Hard Problems of Tagset Conversion[*]

Daniel Zeman
Univerzita Karlova v Praze, Ústav formální a aplikované lingvistiky
Malostranské náměstí 25, CZ-11800, Praha, Czechia
zeman@ufal.mff.cuni.cz

Abstract

Part-of-speech or morphological tags are important means of annotation in a vast number of corpora. However, different sets of tags are used in different corpora, even for the same language. Tagset conversion is difficult, and solutions tend to be tailored to a particular pair of tagsets. We discuss Interset, a universal approach that makes the conversion tools reusable. While some morphosyntactic categories are clearly defined and easily ported from one tagset to another, there are also phenomena that are difficult to deal with because of overlapping concepts. In the present paper we focus on some of such problems, discuss their coverage in selected tagsets and propose solutions to unify the respective tagsets' approaches.

1 Introduction

Most annotated corpora use various types of tags to encode additional information on words. In some cases this information is merely the part of speech ("noun", "verb" etc.—hence the term *part-of-speech* or *POS tags*). In many cases, however, the string of characters comprising the tag is a compressed representation of a feature-value structure. Most of the features encoded this way are morphosyntactic (e.g. "gender = masculine", "number = singular"), hence the term *morphological tags*.

Unfortunately, it is very rare to see two corpora sharing a common set of tags. Language differences are only partially responsible—it is the corpus designers, their diverse views, theories and intended uses of the corpora, what matters most. Even two corpora of the same language may define two completely incompatible tagsets.

Such diversity proves disadvantageous for both human users and NLP software. A human user (linguist) typically wants to submit queries such as "show me all occurrences of a noun in plural, preceded by a preposition". Tags however rarely contain statements like "number = plural" literally. That would be prohibitively space-consuming. Instead we have to know that e.g. the fourth character of the tag being "P" means "plural". For instance, the tag `NNIS7-----A----`[1] may read as "part of speech = noun, detailed part of speech = common noun, gender = masculine inanimate, number = singular, case = 7th (instrumental), negativeness = affirmative". To work with the corpus efficiently, a linguist either needs to interpret the tags using specialized software, or to memorize the particular tag scheme. Obviously, if the same linguist has to switch to a different corpus, he/she must memorize more schemes or replace the tag interpretation software.

For many tagset pairs, designing the conversion procedure is not easy. On one hand, there are rare tagsets fitting at the same time languages as distant as Czech and Estonian; on the other hand, tagsets of two closely related languages (e.g. Danish and Swedish) or even two tagsets of the same language may differ substantially (for instance, the Mamba tagset of Swedish (Nivre et al., 2006) contains detailed classification of auxiliary verbs and punctuation but lacks features like number, mood, tense etc.; this is in sharp contrast to another Swedish tagset, Parole (Cinková and Pomikálek, 2006), which in turn is not compatible with the Danish Parole (Kromann et al., 2004) tagset.

From the above said it follows that the typical tag conversion is an information-losing process. Though it is often desirable to perform it anyway and preserve as much information as possible. Creation of a conversion procedure between two

[*] The research has been supported by the grant MSM0021620838 (Czech Ministry of Education).

[1] This example is taken from the Prague Dependency Treebank (Böhmová et al., 2003).

tagsets requires hours of tedious work, consisting mostly of reading the tagging guidelines and translating them into a programming language. A universal description, to which all tagsets map, could make this process easier, and its results reusable. One attempt to find such description and deploy it in the conversion task is DZ Interset (Zeman, 2008). In the present paper we discuss the development of the universal description and focus on selected hard problems that arise when comparing various existing tagsets.

The rest of the paper is organized as follows: In Section 2 we describe Interset and how it works. Then, Section 3 lists decisions that are difficult w.r.t. universality, demonstrates them on real tagsets, and proposes solutions.

2 Interset

Interset is a universal set of features and their values. It shall be able to store all features that are usually encoded in tags. The role of this universal set is similar to the role of Interlingua in Interlingua-based machine translation (Richens, 1958) or the role of Unicode among character sets. The Interset serves as an intermediate step on the way from tagset A to tagset B. The interaction between the Interset and tagsets A and B, respectively, is described in *tagset drivers*. Once the drivers have been implemented, we can do the two-way conversion A to B and B to A, plus the conversion to/from any other tagset that has been defined so far.

Besides abstract concept definitions, Interset also comprises some real software—supporting procedures that make adding new tagsets easier. Thanks to the encoding algorithm implemented in the support library, developers adding new tagsets need not (not necessarily) consider all features that are irrelevant to the tagset being added (but which may become relevant during conversion to a particular other tagset).

