CoBaLD Annotation: the Enrichment of the Enhanced Universal Dependencies with the Semantical Pattern

M. Petrova, A. Ivoylova, A. Tishchenkova
A4 Technology, RSUH, RSUH
Moscow
g-fox-ive@mail.ru, a.m.ivoylova@gmail.com, nas.tishenkova@ya.ru

Abstract

The paper is devoted to the annotation format aimed at morphological, syntactic and especially semantic markup. The format combines the Enhanced UD morphosyntax and the Compreno semantic pattern, enriching the UD annotation with word meanings and labels for semantic relations between words. To adapt the Compreno semantics for the current purpose, we reduced the number of the semantic fields denoting lexical meanings by using hyperonym fields. Moreover, we used a generalized variant of the semantic relations as the original roles possess rather narrow meanings which makes them too numerous. Creating such a format demands the Compreno-to-UD morphosyntax conversion as well, which, in turn, demands solving the asymmetry problem between the models. The asymmetry concerns tokenization, lemmatization, POS-tagging, sets of grammatical features and dependency heads. To overcome this problem, the Compreno-to-UD converter was created. As an application, the work presents a 150,000 token corpus of English news annotated according to the standard.

Keywords: Enhanced Universal Dependencies, semantic annotation, Compreno semantics, CoBaLD

1. Introduction

The purpose of the current work is to enrich the Enhanced Universal Dependencies (E-UD) annotation (Schuster and Manning, 2016) with a semantic pattern and, therefore, to develop a markup format supporting morphological, syntactic and semantic levels.

To achieve it, we have supplemented the UD annotation schema with the simplified version of the Compreno semantics[1] (Anisimovich et al., 2012) and labeled the new format as Compreno-Based Linguistic Data Annotation, or CoBaLD Annotation.

Its pilot version was first applied for annotating news corpus for Russian language, consisting of approximately 400,000 tokens. The applicability of the format was tested during the SEMarkup-2023 Shared Task (Petrova et al., 2023) aimed at creating parsers for the integral morpho-syntactic and semantic annotation. The baseline parser founded on ruBERT-tiny achieved around 90% F1-score for both SCs and DSs, which proves that the standard suits well for neural network parsing and can be useful for practical NLP tasks.

In the current paper, we suggest an advanced version of the given format. Namely, we have switched from the basic UD annotation schema to E-UD which allows one to take the ellipted nodes into account, processes conjunction in a more reasonable way and suggests other optimizations which will be discussed further. Besides, we have switched from the CONLL-based format[2] to the CONLL-Plus format[3] (for detail, see below), and created an English corpus annotated according to the CoBaLD standard, which includes about 150,000 tokens[4].

Further, we will first consider the related works and discuss arguments for choosing the E-UD modification and the Compreno semantic pattern for CoBaLD annotation. After it, we'll describe the semantical pattern of the markup, and the conversion process of the Compreno morphosyntax into the E-UD standard. Then we will demonstrate the CoBaLD annotation itself and present the CoBaLD-annotated English dataset. In conclusion, we will sum up the results and define further perspectives.

2. Related work

The importance of linguistic markup cannot be underestimated as such annotation is used not only for theoretical purposes but for various NLP tasks as well. Linguistic data, especially semantic information, can be used to enrich language model embeddings for NLP tasks such as sentiment analysis (Baly et al., 2017), metaphor detection (Li et al., 2023), or cross-lingual transfer (Ponti et al., 2018), as well as for solving other NLP problems.

Practical applicability of linguistic annotation depends on its simplicity, fullness, accuracy, and to a large extent, its suitability for machine processing, especially for neural network parsing.

The idea of linguistic text annotation is closely

[1] The access to the Compreno data is provided according to the CC BY-NC 4.0 License which allows non-commercial use.

[2] https://universaldependencies.org/format.html

[3] https://universaldependencies.org/ext-format.html

[4] https://github.com/CobaldAnnotation/CobaldEng

LREC-COLING 2024, pages 3422–3432
20-25 May, 2024.

related to corpora creation, so the first annotation standards were developed for this research area.

Morphological markup is a basic annotation level, so the first corpora such as the Brown Corpus were morphologically annotated. As for dependency trees, one of the first annotated corpora was the SUSANNE Corpus (Sampson, 2002), created as early as in 90s. Another popular markup standard would be the Penn Treebank scheme (Marcus et al., 1993). At that time, there was no universal annotation standard, and the markup schemes could vary across different corpora. It could be a minor issue for theoretical research, but when machine learning techniques became popular, the need for the universal standard arose.

(De Marneffe et al., 2006) proposed 'a system for automatically extracting typed dependency parses of English sentences from phrase structure parses', which is known as Stanford Dependencies (SD). This framework was dominantly based on Chomskian syntax views, more exactly, on HPSG (head-driven phrase structure grammars, (Pollard and Sag, 1994)), although it pursued more practical goals.

SD served as a foundation for the famous UD annotation scheme which differs from it in several aspects (Nivre et al., 2017). Its main purpose is to create a universal annotation standard applicable for any language. The strive for universality supposes that content and not functional words should be heads of dependency relations. The UD scheme in its base variant represents directed acyclic graphs with morphological and dependency information[5]. Although its dependency relations are not purely syntactic, there are no categories entirely devoted to semantics.

Being convenient for automatic processing, UD is widely used for practical tasks. Nevertheless, the absence of semantical domain imposes some restrictions on its applicability. Therefore, enrichment with the semantical pattern would provide significant benefits and enlarge the range of the tasks such annotation could be used for.

Currently, there are several markup schemas that suggest semantic annotation.

