The LuNa Open Toolbox for the Luxembourgish Language

Joshgun Sirajzade and Christoph Schommer

University of Luxembourg
Dept of Computer Science and Communication, ILIAS Lab
Campus Belval, Maison du Nombre,
L-4365 Esch-sur-Alzette, Luxembourg

Abstract. Despite some recent work [17, p. 5], the ongoing research for the processing of Luxembourgish is still largely in its infancy. While a rich variety of linguistic processing tools exist, especially for English, these software tools offer little scope for the Luxembourgish language. LuNa (a Tool for Luxembourgish National Corpus) is an Open Toolbox that allows researchers to annotate a text corpus written in Luxembourgish language and to build/query an annotated corpus. The aim of the paper is to demonstrate the components of the system and its usage for Machine Learning applications like Topic Modelling and Sentiment Detection. Overall, LuNa bases on a XML-database to store the data and to define the *XML* scheme, it offers a Graphical User Interface (GUI) for a linguistic data preparation such as tokenization, Part-Of-Speech tagging, and morphological analysis – just to name a few.

1 Introduction

Luxembourgish is one of the youngest languages of Europe and is a native language for ca. half a million people [7]. Despite the fact that it has more written and digital sources in comparison to other languages of its size, its research is relatively sparse set by side to its neighboring languages like French or German. The same applies to the building of an universal annotated corpus of Luxembourgish, which can be used in research projects, as well as in NLP applications of various kinds. The aim of LuNa is to make a contribution to compiling a corpus for a language with lack of resources, which is also the case for Luxembourgish language.

LuNa's functionality is to tokenize and to standardize a text written in Luxembourgish, e.g., if orthographic variations for words appear. Additionally, LuNa foresees a tagging of words using a POS Tagger. To analyze a text, LuNa supports the search of word formation affixes as well as the annotation of them as such. In the context of the annotation process of word formation affixes, some other analysis can be carried out, for example the analysis of morphological productivity, or the search of the stems of the words with word formation suffixes in the entire corpus [18]. Also, LuNa offers a simple lemmatization for the Luxembourgish language. Several approaches have been investigated, and a hybrid

(rule-based and statistical) lemmatizer is chosen because of its prominent performance for Luxembourgish language. Beside data processing components, first implementations in view of a sentiment analysis and topic modeling exist.

2 Implementation

The backbone of the system bases on a XML-Database eXist[1] (v4.6), which is run on an Ubuntu 14.4 Server. The corpus stored in this database is structured in *TEI-Format* (Text Encoding Initiative, vP5). The frontend consists of an application programmed in Java (v1.8). LuNa can act as a standalone application to process a XML-corpus or as a software client, which operates with the XML-Database.

2.1 Graphical User Interface

The following section describe parts of LuNa's components. A video about LuNa is available here[2].

2.2 TEI

The Text Encoding Initiative (*TEI*) is a kind of *XML* dialect for the organization and structurization of text data. The plain benefit of using *TEI* lies in the fact that it is standardized and well-documented. Thus, it is a format for exchanging data among project participants and external collaborators.

The downside of *TEI* is that *XML* takes more storage space than, e.g., *JSON*. *TEI* has guidelines for organizing metadata of the text and for structuring different kinds of text genres – in particular manuscripts, interviews, and transcription of speeches. This is useful, if a representative corpus of a language is to be build. Below, there is an extract of a header of the poem 'D'Lidd vum Jengsterdag' (English: *the song of yesterday* from Michel Rodange (1827-1876)):

```
<TEI>
 ...
 <titleStmt>
    <title type="main">D'Lidd vum Jengsterdag</title>
    <title type="sub"/>
     <title type="short">D'Lidd</title>
   <author>
    <forename>Michel</forename>
    <nameLink/>
    <surname>Rodange</surname>
    <addName type="pseudonym"/>
   </author>
   <editor>
```

[1] http://exist-db.org/exist/apps/homepage/index.html, last seen on July 17, 2019
[2] https://youtu.be/iLwz8DeJUoI

```
    <forename>Fernand</forename>
    <surname>Hoffmann</surname>
    </editor>
  </titleStmt>
 <publicationStmt>
   <publisher/>
   <pubPlace/>
   <date>1964</date>
 </publicationStmt>
 ...
</TEI>
```

2.3 XML-Database eXist

The XML-database *eXist* (v4.6) is used to manage the data. *eXist* has many in-build tools – such as the browser based IDE eXide (Figure 1) and the standalone Java Admin Client– and is open for public.

