Identification Key generation WebService : User Guide

Laboratoire Informatique et Systématique

2011

Abstract

Identification keys are widely used by scientists to identify taxa. This new identification key generation WebService generates single-access keys on demand, for single users or research institutions. It receives user input data (using the standard SDD format), accepts several parameters for the key generation (impacting the key topology and representation), and supports several output formats. Furthermore, key generation automation is possible thanks to the WebService architecture.

Contents

List of Figures

1 Using the webservice

1.1 Introduction

This WebService can be used in several cases. First, directly from a Scratchpads instance. In this case, a SDD file is automatically generated by the Scratchpads instance. The WebService can also be called directly, by using the `www.identificationkey.fr` website. In this case the SDD file must be provided by the user (the Xper2[1] software can generate it for you). Since this service is part of an open source project, any other institution can provide such service on its own platform.

1.2 Parameters

Several parameter are available, in order to generate alternative keys in different formats. All parameters are passed to the WebService as simple character strings.

1.2.1 `sddURL`

This parameter contains the URL of the SDD input file.

1.2.2 `representation`

This parameter specifies the architecture[2] of the identification key generated by the WebService. The possible values of this parameter are :

- `tree` = a hierarchical representation.
- `flat` = a flat representation

1.2.3 `format`

The format parameters allows the user to specify the file format desired for the output file. The following formats are available:

- `txt` = a plain-text file.
- `html` = an HTML file.
- `interactivehtml` generates an interactive html identification key. This output format is particularly suitable for smartphones.
- `pdf` = a PDF file.
- `wiki` = a file containing wikiText-formatted text.
- `speciesidwikistatement` = a text file containing wikiText-formatted text, adapted to the `http://species-id.net` website (see `http://species-id.net/wiki/Philaccolilus`).
- `speciesidwikiquestionanswer` = a text file containing wikiText-formatted text, adapted to the `http://species-id.net` website.
- `sdd` = an XML file containing key as SDD format.
- `dot` = a GV file containing key as dot format.
- `zip` = a ZIP file containing all other types of output format.

[1]`http://lis-upmc.snv.jussieu.fr/lis/?q=en/resources/software/cai/xper2/downloads/last`
[2]*cf.* `http://www.identificationkey.fr/index.php/help_representation`

	tree	flat
txt	✓	✓
html	✓	✓
interactivehtml	✗	✓
pdf	✓	✓
wiki	✓	✓
speciesidwikistatement	✗	✓
speciesidwikiquestionanswer	✗	✓
sdd	✗	✓
dot	✓	✗
zip	✓	✓

Figure 1: Possible Key representations depending on the output format

1.2.4 `fewStatesCharacterFirst`

When this parameter is set to `yes`, the WebService artificially increases the score of characters which have fewer states, in order to have them first in the key[3]. The possible values of this parameter are :

- `yes` : enables the option.
- `no` : disables the option.

cf. figure 2 on page 6.

1.2.5 `mergeCharacterStatesIfSameDiscrimination`

When this parameter is set to `yes`, the WebService merges character states when their remaining taxa are similar for both issue[4]. The possible values are :

- `yes` : enables the option.
- `no` : disables the option.

cf. figure 3 on page 6

1.2.6 `pruning`

When this parameter is set to `yes`, the WebService removes the branches of the key in which it is impossible to totally discriminate the remaining taxa [5]. The possible values are:

- `yes` : enables the option.
- `no` : disables the option.

cf. figure 4 on page 7

1.2.7 `verbosity`

This parameter allows to select the level of verbosity of the WebService, which determines the content of the output file. Depending on this parameter, a header (value "`h`") can be included in the output file as well as the branches with undescribed taxa labelled "Other" (value "`o`"), warning messages (value "`w`") and a short statistics table (value "`s`"). The parameters can contain up to 4 characters, for instance :

[3] *cf.* `http://www.identificationkey.fr/index.php/help_fewStatesCharacterFirst`
[4] *cf.* `http://www.identificationkey.fr/index.php/help_mergeCharacterStatesIfSameDiscrimination`
[5] *cf.* `http://www.identificationkey.fr/index.php/help_pruning`

- `h` = will display only the header.
- `hows` = will display the header, the branches with undescribed taxa, the warnings and the statistics table.
- `ws` = will display warnings and the statistics table.

