Syntactic Simplification Patterns for Clausal and Phrasal Disembedding

May 8, 2018

1 Definition Guidelines

The set of hand-crafted transformation rules are based on syntactic and lexical features that are obtained from a sentence's phrase structure, which is generated by using Stanford's pre-trained lexicalized parser [14]. In all cases, these rules make use of regular expressions over the parse trees encoded as Tregex patterns [5]. The definitions of our transformation rules were heuristically determined by analyzing common syntactic structures in hundreds of sentences from Wikipedia.

The main guideline for this task was to provide a best-effort set of rules, targeting the challenge of being applied in a recursive fashion and to overcome biased or incorrect parse trees. Since our approach strongly relies on the output of the lexicalized parser, we aim to compensate for erroneous parse trees by *implementing simplification rule patterns as shallow as possible*, meaning that they preferably operate at the top (root) of the phrasal parse tree instead of accessing deep nested structures. Those nested structures are then exploited by recursive parsing. In order to limit the amount of rules, we aimed for a general set of rule patterns that represent classes of common syntactic constructs that cover different syntactic variations, instead of using a more restrictive approach where each syntactic variant has to be specified explicitly. Regarding the recursive process, a crucial factor was the order in which the set of rule patterns were checked against a sentence's parse tree. During our experiments, we heuristically developed a fixed order that worked best for our purpose.

2 Characteristics of the Rule Patterns

Every rule pattern has the following characteristics: it accepts a sentence's parse tree as an input and encodes a certain parse tree pattern that, in case of a match, will extract textual parts out of the tree that are used to produce the following information:

Simplified sentences The simplified sentences are generated by combining extracted parts from the parse tree.

Constituency Our notion of constituency depicts the semantic relevance between the simplified sentences. If all sentences can be considered to be equally important, we use the term *coordination*. In this case, all generated sentences are labeled as *core* sentences. Otherwise, i.e. if one sentence provides background information or further specifies a core sentence, we use the term *subordinate* and label the sentence as *context* sentence. Just like in Rhetorical Structure Theory (RST) [6], the constituency assignment is directly related to the identified rhetorical relation which defines a certain manifestation of constituency for the connected spans. Therefore, the tasks of determining constituency and classifying rhetorical relations can be hardly separated and are usually carried out simultaneously [3].

Classified rhetorical relation Both syntactic and lexical features are used to classify rhetorical relations that hold between the simplified sentences. Syntactic features are manifested in the phrasal composition of a sentence's phrasal parse tree, whereas lexical features are extracted from the parse tree as so-called *signal spans*. Each signal span represents a sequence of words that are likely to include rhetorical cue phrases (e.g. "*because*", "*after* *that*" or "*in order to*") which indicate a certain rhetorical relation. The way these signal spans are extracted from a sentence's parse tree is specific for each rule pattern. The extracted signal spans are then used to infer the kind of rhetorical relation. For this task, we use a predefined list of rhetorical cue phrases which are assigned to their most likely triggered rhetorical relation. If one of these cue phrases is present in the signal span, the corresponding rhetorical relation is set between the newly constructed sentences. Note that this also affects the constituency, which is defined in a relation-specific manner. If the extracted signal span does not match any cue phrase from the list, the default relations *Unknown-Coordination* or *Unknown-Subordination* are set. Some of the implemented rule patterns are explicitly tailored to identify specific relations that heavily rely on syntactic features such as *Purpose* relations [15]. In this particular case, no signal spans are extracted.

The implemented rule patterns were manually defined for the task of recursive sentence simplification on the basis of common syntactic structures, including *coordinations*, *subordinations* and sentences linked by *conjunct adverbs* [4, 11]. In addition to clausal disembedding, we also consider phrasal disembedding to further recurse on the simplified sentences. These rules encompass *coordinated verb phrases*, *participial phrases*, *coordinated noun phrase lists* and *purposes*. Besides, the sentence simplification system proposed in [10] is used to extract further phrasal components.

The extraction of signal spans from those structures was grounded on the research of [9], who classified explicit discourse connectives into the three classes of *subordinating conjunctions*, *coordinating conjunctions* and *adverbial connectives*. Information about the position, order and characteristics of discourse connectives was obtained from the annotation guidelines of the PDTB [11]. Besides, the mapping of cue phrases to rhetorical relations was adapted from the research of [15], who investigated how rhetorical relations are signaled in the PDTB.

3 Transformation Rules

Below, we depict an extensive selection of the implemented transformation rule patterns. This selection is supposed to show representatives of the the main syntactic constructs that are processed by our framework, rather than giving a full specification of every syntactic variant. For each rule, its Tregex pattern is visualized as a tree structure that matches over a sentence's phrasal parse tree. Single characters, such as x, z_i or w, are used as variables to label optional, unconstrained textual spans (indicated by dotted triangles).

3.1 Coordinated Clauses

In coordinated syntactic structures that are manifested as coordinated clauses, each clause is disembedded from the input sentence and transformed into a self-contained simplified sentence. In order to prevent discarding contextual information (e.g. in prepositional phrases), each simplified sentence concatenates its clause with the textual spans z_1 and z_2 that occur before and after the clausal coordination. In the case of two coordinated clauses, the signal span is extracted as the textual span that occurs between them.

