ISOcat Definition of the National Corpus of Polish Tagset

Agnieszka Patejuk[1,2] **and Adam Przepiórkowski**[2,3]

[1]Jagiellonian University, Cracow
[2]University of Warsaw
[3]Institute of Computer Science, Polish Academy of Sciences, Warsaw

agnieszka.patejuk@gmail.com adamp@ipipan.waw.pl

Abstract

This paper describes the first definition of a complete morphosyntactic tagset, The National Corpus of Polish Tagset, in the ISOcat Data Category Registry. Although the task of implementing such a sophisticated tagset in ISOcat turned out to be significantly more challenging than expected, it was successfully completed. The result of this work, the `nkjp` Data Category Selection containing 85 carefully defined Data Categories owned by the `NKJP` group, is publicly available at `http://www.isocat.org/interface/index.html`. Discussing various solutions considered during this implementation, this paper presents certain limitations of ISOcat and offers some suggestions for its further development.

1. Introduction

The aim of this paper is to report on the process of defining the NKJP Tagset in the ISOcat Data Category Registry, commenting on the experience of using this system and suggesting ways in which it could be improved. First sections provide background information about the National Corpus of Polish, its tagset and ISOcat. Next, the implementation of the tagset is presented, discussing the limitations from which particular alternatives suffered and explaining how the tagset was eventually defined. A section highlighting various technical aspects of ISOcat and how these influenced the implementation of the tagset follows, offering some directions for further development of ISOcat. Finally, a succinct summary of the results achieved is provided.

2. NKJP and its tagset

The National Corpus of Polish (Pol. *Narodowy Korpus Języka Polskiego*; NKJP; `http://nkjp.pl/`) is a 3-year project terminating in December 2010, carried out at 4 Polish institutions. Background description of the project can be found in Przepiórkowski *et al.* 2008, 2010. One of the annotation levels in NKJP is morphosyntax.

The tagset of the National Corpus of Polish (henceforth, the NKJP Tagset; Przepiórkowski 2009) is a slightly modified version of the IPI PAN Tagset (Przepiórkowski and Woliński, 2003), a *de facto* standard tagset for Polish. There are 36 grammatical classes approximately corresponding to parts of speech, 13 grammatical categories and their possible values (36 in total). Each grammatical class has an associated list of appropriate grammatical categories, which may be specified as obligatory or optional for the particular class. Furthermore, there are a number of constraints on the possible values of categories appropriate for some classes, which will be discussed in more detail in § 4.

Most grammatical classes have a list of categories for which they inflect or whose value is specified lexically. Gerunds (`ger`) inflect for number, case and negation while their aspect and gender (always neuter) are lexical. The complete morphosyntactic tag for 'piciem', `ger:sg:inst:n:imperf:aff`, provides the following information: it is a gerund whose values of the categories of number, case, gender, aspect and negation are singular, instrumental, neuter, imperfective and affirmative respectively. Certain categories may be optional – while some prepositions, like 'do' which is always `prep:gen`, have only one form, many others take different forms depending on the context: 'pod' `prep:inst:nwok` as opposed to 'pode' `prep:inst:wok` which have different values of vocalicity. Some classes such as conjunctions (`conj`) or predicatives (`pred`) are non-inflecting and, having no associated categories, their complete tags consist only of grammatical class tags: 'i' (`conj`), 'to' (`pred`). Finally, there are classes such as abbreviation (`brev`), bound word (`burk`) and unknown form (`ign`) which, rather than being traditionally understood parts of speech, serve technical purposes.

Alongside widely known traditional grammatical categories such as case (7 values), number, person, gender (5 values), degree, aspect, there are categories such as negation and accentability as well as some classes rather specific to Polish. These include accommodability which determines the syntactic behaviour of numerals, postprepositionality describing the behaviour of certain pronouns in relation to prepositions, agglutination optionally applicable to one class of verbs and vocalicity which regulates the distribution of agglutinates. The remaining category, fullstoppedness, is a technical category taking one of two values depending on whether the abbreviation segment has to be followed by a full stop.

3. ISOcat and its architecture

The ISOcat project is an implementation of the ISO 12620 standard which is described as follows in the abstract available on the ISO website (`http://www.iso.org/`):

> ISO 12620:2009 provides guidelines concerning constraints related to the implementation of a

> Data Category Registry (DCR) applicable to all types of language resources, for example, terminological, lexicographical, corpus-based, machine translation, etc. It specifies mechanisms for creating, selecting and maintaining data categories, as well as an interchange format for representing them.

