QUALIASSISTANT: Extracting Qualia Structures from Texts

Manuel Biertz
Trier University
biertz@uni-trier.de

Lorik Dumani
Trier University
dumani@uni-trier.de

Markus Nilles
Trier University
nillesm@uni-trier.de

Björn Metzler
Trier University
s4bjmetz@uni-trier.de

Ralf Schenkel
Trier University
schenkel@uni-trier.de

Abstract

In this paper, we present QUALIASSISTANT, a free and open-source system written in Java for identification and extraction of Qualia structures from any natural language texts having many application scenarios such as argument mining or creating dictionaries. It answers the call for a Qualia bootstrapping tool with a ready-to-use system that can be gradually filled by the community with patterns in multiple languages. Qualia structures express the meaning of lexical items. They describe, e.g., of what kind the item is (formal role), what it includes (constitutive role), how it is brought about (agentive role), and what it is used for (telic role). They are also valuable for various Information Retrieval and NLP tasks. Our application requires search patterns for Qualia structures consisting of POS tag sequences as well as the dataset the user wants to search for Qualias. Samples for both are provided alongside this paper. While samples are in German, QUALIASSISTANT can process all languages for which constituency trees can be generated and patterns are available. Our provided patterns follow a high-precision low-recall design aiming to generate automatic annotations for text mining but can be exchanged easily for other purposes. Our evaluation shows that QUALIASSISTANT is a valuable and reliable tool for finding Qualia structures in unstructured texts.

1 Introduction

In the field of Natural Language Processing, knowledge bases and thesauri are often used to improve methods such as information extraction (Stevenson and Greenwood, 2006), question answering (Choi et al., 2003), the validation of statements (Hassan et al., 2017) or the generation of arguments (Alshomary and Wachsmuth, 2021; Schiller et al., 2021). In the domain of argument retrieval, for example, there is usually a set of arguments pre-stored in an argument base to improve performance. Given a query entered by a user, these arguments are returned in a ranking (Stab et al., 2018; Wachsmuth et al., 2017). When there are arguments suitable for the query, this ranking works well up to now (Stab et al., 2018). As the arguments are pre-stored in the bases, they are consequently finite and cannot be used for arbitrary queries. With the help of thesauri such as WORDNET, it is possible to modify existing arguments in the base to fit the query.[1] For instance, if the user enters the query *Does every worker get a pension?*, and the most appropriate argument in the base is *Each employee is eligible for pension*, a system applying a thesaurus should be able to recognize, for example, that the category of employees is wider and includes workers. Hence, given the appropriate context, the argument could be evolved to *Each worker is eligible for a pension*. However, thesauri and knowledge bases are also limited because they also have only a finite number of entries, although there could be many more arguments. While they only contain given truths such as that a worker can be seen as employee, this approach could not straightforwardly determine more nuanced facts, like they occur in topics that are discussed in parliaments. For instance, given a query *Should I rely on state pension?*, one result could be that *Pensions are no insurance sum that is simply paid out once*[2], containing valuable though very nuanced knowledge dealing with common misconceptions.

In this paper we present a free and open-source tool called QUALIASSISTANT which is written in Java and helps to create dictionaries, e.g., to mitigate the aforementioned problem of thesauri and knowledge bases.[3] It is based on the work of

[1] https://wordnet.princeton.edu/

[2] This is an actual example from our corpus translated from German.

[3] The datasets as well as a ready to use JAR file are available at https://doi.org/10.5281/zenodo.6805590. The full source code is available at https://github.com/recap-utr/qualiAssistant.

Proceedings of the 9th Workshop on Argument Mining, co-located with the 29th International Conference on Computational Linguistics, pages 199–208
October 17, 2022.

Saint-Dizier (2017), who uses Qualia roles – multi-dimensional aspects of meaning – for automatized argument mining (more on Qualia roles in Section 2) and it answers his call for a "bootstrapping method". One issue with his approach is that it takes a lot of time to acquire a high-quality corpus of such roles (his estimate: several months). While this manual approach facilitates quality control, it is also laborious and should be supported with automatized methods where possible. One such method has been introduced by Cimiano and Wenderoth (2007) in their paper on automatized creation of Qualia roles, using a set of clues, corresponding syntactic patterns (see Section 3), and a large corpus of Web documents in English language. In this paper, we not only enhance their approach so that it becomes applicable to other languages than English (German for instance), we also provide a more extensive list consisting of 142 patterns, and a tool that allows to query arbitrary words such as *pension* in user provided texts in order to extract and return their Qualia roles.[4]

The contributions of this paper are the following:

(1) We introduce the Java application QUALIASSISTANT, which (i) can pre-process texts so that Qualia roles can be found in them and (ii) can also search for Qualia roles in these pre-processed texts.[5] Further, we also explain the abstract concept and the straightforward handling of the application.