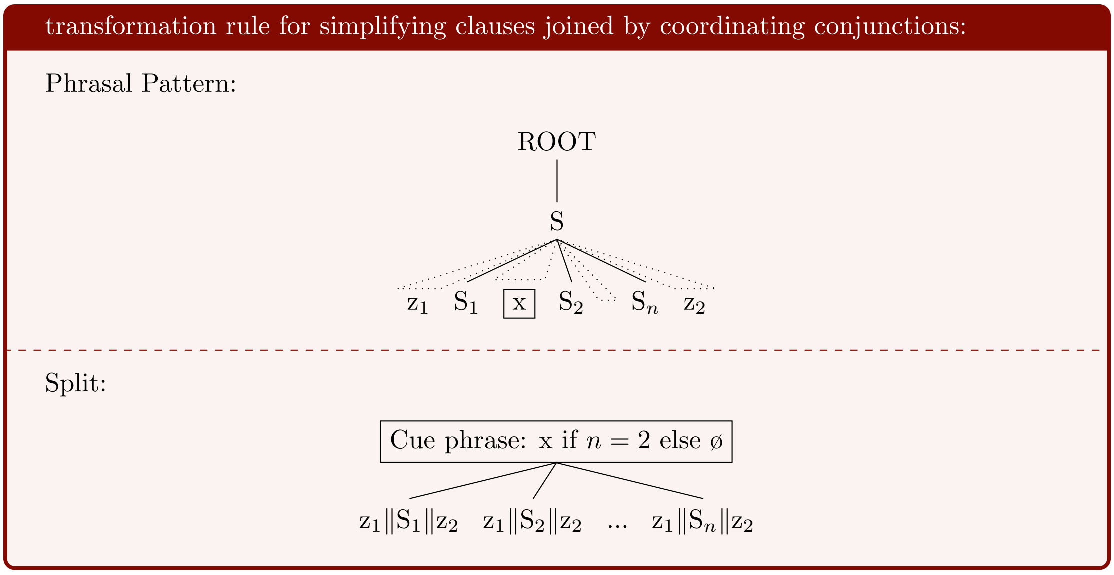

transformation rule for simplifying clauses joined by coordinating conjunctions:
Phrasal Pattern:
ROOT
S
z1
S1
x
S2
Sn
z2
Split:
Cue phrase: x if n = 2 else ø
z1||S1||z2
z1||S2||z2
...
z1||Sn||z2

Figure 1: Rule for coordinated clauses

3.2 Subordinated Clauses and Attributions

In case of sentences that present subordinated structures, two simplified sentences are generated. One sentence corresponds to the superordinate statement, whereas the second sentence embodies the subordinate clause. Syntactic variations of such structures cover the following linguistic expressions, differing in the order of their clausal components: 1) the subordinate clause follows the superordinate span; 2) the subordinate clause precedes the superordinate span; or 3) the subordinate clause is positioned between discontinuous parts of the superordinate span. In all cases, the discourse connective (*cue phrase*) is present inside of the subordinate clause *SBAR* [11]. As an example, the implementation of a rule pattern where the subordinate clause follows the superordinate span is shown in Figure 2.

In addition, special attention was given to attribution relationships, e.g. "*The pilot announced [that there had to be made a special technical check of the aeroplane]*$_{SBAR}$", which, too, fall into the syntactic category of subordinations. To distinguish such relationships from other subordinate constructions, we have crafted additional rule patterns for attributions. Similar to [8] and their Open IE system Ollie, we identify attributions by matching the lemmatized version of the head verb of the sentence (here: "*announce*") against a list of verbs of reported speech and cognition [3].

