14 Weakly Hierarchical Extraction

14.1 Introduction

Independently of the approach chosen to recognize attribute values, IE systems generally use various *sources of information* (features) to decide which text fragments to extract. Some, such as [Sch01], limit themselves to the words (tokens) contained in a text, but most systems additionally use some kind of linguistic features, relying, e.g., on POS (part-of-speech) tagging or partial parsing (chunk parsing). Other typical sources of information are semantic information about the words in a text (gazetteers/word lists, occasionally thesauri such as WordNet [Fel98]) and features derived from the shape of words (token types, prefixes, suffixes). Less frequent is the use of structural information such as HTML tags (e.g., by *wrapper induction* approaches, cf. Sec. 5.3) or the partial DOM trees of XML documents used in our approach.

Usually new sources of information are introduced with the aim of improving extraction quality, and searching additional sources that can be utilized is one way of advancing the field of IE. In this chapter, we will start to explore a novel source of information that is familiar to everyone in computer science but so far has not been used for information extraction (to our knowledge): *inheritance hierarchies*, i.e., supertype/subtype relations between attributes.

So far, this problem has been left largely unattended in the context of statistical information extraction. [Sut05] have used a cascade of linear-chain Conditional Random Fields (CRFs, cf. Sec. 4.3) to jointly learn models for recognizing names and nominals (e.g. "the nation") of persons, organizations, and geopolitical entities (such as countries and cities) from a named-entity corpus (the *Automatic Content Extraction (ACE)* corpus), and the usual attributes (SPEAKER, LOCATION, START TIME, END TIME) from the *Seminar Announcements* corpus (cf. Sec. 17.1).

Each CRF model is trained independently on the training set of its respective corpus, but for application/evaluation they are combined into a single joint decoding model. For decoding, weights learned by individual CRFs are combined into a factorial CRF which makes predictions for all tasks at once. This is an interesting approach, but it is limited to the specific models they use (CRFs) and cannot be generalized to approaches based on other models which do not allow combining separately trained models into a joint prediction model.[1]

[1] It is also worth pointing out that the results reached by our approach on the *Seminar Announcements* corpus *without* named-entity information generally surpass those of [Sut05] *with* named-entity information (cf. Sec. 17.2—their approach is referred to as "CRF" in Table 17.2 and Fig. 17.3). Hence, simply switching over to their approach to make use of such additional information would not be reasonable.

In the Semantic Web area, some authors, e.g. [Mae01, Blo05], have tackled the problem of learning (acquiring) ontologies, i.e., concept hierarchies similar to (but more complex than) the type hierarchies we will explore. Ontologies learned in such a way could be used as type hierarchies in our setup, but it would still be necessary to provide training instances for all relevant types, e.g. by manual annotation.

In the following sections, we discuss ways of exploiting inheritance hierarchies for information extraction, and the problems that occur in this context. The experimental setup used to test the approach and the results of the evaluation will be described in Chap. 20.

14.2 Inheritance Hierarchies of Attributes

Most existing IE tasks[2] comprise a list of clearly separated attributes without any implicit or explicit inheritance relations between different types. However, there is often a logical *inheritance hierarchy* between domain-specific attributes and generic *named* or *numeric entity* types.

Named entity (NE) recognition is a task that is closely related to information extraction. The aim of NE recognition is to locate named entities (names of persons, organizations/companies, locations, ...) and numeric entities (monetary amounts, percentages, dates, times, ...). NE recognition can thus be considered a special branch of IE where the types of information to extract are generic entities instead of domain-specific entities. Generally, any trainable IE system could be employed for NE recognition, while specialized NE recognizers might contain specific heuristics that prohibit their adaptation to generic IE.

For example, the SPEAKER type of a *Seminar Announcement* must be filled with a PERSON, while START TIME and END TIME are subclasses of TIME. Theoretically, there is no need to limit such inheritance hierarchies to two levels, e.g., several kinds of SPEAKERs could be distinguished in a conference program: INVITEDSPEAKERs (invited by the program committee), RESEARCHSPEAKERs (whose papers got accepted for presentation) and REPRESENTATIVESPEAKERs (representatives of city, university etc. inaugurating the conference).

The applicability of such a model depends on the existence of suitable supertypes. The supertypes in such an inheritance hierarchy need not necessarily be classic named or numeric entities, any suitable types will do as long as training data for them is available. For some IE tasks focusing on the extraction of nontraditional entities such as products, medicines, laws, this will make the approach inapplicable due the lack of possible supertypes or due to the lack of training data for supertypes.