(2) We provide a set of 142 patterns to find the four roles as well as two pre-processed datasets in German language which only need queries to be searched in them. Note that QUALIASSISTANT can be easily extended to other languages. Backing up this assertion, we adapted Cimiano and Wenderoth's (Cimiano and Wenderoth, 2007) English patterns and used them to extract Qualia structures in election programs from 2000 to 2020 from seven English-speaking countries.

2 Foundations and Related Work

In this section, we start by defining Qualia structures and discuss related work and how it differs from our work.

Qualia structures were first introduced by Pustejovsky (Pustejovsky, 1991; Pustejovsky and Jezek, 2015) in 1991 as part of his 'Generative Lexicon' (GL) which "emerged from Aristotle's notion of modes of explanation" and structures "lexical semantics knowledge in conjunction with domain knowledge" (Saint-Dizier, 2016). Qualia roles structure multiple layers of meaning around an entity and form a knowledge repository which can be put to good use in many relevant areas in information retrieval (Cimiano and Wenderoth, 2007), including for reference resolution (Bos et al., 1995) and query expansions (Voorhees, 1994). They are also used in areas of natural language processing like argument mining, most prominently by Saint-Dizier (2017).

Pustejovsky's work is based on Moravcsiks's re-interpretation of Aristotle's "doctrine of four causes" as four viewpoints for understanding the meaning of lexical items (*aitia*) (Cimiano and Wenderoth, 2007; Pustejovsky and Jezek, 2015). In Qualia structures, the meaning of a lexical item is thus divided into four roles: *formal*, *constitutive*, *agentive*, and *telic* as shown and described in more detail in Table 1. In our work, the lexical item will be referred to as the *query* (usually a noun) for which we want to find the corresponding *Qualia roles*. For example, given the query *pension*: A formal role would be *income* (what is it?), a constitutive role would be *monthly payments to the retiree* (what does it include?), an agentive role would be *previously regularly paid contributions by the retiree* (what does it need), and a telic role would be *retirement security* (what is it used for?). For the proposed, condensed notation of a Qualia structure see Figure 1. This is an example for a complete Qualia structure with all four roles; but note that not all Qualia structures need all the roles.

$$\text{Pension:} \begin{bmatrix} \text{FORMAL}: [\text{INCOME}], \\ \text{CONSTITUTIVE}: [\text{MONTHLY PAYMENTS}], \\ \text{AGENTIVE}: [\text{REGULAR CONTRIBUTIONS}], \\ \text{TELIC}: [\text{RETIREMENT SECURITY}] \end{bmatrix}$$

Figure 1: An exemplary Qualia structure for *pension*. Notation as proposed by Saint-Dizier (2016).

[4] In their paper, they work with 27 patterns, 13 of which are plural derivatives of other patterns.

[5] Note that the division into (i) and (ii) is made here solely for performance reasons. Reason being is that (i) is computationally intensive, but only needs to be done once beforehand. In contrast, (ii) depends on the query.

[6] Saint-Dizier uses a broader definition stating that all distinguishing features are part of the formal role, though his examples show only categories (Saint-Dizier, 2016).

Table 1: The four Qualia roles of a lexical item describing its semantic meaning.

Qualia role	description
formal	The categories of an entity (Pustejovsky and Jezek, 2015) or its superclass (Yamada et al., 2007) and how it can be distinguished in a larger domain (Cimiano and Wenderoth, 2007) such as orientation, dimensionality, magnitude, shape, or position. For example, a formal role of "dog" would be "animal".[6]
constitutive	Presents the (physical) properties of an object, such as material, weight, components, etc. (Cimiano and Wenderoth, 2007).
agentive	Explains how an entity is brought about, e.g. how it is produced or what its causal chain looks like (Yamada et al., 2007; Cimiano and Wenderoth, 2007).
telic	Expresses the function or purpose of an entity (Saint-Dizier, 2016), e.g. law is used to govern.

According to Saint-Dizier (2016), a general challenge with Qualia structures is that they must be manually retrieved. Saint-Dizier estimates that one would need about 50 Qualia structures to completely represent a knowledge domain (Saint-Dizier, 2016). Acquiring them by hand would be a tedious and laborious task, so he calls for "a bootstrapping method" (Saint-Dizier, 2017). One of the most promising approaches is presented by Cimiano and Wenderoth (2007). They create clues (i.e. general ideas about how terms or combinations of terms indicate a certain role), and patterns for those clues. They then apply those to a large, scraped Web corpus and find Qualia roles for given entities, i.e. nouns, in English. For their gold standard, they rely on 30 words based on the dataset of Yamada and Baldwin (2004), who developed a supervised Machine Learning technique for automatic acquisition of telic and agentive roles. In their approach, they use template-based contextual features extracted from nouns and provide a ranked list of verbs per noun.[7] In contrast to Yamada and Baldwin, the approach of Cimiano and Wenderoth (2007) is more oriented towards high precision results, which is a goal that we share. However, as Machine Learning methods are sensitive to training data, they can only benefit from our approach in the future since they can use the pattern and the automated annotations.

