**ARTICLE**

# Building a Turkish UCCA dataset

Necva Bölücü[1,2] (ID) and Burcu Can[3]

[1]Department of Computer Engineering, Hacettepe University, Ankara, Turkey, [2]Data61, CSIRO, Sydney, NSW, Australia, and [3]Computing Science, University of Stirling, Stirling, UK
**Corresponding author:** Necva Bölücü; Email: necvaa@gmail.com

## Abstract

Semantic representation is the task of conveying the meaning of a natural language utterance by converting it to a logical form that can be processed and understood by machines. It is utilised by many applications in natural language processing (NLP), particularly in tasks relevant to natural language understanding (NLU). Due to the widespread use of semantic parsing in NLP, many semantic representation schemes with different forms have been proposed; Universal Conceptual Cognitive Annotation (UCCA) is one of them. UCCA is a cross-lingual semantic annotation framework that allows easy annotation without requiring substantial linguistic knowledge. UCCA-annotated datasets have been released so far for English, French, German, Russian, and Hebrew. In this paper, we present a UCCA-annotated Turkish dataset of 400 sentences that are obtained from the METU-Sabanci Turkish Treebank. We provide the UCCA annotation specifications defined for the Turkish language so that it can be extended further. We followed a semi-automatic annotation approach, where an external semantic parser is utilised for the initial annotation of the dataset, which is manually revised by two annotators. We used the same semantic parser model to evaluate the dataset with zero-shot and few-shot learning, demonstrating that even a small sample set from the target language in the training data has a notable impact on the performance of the parser (15.6% and 2.5% gain over zero-shot for labelled and unlabelled results, respectively).

**Keywords:** Universal Conceptual Cognitive Annotation; UCCA; Semantic representation; METU-Sabanci Turkish Treebank; dataset

## 1. Introduction

Semantics is concerned with everything that is related to meaning. In linguistics, a distinction is made between semantic representation and semantics, the former being more concerned with relations between the text components, that is, words, statements, etc. (Abend and Rappoport, 2017). Semantic representation is a way to transform the meaning of a natural language utterance so that it can be understood by machines in much the same way that humans understand natural language. With the increasing attention to semantic representation, a number of semantic representation frameworks have been proposed, such as Elementary Dependency Structures (EDS) (Oepen and Lønning, 2006), DELPH-IN MRS Bi-Lexical Dependencies (DM) (Ivanova *et al.* 2012), Abstract Meaning Representation (AMR) (Banarescu *et al.* 2013), Praque Semantic Dependencies (PSD) (Oepen *et al.* 2016), and Universal Conceptual Cognitive Annotation (UCCA) (Abend and Rappoport, 2013a). There are also additional semantic representations that

---

These authors contributed equally to this work.

are designed to uniformly represent meaning (Van Gysel *et al.* 2021) and serve various application purposes (Giordano, Lopez, and Le, 2023).

Each framework involves a set of formal and linguistic assumptions (Kuhlmann and Oepen, 2016). Kuhlmann and Oepen (2016) define three types: Flavor (0), Flavor (1), and Flavor (2) for a formal graph to distinguish the graph-based semantic frameworks by specifying the form of anchoring in graphs. If we sort the Flavors from the strongest to the weakest form of anchoring, Flavor (0) indicates the strongest form of anchoring, where each node is directly linked to a specific token. Flavor (1) is a more general form of anchored semantic graph, where there is a more relaxed anchoring that allows linkage between nodes and arbitrary parts of the sentence, which can be a sub-token or a multi-token sequence. This provides flexibility in the representation of meaning. EDS (Oepen and Lønning, 2006) and UCCA (Abend and Rappoport, 2013a) are categorised in Flavor (1). UCCA defines semantic graphs with a multi-layer structure, where the foundational layer of representation focuses on the integration of all surface tokens into argument structure phenomena (e.g., verbal, nominal, and adjectival). Terminal nodes of a graph are anchored to discontinuous sequences of surface substrings, whereas interior nodes of a graph are not. Flavor (2) indicates unanchored graphs where there is no correspondence between nodes and tokens. AMR (Banarescu *et al.*, 2013) is categorised in Flavor (2), a sentence-level semantic representation framework that uses unanchored graphs where nodes represent concepts (predicate frames, special keywords, etc.) and edges represent semantic relations between them to avoid explicit mapping of graph elements to surface utterances.

Such semantic representations enable the structured use of meaning in natural language processing (NLP) and natural language understanding (NLU) applications, including text summarisation (Liu *et al.* 2015; Liao, Lebanoff, and Liu, 2018; Zhang *et al.* 2020), paraphrase detection (Issa *et al.* 2018), machine translation (Song *et al.* 2019; Sulem, Abend, and Rappoport, 2020; Nguyen *et al.* 2021; Slobodkin, Choshen, and Abend, 2021), question answering (Xu *et al.* 2021; Kapanipathi *et al.* 2021; Naseem *et al.* 2021), and text simplification (Sulem, Abend, and Rappoport, 2018). Although language models such as BERT (Devlin *et al.* 2019) and RoBERTa (Liu *et al.* 2019) can capture some semantic information implicitly from training data, there is still a big gap between implicitly learned semantic structures and gold semantic representations annotated by humans (Hewitt and Manning, 2019; Bölücü and Can, 2022a). There have been several attempts to integrate semantic information into language models (Naseem *et al.,* 2021; Slobodkin *et al.,* 2021; Ameer *et al.* 2023), and they show that the performance of the models improves substantially with the use of annotated datasets, which provides evidence for the usefulness of such annotated datasets particularly in semantic tasks.

UCCA is a recently proposed semantic annotation framework for representing the "semantic meaning" of a sentence within a multi-layered framework, where each layer corresponds to a semantic module. The ultimate goal of UCCA is to provide a semantic representation, applicable across languages, that enables parsing across languages using cross-lingual machine learning approaches, as well as parsing across different domains. It also supports rapid annotation by non-experts who do not have a proficient linguistic background. Due to these advantages, the UCCA representation has gained remarkable attention and has been part of recent shared tasks such as SemEval 2019 (Hershcovich *et al.* 2019b), MRP 2019 (Oepen *et al.* 2019), and MRP 2020 (Oepen *et al.* 2020). Furthermore, there is an ongoing workshop for designing meaning representations (DMRs) covering several frameworks, including UCCA (Xue *et al.* 2020).

Since UCCA was first proposed by Abend and Rappoport (2013a), English UCCA-annotated datasets, which are *English Wikipedia* (Abend and Rappoport, 2013b) and *English 20K Leagues Under The Sea*, were released (Sulem, Abend, and Rappoport, 2015) along with an annotation guideline.[a] These datasets were followed by other UCCA-annotated datasets in several languages, including French, German, Russian, and Hebrew. For English, the datasets are obtained from

---

[a]https://universalconceptualcognitiveannotation.github.io/

*Wikipedia* (Abend and Rappoport, 2013b), *Web Treebank* (Hershcovich, Abend, and Rappoport, 2019a), *The Little Prince* (Oepen *et al.,* 2020), and an English–French parallel corpus of *Twenty Thousand Leagues Under the Sea* (Sulem *et al.,* 2015) including the first five chapters. The expansion of the UCCA dataset to other languages started with the parallel corpus of *Twenty Thousand Leagues Under the Sea*. The German dataset (Hershcovich *et al.* 2019b) consists of the entire book, while the French dataset contains the first five chapters of the parallel corpus annotated using cross-lingual methods (Sulem *et al.,* 2015). In addition, the book *The Little Prince* is used for German (Oepen *et al.,* 2020), Russian (Hershcovich *et al.* 2019b), and Hebrew, the last two languages being new to UCCA datasets.

In this study, we investigate the UCCA graph-based semantic representation for Turkish, which is a low-resource language for semantic annotation but a decent-resource language for other types of resources such as treebanks (Oflazer *et al.* 2003; Türk *et al.,* 2022) and lexicons (Vural, Cambazoglu, and Karagoz, 2014; Zeyrek and Basıbüyük, 2019).[b] There is only one semantically annotated dataset in Turkish in the literature, which was presented by Acar *et al.* (2024). This study presents an annotated Turkish AMR dataset with 100 sentences obtained from the Turkish translation of the novel *The Little Prince* and 600 sentences obtained from the IMST Dependency Treebank (Sulubacak, Eryiğit, and Pamay, 2016; Sulubacak and Eryiğit, 2018; Şahin and Adalı, 2018). The authors present a rule-based AMR parser to facilitate the annotation process and discuss Turkish constructions that require special rules for AMR representations. The preliminary study of Turkish AMR, as well as a warm-up phase for annotation, is conducted by Azin and Eryiğit (2019). Due to the lack of semantic datasets that have been proven to be effective for NLP and NLU applications, we introduce another semantically annotated dataset for Turkish using the UCCA framework. The preliminary annotation has already been performed on a small corpus with fifty sentences (Bölücü and Can, 2022b) obtained from the METU-Sabanci Turkish Treebank[c] (Atalay *et al.,* 2003; Oflazer *et al.,* 2003) conducting an initial exploration into Turkish grammar rules and assessing the efficacy of employing zero-shot learning alongside an external semantic parser for UCCA annotation. In this study, we further extended the UCCA dataset by increasing the dataset size to 400 sentence, once again obtained from the METU-Sabanci Turkish Treebank and provided a more comprehensive guideline covering all possible syntactic rules required to annotate a Turkish dataset. The rules aim to be comprehensive enough for future researchers aiming to annotate a Turkish dataset and may also contribute to understanding UCCA annotation procedures in other languages.

We perform the annotation procedure in a semi-automatic process using an external semantic parser (see Section 5.3, illustrated in Figure 1). The annotation procedure consists of two steps: (1) an external semantic parser is trained on a dataset that is a combination of English, German, and French UCCA-annotated datasets released in SemEval 2019 (Hershcovich *et al.* 2019b) to produce semantic representations that are partially correct and (2) the semantic representations obtained by the semantic parser are manually corrected as needed by following the rules defined in the original UCCA annotation guideline,[d] and new rules are defined peculiar to the Turkish language if required (see Section 4).

The contributions of the study are as follows:

- An extended UCCA semantic representation dataset for the Turkish language, comprising 400 UCCA-annotated sentences[e] (see Section 5).

---

[b]More detailed information about the Turkish resources are given in Çöltekin *et al.,* (2023).
[c]METU-Sabanci Turkish Treebank is a publicly available morphologically and syntactically annotated treebank and publicly available at https://web.itu.edu.tr/gulsenc/METUSABANCI_treebank_v-1.rar.
[d]https://github.com/UniversalConceptualCognitiveAnnotation/docs/blob/v2.1/guidelines.pdf
[e]The dataset is available at https://github.com/necvabolucu/semantic-dataset
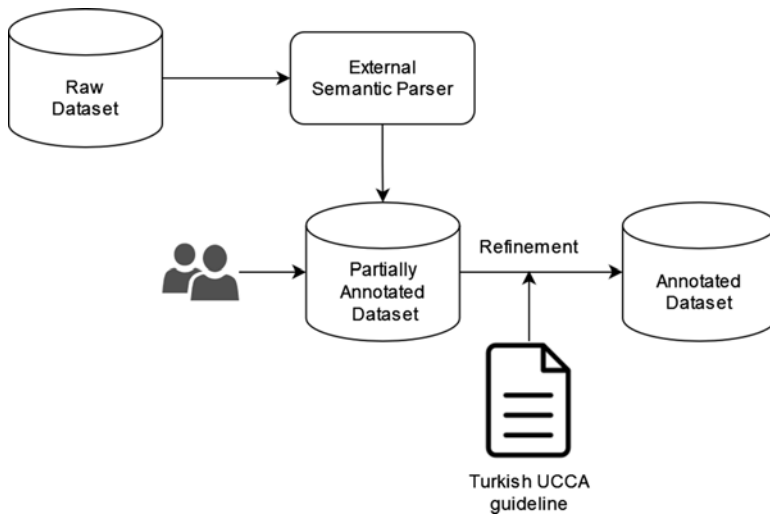
**Figure 1.** Turkish UCCA dataset annotation process comprises two steps: (1) obtaining partially annotated dataset using an external semantic parser and (2) refining the partially annotated dataset by human annotators.

- Turkish UCCA annotation guideline, covering syntactic and morphological rules not covered in the English UCCA guideline (Section 4) and more comprehensive than the initial annotation (Bölücü and Can, 2022b). This resource aims to help annotators annotate the Turkish UCCA dataset and understand the relationships between grammatical rules and the UCCA features. The rules cover both language-specific aspects, such as closed-class words, and more general rules that will help annotate other languages that share similar syntactic features with Turkish. We also provide Turkish examples of UCCA annotation, covering all the categories listed in the English guideline (Appendix A).

- A comprehensive analysis of zero-shot and few-shot learning experimental results using the newly annotated dataset to understand the discrepancies between the annotated sentences in the dataset, providing insights into the use of zero-shot and few-shot techniques in dataset annotation (see Section 6).

This paper is organised as follows: Section 2 discusses the datasets released in Turkish and related work on UCCA semantic parsing, Section 3 describes the UCCA representation framework, Section 4 presents an overview of the Turkish grammar with the UCCA annotation rules defined to annotate Turkish sentences, Section 5 describes the annotation details, including the dataset used, the external semantic parser and the annotation process, as well as the statistical analysis of the annotated Turkish UCCA dataset, Section 6 presents the results obtained by the semantic parser model in zero-shot and few-shot settings using the new Turkish dataset, and finally Section 7 concludes the paper with some potential future goals.

## 2.  Related work

Here, we review the related work, both on annotated datasets for Turkish and on UCCA-based semantic parsing in other languages.

### 2.1 Related work on available Turkish datasets

Turkish is an agglutinative language, where words are formed by productive affixation and multiple suffixes can be attached to a root to form a new word form:

**Example 2.1.**

*gidebilirsek (in English, "if we can go")*

| *gid*[*Verb*] | *Abil*[*Verb* + *Able*] | *Ir*[*Verb* + *PresentPart*] | *sA*[*Verb* + *Cond*] | *k*[*Verb* + *P*1*Pl*] |
|---|---|---|---|---|
| go | can | — | if | we |

It is possible to generate an infinite number of words in Turkish, as shown by Sak *et al.* (2011). This poses a challenge for many NLP tasks, such as language modelling, spell checking, and machine translation, because of the out-of-vocabulary problem. Turkish grammar presents also other challenges, such as free word order in a sentence and the use of clitics (Oflazer, 2014; Azin and Eryiğit, 2019), which may have various meanings depending on the context.

The datasets for many NLP tasks have been released mostly in English. The datasets released in Turkish are very limited. Current Turkish NLP studies generally focus on building datasets for syntactic parsing, such as the METU-Sabanci Treebank (Atalay *et al.*, 2003; Oflazer *et al.*, 2003) and the IMST Dependency Treebank (Sulubacak *et al.*, 2016). Although semantic annotations are crucial for NLP tasks, there are few studies on Turkish semantic annotation (Şahin and Adalı, 2018; Azin and Eryiğit, 2019; Oral *et al.*, 2024). One of the semantic datasets in Turkish is the Turkish Proposition Bank (PropBank) (Şahin and Adalı, 2018), the first semantically annotated corpus built particularly for semantic role labelling (SRL). The other annotated dataset in Turkish is the AMR corpus presented by Azin and Eryiğit (2019) and Oral *et al.*, (2024). Azin and Eryiğit (2019) presented the preliminary investigation on Turkish AMR with 100 annotated sentences obtained from the Turkish translation of the novel *The Little Prince* based on AMR specifications to demonstrate the differences between Turkish and English annotations. Oral *et al.* (2024) extended the annotation procedure by converting the annotation process into a semi-automatic annotation using a rule-based parser in which PropBank (Şahin and Adalı, 2018) sentences are converted into AMR graphs. Human annotators used the output of the rule-based parser to build the dataset rather than annotating it from the very beginning. The presented AMR corpus contains 700 sentences (100 sentences from the Turkish translation of the novel *The Little Prince* and 600 sentences from the IMST Dependency Treebank (Sulubacak *et al.*, 2016; Sulubacak and Eryiğit, 2018; Şahin and Adalı, 2018)).