First, Universal Decompositional Semantics (UDS) project (White et al., 2016) conceived as a natural addition to UD, which presents word senses and semantic roles as simple feature sets. Key benefit is its compatibility with UD, but for practical tasks, it seems easier to use and predict simple token categories instead of feature sets as the task of one tag prediction is a simple multinomial classification task while predicting feature sets with different weights presupposes dealing with multi-label classification, which significantly complicates the problem.

Other popular standards aimed at semantical annotation include Universal Networking Language (UNL) (Uchida and Zhu, 2001), Abstract Meaning Representations (AMR) (Banarescu et al., 2013), and Universal Conceptual Cognitive Annotation (UCCA) (Abend and Rappoport, 2013). Unlike UDS, they do not use decomposition and suggest labels for word meanings and relations directly. However, UNL focuses only on the "core" meanings and does not suggest full semantic description for all meanings and relations, while AMR can hardly be integrated in UD as it is based on different principles of token labeling and may not provide labels for every token as it is in UD. Besides, AMR is not an interlingua and mostly focuses on English.

As for UCCA, its semantic formalism is rather complicated suggesting several category sets: on the one hand, the distinction between the categories is not always obvious, on the other hand, some categories include entities of different nature, which makes its integration problematic as well.

There are also several frameworks containing both morpho-syntactic and semantic annotation, namely, Prague Semantic Depencencies (PSD) (Hajic et al., 2001) and the Compreno model (Anisimovich et al., 2012). PSD consists of morphological, surface syntactic and deep syntactic layers, the latter includes semantic relations (functors) as well, but it does not suggest word meanings.

Compreno, in turn, includes both - semantic relations and word meanings. Moreover, Compreno has other advantages: (1) its semantics suggests a very simple categorial system, containing only two category sets: deep slots (DSs) for semantic relations and semantic classes (SCs) for word meanings; (2) SCs are organized in a form of a thesaurus-like tree, where every level gets all possible DSs for each SC, which provides full description of all possible semantic dependencies including actants, adjuncts, modifiers, and so on; (3) Compreno and UD have similar token labeling principles, so integrating the Compreno semantics into the UD annotation is a feasible task.

However, Compreno has some disadvantages as well. First, it presents morphosyntactic information in the form of parsing trees – not in the markup itself, which makes its usage more inconvenient in comparison with the UD presentation, and, most important, depends on the parser's work. Moreover, the number of the SCs and the DSs is too big due to the detailness of the description which seems excessive for most applicational tasks.

To overcome these problems, we have converted the Compreno morphosyntax to UD (for detail, see part 4) and suggested to use hyperonym SCs and generalized version of the DSs (for detail, see part 3).

[5] https://universaldependencies.org/docs/u/overview /syntax.html

The pilot version of the CoBaLD Annotation used the CONLL-based format. Here, we have switched to the CONLL-Plus format as it allows one to include new categories and preserve compatibility with other datasets. We have added two columns after the standard ten ones: for word meanings and for semantic links with parent nodes (for detail, see part 5).

Moreover, our previous version used the 'basic' UD annotation schema, whereas recent UD modifications, such as Enhanced UD and Surface Universal Dependencies (SUD, (Gerdes et al., 2018)), have more in common in some issues with the Compreno structures than the original UD, so another question was which of the UD-style schemes would better suit for the CoBaLD annotation.

SUD treats copula in the same way Compreno does as both models consider copula to be the core. Nevertheless, all other SUD strategies, including treating functional words as heads, differ significantly from the Compreno formalism.

E-UD, to the contrary, is closer to Compreno, as both models restore the ellipted nodes, or take prepositions into account when specifying dependency labels. For instance, as E-UD restores ellipsis, it does not need the 'orphan' relation which basic UD uses for the dependencies of the ellipted nodes as in Figure 1. Another example is that E-UD does not substitute ellipted nominal heads with their dependents, when, for instance, an ordinal may become 'nsubj' because of an ellipted noun, as in Figure 2 below.

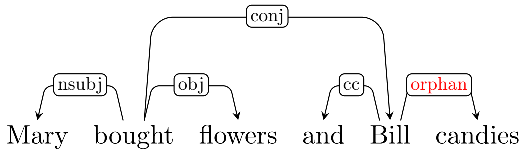


Figure 1: 'orphan' relation in UD: 'Mary bought flowers, and Bill (bought) candies.'

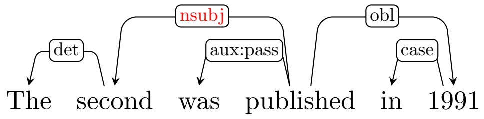


Figure 2: 'nsubj' relation in UD: 'The second (volume) was published in 1991.'

Besides, E-UD allows non-tree links such as reference, and its handling with some syntactic relations, especially with conjunction, seems more logical than in original UD. For these reasons, the E-UD modification was chosen for further development of the format.

3. Enrichment with semantics

Compreno semantics consists of two parts (for more detail, see (Anisimovich et al., 2012; Petrova, 2014; Manicheva et al., 2012)): presentation of word meanings and presentation of relations between words in a sentence. We included both patterns in the annotation. However, we have made some adaptations for the current project.

Important characteristics of the Compreno model are its fullness and detailness. Fullness means that it defines labels for all word meanings and all relations between words, including actants, modifiers, adjuncts, parenthetical expressions, and so on. As far as the detailness is concerned, for word semantics, it means that there are plenty of small semantic fields with a narrow meaning each, thereafter, the number of such fields is rather big. For the relations between words, it means that there are many relations with narrow semantics as well, for example, there are more than 50 relations for different kinds of characteristics: size, speed, weight, colour, smell, evaluation, and alike. Such a detailed description has both advantages and disadvantages, depending on the purposes the datasets can be used for.