The aforementioned promising approach of Cimiano and Wenderoth (2007) has two main shortcomings which we aim to overcome with this paper: First, there is no ready-made tool available to apply their concept, find additional patterns, and execute the whole pipeline. This greatly hinders its application. Second and related, they provide only English clues and patterns. In this paper, we build upon Cimiano and Wenderoth's approach to answer the call of Saint-Dizier (2017) for a Qualia bootstrapping tool with a ready-to-use system that can be gradually filled by the community with patterns in multiple languages. Since the code is also open-source, it is easy to modify if required.

3 Extracting Qualia Roles

3.1 Concept

In this section, we limit ourselves to explaining the concept behind our Qualia structure extraction pipeline (as illustrated in Figure 2), but make the technical details available on the Web page.[8] Our system is based on the approach of Cimiano and Wenderoth (2007) as it uses *Part-of-Speech* (POS) to identify Qualia structures. Given a CSV file, QUALIASSISTANT derives a *constituency tree* that represents the syntactic structure of each sentence from a column specified by the user. We use constituency trees because they contain more detailed information w.r.t. phrases and their hierarchy than 'flat' POS tagging. For instance, some linguistic units in Figure 3 are encircled by a noun phrase (`NP`) tag.[9] These additional POS tags are crucial for a better division in sub clauses, which we rely on when identifying Qualia roles.

The user is expected to provide some specifications in a JSON file from which the application then extracts important information for further processing. These specifications contain information such as the language which is among others nec-

[7] Yamada and Baldwin continued their research in Yamada et al. (2007).

[8] https://github.com/recap-utr/qualiAssistant.

[9] The POS tags in the figure are the ones CoreNLP uses for the German language and can be viewed here: https://www.coli.uni-saarland.de/projects/sfb378/negra-corpus/negra-corpus.html

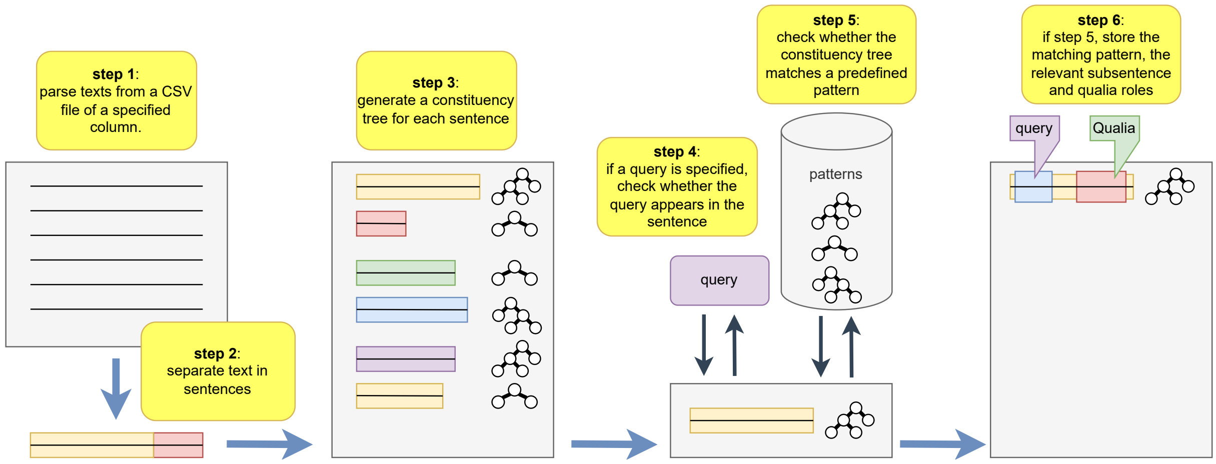

Figure 2: The whole process of QUALIASSISTANT consisting of the dataset pre-processing and the extraction of Qualia roles.

essary to generate the correct constituency trees, paths to input and output CSV files, the name of the column to which the texts should be pre-processed, whether the search for Qualia roles should be limited to a set of queries or be conducted independent from the query, or whether to use stemming (i.e. the heuristic reduction of a word to a common base term) or not to influence precision and recall. A positive example of a match between a pattern and a part of a text can be seen in Figure 3. Then, these entries with matches taken together with the found patterns, the relevant sub-sentences and the identified Qualia roles will be output.

3.2 Rules for Finding POS Tag Sequences

In order to find Qualia roles, we developed a simple convention to search for POS sequences in constituency trees, namely:

1. A *sequence of POS tags* is defined by using **white spaces** between them, e.g.

   `NOUN AUX NP.`

2. In order to allow *multiple selection of POS tags* where exactly one has to be chosen, we use **square brackets**, e.g.

