Transitivity Analysis Annotation Scheme and Its Application to CQPweb: A Study on Stephen King’s Carrie

Amanda Hanna Astriyanti\(^a\), Prihantoro\(^b\)

\(^a\)English Department, Faculty of Humanities, Diponegoro University, 50274, Indonesia
\(^b\)English Department, Faculty of Humanities, Diponegoro University, 50274, Indonesia

amandahanna232@gmail.com
prihantoro@live.undip.ac.id

Abstract

In this study, I aim to create a Transitivity Analysis Annotation Scheme (TAAS) within which, process types and clause patterns are taken into account. Once created, TAAS is applied to corpus data, obtained from a novel entitled Carrie [1]. The scheme is manually implemented using XML format. The XML formatted corpus is converted to a format accepted in CQPweb [2], a corpus query tool. The corpus is tokenised using the Perl tokenizer, in TreeTagger [3]. Once tokenised, the corpus is tagged using English TreeTagger [3], which supplies two grammatical annotations (Part of Speech and Lemma). The annotated corpus is then indexed to CQP web. Once indexed, users are able to search the annotated corpus using transitivity analytic labels, POS tags and lemma, in CQP web.

Keywords: transitivity analysis; annotation; corpus linguistics; CQPweb.

1. Introduction

The general English clause describes a particular process (what kind of activity is involved or what is going on) and a particular participant (who is involved in the process) [4]. This is similar to Halliday’s Systemic Functional Linguistics (SFL), whose analyses are centralised on clauses. SFL provides a reference for linguists to understand language as involving acts of meaning [5]. One of the SFL analytic systems is transitivity. The transitivity system constructs any participant roles depending on its process verb(s) in a clause. Process type is a verbal group in the transitivity system [6]. Subsequently, the participant is a nominal group in the transitivity system, also functionally labelled as an entity [6]. In the arrangement of experience patterns, participant elements also determine the process of happening and doing (material process), sensing (mental process), saying (verbal process), behaving (behavioural process), being (relational process), and existing (existential process), by analysing its clauses. As one of the widely used analytic systems
in SFL, transitivity analysis may be used to examine how certain intrinsic elements (character, plot, theme, and setting) are presented in the literary works [7], [8] or to analyse the figure’s speech using discourse analysis approach [9].

However, none of these studies focuses on the technical aspect of annotation (adding analytic labels to analytic linguistic units) i.e., how to make the annotated text becomes easily accessed. In this study, I will apply transitivity analysis using XML format [10] to be indexed in CQPweb [2] a user-friendly corpus tool. The aim of this study is to propose an annotation scheme of transitivity analysis (TAAS), which may be used as a reference for researchers to carry out SFL studies, using corpus annotation techniques. In line with the results of this study, it also aims to demonstrate how a corpus is annotated and indexed using XML format in CQPweb.

2. Research Methods

2.1 Create and Annotation Scheme

To classify the data using the transitivity features, I created an annotation scheme of transitivity analysis by encoding every feature in each process type to form clause patterns. The scheme is drawn using a chart editor from UAM Corpus Tool version 3.3 developed by O’Donnell (2008).

2.2 Collect Clauses to be Analysed using the Scheme

The unit of analysis in this study is a clause, which Wiranto [11] believes as the actualization of experience patterns by the participants of a particular process type. To understand the experiential meaning embedded in these clauses, I took sample texts from a novel entitled Carrie (1974) written by Stephen King.

2.3 Apply the Scheme to the Data Clauses using XML Format

The eXtensible Markup Language (XML) is a system of markup to corpus texts using <angle brackets> [2]. In this study, I will use XML annotation to encode SFL analyses. In the application of XML, there is a basic rule that tags must appear as boundaries to achieve a well-formed XML file. XML tags are composed of two angle brackets that surround a label, for instance, <sample>. XML can be used to indicate certain features of a text, such as paragraph boundaries, clause boundaries, sentence boundaries, etc. For instance, the <C> tag indicates clause boundaries from its actual text as follows:

\[ \text{The student council committee held a football match at my school} \]  \[ \text{My friend, Sarah, competes against underclassmen.} \]  \[ \text{She accepted a defeat that was only 1 point apart.} \]
XML tags consist of start-tag and end-tag, which serves as boundaries in the region of text [2]. Inside the start-tag, we can include information about the region features after the tag label, for instance, the transitivity analysis. To incorporate the feature, we use attribute-value pairs. Hardie [2] stated that “each attribute has one single value (enclosed in straight quotation), attribute and value are delineated using an equal sign.” In this study, I have done manual annotations on the data using a transitivity analysis scheme. Next, I encoded the analysis in an XML document format. A sample of XML annotated clauses as follows:

<table>
<thead>
<tr>
<th>Actor</th>
<th>Material Process</th>
<th>Goal</th>
</tr>
</thead>
<tbody>
<tr>
<td>My friend</td>
<td>competes against</td>
<td>underclassmen</td>
</tr>
<tr>
<td>&lt;SFLC analysis=&quot;TS&quot; ProcessType=&quot;MA&quot; pattern=&quot;APG&quot;&gt;my team competes against underclassmen&lt;/SFLC&gt;</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

2.4 Convert the Texts to CQPweb Readable Format

Using the Perl tokenizer, in Tree Tagger [3], a corpus text can be indexed to CQPweb if the format complies with the CQPweb-readable format. It is very similar to the XML format in terms of the tags. The main difference is that each token must be presented vertically, as shown in Table 2 below.

<table>
<thead>
<tr>
<th>XML format</th>
<th>CQPweb format</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;text id=&quot;A1&quot;&gt;Sarah is the best football player.&lt;/text&gt;</td>
<td>&lt;text id=&quot;A1&quot;&gt;Sarah is the best football player.&lt;/text&gt;</td>
</tr>
</tbody>
</table>

Once tokenised, the corpus is tagged using English TreeTagger [3], which supplies two grammatical annotations (Part of Speech and Lemma). Annotations for each token can
be added in the subsequent column. For instance, in the italicised section as shown by Text 1 below, the first column is the word form (e.g., ‘player’), the second column consists of the POS tag (e.g., NN), and the third column consists of the lemma (e.g., ‘play’).

<table>
<thead>
<tr>
<th>word level/ token</th>
<th>POS tags</th>
<th>lemma</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sarah</td>
<td>NP1</td>
<td>unknown</td>
</tr>
<tr>
<td>is</td>
<td>VBZ</td>
<td>be</td>
</tr>
<tr>
<td>the</td>
<td>AT</td>
<td>the</td>
</tr>
<tr>
<td>best</td>
<td>RGT</td>
<td>best</td>
</tr>
<tr>
<td>Player</td>
<td>NN</td>
<td>play</td>
</tr>
</tbody>
</table>

Text 1. Sample text in verticalized format

2.5 Index the Texts as Corpus in CQPweb

The Corpus Query Processor (CQP), a web-based tool for indexing corpora, was created by a lecturer at Lancaster University [2]. In this study, I apply CQPweb to index a text, into which SFL analyses are encoded in XML format. Once a corpus is indexed, CQPweb allows users to analyse it using a variety of analytical tools, such as collocation, distribution, categorising, sorting, frequency breakdown, thinning, etc (see Figure 1).
3. Results and Discussion

3.1 Transitivity Analysis Annotation Scheme (TAAS)

The annotation scheme that I propose is called Transitivity Analysis Annotation Scheme - SFL, abbreviated as TAAS (see Figure 2). It is visualised using the tree structure below and composed of categories and their corresponding category labels (grey segment). Users will later be able to search the annotated corpus using the labels. For instance, by supplying TS into the query box, users will be able to search all linguistics units, annotated using the transitivity analysis labels.
Transitivity System can further be subcategorized into 6 categories based on the process type at the clause level, namely material (MA), behavioural (BE), mental (ME), verbal (VE), relational (RE), and existential (EX). Further subcategorization of the patterns for each clause may vary according to the participant functions, present in each process type. The table below consists of process type categories (see Description) with their corresponding labels (see ProcessType) and example clauses.
Table 3. Description of process types

<table>
<thead>
<tr>
<th>ProcessType</th>
<th>Description</th>
<th>Example Clauses</th>
</tr>
</thead>
</table>
| MA          | Material Process          | 1. Desjardin placed the napkin, cleaned the girl up with wet paper towels and got her back into her plain cotton underpants  
2. she slapped Carrie |
| BE          | Behavioural Process       | 1. Miss Desjardin made an irritated cranking gesture  
2. She frowned |
| ME          | Mental Process            | 1. Desjardin could only look at her helplessly  
2. She still believed that she thought all children were good. |
| VE          | Verbal Process            | 1. I’ll talk to them tomorrow  
2. She tried twice to explain the commonplace reality of menstruation |
| RE          | Relational Process        | 1. her shorts were blinding white  
2. A terrible and black foreknowledge grew in Rita Desjardin's mind. |
| EX          | Existential Process       | 1. There was a bright flash overhead  
2. there was silence |

Each category can further be detailed using its corresponding Clause Pattern, encoded as pattern. For instance, I took one of ProcessType as a sample, i.e., MA (material process). The Table 5 below shows examples of clause pattern from MA. Unique labels for each pattern are present, namely APX, APG, APS, APC, and APR.