The first Turkish UCCA dataset in Turkish was introduced by Bölücü and Can (2022b), who also reported a limited set of initial rules for the UCCA annotation framework with 50 annotated sentences. This study extends the Turkish UCCA dataset by providing a preliminary investigation of Turkish grammar and the effectiveness of using zero-shot learning with an external semantic parser for the UCCA annotation by increasing the dataset size and providing a more comprehensive guideline that covers all possible syntactic rules which will be sufficient to annotate a Turkish dataset using the UCCA framework. The annotated dataset will serve as a valuable semantic annotation resource, akin to the Turkish AMR dataset. The general rules, which are not language-specific, have the potential to be used in the annotation of other languages in the UCCA framework. Moreover, the annotated dataset can be used in other NLP tasks (Issa *et al.*, 2018; Naseem *et al.*, 2021; Kapanipathi *et al.*, 2021).

### 2.2 Related work on UCCA semantic parsing

UCCA parsing has been applied to various languages including English, German, and French. The proposed UCCA parsing approaches can be divided into four classes: (1) transition-based, (2) graph-based, (3) compositional-based, and (4) encoder–decoder-based. Transition-based

approaches use new transition actions that handle features of the UCCA framework such as discontinuity and reentrancy. The first UCCA parser is a transition-based model called TUPA parser (Hershcovich, Abend, and Rappoport, 2017), which uses new transition rules (e.g., NODE, LEFT-EDGE, RIGHT-EDGE, LEFT-REMOTE, RIGHT-REMOTE, and SWAP) with new features (e.g., separating punctuation and gap type). Hershcovich *et al.* (2018) extend the TUPA parser with multi-task learning using a set of graph-based frameworks, namely AMR (Banarescu *et al.,* 2013), UD (Universal Dependencies) (Nivre *et al.* 2016, 2020), and PSD (Oepen *et al., 2016). Pütz and Glocker (2019) implement the TUPA parser with a set of features based on the top three items on the stack and the buffer, as well as deep contextualised word embeddings of the rightmost and leftmost parents and children of the respective items by using additional training data. With the popularity of graph-based semantic frameworks, shared tasks have been conducted in several workshops such as SemEval (Hershcovich *et al.* 2019b) and MRP (Oepen *et al.* 2019, 2020), which also introduced new transition-based methods, many of which are extensions of the TUPA parser (Lai *et al.* 2019; Arviv, Cui, and Hershcovich, 2020). Bai and Zhao (2019) extend the TUPA parser with hand-crafted features. Dou *et al.* (2020) use Stack-LSTM instead of BiLSTM used in TUPA parser, and Lyu *et al.* (2019) use TUPA$_{BiLSTM}$ and TUPA$_{MLP}$ as a cascaded parser with a multi-stage training procedure, first training TUPA$_{BiLSTM}$ and retraining the model with TUPA$_{MLP}$.

Graph-based approaches (Cao *et al.* 2019; Koreeda *et al.* 2019; Droganova *et al.* 2019) aim to generate a graph with the highest score among all possible graphs in the graph space, which can be seen as a graph-searching problem. The most common approach tackles the task as a constituency parsing problem (Jiang *et al.* 2019; Li *et al.* 2019; Zhang *et al.* 2019; Cao *et al.,* 2019; Bölücü *et al.,* 2023), where directed acyclic graphs (DAGs) are converted into constituency trees. For example, Bölücü *et al.* (2023) adopts self-attention mechanism proposed by Kitaev and Klein (2018b) to learn UCCA semantic representations, with zero- and few-shot experiments on different languages.

Composition-based approaches follow the compositionality principle and perform semantic parsing as a result of a derivation process that incorporates both lexical and syntactic-semantic rules to develop a semantic graph parser (Che *et al.* 2019; Donatelli *et al.* 2019; Oepen and Flickinger, 2019). Donatelli *et al.* (2019) apply the AM (Abstract Meaning) dependency parser of Lindemann *et al.* (2019) after converting UCCA annotations into AM dependency graphs.

Encoder–decoder approaches use an encoder–decoder architecture to convert an input sentence into a semantic graph as performed in neural machine translation (NMT) (Na *et al.* (2019); Yu and Sagae (2019); Kitaev and Klein (2018a); Ozaki *et al.* (2020)).

For the annotation, we utilised the semantic parser that adopts a graph-based approach proposed by Bölücü *et al.,* (2023) in which the effectiveness of zero- and few-shot experiments on various languages has been already demonstrated. This parser has been also applied to the annotation of the initial Turkish UCCA dataset (Bölücü and Can, 2022b).

## 3. Universal Conceptual Cognitive Annotation

UCCA is a semantic annotation scheme that is strongly influenced by the Basic Linguistic Theory (BLT) (Dixon, 2005, 2010a, 2010b, 2012) and cognitive linguistic theories (Langacker 2007). It was introduced by Abend and Rappoport (2013a). It is a cross-linguistically applicable annotation scheme that is used to encode semantic annotations.

UCCA is a multi-layered framework in which each layer corresponds to a "module" of semantic distinctions. The foundational layer of UCCA focuses on all grammatically relevant information. This layer covers predicate–argument relations for predicates of all grammatical categories (verbal, nominal, adjectival, and others such as tense, and number).

UCCA is represented by a DAG with leaves corresponding to tokens and multi-tokens in a given text. The nodes of the graph are known as units, which are either terminals or non-terminals.
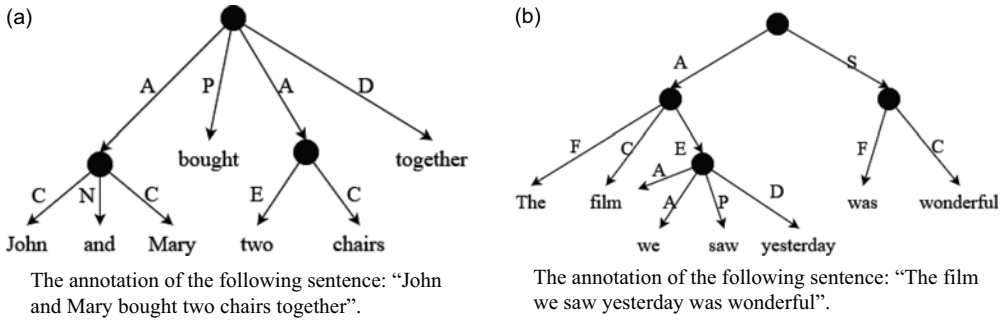
(a) The annotation of the following sentence: "John and Mary bought two chairs together".

(b) The annotation of the following sentence: "The film we saw yesterday was wonderful".

**Figure 2.** Examples of UCCA annotation graphs. Category abbreviations: **A**: Participant, **P**: Process, **D**: Adverbial, **C**: Center, **N**: Connector, **E**: Elaborator, **F**: Function

Multiple tokens correspond to a single entity based on a particular semantic or cognitive consideration. Edges of a graph refer to the categories of its children. There are four main categories in UCCA representation: (i) Scene Elements: Process (P), State (S), Participant (A), Adverbial (D), (ii) Non-Scene Unit Elements: Center (C), Elaborator (E), Connector (N), Relation (R), (iii) Inter-Scene Relations: Parallel Scene (H), Linker (L), Ground (G), and (iv) Other: Function (F).

Two sentences annotated based on the UCCA framework are given in Figure 2. In the sentence given in Figure 2a, there is a Scene with a relation called **Process**, which corresponds to *"bought"*. *"John and Mary"* and *"two chairs"* are the **Participant**s of the Scene, and *"together"* is the **Adverbial** in that Scene. The **Participant** *"John and Mary"* consists of entities of the same type, that is called **Center** for both *"John"* and *"Mary"*, connected by *"and"* which is a **Connector**. The **Participant** *"two chairs"* is composed of a **Center** that is *"chairs"* and an **Elaborator** that is *"two"*, which describes the **Center**. In the second sentence given in Figure 2b, there is a Scene containing a relation called **State** and a **Participant**. The **State** consists of a **Function** and a **Center** and the **Participant** consists of an **Elaborator**, a **Function**, and a **Center**. The **Elaborator** *"film we saw yesterday"* is an **E-Scene**, because *"saw"* evokes another **Scene**. While *"film"* is the **Center** of the **Participant**, it also serves as the **Participant** of the **E-Scene**, resulting in *"film"* having two parents.

### 3.1 UCCA categories

The foundational layer views the text as a collection of Scenes. A Scene describes a movement, an action, or a temporally persistent state. It usually contains information regarding when the Scene happens, the location, and the ground that explains how it happens.

#### 3.1.1 Scene elements

A Scene contains only one main relation, which determines the type of Scene (either dynamic or static). The Relation becomes a Process (P) if there is an action or a movement. However, if a Scene evolves in time, it becomes a State (S), where the Scene describes a temporally persistent state. A Scene contains one or more Participants (A).[f] They can be concrete or abstract. Embedded Scenes are also considered as the Participants of the main Scene. The secondary relations of the Scenes are marked as Adverbials (D). They describe the main relation (P/S) in the Scene. It can refer to the time, frequency, and duration of the process or state, as well as the modality in verbs (e.g., can, will, etc.), the manner relations, and the relations that specify a sub-event (begin, end, finish, etc.).

---

[f]It is possible to have annotations where a Scene may not have a Participant (A) (Abend *et al.* 2020).

### 3.1.2 Non-Scene unit elements

There are also non-Scene relations in the UCCA framework that do not evoke a Scene. Each non-unit contains one or more `Center` (C), which is required for the conceptualisation of the non-Scene unit. It is the main element of the non-Scene unit, and other relations may elaborate or be associated with this element. Class descriptors that determine the semantic type of the parent unit are considered as an `Elaborator` (E) of the main element. Quantifiers describing the quantity of the magnitude of an entity or expression are also identified as an `Elaborator` (E). `Connectors` (N) are the relations between two or more entities with similar features or types.

The other type of relation between two or more units is called `Relator` (R) that does not evoke a Scene. In two different scenarios, the relation is marked as a Relator:

1. A single entity is related to another relation in the same context as a `Relator`. In this case, the `Relator` should be positioned as a sibling of the `Center` (or the `Scene`). It is placed inside the unit they pertain to.
2. Two units attached with different aspects of the same entity are related through a `Relator`.

### 3.1.3 Inter-Scene relations

Linkage is the term that is used for the Inter-Scene relations in UCCA. There are four types of Linkages in the UCCA, adopted from the BLT (Dixon, 2010a):

1. Elaborator Scenes: `E-Scene` adds information to a unit that has been previously established. It answers `which X` or `what kind of X` questions.
2. Center Scenes: `C-Scene` is a `Center` unit of a larger `Scene` that is also a `Scene` that is internally annotated.
3. Participant Scenes: `A-Scene` is a participant in a `Scene` and has a removable role, as it does not add information to a particular participant in the main `Scene`. It is usually the answer to a `what` question in a `Scene`.
4. Parallel Scenes: If a `Scene` is not a `Participant`, `Center`, or `Elaborator` in a `Scene` and is connected to other `Scenes` by a `Linker`, which is a relational word between `Scenes`, the `Scenes` are called `Parallel Scenes` (H).

A unit is marked as a `Ground` (G) if its main purpose is to relate some unit to its speech event; either the speaker, the hearer, or the general context in which the text was uttered/written/conceived. `Ground` is similar to `Linker`, except that `Ground` does not relate the `Scene` to some other `Scene` in the text, but with the speech act of the text (the speaker, the hearer, or their opinions). The speech event is also called `Ground`.

### 3.1.4 Other

There are also `Function` (F) units in which the terminals do not refer to a participant or a relation. They function only as part of the construction in which they are situated in.

## 3.2 Remote and implicit units

While some relations are clearly described in the text, there are instances where a sub-unit in a given unit is not explicitly mentioned:

1. If the missing entity is referenced from another position in the text, we add a reference to the missing unit, which is labelled as a `REMOTE` unit (the minimal unit is used as REMOTE).

2. When the missing entity does not appear in any place in the text, we add an IMPLICIT entity that stands for the missing sub-unit.

The reader may refer to the actual English guideline for an elaborate presentation and example annotations for various scenarios (Abend *et al.* 2020).[g] We provide further examples in Turkish as part of the Turkish guideline in Appendix A.

## 4. Turkish UCCA annotation guideline

In this section, we briefly describe Turkish grammar along with the morphological and syntactic features of the language that were consulted during the annotation process and give the relevant UCCA annotation rules defined for Turkish in addition to the existing rules in the original UCCA guideline for English (Abend *et al.*, 2020), especially for the cases where the rules do not cover the Turkish grammar. We do not describe the existing UCCA annotation rules again here, but only the new UCCA annotation rules that we defined based on Turkish grammar.

### 4.1 Morphology

Turkish is an agglutinative language where suffixes are attached to word roots or stems to form words, known as suffixation. Therefore, the vast majority of Turkish words contain more than one syllable (Lewis 1967).

Morphemes are defined as the smallest meaning-bearing units in a language. There are two types of morphemes, depending on whether they are attached to a word or stand alone in a sentence: **free morphemes**, also called unbounded morphemes can stand alone as if they are words, **bound morphemes** can only be seen attached to a word. Morphemes can be further analysed in two categories, depending on whether they modify the grammatical role (i.e., part-of-speech (PoS)) and the fundamental meaning of a word.

*Derivational morphemes*. Derivational morphemes are bound morphemes that have the ability to modify the meaning and the PoS of a word (e.g., "boya" (the paint) - "boya-**cı**" (the painter)).

**Example 4.1.** *Bir süre **sessiz** yürüdük. (in English, "We walked **quietly** for a while.", i.e. "ses" (Noun, sound) and "sessiz" (Adverb, quietly))*

*Inflectional morphemes*. Inflectional morphemes are bound morphemes attached to nouns, pronouns, nominal phrases, verbs, and verb frames that modify functional relations such as case, tense, voice, mood, person, and number.

**Example 4.2.** *Ahmet ona **ilgiyle** baktı. (in English, "Ahmet looked at him **with interest**.", i.e. "ilgi" (interest) and "ilgiyle" (with interest), where "ilgiyle" is labelled as an adverbial relation in the annotation.)*

They are divided into two categories:

1. **Nominal inflectional morphemes** indicate number, possession, and case (e.g. "çocuk" (kid) -lar (number) -ın (possession)-a (case) - "çocuklarına" (to your children)).
2. **Verbal inflectional morphemes** attach to verbs in two ways: finite verb forms and non-finite verb forms. A finite verb acts as a verb in a sentence and a non-finite verb acts as a nominal in a sentence. The ones that are attached to finite verbs such as voice suffixes,

---

[g]https://github.com/UniversalConceptualCognitiveAnnotation/docs/blob/v2.1/guidelines.pdf

negative markers, tense, and modality markers are not separately labelled but only as a whole verb, which usually refers to an action in the annotation.

**Example 4.3.** *Her şey **bitmişti**, **anlamıştım**. (in English, "It **had ended**, **I had realised**.", i.e. "bit-" (to end), "bitmiş" (ended), "bitmişti" (had ended), "anla-" (to realise), "anlamış" (realised), "anlamıştım" (had realised), where the final verbs refer to an action in the sentence.)*

Copula markers are one of the peculiarities in Turkish grammar (Lewis 1967). A copula is a connecting word or phrase in a particular form of a verb that links a subject and a complement.

Subordinating suffixes appear only in non-finite verb forms and are the main means of forming subordinate clauses in Turkish. They are combined with verb stems to form nominals (e.g., "Okula **git-me-yeceğ-i** belli." (*It is clear **[that s/he won't go to school]**.*)).

*Reduplication.* Reduplication is the repetition of a word or part of a word. There are three types of reduplication in Turkish: emphatic reduplication, m-reduplication, and doubling.

1. **Emphatic Reduplication:** It is used to emphasise the quality of an adjective ("uzun" (long) → "u**p**uzun" (very long)).
2. **m-Reduplication:** In this form, a word is duplicated by adding "m" letter to the beginning of the second word either by replacing the first consonant with "m" or adding "m" if the word begins with a vowel ("etek" (skirt) → "Etek **m**etek" (skirt and the like)).
3. **Doubling:** In this form, a word is doubled with the same form of the word ("koş-" (to run) → "koşa koşa" ("in a hurry" or "willingly", depending on the context)).