   `NOUN AUX [CNP,NP,NOUN].`

3. For *optional POS tags*, we use **round brackets**, e.g.

   `(DET) NOUN AUX (DET) [CNP,NP,NOUN].`

4. To allow only *specific texts for POS tags*, ideally derived from pre-conceptualized clues, we use **slashes** after the POS tags and write the desired text, e.g.

   `(DET) NOUN AUX/ist (DET) [CNP,NP,NOUN].`[10]

5. The specification of the *Qualia role* and the *query* in a pattern can be done by the use of the XML tags **<qualia>** and **<query>**, respectively, e.g.

   `(DET) <query>NOUN</query> AUX/ist (DET) <qualia> [CNP,NP,NOUN]</qualia>.`

3.3 Finding Qualia Roles

If multiple selection of POS tags is used, the system internally works with derivative patterns. Given the aforementioned pattern (used to find *formal* roles)

```
(DET) <query>NOUN</query>
AUX/ist (DET) <qualia>
[CNP,NP,NOUN]</qualia>
```

QUALIASSISTANT derives 12 ($= 2 \cdot 2 \cdot 3$) different search patterns from this input, i.e., by including or excluding the optional POS tags and picking one

[10] The German word *ist* can be translated to the English word *is*.
Note that specifying the POS tag is essential because a specification could be represented by multiple POS tags and thus semantics could be lost. For example, the term *ist* ('is') could also be a specification for the POS tag VERB.

of the options in the multiple selection.[11] One of these derivatives is

```
<query>NOUN</query>
AUX/ist
<qualia>NP</qualia>.
```

where the two optional POS tags (DET) are not included and NP is chosen to be the searched Qualia role from the multiple selection. With regard to the initial phrase

> *Die Sicherheit ist die Grundlage für Freiheit und Wohlstand.*"
> ('Security is the basis for freedom and prosperity.')

the Stanford CoreNLP library provides the (pretty printed) constituency tree shown in Figure 3. Together with the above-mentioned derivation

```
(ROOT
  (NUR
    (S
      (NP (DET Die) (NOUN Sicherheit))
      (AUX ist)
       (NP (DET die) (NOUN Grundlage)
         (PP (ADP für)
           (CNP (NOUN Freiheit)
             (CCONJ und)
             (NOUN Wohlstand)))) )
    (PUNCT .)))
```

Figure 3: Constituency tree of the sentence *Die Sicherheit ist die Grundlage für Freiheit und Wohlstand.* ('Security is the basis for freedom and prosperity.).'

```
<query>NOUN</query>
AUX/ist
<qualia>NP</qualia>
```

of the pattern (the pattern's components are highlighted with double underlining), as well as the query *Sicherheit* (highlighted with wavy underlining; 'security') we get a match with the subsentence

> *die Grundlage für Freiheit und Wohlstand*
> ('the basis for freedom and prosperity')

In the constituency tree, the Qualia role consisting of the content inside the NP tag is highlighted with a yellow background and can be extracted for further usage such as creating and updating knowledge bases or argument mining. QUALIASSISTANT finds matching sequences and, if the query matches, it outputs the Qualia role in form of its leaves by traversing the tree. We thus derive that the term *Sicherheit* ('*security*') contains the formal role *die Grundlage für Freiheit und Wohlstand* ('the basis for freedom and prosperity').

4 Evaluation

In this section, we outline our evaluation, where we measured the performance of QUALIASSISTANT in comparison to a baseline on two datasets quantitatively and qualitatively. To this end, we involved a human annotator.

4.1 Dataset

As already remarked in Section 1, we tested QUALIASSISTANT on two German datasets and one English dataset. One of the German datasets consists of parliamentary speeches from the German Bundestag with 1,367,655 sentences (henceforth: OFFENESPARLAMENT).[12] The other represents written language and consists of user-generated arguments occurring in a forum of German petitions with 124,034 sentences (henceforth: OPENPETITION).[13] These datasets differ fundamentally in language style but they both belong to the domain of politics so they are generally comparable. The difference in language style allows for more meaningful conclusions in the evaluation. While the former dataset is available in JSON format, we scraped the debates of the latter source and converted both into CSVs so that the desired texts could be easily identified and processed based on the corresponding specified column label. The English dataset consists of 179,398 sentences which originate from election programs from 2000 to 2020 from seven English-speaking countries. In addition, we developed a file consisting of 142 search patterns for the German language. For English, we adapted the search patterns from Cimiano and Wenderoth (2007), to show that our application can be used for languages other than German. We intend for these files to be extended and adapted in future work.

[11] Line breaks are added only for improved readability.

