Part V

Conclusions

22 Conclusion and Outlook

22.1 Discussion of Results

The mistake analysis described in the previous chapter has confirmed several of our assumptions, but it has also brought some new insights. We have always assumed that the suitability of texts and attributes to extract is essential for the results we might expect of information extraction systems (cf. Sec. 8.3). The analysis confirms this, but at the same time it refutes some of our conjectures about *which* texts are suitable.

For example, we had assumed that usage of a regular and standardized language would improve extraction quality. However, actually the issue seems to be more complex. On the one hand, in the Seminar corpus we see cases of informal expressions that cause mistakes since the system lacks sufficient similar examples to learn them. But, on the other hand, the language of the newspaper articles forming the Acquisitions corpus appears to be *too* standardized for the system. Here, many phrases are used that are so generic that they do not offer the system sufficient clues to differ between the various roles of companies. In this case, the very regularity of the language becomes a problem for the algorithm.

This is a surprising and important finding—while it confirms that extraction quality depends on the kinds of text to process, it corrects some ostensible hypotheses about what makes texts more or less suitable. Different from what we would have expected, regularity and standardization of the used language is not necessarily an advantage.

This confirms a vital but easily forgotten lesson: it is not enough to form any conjectures, assumptions, or theories—the essential issue is to test them. Another issue where one of our conjectures did not stand the test of reality concerned the type confusions that occurred in the Acquisitions corpus. It had seemed reasonable to us that the differentiation between long and short versions of the same company name would be the main source of confusions among company names, but actually we found *logical* confusions about the roles of companies involved in acquisitions to the major source of confusions. Had we tested this *prior to* evaluating the weakly hierarchical mode (cf. Chap. 20), we could have prepared a better experiment or else saved us the trouble.

An interesting result of the analysis is that many of the mistakes made by our system appear to be almost "intelligent"; for example, if persons involved in organizing or announcing a talk are proposed as SPEAKERs or if the various kinds of companies mentioned in a vague context are confused. In such cases it is imaginable that a human user quickly skimming the text would make similar mistakes, and in some cases it might even be argued whether the attribute values proposed by the system are not quite reasonable.

Of course, there is no "intelligence" in a trainable algorithm such as ours, but sometimes the acquired statistical patterns seem to be not too far away. This glaringly remains us that we will never be able to expect the predictions of an information extraction system to be error-free: wherever true intelligence or understanding would be required, any automatic system will have to fail.

Another issue are the tagging inconsistencies and seemingly forgotten answer keys we occasionally noted. While for a fair comparison with other systems it is essential to treat the answer keys given in a corpus as "gold standard" that is not to be judged or modified, such inconsistencies make a task clearly more difficult (and in a wholly pointless way), since the system has not only to learn to characteristics of the attribute in question but it *also* needs to any learn any apparent patterns influencing the inconsistent tagging (a task that would generally be impossible for humans too—especially if there are no such patterns).

This issue becomes even more problematic by the fact that some other authors appear to have corrected inconsistencies in the used corpora, sometimes without pointing this clearly out. Providing well annotated and consistent test corpora is obviously an important but also a difficult task. Interestingly, this points to another application scenario where the interactive incremental training setup we have proposed can be helpful: since the predictions made by the system during interactive annotation should be largely consistent with the previous annotations on which they are based, interactive annotation based on incremental training is likely to achieve a higher tagging consistency compared to purely manual annotation.

22.2 Summary of Contributions

In Part I, we gave an overview of the current state-of-the-art in information extraction, after introducing the field of *information extraction* and contrasting it with related areas such as *text understanding* and *information retrieval* (Chap. 2). We also discussed the architecture of typical IE systems and the tasks that a *comprehensive* approach to IE would need to handle (Chap. 3).

After comparing the features and characteristics of the presented IE approaches (Chap. 6), we defined in Part II the aims and requirements that our approach should fulfill (Chap. 7). Our guiding principle was to explore issues and investigate problems that before had been neglected, while still preserving the best practices and promising techniques from current approaches. We discussed the assumptions and conjectures motivating why our approach should be tailored for incremental training *("Systems will be used")* and why it should take the document structure of input texts into account *("Structure matters")*; and we treated the common assumptions that tend to be shared by IE approaches in general but are seldom spelled out explicitly (Chap. 8). We further framed the context conditions for our approach by analyzing which target schemas it should be able to handle and which input and output models it should support to comply with the desired requirements (Chap. 9).

