP(s) are first sorted by length (ideally, there is only one chain that included a reference to the question entity). We then select NPs from the chains based on two features: (2) the NP's length and (2) whether the NP appears within an *apposition* construct. *NP Length* has a factor of 1.0, *Apposition* has a factor of 3.0. Apposition is an important text feature for the characterization of persons, since it typically introduces or elaborates on the named entity and thus provides the most useful information for this kind of focused summary. We then extract the sentence the NP belongs to and continue with the next highest-ranking NP. The extracted sentences are sorted by their order within a document and the order of the documents within a cluster, but again no post-processing or smoothing of the summary was performed due to time constraints.

The second strategy relies on NP clustering. The clusters are sorted by size and those that do not contain references to the question are removed. Here too, we rank the NPs in each cluster by (1) NP length and (2) apposition, and select the sentences with the highest ranking NPs.

5.2 Performance and Evaluation

A table correlating our fuzzy merge degree parameter with the different summarization strategies is presented in Figure 8. Here we hold the relative weight of Chain Length and Apposition constant at 1.0 and 3.0.

Run	**1**	**2**	**3**
Fuzzy Merge Degree	0.6	0.8	1.0
NIST baseline	0.31	0.31	0.31
Priority-1	0.33	0.35	0.26
Priority-2	0.33	0.37	0.27
Priority-3	0.33	0.35	0.28

Figure 8: Influence of the fuzzy merge-degree for different strategies.

We also experimented with varying the weights for Chain Length and Apposition for a fixed fuzzy merge-degree for the different strategies. The difference is negligible (most runs score the same: 0.35 compared to a 0.31 baseline) except for the case where Apposition weight is set to 0.0: losing its most discriminative feature, performance uniformly drops.

6 ROUGE

This year's DUC has for the first time mainly automatically scored evaluations called *ROUGE* [8]. ROUGE is fundamentally an n-gram matching scheme between a peer summary and a model summary.

The use of ROUGE has influenced our system development in two ways: a more complex summarization strategy geared towards coherence was entirely abandoned for the competition (mainly because of time pressure) and we attempted only minimal "clean-up" of our extraction-based summaries for coherence. We did, however, not run any form of ROUGE before the competition and did not tailor our summarization strategies for ROUGE in principle.

Our summaries have a consistent flaw: the sequence of sentences in the summary is determined by the order of the source text in the cluster and the order of the sentences within a source document. Temporal coherence in particular is lost. ROUGE does not penalize for this shortcoming.

[1] Note that a named entity recognizer might erroneously split a named entity in two.

[2] Ignore material from sentences that contain quotes, rejecting the sentence and proceeding with the next best-ranking NP.

Our system's performance in the DUC 2004 competition can be summarized as upper third. It is important to note that all submitted systems scored very close to each other, an effect predicted in [9] as a result of the very large space of possible extract based summaries of this length. We also observe that for our system, post-competition tests with different parameter settings show only very small variations. For instance, for Task 2, we tried to find the correlation of summarization strategies, different parameter settings, and different ROUGE algorithms. Figure 9 shows the results.

Strat.	**Fuz.**	**Ign.**		**ROUGE-**		
	Deg.	**Quot.**	**1**	**2**	**4**	**L**
Simple	0.6	T	0.36	0.07	0.008	0.37
Simple	0.6	F	0.35	0.06	0.008	0.36
Cluster	0.6	T	0.35	0.07	0.007	0.36
Cluster	0.6	F	0.34	0.07	0.011	0.35
Simple	0.8	T	0.36	0.08	0.010	0.35
Simple	0.8	F	0.34	0.07	0.010	0.35
Cluster	0.8	T	0.34	0.07	0.010	0.35
Cluster	0.8	F	0.34	0.07	0.001	0.34

Figure 9: Correlation of different summarization strategies, different parameter settings, and ROUGE algorithms. ROUGE-3 is omitted, since results in each row were the same (0.2)

Figure 9 is, however, inconclusive. From other experiments we feel that a merge degree of 0.8 is better and that the simple sentence summarization strategy is currently more mature. But any experimental variation in ROUGE scores is hard to interpret. Although ROUGE-L seems to give us best results in this table, we evaluate using only ROUGE-1.

