Resources and Techniques for Multilingual Information Extraction

Stephan Busemann and Hans-Ulrich Krieger

German Research Center for Artificial Intelligence (DFKI GmbH)
Stuhlsatzenhausweg 3, D-66123 Saarbrücken
{busemann|krieger}@dfki.de

Abstract

Official travel warnings published regularly in the internet by the ministries for foreign affairs of France, Germany, and the UK provide a useful resource for assessing the risks associated with travelling to some countries. The shallow IE system *SProUT* has been extended to meet the specific needs of delivering a language-neutral output for English, French, or German input texts. A shared type hierarchy, a feature-enhanced gazetteer resource, and generic techniques of merging chunk analyses into larger pieces are major reusable results of this work.

1. Introduction

Assessing the risks associated with traveling to some countries is a major task in predicting the number of travelers on a particular flight. The EC-funded project AIRFORCE[1] (AIR FOReCast in Europe) has developed new methods for predicting passenger flow, mining large repositories of passenger data owned by airports and air carriers. Deviations from predictions can sometimes be explained by, e.g., an outbreak of an epidemic, civil war, or increase of criminal activities in certain countries or regions. While everyday news collected from different sources may provide relevant information, another useful resource is provided by official travel warnings published regularly in the Internet by the ministries for foreign affairs. This information is easily accessible and has been exploited to explain deviations of actual passenger data from predictions based on mining previous passenger data. Whenever such explanations are found, further statistical examination can be improved by adding new parameters to the data mining process.

In this paper, we concentrate on the information extraction task emerging from this application. Three governmental web sites are harvested weekly to collect travel warnings and details for each country:

Germany http://www.auswaertiges-amt.de/www/de/laenderinfos/index_html;

France http://www.france.diplomatie.fr/voyageurs/etrangers/avis/conseils/default2.asp – here only the "last minute" warnings were used, see under "derniére minutes";

United Kingdom http://www.fco.gov.uk/servlet/Front?pagename=OpenMarket/Xcelerate/ShowPage&c=Page&cid=1007029390590 – here the summaries were used.

The information available from these sites is written in the respective languages. While it remains relevant where a piece of information originates from (political differences between the countries involved are reflected in the travel recommendations, see, e.g., Iraq), the information should be represented in a homogeneous language-neutral way. Of particular interest are the beginnings and endings of periods during which a certain risk exists. It was expected that, with a little delay, passenger data should reflect these changes by decreasing or increasing, respectively. To identify these periods, the information extracted weekly is time-stamped.

The information of interest consists of a "head" part specifying the country in question, the date the text was put online for the first time, and the date it was last updated, plus zero or more "content" parts, specifying mentions of diseases, crimes, military conflicts, and terrorism. In addition, they may contain recommendations such as travel at own risk, travel to be avoided, travel without risk, etc.

In section 2., the application system is presented. The shallow information extraction system *SProUT* (Becker et al., 2002) has been extended in several respects to meet the specific needs of delivering a language-neutral output for English, French, and German input texts. A shared type hierarchy (section 3.), a feature-enhanced gazetteer resource (section 4.), and generic techniques of merging chunk analyses into larger results (section 5.) are major reusable results of this work.

2. The Application System

The application system consists of a complete Java implementation of a pipeline of the following components:

- the gatherer/stripper that takes weekly snapshots of travel warning summaries from the three web sites and removes HTML code,
- the shallow analysis component *SProUT*,
- the fragment merger, and
- the online MySQL database server that supports querying for particular properties, such as time intervals in which there was, e.g., a SARS warning for Taiwan.

SProUT is a platform for the development and configuration of multilingual unification-based shallow text analysis systems (Becker et al., 2002). Currently, the platform provides as linguistic processing resources a pipeline of tokenizer, gazetteer checker, morphological analysis, and

[1]EC contract no. IST-2000-25045, 2000–2003.

```
head_country_updated :> @seek(date_updated) & [ DAY #dofm,
                                                MONTH #month,
                                                YEAAR #year ]
                        gazetteer & [ GTYPE gaz_country,
                                      CONCEPT #locname ]
                     -> af_title & [ LOCNAME #locname,
                                     LOCTYPE "country",
                                     WEBPAGE-FROM "GB",
                                     UPDATED  #updated ],
    where #updated = Append(#year, "-", #month, "-", #dofm).
```

Figure 1: A *SProUT* grammar rule for English, representing "head" information in travel warnings. Symbols starting with `#` are logical variables (or coreference tags), and the values assigned to them can be used within the scope of the rule. The ampersand character `&` signals the unification of the information to its left and right.

```
np_location :> (det & [INFL #infl]) ?
               (adj & [INFL #infl]) *
               #loc & noun & [INFL #infl, TYPE location]
            -> af_location & [LOCATION #loc].
```

