Part IV

Evaluation

15 Evaluation Goals and Metrics

15.1 Goals and Limitations of Quantitative Evaluation

Quantitative evaluation can serve several goals:

1. Figuring out whether and how useful a proposed approach or a proposed enhancement is, by comparing results of the approach with the results of some baseline. In case of enhancements, the baseline for comparisons is the original approach without the proposed enhancement; in other cases a suitable baseline might be more difficult to determine.
2. Comparing several approaches or several variants of the same approach to find out which of them is better suited for the tested setting.
3. Finding out which configuration of an approach is better suited for a setting, by comparing different configurations ("parameter optimization").

The evaluation done in the following chapters serves the first two goals. As stated while defining the scope of this thesis (Sec. 7.4), we are not interested in detailed parameter optimization and performance tuning studies, leaving such optimizations as future work.

A general caveat of quantitative evaluation is that we can only evaluate specific settings, i.e. specific information extraction corpora and setups. We cannot know for certain whether and in which degree the results are transferable to other tasks and settings. For evaluating the extraction of attribute values, the "classical" IE task, we will use two of the most frequently used standard IE corpora that represent two different kinds of tests. The *CMU Seminar Announcements* corpus comprises newsgroups messages with some partially structured elements such as headers and written in an informal, partially ungrammatical language which is typical for e-mail messages and other kinds of day-to-day "ad-hoc" communication; the *Corporate Acquisitions* corpus, on the other hand, comprises newspaper articles written in a formal and strictly grammatical style. This allows drawing conclusions about these specific corpora; comparing relative performance also allows some insight into how well algorithms can cope with these different types of texts but such general reflections always need to be taken with a certain caution.

Also, frequently, there can be some doubt about which of the results reached by an algorithm actually are correct and which are wrong. The answer keys provided for evaluating IE systems (as well as those provided for training) are the results of human annotation of input texts. Human annotators will almost inevitable make occasional errors by overlooking some answer keys or misplacing the borders of answer keys. Aside from obvious errors (which an annotator her/himself would admit to be erroneous if the problem was pointed out to her/him), there is a considerable "gray area" where

annotators might come to different conclusions about which exact text fragments should be labeled as answer keys and which should not.

Inter-annotator agreement (IAA) can be considered some kind of "top line" (upper bound) for the system performance we can expect, since it is unlikely that quality of extractions performed by an algorithm will ever surpass those done done by humans. In bioinformatical extraction tasks, inter-annotator agreement has been found to reach values from about 70% to 90%, depending on the type of entity to extract,[1] but for other application areas such studies are still rare.[2]

15.2 Evaluation Methodology

As discussed in [Lav04a, Lav04b], there are several issues that need to be addressed to allow a fair comparison of different systems, some of which have often been neglected in previous IE evaluations. An important issue is the size of the split between training and testing set (e.g. 50/50 or 80/20 split) and the procedure used to determine partitions (n-fold cross-validation or n random splits).

Another issue is how to compare predicted answers (attribute values) with the expected (true) answers. Typical options are to require that all occurrences of an attribute in a document should be found (*"one answer per occurrence"* or *"match-all"*) or to expect only a single answer per attribute which is considered most likely to be correct (*"one answer per attribute"* or *"match-best"*).

The latter option ("match-best") is useful if multiple answers for the same attribute are expected to be synonymous (e.g. "2pm" and "2:00 pm"). Regarding relational target schemas, is corresponds to the *text-as-tuple* scenario where there is only a single relation (with any number of attributes) and each text corresponds to at most one tuple in this relation (cf. Sec. 9.1). The former option ("match-all") makes sense if each occurrence is assumed to contain relevant new information; it corresponds to the *single-attribute relations* scenario where several independent single-attribute relations exists.

A less frequently used option would be "one answer per different string" where multiple occurrences of the same string are collapsed into a single occurrence, i.e. different positions in the document are ignored.

To determine the input values for the evaluation metrics that will be presented in the next section, we compare the extractions proposed by the system with the predefined answer keys ("gold standard") to determine their evaluation status. Possible status values are:

true positive: *correct* predictions, i.e. predictions matched by an answer key.

false positive: *spurious* predictions (no corresponding answer key).

[1] [Col05] report 87% IAA (accuracy) for Fly genes, 91% for Yeast genes and 69% for Mouse genes; [Man05] report an average IAA (F-measure) of 71% for protein names; [Dem02] report an average IAA precision of 86% and IAA recall of 92% for terminology recognition (= extraction of attribute values).

[2] Peter Siniakov and Heinz Schweppe are currently attending a bachelor thesis on this topic, but results are not yet in at the time of writing.

false negative: *missing* answer keys (no corresponding prediction).

In *match-best* mode, two additional status values occur:

ignored: for predictions that have been ignored. Since in this mode there is only a single instance of each attribute to predict, we choose the most probably prediction (as per the probability estimates returned by the classifier) of each attribute for evaluation (so it will be evaluated as either **true positive** or **false positive**, depending on whether or not a matching answer key is found). All other predictions are marked as **ignored**.

alternative: for answer keys that could have been proposed as predictions but were not. In this mode, the proposed (most likely) prediction should match one of the answer keys. Either the selected prediction matches and is evaluated as a **true positive**; or there is no selected prediction or it does not match, in which case one of the answer keys (if there are any) is marked as **false negative**. Any further answer keys are marked as **alternative** since they are irrelevant for calculating evaluation metrics.

15.3 Evaluation Metrics

The most commonly used metrics for quantitative evaluation of IE systems are *precision* and *recall*; the joint *F-measure* combines them both in a single figure. For each attribute, results are evaluated by counting *true positives tp* (correct attribute values), *false positives fp* (spurious attribute values), *false negatives fn* (missing attribute values) and calculating

$$\textit{precision } P = \frac{tp}{tp + fp}$$

and

$$\textit{recall } R = \frac{tp}{tp + fn}.$$

The *F-measure* is the harmonic mean of precision and recall:

$$F = \frac{2 \times P \times R}{P + R}.$$

Only exact matches are accepted as *true positives*; partial matches are counted as errors (a partial match between a prediction and an answer key will always result in a false positive *and* a false negative).

In approaches modeling information extraction as a token classification task (cf. Chap. 10), it would theoretically be possible to use the raw token classification accuracy as an evaluation metrics. However, the *P/R/F* metrics focusing the correct extraction of complete attribute values are more interesting since they measure directly the goal of IE—a higher token classification accuracy will not be of any use if information extraction performance suffers. Also, accuracy measurements would be of little interest due to the very unbalanced class distribution among tokens. In the *Seminar Announcements* corpus (cf. Sec. 17.1), our tokenization schema yields 139,021 tokens, only 9820 of which are part of slot fillers. Thus most strategies could already reach an accuracy of 93% by always predicting the O class.

For a corpus containing multiple attributes, there are several ways to combine results of all attributes into a single measure. The *microaverage* is calculated by summing the respective *tp*, *fp* and *fn* counts for all attributes and then calculating P, R, and F over the summed counts. Thus attributes that occur more frequently have a higher impact on the joint measure than rare attributes. On the other hand, the *macroaverage* is calculated by computing the mean of all attribute-specific P and R values, so all attributes are considered of equal importance, no matter how often they occur.

A disadvantage of the *microaverage* is that is depends on knowing the raw counts, which are hardly ever published in research papers. This is addressed by a related metric, the *weighted average* proposed by [Chi02]: here each attribute is weighted by the total number of answer keys (expected attribute values) of this attribute in the corpus. These numbers can be determined by inspecting a corpus, allowing comparisons with other systems evaluated on the same corpus even if no raw counts have been published.