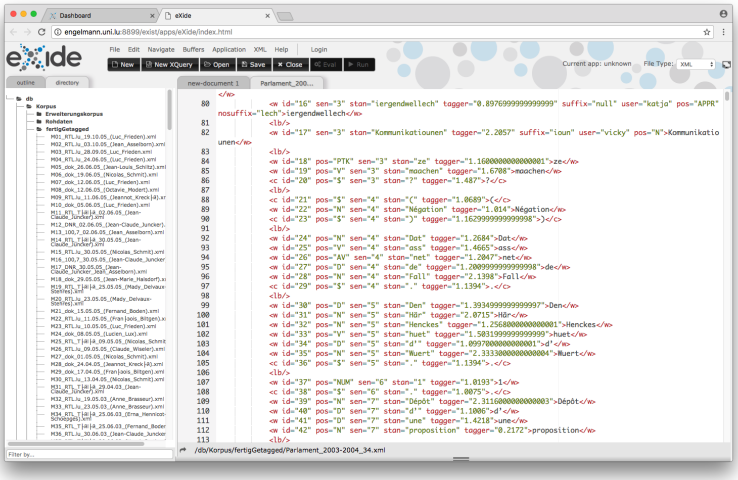

Fig. 1. eXide is a browser based editor provided by eXist. So, LuNa corpus can be viewed with this tool over the internet.

2.4 Tokenization

Luxembourgish texts have different kinds of spelling. Additionally, problems sometimes occur in preprocessing, because some writing applications or computers use different language settings. Thus, when the spelling seems to be correct,

some characters are different according to their unicode encoding. As an example for the last case we name the Luxembourgish article, which can be joined to the following neutral and feminine nouns, like in *d'Wielerin* (English: voter) or *d'Opschwong (English: boost, boom, revival)*. Different Applications from where our Luxembourgish texts are coming have different characters for apostrophes. Therefore, all versions of used apostrophes should be specified in the parameter for word delimiters (Figure 2). Low data quality can sometimes influence the research outcomes in a negative way.

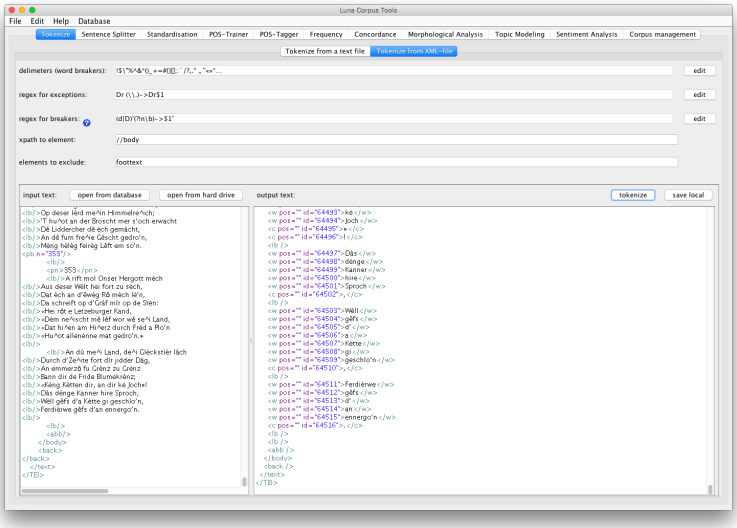

Fig. 2. XML-Tokenizer: Parameters like delimiters for tokenizing can be adjusted. Characters, which can be ambiguous, can be formulated as regular expression, in order to build exceptions.

2.5 Splitting the sentences

Similar to the question what a token is (at least formally) and accordingly how to tokenize a text, there are some debates about splitting the texts into the sentences [13, p. 166]. Without going into details of syntax research and asking for a definition of a sentence or a syntactical unit, it is crucial to point out the benefits of having this kind of information. LuNa uses sentence boundaries a lot, in building concordance lines, POS-tagging, morphological analysis, etc. The tools are delivering better results, if they are using sentence boundaries to carry out further processing, because it is a natural unit for syntactical relationships. The usage of sentence positions in POS-tagging will be discussed bellow. As

Figure 2.5 indicates, the parameter for sentence boundaries – a list of characters that divide sentences – can be specified in the GUI.