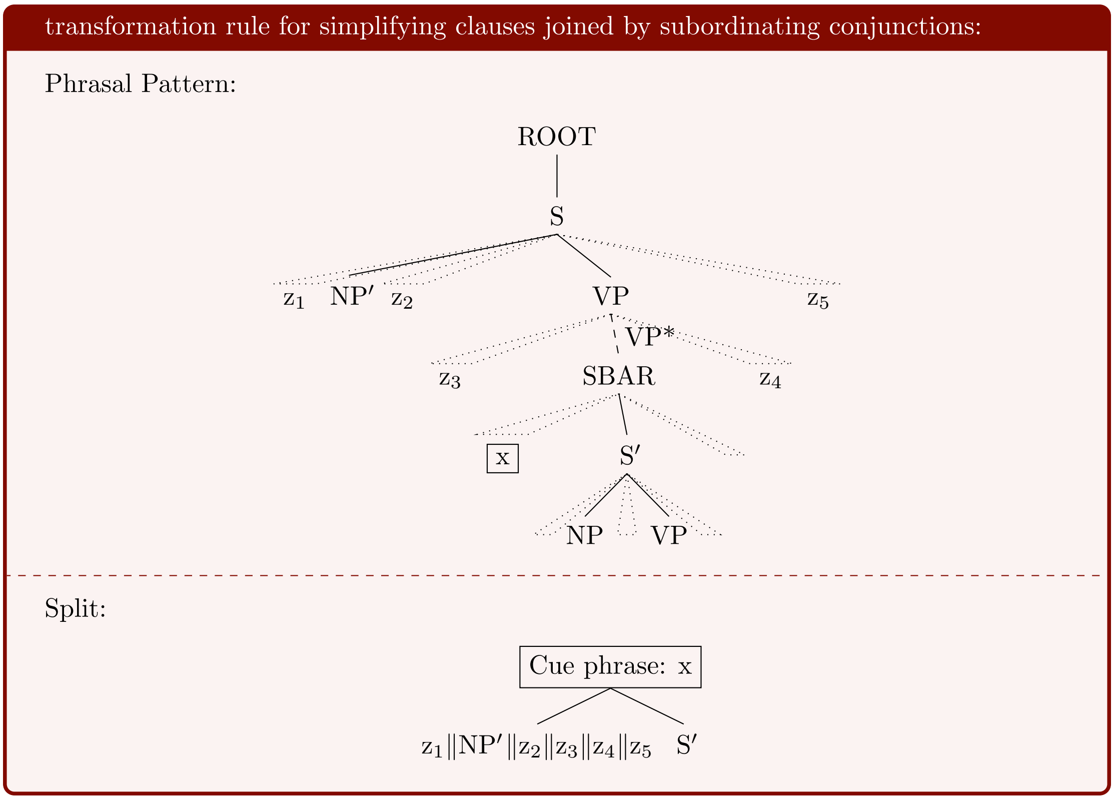


Figure 2: Rule for subordinated clauses with closing subordinative clauses

3.3 Relative Clauses

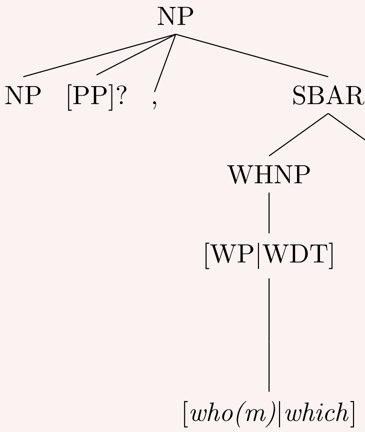
A relative clause is one that is attached to its antecedent by a relative pronoun. There are two types of relative clauses, differing in the semantic relation between the clause and the phrase to which it refers that may be either restrictive or non-restrictive. In the former case, the relative clause is strongly connected to its antecedent, providing information that identifies the noun it modifies (e. g. *”Obama criticized leaders who refuse to step off.”*) Thus, it supplies essential information and therefore cannot be eliminated without affecting the meaning of the sentence. In contrast, non-restrictive relative clauses are parenthetic comments which usually describe, but do not further define their antecedent (e. g. *”Obama brought attention to the New York City Subway System, which was in a bad condition at the time.”*), and hence can be left out without disrupting the meaning or structure of the sentence [12]. In short, restrictive relative clauses are an integral part of the phrase to which they are linked and thus should not be separated out of the sentence. Consequently, only non-restrictive relative clauses are detached from the main clause and transformed into self-contained context sentences by our simplification model. As non-restrictive relative clauses are usually set off by commas - unlike their restrictive counterparts - they can be easily distinguished from one another on a purely syntactic basis.

In the course of the previously conducted rule engineering process, the constituency parse trees of hundreds of sentences containing non-restrictive relative clauses have been analyzed in order to identify a general pattern which typically signifies this type of clause. That way, we have determined that the antecedent of such a relative clause must be a noun phrase that is - if any - usually only separated from it by a prepositional phrase, succeeded by a comma. Thus, the following pattern commonly indicating a non-restrictive relative clause has been deduced: NP [PP]? , SBAR, whereby the subordinate clause has to be introduced by a relative pronoun. In this connection, the relative pronouns *who, whom, which* and *where*, as well as a combination of a preposition and one of the pronouns named above are factored in by our text simplification framework. Two exemplary simplification rules treating non-restrictive relative clauses commencing with the pronouns *who, whom* or *which* (see figure 2.1) and *where* (see figure 2.2), respectively, are shown below.

Figure 3: Example simplification rules for non-restrictive relative clauses

transformation rule for simplifying non-restrictive relative clauses introduced by the pronouns *who(m)* or *which*:

Clausal Pattern:



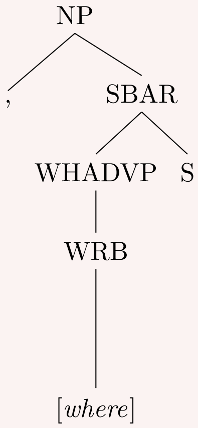
Extracted Context:

NP + [PP]? + S + .

(2.1) Relative pronouns *who, whom, which*

transformation rule for simplifying non-restrictive relative clauses introduced by the pronoun *where*:

Clausal Pattern:



Extracted Context:

’There’ + S + .

(2.2) Relative pronoun *where*

3.4 Coordinated Verb Phrases

If a verb phrase consists of multiple coordinated verb phrases, each verb phrase is disembedded and attached to the shared noun phrase, thus generating two or more simplified sentences with reduced sentence length. In this way, we aim to increase both minimality and recall of the later extracted relational tuples. Analogous to the rule of coordinated clauses, the signal span is extracted as the textual span between two coordinated verb phrases.