At the time of writing there are drivers for 20 tagsets of 10 languages, freely available on-line.[2]

2.1 A New Standard?

Interset is not a new annotation standard. There have been attempts to standardize morphosyntactic tagging and it is not Interset's mission to compete with them. Instead, the goal is to cover as many existing tagsets as possible whether they conform to a standard or not. The set of Interset features and values could of course be compared to those defined in standards. There have been several European projects concerning tagset standardization. The EAGLES project (EAGLES, 1996; Leech and Wilson, 1999) produced a set of recommendations for tagsets. Output of the LE-PAROLE project (Volz and Lenz, 1996) was a multilingual corpus of 14 European languages, morphosyntactically annotated according to a common core PAROLE tagset, extended with a set of language specific features. Another multilingual corpus with common tagset is MULTEXT (Ide and Véronis, 1994) for six European languages (en, fr, es, de, it, nl), and later its spin-off MULTEXT-EAST (Erjavec, 2004) for 12 languages (en, bg, cs, ee, hu, ro, sl, later also hr, lt, ru, sl-res); the tagsets used in MULTEXT corpora comply with EAGLES. Various EAGLES-compliant tagsets can be added to our system and their mutual similarity will probably make adding them all easier. Weakly related is also the Gold Ontology project (Farrar and Langendoen, 2003) that defines various linguistic concepts, some of which serve as feature names and feature values in Interset. Similarly, morphosyntactic and other terms are included in IsoCat.[3]

Interset has been used in cross-language parser adaptation (Zeman and Resnik, 2008), and in MorphCon, a GUI program that brings tag conversion to corpus users (Pořízka and Schäfer, 2009). Currently, it is being deployed as a means of unified access to morphosyntactic annotation for the users of the parallel multilingual corpus Intercorp.[4]

2.2 A New Tagset?

Interset is not primarily meant as a new *physical* tagset for annotation. Although it obviously could be used that way (possibly after compressing the feature values), it is better thought of as a set of concepts that physical tagsets map to. Physical tagsets often need to conform to linguistic tradition and terminology of the given language, minimize tagging errors etc. In contrast, the most important design constraint for Interset is the portability of information from one tagset to the others. If a feature value X is known in tagset A and unknown in B, and users of B are likely to tag the same words with feature value Y, then the Interset algorithms should replace X by Y.

[2] https://wiki.ufal.ms.mff.cuni.cz/user:zeman:interset

[3] http://www.isocat.org/

[4] http://www.korpus.cz/intercorp/?req=doc:uvod

Conversion via Interset often looses information but never adds new information. Interset may define feature value X but it just won't be set unless the source tagset defines it, too. Specifically, the conversion procedure does not retag words. For instance, the source tagset may define one tag `IN` for both prepositions and subordinating conjunctions. The target tagset may have separate tags for each of those categories but Interset will not sort out the words tagged `IN`. In fact, the procedure *never* looks at the word the tag is assigned to. It only works with the tag itself.

3 The Hard Problems

3.1 Pronouns, Determiners, Wh-Adverbs

Most languages and tagsets have personal pronouns, i.e. words like "I", "you", "he". Various interrogative and relative function words are often also considered pronouns. In addition, grammars of some languages distinguish *determiners* while others prefer to categorize the same thing as a sort of pronouns. According to the definition of EAGLES, **pronoun** is a function word that *replaces* a noun phrase, while **determiner** is a function word that *modifies* a noun phrase. As a result, proper EAGLES-pronouns behave like nouns and determiners behave like adjectives. Note that possessive pronouns (i.e. "my", "your"), also found in many languages, are personal possessive determiners in the sense of the EAGLES definition.

Because tagsets often disagree in what is pronoun, what is determiner etc., it is difficult to find a unifying approach. We decided to limit the number of the major parts of speech in order to minimize the cases where a word would end up with an empty part of speech. If there was a part of speech called *determiner*, drivers of tagsets not having determiners would either have to check whether **pos** `= det` during encoding, or they would fall back into a residual word class. On the other hand, if we tag determiners as special cases of adjectives (which is what Interset does), such drivers will simply encode determiners as adjectives (which are much more common).

We also followed this solution with pronouns because of the following reasons:

- Although pronouns are found in most tagsets, there is much controversy about the precise extent of that category.
- Some tagsets allow for distinguishing between *substantive* and *attributive* pronouns. Assigning pronouns to nouns and adjectives respectively helps preserve that distinction.
- Some of the features that distinguish pronouns from real nouns and adjectives (interrogativeness, for instance) are found with adverbs and numerals as well and thus it makes sense to separate them.