All clause patterns begin with actor (A) and process (P). Thus, each label begins with AP. They can be followed by other functions such as goal (G), scope (S), Client (C), and Recipient (R). If the additional participant is absent, then X is appended. See full descriptions of the labels from the pattern of other ProcessType in the appendix.

Table 4. Material clause patterns
pattern | Description | Example clauses
--- | --- | ---
APX | actor + process | The thought **trailed off** -
APG | actor + process + goal | She **took** one of the sanitary napkins
APS | actor + process + scope | Miss Desjardin **employed** the standard tactic
APC | actor + process + client | Miss Desjardin **led** Carrie
APR | Actor + process + recipient | She **handed** him

3.2 XML Annotation Format

Presently, the only tag for further XML analysis is SFLC (SFL analysis at clause level). For the attributes and values, the following labels are used (see Table 5). The headings are used as attributes, while the labels are used as values. As a reference for readers to understand TAAS easily, I compiled the tagset above into a table below.

**Table 5. Tagset of Transitivity Analysis**

<table>
<thead>
<tr>
<th>Attributes</th>
<th>analysis</th>
<th>ProcessType</th>
<th>patterns</th>
</tr>
</thead>
<tbody>
<tr>
<td>TS (Transitivity System)</td>
<td>MA (Material Process)</td>
<td>APX</td>
<td>APG</td>
</tr>
<tr>
<td></td>
<td>BE (Behavioural Process)</td>
<td>APS</td>
<td>APC</td>
</tr>
<tr>
<td></td>
<td>ME (Mental Process)</td>
<td>APR</td>
<td></td>
</tr>
<tr>
<td></td>
<td>VE (Verbal Process)</td>
<td>RE (Relational Process)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>EX (Relational Process)</td>
<td>TPE</td>
<td></td>
</tr>
</tbody>
</table>
Based on the table of the transitivity tagset above, the main attribute of the SFLC tag is analysis, which is equipped with a value called TS. Consider an example clause: Desjardin said. From its verb, the clause can be categorised as a verbal process (VE). According to the type of clause, it can further be categorised into SPX as it is composed of a sayer and a process, without any additional participant. The labels in the XML annotation sample can be described as follows.

1. First of all, we will carry out an SFL analysis at clause level. For instance, Desjardin said is a clause that will be analysed using the SFL analysis system. For this reason, we supply the following XML tag <SFLC>. We are required to make sure the proper XML tag is supplied with both the start and end tag.
   <SFLC>Desjardin said</SFLC>

2. Then, we need to incorporate the type of SFL analysis that we are going to apply to this clause, which is the transitivity system (TS). For this reason, we need to add the attribute analysis that informs the type of analysis as TS. It must be supplied as an attribute-value pair inside the start tag.
   <SFLC analysis="TS">Desjardin said</SFLC>

3. Once, we decided that this clause is going to be analysed by the transitivity analysis system. We will need to determine the ProcessType of the clause, which in this example is a verbal process on the basis of its verb. Therefore, we need to supply the following attribute-value pair ProcessType="VE" to indicate that this clause belongs to the verbal clause.
   <SFLC analysis="TS" ProcessType="VE">Desjardin said</SFLC>

4. The final step is annotating the pattern of this clause. As you can see, the clause consists of only a sayer and its process without any additional participant. For this reason, the proper label for the clause pattern is SPX. It is a value of pattern, which refers to the category of verbal clause pattern. See full descriptions of the labels from the pattern of other ProcessType in the appendix.
   <SFLC analysis="TS" ProcessType="VE" pattern="SPX">Desjardin said</SFLC>

5. Note that the relational clause patterns can be categorised into identifying and attributive, whose values are TPV, VPT, and CPA (see Table 5). Earlier TPV
(token + process + value) and VPT (value + process + token) fall into identifying categories, while CPA (carrier + process + attribute) falls into attributive categories.