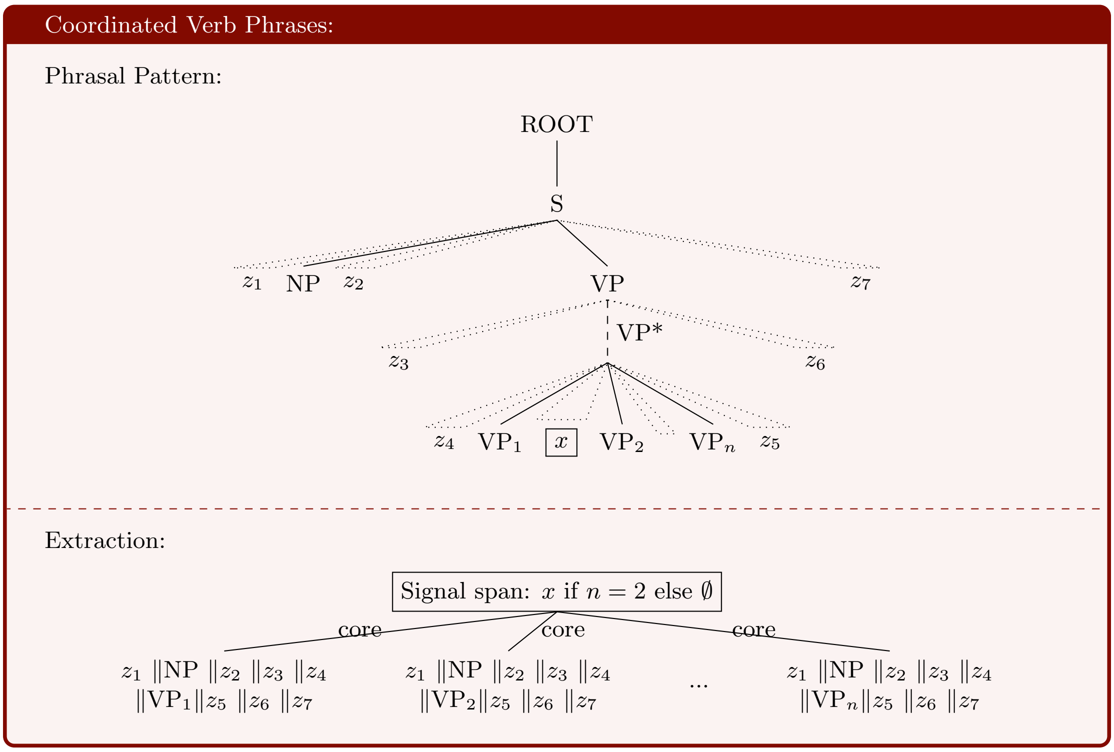

Coordinated Verb Phrases:
Phrasal Pattern:
ROOT
S
z_1
NP
z_2
VP
z_7
VP*
z_3
z_6
z_4
VP$_1$
$\boxed{x}$
VP$_2$
VP$_n$
z_5
Extraction:
Signal span: x if $n = 2$ else $\emptyset$
core
core
core
z_1 ||NP || z_2 || z_3 || z_4 ||VP$_1$|| z_5 || z_6 || z_7
z_1 ||NP || z_2 || z_3 || z_4 ||VP$_2$|| z_5 || z_6 || z_7
...
z_1 ||NP || z_2 || z_3 || z_4 ||VP$_n$|| z_5 || z_6 || z_7

Figure 3: Rule for coordinated verb phrases

3.5 Coordinated Noun Phrase Lists

According to the minimality principle of the Open IE task, we assembled rules for breaking up lists of entities. In this way, the recall of the extracted relational tuples is increased. The implemented rules match patterns of coordinated noun phrases within a parent noun phrase. In order to avoid inadvertently mistaking coordinated noun phrases for appositives (e.g. "*He returned to Kenya for a visit to his father's birthplace, a village near Kisumu in rural western Kenya*"), we generated a heuristic pattern that is determined by the regular expression $(NP)(,NP)*,?(and|or)(.+)$ which matches over the topmost noun phrases in subject and object position. By considering topmost noun phrases, we compensate for parsing errors that we have frequently identified in deep nested noun phrase structures. For each extracted entity, a simplified sentence is generated. The signal span is considered as one of the two coordinate conjunctions "*and*" or "*or*".