Pronouns are recognized by nonempty value of the `prontype` feature. The drawback of this approach is that the encoding procedure of a driver that recognizes pronouns must define its own method of asking the question *"Does the current feature structure correspond to a pronoun?"*

3.2 Numerals

Numerals form an analogy to the pronoun-determiner problem. Most tagsets recognize cardinal numbers as a separate category. However, the rest is less obvious. Some tagsets recognize ordinal numbers while others classify them as adjectives because of their syntactic behavior. Some Slavic tagsets define complex systems of numerals, according to traditional local grammars: besides cardinals and ordinals, there are separate tags for fractions, multiplications, adverbial ordinals, generic ordinals, interrogative, relative, demonstrative or indefinite numerals.

For the sake of consistency, the solution should be parallel to that of pronouns. Cardinal numbers, as the most specific and most widely recognized category, should retain their independence. The rest will be split among nouns (fractions), adjectives (ordinals and some generics) and adverbs (multiplications and ordinal points in time). Non-empty `numtype` feature will distinguish them from real nouns, adjectives and adverbs. Parallelly, `prontype` could be set, too, for interrogative, demonstrative and indefinite numerals.

3.3 Non-Finite Verb Forms

Non-finite verb forms often constitute mixed categories from the syntactic point of view. The syntactic properties of participles overlap with adjectives. Similarly, gerunds resemble nouns and transgressives function as adverbs. At the same time however, they retain their verbal arguments.

Usually, these words are tagged as forms of verbs. However, there are exceptions. In the Swedish Parole tagset, participles are tagged as adjectives. Czech equivalent of gerunds is considered a deverbative noun. The Interset guidelines prefer marking all three categories (participles, gerunds and transgressives) as verbs.[5]

3.4 Alternate Values

Some tagsets contain tags that encode two or more values of the same feature at the same time. For instance, the Czech PDT tags define 11 values for the third character of the tag, which denotes gender and animateness. Four are more or less independent values: `M` = "masculine animate", `I` = "masculine inanimate", `F` = "feminine" and `N` = "neuter". The rest are various combinations: `Y=(M|I)`, `T=(I|F)`, `W=(I|N)`, `H` and `Q=(F|N)`, `Z=(M|I|N)`, `X=(M|I|F|N)`. The obvious goal here is to make life easier for taggers because some word forms are systematically ambiguous.

Interset does not define separate feature values for all combinations. However, it does allow for storing alternate values if necessary:

```
$f{gender} = ["fem", "neut"];
```

At the first glance, such option poses a serious obstacle for encoder design. Before, the encoder was a nice sequence of `if` statements, merely saying "if gender is `fem`, output character `F`". Do we now need to check whether gender is array first? Even if we are writing driver for a tagset that never works with alternate values? Fortunately, no. Once again, the support library provides a procedure that can convert arrays to single values in all features except those that we acutally expect to contain arrays.

There is nonetheless one open question about alternate values. Interset cannot express permissible combinations of multiple features. In the Czech example above, `Q` actually means more than just "feminine or neuter". It means "feminine singular or neuter plural". Interset can store both genders and both numbers but by that it will also cover feminine plurals and neuter singulars.

3.5 Joint Categories

Every tagset assumes a certain tokenization of the tagged text. The tokenization guidelines may leave unsplit tokens that evolved historically from two words with separate categories. Examples include German *zum = zu dem* "to the", Czech *proň = pro něj* "for him". We then have a token corresponding to two other tokens also occurring in the corpus, with different POS.

Much more difficult is Arabic where orthographic rules dictate not to space-separate certain sequences of words. So for instance, the conjunction و *(wa)* "and", often thrown in at the sentence beginnings, is connected to the following word, as are prepositions. Tokenization cannot be finished before morphological analysis because word segmentation may have ambiguous solutions. Thus, in the following example, we'll end up with long tags `CONJ+PREP+NOUN_PROP` and `NOUN+CASE_DEF_NOM`, respectively.

```
<token_Arabic>وبالفالوجة
  <pos>wa/CONJ + bi/PREP +
        AlfAlwjp/NOUN_PROP</pos>
<token_Arabic>مثال
  <pos>mivAl/NOUN + u/CASE_DEF_NOM</pos>
```

Note that the first of the two examples corresponds to a sequence of three words (at least from the English perspective) while the second example describes just one noun (stem + suffix). The latter could be perfectly described in Interset; the former is the problem.