*4.1.1 UCCA annotation guideline based on morphology*
Here, we describe the rules that do not exist in the original UCCA guideline in English and that we have defined on the basis of Turkish morphology:

1. **Derivationally modified words:** During annotation, we only considered the final PoS to define the UCCA category of a derived word.[h]

   **Example 4.4.**

   | *Soruyu* | $\langle$*hızlıca*$\rangle_D$ | *cevapladı* |
   |---|---|---|
   | *The question* | *quickly* | *S/he answered* |

   *She answered the question $\langle$quickly$\rangle$*

2. **Nominal morphemes:** The morphemes are not labelled separately, but the words they are attached to are labelled mostly based on their semantic roles in a sentence, which are mainly shaped by these inflectional morphemes (particularly with the last inflectional morpheme within the word).
3. **Inflectional suffices added to proper noun:** While inflectional suffixes are added to the end of the word when it is a common noun (e.g., "kedi" (cat)), they are separated from the proper noun (e.g., "Kerem", "İstanbul;") by an apostrophe. Since they relate the word to

---

[h]The word or groups of words defining the semantic label are indicated in the examples by $\langle\rangle$.

the Scene, we separated inflectional suffixes from the proper nouns and labelled them as `Relator`. However, we left the common nouns as they are.

**Example 4.5.**

> *Erkekler Park*ı    ⟨*'na*⟩$_R$    *gidiyorsun*
> *the Erkekler Park*    *to*    *you are going*
> *You are going* ⟨*to*⟩ *the Erkekler Park*

4. **Genitive case:** To express possession in Turkish, the genitive case suffix is attached to the possessor, and the possessive suffix is attached to the possessed noun. Pronominal possessors of possessive nouns can also be omitted because of the possessive suffix that already contains the possessive meaning. Omitted pronominal possessors of possessive nouns are labelled with `E-IMPLICIT`.

**Example 4.6.**

> ⟨(*Onun*)⟩$_{E-IMPLICIT}$    *gözleri*    *buğulanmıştı*    *bir*    *an*
> *Her/his*    *eyes*    *were fogged*    *a*    *moment*
> ⟨*Her/his*⟩ *eyes were fogged for a moment*

5. **Subordinating suffixes:** They are labelled based on their final semantic relation within the sentence (e.g., participant), but since they refer to an action, they are labelled as a process within the phrase (see Rule 6.b in Section 4.2.1).

6. **Reduplication:**

   - **Emphatic Reduplication:** We label the word with its actual category as defined in the UCCA guideline since it does not require any additional rule.[i]

     **Example 4.7.**

     > ⟨*Simsiyah*⟩$_D$    *olmaya*    *başladı*
     > *Pitch black*    *to turn*    *it started*
     > *It started to turn* ⟨*pitch black*⟩

   - **m-Reduplication:** We combine the reduplicated form with the word during annotation, as the reduplicated form is usually not a valid word♦.

     **Example 4.8.**

     > ⟨*Etek*    *metek*⟩$_A$    *almadı*
     > *Skirt*    *the like*    *s/he did not buy*
     > *S/he didn't buy* ⟨*skirt and the like*⟩

---

[i]During annotation, we did not come across any example of this type of marker. Therefore, the example does not exist in the METU dataset.

- **Doubling:** We combine the doubled form with the word to be concise with the annotation. We label the word with its actual category as also defined in the UCCA guideline.

**Example 4.9.**

$$\langle Koşa \quad koşa \rangle_D \quad geldim \quad buraya$$

*Running   running   Icame      here*

*I came here ⟨in a hurry⟩*

### 4.2 Syntax

Turkish has a free-order sentence structure with features such as head-final, pro-drop, and subject-verb agreement. Here, we briefly describe the grammatical structure of Turkish, which will help to understand the UCCA annotation guideline proposed for Turkish in this study.

*Constituents of a sentence.* In Turkish, sentences can be simple or complex in terms of the subordinate clauses that are semantically dependent on the main clause.[j] Two main components of a simple sentence are subject and predicate. In a sentence, the **predicate** expresses an event, process, or state of affairs with an agreement of the subject, and the **subject** of the sentence may be a person, place, or thing that possesses or is affected by the predicate:

**Example 4.10.** *Kerem bir an durdu. (in English, "Kerem stopped for a moment.")*

**Example 4.11.** *O öğretmendi. (in English, "S/he was a teacher.")*

Although Turkish is a syntactically free-order language, the unmarked order (neutral word order) is subject-(object)-predicate (SOV) (e.g., "Ahmet okula gitti." (in English, *"Ahmet went to the school."*)).

The sentence types are divided into two classes based on the predicate:

1. Verbal sentences have predicates that are finite verbs and they indicate a process in a given sentence:

   **Example 4.12.** *Kerem bana baktı. (in English, "Kerem looked at me.")*

2. Nominal sentences have predicates that do not contain a verb or use a verb in the form of a copula (e.g., ol- (be)). They indicate a state in a given sentence:

   **Example 4.13.** *Bir tutsağım ben. (in English, "I am a prisoner.")*

   **Example 4.14.** *Ahmet doktor olacaktı. (in English, "Ahmet was going to be a doctor.")*

*Phrases.* A phrase is a syntactic unit that is a collection of syntactically coherent words that function as a unit in a sentence and can be nested. For example, VPs can contain NPs. There are five types of phrases: verb phrases, noun phrases, adjectival, adverbial, and postpositional phrases.

1. A verb phrase consists of a verb and additionally may contain a complement of a verb or an adverbial. A verb usually refers to a process in a sentence.

   **Example 4.15.** *Ahmet suyu soğuk soğuk içti. (in English, "Ahmet drank the cold water too fast.")*

---

[j]Subordinate clauses are indicated by [] as defined in Göksel and Kerslake (2004).

2. A noun phrase primarily contains a noun or a group of words containing a noun. The role of a noun phrase is that of a subject and some kind of complement such as an object, subject complement, and complement of a postposition.

**Example 4.16.** ***Ahmet*** *geldi. (in English, "**Ahmet** came.")*

**Example 4.17.** *[**Ahmet'in sevdiği] ekşi elma** bitmişti. (in English, "[**Ahmet's favorite] sour apple** was finished.")*

Noun phrases usually correspond to participants (e.g., "Ahmet'in sevdiği ekşi elma") in a sentence where they may have their own participants (e.g., "Ahmet") and processes (e.g., "sevdiği") in the phrase and may constitute a sub-scene in the entire sentence.

3. An adjective phrase contains a group of words that act as modifiers of nouns, providing additional information about the quality, characteristics, or attributes of the nouns they describe.

**Example 4.18.** *Bu restoran, **şehirdeki en iyi** yemekleri sunuyor. (in English, This restaurant offers **the best meals in the city**)*

4. An adverb phrase is composed of a group of words that act as modifiers of adjectives, verbs, or other adverbs to provide additional information about the cause, effect, space, time, and condition.

**Example 4.19.** ***Oldukça yavaş*** *koştu. (in English, S/he ran **quite slowly**)*

**Example 4.20.** ***Sabah*** *evde çalışacağım. (in English, I will work at home **in the morning**)*

5. A postpositional phrase is usually composed of a noun phrase and a postposition that follows the noun phrase. While the postposition is the head of a phrase, the noun phrase is the complement of the phrase:

**Example 4.21.** ***Akşama doğru*** *yağmur yağabilir. (in English, "It may start raining **towards evening**.")*

Depending on its meaning, a postpositional phrase may correspond to an adverbial clause in a sentence.

*Complex sentences and subordination.* Complex sentences contain at least one subordinate clause in addition to the main clause, where a subordinate clause is a group of words that functions as a unit in a sentence. In Turkish, there are three types of clauses: noun, relative, and adverbial clause.

1. Noun Clause occurs as a noun phrase in a complex sentence as the subject or object in the sentence:

**Example 4.22.** *Ahmet [**senin okula gittiğini**] bilmiyordu. (in English, "Ahmet didn't know that [**you were going to school**].")*

2. Relative Clause occurs as an adjectival phrase that is used to modify noun phrases:

**Example 4.23.** *[**Öğretmen olan**] oğlu, Ankara'da yaşıyor. (in English, "Her/His son, [**who is a teacher**], lives in Ankara.")*

3. Adverbial Clause occurs as an adverbial phrase that expresses the verb in terms of time, place, manner, and degree:

   **Example 4.24.** *[Kedimiz FC;ş FC;mesin diye] ısıtıcıyı açtık. (in English, "We turned on the heater [so that our cat would not get cold].")*

*Conditional sentences.* Conditional sentences are based on a condition that is met when a certain action takes place.

**Example 4.25.** *[Okula gitmezsen] sınıfta kalacaksın. (in English, "[If you do not go to school] you will fail the class.")*

### 4.2.1 UCCA annotation guideline based on syntax

Here, we describe the rules that do not exist in the original UCCA guideline in English and that we have defined on the basis of Turkish syntax:

1. **Nominal sentences without copula marker:** If the predicate is not indicated by copula markers (e.g., "-(y)DI" (was), "-(y)mIş" (was), and "-(y)sA" (if it is)), we inserted a generalising modality marker "-DIr" (is) as an IMPLICIT unit.

   **Example 4.26.**

$$Nasıl \quad bir \quad kadın\langle(dır)\rangle_{S-IMPLICIT} \quad bu$$
$$How \quad a \quad woman\ is \quad this$$
$$What\ kind\ of\ woman\langle is\rangle this$$

2. **Pronoun-dropping (Göçmen *et al.*, 1995):** Pronoun subjects may be omitted in a sentence since Turkish is a pro-drop language. Omitted pronoun subjects are marked with the label A-IMPLICIT.[k]

   **Example 4.27.**

$$\langle(Ben)\rangle_{A-IMPLICIT} \quad bilmiyorum$$
$$I \quad I\ don't\ know$$
$$\langle I\rangle\ don't\ know$$

3. **Juxtaposition:** One of the most common methods to coordinate two or more phrases or sentences is simply to list them without using a connector. We combined such phrases and labelled them as a whole. Additionally, we labelled each of them as a Parallel Scene.

   **Example 4.28.**

$$\langle Bağışlayın\rangle_H \quad \langle koşa \quad koşa \quad geldim \quad buraya\rangle_H$$
$$Forgive \quad running \quad running \quad Icame \quad here$$
$$\langle Forgive\ me\rangle \langle I\ came\ running\ here\rangle$$

---

[k]The word given in () indicates the omitted word in the examples.

**Example 4.29.**

| ⟨*Onu* | *elinden* | *kaçırmış*⟩_H | ⟨*onu* | *bir* | *başka* | *erkeğe* | *kaptırmıştı*⟩_H |
|---|---|---|---|---|---|---|---|
| *Her/him* | *from her/his hand* | *lost* | *s/he* | *a* | *another* | *man* | *had lost* |

⟨*S/he missed her/him*⟩ ⟨*S/he had lost her/him to another man*⟩

4. **Subordinators (Zeyrek and Webber, 2008):** Subordinators link the clauses to superordinate clauses. A subordinator is a word or suffix that introduces a subordinate clause and connects it to a main clause or another clause.

   (a) **Word subordinators** The word subordinators in Turkish are "diye", "ki", "mI", and "dA", as well as other obsolescent subordinators that contain "ki' ("ola ki", "meğer ki", etc.). Since we already described the other clitics above, we only describe "diye" here. "**diye**" relates the clause to a superordinate clause. Therefore, it is labelled as `Relator`.

   **Example 4.30.**

   | *Nereden* | *biliyorsunuz* | ⟨*diye*⟩_R | *sordu* |
   |---|---|---|---|
   | *From where* | *you know* | *that* | *he asked* |

   *He asked* ⟨*(that)*⟩[1] *how do you know*

   (b) **Suffix subordinators**

   • **Simplex subordinators:** In Turkish "*-(y)Ip*" and "*-(y)ArAk*" are used as subordinating suffixes in simplex subordinators for nominal and adverbial clauses to link clauses. We labelled the clauses with "*-(y)Ip*" as `Parallel Scene` since they define a new Scene, whereas we labelled the clauses with "*-(y)ArAk*" as `D-Scene` since they modify the relation (Process or State) in the Scene.

   **Example 4.31.**

   | [*Dar* | *yollarda* | ⟨*koşarak*⟩_D | *giden*]_D | *Kerem'i* | *yakaladım* |
   |---|---|---|---|---|---|
   | *Narrow* | *on roads* | *by running* | *went* | *Kerem + dative* | *I caught* |

   *I caught Kerem*[⟨*running*⟩*on narrow roads*]

   **Example 4.32.**

   | [*Kerem* | *çayına* | *iki* | *şeker* | ⟨*atıp*⟩_P]_H | [*yavaşça* | *karıştırdı*]_H |
   |---|---|---|---|---|---|---|
   | *Kerem* | *to his tea* | *two* | *sugar* | *added* | *slowly* | *stirred* |

   [*Kerem* ⟨*added*⟩ *two sugars to his tea*] *and* [*stirred it slowly*]

   • **Complex subordinators:** They consist of a postposition and a nominalising suffix. We labelled the postposition in the complex subordinators as a `Linker` since it connects the adverbial clause evoking a Scene with the main Scene.

---

[1]*The word "that" is omitted in the English corresponds to "diye" in Turkish.*

**Example 4.33.**

$$[Ona \quad yetişebilmek_P]_H \quad \langle için \rangle_L \quad [peşinden \quad koşuyordum]_H$$

*Her/him    to catch up    for    after her/him    I was running*

*I was running after her/him ⟨to⟩ catch up*

5. **Complex sentences:** There are also Adverbial clauses in Turkish because of the derivational suffixes. Therefore, we defined D-Scene to label such clauses.

**Example 4.34.**

$$Dar \quad yollarda \quad \langle koşarak \rangle_D \quad giden \quad Kerem'i \quad yakaladım$$

*Narrow    on ways    running    who let    Kerem    I caught*

*I caught Kerem ⟨running⟩ on narrow roads*

### 4.3 Closed-class words

Closed-class words in Turkish, also known as function words, are crucial for constructing sentences in Turkish, providing grammatical structure, and conveying relationships between different elements in a sentence. They are considered closed-class because they are relatively stable and less prone to change compared to open-class words such as nouns and verbs. Although they fall into pre-existing UCCA categories, we describe them separately because they are language-specific. The closed-class word categories in Turkish are as follows:

*Pronouns.* Pronouns refer to the persons, objects, or situations mentioned above, their references being clear from the context or only partially specified. They have different forms based on inflectional suffixes added to them, such as number and possession.

*Postpositions.* Postpositions are closed-class words that follow a noun or a pronoun to indicate its relationship to another element in a sentence. They also indicate relationships such as location, direction, and possession.

*Clitics.* Clitic is a type of bound morpheme that functions similarly to an affix but is usually in the form of a free morpheme (e.g., *mI*, *dA*, *ya*, *ki* and *bile*, *ile*). They don't have stress of their own and depend on the host word for their pronunciation.

*Conjunctions and discourse connectives.* In Turkish, conjunctions and discourse connectives are used to link two or more items that have the same syntactic function. While conjunctions are used to link phrases, subordinate clauses, or sentences, discourse connectives are used to link sentences.

#### 4.3.1 UCCA annotation guideline based on closed-class syntactic words

Closed-class words are language-specific, but due to the inherent meanings of closed-class words, we need to define UCCA rules for efficient and fast annotation during the annotation procedure.

1. **Clitics:** Clitics have to be carefully annotated because they may have different meanings in different contexts. The new rules defined for the Turkish clitics are given below:

   - "**mI**"[m] can be used in three different meanings:

---

[m]The clitic also involves the other forms of "mı", such as "mi", "mısın", "mısınız", etc. depending on the vowel harmony and the person type.

(a) **Yes/No condition:** It is used to make a question sentence. In this case, it is labelled with a `Function` since it does not refer to a participant or a relation.