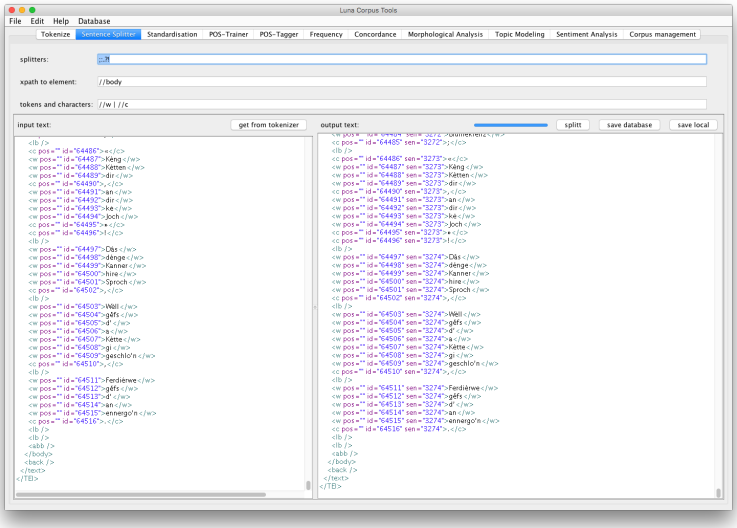

Fig. 3. Sentence splitter: Characters, which can denote sentence borders, can be adjusted. Each sentence gets an attribute, which signals its belonging to a certain sentence.

2.6 Normalization (Standardization)

Text normalization plays a crucial role in text mining, which is represented in the third tab in GUI of LuNa. Like the situation with English words *center* and *centre*, Luxembourgish also contains such variations for many words. However, comparing to English, it is particularly difficult in Luxembourgish text to deal with these variations. On the one hand there is no sufficient amount of text in Luxembourgish to support an automatic retrieval of these variations. On the other hand writing in Luxembourgish has not been fully standardized (also in comparison to German and French languages) for a long time. The official orthography of the Luxembourgish language is relatively young. The situation for the LuNa is peculiar for two reasons: First if one has literature or text in Luxembourgish language that are already older than lets say 50 years, you will already find many differences in spelling. Secondly, the aim of LuNa, is not only its usage in building big research corpora, but also to be used by others, like to be deployed behind language applications. And in such cases a good system for standardisation is very important. LuNa uses for that a table, which can hold regular expressions for normalizing the data (see Figure 4). The original

text stays as always untouched in LuNa, and a new annotation is added with normalized versions for carrying out searches and other investigations. This is because some of the users might indeed be interested in spelling variations. By doing so, no information is lost. Searching *center* may reveal that in some texts *centre* occurs.

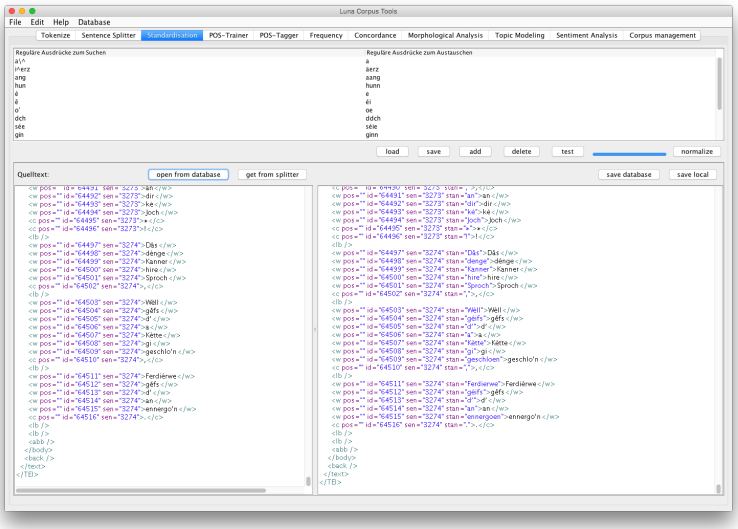

Fig. 4. Text Normalization in LuNa: A list of regular expressions can be specified, in order to normalize the text. The normalized forms are again added as attributes, so the original forms are not lost.

2.7 POS-tagging

Annotating a corpus with part of speeches is one of the most widely used methods in text mining, as well as in corpus linguistics. It is a crucial step for a wide range of tasks with various purposes. A vast amount of research has been carried out in this field and there are already many ready-to-use tools [13, p. 29]. Nevertheless the developers behind LuNa took the chance and the challenge to implement a new one. As there are many models available, it was easy to implement one. Mason[11] shows, how a tagger can be implemented in the programming language Java. Manning and Schütze[10, p. 341] discuss statistical backgrounds of taggers. The history of POS-Taggers began rule based with poor results. But they are performing better, since the machine learning algorithms are applied to them. Especially those like baayesian networks, decision trees and neuronal networks have been successfully applied in POS-tagging [14, 15]. And later on, the so called hybrid approach was introduced, which makes use of rules, where they are

deterministic and classifying in all other cases. The developers of LuNa decided for baayesian statistics and decision trees, sine they are simple to implement and easy to understand. The intermediate results of a training process can be seen with the POS-Trainer component in LuNa (Figure 5). Such a functionality is beneficial for both research and application purposes. Users can repeat the calculations behind the algorithms to a certain point.