Below, we regard the semantic items in more detail and consider the modifications they have undergone in the CoBaLD format while integrating in the E-UD formalism.

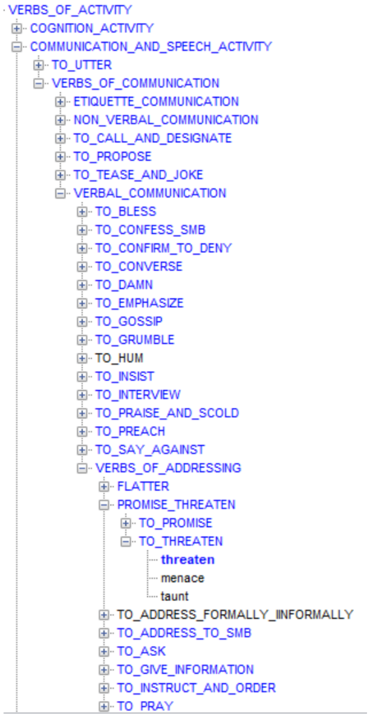
3.1. Word meanings

Lexical meanings are presented in the form of so called Semantic Classes (SCs). SCs form the semantic hierarchy – a thesaurus-like tree, consisting of universal senses, such as MOTION, HUMAN, ANIMAL, ORGANIZATION, or SPORT which are filled with lexical contents for different languages – lexical classes. A SC denotes a place where a word in the relevant meaning is positioned in the tree. The fragment of the hierarchy is shown in Figure 3. SCs are written in capitals and lexical classes in small letters; when opening each plus-sign, one can see the descendants of the parent class, both SCs and lexical classes.

The whole tree includes more than 200,000 universal SCs. In the current version of the markup, we decided to reduce it and used the shortened variant which consists of the hyperonym SCs. It means that we do not show classes like CAT, TORTOISE or ELEPHANT in the annotation, but point out the hyperonym SC ANIMAL instead.

For instance, sentence (1) is attributed with the SCs of the full hierarchy in (1a) and with hyperonym SCs in (1b). The SCs in both cases are spelt in green capitals with quotes:

(1) The players ran hard all the time.



- VERBS_OF_ACTIVITY
  - COGNITION_ACTIVITY
  - COMMUNICATION_AND_SPEECH_ACTIVITY
    - TO_UTTER
    - VERBS_OF_COMMUNICATION
      - ETIQUETTE_COMMUNICATION
      - NON_VERBAL_COMMUNICATION
      - TO_CALL_AND_DESIGNATE
      - TO_PROPOSE
      - TO_TEASE_AND_JOKE
      - VERBAL_COMMUNICATION
        - TO_BLESS
        - TO_CONFESS_SMB
        - TO_CONFIRM_TO_DENY
        - TO_CONVERSE
        - TO_DAMN
        - TO_EMPHASIZE
        - TO_GOSSIP
        - TO_GRUMBLE
        - TO_HUM
        - TO_INSIST
        - TO_INTERVIEW
        - TO_PRAISE_AND_SCOLD
        - TO_PREACH
        - TO_SAY_AGAINST
        - VERBS_OF_ADDRESSING
          - FLATTER
          - PROMISE_THREATEN
            - TO_PROMISE
            - TO_THREATEN
              - threaten
              - menace
              - taunt
          - TO_ADDRESS_FORMALLY_INFORMALLY
          - TO_ADDRESS_TO_SMB
          - TO_ASK
          - TO_GIVE_INFORMATION
          - TO_INSTRUCT_AND_ORDER
          - TO_PRAY

Figure 3: A fragment of the full semantic hierarchy

(1a) The players "PLAYER_OF_GAMES" ran "TO_RUN" hard "INTENSITY_OF_ACTIVITY" all the time "TIME".

(1b) The players "HUMAN" ran "MOTION" hard "CH_OF_INTENSITY" all the time "TIME".

Currently, the hyperonym hierarchy includes more than 650 classes and is available on Github[6]. All classes include comments and nearly all of them – sets of examples extracted from our corpus as well. If a SC has no instances in the "Examples" field, it means that the class is divided into the subclasses with narrower meanings and, therefore, is not used in the annotation itself, as the subclasses are used instead of the parent class.

For example, on Figure 4, "ENTITY" class is a hyperonym for the SCs "FOOD", "ORGANIZATION", "PHYSICAL OBJECT", "MENTAL OBJECT", "SUBSTANCE", and its other descendants, therefore, these child classes will be indicated in the annotation instead of "ENTITY".

The hierarchy of the hyperonym classes is easier to operate with and more understandable for users. Its other benefit is that it gives better opportunities for elaborating parsers, especially neural networks based, which can reproduce the markup of the given format. As our Russian dataset is about 400,000 tokens and the English one about 150,000, the number of 200,000 SCs would be too much for the machine learning of the parser.

Nevertheless, it is not always evident which generalization level can be considered optimal. On the one hand, the purpose is to present a structurally balanced semantic tree; on the other hand, the labels of the classes used in the annotation should be intuitively understandable.

Moreover, in spite of the fact that most hyperonym classes seem to be exact enough to denote definite word meanings, there are still cases where such a generalization leads to losing the distinction between homonyms which are positioned closely in the tree. For instance, we can differentiate between 'pour' as 'flow in a stream' (like 'the river poured into the sea', hyperonym SC "MOTION") and 'pour' as 'to rain hard' (like 'The rain is just pouring down', hyperonym SC "TO_TAKE_PLACE_IN_NATURE"). But we can not differentiate between the first meaning and 'pour' as 'to move in large amounts or numbers' (like 'The people poured along the street'), because the hyperonym class here is also "MOTION".