**Example 4.35.**

| *Sakinleştin* | $\langle mi \rangle_F$ | *biraz* |
|---|---|---|
| *You calmed down* | *have* | *abit* |

$\langle$*Have*$\rangle$ *you calmed down a bit?*

(b) **Adverbial clause marker:** It has the meaning of *as soon as* or *once* and connects two clauses. Therefore, it is labelled as `Linker` which corresponds to the same category in the UCCA guideline♦.

**Example 4.36.**

| *Okula* | *gittin* | $\langle mi \rangle_L$ | *arkadaş* | *edinirsin* |
|---|---|---|---|---|
| *To school* | *you went* | *when* | *friend* | *you make* |

$\langle$*When*$\rangle$ *you go to school, you make friends*

(c) **Intensifier in doubled forms:** It is used in a doubled form to connect the two same adjectives in order to intensify the quantity. We combine the doubled forms (e.g., "karanlık mı karanlık" (in English, very dark)) and label as a whole♦.

**Example 4.37.**

| $\langle$*Karanlık* | *mı* | *karanlık*$\rangle_E$ | *bir* | *yolda* | *yürüyoruz* |
|---|---|---|---|---|---|
| *Dark* | — | *dark* | *a* | *inroad* | *we are walking* |

*We are walking in a $\langle$very dark$\rangle$ road*

- "**dA**" can be used in six different meanings:

(a) **Additive function:** It is labelled as an `Adverbial` since it attributes the meaning of "moreover" to a sentence or a clause♦.

**Example 4.38.**

| *Filmi* | *izlemedim* | $\langle de \rangle_D$ |
|---|---|---|
| *The movie* | *I did not watch* | *moreover* |

$\langle$*Morover*$\rangle$ *I did not watch the movie*

(b) **Adversative Function:** It is labelled as a `Linker` since it links two or more sentences with the meaning of "but"♦.

**Example 4.39.**

| *Filmi* | *izlemedim* | $\langle de \rangle_L$ | *anlattılar* |
|---|---|---|---|
| *The movie* | *I didn't watch* | *but* | *they told* |

*I didn't watch the movie $\langle$but$\rangle$ they told me*[n]

---

[n]*The word "me" does not correspond to a word in Turkish, but it is expressed rather implicitly as a morpheme in the verb "anlattılar" (in English, "they told me").*

(c) **Continuative/topic shifting:** It has the meaning of "also" or "either", depending on the position of the clitic in the sentence. It is labelled as `Elaborator` if the clitic refers to the Participant, and as `Adverbial` if it refers to the adverb.

**Example 4.40.**

| O | $\langle da\rangle_E$ | burada | bekler |
|---|---|---|---|
| S/he | also | here | wait |

S/he ⟨also⟩ waits here

**Example 4.41.**

| Sanırım | o | zaman | $\langle da\rangle_D$ | gelmemişti |
|---|---|---|---|---|
| I guess | that | time | either | s/he didn't come |

I guess s/he didn't come then ⟨either⟩

(d) **Enumerating:** This role is similar to Continuative/topic shifting and is also labelled as `Elaborator`♦.

**Example 4.42.**

| Ayşe | $\langle de\rangle_E$ | Sema | $\langle da\rangle_E$ | filmi | izlediler |
|---|---|---|---|---|---|
| Ayşe | both | Sema | and | the movie | watched |

⟨Both⟩ Ayşe ⟨and⟩ Sema watched the movie

(e) **Modifier of adverb:** It is labelled as `Adverbial` since it modifies the Adverbial in the annotation.

**Example 4.43.**

| Şimdi | adımlarını | daha | $\langle da\rangle_D$ | hızlandırmıştı |
|---|---|---|---|---|
| Now | her/his steps | more | even | had accelerated |

Now s/he had accelerated her/his steps ⟨even⟩ more

(f) **Discourse connective:** It is labelled as `Linker` since it connects Scenes in the sentence.

**Example 4.44.**

| Şimdi | düşünüyorum | $\langle da\rangle_L$ | galiba | o | parkın | dışında | yapamayacağım | ben | dedi | Kerem |
|---|---|---|---|---|---|---|---|---|---|---|
| Now | I'm thinking | that | I guess | that | park | outside | I can't do | I | said | Kerem |

Now I'm thinking about it, ⟨and⟩[o] I guess I can't do it outside of that park, Kerem said

---

[o] *The word "and" does not correspond to a word in Turkish, but it is expressed by clitic "da".*

- "**ya**" has four different meanings:

  (a) **Contrastive adversative conjunction:** It is labelled as a `Linker` that adds a contrastive meaning such as "but"♦.

   **Example 4.45.**

   | *İzledim* | *⟨ya⟩_L* | *anlamadım* |
   |---|---|---|
   | *I watched* | *but* | *I couldn't understand it* |

   *I watched ⟨but⟩ couldn't understand it*

  (b) **Repudiative discourse connective:** It generally occurs at the end of a sentence and is usually punctuated with an exclamation mark to express the speaker's opinion with a firm tone. Therefore, it is labelled as `Ground`♦.

   **Example 4.46.**

   | *Gitmedim* | *dedim* | *⟨ya⟩_G* |
   |---|---|---|
   | *I did not go* | *I told* | — |

   *I told you I did not go*

  (c) **Reminding discourse connective:** It generally occurs in a Scene-final position that has the same role as the Repudiative discourse connective, but this time it is used for reminding purposes. Therefore, it is also labelled as `Ground`♦.

   **Example 4.47.**

   | [*Sana* | *söylemiştim* | *⟨ya⟩_G*]_H | [*işte* | *o* | *okul*]_H |
   |---|---|---|---|---|---|
   | *To you* | *I told* | *that* | *here it is* | *that* | *school* |

   *I told you that is the school*

  (d) **Stressable discourse connective:** It precedes a phrase and introduces an alternative question. Since it precedes a phrase, we label it as a `Linker`♦.

   **Example 4.48.**

   | *⟨Ya⟩_L* | *orayı* | *bir daha* | *bulamazsam* |
   |---|---|---|---|
   | *What if* | *there* | *gain* | *What if I cannot find* |

   *⟨What if⟩ I cannot find there again*

- "**ki**" has four different meanings:

  (a) **Subordinator connective:** It connects a noun, an adverbial clause, or a clause with a sentence. If it connects a noun or a noun clause it is labelled as a `Relator`, otherwise as a `Linker`.

   **Example 4.49.**

   | *İnanıyorum* | *⟨ki⟩_R* | *onlar* | *gelecek* |
   |---|---|---|---|
   | *I believe* | *that* | *they* | *will come* |

   *I believe ⟨that⟩ they will come*

**Example 4.50.**

*O kadar iyisin ⟨ki⟩ₗ özgürlüğüm     kısıtlanıyor     istediğim gibi davranamıyorum*

*—   so    good   that   my freedom   is being restricted   I want    as      I can't act*

*You're so good⟨that⟩my freedom is being restricted I can't act as I want*

(b) **Repudiative discourse connective:** Since it attributes the meaning of "just" or "such" to a sentence and expresses the opinion of the speaker, it is labelled as Ground.

**Example 4.51.**

*Öyle    bir     şey     ⟨ki⟩_G*

*Such    a      thing    —*

*It is such a thing*

(c) **Exclamations:** It is labelled as a Ground since it has the meanings of "o kadar" (such) or "öyle(sine)" (so) and expresses the speaker's opinion.

**Example 4.52.**

*Anlatacağım     öyle    bir     şey     ⟨ki⟩_G*

*I will tell     such    a      thing     —*

*I will tell something like that*

(d) **Relative clause marker**: It has two types: non-restrictive relative clauses and restrictive relative clauses. Both types connect a clause with a sentence. Therefore, it is labelled as a Relator.

**Example 4.53.**

*Ahmet    ⟨ki⟩_R   müzik    sevmez    o     bile    geldi*

*Ahmet    who    music    not like   he    even    came*

*Even Ahmet ⟨who⟩ does not like music came)*

**Example 4.54.**

*Bugün   yağmur   yağarsa   ⟨ki⟩_R   yağacağını   hiç   sanmıyorum    maç    iptal   olacakmış*

*Today    rain    if rains   which    it will    not   I don't think   the match   cancel   would be*

*'If it rains today ⟨which⟩ I don't think it will the match would be cancelled*

- "**bile**" It is an additive connective that has the meanings such as "already" or "even". Therefore, it is labelled as an Adverbial♦.

**Example 4.55.**

*Gönderdim     ⟨bile⟩_D*

*I sent        already*

*I have ⟨already⟩ sent it*

2. **Conjunctions and discourse connectives:** We only provide the rules for conjunctions and discourse connectives that are not covered in the other rules described above.

- "**halbuki/oysa**": The course connectives "halbuki"/"oysa" (whereas/however) indicate a contrast between the two states. Since it is used to link different states, it is labelled with a Linker.

  **Example 4.56.**

  | ⟨*Oysa*⟩*L* | *benim* | *sizlere* | *ne* | *kadar* | *çok* | *anlatacağım* | *vardı* |
  |---|---|---|---|---|---|---|---|
  | *However* | *me* | *to you* | *what* | *much* | *so* | *I will tell* | *had* |

  ⟨*However*⟩ *I had so much to tell you*

- "**peki**" It expresses the speaker's agreement on the subject. Therefore, it is labelled as Ground.

  **Example 4.57.**

  | ⟨*Peki*⟩*G* | *senin* | *yerin* | *neresi* |
  |---|---|---|---|
  | *So* | *your* | *place* | *where is* |

  ⟨*So*⟩ *where is your place*

- "**Demek**" It is used at the end or beginning of a sentence and adds inferential meaning by referring to the previous sentence. Since it also contains the attitude of the speaker, it is labelled as Ground.

  **Example 4.58.**

  | *Siz* | *o* | *dünyayı* | *biliyorsunuz* | ⟨*demek*⟩*G* |
  |---|---|---|---|---|
  | *You* | *that* | *world* | *you know* | *so* |

  ⟨*So*⟩ *you know that world*

- "**Yoksa**" It is an inferential connective and is used in yes/no questions. It is used when the speaker realises that the situation is different from what s/he expects. It is labelled as Adverbial.

  **Example 4.59.**

  | ⟨*Yoksa*⟩*D* | *biliyor* | *musunuz* | *orayı* | *diye* | *hayretle* | *sordu* |
  |---|---|---|---|---|---|---|
  | — | *know* | *do you* | *that place* | *that* | *amazedly* | *he asked* |

  *Do you know where it is?" he asked amazedly*

3. **Postposition "gibi":** The role of "*gibi*" is derived from its primary function as a postposition. It expresses the opinion of a speaker. Therefore, it is labelled as Ground.

   **Example 4.60.**

   | *Ne* | *tuhaf* | *başka* | *şeyler* | *önem* | *kazandı* | ⟨*gibi*⟩*G* |
   |---|---|---|---|---|---|---|
   | *What* | *weird* | *other* | *things* | *matter* | *gain* | *like* |

   *How weird other things* ⟨*seem*⟩ *to matter*

**Table 1.** A sentence "Ama hiçbir şey söylemedim ki ben sizlere" (in English, *"But I didn't say anything to you"*) in the METU-Sabanci Turkish Treebank (Atalay *et al.,* 2003; Oflazer *et al.,* 2003). The columns correspond to the positions of the words within the sentence, surface forms, lemmas, parts-of-speech (PoS) tags, morphological features separated by |, head-word indices (index of a syntactic parent, 0 for ROOT), and syntactic relationships between HEAD and the word, respectively.

| Id | Form | Lemma | PoS | Morphological features | Head | Dependency Rel. |
|---|---|---|---|---|---|---|
| 1 | Ama | ama | Conj | _ | 8 | S.MODIFIER |
| 2 | hiçbir | hiçbir | Det | _ | 3 | DETERMINER |
| 3 | şey | şey | Noun | A3sg|Pnon|Nom | 4 | OBJECT |
| 4 | söylemedim | söyle | Verb | Neg|Past|A1sg | 8 | SENTENCE |
| 5 | ki | ki | Conj | _ | 4 | INTENSIFIER |
| 6 | ben | ben | Pron | A1sg|Pnon|Nom | 4 | SUBJECT |
| 7 | sizlere | siz | Pron | A2pl|Pnon|Dat | 4 | DATIVE.ADJUNCT |
| 8 | . | . | Punc | _ | 0 | ROOT |

## 5. Turkish UCCA annotation

We annotated 400 sentences obtained from the METU-Sabanci Turkish Treebank (Atalay *et al.,* 2003; Oflazer *et al.,* 2003). Here, we present the details of the dataset used in this study along with the external semantic parser used during the annotation. We explain the annotation process defined in Figure 1.

### 5.1 Dataset

METU-Sabanci Turkish Treebank (Atalay *et al.,* 2003; Oflazer *et al.,* 2003) is a morphologically and syntactically annotated treebank. It consists of 7262 sentences taken from 10 different genres, that is, narratives, memories, research papers, travel writings, diaries, newspaper columns, articles, short stories, novels, and interviews. Since the corpus is annotated at both lexical and syntactic levels, it is one of the most valuable syntactically annotated datasets in Turkish. By adding graph-based semantic representations of the sentences in the same dataset, we aim to make the dataset more accessible for other NLP tasks that require semantic information as well as syntactic and morphological information.

A sample annotation from the METU-Sabanci Turkish Treebank is given in Table 1 for Example 5.1. As seen, each sentence involves PoS tags, dependencies, dependency labels, as well as morphological tags of each word. The UCCA annotation of the sentence is illustrated in Figure 3.

**Example 5.1.**

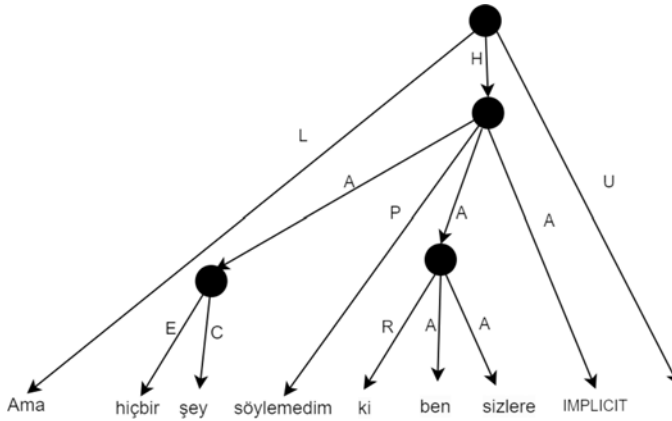| Ama | hiçbirşey | söylemedim | ki | ben | sizlere |
|---|---|---|---|---|---|
| *But* | *anything* | *I didn't say* | *—* | *I* | *to you* |
| *Conj* | *Det Noun* | *Verb* | *Conj* | *Pron* | *Pron* |

*But I didn't say anything to you*

**Figure 3.** UCCA Annotation of "Ama hiçbir şey söylemedim ki ben sizlere" (in English, *"But I didn't say anything to you"*)
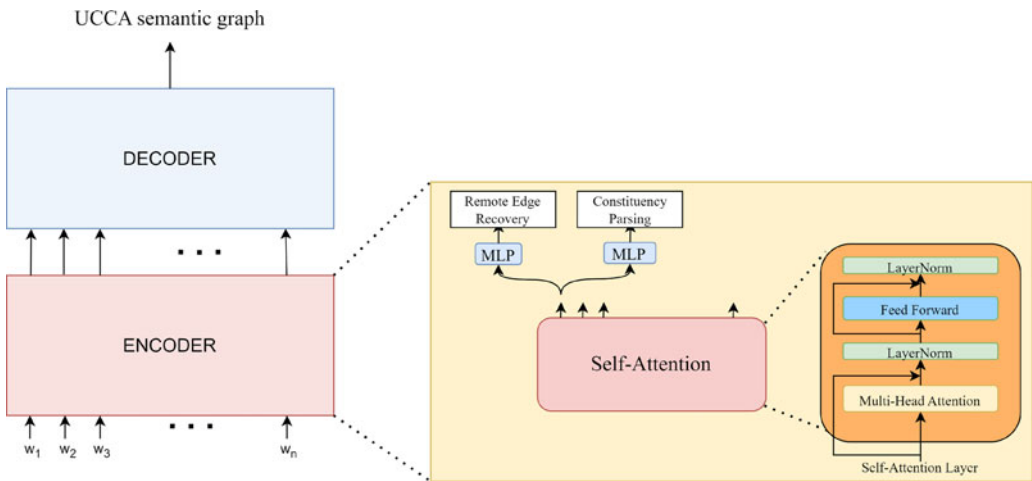


**Figure 4.** An overview of the external semantic parser

### 5.2 Self-attentive semantic parser

We utilised an external semantic parser to perform the annotation with a semi-automatic approach that significantly reduced the annotation effort. For this purpose, we used the semantic parser proposed by Bölücü *et al.*, (2023). The model is based on an encoder–decoder architecture that tackles parsing as a chart-based constituency parsing problem. In the encoder, self-attention layers are used along with two multi-layer perception (MLP) classifiers, two fully connected layers, and a non-linear activation function ReLU in the output layer. The output layer of the encoder produces the per-span scores where spans correspond to the constituents in the constituency tree. The CYK (Cocke-Younger-Kasami) algorithm (Chappelier and Rajman, 1998) is used in the decoder to generate the tree with the maximum score using the scores that are produced by the encoder. The architecture of the model is given in Figure 4.