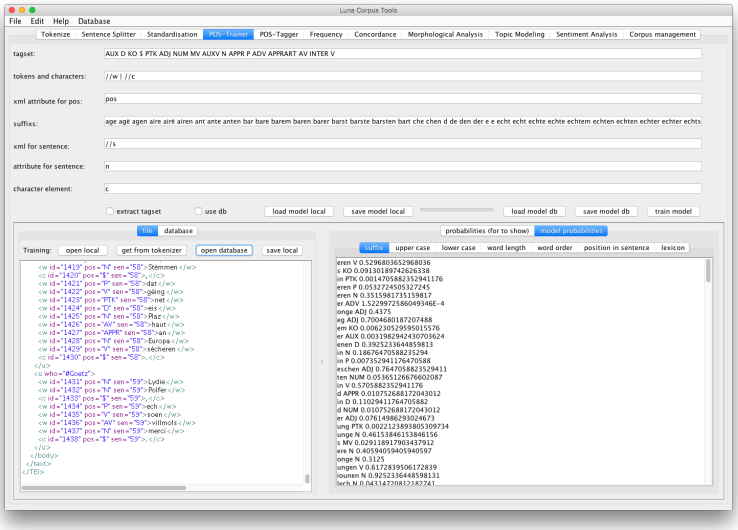

Fig. 5. POS Trainer: Here it is possible to choose training files and see the results of the training process. They are conditional probabilities for given feature. The model can be stored locally or into the database.

2.8 Other Features in POS Tagging

Approaching the problem from the linguistic point of view new features were added. LuNa uses not only the word order which is one of the most popular features used in POS-tagging, e.g. part of speeches like articles and adjectives are normally followed by a noun, but also uses the positions of the words in a sentence. This feature takes the information into account, which part of speeches are likely to occur in the first, second till the last position of the sentence. The reason of using positions is that for languages like Luxembourgish and German, the word order can be relatively flexible. These languages have more morphological means to express grammatical information than english such as declination of nouns. Because of the declination, one can change word order, but the subject and the object will still remain the same. Furthermore, in these Languages the positions for verbs can be at the end of a relatively long sentence. In this case,

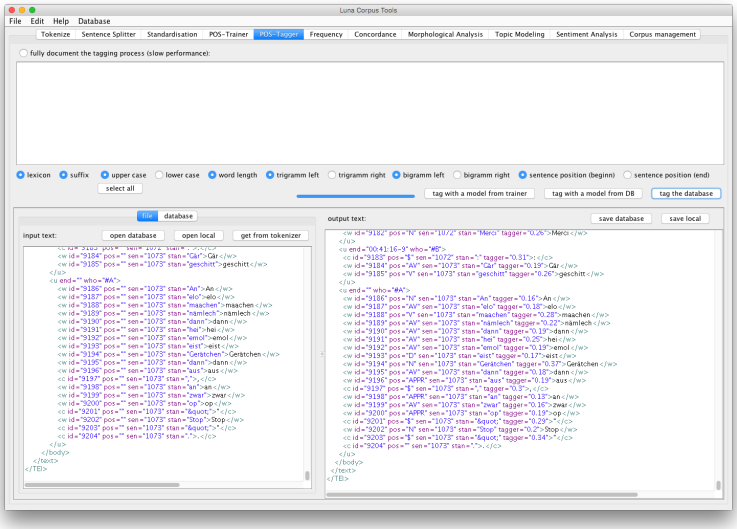

Fig. 6. POS-Tagger: The user can tag new texts with pre-trained models. It is also possible to choose between features to use.

the use of n-gramms is not always sufficient to capture such information. For this purpose, word positions in the sentence might be again very informative.