However, if further usage of the format demands to make the hierarchy more detailed, we can surely make the necessary supplements. The feedback in this respect is very important and will help us to define the optimal detailness level of the hierarchy.

3.2. Relations between words

In Compreno, there are syntactic roles, called surface slots, and semantic roles, called deep, or semantic, slots.

Syntactic roles are language-specific and determine only surface relations between words. It means that in examples (2a)-(2d) we have one surface slot: Object_Indirect_To, which corresponds to different semantic slots indicated after the sentences:

(2a) I talked [to Peter] – Addressee;

(2b) The rule refers only [to children] – Object;

(2c) The tune [to the song] – Purpose;

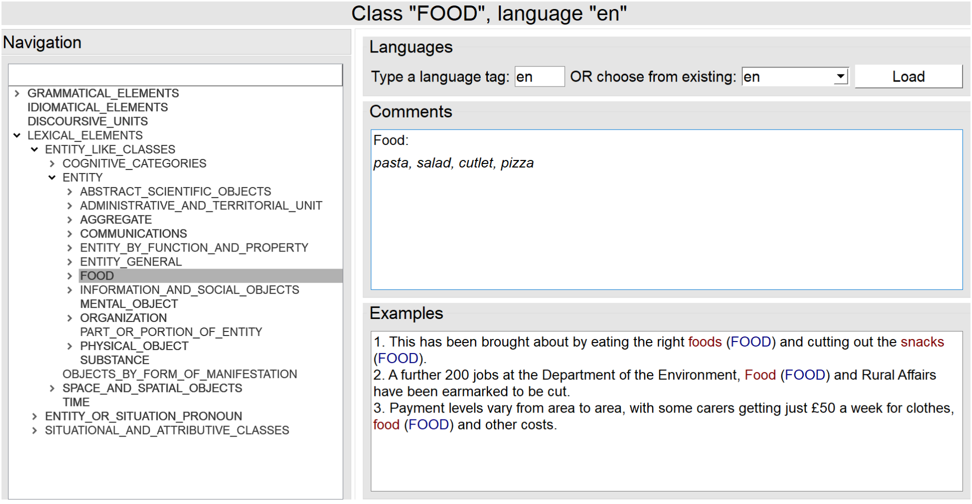
(2d) His reply [to a question] – Stimulus.

General differences in E-UD and Compreno approaches towards the presentation of the relations between words can be shown in Table 1:

Model	E-UD	Compreno
Have semantic dependencies	no	yes
Have syntactic dependencies	yes	yes
Actant and circumstantial dependencies with similar surface realizations get different syntactic roles	yes	no
Syntactic dependencies with similar surface realizations which depend on nominal vs verbal cores get different syntactic roles	yes	no

Table 1: E-UD and Compreno differences in the description of the relations between words

[6] https://github.com/compreno-semantics/semantic-hierarchy



Class "FOOD", language "en"

- GRAMMATICAL_ELEMENTS
- IDIOMATICAL_ELEMENTS
- DISCOURSIVE_UNITS
- LEXICAL_ELEMENTS
  - ENTITY_LIKE_CLASSES
    - COGNITIVE_CATEGORIES
    - ENTITY
      - ABSTRACT_SCIENTIFIC_OBJECTS
      - ADMINISTRATIVE_AND_TERRITORIAL_UNIT
      - AGGREGATE
      - COMMUNICATIONS
      - ENTITY_BY_FUNCTION_AND_PROPERTY
      - ENTITY_GENERAL
      - FOOD
      - INFORMATION_AND_SOCIAL_OBJECTS
      - MENTAL_OBJECT
      - ORGANIZATION
      - PART_OR_PORTION_OF_ENTITY
      - PHYSICAL_OBJECT
      - SUBSTANCE
    - OBJECTS_BY_FORM_OF_MANIFESTATION
    - SPACE_AND_SPATIAL_OBJECTS
    - TIME
  - ENTITY_OR_SITUATION_PRONOUN
  - SITUATIONAL_AND_ATTRIBUTIVE_CLASSES

Type a language tag: en OR choose from existing: en

Food:
pasta, salad, cutlet, pizza

1. This has been brought about by eating the right foods (FOOD) and cutting out the snacks (FOOD).
2. A further 200 jobs at the Department of the Environment, Food (FOOD) and Rural Affairs have been earmarked to be cut.
3. Payment levels vary from area to area, with some carers getting just £50 a week for clothes, food (FOOD) and other costs.

Figure 4: The semantic hierarchy of the hyperonym semantic classes

As the surface slots do not strictly depend on semantic roles and can refer both to actant and circumstantial dependencies, one surface slot can correspond to different UD dependencies, for instance, 'iobj' or 'obl'. Besides, verbal and nominal cores attach the same surface slots: in (3a) and (3b) there is one slot for [on the hill]-dependency, unlike it is in UD, where (3a) corresponds to 'nmod' for nominal cores and to 'obl' for verbal cores:

(3a) The house [on the hill];

(3b) He stood [on the hill].

On the other side, introducing preposition extensions in E-UD, such as 'nmod:on' or 'obl:on', makes the Compreno surface slots a bit closer to the UD dependencies. As we adopted the UD syntax, we did not use Compreno surface slots in CoBaLD annotation. Nevertheless, we extract this information from the parsing trees and use it during the Compreno-to-UD conversion process.

Unlike surface slots, deep slots (DSs) are universal across languages and denote semantic relations between words. Their key difference from valencies is that they define not only actant dependencies, but all dependencies a word can attach. Moreover, each deep slot can be filled with a strict set of SCs: Agent slot with beings, organizations, countries, and so on, Time slot with words of temporal semantics, Reason slot with all SCs.

Most dependencies get both a surface and a semantic role. The exceptions are grammatical dependencies (articles or prepositions, for instance), and idiomatical ones (like 'beans' in 'spill the beans').