Figure 2: Another grammar rule, describing a locative NP and enforcing morpho-syntactic agreement among the optional determiner (`?`), followed by arbitrary many adjectives (`*`), and the final noun of type `location`.

```
[ LOCNAME "c_afghanistan",
  LOCTYPE "country",
  WEBPAGE-FROM "GB",
  UPDATED "2003-10-02" ]
```

Figure 3: A piece of *SProUT* output of type `af_title`, as it can be derived by applying the rule from figure 1.

grammar interpreter. Lingware is available for nine languages, including Chinese, Japanese, Dutch, Spanish and Slavic languages.

SProUT grammars rules are composed by a regular expression of left-hand side (LHS) elements (to be matched by the text input) and a single right-hand side (RHS) element, the output. All elements are typed feature structures (TFSs for short; see (Carpenter, 1992)) and originate from instantiations of type definitions wrt. a type hierarchy (Krieger and Schäfer, 1995). Applicability of the LHS is checked by a fast and cheap TFS unifiability test. The RHS is instantiated using TFS unification and information from the LHS can be "moved" to the RHS using coreference tags. *SProUT* provides a couple of standard regular operators, such as sequence, Kleene star/plus, disjunction, etc. A predefined set of functional operators (cf. figure 1) adds further functionality and can be seen as a "door" to the outside world of *SProUT*. The use of `@seek` (cf. figure 1) clearly extends the flexibility of the formalism, even making it context-free.

The rule in figure 1 matches input like "Updated: 3 October 2003 Iraq" The second LHS element must be a country name, as recognized by the gazetteer (cf. section 4.). The first LHS element is analyzed by calling—through `@seek`—the rule named `date_updated` that allows the two tokens "Updated" and ":" to be matched before a date expression is required. The values for day, month, and year bound by the rule `date_updated` are concatenated in the rule in figure 1, with intermediate dashes, by applying the functional operator `Append`. A sample instantiated output of the rule is shown in figure 3.

Figure 2 displays another *SProUT* rule and shows the use of the regular operators `?` and `*`, together with the use of coreference markers among several elements (in this case, to enforce inflectional agreement).

For the *SProUT* system instance in hand, the pre-existing tokenizers and morphological analysis components could be re-used without any modification. *SProUT* includes a development environment for configuring the components and defining the grammar rules. The three grammars modeling the warning information for English, French, and German have been written from scratch. The set of application-specific output types is relatively small (approx. 100). *SProUT* can be used as a runtime system through a Java API. Information between the components in our system is exchanged as TFSs, encoded in XML.

3. Resource: Multilingual Type Definitions for Travel Warnings

IE systems for, e.g., English usually match certain pieces of the input text against a grammar and produce a more or less textual output (cf. MUC competitions (Grishman and Sundheim, 1996)). For instance, the advice for a travel may be encoded as a TFS `[TRAVEL advised_against]`. Applying the same type of grammar to French might result in `[TRAVEL déconseillé]`. Multilingual IE requires language-neutral results though. What is more, the results must follow the same structure, irrespective of the grammar and language used. Our grammars thus abstract away from the surface form of words by using identical output types, leading to, e.g., `[TRAVEL not_advised]`.

While the above type definitions are specific to the travel warning domain, the use of TFSs for predefining homogeneous output representations for multiple grammars is of considerable general value. Existing grammars can be equipped with different output encodings. Note that, e.g., an XSLT-based post processor cannot provide the same flexibility as it works only on the final *SProUT* output.

```
cote d'ivoire | GTYPE:gaz_country | LANG:french | CONCEPT:c_ivory_coast
côte d'ivoire | GTYPE:gaz_country | LANG:french | CONCEPT:c_ivory_coast
elfenbeinkueste | GTYPE:gaz_country | LANG:german | CONCEPT:c_ivory_coast
elfenbeinkuste | GTYPE:gaz_country | LANG:german | CONCEPT:c_ivory_coast
elfenbeinküste | GTYPE:gaz_country | LANG:german | CONCEPT:c_ivory_coast
ivory coast | GTYPE:gaz_country | LANG:english | CONCEPT:c_ivory_coast
```

Figure 5: Gazetteer entries (excerpt) in multiple spellings and languages, relating to the same concept `c_ivory_coast`.

```
gazetteer := sign & [ GTYPE gaz_type,
                      LANG language,
                      CONCEPT string ].
```

Figure 4: The gazetteer type interface (see section 4.).

