applied sciences

MDPI

Article

Achieving Semantic Consistency for Multilingual Sentence Representation Using an Explainable Machine Natural Language *Parser* (*MParser*)

Peng Qin [1], Weiming Tan [1], Jingzhi Guo [1], Bingqing Shen [2] and Qian Tang [1,3,*]

[1] Faculty of Science and Technology, University of Macau, Macau 999078, China; yb77428@connect.um.edu.mo (P.Q.); wade.tan@connect.um.edu.mo (W.T.); jzguo@um.edu.mo (J.G.)
[2] School of Software, Shanghai Jiao Tong University, Shanghai 200240, China; sunniel@sjtu.edu.cn
[3] College of Business, Beijing Institute of Technology, Zhuhai 519088, China
* Correspondence: tang_qiansxy2022@126.com

Abstract: In multilingual semantic representation, the interaction between humans and computers faces the challenge of understanding meaning or semantics, which causes ambiguity and inconsistency in heterogeneous information. This paper proposes a Machine Natural Language Parser (MParser) to address the semantic interoperability problem between users and computers. By leveraging a semantic input method for sharing common atomic concepts, MParser represents any simple English sentence as a bag of unique and universal concepts via case grammar of an explainable machine natural language. In addition, it provides a human and computer-readable and -understandable interaction concept to resolve the semantic shift problems and guarantees consistent information understanding among heterogeneous sentence-level contexts. To evaluate the annotator agreement of MParser outputs that generates a list of English sentences under a common multilingual word sense, three expert participants manually and semantically annotated 75 sentences (505 words in total) in English. In addition, 154 non-expert participants evaluated the sentences' semantic expressiveness. The evaluation results demonstrate that the proposed MParser shows higher compatibility with human intuitions.

Keywords: document representation; semantic analysis; natural language processing; conceptual modeling; universal representation

check for updates

Citation: Qin, P.; Tan, W.; Guo, J.; Shen, B.; Tang, Q. Achieving Semantic Consistency for Multilingual Sentence Representation Using an Explainable Machine Natural Language *Parser* (*MParser*). *Appl. Sci.* **2021**, *11*, 11699. https://doi.org/10.3390/app112411699

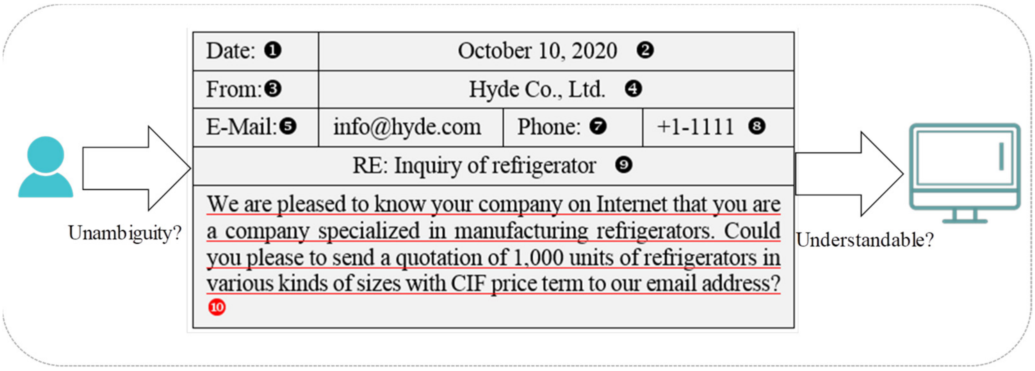
Academic Editors: Arturo Montejo-Ráez and Salud María Jiménez-Zafra