The architecture of ISOcat is therefore determined by the requirements set by the ISO 12620 standard: the DCR model consists of administrative information, descriptive information and linguistic information. These components of the DCR are reflected in the organisation of the data contained in individual Data Categories (DCs), all of which have an Administration Information Section, Description Section and potentially Conceptual Domains whose values are specified independently of the content of each other for different languages.

The Administration Information Section contains the Administration Record which provides information about the mnemonic identifier (Identifier, as opposed to PID, the unique Persistent IDentifier), version, registration status, justification together with its origin as well as the dates of creation and the last change made to the DC.

The Description Section (DS) contains the Language Section which organises information about a given DC according to language. It provides details such as the definition, its source and, optionally, some notes in the Definition Section. There is also the Name Section which specifies the DC names together with their status in the given language and the Example Section where relevant examples together with their sources can be provided. The DS also contains the Data Element Section which is intended as the place for storing language-independent names of the DC.

The Conceptual Domain (CD), an inherent feature of *complex* DCs, contains the information about its possible values in a given language, which in turn depend on the type of the particular DC. There are three types of *complex* DCs: *closed*, *open* and *constrained*. The CDs of a *complex/closed* DC are represented as finite sets of *simple* DCs which, being the only non-complex DC type, do not have any associated CDs and therefore have no associated values themselves. The two remaining types of *complex* DCs, *complex/open* and *complex/constrained*, are characterised by the fact that the sets of their values cannot be enumerated exhaustively: the *open* DC is a 'complex data category whose conceptual domain is not restricted to an enumerated set of values' while the *constrained* DC is a 'complex data category whose conceptual domain is non-enumerated, but is restricted to a constraint specified in a schema-specific language or languages' (ISOcat Glossary, 2010).

More information about ISOcat can be found on the website of the project (`http://www.isocat.org/`).

4. Defining the NKJP Tagset in ISOcat

The original idea was to enter into the ISOcat DCR the grammatical classes, grammatical categories and their corresponding values. After having completed this stage, the values would be attached to appropriate grammatical categories and these, subsequently, would be related to appropriate grammatical classes as their attributes. Ideally, the relations between a given grammatical class or category and its possible values or attributes would be expressed in its CD for the particular language. Adopting such a strategy would be preferable not only because of being the most economic one in terms of the amount of time necessary to define the NKJP Tagset as ISOcat DCs but also because it would closely reflect the design and structure of the tagset.

4.1. Elegant but impossible

Regrettably, such a solution, even though it would certainly be the most elegant one, could not be implemented because of the architecture of the ISOcat DC types. The DC type which matches best the requirements set by this task is, in most cases, the *complex* one as every complete morphosyntactic NKJP tag consists of a tag signalling the grammatical class followed by tags corresponding to values of appropriate grammatical categories, if there are any. Using the range of DC types offered by ISOcat, the values of grammatical categories were classified as *simple* DCs since, being simple atoms, they have no values themselves. They were subsequently related to corresponding grammatical categories whose DC type was set to *complex/closed* because they have well-defined repertoires of enumerable values. Ideally, it would be possible to list in an analogous way all the corresponding grammatical categories in the CD of a given grammatical class as its attributes. This way, both grammatical classes and categories would be classified as *complex/closed* DCs while values of grammatical categories would be *simple* DCs – it is here that a serious problem is encountered. While it is possible to provide *simple* DCs as values in the CD of a *complex/closed* DC, it is not possible to specify such a *complex/closed* DC as one of the attributes of another *complex/closed* DC, which would be the case here. The reasons are manifold: non-simple DC types cannot be linked to the CD of a *complex* DC, only *complex/closed* DCs can have CDs with enumerated content and, more importantly, there is no support for representing any relations other than that of *Value* in the CD at the moment.

4.2. Clever but impossible

Another approach at defining the tagset in ISOcat was based on the idea of entering complete NKJP tags directly into the DCR as *complex/open* DC types. A complete morphosyntactic NKJP tag, for instance `subst:sg:nom:m2`, the template for which would be `class:number:case:gender` has the following structure: the first element represents the grammatical class, followed by values of appropriate grammatical categories, if there are any. If the complete tag consists of more elements than the obligatory grammatical class, every segment is separated from the following one with a colon. With 36 grammatical classes, 13 grammatical categories and 36 values in total, there are more than 1500 possible complete tags, which makes the task of creating and entering them manually unfeasible. Due to this fact, the complete tags need to be either generated or extracted from the corpus. The latter solution is given preference because it avoids problems encountered in the case of baseline tag generation which include accounting for restrictions on the

values of grammatical categories as well as the optionality of some categories. On the other hand, choosing to extract the complete tags from the corpus, there is the risk of some not being represented due to their absence from the data. Having obtained the complete tags by either method, the original tag separators must be replaced by some other character due to the fact that the use of colons in the DC Identifier is restricted and the system will not accept such DCs. Subsequently, automatic descriptions of DCs would be generated (`subst:sg:nom:m2` = noun, singular, nominative, animate masculine) and, together with corresponding complete morphosyntactic tags as identifiers, fitted into the frame provided by the Data Category Interchange Format (DCIF), the XML export format for DCs grouped into Data Category Selections (DCSs). Finally, the modified DCIF file would be fed into the DCR.