The inconvenience is that the full list of the Compreno DSs is more than 300. Such a detailed description can be useful for some special tasks (like building semantic sketches (Ponomareva et al., 2021)), but seems too heavy for most purposes. Namely, it can significantly hamper parsers learning on the datasets of our volume. For this reason, we made some generalizations here, too, as we did with the SCs. We joined all characteristic slots together, parentheticals, specifications, and united a number of slots with the same semantics but different filling.

For example, the Compreno markup suggests several slots for locative and temporal adjuncts, as shown in tables 2 and 3 correspondingly. Locative slot can be filled with words denoting places and locations, Locative Event – with events in locative contexts, Locative Orientation – with orientational adjectives and adverbs, – see the examples of each slot in Table 2. In the generalized description, these DSs are merged to form one Locative DS.

Table 3, in turn, shows the correlation between temporal slots in full Compreno markup and in the generalized version: five Time slots in full markup differentiating mostly through their fillers correspond to one Time slot in CoBaLD annotation.

Locative DSs in Compreno	Locative Examples	Locative DS in Co-BaLD
Locative	anywhere [in the world], hide [under the table]	Locative
Locative_Event	they were [on the rehearsal]	
Locative_Orientation	turn [to the right], to go [up]	

Table 2: Correlation between Locative DSs in full Compreno markup and in CoBaLD annotation

Time DSs in Compreno	Time examples	Time DS in CoBaLD
Time	He came [yesterday/at 5 o’clock].	Time
Time_Being	[post-Bush] economy	
Time_Entity	[After a sandwich and a pint], we headed to Trinity College.	
Time_Situation	[When the war started], nobody believed it.	
Time_Source	someone [from 1860]	

Table 3: Correlation between Time DSs in full Compreno markup and in CoBaLD annotation

The shortened number of the DSs is 143. Their list is available on Github[7] with comments and examples for every slot. Depending on further work with the format, we will probably make more generalizations in the DS pattern to make it simpler if the current set would seem too detailed.

As the UD markup does not suggest semantic domain, we just added it into the annotation schema by creating two new columns in CONLL-U Plus for the SCs and the DSs. Nevertheless, the creation of the dataset annotated in the new format presupposed that we made automatic Compreno markup first, extracted its semantic pattern and converted its morphosyntax into E-UD. Otherwise, if tokenization or heads in a sentence would differ in two formats, homogeneous markup of all three levels would be impossible.

4. Conversion

As we showed in (Ivoylova et al., 2023), the conversion process is a challenging task due to the asymmetries between the formalisms. Below, we highlight key issues related to it, focus on the distinctions of the conversion to E-UD as compared with the conversion to basic UD, and compare the Compreno-To-UD conversion process for English and Russian.

4.1. Morphology

In morphology, the asymmetry between the models concerns the following areas: tokenization, lemmatization, POS-tagging, and defining the sets of grammatical features. In the current version, we have made some optimizations for the Russian converter and elaborated the converter for English.

The description of the Russian converter is given in (Ivoylova et al., 2023). Its optimizations were not crucial. Most significant changes affected lemmas: first of all, verb aspect presentation. In Compreno, verb lemmas are presented in perfective forms, whereas in UD, the verbal lemma must correspond to the aspect of its form in a sentence. We expanded the list of tokens with incorrect lemmas and added the following columns for each verb (instead of two columns ’perfective’/’imperfective’ in the previous variant) – reflexive form, perfective aspect / non-reflexive form, perfective aspect / reflexive form, imperfective aspect / non-reflexive form, imperfective aspect.

As for tokenization, we expanded the list of tokens which were merged in Compreno and re-tokenized them. We also added a new column ‘XPOS’ showing the token’s POS-tag before the conversion, introduced the ’NumForm’ feature which marks the way numerals are written (Word(’six’), Digit(6), or Roman(XI)), and returned features to words which were marked as abbreviations (they had only ‘ABBR=Yes’ feature in the previous version).

Now let us discuss the English conversion process.

Tokenization

One of the problems we faced was to convert tokens that have:

- possessive ‘s
- contracted forms of verbs (‘ve, ‘re, ‘ll etc.)

We split such multiword expressions rule-based and add a string above them as the UD format requires:

```
11-12   government’s
11      government      government
12      ‘s              ‘s
```

Another challenge was the conversion of multiword expressions (as in Russian): in Compreno, items like ‘more than’ are represented as one token, while in UD, these are two tokens, which entails the necessity to re-tokenize them.

We solved this problem the same way as in the previous version except for one change: as we expanded the list of merged tokens (now it contains approximately 10,000 merged multiword expressions), we decided to use automatic pre-marking and then check the markup manually. We had to find a morphological parser to process the list and chose the Spacy UDPipe library[8] as it parses in the UD format. To integrate these tokens, we used the same script as for the Russian converter.

[7] https://github.com/compreno-semantics/semantic-slots

[8] https://spacy.io/universe/project/spacy-udpipe

Lemmatization

The conversion of lemmas seems to be one of the easiest parts of the process but there are also some inconsistencies between the Compreno and the UD markups. For instance, there are hash lemmas in Compreno which are attributed to some special sets of tokens. Among them, the #UnknownWord lemma, which is given to words when Compreno struggles to recognize its lemma or the #Number lemma meant for numerals. The vast majority of such lemmas can be replaced with these tokens, for instance, 'Speakerboxxx' → #UnknownWord.