[12] https://offenesparlament.de/

[13] https://www.openpetition.de/

4.2 Setup

Since QUALIASSISTANT is able to find Qualia structures for queries as well as independent from queries, we conducted our evaluation for both. W.r.t. the query dependent approach, we selected 52 German query terms with high political relevance in Germany such as *Rente* ('pension'), *Sicherheit* ('security'), or *Bildung* ('education') and obtained 869 Qualia structures for the dataset OFFENESPARLAMENT and 207 Qualia structures for OPENPETITION. W.r.t. the query independent approach, we let the system search for queries and Qualia roles and obtained 16,090 Qualia roles for 5,210 different identified queries for OFFENESPARLAMENT as well as and 2,811 Qualia roles for 1,833 different queries for OPENPETITION.

Since our preliminary experiments showed that single terms are not always useful as queries, e.g. because they only make sense in context (as with genitive constructions), we expanded the queries by traversing the constituency tree if they are surrounded by a certain tag - for the German patterns we took `NP`. In this way, for the example in Figure 3, we only get the extension *Die Sicherheit* ('The security') because there are no other `NP` tags placed higher. However, assuming the query was *Grundlage* ('basis') in the same example, then the extended query *Die Grundlage für Freiheit und Wohlstand* ('the basis for freedom and prosperity') would be derived. Thus, we obtained triples of the form (query, expanded query, Qualia role), where "expanded query" is the top level of extensions for a given tag. Since some queries did not include an expanded query, we ignored them for the evaluation of the four datasets. Further, we did not include more than 10 triples for each query for each the two query dependent datasets for the evaluation. Thus, w.r.t. the query dependent approach, we obtained 231 Qualia structures for the dataset OFFENESPARLAMENT and 89 Qualia structures for OPENPETITION. From each of the two sets following the query independent approach, we drew a random sample of 50 Qualia structures that also include expanded queries. Thus, our final evaluation set consists of 420 (=231+89+50+50) (query, expanded query, Qualia role) triples. Figure 4 visualizes the setup to obtain the evaluation set. Among these, 414 hold the role "formal", 4 hold the role "constitutive" and 2 hold the role "agentive". Obviously, formal roles appear most often. On the one hand, this is due to the fact that we were able to find more patterns for these roles because these are easier to determine. On the other hand, this is a result of the texts in which formal roles are more likely to be found. However, we do not see this as a disadvantage since we are currently only using small datasets as a proof of concept. In the future, we plan to use huge datasets like Wikipedia and datasets of different domains, where we can find the other roles as well.

4.3 Baseline

Since we are not aware of any existing system for finding Qualia structures, we developed an intuitive baseline. For a given query, this baseline first searches for all subtrees of the constituency tree that branch off `NP` tags and then randomly picks one in which the query does not occur. If no query is given, a random noun is chosen from the original input sentence and used as the query for the process described above. The intuition behind this is the following:

(1) Queries are mostly nouns. Thus, it makes sense to randomly select a noun as a query if none is supplied by the user. Apart from that, the number of nouns in a sentence is moderate.

(2) Qualia roles to queries are mostly noun phrases. Hence, it makes sense to select such as Qualia roles. Particularly in this case, the number of noun phrases in a sentence is small as there cannot be that many noun phrases of this kind, since we are only considering constituency trees of sentences here.

We ran this baseline system for all of the four datasets. W.r.t. the two query independent datasets, similar to our proposed system, we drew a random sample of 50 triples for each. W.r.t. the two query dependent datasets, we obtained 142 triples for OFFENESPARLAMENT and 70 for OPENPETITION. Overall, this resulted in 312 (=50+50+142+70) triples delivered by the baseline system. Assuming that this process provided reasonable Qualia structures, the approach still fails in assigning the roles agentive, constitutive, formal, and telic. Since our approach mostly retrieved formal roles, we assigned the role formal for the baseline to each found Qualia structure so that the annotator should not be able to recognize from the assigned role which system returned it.