Unfortunately, this solution could not be implemented as DC import is not supported at all at the moment. According to the obtained information, although the DCIF DC import is given priority, there is no set date of its introduction and, more importantly, it is either not going to be publicly available or it is going to be subject to certain restrictions on the allowed data import limit.

4.3. Successful but time-consuming

Due to the fact that the implementation of the previous solution was not possible because of technical limitations, another strategy had to be adopted. 36 DCs defining grammatical classes which roughly correspond to parts of speech were created manually. Due to the uniqueness of many solutions adopted in the tagset which include, for instance, a separate class for depreciative forms (`depr`) and two distinct classes of pronouns, it was not possible to use already existing DCs and new ones tailored to the needs of NKJP were created. Definitions of NKJP DCs were written, with minor modifications, on the basis of extracts from publications about the IPI PAN Corpus (mainly, Przepiórkowski 2004) and NKJP with appropriate bibliographic source provided, following ISOcat guidelines.

Since the appropriate grammatical categories could not be represented in the CD of grammatical classes as their attributes, the definition is followed by a line containing detailed information about the grammatical categories associated with the particular grammatical class. In order to make it easier to trace associated grammatical categories, the list is accompanied by corresponding PIDs in plain text since the use of hyperlinks in definitions is not supported. Furthermore, if a category happens to be optional for some class, information about its optionality is also provided in brackets after the corresponding PID. Since grammatical classes are sets of lexemes which are not defined in ISOcat themselves, the DC type of defined grammatical classes was set to *complex/open*.

4.4. Another alternative

There is an alternative solution which, although it has not been implemented, is worth mentioning as it could improve the results achieved through the application of the previous one. This would, however, come at a considerable cost – grammatical classes would need to be reclassified as *complex/closed*, which would distort the ontology modelled in this implementation where grammatical classes are consistently *complex/open* DCs whose values are appropriate lexemes. The values of grammatical categories could be related directly, bypassing the level of categories, to appropriate grammatical classes in their CDs. Though the intermediate level of grammatical categories would not be represented in the CD of the given grammatical class, this information would be still available in its justification as well as definition. In this way, the information provided in the description of the Data Category (DC) would complement the specification of grammatical category values in its CD. Furthermore, such a solution would make it possible to account in a straightforward way for most constraints on the values of categories appropriate for classes, with the exception of more complex ones as in the case of imperative (`impt`) where the range of appropriate values of the category of person is restricted by the value of the category of number.

However, there are some serious drawbacks which have to be taken into consideration. These include a great deal of manual work due to the complete lack of support for templates or multiple changes to the DC or even the entire DCS. Moreover, since values in the CD are listed in an alphabetical order, the values of corresponding grammatical categories would not be grouped. Finally, there is no means to account for the optionality of values of certain categories directly in the CD of the given grammatical class – such information could only be retrieved in the definition and justification of the DC.

5. Technical issues

As it is openly acknowledged on the project website, ISOcat is constantly under development – it is emphasised that the Web Interface (WI) available at the moment is a beta version. As a result of this implementation which required a considerable amount of time spent using ISOcat WI, a few bugs were reported and many more features were requested. Most of the identified bugs were fixed while only some of the suggested functionalities have been introduced. This section recaps the main points concerning the technical side of the defining the NKJP Tagset in ISOcat and brings into focus some issues which require particular attention.

5.1. What has been done

So far, only two of the requested features have been implemented, the first one being the update of the ISOcat DC search following earlier requests from other users. It is now possible to refine the search results using a variety of parameters such as the matching method, the language of keywords as well as fields, profiles and scopes to be considered. On the one hand the update introduces many more features than requested, but on the other it does not include an important functionality that was suggested – the possibility to use DC search when specifying the values in the CD of a DC. The second update brought the possibility to delete a DCS but not a DC which, being persistent, which is a part of ISOcat policy, cannot be removed once created. Since there is no way to delete a DC, the only solution at the moment is to recycle it – the only element of the DC

which cannot be changed is its PID. In the future, however, an alternative in the form of having the possibility to deprecate a DC is going to be made available while at the moment it is only possible to set the name status of the DC to deprecated.

