

Lexicon Webservice - Training Material | WP3

Author: Mathieu Fannee, INL Internal Review: Katrien Depuydt – November 2013

The INL LexiconService is a webservice that gives any piece of software quick online access to a lexicon by means of http requests. The LexiconService is designed to access a computational lexicon with an Impact Lexicon Database structure.¹ The service is now deployed at INL and gives access to INL's GiGaNT lexicon of 15th to 20th century Dutch.

This webservice offers various possibilities. One can obtain the word forms belonging to a given lemma, or the other way round, one can get the lemma corresponding to a given word form. It is also possible to expand any word with its complete paradigm. And one can limit the results to a given period of history, or to a given part-of-speech. Finally, the lexical information provided by the webservice can be given in both XML or JSON format.

In the following, we'll be describing how requests need to be formulated to get the information you need.

Query basics

To be able to get lexical information from the LexiconService, you need to provide it with at least three things:

- A word
- A lexicon to look up this word in
- And what you need to get in return (a lemma, a paradigm, ...)

Let's start with the lexicon in which the word should be looked up. Let's say we want to access some Dutch lexicon, which is named '*lexicon_service_db*'.² Your request will have to contain this part:

```
...database=lexicon_service_db...
```

Then you have to tell the LexiconService what you need to get. Three distinct operations are possible:

- Get the lemma of a word form
- Get the word forms of a lemma (=its paradigm)
- Expand a word to its complete paradigm (=lemma and all word forms)

Telling the LexiconService which word your question is about can be done in different ways, depending on the operation you need to be performed. So we're going to describe these three possible operations now.

Get a lemma

¹ The current Dutch lexicon consists of a lemmata table with grammatical categories (part of speech), a wordforms table, a attestation table, a document table with metadata (source information and date).

² The list of available lexica depends on the specific LexiconService instance you're using and also on the period of time (as new lexica might be added and old ones removed). Please contact the administrator of the LexiconService instance you are using to get an up-to-date list of the available lexica.

Telling the LexiconService to get the lemma of a word form is done by simply telling 'get_lemma'. The first part of your http requests will therefore look like this:

```
http://sk.taalbanknederlands.inl.nl/LexiconService/lexicon/get_lemma?...
```

You then need to set a few parameters: the word form you want the lemma from, and the name of the lexicon where to look it up. Let's say your word form is *liep* (the Dutch for *walked*). Your complete request will look like this:

```
http://sk.taalbanknederlands.inl.nl/LexiconService/lexicon/get_lemma?  
database=lexicon_service_db&wordform=liep
```

Provided that you want JSON output, the LexiconService will send a response like this:

```
{"lemmata": [  
  {"lemma": "lopen", "pos": "VRB"},  
  {"lemma": "lijp", "pos": "ADJ ADV"}  
]}
```

Get wordforms

Telling the LexiconService to get the word forms of a lemma is done by simply telling 'get_wordforms'. The first part of your http requests will therefore look like this:

```
http://sk.taalbanknederlands.inl.nl/LexiconService/lexicon/get_wordforms?...
```

You then need to set a few parameters: the lemma you want the word forms from, and the name of the lexicon where to look it up. Let's say your lemma is *lopen* (the Dutch for *to walk*). Your complete request will look like this:

```
http://sk.taalbanknederlands.inl.nl/LexiconService/lexicon/get_wordforms?  
database=lexicon_service_db&lemma=lopen
```

Provided that you want JSON output, the LexiconService will send a response like this:

```
{"wordform": [  
  "loop", "loope", "lopen", "loopen", "lope", "loepen", "loope", "liept",  
  ...  
  "liepen", "loopen"  
]}
```

Expand a wordform or lemma

Telling the LexiconService to expand a given word form or lemma to its complete paradigm is done by simply telling 'expand'. The first part of your http requests will therefore look like this:

```
http://sk.taalbanknederlands.inl.nl/LexiconService/lexicon/expand?...
```

You then need to set a few parameters: the lemma you want the word forms from, and the name of the lexicon where to look it up. Let's say your word is *loopt* (the Dutch for *(he) walks*). Your complete request will look like this:

```
http://sk.taalbanknederlands.inl.nl/LexiconService/lexicon/expand?  
database=lexicon_service_db&wordform=loopt
```

Provided that you want JSON output, the LexiconService will send a response like this:



```
{ "wordForm": [
  "eloopen", "gelopen", "gelopen", "ghelopen", "ghelopen", "laupe", "liep",
  ...,
  "loopen", "loopen", "loopen", "loopen", "loopt", "lopen", "lopende"
]}
```

Advanced queries

Limit the search to some part-of-speech

We've seen before how to get the lemma of a given word form. The needed query for the Dutch word form *liep* was:

```
http://sk.taalbanknederlands.inl.nl/LexiconService/lexicon/get_lemma?
database=lexicon_service_db&wordform=liep
```

The result consisted of two very different lemmata, a verb and an adjective/adverb:

```
{ "lemmata": [
  { "lemma": "lopen", "pos": "VRB" },
  { "lemma": "lijp", "pos": "ADJ ADV" }
]}
```