1.2.8 `scoreMethod`

This parameter determines which scoring method is used to sort the characters by their discriminatory power. There are three possible values :

- `xper` : Xper scoring method
- `jaccard` : Jaccard scoring method
- `sokalandmichener` : Sokal & Michener scoring method

These scoring methods are quantitative assessments of a character's ability to distinguish between different taxa. The three discriminatory power are the result of the sum, for all pairs of taxa, of a measure of dissimilarity between the taxa for which the character is evaluated.

For the Xper discriminatory power, the dissimilarity between two taxa for a given character is either 0 or 1 depending on whether or not these two taxa have common character states.
In the case of the discriminatory power based on the Jaccard dissimilarity index or based on the Sokal & Michener index, the dissimilarity between two taxa is comprised between 0 and 1 and depends on the number of states of the character in common.

The following variables are used to calculate a character's (C) discriminant power for two Taxa T_a and T_b :
n_{11} : the number of states of C that occur for both T_a and T_b.
n_{10} : the number of states of C that occur for T_a only.
n_{01} : the number of states of C that occur for T_b only.
n_{00} : the number of states of C that occur for neither T_a or T_b.

- the dissimilary according to Xper[1] is based on incompatibility between descriptions. Two taxa are incompatibles (or dissimilar or discriminated) if, for one character, there is no common states of characters, i.e. if $n_{11} = 0$.

$$n_{11} = 0 \Rightarrow d_{Xper} = 1 \; otherwise \; d_{Xper} = 0$$

- the dissimilarity using the jaccard method[2] take into account at least the states of the characters of one of the two taxa that we compare.

$$d_{Jaccard} = (n_{01} + n_{10})/(n_{01} + n_{10} + n_{11})$$

Similarity = (1 - Dissimilarity), i.e. two taxa are even more similar than their number of common states of characters increase.

$$S_{Jaccard} = n_{11}/(n_{01} + n_{10} + n_{11})$$

- the dissimilarity according to Sokal et Michener[3] between two taxa for one character is measured by taking into account all the possible states existing for the character and not only those existing for at least only one of the taxa.

$$d_{SM} = (n_{01} + n_{10})/(n_{00} + n_{01} + n_{10} + n_{11})$$

Similarity = (1 - Dissimilarity), two taxa are even more similar than they share common present and absent states.

$$S_{SM} = (n_{00} + n_{11})/(n_{00} + n_{01} + n_{10} + n_{11})$$

1.2.9 Weight parameters

If you want, you can apply weights to characters and taxa to alter the structure of your identification key. For example, if you want a certain character to appear sooner in the key, you can give it a greater weight. The SDD input format supports character weight with the following characteristics :

- the weight (W) is a positive integer, such that : $W \in \{1, 2, 3, 4, 5\}$, but is stored as a character string in the SDD file with the format "Rating$W$of5"
- a weight can be given to any character, on a per-taxon basis, *i.e.* the same character can have different weights, depending on the taxon considered.
- furthermore, the SDD format allows the attribution of more than one weight per character and taxon. Up to 6 different weights may be given to any taxon-character couple, depending on the context in which they should be applied :
  - `ObservationConvenience`
  - `Availability`
  - `Repeatability`
  - `CostEffectiveness`
  - `PhylogeneticWeighting`
  - `RequiredExpertise`

  Keep in mind that even if your SDD file contains only one kind of character weight context, the SDD format specification requires that this weight context be defined in the SDD file and that it belong to one of the six weight context defined above.

`weightType` **parameter** By default (or if `weightType` is set to "`global`"), the character weight is applied globally, *i.e.* even if, in the SDD input file, some taxa have different weights for a given character D, the weight that will actually be applied is the average weight of every taxon-character couple in with D is present (In the case of taxon-character couples for which the weight is not defined in the SDD file, the median weight value, 3, is used).

If the parameter `weightType` is set to "`contextual`", the weight applied for each character D is the average weight of every taxon-character couple in which D is present, but only taking in consideration the couples for which the taxon has not already been identified in an earlier step of the algorithm (again, if a weight is not set for a given taxon-character couple, the median value, 3 is used).