14.3 Strictly Hierarchical Approach and Related Problems

So far, statistical IE approaches tend to be flat, they only consider a single level of attributes without taking hierarchical dependencies into account. One possible way to

[2] Such as those available in the *RISE Repository* ([RISa]).

consider inheritance hierarchies would be a **strictly hierarchical (SH) approach**:

In an initial step, only the top-most types (those without a superclass, i.e., typically named and numeric entities) of attribute values are extracted in the usual way. In further steps, the classification is iteratively refined, determining for each found fragment whether it belongs to one of the direct subclasses of the original type (e.g., whether a TIME is a START TIME or an END TIME or neither).

An advantage of this approach is that it reduces the workload of the system—processing full texts is only necessary for a limited number of top-level types; for subtypes, only attribute values of the supertype must be considered.

However, there are serious problems with such an approach. For one, the *problem of error propagation:* most errors of the top-most classifier cannot be corrected later, since subsequent steps rely on attribute values identified in the first step. This means that false negatives (missing attribute values) and misplaced borders of a top-level type will propagate as errors through all subtypes; only false positives (spurious attribute values) stand a chance of being corrected (by being classified as OTHER in a later step).

This is the especially serious because of the *different corpora problem:* generic top-level entities (i.e., named and numeric entities) are usually *not* marked in domain-specific target corpora (e.g., Seminar Announcements). Thus the top-level entities must be trained on another corpus, e.g., a generic NE corpus. Using different corpora for training and for extraction will generally increase the error rate since the resulting recognizer is better adjusted to the training corpus. This makes the fact that top-level errors cannot be corrected in later steps even worse.

An associated problem is that *annotation styles or semantics might be different.* The annotation guidelines used for the preparation of different corpora might have been different, e.g., PERSON names might be tagged without preceding titles in a generic NE corpus, while a domain-specific Seminar Announcements corpus might require the inclusion of titles into the names of SPEAKERs. In this case, the SH approach would have no chance of extracting a SPEAKER name such as **"Professor Iris Young"** correctly, since even in the best case, the higher-level PERSON recognizer will identify the name without the preceding "Professor", leaving no chance of correction in the later step.

The situation is even worse if the *semantics* of related types are different. For example, the LOCATION of a seminar might appear to be a subtype of the LOCATION type that is a typical constituent of named entity corpora. However, LOCATIONs in named entity corpora comprise geographic entities such as countries, cities, or streets, but the LOCATION of a seminar typically identifies a room, possible (but not necessarily) giving additional details on building, street address, or university. Thus, while named entity LOCATIONs might be part of a seminar LOCATION, full seminar LOCATIONs will almost certainly *not* be identified as LOCATIONs by the named entity model.

14.4 Weakly Hierarchical Approach

To address these problems, we propose a **weakly hierarchical (WH) approach** as a less fragile alternative: again, there is one step for each level of types in the inheritance

hierarchy, meaning that top-level types (root types) are recognized in the first step, second-level types in the second step and so on. However, in all steps, extractions are possible from the complete texts—"subtype" recognizers are *not* limited to attribute values found by the corresponding "supertype" recognizer. Instead, the "supertype" annotations derived in prior steps are added as additional *features* for the subtype recognizers. Information about supposed "supertype" attribute values is thus available for locating "subtype" attribute values, but "subtype" recognizers are not forced to honor this information (because of which it is more appropriate to speak about *loose super/subtypes* or to use quotes when using these terms). If a classification-based approach to IE is used (as in this thesis), this means that the trainable classifiers will automatically determine whether and how to use this information while building their classification models.

However, if this is done *indiscriminately*, there is a risk of adding too much noise for "subtype" recognition, which might negatively affect extraction quality (esp. classification-based approaches tend to be susceptible to noise). This can be addressed by *discriminatively* making information regarding each loose supertype available only to the corresponding subtype recognizers. For example, PERSON features are added to the information used by the SPEAKER recognizer, but are invisible for recognizers of types that are not PERSON "subtypes".

14.5 Integration into Information Extraction Approach

For the experiments reported in Chap. 20, we have used our system in the default setting with the *IOB2* strategy.

The utilized context representations (cf. Sec. 12.2) are augmented by Information on whether tokens belong to the loose supertype (in *indiscriminate* mode) or to any of the attributes recognized by the higher-level classifiers (in *discriminate* mode) is included as additional semantic information. The fact that multiple binary Winnow classifiers are combined in a "one-against-the-rest" setup (as described in Sec. 11.1) makes the discriminate variant of the weakly hierarchical approach possible, by providing appropriate information about each "supertype" to the binary classifiers for the corresponding "subtypes" only.