Another useful feature in Luxembourgish language is the so called 'upper case', because nouns are capitalized regardless of their positions in the sentence. Thus, this indicator is used to distinguish nouns from other word classes. (But this indicator is not useful for words appearing at the beginning of the sentence.) Besides, LuNa POS-Tagger uses rules, which are applied in two steps. 1) Before the tagging process, e.g. numbers are recognized as such or characters in a sentence. 2) After the tagging process. Here sometimes rules help to choose one candidate out of two possible ones. The features for the tagging can be selected in the GUI (Figure 6). After the tagging process the xml file has the following annotations.

```
<lb/>
<w pos="P" id="12" sen="3">Ech</w>
<w pos="V" id="13" sen="3">géing</w>
<w pos="D" id="14" sen="3">d'</w>
<w pos="N" id="15" sen="3">Regierung</w>
<w pos="V" id="16" sen="3">froen</w>
<c pos="$" id="17" sen="3">,</c>
<w pos="KO" id="18" sen="4">ob</w>
<w pos="P" id="19" sen="4">si</w>
<lb/>
<w pos="D" id="20" sen="4">iergendeng</w>
<w pos="N" id="21" sen="4">Kommunikatioun</w>
```

```
<w pos="APPR" id="22" sen="4">un</w>
<lb/>
<w pos="D" id="23" sen="4">d'</w>
<w pos="N" id="24" sen="4">Chamber</w>
<w pos="PTK" id="25" sen="4">ze</w>
<w pos="V" id="26" sen="4">maachen</w>
<w pos="V" id="27" sen="4">huet</w>
<c pos="$" id="28" sen="4">.</c>
<lb/>
```

2.9 Morphological Analysis

Morphological Analysis in LuNa concerns mainly the identifying and annotation of different affixes in the words. So far, the word formation affixes of Luxembourgish could be annotated successfully. Figure 2.9 shows how one can annotate the Luxembourgish prefix *on* in the words like *onbedéngt* (en. unconditionally), or *ongesond* (en. unhealthy). Annotating of morphological information can be useful for many reasons, especially in understanding of the structure of a language [18] or even in practical tasks like language generation etc.

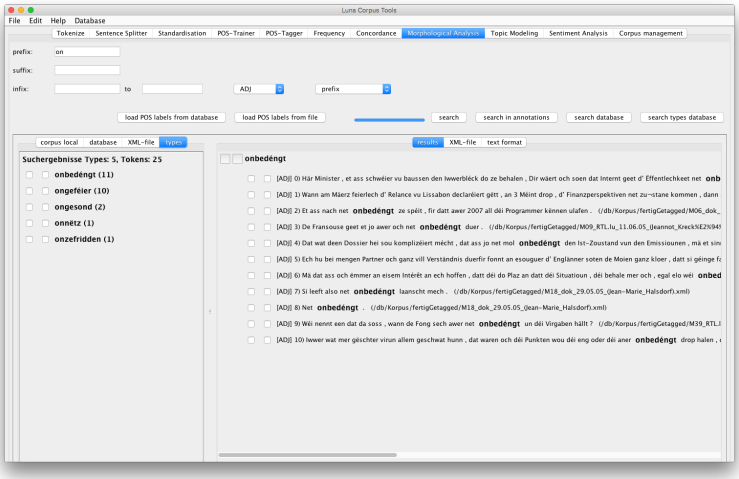

Fig. 7. Finding the tokens with the prefix *on* (en. *un* like in *uncertain*).

2.10 Exploring the Corpus

Under the tab Frequency an X-Path expression can be formulated, in order e.g. to see the frequent used words in the corpus. Because the corpus is well structured,

it is possible to see the word counts per document or in the entire corpus, in order to carry out further analysis. The word counts can also be narrowed down to specific part-of-speeches, for example, it is possible to see the most used nouns or verbs. Figure 2.9 shows the extraction of frequent nouns from parliament speeches.

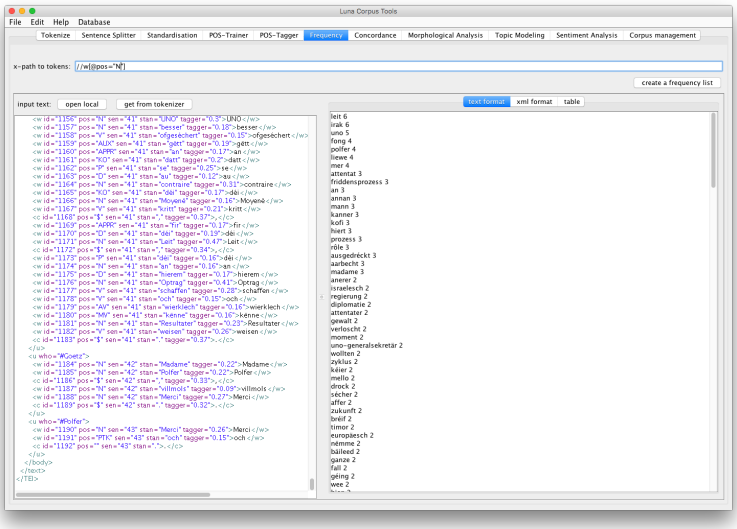

Fig. 8. Frequency analysis with x-path.