Another important question is what the correlation between ROUGE-N scores and usefulness is. To this end, we manually assigned usefulness scores to the seven clusters that were ranked best and to the eight clusters that were ranked worst by ROUGE-N, N∈{1,2,3,4}. Usefulness was assessed as a number between 0 and 1, where 0 meant completely useless and 1 "perfect." A usefulness of 0.5 meant an average summary.

Cluster	Description	Usefulness
31043	Lebonan Civil Politic Flurry	0.9
31009	Turkish Civil Politic Flurry	0.75
30036	Nobel Award	0.7
30050	United States Midterm Election	0.4
30007	Congolese Civil War	0.8
30001	Combodian Politic Issue	0.6
30040	Open of Gaza Strip	0.75

Figure 10: Clusters with best ROUGE-1 scores for Task 2.

Figure 10 indicates that for those clusters that get good ROUGE scores, usefulness is also high (with the exception of cluster d30050, discussed above). Cluster d31043, for instance, is ranked among the better results by ROUGE-N (1-4) and it turned out to be one of our best summaries).

Cluster	Description	Usability
31031	the United State's Congress's arguement on annual budget	0.2
30017	North Korea's long lasting famine and circumstances	0.3
30002	Catastrophy from Hurrican Mitch in several contries	0.4
30011	Malaysian Ecconomic and Political Trouble	0.5
31022	Fire disaster in Sweden dance hall	0.2
30007	Congolese Civil War	0.8
30003	Pinochet's arrested	0.4
30053	Cliton's visit to mid east for peace	0.4

Figure 11: Clusters with worst ROUGE-1 scores for Task 2.

Figure 11 shows that those clusters that get the worst ROUGE-N scores have low usefulness scores in general. A notable exception is cluster 31022, which has a high usefulness score even though his ROUGE score is low (note that another run of the system places the same cluster as one of our best summaries. The two summaries are, in fact, barely distinguishable to a human assessor. This case is interesting, because it illustrates that ROUGE cannot predict usefulness in general. A drastically different summary, not extract-based, but crafted with terms that allow for high compression would be penalized as compared to our barely coherent summaries, because the n-gram overlap is not there. Our 10-word summaries for DUC-2003, for instance, included a classification in one of the standard newspaper article categories (Politics, Science, Business, . . .). This "wasted" valuable words that didn't contain text extracts, but the manually assigned usefulness scores seemed to indicate that the human assessors felt that loss was well-compensated by the classification. A ROUGE score cannot assign value to such extra-textual material, which humans on the other hand value.

Figure 12 shows almost 100% increase in the ROUGE score of summaries that are barely more informative. It illustrates how ROUGE scoring can limit future research and system development, and serves to streamline summary style by penalizing unconventionally presented summaries. Given that the submissions for DUC 2004 scored already so close to each other, unreflected use of ROUGE in lieu of more careful evaluation could be stifling to the

Document APW19981016.0240 in d30001t:

Summary with classification (ROUGE score: 0.28): People & Politics: country's next president; only other army commander; Syr

Summary without header (ROUGE score: 0.49): country's next president; only other army commander; Syria; Lebanon; politi

Document APW19981221.0719 in d31050t:

Summary with classification (ROUGE score: 0.23): People & Politics: Fischer; authorities; China's sentencing; Wei Jingsheng

Summary without header (ROUGE score: 0.50): China's sentencing; German Foreign Minister Joschka Fischer; drew China's a

Figure 12: Effect of extra-textual material on ROUGE-1 scores.

field. It does, however, provide a powerful tool for large scale, automatic assessment for both, development and comparison purposes, and will become a standard in the field.