The input given to the encoder is a sentence $s = \{w_1, \cdots, w_n\}$, where each word $w_t$ is mapped into a dense vector that is the concatenation of multilingual contextualised embeddings $con_t$ and syntactic embeddings (i.e., PoS tags, dependency tags, entity types, and entity iob types)— extracted from an off-the-shelf open-source NLP library— which are word embeddings $e_{w_t}$,

PoS tag embeddings $e_{p_t}$, dependency label embeddings $e_{d_t}$, entity type embeddings $e_{e_t}$, and entity iob (inside-outside-beginning) category embeddings $e_{e_o b_t}$:

$$x_t = e_{w_t} \oplus e_{p_t} \oplus e_{d_t} \oplus e_{e_t} \oplus e_{e_o b_t} \tag{1}$$

$$w_t = x_t \oplus con_t \tag{2}$$

The output of the semantic parser model is a UCCA (Abend and Rappoport, 2013a) representation of the sentence in the form of a directed graph.

### 5.3 Dataset annotation

As stated in the study, dataset annotation consists of two steps: (1) obtaining a partially annotated dataset using an external semantic parser (see Section 5.2), and (2) refining the partially annotated dataset by human annotators. Here, we explain the steps of annotating the Turkish UCCA dataset.

#### 5.3.1 Partial annotation

Zero-shot learning is a machine learning paradigm used when there is a lack of annotated examples in the target language. Various approaches, such as transfer learning and meta-learning, are used to transfer knowledge between languages. Transfer learning often involves using a pretrained model as a starting point and fine-tuning it on the target language. Meta-learning, on the other hand, requires minimal additional training.

Due to the lack of a UCCA-annotated dataset in Turkish, we used zero-shot learning instead of alternative approaches. The parser was trained on a merged dataset consisting of English (Abend and Rappoport, 2013a), German (Hershcovich *et al.* 2019b), and French (Sulem *et al.,* 2015)[p] datasets (without using any Turkish examples in training) and used the trained parser to get the partially annotated Turkish dataset (i.e., 400 sentences from the METU-Sabanci Turkish Treebank).[q]

#### 5.3.2 Refinement

The semantic representations of the Turkish sentences (partially annotated) obtained from the parser model are partially correct (with some errors that need to be corrected). We manually revised the representations based on the Turkish guidelines (see Section 4). Finally, we obtained the gold semantic representations of the 400 Turkish sentences.

### 5.4 Inter-annotator agreement

We employed two annotators who are native Turkish speakers and fluent in English and have expertise and knowledge in computational linguistics. The annotators were initially trained for UCCA annotation based on the official UCCA guideline (Abend and Rappoport, 2013a). The annotation step was performed individually, which was followed by a comparison phase in which the two annotations were compared to each other to identify annotation agreements. To show how clear our annotation guideline is and how consistently it was understood by the annotators, we used accuracy and Cohen's kappa ($\kappa$) (Cohen, 1960). There was disagreement on 616 edges out of 3, 981 edges in a total of 400 sentences. Thus, the disagreement between the two annotators is 15.47% and Accuracy is 84.53% based on the annotated tokens. Accuracy does not consider the expected chance of agreements that are likely to occur. Therefore, we also calculated

---

[p]The semantic parser is trained with the same hyperparameters as in the cross-lingual experiments as given by Bölücü *et al.* (2023).
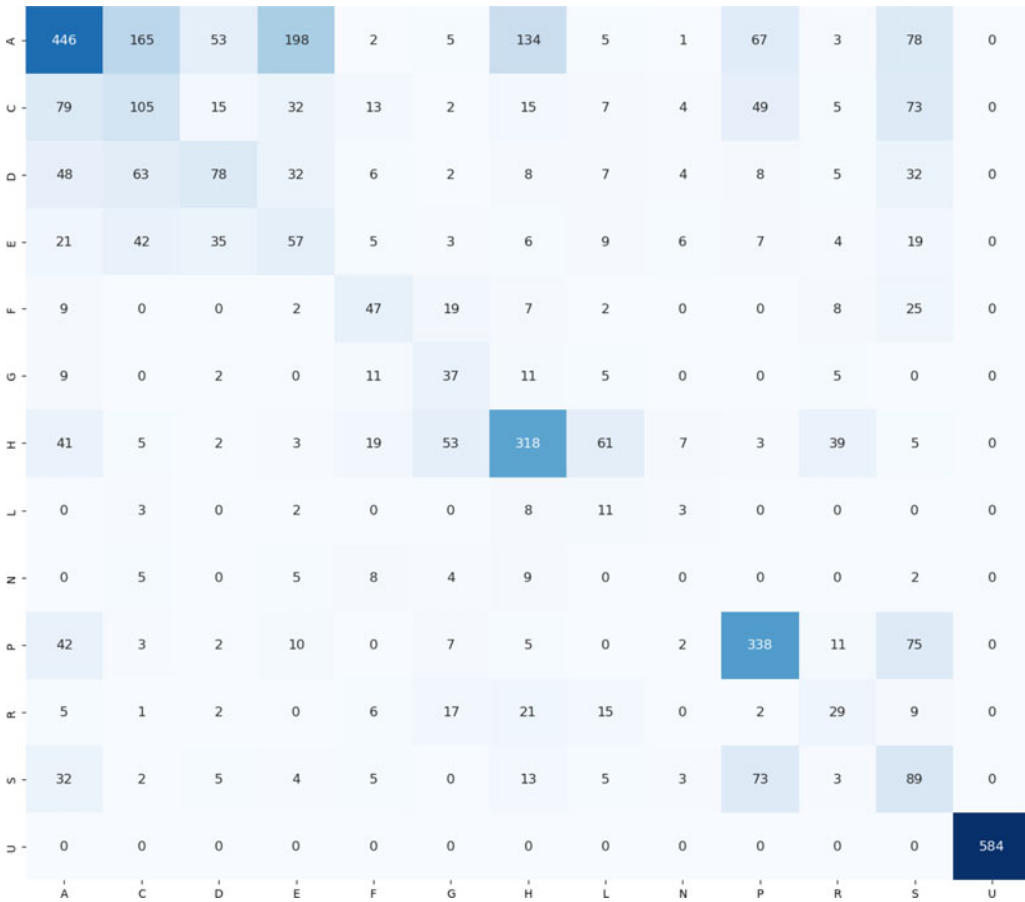
[q]Training details can be found in Section 6.1.2.

| | A | C | D | E | F | G | H | L | N | P | R | S | U |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **A** | 446 | 165 | 53 | 198 | 2 | 5 | 134 | 5 | 1 | 67 | 3 | 78 | 0 |
| **C** | 79 | 105 | 15 | 32 | 13 | 2 | 15 | 7 | 4 | 49 | 5 | 73 | 0 |
| **D** | 48 | 63 | 78 | 32 | 6 | 2 | 8 | 7 | 4 | 8 | 5 | 32 | 0 |
| **E** | 21 | 42 | 35 | 57 | 5 | 3 | 6 | 9 | 6 | 7 | 4 | 19 | 0 |
| **F** | 9 | 0 | 0 | 2 | 47 | 19 | 7 | 2 | 0 | 0 | 8 | 25 | 0 |
| **G** | 9 | 0 | 2 | 0 | 11 | 37 | 11 | 5 | 0 | 0 | 5 | 0 | 0 |
| **H** | 41 | 5 | 2 | 3 | 19 | 53 | 318 | 61 | 7 | 3 | 39 | 5 | 0 |
| **L** | 0 | 3 | 0 | 2 | 0 | 0 | 8 | 11 | 3 | 0 | 0 | 0 | 0 |
| **N** | 0 | 5 | 0 | 5 | 8 | 4 | 9 | 0 | 0 | 0 | 0 | 2 | 0 |
| **P** | 42 | 3 | 2 | 10 | 0 | 7 | 5 | 0 | 2 | 338 | 11 | 75 | 0 |
| **R** | 5 | 1 | 2 | 0 | 6 | 17 | 21 | 15 | 0 | 2 | 29 | 9 | 0 |
| **S** | 32 | 2 | 5 | 4 | 5 | 0 | 13 | 5 | 3 | 73 | 3 | 89 | 0 |
| **U** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 584 |

**Figure 5.** Confusion matrix for the outputs in partial annotation (predicted) and refined annotation (gold). Category abbreviations: **A**: Participant, **C**: Center, **D**: Adverbial, **E**: Elaborator, **F**: Function, **G**: Ground, **H**: Parallel Scene, **L**: Linker, **N**: Connector, **P**: Process, **R**: Relator, **S**: State, **U**: Punctuation

Cohen's kappa ($\kappa$), a statistical measure of the reliability of annotations between different annotators. Cohen's kappa ($\kappa$)($*100$) score for the annotation is 82.29. The scores between 80 and 90 indicate strong agreement between the annotators. The general disagreement that recurs in the training procedure mainly concerns the annotation of the clitics (e.g., mI, dA).

### 5.5 Comparison of the outputs in partial and refined annotation

Figure 5 illustrates the confusion matrix acquired from the outputs of the partial annotation (predicted) and refined annotation (gold). We analyse the results of the semantic parser model, which correspond to the partial annotations, to identify the most commonly corrected errors in the refined annotation. We added all *IMPLICIT* edges manually because the model cannot predict the *IMPLICIT* edges. Typically, the external parser model does not confuse the Punctuation (U) label, as it is a straightforward task to recognise punctuation. However, the model often has difficulty distinguishing between labels such as Participant (A), Center (C), Adverbial (D), Elaborator (E), and State (S). This makes sense because Turkish allows the creation of new words by adding derivational morphemes, and words with these labels may share common roots. Other commonly confused labels include Relator (R), Ground (G), and Linker (L), which are
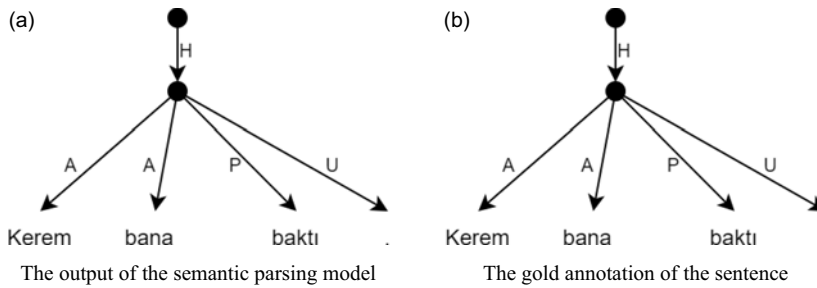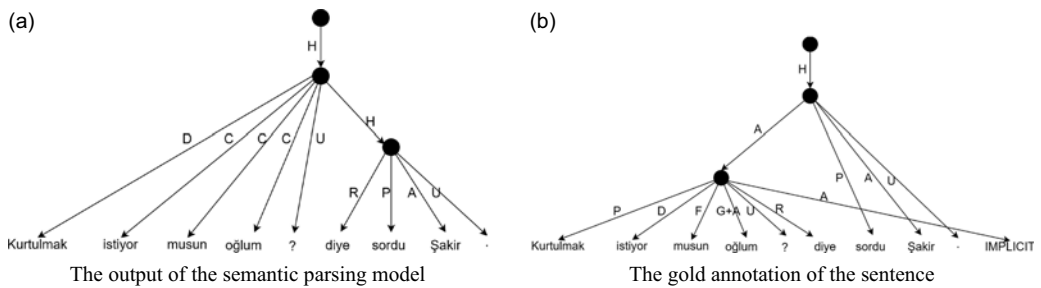
**Figure 6.** The semantic parse tree obtained from the semantic parsing model and the gold annotation obtained from the manual annotation of the sentence *"(O) Yerinden kalkmıştı."* (in English, *"S/he had stood up."*). Category abbreviations: **H**: Parallel Scene, **A**: Participant, **P**: Process, **U**: Punctuation



**Figure 7.** The semantic parse tree obtained from the semantic parsing model and the gold annotation obtained from the manual annotation of the sentence *"(Sen) Kurtulmak istiyor musun oğlum? diye sordu Şakir."* (in English, *"Do you want to be saved son? asked Şakir."*). Category abbreviations: **H**: Parallel Scene, **D**: Adverbial, **C**: Center, **U**: Punctuation, **R**: Relator, **P**: Process, **A**: Participant, **F**: Function, **G**: Ground

known to be challenging even for human annotators. In addition, `Relator` (R) presents complexity due to the presence of clitics (please see Section 4 for the relevant rules), which leads the semantic parser to occasionally misclassify it as `Ground` (G), `Parallel Scene` (H), or `Linker` (L). Finally, despite the absence of any words labelled as `Connector` (N) in the refined annotation dataset, the model annotated some words as `Connector` (N) possibly due to their similarity to words in other languages (English, German, and French).

Two example sentences parsed with the external semantic parser are given in Figure 6 and 7 along with their gold (refined) annotations. For the first sentence in Figure 6, we added only an `IMPLICIT` edge based on the annotation rule for pronoun-dropping. The model cannot learn the implicit edges, which are all added manually during annotation. Morphological and more contextual information (possibly in paragraph level) should be provided to learn such implicit relations. For the second sentence given in Figure 7, most of the labels were incorrect and therefore had to be corrected manually.

When correcting the annotations obtained from the semantic parser model, we made almost no additional corrections for the short sentences (with less than five words). The labels of the majority of the terminal nodes were also mostly correct. We corrected most of the annotations with a `Parallel Scene` and, for that matter, the entire annotation of the sentence.

### 5.6 Annotation Statistics

We provide the statistical distributions in the annotated dataset with a comparison to other annotated datasets in English, German, and French (see Table 2). In particular, we provide the proportions of the edges and the labels in the final annotated dataset.

**Table 2.** Proportions of the edges and labels as well as the number of sentences and tokens in the UCCA datasets in Turkish, English, French, and German. The statistical details of English, French, and German datasets are taken from Hershcovich *et al.* (2019b).

|  | Turkish | En-Wiki | En-20K | Fr-20K | De-20K |
|---|---|---|---|---|---|
| # sentences | 400 | 5,141 | 492 | 492 | 6,154 |
| # tokens | 2,474 | 158,739 | 12,638 | 13,021 | 144,529 |
| # edges | 3,981 | 208,937 | 16,803 | 17,520 | 187,533 |
| % primary | 96.66 | 97.40 | 96.79 | 97.02 | 97.32 |
| % remote | 3.34 | 2.60 | 3.21 | 2.98 | 2.68 |
| % Participant (A) | 26.78 | 17.17 | 18.1 | 17.08 | 19.86 |
| % Center (C) | 10.02 | 18.74 | 16.31 | 18.03 | 14.32 |
| % Adverbial (D) | 7.36 | 3.65 | 5.25 | 4.18 | 5.67 |
| % Elaborator (E) | 5.38 | 18.98 | 18.06 | 18.65 | 14.88 |
| % Function (F) | 2.41 | 3.38 | 3.58 | 2.58 | 2.98 |
| % Ground (G) | 1.41 | 0.03 | 0.56 | 0.37 | 0.57 |
| % Parallel Scene (H) | 13.06 | 6.02 | 6.3 | 6.15 | 7.54 |
| % Linker (L) | 0.68 | 2.19 | 2.66 | 2.57 | 2.49 |
| % Connector (N) | 0.00 | 1.26 | 0.93 | 0.84 | 0.65 |
| % Process (P) | 11.33 | 7.1 | 7.51 | 6.91 | 7.03 |
| % Relator (R) | 1.81 | 8.58 | 8.09 | 9.6 | 7.54 |
| % State (S) | 5.10 | 1.62 | 2.1 | 1.88 | 3.34 |
| % Punctuation (U) | 14.67 | 11.28 | 10.55 | 11.16 | 13.15 |

The average sentence length in the Turkish dataset is comparably shorter than that of the other languages. This is due to the morphological complexity of the language although the semantic complexity and relations are included but encoded differently. Turkish sentences in the dataset are generally syntactically simple and involve one predicate and one subject. The average length of the sentences is 6.19 (# of tokens/ # of sentences), the average number of edges is 9.95 (# of edges/ # of sentences), and finally, the average number of IMPLICIT edges in the dataset is 0.89 (# of IMPLICIT edges/ # of sentences).