An example of the application of TAAS, can be observed in Text 2 as a part of the selected data from the novel Carrie in Part I BLOOD SPORT. Overall, the sample text contains several clauses marked with clause-level analytic tags, as shown in <SFLC>...</SFLC>. Pity and self-shame met in her is a sample clause in Text 2. The clause contains attribute-value pairs inside the start tag showing that this clause has been analysed using different categories or processes. TS (transitivity system) refers to the type of analysis. It shows that this clause is analysed using transitivity analysis. Subsequently, according to the ProcessType, this clause belongs to ME (mental process). The category clause pattern of this mental process is PPS (phenomenon + process + senser): Pity and self-shame is the phenomenon, met is the process, and her is the senser.

Text 2. Part I BLOOD SPORT

This XML Annotation Scheme may not necessarily be accepted by all corpus tools. Commonly, an annotated text is modified following a format, acceptable in a tool used to index the corpus. In this project, I will show how to convert the document format from native XML into the CQPweb readable format, as shown in the subsequent section.

3.3 Adaptation to CQPweb

The CQPweb program requires that a text is presented vertically. This stands in contrast to the previous text format, in which the XML annotations are presented in a horizontal form (see Text 3). As shown below, we can see that each line consists of a token (word or punctuation or symbol) or a tag (start or end).
Pity and self shame met in her.

There was a bright flash overhead.

**Text 3. Verticalized text of BLOOD SPORT.vrt**

The clause inside the XML tags is considered as a group of tokens, which must appear in the first column. In the columns separated by tab characters, each token can be supplemented by optional annotations [6]. They can be POS (part of speech) tags, lemmas, semantic tags, as well as additional SFL analysis annotations. Consider a sample clause, *There was a bright flash*, from the Text 3.

<table>
<thead>
<tr>
<th>word level/ token</th>
<th>POS tags</th>
<th>lemma</th>
</tr>
</thead>
<tbody>
<tr>
<td>There was a bright flash</td>
<td>EX VBD DT JJ NN</td>
<td>there be a bright flash</td>
</tr>
</tbody>
</table>

**Text 4. Sample clause with POS and Lemma annotation**
Following the Part-of-Speech tags in the Penn Treebank Project [9], the token *was* is encoded as *VBD*, which refers to copula verbs (e.g. *is, am, are, was, were*). Under the lemma, there exists *be* as the lemma form of *was*.

### 3.4 POS Tags Based Search

After converting the document format from native XML into the CQPweb readable format, the text containing the transitivity analysis annotations is ready to index on CQPweb. The indexed corpus appears as Transitivity Corpus SFL: powered by CQPweb, available from https://cqpweb.lancs.ac.uk/transitivity/. In this section, I will show how to search the corpus data from the CQPweb query box.

To search the corpus using POS (Part of Speech) tags, users have to add an underscore before the POS tag, here Penn Treebank POS tags [9]. For instance, a user wants to search for a group of verbs, specifically past tense. Then, 

\[ _{VBD} \]

needs to be supplied into the query box.

![Figure 3. POS tags based search](image)

### 3.5 TAAS Tags Based Search

To search the annotated corpus using TAAS (Transitivity Analysis Annotation Scheme) tags, users have to supply a CQP syntax query into the query box. On the query mode options, users need to select *CQP Syntax*. To search using CQP Syntax, a user must understand three labels: tag, attribute, and value. For instance, if we want to find all clauses annotated using TS, we also need to understand its attribute on the tag (see *Table 5. Tagset of Transitivity Analysis*). The CQP syntax query is shown as follows.
Figure 4. TAAS based search

\[
\text{<SFLC> a:[] []* </SFLC> :: a.SFLC\_analysis = "TS"}
\]

The above query can be used as a template. The users just need to change the attribute and value. The tag for SFL search is always SFLC. Thus, it does not require any modification. Below is the corpus data concordance of annotated corpus within TAAS.
4. Conclusion

This study aims to demonstrate the creation of the Transitivity Analysis Annotation Scheme (TAAS), how a corpus is annotated in XML format using TAAS and indexed into CQPweb. The results of this study show that users are able to search the annotated corpus using the TAAS tags in CQPweb. The quantity of this corpus is indeed small, as it is considered a new project in the Linguistics Department FIB UNDIP. Subsequent studies can be geared towards improving the quantity, by accumulating TAAS annotated corpus into CQPweb. In line with the aim of TAAS, this annotation scheme can also be combined with other analytical systems, such as Mood, appraisal, theme, etc. Thus, this research is sustainable for further research that discusses system analysis in SFL.

References