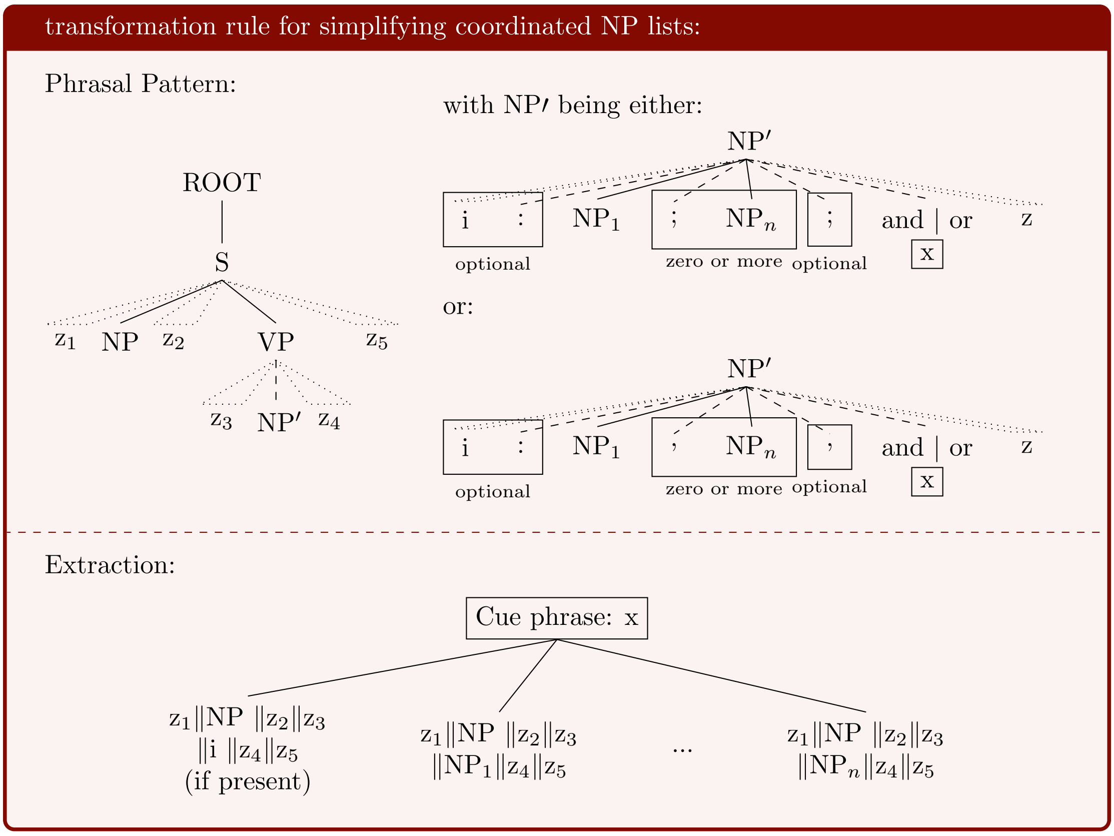


Figure 4: Rule for coordinated noun phrase lists in object position

3.6 Lead Noun Phrases

Occasionally, sentences may start with an inserted noun phrase, which in the majority of cases indicates a temporal expression. Hence, such a phrase generally represents background information that can be eliminated from the main sentence without resulting in a lack of key information. This is achieved by applying the simplification rule displayed in figure 5.

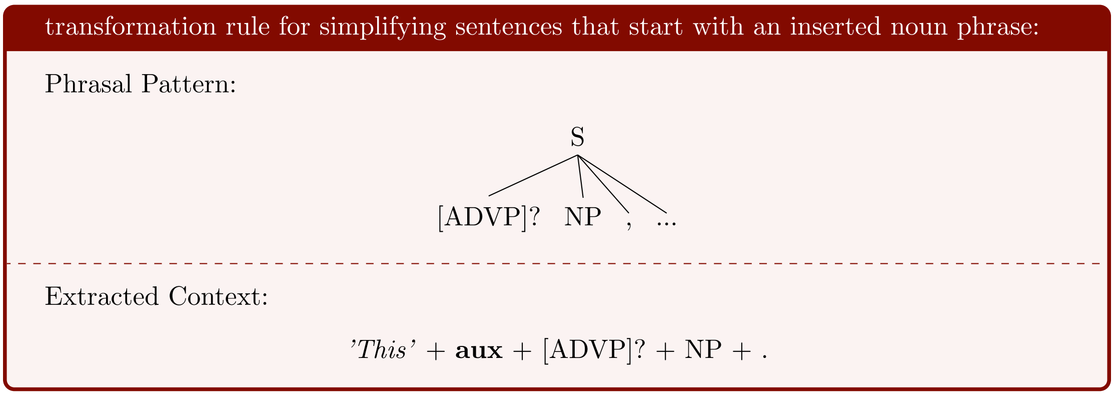
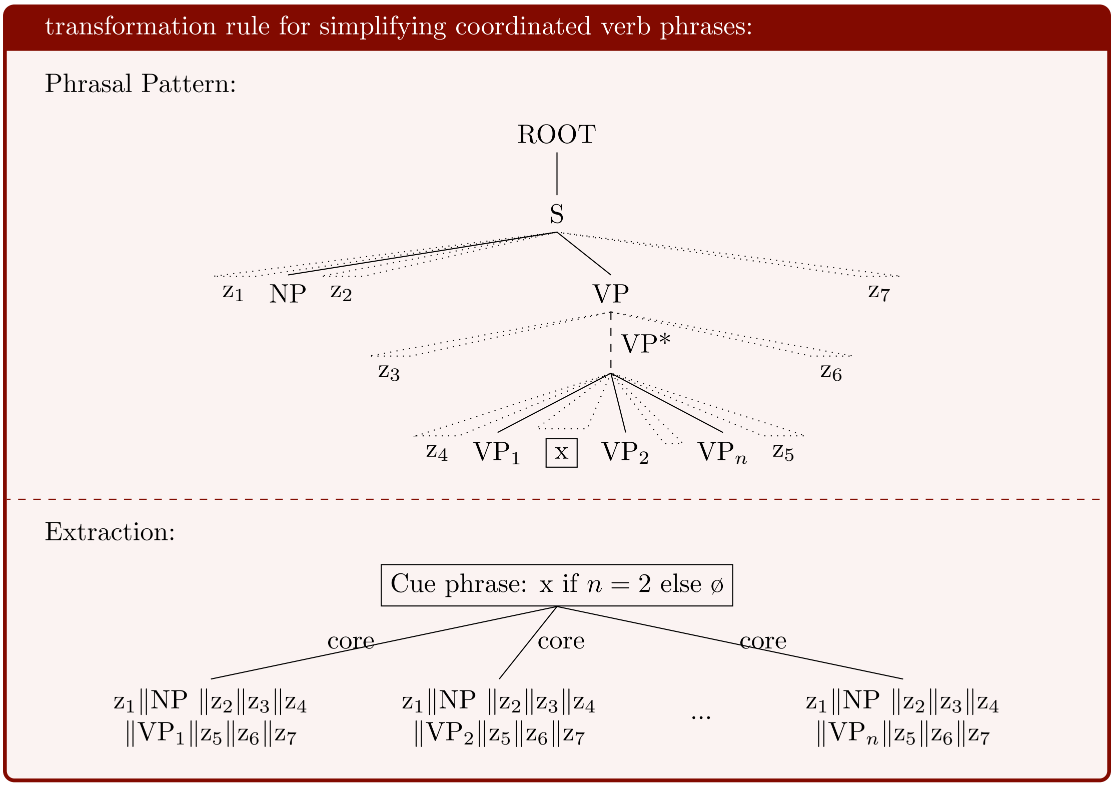

transformation rule for simplifying sentences that start with an inserted noun phrase:
Phrasal Pattern:
S
[ADVP]? NP , ...
Extracted Context:
'This' + aux + [ADVP]? + NP + .