2.11 Topic Annotation

LuNa Corpus Tools integrates the ready to use java library MALLET (MAchine Learning for LanguagE Toolkit)[3], which has an implementation of various machine learning algorithms [12]. Luna uses MALLETs module for Topic Modeling, which is an implementation of Gibbs Sampling for Latent Drichlet Allocation [3, 19]. Topic Modeling has already been used successfully in many application cases, e.g. in financial news [16, 9]. [1] discusses the benefits of using topic annotation in corpus building. The obvious benefit of Topic Modeling for law resourced languages lies in its being a unsupervised technique. So, no labelled training set is needed. At the moment, LuNa is able to display the topics extracted from the corpus (Figure 9). The following xml file shows the first five topics with first five words in them extracted from parliament texts.

```
<topic>
```

[3] http://mallet.cs.umass.edu/, last seen on July 17, 2019

```
    <word rank="1" count="35.0">sécher</word>
    <word rank="2" count="16.0">verfassung</word>
    <word rank="3" count="12.0">géife</word>
    <word rank="4" count="12.0">eent</word>
    <word rank="5" count="12.0">hätten</word>
</topic>
<topic>
    <word rank="1" count="6.0">décidéiert</word>
    <word rank="2" count="2.0">fraktur</word>
    <word rank="3" count="2.0">statsfinanze</word>
    <word rank="4" count="2.0">wochen</word>
    <word rank="5" count="2.0">éischte</word>
</topic>
<topic>
    <word rank="1" count="13.0">weisen</word>
    <word rank="2" count="13.0">éischter</word>
    <word rank="3" count="10.0">bon</word>
    <word rank="4" count="8.0">ëffentlech</word>
    <word rank="5" count="8.0">hëllefen</word>
</topic>
<topic>
    <word rank="1" count="4.0">aféieren</word>
    <word rank="2" count="3.0">solle</word>
    <word rank="3" count="3.0">zil</word>
    <word rank="4" count="2.0">gesondheetspolitik</word>
    <word rank="5" count="2.0">populär</word>
</topic>
<topic>
    <word rank="1" count="3.0">schoulen</word>
    <word rank="2" count="3.0">suivi</word>
    <word rank="3" count="3.0">iwwerhaapt</word>
    <word rank="4" count="3.0">kéier</word>
    <word rank="5" count="2.0">iraneschen</word>
</topic>
```

2.12 Sentiment Detection

Some recent advances in sentiment analysis in various social platforms [6, 8] makes its application easy. There are attempts in building corpora with sentiment annotations and their linguistic description [20, 4, 5]. However, low resourced languages have special challenges, the accuracy may drop depending on the size of training data [2]. There is a tab with a functioning sentiment training and analysis in LuNa Corpus Tools. The concrete application case for sentiment analysis for Luxembourgish language emmerged from the collaboration with RTL. The RTL web presence began since the year 2008 allow readers to write commentaries on the news. After then more than half million commen-

taries are written on the RTL homepage for different news, articles and posts. Currently, LuNa uses the LibSVM[4] library, in order to classify the sentiments.

```
<sentence value="positive">
          <w id="97" pos="N" sen="7" tagger="0,2">Et</w>
          <w id="98" pos="AUX" sen="7" tagger="0,18">ginn</w>
          <w id="99" pos="PTK" sen="7" tagger="0,17">net</w>
          <w id="100" pos="AV" sen="7" tagger="0,15">nemmen</w>
          <w id="101" pos="N" sen="7" tagger="0,25">Jempi&apos;en</w>
          <w id="102" pos="AV" sen="7" tagger="0,23">hei</w>
          <w id="103" pos="APPRART" sen="7" tagger="0,16">am</w>
          <w id="104" pos="N" sen="7" tagger="0,56">Land</w>
          <c id="105" pos="$" sen="7" tagger="0,33">.</c>
</sentence>
<sentence value="positive">
          <w id="106" pos="P" sen="8" tagger="0,24">Et</w>
          <w id="107" pos="V" sen="8" tagger="0,26">gin</w>
          <w id="108" pos="AV" sen="8" tagger="0,15">och</w>
          <w id="109" pos="AV" sen="8" tagger="0,19">nach</w>
          <w id="110" pos="ADJ" sen="8" tagger="0,18" value="positive">gudd</w>
          <w id="111" pos="APPR" sen="8" tagger="0,12">an</w>
          <w id="112" pos="ADJ" sen="8" tagger="0,14" value="positive">diplômé&apos;ert</w>
          <w id="113" pos="N" sen="8" tagger="0,4">Studenten</w>
          <c id="114" pos="$" sen="8" tagger="0,31">.</c>
</sentence>
```