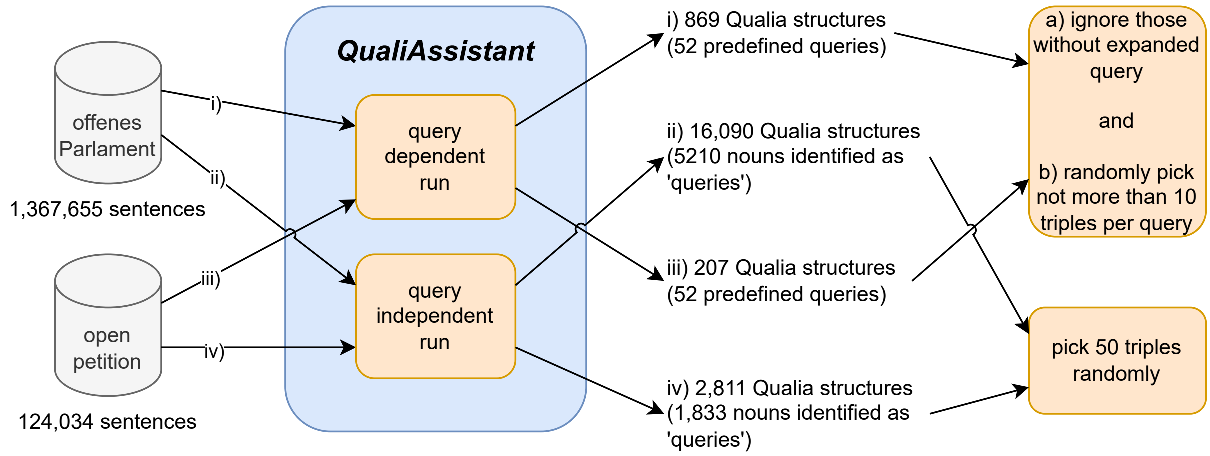

Figure 4: Setup to obtain the final evaluation set.

4.4 Annotation

Then, we aggregated the 732 (=420+312) triples in a single file and asked a human annotator to assess these triples with respect to their meaningfulness on a three fold scale. The annotator is a doctoral student in computer science working on computational argumentation for more than four years. Note that we veiled the triples' origins to the annotator and shuffled the order. For both query and expanded query, the annotator assigned the value `0` with respect to the Qualia role if the role does not make any sense to be included in a text mining process. For example, if the query was *poverty* and the Qualia role was *work*. If the Qualia role was a perfect fit without compromises and could be seen as a gold standard in further systems, the annotator assigned the value `2`. For example, if the query was *employment* and the (constitutive) Qualia role was *work*. If the Qualia structure could be seen as tenable (even with only little drawbacks) the annotator assessed it with the value `1`. This could be, e.g., when the Qualia role to the query was in the genitive form.

4.5 Results of the Quantitative Evaluation

We measured the annotations with micro average precision. Table 2 shows these results for standard as well as expanded queries for the two datasets OFFENESPARLAMENT and OPENPETITION. There, we not only distinguish between query dependent and query independent searches but also compare perfect pairs (label = `2`) to pairs that need to be processed (label = `1` and label = `2`) to be included in a database.

We can observe from the table that our method performs significantly better than the baseline for both datasets and for both query dependent and query independent searches. We can also notice that for each method and dataset, the results improve by expanding the queries with their context. The query dependent search seems to perform better on the dataset OFFENESPARLAMENT, which consists of sophisticated texts, while the query independent search performs better on the dataset OPENPETITION which consists of user-generated content. Nevertheless, these results should be treated with some caution, since only one annotator carried out the assessments. It is therefore more important to pay attention to the tendency, which clearly shows the value of QUALIASSISTANT, and less to the absolute numbers.

4.6 Results of the Qualitative Evaluation

Some observations of the annotations were that queries without context rarely make sense, and thus it is good to provide additional contexts. For example, for the query *Armut* ('poverty') we get the wrong Qualia role *Arbeit* ('work'). Adding the extension of the query, we obtain the expanded and reasonable query *Das beste Mittel gegen Armut* ('the best remedy for poverty'). Sometimes queries without context also made sense, but then there is the risk of losing the semantics of the text. For example, for the query *Gesundheit* ('health') we get the Qualia role *keine Nebensächlichkeit* ('no minor matter'), which is completely reasonable. However, expanding the query automatically to include context provides the expanded query *Schutz der Gesundheit* ('protection of health'), which provides a more accurate description of the text. Therefore, we recommend including the expanded query when intending to reflect the text for future tasks.

Table 2: Precision values for standard as well as expanded queries for the datasets OFFENESPARLAMENT and OPENPETITION when only **perfect** matching pairs (label = 2) are considered to be true positives (left side) as well as when also **tenable** matching pairs (label = 1 and label = 2) are considered to be matching pairs (right side). The upper part shows the results for the **query dependent** search. The lower part shows the results for the **query independent** search.

query dependent search	**method**	**consider only perfect pairs** OFFENESPARLAMENT	OPENPETITION	**consider also tenable pairs** OFFENESPARLAMENT	OPENPETITION
✓	QUALIASSISTANT$_{\text{expanded}}$	**0.714**	**0.551**	**0.887**	**0.73**
✓	QUALIASSISTANT$_{\text{standard}}$	0.377	0.36	0.736	0.64
✓	BASELINE$_{\text{expanded}}$	0.218	0.186	0.415	0.414
✓	BASELINE$_{\text{standard}}$	0.134	0.057	0.415	0.357
✗	QUALIASSISTANT$_{\text{expanded}}$	**0.46**	**0.64**	**0.8**	**0.8**
✗	QUALIASSISTANT$_{\text{standard}}$	0.22	0.34	0.5	0.6
✗	BASELINE$_{\text{expanded}}$	0.02	0.06	0.04	0.16
✗	BASELINE$_{\text{standard}}$	0.0	0.0	0.04	0.14

In order to shed more light on the numbers of Table 2, we randomly picked a sample of the four datasets to qualitatively inspect the results and will now discuss these impressions.