Figure 5: Simplification rule for lead noun phrases



transformation rule for simplifying coordinated verb phrases:
Phrasal Pattern:
ROOT
S
z1 NP z2
VP
z7
VP*
z3
z6
z4 VP1 x VP2 VPn z5
Extraction:
Cue phrase: x if n = 2 else ø
core
core
core
z1||NP ||z2||z3||z4 ||VP1||z5||z6||z7
z1||NP ||z2||z3||z4 ||VP2||z5||z6||z7
...
z1||NP ||z2||z3||z4 ||VPn||z5||z6||z7

Figure 6: Rule for coordinated verb phrases

3.7 Appositive Phrases

An appositive is a noun phrase that further characterizes the phrase to which it refers. Just like relative clauses, appositions can be classified as restrictive and non-restrictive, respectively. Non-restrictive appositives are separate information units, marked by their segregation through punctuation [12]. Representing parenthetical information, they can be omitted without changing the meaning of the sentence. For resolving such an apposition, our simplification model uses a pattern which matches a noun phrase followed by a comma that is again followed by a noun phrase, with each of the two noun phrases potentially succeeded by a prepositional phrase: NP [PP]? , NP [PP]? [,—EOS].

In order to avoid inadvertently mistaking coordinated noun phrases for appositives, the following heuristic is applied: From the phrase that is deemed an appositive by matching the pattern above, we scan ahead, looking one after the other at his sibling nodes in the parse tree. If a conjunction *and* or *or* is encountered, the analysis of the appositive is rejected [13]. That way, we avoid wrong analyses like: *”Obama has talked about using alcohol,* [$_{appos}$ *marijuana*], *and cocaine.”*

An appositive is converted into a contextual sentence by taking the phrase including a proper noun - if any, otherwise simply the one before the comma -, as its first component and appending the appropriate form of the verb *”to be”* followed by the remaining noun phrase.

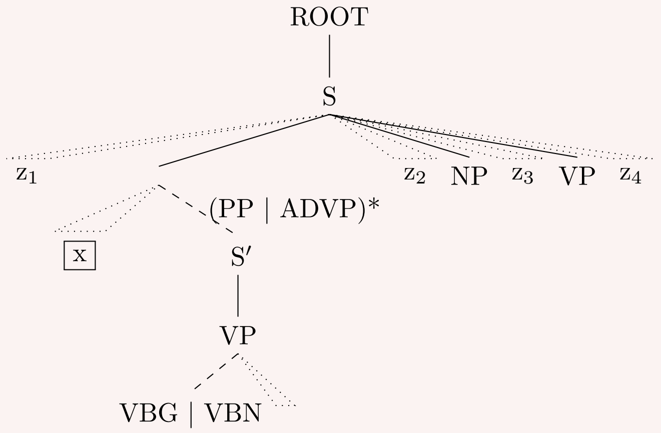
The second type of appositives, restrictive apposition, does not contain punctuation [12]. These constructs are simplified as well, provided they match a particular sequence of named entities and POS tags. To be exact, we first make use of the named entity tagged representation of the given input sentence, searching for words marked with either a ”PERSON” or ”ORGANIZATION” tag. On condition that such entities have been detected, we next check the POS tags of the respective preceding words. If we encounter a sequence of nouns or proper nouns, potentially with some prepending adjectives, determiners, numbers and/or possessive pronouns, we assume that this prefix string plays the role of a restrictive appositive phrase. In this case, the constituents ahead of the proper nouns are detached and transformed into an isolated context sentence by linking them via an auxiliary verb (one of *is, are, was, were*) to the identified person or organization entity, respectively.

3.8 Participial Phrases

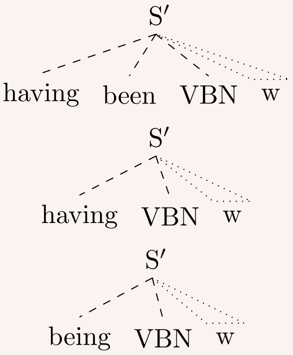
Often, if one or more actions are carried out by the same subject, participial phrases are used to express one of them. This can be done by replacing a main clause (e.g. "*Opening the drawer he took out a revolver*") or a subordinate clause (e.g. "*Fearing that the police would recognize him he never went out in daylight*") [7]. Participial phrases do not contain a subject of their own, but include active participles (e.g. "*warning*"), passive participles ("*warned*") or perfect participles active/passive ("*having warned*"/"*having been warned*") besides the objects of the participle and verbal modifiers. Furthermore, they can be introduced by adverbial connectors such as "*although*" or "*when*" [1]. In addition, we consider gerund constructions that occur after prepositions. For such constructions, the implemented rule patterns generate simplified sentences for each action, including the shared subject noun phrase to which they refer. The definition of a rule pattern for extracting initial participial phrases is shown in Figure ??. Note that the construction of the simplified sentences now requires a paraphrasing stage in order to generate grammatically sound sentences.