3 Tests

In the moment the corpus of Luxembourgish language is composed of two parts. The first part is the fully annotated part of the corpus. It provides rich meta data on the genre, date of origin, author(s) etc. The texts here are fully tokenized, normalized and POS-annotated. Currently this part has ca. 20 mio. running tokens. 10 mio. of these tokens belong to the transcriptions of speeches in the Luxembourgish Parliament (Chambre des Députés) from the year 2003 ongoing, divided in ca. 300 documents. Approximately 5 mio. tokens are gathered from the news from the web presence of RTL (Radio Télévision Luxembourg). These are mainly interviews. Furthermore, there is a part of corpus containing documents from Luxembourgish literature (literature in Luxembourgish language to be precise) with ca. 2.5 mio. running tokens. The rest is also interviews in the Luxembourgish language, but this time, carried out for research purposes at the university of Luxembourg. In the second part of the corpus there are some other 70 mio. token text data from the web presence of RTL, which are already digital and are going to be annotated. These data can be already used for several other purposes, fist of all for statistical analysis. Moreover, we are constantly provided

[4] https://www.csie.ntu.edu.tw/cjlin/libsvm/, last seen on July 17, 2019

with new text data, which must be digitized in order to be used to enrich the corpus.

As mentioned before, using rules improves the performance of the tagger, especially in the case of low resourced languages. Trained with ca. 17 thousand pre-tagged tokens without the rules the tagger gives the accuracy of 87% with decision trees. Applying the mentioned rules improves the accuracy up to 92%, which is not good for languages like English or German, but seems to be sufficient considering the low resourced background of Luxembourgish language.

For Sentiment Analysis training the models with existing relative languages like German, does not help the performance much here. The same applies to the translation of existing English or German resources into Luxembourish, because due to lack of resources the translation in itself does not deliver good performance. That is why, the linguistic department of the university has decided to annotated sentiments manually. Currently, LuNa riches the accuracy of 67%, when trained with 2081 pre-labeled sentences. These sentences contain the classical notation for sentiments; *positive*, *neutral* and *negative*.

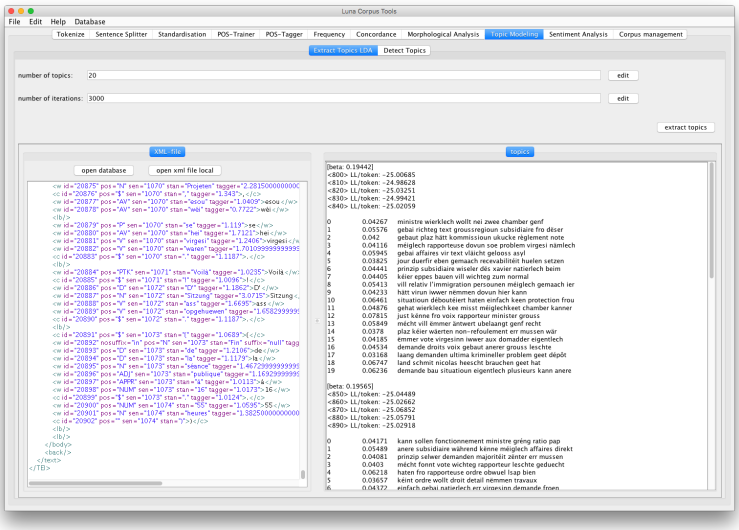

Fig. 9. Topic Modeling with Parliament text.

4 Conclusions and Future Work

LuNa has a strong decline to a corpus building. Many of the procedures like tokenization, normalization, POS-tagging, Lemmatization are a crucial steps in the processing of natural language. LuNa is currently applied in the processing

of news messages (provided by RTL Luxembourg) as well as in the processing of users' comments. Working with such commentaries has its specific problems: firstly, the writing style is much more diverse than it is for the standard Luxembourgish language. Secondly, research fields like, e.g., Sentiment Analysis deliver consequently poor results in such low resourced languages. A training of the models with existing relative languages – like for example German – is not really supportative. The same applies to the translation of existing English or German resources into Luxembourgish, because due to lack of resources the translation in itself does not deliver good performance. Future work concern the support of research disciplines like Topic Modeling or Sentiment Analysis. It should be not only possible to extract Topics from the corpus, but also to store annotations for further usage.