## 6. Experiments and results

We utilised the annotated dataset to train and test the semantic parser model that is used to obtain partially correct annotations. We performed two experiments, where we only used the dataset for testing purposes in the zero-shot learning framework in the first one (zero-shot learning), and we used some part of the dataset for training and the rest for testing purposes in the second experiment (few-shot learning).

**Table 3.** The number of sentences in each UCCA-annotated dataset provided by SemEval 2019 (Hershcovich *et al.* 2019b)

|            | English-Wiki | English-20K | German-20K | French-20K |
|------------|--------------|-------------|------------|------------|
| Train      | 4,113        | 0           | 5,211      | 15         |
| Validation | 514          | 0           | 651        | 238        |
| Test       | 515          | 492         | 652        | 239        |

### 6.1 Experimental details

#### 6.1.1 Datasets for external parser

For training the external semantic parser (Bölücü *et al.,* 2023), we used the combination of English, German, and French UCCA datasets released in SemEval-2019 (Hershcovich *et al.* 2019b). The details of the datasets are given in Table 3.

#### 6.1.2 Training details

All syntactic embedding features (i.e., PoS tags, dependency tags, entity types, and entity iob types) are extracted from Stanza library (Qi *et al.* 2020)[r] that includes a model trained on the IMST Turkish Dependency Treebank (Sulubacak *et al.,* 2016; Sulubacak and Eryiğit, 2018; Şahin and Adalı, 2018).

In zero-shot learning, we merged the train sets of the English, German, and French UCCA datasets to train the parser model and used the merged validation sets to fine-tune the parameters of the model. We tested the trained model on our annotated dataset in Turkish.

In few-shot learning, we used 5-fold cross-validation by adding 320 sentences in Turkish to the merged training set in other languages and using 80 sentences in Turkish as the test set. We report the average of the scores obtained from each fold and the standard deviation of the scores to analyse the variance between the folds.

#### 6.1.3 Evaluation metrics

We followed the official evaluation metrics used in Semeval-2019 (Hershcovich *et al.* 2019b) which are Precision, Recall, and $F_1$ score. The metrics measure a matching score between the graphs $G(V, E, l)$ with a set of nodes $V$, edges $E$ between nodes, and the labels $l$ of the edges. The outputs of the model and the gold/annotated graphs are denoted by $G_o = (V_o, E_o, l_o)$ and $G_g = (V_g, E_g, l_g)$, respectively. Labelled precision and recall metrics are calculated by dividing the number of matched edges in $G_o$ and $G_g$ with corresponding labels by the total number of edges in $E_o$ and $E_g$, respectively. Unlabelled precision and recall metrics are calculated by considering only the number of matched edges in $G_o$ and $G_g$ without taking into account the edge labels.

Additionally, we measured the statistical significance of the macro $F_1$ score with an approximate randomisation test (Chinchor, 1992) with $99,999$ iterations and a significance level of $\alpha = 0.05$ for few-shot and zero-shot learning. For the significance testing, we used the outputs yielding the highest $3^{rd}$-best score obtained from the folds.

### 6.2 Results

The experimental results of zero-shot and few-shot learning are given in Table 4. As we can see, the model struggles to predict the remote edges, including unlabelled edges, which is considered an easier task than predicting labelled edges in zero-shot learning. However, when a limited

---

[r]https://stanfordnlp.github.io/stanza/index.html

**Table 4.** F-1 results obtained from zero-shot and few-shot learning on the Turkish UCCA dataset. Avg is the macro average of F1 metric. ↑ means a statistically significant improvement over the zero-shot learning.

| | Labelled | | | Unlabelled | | |
|---|---|---|---|---|---|---|
| | Primary | Remote | Avg | Primary | Remote | Avg |
| Zero-shot | 58.2±0.71 | 0.0 | 57.9±0.63 | 85.9±1.44 | 0.0 | 85.7±1.25 |
| Few-shot | 74.8±0.96 ↑ | 21.2±0.84 ↑ | 73.5±0.60 ↑ | 89.5±1.04 ↑ | 23.4±1.06 ↑ | 88.2±0.96 ↑ |
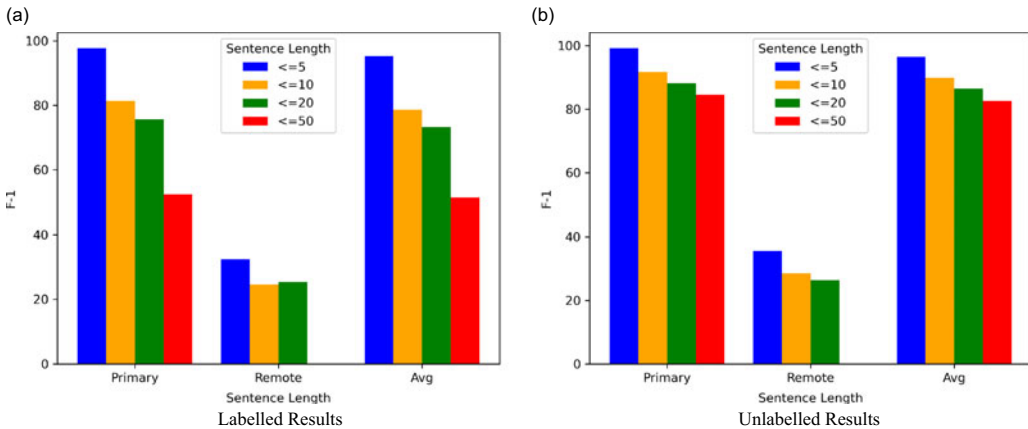


**Figure 8.** Results obtained from few-shot learning according to their sentence length

Turkish dataset is used to train the parser model in the few-shot learning setting, the model shows effectiveness in predicting remote edges. Although the amount of Turkish training data is small compared to the datasets in the other languages (English, German, and French) used for training in few-shot learning, the model shows a remarkable improvement compared to the results obtained in the zero-shot setting. In addition to better results, few-shot learning proves to make the parser model more stable (with lower variance) across folds compared to zero-shot learning. This shows that adding Turkish data improves the generalisability of the parser model. Finally, although the performance of the unlabelled remote edge prediction is higher than the labelled one, the improvement is not significant, showing the challenge of `Remote` edge prediction in Turkish.

### 6.3 Error analysis

#### 6.3.1 Sentence length

In Figure 8, we present the labelled and unlabelled results obtained by the parser trained in the few-shot learning setting for different sentence lengths. It can be seen that the model performs better in predicting shorter sentences which make up the majority of the UCCA dataset compared to the other datasets in English, German, and French. In addition, shorter Turkish sentences are mostly simple sentences consisting of a `Participant` (A), a `Process` (P) or `State` (S), and a `Punctuation` (U) such as "⟨*(Sen)*⟩ *Anlattın.*" (in English, *"You told."*) and "⟨*(Ben)*⟩ *Biliyorum.*" (in English, *"I know."*), making it easier for the parser to predict UCCA categories for shorter sentences. These UCCA labels are also the most frequent categories in the datasets (see Table 2). However, the performance of the parser for longer sentences is superior in unlabelled prediction, showing that predicting UCCA categories is a challenge in semantic parsing. In other words, even
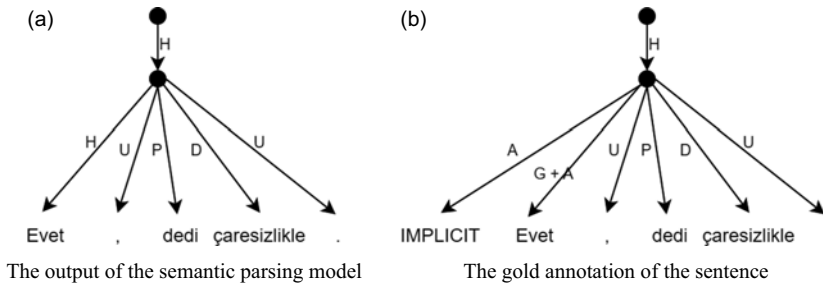
(a) The output of the semantic parsing model

(b) The gold annotation of the sentence

**Figure 9.** The semantic parse tree obtained from the semantic parsing model and the gold annotation obtained from the manual annotation of the sentence *"(O) Evet, dedi çaresizlikle."* (in English, *"S/he said yes with desperation."*). Category abbreviations: **H**: Parallel Scene, **D**: Adverbial, **G**: Ground, **U**: Punctuation, **P**: Process, **A**: Participant

if we parse a sentence correctly, we may not be able to predict the correct UCCA category of the tokens. Since the labelled results of shorter sentences are remarkably high, we could not see a big gap between labelled and unlabelled results in short sentences. However, it becomes noticeable in longer sentences due to the lower performance of the UCCA parser on these sentences.

The results for different sentence lengths presented in Figure 8 show the reason for the improvement in the unlabelled results. The prediction of unlabelled UCCA representation of longer sentences is more accurate than for the labelled ones and has more impact on the improvement of the unlabelled results given in Table 4.

### 6.3.2 Error types

We identified three categories of prediction errors made by the semantic parser trained with few-shot learning:

1. Linkage error: A linkage error is a type of error that occurs in Inter-Scene relations, including `Parallel Scene`, `E-Scene`, `A-Scene`, `C-Scene`, and `D-Scene`. This error can be divided into two types of errors:

   (a) Overgeneration: This type of error occurs when the parser generates scenes that are too large by combining scenes that should be separate. For example, in the sentence *"Evet, dedi çaresizlikle."* (in English, "S/he said yes with desperation."), the parser generates two `Scenes` ⟨Evet⟩$_H$ and ⟨dedi çaresizlikle⟩$_H$, although "Evet" is the `Ground` (G) of the main `Scene` (see Figure 9).

   (b) Undergeneration: In contrast to overgeneration, undergeneration error occurs when the parser fails to create scenes for elements that should be grouped together. It can be seen as mislabelling/merging separate `Parallel Scenes`. For example, the sentence *"Kurtulup buraya gelmeyi başardım."* (in English, *"I managed to escape and come here."*) should be labelled as ⟨Kurtulup⟩$_H$ ⟨buraya gelmeyi başardım⟩$_H$, whereas the output of the parser is ⟨Kurtulup buraya gelmeyi başardım⟩$_H$ where "Kurtulup" is labelled as the `Process` (P) of the main `Scene` (see Figure 10). Another example is the prediction of the nested `Parallel Scenes`. In the sentence *"Kaçıp kurtulmak istedin."* (in English, "You wanted to escape and get away."), "Kaçıp" (to escape) and "kurtulmak" (to get away) are the `A-Scenes` of the main `Scene`, but they are parsed as `Process` (P) of the main `Scene` by the parser (see Figure 11).

2. Mislabelling: Mislabelling occurs when the parser correctly parses the tokens but assigns an incorrect UCCA category. This error is particularly observed between `Center` (C) and
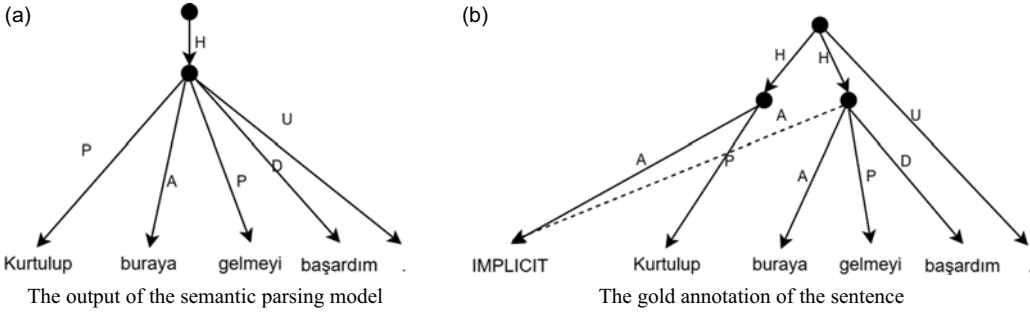
**Figure 10.** The semantic parse tree obtained from the semantic parsing model and the gold annotation obtained from the manual annotation of the sentence, *"(Ben) Kurtulup buraya gelmeyi başardım."* (in English, *"I managed to escape and come here."*). Category abbreviations: **H**: Parallel Scene, **D**: Adverbial, **U**: Punctuation, **P**: Process, **A**: Participant
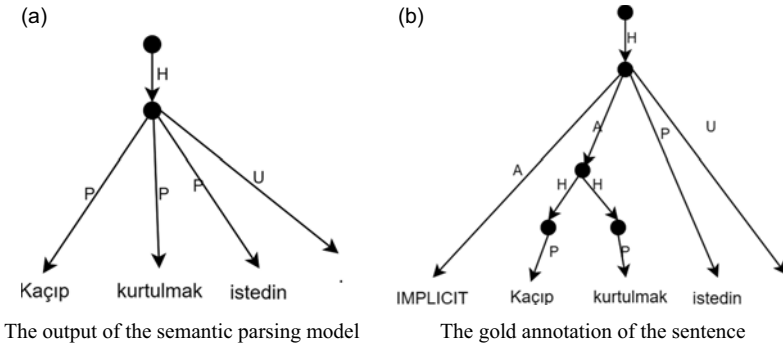


**Figure 11.** The semantic parse tree obtained from the semantic parsing model and the gold annotation obtained from the manual annotation of the sentence *"(Sen) Kaçıp kurtulmak istedin."* (in English, *"You wanted to escape and get away."*). Category abbreviations: **H**: Parallel Scene, **U**: Punctuation, **P**: Process, **A**: Participant

Participant (A), State (S); between Elaborator (E) and Adverbial (D), State (S), and Participant (A); and between Relator (R) and Ground (G), Parallel Scene (H), and Linker (L). For example, "başka" (another) should be labelled as Elaborator (E) in the sentence *"Onu elinden kaçırmış bir başka erkeğe kaptırmıştı."* (in English, *"S/he missed her/him s/he had lost her/him to another man."*). The parser correctly parses the sentence but labels "başka" as Center (C) while its correct label is Elaborator (E) (see Figure 12).

3. Attachment error: Attachment error occurs when the Relator is not properly attached to the Center (C) (or the Scene) where it should be. For instance, in the sentence *"Geldik! diye bağırdı Kerem."* (in English, *"Kerem shouted that we had arrived."*), there is an attachment error where the Relator "diye" (that) is not correctly attached to the associated Scene "Geldik" (we arrived) (see Figure 13).