transformation rule for simplifying shared NP pre-participial:

Phrasal Pattern:



Extraction:

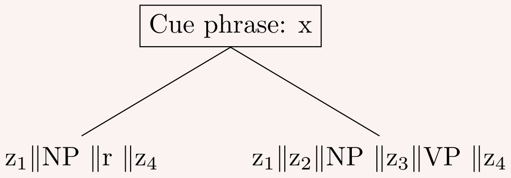


r ← {has/have} ||been ||VBN ||w

r ← {has/have} ||VBN ||w

r ← {is/are/was/were} ||VBN ||w

else: r ← {is/are/was/were} ||S′



3.9 Prepositional Phrases

A prepositional phrase is composed of a preposition and a complement which is characteristically a noun phrase (e. g. *on the table, in terms of money*), a nominal *wh*-clause (e. g. *from what he said*) or a nominal *-ing* clause (e. g. *by signing a peace treaty*). They may function as a postmodifier in a noun phrase, an adverbial phrase or a complement of a verb or an adjective [12]. Depending on their particular syntactic function, they contribute more or less fundamental information to the sentence which they are part of.

Among the different types of prepositional phrases mentioned above, those that play the role of adverbial phrases which are offset by commas are by far the easiest to handle, as they may generally be pruned without losing vital information. Assuming that they match one of the constituency parse tree patterns that have been identified for this type of phrase (most notably the ones displayed in figure 7), the constituents of the prepositional phrase under consideration are separated out of the given input sentence and transformed into simpler detached ones with the help of the corresponding rule.

In fact, things get much more complicated in case of prepositional phrases used as post-modifying adverbials or noun phrases without segregation through punctuation, since in this context, they may either provide information that identifies the phrase to which they refer (e. g. *"Obama's election as the first black president of the Harvard Law Review gained national media attention."*) or just describes, but not further defines, their antecedent (e. g. *"Paris has an extensive road network with more than 2,000 km of highways and motorways."*) Distinguishing between such defining and describing prepositional phrases is extremely demanding.

As a first approach to address this issue, we extract only a subgroup of suchlike prepositional phrases which feature specific properties. By examining several hundreds of sample sentences from various Wikipedia articles, we have discovered that in many cases prepositional phrases constituting the last component of a sentence, in particular when either relating to a location, person or organization (as indicated by their respective named entity tags), or representing a date (as signified by the POS tag 'CD'), may be removed without corrupting the meaning of the source sentence (cf. *"Obama formally announced his candidacy in January 2003."* or "*Obama delivered the keynote address at the 2004 Democratic National Convention.*") Hence, we take the current version of the core sentence - after having applied all other simplification rules defined within this framework - and look at its last prepositional phrase, if any. Provided that it either contains a word which is named entity tagged with 'LOCATION', 'PERSON' or 'ORGANIZATION', or ends with a number (i. e. POS tag 'CD') or a proper noun (i. e. POS tag 'NNP' or 'NNPS'), we separate it out into a stand-alone context sentence. This process is recursively repeated until a prepositional phrase that cannot be extracted due to missing aforementioned characteristics is encountered. In addition, it must be noted that phrases starting with the preposition "of" (cf. *"Obama served on the boards of directors of the Woods Fund of Chicago."* or *"His mother spent most of the next two decades in Indonesia."*), as well as those acting as a complement in a participial phrase (e. g. *"His remarks were made to a group of Marines preparing for deployment to Afghanistan."*) and sentences including an adjective or adverb in comparative or superlative form (like in *"Google announced the setting up of its largest campus outside the United States."*) are generally not eliminated, even though fulfilling the previously named requirements. The reason for this is that they usually serve as an integral component of the phrase to which they refer and therefore cannot be removed without resulting in sentences that are too curt or even gain a different meaning.

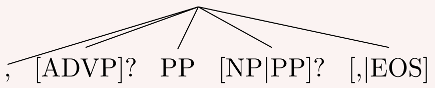
Furthermore, resolving prepositional phrases acting as complements of verbs or adjectives is of similar complexity, as they are often closely connected to their antecedent [12] and therefore cannot be left out without corrupting the meaning of the sentence (cf. *"Obama would become the first President to have been born in Hawaii."*, *"This took place in July 2011."* or *"Radio France is headquartered in Paris' 16th arrondissement."*) However, many times such verb or adjective phrase modifiers contribute no more than some form of additional background information which can be eliminated, resulting in a sentence that is still both meaningful and grammatical (e. g. *"Obama won with 70 percent of the vote.", "Obama's parents divorced in March 1964." or "Obama and Joe Biden were formally nominated by former President Bill Clinton."*) Beyond that, there are a lot of cases where it is well arguable whether or not to extract prepositional phrases complementing verbs or adjectives. Here, both source sentences are shortened to a grammatical, but terse core sentence. Hence, it might be preferable to preserve at least one of the prepositional phrases that have been included in the source.

Finally, prepositional phrases starting with the preposition *"to"* have been analyzed in more detail as they incorporate a very important type of extractable constituent, namely phrases describing intentions (cf. *"Obama commissioned a poll to assess his prospects in a 2004 U.S. Senate race."*) As before, based on the constituency parse trees of numerous sample sentences which contain this kind of phrase, we have been searching for patterns that can typically be separated out from the core, eventuating in compressed, though still informative, grammatical sentences.