`weightContext` **parameter** This parameter determines which character weight context will be applied for the key generation process. If set, this parameter's value must be one of the 6 character weight context defined above (*e.g.* `CostEffectiveness`).
Only one weight context is applied during the key generation process. If this parameter is not set, the key will be generated without considering the weights that may be present in the SDD input file at all.

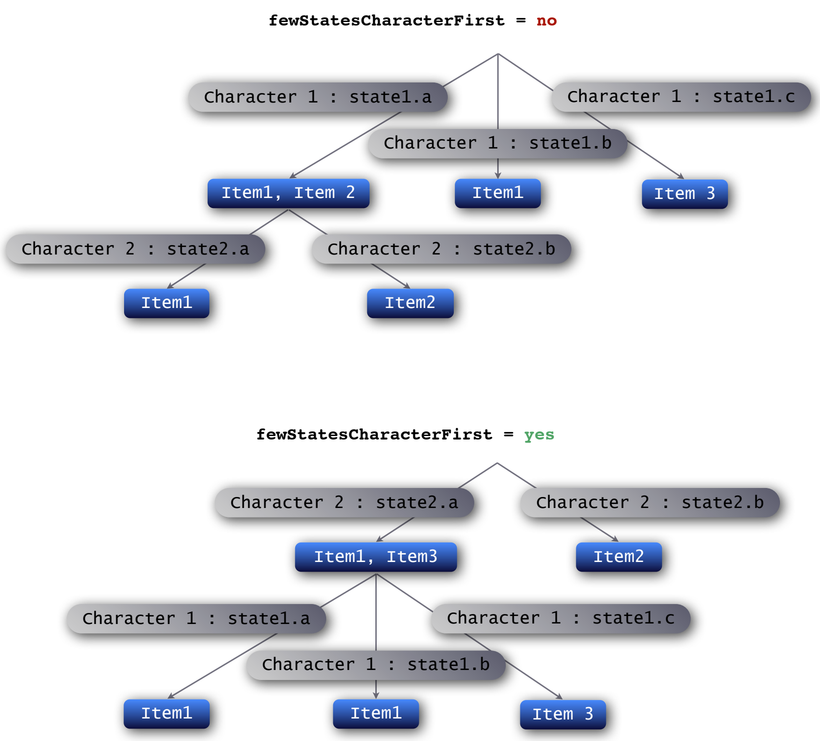


Figure 2: `fewStatesCharacterFirst` example

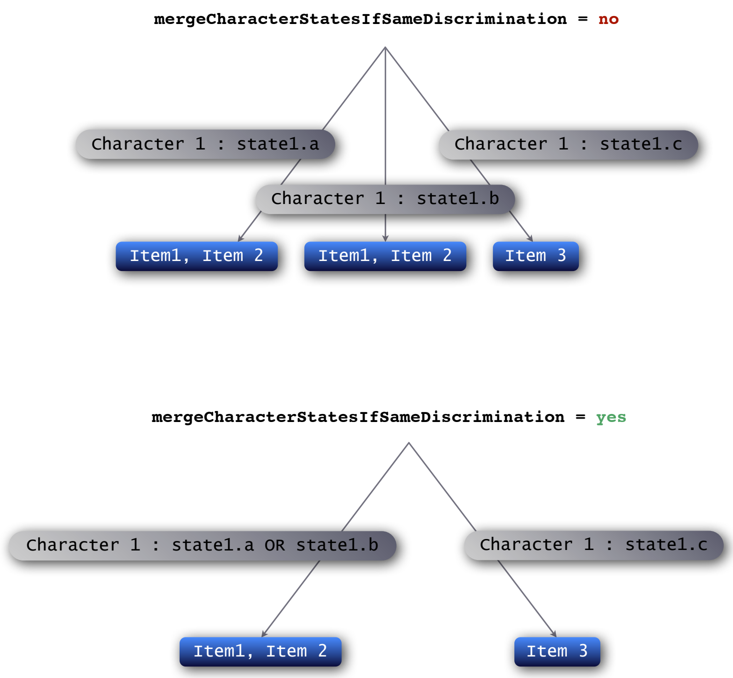
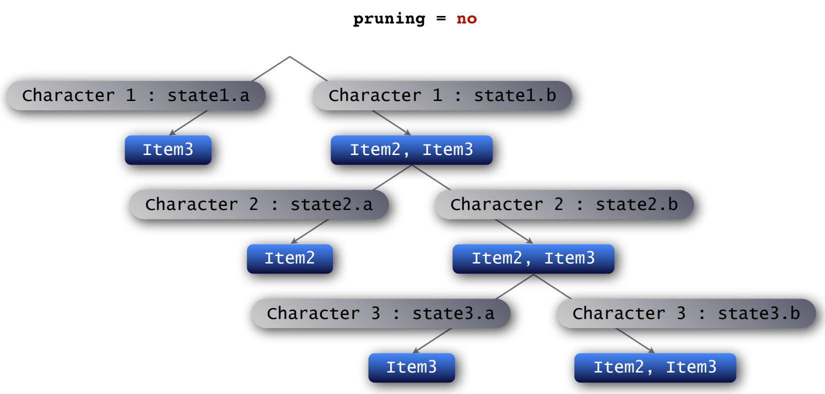