POS-tagging

Key differences between UD and Compreno POS-tags were already described in (Ivoylova et al., 2023). Shortly, some of the Compreno POS-tags do not correspond to the UD tags. Compreno does not distinguish 'Auxiliary' and 'Verb' POS-tags – the 'Verb' tag is used for all lexical and auxiliary verbs, unlike it is in UD. Besides, there is no 'Determiner' POS-tag in Compreno, attributed to articles and demonstrative pronouns in UD: instead, Compreno has the 'Article' tag for articles and the 'Pronoun' tag for pronouns, which are additionaly marked with features denoting the pronoun type (personal, demonstrative, and so on). Such POS-tags were converted with the help of morphological features and the syntax module.

Also, there is a special POS-tag 'Invariable' in Compreno: usually, it refers to discourse units and parenthetical constructions. In Russian, we created a list of such tokens for further conversion. In English, we converted this tag with the help of other morphological features. After the conversion, such tokens normally get ADV/ADJ POS-tags.

Grammatical Features

Another source of asymmetry concerns grammatical features.

Some types of grammatical information in Compreno have to be collected from different syntactic categories, though in UD, it is stored in one slot. For instance, this concerns pronoun types. Features that mark reciprocal and personal pronouns are located in different slots of morphological markup, and to distinguish relative and interrogative pronouns, we had to use syntax data.

Another questionable moment concerns gerunds: we decided to follow the UD instruction in case of their definition despite the fact that it seems to contradict both Compreno principles and English grammar rules saying that 'The primary difference between a gerund and a participle, therefore, is that while a participle is functionally comparable to an adjective, a gerund is functionally comparable to a noun. There is also a secondary difference: that gerunds do not combine with auxiliaries in the way that participles do' (Huddleston, 2002). UD documentation distinguishes present participles and gerunds by the precedence of 'to be', which entails marking the form 'decreasing' in 'Another contributory factor has been the decreasing consumption of iodised salt used in foods.' as 'Gerund' instead of participle although the context here is functionally comparable to an adjective. Probably, this decision comes from practice-oriented format of the UD markup.

Comparison of Russian and English conversion

As morphology is language-specific, Russian and English have different sets of grammatical features. It means we had to re-write the code for the English converter almost completely except for some special parts mentioned above.

4.2. Syntax

Compreno provides highly detailed data concerning syntactic relations, namely, surface slots which are language specific. However, slots like 'Subject' or 'Object Direct' are present in different languages of the model.

During the conversion, we tried to match the UD dependency relations with Compreno surface slots where possible, but as the dependencies in UD are not strictly syntactical, in some cases we had to resort to other sources of information such as POS or DSs. For instance, there are surface slots which would correspond to both 'ccomp' and 'acl' relations, so the conversion should depend on the POS-tag of the head here. That is, in sentences 'Statistics released last week showed that stockpiles of oil products in the US had risen', and 'An indication that severe supply disruptions may not arise this winter, barring any serious incident' underlined words are both marked as 'Clause Finite' in Compreno, but 'risen' is 'ccomp' while 'arise' is 'acl' in UD.

As the E-UD standard is more syntax-oriented in comparison to basic UD, the Compreno data is easier to convert to E-UD. For this reason, the conversion was made first to E-UD and then to basic UD. Compared to our previous work on Russian, we had to re-write the entire conversion script in order to introduce E-UD features such as ellipsis restoration, 'ref' tag and some others, for which the Compreno model provided all the information we needed. Nevertheless, language specific changes of the script are minimal: we have to switch the surface slot lists, and there are some differences in 'det' conversion.

The distinctions between the UD and Compreno annotation schemes mainly concern dependency heads. Usually, Compreno considers content words to be heads, but there is an exception for the copula. Unlike UD, Compreno considers copula to be the core. Therefore, we had to swap the dependency heads for cases like ‘the girl is beautiful’ and label dependency relations accordingly in order to comply with the UD principles, although this might disrupt semantic dependencies.

Another difference concerns heads of ‘including’, ‘according to’, and ‘such as’. For instance, the Compreno model would label the sentence ‘According to her, it is correct.’ as in Figure 5, whereas in UD, it would look like in Figure 6.

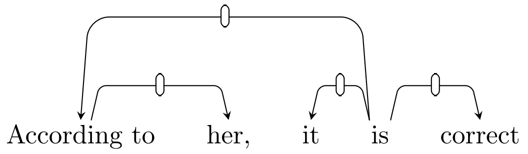


Figure 5: Compreno markup for ‘According to her, it is correct.’

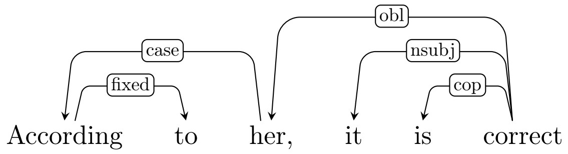


Figure 6: UD markup for ‘According to her it is correct.’

We managed to resolve practically all such issues by converting them to the UD standard, except for one E-UD feature. As one can see in Figure 7, there is ‘nsubj’ relation for ‘cake’ depending on ‘beautiful’. The Compreno model does not annotate this type of control, therefore, we cannot convert it.

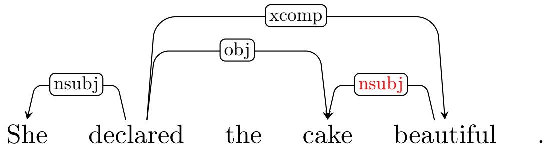


Figure 7: Secondary ‘nsubj’ relation

On the other hand, there is linguistic information available to us which we have not used yet: reference links. The Compreno model also provides information on anaphor and coreference. So far we have only introduced the antecedent links for relative pronouns as the tag ‘ref’ implies, that is, for ‘Otherwise, you see people make agreements which then fall apart.’ the word ‘which’ refers to ‘agreements’ and thus would get an E-UD tag ‘8:ref’, while the referred noun would get an additional relation ‘nsubj’.