Figure 7: Most important simplification rules for offset prepositional phrases

example transformation rule for simplifying offset prepositional phrases:

Phrasal Pattern:



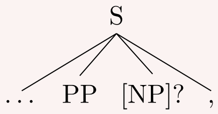
Extracted Context:

'This' + **aux** + [ADVP]? + PP + [NP—PP]? + .

(6.1) Parse tree pattern 1

example transformation rule for simplifying offset prepositional phrases:

Phrasal Pattern:



Extracted Context:

'This' + **aux** + PP + [NP]? + .

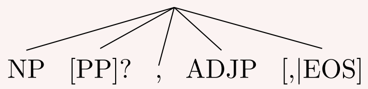
(6.2) Parse tree pattern 2

3.10 Adjectival Phrases

An adjectival phrase is one whose head is an adjective, optionally complemented by a number of dependent elements. It further characterizes the noun phrase it is modifying [2, 12].
Similar to participial phrases, only those adjectival phrases that are set off by commas are detached and transformed into contextual sentences with the help of the simplification rule depicted in figure 7. On the contrary, adjectival phrases that are not separated by punctuation customarily represent an integral part of the phrase to which they refer (cf. *"It is the subdivision responsible for providing emergency services."* or *"The council plays a largely passive role in the city government."*) As a result, such phrases are not extracted from the input sentence.

transformation rule for simplifying offset adjective phrases:

Phrasal Pattern:



Extracted Context:

NP + [PP]? + **aux** + ADJP + .

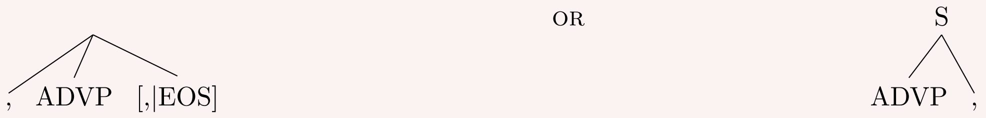
Figure 7: Simplification rule for adjective phrases

3.11 Adverbial Phrases

An adverbial phrase consists of an adverb as its head, together with an optional pre- or post-modifying complement [2, 12]. To guarantee that a phrase introduced by an adverb is not a fundamental constituent of the currently treated sentence, they are separated out only if they are offset by commas - as is the case with adjectival phrases. This is done by applying the respective transformation rule (see figure 8).

transformation rule for simplifying offset adverb phrases:

Phrasal Pattern:



Extracted Context:

'This' + **aux** + ADVP + .

Figure 8: Simplification rule for adverb phrases

References

[1] Roberta G Abraham. Field independence-dependence and the teaching of grammar. *Tesol Quarterly*, 19(4):689–702, 1985.

[2] Laurel J. Brinton. *The Structure of Modern English: A linguistic introduction*. John Benjamins B.V., Amsterdam, The Netherlands, 2000.

[3] Lynn Carlson and Daniel Marcu. Discourse tagging reference manual. *ISI Technical Report ISI-TR-545*, 54:56, 2001.

[4] Alistair Knott and Robert Dale. Using linguistic phenomena to motivate a set of coherence relations. *Discourse processes*, 18(1):35–62, 1994.

[5] Roger Levy and Galen Andrew. Tregex and tsurgeon: tools for querying and manipulating tree data structures. In *Proceedings of the fifth international conference on Language Resources and Evaluation*, pages 2231–2234, 2006.

[6] William C Mann and Sandra A Thompson. Rhetorical structure theory: Toward a functional theory of text organization. *Text-Interdisciplinary Journal for the Study of Discourse*, 8(3):243–281, 1988.

[7] A.V. Martinet and A.J. Thomson. *A Practical English Grammar. 4th edition.* Oxford University Press, 1996.

[8] Mausam, Michael Schmitz, Stephen Soderland, Robert Bart, and Oren Etzioni. Open language learning for information extraction. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 523–534, Jeju Island, Korea, July 2012. Association for Computational Linguistics.

[9] Eleni Miltsakaki, Rashmi Prasad, Aravind K Joshi, and Bonnie L Webber. The penn discourse treebank. In *LREC*, 2004.

[10] Christina Niklaus, Bernhard Bermeitinger, Siegfried Handschuh, and André Freitas. A sentence simplification system for improving relation extraction. In *Prooceedings of COLING 2016: System Demonstrations, The 26th International Conference on Computational Linguistics, Osaka, Japan, December 11-16, 2016*, pages 170–174, 2016.

[11] Rashmi Prasad, Eleni Miltsakaki, Nikhil Dinesh, Alan Lee, Aravind Joshi, Livio Robaldo, and Bonnie L Webber. The penn discourse treebank 2.0 annotation manual. 2007.

[12] Randolph Quirk, Sidney Greenbaum, Geoffrey Leech, and Jan Svartvik. *A Comprehensive Grammar of the English Language*. Longman, London, 1985.

[13] Advaith Siddharthan. Syntactic simplification and text cohesion. *Research on Language and Computation*, 4(1):77–109, 2006.

[14] Richard Socher, John Bauer, Christopher D. Manning, and Andrew Y. Ng. Parsing With Compositional Vector Grammars. In *ACL*. 2013.

[15] Maite Taboada and Debopam Das. Annotation upon annotation: Adding signalling information to a corpus of discourse relations. *D&D*, 4(2):249–281, 2013.