In general, we can state that formal roles can be found very well in German texts. Agentive and constitutive roles also provide quite good but few results. We also found Qualia structures which could be declared as telic, but they were classified as formal. Nevertheless, in the found cases the results are also valid as formal roles, underscoring the general ambiguity of language(s). For example, the query *Arbeit* ('work') provided the formal Qualia role *ein wichtiger Hebel für Integration* ('an important lever for integration') as it also matched such a pattern. However, this role could also be considered telic, since the role not only shows what it is (formal), but also what it is needed for (telic). At least in German, there is a need for much more specific telic patterns catching different grammar, such as active and passive. The current telic patterns could not throw any results, underscoring that development of good patterns remains a continuous task and challenge.

Since the dataset OPENPETITION, contrary to the dataset OFFENESPARLAMENT, is user-generated, there were occasionally grammatical errors made by users, thus leading to minimally incorrect constituency trees. Still, the extracted Qualia roles were semantically meaningful, although they could include syntactical errors. For example, the query *Abtreibung* ('abortion') yielded the Qualia role *Mord an einem Menschen das* ('Murder of a human being the') since there was no punctuation mark between two sentences in the corresponding text and the second sentence started with *das* ('the'), so only this article was appended to the end of a Qualia role.[14]

Apart from that, we noticed that good queries can produce properly good results, some of which we will briefly mention below. For example, for the query *Rente* ('pension') we get the Qualia role *keine Versicherungssumme , die einfach nur einmal ausbezahlt wird* ('no insurance sum that is simply paid out once') and for the query *Subsidiarität* ('subsidiarity') we get the role *der Violinschlüssel dafür, dass dieser Ausgleich in angemessener Form gelingen kann* ('the treble clef for this compensation to succeed in an appropriate form'). These cases show that QUALIASSISTANT is able to grasp a complex political issue. We noticed that in particular those roles that contain a negation seem to contain strong political demarcations. The query *Bildung* ('education'), for example, returns the Qualia role *Schlüssel zu einem erfolgreichen Berufsleben* ('key to a successful professional life'), which could as well have been passable as telic. The query *Staatsverschuldung* ('national debt') yields, among other things, the Qualia role *die Basis für eine weiter schlechte wirtschaftliche Entwickung* ('basis for still deteriorating economic development'), which can also be seen as an argument, e.g., when asking an argument search engine for reasons against increasing national debts. Note that argument search engines work in such a way that they take queries such as "*should national debts be taken care of?*" as input and output a list of ranked arguments either supporting or attacking the query's topic. In this case *basis for still deteriorating economic development* would be an attacking argument. That means that QUALIASSISTANT

[14]This issue disappeared after updating CoreNLP to version 4.4.0, i.e., CoreNLP correctly recognized the superfluous term *the* and does not include it in that sentence anymore.

might also help in argument mining tasks (see Section 1).

For the query *Parlament* ('parliament') we got the Qualia role *keine Versammlung von Helden und Heiligen* ('not a gathering of heroes and saints'), which on the one hand shows that the application is able to reflect the point of view of the texts. On the other hand, it shows that it is a find that would most likely not be found in such a way in a knowledge base or thesaurus.

If the query was not carefully designed (or automatically assigned if the user does not provide any), the results could become more noisy. For example, for the query *Redner* ('speaker') we get Qualia roles such as *the colleague* x (where x is a name of a politician), which offers no genuine added value. Especially with references we noticed problems in the German language at an early stage. For example, for the query *Krankheit* ('illness') the role *ein Grundbedürfnis* ('a basic need') was assigned, which is nonsense. In fact, the original sentence is *Die Versorgung bei Krankheit ist ein Grundbedürfniss* ('the care in case of illness is a basic need'). This kind of false results can be easily removed by taking only sentences that start with the desired pattern. However, in the future we want to enable researchers to explore other possibilities, namely how the results change when the context of a query is automatically included. Thus, we added an additional column in the output that contains expanded queries. As the manual investigations in the last section showed this query expansion immensely helps to understand the relationships between query and Qualia role.

5 Conclusion and Future Work

In this paper, we presented QUALIASSISTANT for finding Qualia structures in texts. In our evaluation utilizing two different datasets in German and one in English, we showed that our approach works reasonably well. However, in this research, the most challenging part remains the gathering of patterns in the form of POS sequences for texts. This is an ongoing process that can take years of development to become sophisticated. We hope for the community to assist in this process by contributing to updating the patterns in our repository since Qualia structures are important for many NLP tasks such as argument mining which is a growing area of research.