Part III described and Part IV evaluated the approach that we designed and implemented to fulfill the defined aims and requirements—we will resume both parts

together due to the interdependencies between models and their evaluations. In Chapter 10 we explicated how information extraction can be modeled as a series of token classification tasks. We introduced the concept of *tagging strategies* that are necessary to translate between logical states and class labels; we identified and characterized the tagging strategies that can be found in the literature, also introducing a new strategy, the so-called *BIA* (or *Begin/After*) tagging. A comparison of the different strategies is contained in Chapter 19—we concluded that the choice of a tagging strategy, while not crucial, should not be neglected when implementing a statistical IE system. The popular *IOB2* strategy which we had chosen as default strategy was found to be the best of all established tagging strategies, closely rivaled by the new *BIA* strategy.

In Chapter 11 we introduced our choice of the second core component of classification-based IE, the classification algorithm to use. Since incremental training is a major concern of our work, we chose the online learning algorithm Winnow as default classifier. Our Winnow implementation uses a sparse architecture which makes it specifically suitable if the overall number of features is very high and a thick threshold training setup which makes it a large margin classifier somewhat similar to SVMs. We introduced the feature combination techniques SBPH and OSB as ways to enrich the feature space for the (linear) classifier, allowing it to learn the relevance not just of isolated features, but also of combinations of related features.

Since in classification-based information extraction the used classification algorithm is only one of several factors influencing the results, we also evaluated this classifier setup for text classification to get a better impression of its performance and to optimize parameters (Chap. 16). We found the results reached by the combination of Winnow and the novel OSB combination technique to be excellent, making it one of the best (if not the best, according to the logistic average misclassification rate and the medium area of the ROC curve) classifiers participating in the 2005 Spam Filtering Task of the renowned *Text REtrieval Conference (TREC)*.

In Chapter 12 we introduced the third and final core component of our token classification approach: the context representations we generate as feature vectors to allow the classifier to learn the features relevant for distinguishing tokens of different classes from among a rich and expressive choice of features. We also covered the preprocessing and tokenization steps that are necessary to prepare the input and to generate suitable tokens. For preprocessing, we convert documents into a DOM (XML) tree structure that unifies both document structure (paragraphs, lists, emphasized blocks, etc.) and linguistic structure (sentences, sentence constituents, etc.). This makes it possible to generate context representations on the basis of *inverted subtrees* of the whole document tree that use the leaf node containing the token to classify as new root and extend from there, covering an overall context that is far larger than the flat context window (usually just the token itself and some tokens/word to its right and left) considered by other IE algorithms.

Preparing the input documents in this way requires the unification of various and partially conflicting sources of information (such as structural markup and linguistic annotations) in a single DOM tree structure. For this purpose, we developed a merging algorithm that can repair nesting errors and related problems in XML-like input

(Chap. 13).

After thus presenting the concept and the components of our approach and after introducing the metrics and methodology for evaluation (Chap 15), we were ready to evaluate the overall performance of the approach in Chapter 17. For evaluation, we used two of the most popular information extraction corpora, the *CMU Seminar Announcements Corpus* and the *Corporate Acquisitions Corpus*. For the Acquisitions corpus our system reached the best of all results known to us, for the Seminar corpus it was among the best two (while additionally allowing incremental training). We noted that our system is especially biased towards reaching a high precision, and that in general results tend to depend more strongly on the kinds of attributes and texts to handle than on the specifics of the used system.

In Chapter 18 we studied the influence of the various sources of information that we include in our rich context representations. This ablation study confirmed that all the investigated sources of information contributed to the good results reached by our system, though the semantic sources we had used tended to be of little importance. It also confirmed the *"Structure matters"* conjecture which had motivated us to consider information regarding document structure, in spite of the fact that we had to rely on a heuristic recognition of the implicit "ASCII markup" contained in the corpora since conventional IE corpora such as the ones we were using do not contain any explicit structural markup (and we did not have the resources to prepare new ones). We also investigated the utility of interactive incremental training and found that it can reduce the human effort for providing training data and for correcting predictions in a substantial way, justifying our decision to introduce incremental trainability into the field of IE.

In Chapter 14 we investigated approaches of making supertype/subtype relations between attributes fruitful for information extraction. We discussed the idea of a *strictly hierarchical approach* and why it would hardly work due to the problems regarding error propagation and differences in the used corpora, annotation styles, and semantics. We also proposed a *weakly hierarchical approach* as a less fragile alternative. This attempt, however, was the one part of our work where the reached results were largely disappointing—evaluation on the two test corpora (Chap. 20) showed very mixed results. We concluded that this approach might make sense in some cases if suitable information regarding "loose" supertypes is available and can easily be integrated but that it is hardly recommendable as a general-purpose solution.

We completed the evaluation of our system with a mistake analysis (Chap. 21) to learn more about the nature and the (likely) causes of the mistakes that occur. The conclusions that can be drawn from this analysis have been discussed in the preceding section.