References

1. Akira, M., Paul, T., Susan, H., Dominik, V.: 'what is this corpus about?': using topic modelling to explore a specialised corpus. Corpora **12**(2), 243–277 (2017)
2. Anh Le, T., Moeljadi, D., Miura, Y., Ohkuma, T.: Sentiment analysis for low resource languages: A study on informal indonesian tweets (12 2016)
3. Blei, D.M.: Probabilistic topic models. Commun. ACM **55**(4), 77–84 (Apr 2012)
4. Bolioli, A., Bosco, C., Patti, V.: Developing corpora for sentiment analysis: The case of irony and senti-tut. IEEE Intelligent Systems **28**, 55–63 (03 2013)
5. Bosco, C., Patti, V., Bolioli, A.: Developing corpora for sentiment analysis: The case of irony and senti-tut (extended abstract). In: Yang, Q., Wooldridge, M. (eds.) Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence, IJCAI 2015, Buenos Aires, Argentina, July 25-31, 2015. p. 4188. AAAI Press (2015), http://ijcai.org/Abstract/15/587
6. Feldman, R.: Techniques and applications for sentiment analysis. Commun. ACM **56**(4), 82–89 (Apr 2013)
7. Gilles, P.: Luxembourgish. In: Boas, H.C., Deumert, A., Louden, M. (eds.) Varieties of German Worldwide. Oxford University Press, Oxford (2018)
8. Guo, S., Höhn, S., Xu, F., Schommer, C.: Perseus: A personalization framework for sentiment categorization with recurrent neural network. In: International Conference on Agents and Artificial Intelligence, Funchal 16-18 January 2018. p. 9 (2018)
9. Kampas, D., Schommer, C., Sorger, U.: A hidden markov model to detect relevance in nancial documents based on on/off topics. In: European Conference on Data Analysis (2014)
10. Manning, C.D., Schütze, H.: Foundations of Statistical Natural Language Processing. MIT Press, Cambridge, MA, USA (1999)
11. Mason, O.: Programming for Corpus Linguistics. Edinburgh University Press (2001)
12. McCallum, A.K.: Mallet: A machine learning for language toolkit (2002), http://mallet.cs.umass.edu
13. McEnery, T., Hardie, A.: Corpus Linguistics : method, theory and practice. Cambridge Univ. Press, Cambridge (2011)
14. Schmid, H.: Part-of-speech tagging with neural networks. In: Proceedings of the 15th Conference on Computational Linguistics - Volume 1. pp. 172–176. COLING '94, Association for Computational Linguistics, Stroudsburg, PA, USA (1994). https://doi.org/10.3115/991886.991915, https://doi.org/10.3115/991886.991915

15. Schmid, H., Laws, F.: Estimation of conditional probabilities with decision trees and an application to fine-grained pos tagging. In: Proceedings of the 22Nd International Conference on Computational Linguistics - Volume 1. pp. 777–784. COLING '08, Association for Computational Linguistics, Stroudsburg, PA, USA (2008), http://dl.acm.org/citation.cfm?id=1599081.1599179
16. Schommer, C., Kampas, D., Bersan, R.: A prospect on how to find the polarity of a financial news by keeping an objective standpoint. Proceedings ICAART 2013 (2013)
17. Sirajzade, J.: Das luxemburgischsprachige Oeuvre von Michel Rodange (1827-1876). Editionsphilologische und korpuslinguistische Analyse. doctoralthesis, Universität Trier (2015)
18. Sirajzade, J.: Korpusbasierte Untersuchung der Wortbildungsaffixe im Luxemburgischen. Technische Herausforderungen und linguistische Analyse am Beispiel der Produktivität. Zeitschrift für Wortbildung = Journal of Word Formation **18**(1) (2018)
19. Steyvers, M., Griffiths, T.: Probabilistic topic models. In: Landauer, T., McNamara, S.D., Kintsch, W. (eds.) Latent Semantic Analysis: A Road to Meaning, chap. Probabilistic topic models. Laurence Erlbaum (2007)
20. Stuart, K., Botella, A., Ferri-Miralles, I.: A corpus-driven approach to sentiment analysis of patient narratives. In: Ortiz, A.M., P\'erez-Hern\'andez, C. (eds.) CILC2016. 8th International Conference on Corpus Linguistics. EPiC Series in Language and Linguistics, vol. 1, pp. 381–395. EasyChair (2016)