In the future, we aim to identify Qualia structures on larger datasets such as Wikipedia using our application, e.g., to set up knowledge bases with them, or for mining arguments. Furthermore, we will investigate the development of further patterns in multiple languages as we believe that many fields of research benefit from having Qualia structures with high-precision, e.g., the validation of statements. Moreover, we want to develop Machine Learning methods that are able to find new Qualia Roles by making use of the automated annotations. We will also improve usability, for example by returning complete Qualia objects as query response.

Acknowledgements

The authors sincerely thank both anonymous reviewers 2 and 3 for their invaluable feedback.

This work has been funded by the Deutsche Forschungsgemeinschaft (DFG) within the projects ReCAP and ReCAP-II, Grant Number 375342983 - 2018-2024, as part of the Priority Program "Robust Argumentation Machines (RATIO)" (SPP-1999).

References

Milad Alshomary and Henning Wachsmuth. 2021. Toward audience-aware argument generation. Patterns, 2(6):100253.

Johan Bos, Paul Buitelaar, and Anne-Marie Mineur. 1995. Bridging as coercive accommodation. arXiv preprint cmp-lg/9508001.

Key-Sun Choi, Jae-Ho Kim, Masaru Miyazaki, Jun Goto, and Yeun-Bae Kim. 2003. Question-answering based on virtually integrated lexical knowledge base. In Proceedings of the Sixth International Workshop on Information Retrieval with Asian Languages, 2003, Sappro, Japan, July 7, 2003, pages 168–175. ACL.

Philipp Cimiano and Johanna Wenderoth. 2007. Automatic acquisition of ranked qualia structures from the web. In ACL 2007, Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics, June 23-30, 2007, Prague, Czech Republic. The Association for Computational Linguistics.

Naeemul Hassan, Gensheng Zhang, Fatma Arslan, Josue Caraballo, Damian Jimenez, Siddhant Gawsane, Shohedul Hasan, Minumol Joseph, Aaditya Kulkarni, Anil Kumar Nayak, Vikas Sable, Chengkai Li, and Mark Tremayne. 2017. Claimbuster: The first-ever end-to-end fact-checking system. Proc. VLDB Endow., 10(12):1945–1948.

James Pustejovsky. 1991. The generative lexicon. Comput. Linguistics, 17(4):409–441.

James Pustejovsky and Elisabetta Jezek. 2015. A Guide to Generative Lexicon Theory.

Patrick Saint-Dizier. 2016. Argument mining: the bottleneck of knowledge and language resources. In Proceedings of the Tenth International Conference on Language Resources and Evaluation LREC 2016, Portorož, Slovenia, May 23-28, 2016. European Language Resources Association (ELRA).

Patrick Saint-Dizier. 2017. Knowledge-driven argument mining based on the qualia structure. Argument Comput., 8(2):193–210.

Benjamin Schiller, Johannes Daxenberger, and Iryna Gurevych. 2021. Aspect-controlled neural argument generation. In Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2021, Online, June 6-11, 2021, pages 380–396. Association for Computational Linguistics.

Christian Stab, Johannes Daxenberger, Chris Stahlhut, Tristan Miller, Benjamin Schiller, Christopher Tauchmann, Steffen Eger, and Iryna Gurevych. 2018. Argumentext: Searching for arguments in heterogeneous sources. In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 2-4, 2018, Demonstrations, pages 21–25. Association for Computational Linguistics.

Mark Stevenson and Mark A Greenwood. 2006. Learning information extraction patterns using wordnet. In Proceedings of the 5th Intl. Conf. on Language Resources and Evaluations (LREC), pages 95–102.

Ellen M. Voorhees. 1994. Query expansion using lexical-semantic relations. In Proceedings of the 17th Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval. Dublin, Ireland, 3-6 July 1994 (Special Issue of the SIGIR Forum), pages 61–69. ACM/Springer.

Henning Wachsmuth, Martin Potthast, Khalid Al Khatib, Yamen Ajjour, Jana Puschmann, Jiani Qu, Jonas Dorsch, Viorel Morari, Janek Bevendorff, and Benno Stein. 2017. Building an argument search engine for the web. In Proceedings of the 4th Workshop on Argument Mining, ArgMining@EMNLP 2017, Copenhagen, Denmark, September 8, 2017, pages 49–59. Association for Computational Linguistics.

Ichiro Yamada and Timothy Baldwin. 2004. Automatic discovery of telic and agentive roles from corpus data. In Proceedings of the 18th Pacific Asia Conference on Language, Information and Computation, PACLIC 18, Waseda University, Tokyo, Japan, December 8-10, 2004. ACL.

Ichiro Yamada, Timothy Baldwin, Hideki Sumiyoshi, Masahiro Shibata, and Nobuyuki Yagi. 2007. Automatic acquisition of qualia structure from corpus data. IEICE transactions on information and systems, 90(10):1534–1541.