Besides these application-specific types defining the output of the system, there are component-specific types that can be reused together with the respective components. In *SProUT*, the output of all components follow certain type definitions. The type interfaces to tokenizer, morphology, and gazetteer guarantee that the results delivered by these components can smoothly be integrated. For instance, the type `gazetteer` specializes the unification of the type `sign` that, among other things, represents the textual surface, and a structure containing type-restricted gazetteer-specific features (cf. figure 4). The type `gazetteer` is used in *SProUT* grammar rules as shown in figure 1.

4. Resource: Multilingual Feature-Enhanced Gazetteers

Gazetteers encode word or phrase lists to be matched against input text. Gazetteer entries are typed, which allows the system to interpret a textual match as, e.g., a country or a date. A *SProUT* rule can thus interpret "Ivory Coast", "Côte d'Ivoire", or "Elfenbeinküste" as country names. Again, the problem of homogeneous output across linguistic variations occurs. The solution adopted was to extend the gazetteer by allowing sequences of characters to be associated with feature-value pairs, and thus our enhanced gazetteer is quite similar to a lexicon in a unification-based grammar. The feature `CONCEPT` carries identical non-linguistic identifiers for entries denoting the same objects (cf. figure 5). In *SProUT*, a rule can refer in its LHS to the gazetteer and assign the value of the feature `CONCEPT` to a variable that can be used in the output (cf. figure 1).

A complete country gazetteer has been developed as an independent, reusable resource, that can be used by *SProUT*. For each country the capital is defined. City states like Singapore are marked to account for possible ambiguities between cities and states. Entries are completely listed for English, French, and German. This multilingual coverage goes beyond what existing resources such as WordNet (Fellbaum, 1998), the World Gazetteer[2] or the CIA Factbook[3] have to offer.

Similarly, a complete gazetteer for date expressions has been developed for the three languages in question. Days, months and years are mapped onto coherent formats, allowing *SProUT* to come up with a homogeneous representation. For instance, "October 2nd, 2003", "2. Okt. '03", "2.10.03", "10/02/2003", etc. are all mapped onto "2003-10-02" (cf. figure 3).

Other, application-specific gazetteer entries comprise diseases, expressions suggesting crime, civil war, and terrorism.

[2] http://www.world-gazetteer.com/

[3] http://www.cia.gov/cia/publications/factbook/

5. Resource: Merging Techniques for Shallow Analysis Results

Merging analyzed pieces of text is realized by a two-step process.

5.1. Merging at the Chunk Level

Shallow analysis in *SProUT* (and other IE systems) often yields multiple results for a single chunk, originating from different ambiguity sources.

Local Ambiguities. The lexicon contains morpho-syntactic (e.g., gender, number, person) and semantic (senses) variations which might blow up the search space of the grammar interpreter/parser, resulting in multiple readings. There exist, however, several techniques which help to lower the ambiguity rate by compacting and unifying lexicon entries (Krieger and Xu, 2003). Typed feature structures are a necessary requirement for applying such techniques.

Spurious Ambiguities. Depending on the form of the grammar, multiple recursive rule calls might lead to attachment ambiguities which however produce equivalent RHS structures. In case we are only interested in the output (as we are in AIRFORCE), we are allowed to remove such duplicate TFSs.

Rule Ambiguities. We have often encountered rule sets, which, for specific input items, produce output structures that are related according to their degree of informativeness. I.e., we have found structures within these results which are more general or more specific than others.

In each of the above cases, we locally reduce the number of output TFSs for a chunk without giving up any information. This is achieved by virtue of TFS subsumption, yielding the most specific TFSs, which are guaranteed to carry the full information of more general ones. Very often, a single TFS remains at the end.

Due to the fact that the *SProUT* interpreter employs a longest-match strategy, further ambiguities are avoided at this stage.

5.2. Merging at the Sentential Level

Since chunk analyses may contribute to a travel warning analysis in different ways, we need a mechanism to combine complementary partial structures into reasonable superstructures. Relatively little is known in the literature on chunk or template merging; see, e.g., (Hobbs et al., 1997; Surdeanu and Harabagiu, 2002). Important hints for merging chunks are (i) overlap of slot names and slot fillers, (ii) satisfaction of domain constraints, and (iii) distance between chunks in the text. Criterion (i) and (ii) has been implemented in AIRFORCE by TFS unification and appropriateness conditions on the features of a TFS.

The strategy for combining information snippets is a kind of greedy forward merging. We incorporate new information from left to right as long as unification is successful. When it fails, we back up to the last successful matching point, store this structure, and restart the merging process with the TFS which has caused the failure. By using this heuristic strategy, the bulk of the partial TFSs in AIRFORCE for a given warning can be usually represented by one or two result structures. This strategy accommodates the fact that adjacent TFS should be merged first, before taking farther structures into account. A similar, although more general idea, was presented in (Xu and Krieger, 2003) who showed how to integrate even conflicting information. The operation used in this paper was a kind of prioritized default unification, viz., priority union (Kaplan, 1987).