For the Czech and German examples above, Interset encodes one of the joint categories as a feature of the other. Thus for *zum*, Interset would note that it's a preposition with a special property *attached article*. It would not express any relation to how independent articles are tagged. Alternatively, we could use the array approach described in Section 3.4 and assign two values to the `pos` feature. Neither solution is optimal because both are far from universal. In theory any part of speech could combine with any other and there could be more than two concatenated tokens, each with its own features that should not be mixed all together.

4 Conclusion

We have described Interset, a reusable and universal method for tagset conversion via common set of features and their values. We briefly compared Interset to tagset standardization efforts and pointed out the differences in goals between standards and Interset. Then we presented numerous examples from real tagsets to illustrate the most difficult parts of tagset conversion. The solutions we proposed pursue the ultimate goal that information loss is minimized and similar information is encoded similarly, across tagsets and languages.

[5] Note however that any guidelines are only to ensure unified approach to different presentations of the same information. If participles were tagged as *normal* adjectives without sub-dividing adjectives into "normal ones" and participles, they would remain so in Interset and also in the target tagset.

References

Alena Böhmová, Jan Hajič, Eva Hajičová, and Barbora Hladká. *The Prague Dependency Treebank: A Three-Level Annotation Scenario*, chapter 7, pages 103–128. Kluwer Academic Publishers, Dordrecht, The Netherlands, 2003. 1

Silvie Cinková and Jan Pomikálek. Lempas: A make-do lemmatizer for the swedish parole-corpus. *The Prague Bulletin of Mathematical Linguistics*, 86:47–54, 2006. 1

EAGLES. Recommendations for the morphosyntactic annotation of corpora, 1996. URL `http://www.ilc.cnr.it/EAGLES96/annotate/annotate.html`. 2.1

Tomaž Erjavec. Multext-east version 3: Multilingual morphosyntactic specifications, lexicons and corpora. In *Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC 2004)*, Lisboa, Portugal, 2004. 2.1

Scott Farrar and D. Terence Langendoen. A linguistic ontology for the semantic web. *GLOT International*, 7(3):97–100, 2003. URL `http://www.linguistics-ontology.org/gold.html`. 2.1

Jan Hajič, Otakar Smrž, Petr Zemánek, Jan Šnaidauf, and Emanuel Beška. Prague arabic dependency treebank: Development in data and tools. In *Proceedings of NEMLAR-2004*, pages 110–117, 2004.

Nancy Ide and Jean Véronis. Multext (multilingual tools and corpora). In *Proceedings of the 15th International Conference on Computational Linguistics (COLING-94)*, Kyoto, Japan, 1994. URL `http://www.aclweb.org/anthology/C/C94/C94-1097.pdf`. 2.1

Matthias T. Kromann, Line Mikkelsen, and Stine Kern Lynge. Danish dependency treebank. In *http://www.id.cbs.dk/ mtk/treebank/*, København, Denmark, 2004. 1

Geoffrey Leech and Andrew Wilson. Standards for tagsets. In *Syntactic Wordclass Tagging. Text, Speech and Language Technology*, pages 55–80, Dordrecht, The Netherlands, 1999. Kluwer Academic Publishers. ISBN 0-7923-5896-1. 2.1

Joakim Nivre, Jens Nilsson, and Johan Hall. Talbanken05: A swedish treebank with phrase structure and dependency annotation. In *Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC 2006)*, Genova, Italy, 2006. 1

Petr Pořízka and Markus Schäfer. Morphcon – program pro konverzi českých morfologických tagsetů (morphcon – a program for conversion of czech morphosyntactic tagsets). In *Čeština ve formální gramatice*, Brno, Czechia, February 2009. 2.1

Richard Hook Richens. Interlingual machine translation. *The Computer Journal*, 1(3):144–147, 1958. 2

Norbert Volz and Suzanne Lenz. Multilingual corpus tagset specifications. mlap parole 63–386 wp 4.1.4, 1996. URL `http://www.elda.org/catalogue/en/text/doc/parole.html`. 2.1

Daniel Zeman. Reusable tagset conversion using tagset drivers. In *Proceedings of the 6th International Conference on Language Resources and Evaluation (LREC 2008)*, Marrakech, Morocco, 2008. European Language Resources Association. ISBN 2-9517408-4-0. 1

Daniel Zeman and Philip Resnik. Cross-language parser adaptation between related languages. In *Proceedings of IJCNLP workshop on NLP of less privileged languages (NLPLPL)*, Hyderabad, India, 2008. 2.1