5. CoBaLD annotation

The annotation process was organized the same way we did it with Russian corpus. First, the dataset was annotated automatically with the help of the Compreno parser. The annotation includes SCs, DSs, and boundaries of the constituents. At this stage, full Compreno markup was used, including full sets of SCs and DSs. After that, we have checked all the annotated sentences manually and corrected them.

Nevertheless, there still can be ambiguity cases, which different annotators understand differently.

To evaluate the ambiguity level, we have measured inter-annotator agreement on two small samples - 100 sentences concerning politics and 100 sentences concerning technologies. The results are presented in Table 4:

		Full markup, %	Simplified markup, %
Tech	SemSlot	97.12	97.36
	SemClass	97.7	97.78
	Heads	98.29	98.29
	Overall	93.10	93.44
Politics	SemSlot	98.75	98.8
	Semclass	98.34	98.65
	Heads	99.58	99.58
	Overall	96.67	97.04

Table 4: Inter-annotator agreement for semantic markup, joint probability

Earlier, we have done the same measurements for the Russian corpus on a 100 sentence sample as well. For Russian, overall inter-annotator agreement was 94.17% (97.28% - for SCs, 97.36% - for DSs, and 99.07% - for heads of the constituents, for details, see (Petrova et al., 2023)).

After the manual check and the correction of the semantic pattern, the parser builds parsing trees according to the semantic annotations. Morphological and syntactic information is taken from the trees as in Compreno, such information is not present in the markup itself. Further, we convert morpho-syntactic annotation into the UD format with the Compreno-To-UD Converter, substitute full semantic annotation with hyperonym classes and generalized DSs, and add the semantic pattern to the annotation.

The results of the conversion are also manually checked, but the verification algorithm is different here. First, the idea was to check each sentence and make manual edits if necessary, but it turned out to be more effective to check small samples and to fix the bugs in the converter. We have made

```
# text = The full economic costs of the disaster remain unclear.
1   The the DET Article Definite=Def|PronType=Art   4   det 4:det   _   _   ARTICLES
2   full    full    ADJ Adjective   Degree=Pos  4   amod    4:amod  _   Characteristic  CH_SPHERE_OF_COVERAGE
3   economic    economic    ADJ Adjective   Degree=Pos  4   amod    4:amod  _   Sphere  ECONOMY
4   costs   cost    NOUN    Noun    Number=Plur 8   nsubj   8:nsubj _   Object  MONEY
5   of  of  ADP Preposition _   7   case    7:case  _   _   PREPOSITION
6   the the DET Article Definite=Def|PronType=Art   7   det 7:det   _   _   ARTICLES
7   disaster    disaster    NOUN    Noun    Number=Sing 4   nmod    4:nmod:of   _   Object_Situation    BAD_DANGEROUS_EVENT
8   remain  remain  VERB    Verb    Mood=Ind|Number=Plur|Person=3|Tense=Pres|VerbForm=Fin   0   root    0:root  _   Predicate   BE
9   unclear unclear ADJ Adjective   Degree=Pos  8   xcomp   8:xcomp SpaceAfter=No   State   CH_PERCEPTIBILITY
10  .   .   PUNCT   PUNCT   _   8   punct   _   _   _   _
```

Figure 8: Annotation example

several iterations, and currently, the quality of the latest test sample is as presented in Table 5.

Lemma	POS	Feats	Heads	Deprel	E-UD	Overall
98.68	95.48	94.92	95.48	96.80	93.97	85.12

Table 5: Joint probability for automatic conversion and human annotators, %

Nevertheless, we continue the improvement of the converter, and renew the information about the conversion quality on the project page.

Final annotation is shown on Figure 8. Words of the sentence are written in a column; each word's line contains the annotated information. Columns 1-10 correspond to default columns in CONLL: word index, word form, lemma, universal POS tag, optional language-specific (or treebank-specific) POS tag, morphological features, head of the word, UD relation to the head (deprel), head - deprel pairs, other information[9]. Columns 11 and 12 are the new ones introduced to the markup for denoting word meanings and semantic relations between words respectively.

6. Dataset

For English annotation, we have chosen the BBC dataset[10] (Greene and Cunningham, 2006), as it is freely available, and besides, quite similar to the Russian news dataset we had used for the annotation. The BBC dataset is divided into five topics, namely, business, entertainment, politics, sport and tech. For our project, we selected sentences from every topic evenly.

The whole BBC corpus contains around 963,000 tokens. Our dataset is a bit more than 150,000 tokens, which corresponds to approximately 15% of the whole corpus.

It is important to note that the BBC dataset consists of complete texts while our annotation takes separate sentences into account. It causes inconvenience in cases of direct speech such as 'Chelsea assistant boss Steve Clarke said: "I would rather talk about the football but we think it was something thrown from the crowd. He did not require stitches."' Splitting such fragments in separate sentences, we get two sentences with only one quotation mark each. Therefore, we manually added or removed quotation marks here during the annotation process.

We have labeled the corpus CoBaLD Eng Dataset. It is available on CoBaLD Github[11] page, and includes morphological, syntactic and semantic markup in the CONLL-Plus format described above.

7. Conclusion and further plans

In the present paper, we have suggested a new annotation format aimed at three level markup including morphology, syntax and semantics. The format is based on the widely acknowledged E-UD annotation schema which is supplemented with the markup of lexical meanings and relations between words. These two parts of the semantic pattern come from the Compreno model and represent simplified and generalized version of the Compreno semantics. Using hyperonyms instead of full SC set as well as the current generalized variant of the DS set may demand further optimization, for instance, further reduction of the DS set or, on the contrary, a more detailed variant of the semantic hierarchy. The feedback on using the dataset annotated according to the standard would help to define the optimal level of the semantic description detailness.