### 6.4 Discussion

In the literature, semantic representations are widely employed in NLP and NLU applications that require comprehension of text (Liu *et al.,* 2015; Issa *et al.,* 2018; Kapanipathi *et al.,* 2021), making annotated datasets crucial resources. However, the annotation of datasets is a labour-intensive and expensive process, demanding expertise and strict adherence to annotation guidelines.
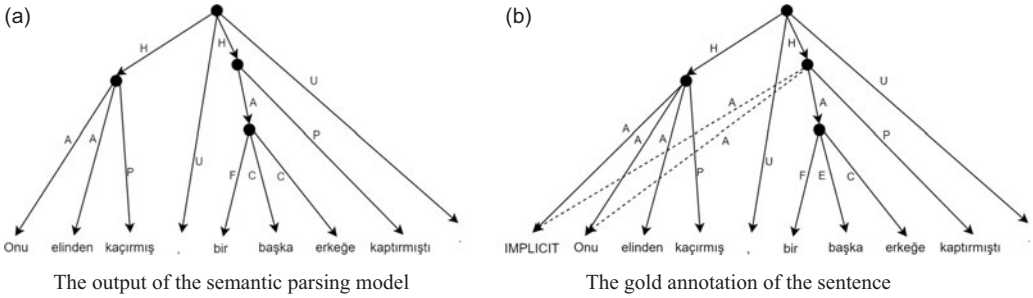
**Figure 12.** The semantic parse tree obtained from the semantic parsing model and the gold annotation obtained from the manual annotation of the sentence, *"(O) Onu elinden kaçırmış, bir başka erkeğe kaptırmıştı."* (in English, *"S/he missed her/him, s/he had lost her/him to another man."*). Category abbreviations: **H**: Parallel Scene, **D**: Adverbial, **F**: Function, **U**: Punctuation, **P**: Process, **E**: Elaborator, **C**: Center, **A**: Participant
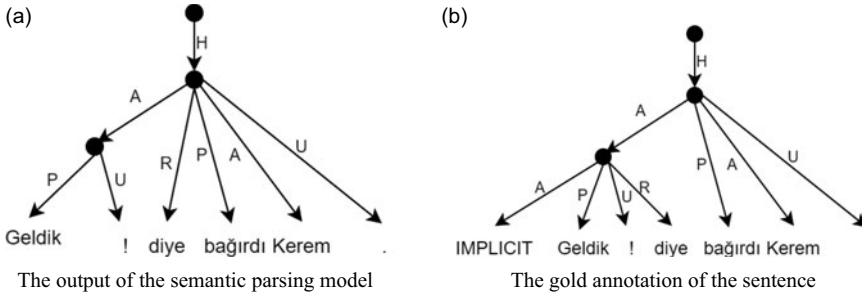


**Figure 13.** The semantic parse tree obtained from the semantic parsing model and the gold annotation obtained from the manual annotation of the sentence *"Geldik! diye bağırdı Kerem."* (in English, *"Kerem shouted that we had arrived."*). Category abbreviations: **H**: Parallel Scene, **R**: Relator, **U**: Punctuation, **P**: Process, **A**: Participant

UCCA facilitates knowledge transfer across languages. Bölücü *et al.* (2023) show the efficacy of zero-few and few-shot learning, demonstrating that a semantic parser trained in one language can perform reasonably well in another, even when linguistic structures differ. Motivated by this, we propose a pipeline annotation procedure that employs an external semantic parser trained on UCCA datasets from various languages to generate a partially annotated dataset in Turkish. Despite the linguistic differences, we successfully transfer cross-lingual features, exemplified by achieving an F-1 unlabelled score of 85.9% for Turkish in the zero-shot setting. The substantial reduction in total annotation time, approximately 1/3rd of the expected duration, justifies the inclusion of an external semantic parser in the annotation process.

In the context of Turkish UCCA annotation, previous work (Bölücü and Can, 2022b) explored a similar approach on a smaller corpus, while our study introduces a larger dataset of 400 sentences. The guideline provided in the prior study covered only a subset of rules, insufficient for fully annotating a Turkish dataset within the UCCA framework. In contrast, our more comprehensive rules (Section 4) encompass all possible syntactic rules necessary for UCCA annotation in Turkish that involve both language-specific closed-class words and more generic rules applicable to other syntactically similar languages. Therefore, the present study significantly expands upon previous research, offering a detailed analysis of results and insights applicable to researchers working on semantic annotation in other languages.

A recognised limitation is the involvement of an external semantic parser and the need to analyse the outputs of the parser (to obtain a partially annotated dataset), which complicates a fully automatic annotation process. For researchers venturing into new languages, it may be required to

define rules that may deviate from the English guideline. However, extending the training to new languages can potentially streamline the annotation time for building diverse language models, and our guideline will be a potential guideline for use in such other language annotations.

## 7. Conclusion

In this study, we presented the Turkish UCCA dataset with 400 sentences obtained from the METU-Sabanci Turkish Treebank. The annotation was performed in a semi-automatic framework in which we used an external semantic parser in zero-shot learning trained on UCCA datasets in other languages and tested with the raw Turkish dataset to obtain a partially annotated dataset. Then, we analysed the discrepancies between the annotated sentences and the English guideline to define new rules in line with Turkish grammar in addition to the ones that are already defined in the actual UCCA guideline. In doing so, we either utilised the current specifications by describing how each linguistic construction should be annotated to ensure consistent annotation based on the original guideline, or we defined new rules that cover the syntactic rules peculiar to Turkish.

We believe that this corpus will be a crucial resource for advancing the state of the art in semantic parsing in Turkish, particularly in Turkish UCCA parsing. This will also be useful for other NLP tasks that require semantic information, such as question answering, text summarisation, and machine translation. Furthermore, the provided Turkish guideline, incorporating new rules specific to Turkish grammar alongside the English guideline, will be beneficial for annotating datasets in other languages. In the future, we plan to annotate a new version of the Turkish UCCA dataset, this time at the morphological level.

## References

**Abend O. and Rappoport A.** (2013a). UCCA: A semantics-based grammatical annotation scheme. In *Proceedings of the 10th International Conference on Computational Semantics (IWCS 2013)–Long Papers*, pp. 1–12.

**Abend O. and Rappoport A.** (2013b). Universal conceptual cognitive annotation (UCCA). In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 228–238.

**Abend O. and Rappoport A.** (2017). The state of the art in semantic representation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 77–89.

**Abend O., Schneider N., Dvir D., Prange J. and Rappoport A.** (2020). UCCA's Foundational Layer: Annotation Guidelines v2.1. arXiv preprint arXiv:2012.15810.

**Ameer I., Bölücü N., Sidorov G. and Can B.** (2023). Emotion Classification in Texts over Graph Neural Networks: Semantic Representation is better than Syntactic. IEEE Access, **11**, 56921–56934.

**Arviv O., Cui R. and Hershcovich D.** (2020). HUJI-KU at MRP 2020: Two transition-based neural parsers. In *Proceedings of the CoNLL. 2020 Shared Task: Cross-Framework Meaning Representation Parsing*, pp. 73–82.

**Atalay N. B., Oflazer K. and Say B.** (2003). The annotation process in the Turkish treebank. In *Proceedings of 4th International Workshop on Linguistically Interpreted Corpora (LINC-03) at EACL 2003*.

**Azin Z. and Eryiğit G.** (2019). Towards Turkish abstract meaning representation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, pp. 43–47.

**Bai H. and Zhao H.** (2019). SJTU at MRP 2019: A transition-based multi-task parser for cross-framework meaning representation parsing. In *Proceedings of the Shared Task on Cross-Framework Meaning Representation Parsing at the 2019 Conference on Natural Language Learning*, pp. 86–94.

**Banarescu L., Bonial C., Cai S., Georgescu M., Griffitt K., Hermjakob U., Knight K., Koehn P., Palmer M. and Schneider N.** (2013). Abstract meaning representation for sembanking. In *Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse*, pp. 178–186.

**Bölücü N. and Can B.** (2022a). Analysing syntactic and semantic features in pre-trained language models in a fully unsupervised setting. In *Proceedings of the 19th International Conference on Natural Language Processing (ICON)*, pp. 19–31.

**Bölücü N. and Can B.** (2022b). Turkish universal conceptual cognitive annotation. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pp. 89–99.

**Bölücü N., Can B. and Artuner H.** (2023). A Siamese neural network for learning semantically-informed sentence embeddings. *Expert Systems with Applications* **214**, 119103.

**Cao J.**, **Zhang Y.**, **Youssef A. and Srikumar V.** (2019). Amazon at MRP 2019: Parsing meaning representations with lexical and phrasal anchoring. In *Proceedings of the Shared Task on Cross-Framework Meaning Representation Parsing at the 2019 Conference on Natural Language Learning*, pp. 138–148.

**Chappelier J.-C. and Rajman M.** (1998). A generalized CYK algorithm for parsing stochastic CFG. In *Proc. of 1st Workshop on Tabulation in Parsing and Deduction (TAPD'98), number CONF*, pp. 133–137.

**Che W.**, **Dou L.**, **Xu Y.**, **Wang Y.**, **Liu Y. and Liu T.** (2019). HIT-SCIR at MRP 2019: A unified pipeline for meaning representation parsing via efficient training and effective encoding. In *Proceedings of the Shared Task on Cross-Framework Meaning Representation Parsing at the 2019 Conference on Natural Language Learning*, pp. 76–85.

**Chinchor N.** (1992). The statistical significance of the muc-4 results. In *Proceedings of the 4th Conference on Message Understanding*, pp. 30–50.

**Cohen J.** (1960). A coefficient of agreement for nominal scales. *Educational and Psychological Measurement* **20**(1), 37–46.

**Çöltekin Ç.**, **Doğruöz A. S. and Çetinoğlu Ö.** (2023). Resources for Turkish natural language processing: a critical survey. *Language Resources and Evaluation* **57**(1), 449–488.

**Devlin J.**, **Chang M.-W.**, **Lee K. and Toutanova K.** (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 4171–4186.

**Dixon R.** (2010a). *Basic Linguistic Theory*. Oxford University Press.

**Dixon R. M.** (2005). *A Semantic Approach to English Grammar*. Oxford University Press.

**Dixon R. M.** (2010b). *Basic Linguistic Theory*. Volume 2: Grammatical topics. Oxford University Press.

**Dixon R. M.** (2012). *Basic Linguistic Theory*. Volume 3: Further Grammatical Topics. Oxford: Oxford University Press.

**Donatelli L.**, **Fowlie M.**, **Groschwitz J.**, **Koller A.**, **Lindemann M.**, **Mina M. and Weißenhorn P.** (2019). Saarland at MRP 2019: Compositional parsing across all graphbanks. In *Proceedings of the Shared Task on Cross-Framework Meaning Representation Parsing at the 2019 Conference on Natural Language Learning*, pp. 66–75.

**Dou L.**, **Feng Y.**, **Ji Y.**, **Che W. and Liu T.** (2020). HIT-SCIR at MRP 2020: Transition-based parser and iterative inference parser. In *Proceedings of the CoNLL. 2020 Shared Task: Cross-Framework Meaning Representation Parsing*, pp. 65–72.

**Droganova K.**, **Kutuzov A.**, **Mediankin N. and Zeman D.** (2019). ÚFAL-Oslo at MRP 2019: Garage sale semantic parsing. In *Proceedings of the Shared Task on Cross-Framework Meaning Representation Parsing at the 2019 Conference on Natural Language Learning*, pp. 158–165.

**Giordano B.**, **Lopez C. and Le I.** (2023). MR4AP: Meaning representation for application purposes. In *Proceedings of the 4th International Workshop on Designing Meaning Representations*, pp. 110–121.

**Göçmen E.**, **Sehitoglu O. T. and Bozsahin C.** (1995). An outline of Turkish syntax. *Ms. Department of Computer Engineering*, 1–36.

**Göksel A. and Kerslake C.** (2004). *Turkish: A Comprehensive Grammar*. Routledge.

**Hershcovich D.**, **Abend O. and Rappoport A.** (2017). A transition-based directed acyclic graph parser for UCCA. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1127–1138.

**Hershcovich D.**, **Abend O. and Rappoport A.** (2018). Multitask parsing across semantic representations. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 373–385.

**Hershcovich D.**, **Abend O. and Rappoport A.** (2019a). Content differences in syntactic and semantic representation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 478–488.

**Hershcovich D.**, **Aizenbud Z.**, **Choshen L.**, **Sulem E.**, **Rappoport A. and Abend O.** (2019b). SemEval-2019 task 1: Cross-lingual semantic parsing with UCCA. In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pp. 1–10.

**Hewitt J. and Manning C. D.** (2019). A structural probe for finding syntax in word representations. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 4129–4138.

**Issa F.**, **Damonte M.**, **Cohen S. B.**, **Yan X. and Chang Y.** (2018). Abstract meaning representation for paraphrase detection. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pp. 442–452.

**Ivanova A.**, **Oepen S.**, **Øvrelid L. and Flickinger D.** (2012). Who did what to whom? A contrastive study of syntacto-semantic dependencies. In *Proceedings of the sixth linguistic annotation workshop*, pp. 2–11.

**Jiang W.**, **Li Z.**, **Zhang Y. and Zhang M.** (2019). HLT@SUDA at SemEval 2019 task 1: UCCA graph parsing as constituent tree parsing. In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pp. 11–15.

**Kapanipathi P.**, **Abdelaziz I.**, **Ravishankar S.**, **Roukos S.**, **Gray A.**, **Astudillo R. F.**, **Chang M.**, **Cornelio C.**, **Dana S.**, **Fokoue-Nkoutche A. and et al.** (2021). Leveraging abstract meaning representation for knowledge base question answering. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pp. 3884–3894, *2021*

**Kitaev N. and Klein D.** (2018a). Constituency parsing with a self-attentive encoder. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 2676–2686.

**Kitaev N. and Klein D.** (2018b). Multilingual constituency parsing with self-attention and pre-training. *CoRR*, abs/1812.11760.

**Koreeda Y.**, **Morio G.**, **Morishita T.**, **Ozaki H. and Yanai K.** (2019). Hitachi at MRP 2019: Unified encoder-to-biaffine network for cross-framework meaning representation parsing. In *Proceedings of the Shared Task on Cross-Framework Meaning Representation Parsing at the 2019 Conference on Natural Language Learning*, pp. 114–126.

**Kuhlmann M. and Oepen S.** (2016). Towards a catalogue of linguistic graph banks. *Computational Linguistics* **42**(4), 819–827.

**Lai S.**, **Lo C. H.**, **Leung K. S. and Leung Y.** (2019). CHUK at MRP 2019: Transition-based parser with cross-framework variable-arity resolve action. In *Proceedings of the Shared Task on Cross-Framework Meaning Representation Parsing at the 2019 Conference on Natural Language Learning*, pp. 104–113.

**Langacker R. W.** (2007). Cognitive grammar. In D. Geeraets & H. Cuyckens (Eds.), *The Oxford Handbook of Cognitive Linguistics* (pp. 421–462). Oxford: Oxford University Press.

**Lewis G.** (1967). *Turkish Grammar.* Clarendon Press.

**Li Z.**, **Zhao H.**, **Zhang Z.**, **Wang R.**, **Utiyama M. and Sumita E.** (2019). SJTU-NICT at MRP 2019: Multi-task learning for end-to-end uniform semantic graph parsing. In *Proceedings of the Shared Task on Cross-Framework Meaning Representation Parsing at the 2019 Conference on Natural Language Learning*, pp. 45–54.

**Liao K.**, **Lebanoff L. and Liu F.** (2018). Abstract meaning representation for multi-document summarization, 1178–1190.

**Lindemann M.**, **Groschwitz J. and Koller A.** (2019). Compositional semantic parsing across graphbanks. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 4576–4585.

**Liu F.**, **Flanigan J.**, **Thomson S.**, **Sadeh N. and Smith N. A.** (2015). Toward abstractive summarization using semantic representations. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 1077–1086.

**Liu Y.**, **Ott M.**, **Goyal N.**, **Du J.**, **Joshi M.**, **Chen D.**, **Levy O.**, **Lewis M.**, **Zettlemoyer L. and Stoyanov V.** (2019). RoBERTa: A robustly optimized BERT pretraining approach, arXiv preprint arXiv:1907.

**Lyu W.**, **Huang S.**, **Khan A. R.**, **Zhang S.**, **Sun W. and Xu J.** (2019). CUNY-PKU parser at semEval-2019 task 1: Cross-lingual semantic parsing with UCCA. In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pp. 92–96.

**Na S.-H.**, **Min J.**, **Park K.**, **Shin J.-H. and Kim Y.-G.** (2019). JBNU at MRP 2019: Multi-level biaffine attention for semantic dependency parsing. In *Proceedings of the Shared Task on Cross-Framework Meaning Representation Parsing at the 2019 Conference on Natural Language Learning*, pp. 95–103.