Now imagine you're not interested in adjectives (ADJ), but only in verbs (VRB).³ You can set an extra parameter 'pos' (part-of-speech) to say just that:

```
http://sk.taalbanknederlands.inl.nl/LexiconService/lexicon/get_lemma?
database=lexicon_service_db&wordform=liep&pos=VRB
```

Now the LexiconService output will be:

```
{ "lemmata": [
  { "lemma": "lopen", "pos": "VRB" }
]}
```

Exactly the same can be achieved for the other operations 'get_wordforms' and 'expand'. The request for word forms of lemma *lopen* limited to a part-of-speech *VRB* will look like:

```
http://sk.taalbanknederlands.inl.nl/LexiconService/lexicon/get_wordforms?
database=lexicon_service_db&lemma=lopen&pos=VRB
```

And the request for expansion of *loopt* limited to a part-of-speech *VRB* will be:

```
http://sk.taalbanknederlands.inl.nl/LexiconService/lexicon/expand?
database=lexicon_service_db&wordform=loopt&pos=VRB
```

Limit the search to a period of time

The LexiconService offers the possibility to limit a search to a given period of time. This can be achieved by adding two parameters, 'year_from' and 'year_to'. It is possible to use only one of them, or both at the same time.

³ The parts-of-speech in use in a given lexicon depends on the choices made by the authors of the lexicon in question. Please contact the authors of the lexicon you'd like to use to get a list of relevant parts-of-speech.

Let's say you'd like to get word forms of the word *lopen* before the year 1700, your request will look like:

```
http://sk.taalbanknederlands.inl.nl/LexiconService/lexicon/get_wordforms?
database=lexicon_service_db&lemma=lopen&year_to=1600
```

This will give quite some oldish word forms:

```
{ "wordform": [
  "ghelooopen", "ghelooopen", ...
]}
```

To get the modern word forms (after 1900) of *lopen* instead, we can use 'year_from':

```
http://sk.taalbanknederlands.inl.nl/LexiconService/lexicon/get_wordforms?
database=lexicon_service_db&lemma=lopen&year_from=1900
```

With more modern forms as a result:

```
{ "wordform": ["gelopen", "liep", "lopen", "loopend", "loopende", "loopt"] }
```

Of course, the 'year_from' and 'year_to' parameters can be used together so as to isolate a given period of time. Say we want the paradigm of *lopen* in the period 1600-1700, our request will be:

```
http://sk.taalbanknederlands.inl.nl/LexiconService/lexicon/get_wordforms?
database=lexicon_service_db&lemma=lopen&year_from=1600&year_to=1700
```

Prevent caching

As part of their optimization strategies, some servers might cache requests and responses, in such a way that they can reply faster and with less CPU use when receiving a request they had to process before. A bad thing about this is that if you're working with a growing lexicon, you won't be able to get newly added information for a word you already send a request about.

This can be solved easily, just by adding a 'dummy' parameter with some random number to the http request, in such a way that the request will always look different from requests sent before, even if those were about the same word. For example:

```
http://sk.taalbanknederlands.inl.nl/LexiconService/lexicon/get_lemma?
database=lexicon_service_db&wordform=liep&dummy=1384187319550
```

Output type

The LexiconService can give both XML and JSON output. The output type cannot be set by an explicit parameter of the http request: you have to set it in the AJAX call of the application you're using to connect to the LexiconService.

For example, an AJAX call to the LexiconService written in jQuery will look like this:

```
$.ajax( {
  "type": "GET",

  "url": "../LexiconService/lexicon/get_wordforms",

  "data": {},

  "dataType": "xml", // put xml of json here
```



```
"success": function(xml) {  
    ...  
}  
});
```

Of course, the XML datatype hereabove can be changed for JSON if that is what you need.



Code example to use the LexiconService

The following is a full code example showing how to send requests to the LexiconService and process responses from it. The example is written in jQuery within an HTML test page, but this example can of course be rebuilt in any other language that supports AJAX calls.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
<head>
  <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=9; IE=8; IE=7; IE=EDGE" />

  <title>INL LexiconService Test Page</title>

  <script type="text/javascript" language="javascript" src="js/jquery.js"></script>

  <script>

    // we have a GUI with two buttons to click on, for testing JSON or XML output
    $(document).ready(function(){

      // send a request when the button is clicked upon
      // and send the button id ('xml' or 'json') as required output type
      $(".send_request").click(function(){

        getData( $(this).attr("id") );

      });

    });

    // get webservice output, given the chosen output type (JSON or XML)
    function getData(sOutputType){

      // read which action is required (expand, get lemma, get wordforms)
      var sActionToPerform = $("#action_to_do").val();

      // remove the output of a previous round
      $("#result").empty();

      // read the user input
      var sDatabase = $("#database").val();
      var sWord = $("#input_word").val();
      var sPos = $("#input_pos").val();
      var sYearFrom = $("#input_year_from").val();
      var sYearTo = $("#input_year_to").val();

      // build URL to send the request to
      var sBaseUrl = "../LexiconService/lexicon/";
      sUrl = sBaseUrl + sActionToPerform;