Our further perspectives concern several areas. First, the work on DL-based parser is in progress, which would produce the markup in the given standard as we did for Russian. It presupposes the enlargement of the annotated datasets, especially the English corpus. Second, we are planning to add corpora in other languages, both by annotating new corpora and with the help of Cross-Lingual Transfer techniques. Third, we are going to add new features to the format itself, namely, to widen the coreference description as it is possible to extract all necessary information from the Compreno parsing trees, and to experiment with adding a new column for the Compreno surface slots, at least, their simplified version.

[9] for details, see https://universaldependencies.org/format.html

[10] All rights, including copyright, in the content of the original articles are owned by the BBC.

[11] https://github.com/CobaldAnnotation/CobaldEng

References

Omri Abend and Ari Rappoport. 2013. Universal conceptual cognitive annotation (ucca). In Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 228–238.

KV Anisimovich, K Yu Druzhkin, KA Zuev, FR Minlos, MA Petrova, and VP Selegei. 2012. Syntactic and semantic parser based on abbyy compreno linguistic technologies. In Computational Linguistic and Intellectual Technologies, pages 91–103.

Ramy Baly, Hazem Hajj, Nizar Habash, Khaled Bashir Shaban, and Wassim El-Hajj. 2017. A sentiment treebank and morphologically enriched recursive deep models for effective sentiment analysis in arabic. ACM Transactions on Asian and Low-Resource Language Information Processing (TALLIP), 16(4):1–21.

Laura Banarescu, Claire Bonial, Shu Cai, Madalina Georgescu, Kira Griffitt, Ulf Hermjakob, Kevin Knight, Philipp Koehn, Martha Palmer, and Nathan Schneider. 2013. Abstract meaning representation for sembanking. In Proceedings of the 7th linguistic annotation workshop and interoperability with discourse, pages 178–186.

Marie-Catherine De Marneffe, Bill MacCartney, Christopher D Manning, et al. 2006. Generating typed dependency parses from phrase structure parses. In Lrec, volume 6, pages 449–454.

Kim Gerdes, Bruno Guillaume, Sylvain Kahane, and Guy Perrier. 2018. Sud or surface-syntactic universal dependencies: An annotation scheme near-isomorphic to ud. In Universal dependencies workshop 2018.

Derek Greene and Pádraig Cunningham. 2006. Practical solutions to the problem of diagonal dominance in kernel document clustering. In Proc. 23rd International Conference on Machine learning (ICML'06), pages 377–384. ACM Press.

Jan Hajic, Barbora Vidová-Hladká, and Petr Pajas. 2001. The prague dependency treebank: Annotation structure and support. In Proceedings of the IRCS workshop on linguistic databases, pages 105–114.

Rodney Huddleston. 2002. Verb. The Cambridge grammar of the English language, page 81.

Alexandra Ivoylova, Darya Dyachkova, Maria Petrova, and Mariia Michurina. 2023. The problem of linguistic markup conversion: the transformation of the compreno markup into the ud format. In International Conference on Computational Linguistics and Intellectual Technologies «Dialog.

Yucheng Li, Shun Wang, Chenghua Lin, Frank Guerin, and Loïc Barrault. 2023. Framebert: Conceptual metaphor detection with frame embedding learning. arXiv preprint arXiv:2302.04834.

Ekaterina Manicheva, Maria Petrova, Elena Kozlova, and Tatiana Popova. 2012. The compreno semantic model as integral framework for multilingual lexical database. In Proceedings of the 3rd Workshop on Cognitive Aspects of the Lexicon, pages 215–230.

Mitchell Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. 1993. Building a large annotated corpus of english: The penn treebank.

Joakim Nivre, Daniel Zeman, Filip Ginter, and Francis Tyers. 2017. Universal dependencies. In Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Tutorial Abstracts.

MA Petrova. 2014. The compreno semantic model: the universality problem. International Journal of Lexicography, 27(2):105–129.

Maria Petrova, Alexandra Ivoylova, Ilya Bayuk, Darya Dyachkova, and Mariia Michurina. 2023. The cobald annotation project: the creation and application of the full morpho-syntactic and semantic markup standard. In Proceedings of the International Conference "Dialogue, volume 2023.

Carl Pollard and Ivan A Sag. 1994. Head-driven phrase structure grammar. University of Chicago Press.

Maria Ponomareva, Maria Petrova, Julia Detkova, Oleg Serikov, and Maria Yarova. 2021. Semsketches2021: experimenting with the machine processing of the pilot semantic sketches corpus. In Komp'juternaja Lingvistika i Intellektual'nye Tehnologii, pages 560–570.

Edoardo Maria Ponti, Ivan Vulić, Goran Glavaš, Nikola Mrkšić, and Anna Korhonen. 2018. Adversarial propagation and zero-shot cross-lingual transfer of word vector specialization. arXiv preprint arXiv:1809.04163.

Geoffrey Sampson. 2002. Briefly noted-english for the computer: the susanne corpus and analytic scheme. Computational Linguistics, 28(1):102–103.

Sebastian Schuster and Christopher D Manning. 2016. Enhanced english universal dependencies: An improved representation for natural language understanding tasks. In Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16), pages 2371–2378.

Hiroshi Uchida and Meiying Zhu. 2001. The universal networking language beyond machine translation. In International Symposium on Language in Cyberspace, Seoul, pages 26–27.

Aaron Steven White, Drew Reisinger, Keisuke Sakaguchi, Tim Vieira, Sheng Zhang, Rachel Rudinger, Kyle Rawlins, and Benjamin Van Durme. 2016. Universal decompositional semantics on universal dependencies. In Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, pages 1713–1723.