**Naseem T.**, **Ravishankar S.**, **Mihindukulasooriya N.**, **Abdelaziz I.**, **Lee Y.-S.**, **Kapanipathi P.**, **Roukos S.**, **Gliozzo A. and Gray A.** (2021). A semantics-aware transformer model of relation linking for knowledge base question answering. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pp. 256–262.

**Nguyen L. H.**, **Pham V. H.**, **Dinh D.**, **Ahmad A. and et al.** (2021). Improving neural machine translation with AMR semantic graphs. *Mathematical Problems in Engineering* **2021**, pp. 1–12.

**Nivre J.**, **de Marneffe M.**, **Ginter F.**, **Hajic J.**, **Manning C. D.**, **Pyysalo S.**, **Schuster S.**, **Tyers F. M. and Zeman D.** (2020). Universal dependencies v2: An evergrowing multilingual treebank collection. *CoRR*, abs/2004.10643.

**Nivre J.**, **De Marneffe M.-C.**, **Ginter F.**, **Goldberg Y.**, **Hajic J.**, **Manning C. D.**, **McDonald R.**, **Petrov S.**, **Pyysalo S.**, **Silveira N. and et al.** (2016). Universal dependencies v1: A multilingual treebank collection. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pp. 1659–1666.

**Oepen S.**, **Abend O.**, **Abzianidze L.**, **Bos J.**, **Hajic J.**, **Hershcovich D.**, **Li B.**, **O'Gorman T.**, **Xue N. and Zeman D.** (2020). MRP 2020: The second shared task on cross-framework and cross-lingual meaning representation parsing. In *Proceedings of the CoNLL. 2020 Shared Task: Cross-Framework Meaning Representation Parsing*, pp. 1–22.

**Oepen S.**, **Abend O.**, **Hajic J.**, **Hershcovich D.**, **Kuhlmann M.**, **O'Gorman T.**, **Xue N.**, **Chun J.**, **Straka M. and Urešová Z.** (2019). MRP 2019: Cross-framework meaning representation parsing. In *Proceedings of the Shared Task on Cross-Framework Meaning Representation Parsing at the 2019 Conference on Natural Language Learning*, pp. 1–27.

**Oepen S. and Flickinger D.** (2019). The ERG at MRP 2019: Radically compositional semantic dependencies. In *Proceedings of the Shared Task on Cross-Framework Meaning Representation Parsing at the 2019 Conference on Natural Language Learning*, pp. 40–44.

**Oepen S.**, **Kuhlmann M.**, **Miyao Y.**, **Zeman D.**, **Cinková S.**, **Flickinger D.**, **Hajic J.**, **Ivanova A. and Uresova Z.** (2016). Towards comparability of linguistic graph banks for semantic parsing. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pp. 3991–3995.

**Oepen S. and Lønning J. T.** (2006). Discriminant-based MRS banking. In *LREC*, pp. 1250–1255.

**Oflazer K.** (2014). Turkish and its challenges for language processing. *Language Resources and Evaluation* **48**(4), 639–653.

**Oflazer K.**, **Say B.**, **Hakkani-Tür D. Z. and Tür G.** (2003). *Building a Turkish Treebank*. In A. Abeillé (Ed.), *Treebanks: Building and using Syntactically Annotated Corpora* (pp. 261–277). Kluwer Academic Publishers.

**Oral E.**, **Acar A. and Eryiğit G.** (2024). Abstract meaning representation of Turkish. *Natural Language Engineering*, **30**(1), 171–200. doi:10.1017/S1351324922000183.

**Ozaki H.**, **Morio G.**, **Koreeda Y.**, **Morishita T. and Miyoshi T.** (2020). Hitachi at MRP 2020: Text-to-graph-notation transducer. In *Proceedings of the CoNLL. 2020 Shared Task: Cross-Framework Meaning Representation Parsing*, pp. 40–52.

**Pütz T. and Glocker K.** (2019). Tüpa at SemEval-2019 task1: (Almost) feature-free semantic parsing. In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pp. 113–118.

**Qi P., Zhang Y., Zhang Y., Bolton J. and Manning C. D.** (2020). Stanza: a Python natural language processing toolkit for many human languages. arXiv preprint arXiv:2003.07082.

**Şahin G. G. and Adalı E.** (2018). Annotation of semantic roles for the Turkish proposition bank. *Language Resources and Evaluation* **52**(3), 673–706.

**Sak H., Güngör T. and Saraçlar M.** (2011). Resources for Turkish morphological processing. *Language Resources and Evaluation* **45**(2), 249–261.

**Slobodkin A., Choshen L. and Abend O.** (2021). Semantics-aware attention improves neural machine translation, arXiv preprint arXiv:2110.06920.

**Song L., Gildea D., Zhang Y., Wang Z. and Su J.** (2019). Semantic neural machine translation using AMR. *Transactions of the Association for Computational Linguistics* **7**, 19–31.

**Sulem E., Abend O. and Rappoport A.** (2015). Conceptual annotations preserve structure across translations: A French-English case study. In *Proceedings of the 1st Workshop on Semantics-Driven Statistical Machine Translation,* pp. 11–22.

**Sulem E., Abend O. and Rappoport A.** (2018). Simple and effective text simplification using semantic and neural methods. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 162–173.

**Sulem E., Abend O. and Rappoport A.** (2020). Semantic structural decomposition for neural machine translation. In *Proceedings of the Ninth Joint Conference on Lexical and Computational Semantics*, pp. 50–57.

**Sulubacak U. and Eryiğit G.** (2018). Implementing universal dependency, morphology, and multiword expression annotation standards for Turkish language processing. *Turkish Journal of Electrical Engineering & Computer Sciences* **26**(3), 1662–1672.

**Sulubacak U., Eryiğit G. and Pamay T.** (2016). IMST: A revisited Turkish dependency treebank. In *1st International Conference on Turkic Computational Linguistics*, EGE UNIVERSITY PRESS, pp. 1–6.

**Türk U., Atmaca F., Özateş Ş.B., Berk G., Bedir S. T., Köksal A., Başaran B.Ö., Güngör T. and Özgür A.** (2022). Resources for Turkish dependency parsing: introducing the BOUN treebank and the BoAT annotation tool. *Language Resources and Evaluation*, **56**, 259–307.

**Van Gysel J. E. L., Vigus M., Chun J., Lai K., Moeller S., Yao J., O'Gorman T., Cowell A., Croft W., Huang C.-R., Hajič J., Martin J. H., Oepen S., Palmer M., Pustejovsky J., Vallejos R., Xue N.** (2021). Designing a uniform meaning representation for natural language processing. *KI-Künstliche Intelligenz* **35**(3-4), 343–360.

**Vural A. G., Cambazoglu B. B. and Karagoz P.** (2014). Sentiment-focused web crawling. *ACM Transactions on the Web (TWEB)* **8**(4), 1–21.

**Xu W., Zhang H., Cai D. and Lam W.** (2021). Dynamic semantic graph construction and reasoning for explainable multi-hop science question answering, 1044–1056.

**Xue N., Croft W., Hajič J., Huang C.-R., Oepen S., Palmer M., Pustejovsky J., Abend O., Aroonmanakun W. and Bender E., et al.** (2020). *The First International Workshop on Designing Meaning Representations (DMR)*

**Yu D. and Sagae K.** (2019). UC Davis at Semeval-2019 Task 1: DAG semantic parsing with attention-based decoder. In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pp. 119–124.

**Zeyrek D. and Başıbüyük K.** (2019). TCL-A lexicon of Turkish discourse connectives. In *Proceedings of the First International Workshop on Designing Meaning Representations*, pp. 73–81.

**Zeyrek D. and Webber B.** (2008). A discourse resource for Turkish: Annotating discourse connectives in the METU corpus. In *Proceedings of the 6th workshop on Asian language resources*.

**Zhang X., Zhao H., Zhang K. and Zhang Y.** (2020). SEMA: Text simplification evaluation through semantic alignment. In *Proceedings of the 6th Workshop on Natural Language Processing Techniques for Educational Applications*, pp. 121–128.

**Zhang Y., Jiang W., Xia Q., Cao J., Wang R., Li Z. and Zhang M.** (2019). SUDA-Alibaba at MRP 2019: Graph-based models with BERT. In *Proceedings of the Shared Task on Cross-Framework Meaning Representation Parsing at the 2019 Conference on Natural Language Learning*, pp. 149–157.

## Appendix.

## A. Turkish UCCA-annotated sentences[s]

Here, we give sample sentences corresponding to the UCCA categories defined in English guideline.

---

[s]Category abbreviations for UCCA annotation: **A**: Participant, **C**: Center, **D**: Adverbial, **E**: Elaborator, **F**: Function, **G**: Ground, **H**: Parallel Scene, **L**: Linker, **N**: Connector, **P**: Process, **R**: Relator, **S**: State, **U**: Punctuation.

A text in the foundational layer of UCCA representation consists of `Scenes`. It may consist of one or more scenes as shown in the following examples.

- Ahmet okula gitti (in English, *"Ahmet went to the school"*) (1 Scene)
- Ahmet eve döndü ve duş aldı (in English, *"Ahmet went back home and took a shower"*) (2 Scene)

*A.1 Categories*

*A.1.1 Scene elements.* Each `Scene` should have a main relation describing a movement or action called `Process` (P) or a temporally persistent state called `State` (S). The other components of a `Scene` are `Participant` (A), which can be one or more, and `Adverbial` (D), which describes the relation in time, location, or ground. Below are Turkish examples of Scene elements.

**Example A.1.** *Ben ⟨bir tutsağım⟩$_S$ (in English, "I am a prisoner")*

**Example A.2.** *Ahmet okula yürüyerek ⟨gitti⟩$_P$ (in English, "Ahmet went to school on foot")*

**Example A.3.** *⟨Elmayı⟩$_A$ alabilir (in English, "He may take the apple")*

**Example A.4.** *⟨Ayşe⟩$_A$ ⟨okulda⟩$_A$ kaldı (in English, "Ayşe stayed at school")*

**Example A.5.** *Ahmet yüzmeye ⟨başladı⟩$_D$ (in English, "He started swimming")*

**Example A.6.** *Soruyu ⟨hızlıca⟩$_D$ cevapladı (in English, "She answered the question quickly")*

**Example A.7.** *Ayşe ⟨sık sık⟩$_D$ spora gider (in English, "Ayşe often goes to the gym")*

*A.1.2 Non-Scene unit elements.* Non-Scene relations do not evoke a `Scene`, which is the main difference with the Scene elements. The main concept in non-Scene units is `Center` (C), and other relations that detail the `Centers`. While `Elaborator` (E) determines the semantic type or quantification of the magnitude of the parent entity that is a `Center`, `Connector` (N) connects entities that have similar features or types. Finally, `Relator` (R) relates an entity to other relations or units that are attached with different aspects.

**Example A.8.** *⟨1996⟩$_C$ ⟨yılı⟩$_E$ (in English, "the year 1966")*

**Example A.9.** *⟨onun⟩$_E$ ⟨eli⟩$_C$ (in English, "her/his hand")*

**Example A.10.** *⟨biraz⟩$_E$ ⟨şeker⟩$_C$ (in English, "some sugar")*

**Example A.11.** *⟨⟨Ben⟩$_C$ ⟨ve⟩$_N$ ⟨⟨(benim)⟩$_{E-IMPLICIT}$ ⟨arkadaşım⟩$_C$⟩$_C$ okula beraber gittik (in English, "I and my friend went to school together")*

**Example A.12.** *Ali [⟨fırının⟩$_C$ ⟨içindeki⟩$_R$] kurabiyeleri] aldı (in English, "Ali took the cookies from the oven")*

*A.1.3 Inter-Scene relations.* The Inter-Scene relations category is composed of `Parallel Scene` (H), `Linker` (L), and `Ground` (G). `Parallel Scene` is a `Scene` that does not take place in the main Scene as a `Participant`, a `Center`, or an `Elaborator`. The `Parallel Scenes` can be linked to other Scenes with `Linker`, which is a relational word between `Parallel Scenes`. `Ground` is a unit that relates units to their speech event, which can be either a speaker or a hearer. The main difference with `Linker` is that it does not relate to Scenes. Linkage is a term used in

Inter-Scene relations in which a `Scene` is a unit in one of `Participant`, `Center`, `Elaborator`, `Adverbial` (described in Section 4), or `Parallel Scene`.

**Example A.13.** ⟨*Eğer*⟩*L* ⟨*okula gidersen*⟩*H* ⟨*Ahmet ile karşılaşırsın*⟩*H* (in English, *"If you go to school, you will meet Ahmet"*)

**Example A.14.** ⟨*Arkadaşını beklerken*⟩*H* ⟨*ayakkabısını boyadı*⟩*H* (in English, *"While waiting for her/his friend, s/he polished her/his shoes"*)

**Example A.15.** ⟨*Sadece kendi istediklerini söyledin*⟩*H* ⟨*çünkü*⟩*L* ⟨*sen de suçlusun*⟩*H* (in English, *"You just said what you wanted because you're guilty too"*)

**Example A.16.** ⟨*İlginçtir*⟩*G* *okumakta zorlanmadı* (in English, *"Interestingly, it wasn't hard to read"*)

**Example A.17.** ⟨*Gördüğünüz gibi*⟩*H* ⟨*gelmediler*⟩*H* (in English, *"They didn't come as you see"*)

**Example A.18.** ⟨*Eski kocası*⟩*A* *her zaman oradadır* (in English, *"Her/his ex husband is always there"*)

**Example A.19.** ⟨*Seni üzmekten*⟩*A* *korkuyorum* (in English, *"I'm afraid to upset you"*)

**Example A.20.** ⟨⟨*Her*⟩*E* ⟨*istediğini*⟩*P*⟩*C* *yerine getiriyordum.* (in English, *"I was doing whatever s/he want"*)

**Example A.21.** *Ürkütücü şeyler* ⟨⟨*bu*E* ⟨*anlattıklarınız*⟩*P*⟩*C* (in English, *"These are the scary things you're talking about"*)

**Example A.22.** ⟨*Dar yollarda koşarak giden*⟩*D* *Kerem'i yakaladım* (in English, *"I caught Kerem running on narrow roads"*)

**Example A.23.** ⟨*Bahçeye giren (köpek)*⟩*E* *köpek kahverengidir* (in English, *"The dog entering the garden is brown"*)

**Example A.24.** ⟨*[Yan daireye] taşınan (Ahmet)*⟩*E* *Ahmet evime geldi* (in English, *"Ahmet who moved to the next flat, came to my house"*)

*A.1.4 Other.* The final category is Other in which the `Function` (F) unit is only a part of the construction.

**Example A.25.** ⟨*Ayy*⟩*F* *sandalyeden düştü* (in English, *"Ouch he fell from the chair"*)

**Example A.26.** *İstanbul* ⟨*'a*⟩*F* *mı gidiyorsun* (in English, *"Your are going to the Erkekler Park"*)

**Example A.27.** *Kerem* ⟨*bir*⟩*F* *an durdu* (in English, *"Kerem stopped for a moment"*)

*A.2 Remote and implicit units*
If an entity is missing and referred from another position in the text, an edge is added for the entity as `IMPLICIT`. If it is not referred to in the text, a new token is created as a `REMOTE` unit.

**Example A.28.** ⟨*(O)*⟩*A−IMPLICIT* *okula gelmedi* (in English, *"He didn't come to school"*)

**Example A.29.** [⟨*(Benim)*⟩*E−IMPLICIT* *Çocukluğum*] *aklıma geldi* (in English, *"I remembered my childhood"*)

**Example A.30.** [*Ali okuldan geldi*]*H* *ve* [*televizyon izledi* ⟨*(Ali)*⟩*A−REMOTE*] (in English, *"Ali came from school and watched television"*)

**Example A.31.** [*Okula yeni kayıt olan* ⟨*(çocuk)*⟩$_{A-REMOTE}$*] çocuk] bugün gelmedi (in English, "The newly enrolled child did not come today")*

**Example A.32.** *Ne [ondan bahsedebildim] ne [yaşadıklarımdan* ⟨*(bahsedebildim)*⟩$_{P-REMOTE}$*] (in English, "I could neither talk about her/his nor talk about my experiences")*