

# Facilitating Annotation of Collective Agreements with Keywords Extraction

(Update 18/9/2020)

Daniela Ceccon  
WageIndicator Foundation, Amsterdam – CLARIN, Utrecht  
danielaceccon@wageindicator.org

Stefano Ceccon  
PhD, Data Science Lead

## SSHOC Partners



## Join our Community

✉ [info@sshopencloud.eu](mailto:info@sshopencloud.eu)

🔗 [sshopencloud.eu](http://sshopencloud.eu)

🐦 [@SSHOpenCloud](https://twitter.com/SSHOpenCloud)

🌐 [/company/sshoc](https://www.linkedin.com/company/sshoc)

DATA



TOPIC assigned to a  
CLAUSE

Previous  
script:

Find KEYWORDS per topic  
while looking at CLAUSES

Issue: many words are  
common in all the topics.  
We could not tell whether  
a word was more common  
than in the full text.



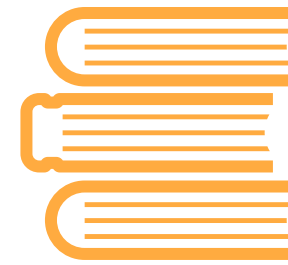
New script:

1. Split the full text into paragraphs.
2. Finds the clauses in the paragraphs and assigns a topic to each paragraph (or no topic)
3. Looks for keywords in the full text (in all paragraphs).

- CLARIN services for the SSH
- The SSH Open Marketplace: it's relevance for the CLARIN communities and CLARIN services
- Social Science and Humanities Open Cloud: why vocabularies matter

Social Sciences & Humanities Open Cloud (SSHOC) is a project funded by the EU framework programme Horizon 2020 and unites 20 partner organisations and their 27 associates in developing the social sciences and humanities area of the European Open Science Cloud (EOSC).

1440 Collective bargaining agreements (CBAs) texts from the WageIndicator CBA Database (since 2012)



[WageIndicator.org/cbadatabase](http://WageIndicator.org/cbadatabase)

28 languages

50+ countries

Annotated: answers to 249 labour rights related questions on 9 topics (eg Employment Contracts, Gender Equality Issues, etc) + clauses selected

## AIM OF THIS WORK



To ease future CBA texts annotation by finding the parts of texts where a question is answered = assign a 'bind' to paragraphs in new CBA texts

## Python script

1

Dataset **TEXTS**  
(.csv dump with all CBA texts in html)

2

Dataset **CLAUSES**  
(.csv dump with all clauses assigned to a question (= 'bind'))

### DATA PROCESSING:

1. We parse texts in paragraphs and create a 'paragraphs dictionary' with languages as keys, containing all the paragraphs for each language.
2. For each clause selected, we check whether it is contained in a paragraph. If that is the case, then the bind is assigned to the paragraph in a new data frame with 1 or 0 identifying whether the bind is assigned to that paragraph or not.

3. We can only do the training on binds that have a sufficient number of assigned paragraphs: we decide for 5 as a minimum.
4. **We perform cleaning (tokenisation - lemmatisation - stop words removal) using NLTK tools (WordNetLemmatizer for English, Snowball Stemmer for other languages).**
5. We add a column with cleaned paragraphs to our data frame.

Dataset **PARAGRAPHS**  
(all cleaned paragraphs with 1 or 0 for each bind)

### FREQUENCY MODELS:

1. Our own model, **relative frequency**:
  - a) Training set: 70% of paragraphs; testing set: 30%.
  - b) We calculate the relative frequency of the 30 most common words for each bind in the training paragraphs with bind and without bind (we exclude from the 30 words the 100 most common words in par. without bind)
  - c) We do the same for the testing paragraphs.
  - d) We plot the ROC curve for each bind on the testing paragraphs to see if the model works well and to determine the optimal threshold.
2. Try with **Bag of Words**.
3. Explore other models of more advanced machine learning (Bert - Par2Vec).

## SSHOC Partners