7 Preliminary Conclusion

While we are still evaluating our system and its evaluation results, we can sum up our DUC 2004 experience as follows. The knowledge-poor, coreference-based summarization strategy using fuzzy set theory has been successfully extended to 100-word (665-character) multi-document summaries. Our systems score, in fact, in the upper third of this year's competition.

Obvious next steps are to tailor Multi-ERSS to Task 5, to develop a summarization strategy that deals better with clusters of documents with very little overlap and to use text classification to field the text to one strategy or the other. But this is tinkering, interesting from the perspective of the CL researcher, but results will remain frustrating to the human user.

We propose to further develop the targeted summary idea and try to address the issue of usefulness while avoiding the variations in human assessments of quality. We will have to experiment with secondary performance measures of both automated systems and human summary "consumers" on standardized tasks, working under the assumption that a better summary will result in better secondary performance.

8 Acknowledgments

Our thanks to Jennifer Scott, whose annotations and evaluations are invaluable.

This work was funded in part by an NSERC discovery grant and by a NATEQ nouveau chercheur, volet équipe grant.

References

[1] Sabine Bergler, René Witte, Michelle Khalife, Zhuoyan Li, and Frank Rudzicz. Using Knowledge-poor Coreference Resolution for Text Summarization. In *Workshop on Text Summarization*, Document Understanding Conference (DUC), Edmonton, Canada, May 31–June 1 2003. NIST. `http://duc.nist.gov/`.

[2] Claire Cardie and Kiri Wagstaff. Noun phrase coreference as clustering. In *Proceedings of the Joint Conference on Empirical Methods in NLP and Very Large Corpora*, Maryland, 1999.

[3] Earl Cox. *The Fuzzy Systems Handbook*. AP Professional, 2nd edition, 1999.

[4] H. Cunningham. GATE, a General Architecture for Text Engineering. *Computers and the Humanities*, 36:223–254, 2002. `http://gate.ac.uk`.

[5] H. Cunningham, D. Maynard, K. Bontcheva, and V. Tablan. GATE: A framework and graphical development environment for robust NLP tools and applications. In *Proceedings of the 40th Anniversary Meeting of the Association for Computational Linguistics*, 2002.

[6] Christiane Fellbaum, editor. *WordNet: An Electronic Lexical Database*. MIT Press, 1998.

[7] George J. Klir and Tina A. Folger. *Fuzzy Sets, Uncertainty, and Information*. Prentice-Hall, 1988.

[8] Chin-Yew Lin and E.H. Hovy. Automatic evaluation of summaries using n-gram co-occurrence statistics. In *Proceedings of the 2003 Human Language Technology Conference HLT-NAACL 2003*, Edmonton, Canada, May 27 – June 1 2003.

[9] Chin-Yew Lin and E.H. Hovy. The potential and limitations of sentence extraction for summarization. In *Proceedings of the Workshop on Automatic Summarization, DUC-2003, post-conference workshop of HLT-NAACL-2003*, Edmonton, Canada, May 31 – June 1 2003.

[10] René Witte. *Architektur von Fuzzy-Informationssystemen*. BoD, 2002. ISBN 3-8311-4149-5.

[11] René Witte and Sabine Bergler. Fuzzy Coreference Resolution for Summarization. In *Proceedings of 2003 International Symposium on Reference Resolution and Its Applications to Question Answering and Summarization (ARQAS)*, pages 43–50, Venice, Italy, June 23–24 2003. Università Ca' Foscari. `http://rene-witte.net`.

[12] L.A. Zadeh. Fuzzy sets. In R.R. Yager, S. Ovchinnikov, R.M. Tong, and H.T. Nguyen, editors, *Fuzzy Sets and Applications: Selected Papers by L.A. Zadeh*, pages 29–44. Wiley&Sons, 1987. Originally published in *Information and Control*, Vol. 8, New York: Academic Press, 1965, pages 338–353.