The merging process can be restricted to TFSs within a certain distance, representing the heuristics that structures for distant pieces of text usually contribute to quite different aspects of a text, and thus should not be combined into a single structure. This implements criterion (iii) above.

The techniques are generic and can be parameterized to different text types so as to yield optimal results. Since the techniques have proven to be useful, we have integrated them into the *SProUT* environment.

Properties of a warning message in AIRFORCE are normally functional, meaning that they are associated with exactly a single value (but not with multiple values). For instance, the property `UPDATED` is assigned value `"2003-10-02"` which we express as a TFS of the form `[UPDATED "2003-10-02"]` (see figure 3). There is, however, one notable exception to this assumption: there might exist several reasons (and even none at all) which have led to a warning message, e.g., terrorism *and* crime. We will notate such relational or set-valued features using set braces and write, for example, `[CAUSE {terrorism, crime}]`. During TFS unification, set values are handled differently in that the elements of two sets are not unified pairwise (as is done in disjunctive unification), but instead the union of the sets is taken as the result value. Our use of sets here reminds us of the Lexical Functional Grammar framework (Kaplan and Maxwell III, 1988). Since relational values are extremely useful during the information extraction task, the *SProUT* system has been extended by this new data type and by the set unification operation.

6. Conclusion

We described resources and techniques for multilingual shallow analysis of travel warnings published regularly on the Internet. The homogeneous type definitions as well as the feature-enhanced gazetteer techniques contribute to homogeneous output across languages and grammars. The country and date gazetteers are fully reusable data. Dealing with multiple partial results to reduce the complexity of subsequent processing is mandatory for any shallow IE application.

Acknowledgments

This work has partially been funded by the European Union under contracts no. IST-1999-12457 to the project AIRFORCE and no. IST-2000-25045 to the project MEMPHIS. We are indebted to the DFKI *SProUT* team, in particular to Witold Drożdżyński, Jakub Piskorski and Ulrich Schäfer. We thank our colleagues Walter Kasper and Tim vor der Brück who implemented and adapted the gatherer/stripper component.

7. References

Becker, Markus, Witold Drożdżyński, Hans-Ulrich Krieger, Jakub Piskorski, Ulrich Schäfer, and Feiyu Xu, 2002. SProUT—shallow processing with unification and typed feature structures. In *Proceedings of the International Conference on Natural Language Processing, ICON-2002*.

Carpenter, Bob, 1992. *The Logic of Typed Feature Structures*. Tracts in Theoretical Computer Science. Cambridge: Cambridge University Press.

Fellbaum, Christiane (ed.), 1998. *WordNet: An Electronic Lexical Database*. MIT Press.

Grishman, Ralph and Beth Sundheim, 1996. Message understanding conference - 6: A brief history. In *Proceedings of the 16th International Conference on Computational Linguistics, COLING 1996*.

Hobbs, Jerry, Douglas Appelt, John Bear, David Israel, Megumi Kameyama, Mark Stickel, and Mabry Tyson, 1997. FASTUS: A cascaded finite-state transducer for extracting information from natural-language text. In Emanuel Roche and Yves Schabes (eds.), *Finite State Devices for Natural Language Processing*. MIT Press.

Kaplan, Ronald M., 1987. Three seductions of computational psycholinguistics. In P. Whitelock, M.M. Wood, H.L. Sommers, R. Johnson, and P. Bennett (eds.), *Linguistic Theory and Computer Applications*. London: Academic Press, pages 149–188.

Kaplan, Ronald M. and John T. Maxwell III, 1988. Constituent coordination in lexical-functional grammar. In *Proceedings of the 12th International Conference on Computational Linguistics, COLING-88*.

Krieger, Hans-Ulrich and Ulrich Schäfer, 1995. Efficient parameterizable type expansion for typed feature formalisms. In *Proceedings of the 14th International Joint Conference on Artificial Intelligence, IJCAI-95*. Also available as DFKI Research Report RR-95-18.

Krieger, Hans-Ulrich and Feiyu Xu, 2003. A type-driven method for compacting MMorph resources. In *Proceedings of RANLP 2003*.

Surdeanu, Mihai and Sanda M. Harabagiu, 2002. Infrastructure for open-domain information extraction. In *Proceedings of the Human Language Technology Conference (HLT 2002)*.

Xu, Feiyu and Hans-Ulrich Krieger, 2003. Integrating shallow and deep NLP for information extraction. In *Proceedings of RANLP 2003*.