22.3 Future Work

Throughout this work we have already mentioned various issues that could be investigated as future work. To resume them quickly:

- The IE system introduced in this thesis is designed as a generic framework for classification-based information extraction that allows modifying and exchanging all core components (classification algorithm, context representations, tagging strategies) independently of each other. We have already performed a systematic analysis of switching the tagging strategies, but for the other components this remains as future work.
- We have consciously refrained from performing extensive parameter variation tests. Optimizing the various parameters influencing the Winnow classifier and the extend and coverage of the generated context representations (parameters controlling the size and details of the considered context, the list of semantic sources used, etc.) has been left as future work, especially since such exact tunings will often be corpus-specific.
- While our attempt to utilize inheritance hierarchies between attributes as another source of information for improving extraction quality turned out to be of very limited success, we still believe that identifying and making accessible additional sources of information is a relevant area of future research.

These future research directions are within the scope of work that we have addressed in this thesis: the extraction of explicit information where the suitability of attributes to extract and of texts to extract from is not to be judged. While this is certainly a very important research area that will remain of core importance for future information extraction approaches, we believe that the most important issues for work which still needs to be done will lie *outside* this area which by now has been covered fairly well.

These more important challenges for future work, as we see them, will lie in two areas:

Suitability of Tasks

As noted above, the results reached by various systems appear to depend more strongly on the kinds of attributes and texts to handle than on the used approach; and, as we learned in the mistake analysis, sometimes in non-obvious and surprising ways. Gaining a better insight into the characteristics that make tasks more or less suitable for automatic extraction will probably be more important for the future practical advancement of information extraction than further improvements in extraction quality which will generally only be gradual.

We had noted before (Sec. 17.3) that there are at least three general factors on which the suitability of a task depends: the amount of training data available, the characteristics of the attributes to extract, and the characteristics of the texts to process. Future research should try to acquire more detailed knowledge of these (and, if relevant, other) factors, defining more exact qualitative and, as far as possible, quantitative criteria about the requirements that tasks need to fulfill to be suitable candidates for automatic information extraction.

Such a deeper suitability model would ideally make it possible to estimate what magnitude of results to expect for a certain corpus *without having to annotate many sample texts*, since for trainable systems the annotation of training data remains the most serious burden, and even incremental trainability can only lower, but not remove

this burden. Of course, with regard to the characteristics of attributes to extract this ideal is somewhat paradoxical—how should these characteristics be measured if no sample attribute values are known? In some cases this paradox might be resolved due to the fact that the typical purpose of a comprehensive IE system will be to populate a database from text documents: if the target database already exists and contains sample values of the attributes to extract gained in some other ways, these sample values might provide sufficient data to find out whether a task is promising for automatic handling, obviating the need to provide annotate sample texts *before* this is known.

Comprehensive Approach for Text-to-Database Integration

Which brings us to the second issue, namely, that we have handled only one of the various tasks that need to be addressed to get a *comprehensive* information extraction system. We believe that the most important challenge for future IE systems is to move beyond this one step, the extraction of explicit information (where gradual further improvements will certainly occur, but where we suppose substantial new breakthroughs to be unlikely), and to cover more of the other steps sketched in Section 3.1.

The *text filtering* and *extraction of implicit information* steps might be suitable candidates for text classification so they could be easily integrated into our classification-based approach (if they are required it all, which will depend on the task). Later steps such as *value normalization* and *relationship resolution*, however, are of a different nature and will require other ways of handling them, but they are essential to support more complex relational target schemas where attributes are strictly typed and where several relations with dependencies between them exist. Value normalization will often be a largely attribute-specific process that requires specialized rules and heuristics for dates, person names, geographic entities etc. Relationship resolution can be handled in a more general way, but how to extend such approaches as developed, for example, by [Rot02] from binary relations to general n-ary relations is an issue that will need to be addressed.

Instance unification would allow identifying and merging complementary or conflicting pieces of information about a real-world entity from different texts of different parts of a text. While there are numerous papers about this problem (often referred to as *record linkage*) in the general database context, the interesting question for information extraction would be whether the additional textual context provided by the texts from which attribute values have been extracted could be made fruitful for improving the quality of unifications.

Recently, IBM has presented the *UIMA* (Unstructured Information Management Architecture) framework [Per04] as an architecture for extracting information from unstructured sources which points in the direction we have in mind. So far, however, this architecture is focused on preprocessing and fragment extraction, more complex tasks such as relationship resolution have not (yet?) been addressed; the same is true of earlier initiatives such as *GATE* [Cun02] and *ATLAS* [Lap02]. Creating a more comprehensive approach for text-to-database integration remains an open issue.