      // default action is 'expand'
      // this action has no pos parameter since some
      // word of the paradigm might have different
      // parts-of-speech
      var aDataToSend = {
        "database": sDatabase,
        "wordform": sWord,
        "year_from": sYearFrom,
```



```

        "year_to":    sYearTo,
        "dummy":    getUniqueNumber() // prevent caching
    };

    // 'get_lemma'
    if (sActionToPerform == 'get_lemma')
    {
        aDataToSend = {
            "database":    sDatabase,
            "wordform":    sWord, // word form needed here
            "pos":        sPos,
            "year_from":    sYearFrom,
            "year_to":    sYearTo,
            "dummy":    getUniqueNumber() // prevent caching
        };
    }

    // 'get_wordforms'
    else if (sActionToPerform == 'get_wordforms')
    {
        aDataToSend = {
            "database":    sDatabase,
            "lemma":    sWord, // lemma needed here
            "pos":        sPos,
            "year_from":    sYearFrom,
            "year_to":    sYearTo,
            "dummy":    getUniqueNumber() // prevent caching
        };
    }

    // send an ajax request to the webservice
    $.ajax( {
        "type": "GET",
        "url": sUrl,
        "data": aDataToSend,
        "dataType": sOutputType, // get response as xml or json
        "success": function( xmlOrJsonOutput ) {

            // send the output to the screen
            var sNeatOutput = getNeatOutput( sOutputType, sWord,
                sActionToPerform, xmlOrJsonOutput );
            print( sNeatOutput );

        },
        "error": function( jqXHR, textStatus, errorThrown ){
            print( "Something went wrong: "+textStatus+" "+errorThrown );
        }
    } );

}

// send some output to the screen (within a SPAN-tag)
function print( str ){
    $("#result").append(
        $("<span></span>").text(str).append($("<br/>"))
    );
}

```

```

// process the output of the webservice

function getNeatOutput( sOutputType, sWord, sActionToPerform, xmlOrJsonOutput ){

    var sOutput = "";

    // process the lemmata if we requested that
    if ( sActionToPerform == 'get_lemma' )
    {
        sOutput += "The word form '" + sWord + "' has the following lemmata: ";
        var aLemmaArray = new Array();

        if (sOutputType == "xml")
        {
            $( xmlOrJsonOutput ).find( "lemmata" ).each(function(){

                var sLemma = $( this ).find( "lemma" ).text();
                var sPos = $( this ).find( "pos" ).text();

                aLemmaArray.push(
                    "'" + sLemma + "' with part-of-speech '" + sPos + "'" );
            });
        }
        else if (sOutputType == "json")
        {
            var aaLemmata = xmlOrJsonOutput.lemmata;

            for (var i=0; i<aaLemmata.length; i++)
            {
                var sLemma = aaLemmata[ i ].lemma;
                var sPos = aaLemmata[ i ].pos;

                aLemmaArray.push(
                    "'" + sLemma + "' with part-of-speech '" + sPos + "'" );
            }
        }

        // gather all lemmata, separated by a comma
        sOutput += aLemmaArray.join( ", " ) + ".";
    }

    // 'get_wordforms' and 'expand' have the same output
    else if ( sActionToPerform == 'get_wordforms' || sActionToPerform == 'expand' )
    {
        sOutput += "The word '" + sWord + "' has the following paradigm: ";

        var aWordforms = new Array();

        if (sOutputType == "xml")
        {
            $( xmlOrJsonOutput ).find( "wordform" ).each(function(){

                sWordform = $( this ).text();
                aWordforms.push( "'" + sWordform + "'" );
            });
        }
        else if (sOutputType == "json")
        {
            aJsonWordforms = xmlOrJsonOutput.wordform;

            for (var i=0; i<aJsonWordforms.length; i++)
            {
                sWordform = aJsonWordforms[i];
            }
        }
    }
}

```



```

        aWordforms.push( "" + sWordform + "" );
    }
}

// gather all word forms, separated by a comma
sOutput += aWordforms.join( ", " ) + ".";
}

    return sOutput;
}

// get a unique number to be added as an arg to the http-requests
// so as to make sure the browser won't cache the request
function getUniqueNumber() {
    return new Date().getTime();
}

</script>
</head>

<body>
    <h2>INL LexiconService Test Page</h2>
    <br><br><br>LexiconName:<input type="text" id="database"
value="lexicon_service_db"></input><br><br><br>
    Type a word (and part of speech if needed) in here, and click a button to test query
    expansion and such within the INL Lexicon webservice.<br><br>

    Word:<input type="text" id="input_word"></input>
    POS:<input type="text" id="input_pos"></input>
    Query:<select id="action_to_do">
        <option value="expand">Expand</option>
        <option value="get_lemma">Get lemma</option>
        <option value="get_wordforms">Get wordforms</option>
    </select><br>
    Year from:<input type="text" id="input_year_from"></input>
    Year to:<input type="text" id="input_year_to"></input>
    <br><br>
    <button type="button" id="json" class="send_request">
    Test JSON output
    </button>
    <button type="button" id="xml" class="send_request">
    Test XML output
    </button>
    <br>
    <br>
    <div id="result"></div>
</body>
</html>

```