Figure 3: `mergeCharacterStatesIfSameDiscrimination` example



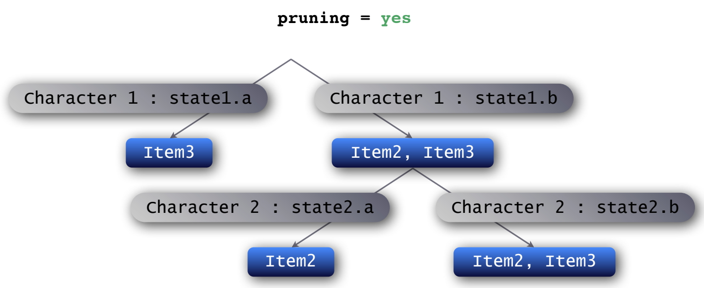


Figure 4: `pruning` example

2 key generation algorithm

The method for creating the graph (the identification key) is a heuristic that selects, step by step, the "best" character to form a node of the graph according to the criteria to optimize, and then create branches from this node and continues the process in each branch until a stop criterion is reached.

Here is a more detailed description of each step of the algorithm :

- The $BEST\_CHAR$ function selects the best character, based on the discriminant power (score) of the character, which optimizes the various global properties of the graph.
- The $REMAINING\_ITEMS$ function updates the list of the remaining items compatible with the path.
- At the beginning of the algorithm the list of character is initialized with all existing characters. At each iteration, all the remaining characters are calculated by the $REMAINING\_CHARS$ function that takes in consideration the relationship between characters.

```
$characters \leftarrow the\ list\ of\ characters$
$items \leftarrow the\ list\ of\ items$

$KEY(items, characters)$
**if** $STOP(items, characters) = TRUE$ **then**
  $create\ a\ leaf\ labeled\ by\ items$
**else**
  $best\_desc \leftarrow BEST\_DESC(items, characters)$
  $create\ a\ node\ labeled\ with\ best\_char$
  **for all** $Si \in Sbest\_char$ **do**
    $remaining\_items \leftarrow REMAINING\_ITEMS(items, best\_char, Si)$
    **if** $SIZE(remaining\_items) > 0$ **then**
      $create\ a\ branche\ labeled\ with\ Si$
      $remaining\_characters \leftarrow REMAINING\_CHARS(characters, best\_desc)$
      $KEY(remaining\_items, remaining\_characters)$
    **end if**
  **end for**
**end if**
```

References

[1] Régine Vignes, Jacques Lebbe, and S. Darmoni. Symbolic-numeric approach for biological knowledge representation : a medical example with creation of identification graphs. *Data Analysis, learning symbolic and numeric knowledge*, pages 389–398, 1989.

[2] Paul Jaccard. Étude comparative de la distribution florale dans une portion des alpes et des jura. *Bulletin de la Société Vaudoise des Sciences Naturelles*, (37):547–579, 1901.

[3] R. Sokal and C. Michener. A statistical method for evaluating systematic relationships. *The University of Kansas Scientific Bulletin*, (38):1409–1438, 1958.