5.2. What needs to be done

There are many vital functionalities which could make the work with ISOcat significantly easier and more efficient but, unfortunately, are not supported at the moment. These include the introduction of basic tools such as DC templates which could be created from scratch or on the basis of a particular DC chosen by the user. A multiple change tool, possibly with regular expression support, making it possible to apply multiple changes at the same time instead of doing it manually would certainly be in place. It could easily be applied to editing the definitions (to change some key term shared by a number of DCs), their source, but also to changing features such as DC name status, DC type or even CD values (if some of them are shared). Multiple change tool would also be particularly useful when changing the scope of chosen DCs or even the entire DCS as currently changing the scope of a DCS does not result in an automatic change of the scope of DCs it contains. At the moment all of the above must be done manually, which is extremely time consuming when handling a greater number of DCs. The next feature request, whose importance is supported by ample evidence presented earlier, is the implementation of the DCIF import which would not only enable automating the management of the DCS to a large extent but it would also provide the first basic alternative to managing DCs via the WI which is currently the only means of accessing ISOcat.

Finally, there are some minor issues which could still have a considerable positive impact on the experience of using ISOcat. The introduction of password change would certainly be appreciated by many, not only for security reasons. It might be a good idea to resign from the obligatory comment required by the WI when saving a new DC, which would make working with ISOcat even more smooth and reduce the number of whitespace comments. Last but not least, the ISOcat WI provides a brilliant platform for work supported by state-of-the-art technology but a faster, more lightweight alternative would certainly be welcome.

6. Conclusion

In spite of a number of problems encountered during this implementation, the main objectives were achieved – the NKJP Tagset has been successfully defined in the ISOcat DCR and it is now available as a public DCS, `nkjp`, owned by the `NKJP` group which is the first and used to be the only group with a public DCS in the ISOcat DCR. The `nkjp` DCS contains 36 *complex/open* DCs corresponding to grammatical classes, 13 *complex/closed* DCs defining grammatical categories and 36 *simple* DCs specifying values of these categories. In total, 85 DCs were created, all of which have an associated definition describing its function in the tagset, accompanied by relevant examples from Polish. Due to the technical limitations discussed at length, it was not possible to reproduce the original design of the tagset and alternative solutions had to be adopted.

At a glance, the NKJP Tagset was modelled in the ISOcat DCR in the following way: grammatical classes are *complex/open* DCs with appropriate lexemes as their values, grammatical categories are *complex/closed* DCs whose CDs are populated with their possible values modelled as *simple* DCs. Due to the fact that, using currently implemented ISOcat solutions, it was not possible to express formally the relation of *Attribute* between grammatical classes and categories, definitions of grammatical classes provide additional information about appropriate categories together with their PIDs, details about their optionality and, if applicable, constraints on their values.

The idea of standardising linguistic concepts is undeniably appealing and ISOcat provides a convenient platform to assist this process. It offers functionalities such as DC checking which support the creation of DCs in accordance with ISO standards and gives the unique possibility to submit DCs for standardisation. Though the experience of defining the NKJP Tagset in the ISOcat DCR suggests that there is still some room for improvement, the current implementation of ISOcat was flexible enough to allow a successful realisation of this task – this is the first public ISOcat definition of any complete tagset, for any language.

References

ISOcat Glossary (2010). `http://www.isocat.org/interface/JSXAPPS/ISOcat/help/ISOcatGlossary.html`.

Przepiórkowski, A. (2004). *The IPI PAN Corpus: Preliminary version*. Institute of Computer Science, Polish Academy of Sciences, Warsaw.

Przepiórkowski, A. (2009). A comparison of two morphosyntactic tagsets of Polish. In V. Koseska-Toszewa, L. Dimitrova, and R. Roszko, editors, *Representing Semantics in Digital Lexicography: Proceedings of MONDILEX Fourth Open Workshop*, pages 138–144, Warsaw.

Przepiórkowski, A. and Woliński, M. (2003). The unbearable lightness of tagging: A case study in morphosyntactic tagging of Polish. In *Proceedings of the 4th International Workshop on Linguistically Interpreted Corpora (LINC-03), EACL 2003*, pages 109–116.

Przepiórkowski, A., Górski, R. L., Lewandowska-Tomaszczyk, B., and Łaziński, M. (2008). Towards the National Corpus of Polish. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation, LREC 2008*, Marrakech. ELRA.

Przepiórkowski, A., Górski, R. L., Łaziński, M., and Pęzik, P. (2010). Recent developments in the National Corpus of Polish. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation, LREC 2010*, Valletta, Malta. ELRA.