Received: 16 November 2021
Accepted: 8 December 2021
Published: 9 December 2021

1. Introduction

Multilingual semantic representation [1] presents words, phrases, texts, or documents in heterogeneous parties (e.g., English and Chinese) to achieve semantic consistency. It has been applied in several areas, such as machine translation [2], question answering [3], and document representation [4,5]. The process of parsing a natural language sentence to its semantic representation is called semantic parsing [6], which parses the sentences without representing the syntactic classification of the components of the sentence. Semantic parsing is an essential process and has attracted great attention in multilingual semantic representation and NLP research over the last few decades [6]. Typically, a semantic parser labels each word in the original sentence according to its semantic role or represents each compound component based on its meaning [7]. Several semantic approaches are proposed for parsing natural language sentences in semantic representation, such as Groningen Meaning Bank (GMB) [8] and abstract meaning representation (AMR) [9]. Still, their annotation schemes are designed for individual languages that have language-dependent features. Because many applications require multilingual capabilities, several efforts are underway to create more cross-lingual natural language resources such as universal conceptual cognitive annotation (UCCA) [10], universal networking language (UNL) [11], and universal dependencies (UD) [12]. They are the framework for cross-linguistically consistent grammatical annotation. Despite these efforts, some remaining interlanguage variations important for practical usage are not yet captured by the efforts. They create obstacles to a truly cross-lingual meaning representation that enables downstream applications written in one language to be applicable for other languages. Using cross-lingual language to perform cross-lingual semantic parsing for one language to improve the representation of another language remains a largely under-explored research question. This paper focuses on the problem of multilingual semantic interoperability in semantic representation.

In semantic analysis and labeling, texts and documents are generally very complex because of flexible structural and complex morphological grammars. The state-of-art semantic parser methods and applications have not achieved satisfying results. One technical challenge is the lack of consistent conversions across domains. The heterogeneous text may share heterogeneous meaning and cause semantic loss or misunderstanding between a computer and a user [13]. For example, Figure 1 shows an English inquiry sheet for illustrating the multilingual semantic interoperability problems. The table consists of 10 cells; cells 1–9 contain a single atomic concept, i.e., "*one cell one atomic concept*" (e.g., Date in cell 1). However, one atomic concept may have multiple meanings. For instance, the word "*company*" in cell 10 refers to several meanings such as "*a commercial business*" and "*the fact or condition of being with another or others, especially in a way that provides friendship and enjoyment*". To achieve accurate atomic concept exchange and guarantee semantic consistency in cells 1–9, several document representation approaches [5,14] are proposed to solve the heterogeneous concept or meaning exchange problem. An effective solution is the collaboration mechanism that connects heterogeneous domains or contexts, allowing the exchange of heterogeneous semantic documents by a semantic input method (SIM) approach [15]. However, some sentences also contain sequences of atomic concepts for a free-text cell (e.g., cell 10), which makes it hard to ensure that the meaning (M_1) of an English sentence *Ei:* = *List* $(w_1, w_2, \ldots, w_n)$ and the meaning (M_2) of a translated Chinese sentence *Cj* := *List* $(w_1, w_2, \ldots, w_n)$ will be semantically equivalent. The reasons for causing $M_1 \neq_s M_2$ ("$\neq_s$" refers to not semantically equal) include:



Date: ❶	October 10, 2020 ❷		
From:❸	Hyde Co., Ltd. ❹		
E-Mail:❺	info@hyde.com	Phone: ❼	+1-1111 ❽
RE: Inquiry of refrigerator ❾			
We are pleased to know your company on Internet that you are a company specialized in manufacturing refrigerators. Could you please to send a quotation of 1,000 units of refrigerators in various kinds of sizes with CIF price term to our email address? ❿			

Figure 1. Complex document interaction between computers and users.

(1) Heterogeneous grammatical rules: The language grammars of the components in E_i and C_i have their own rules to generate a sentence and it is impossible to achieve a one-to-one mapping.
(2) Synonyms and homonyms: Each term in E_i may have several synonyms or homonyms. A wrong term in meaning may cause semantic ambiguity.
(3) Peculiar language phenomena: Some phenomena in E_i never appear in C_i, resulting in asymmetric mapping. For example, the particles of "を, に, で, へ, より" in Japanese do not have counterparts in Chinese.

Therefore, the same sentence will produce completely different scenarios in a heterogeneous text, and the original meaning in mind may be shifted to another meaning. The above problems are called semantic shift problems that change a sentence's original meaning in multilingual semantic representation. Moreover, in natural language texts, users cannot express their information needs in a computer-understandable way or interpret the representation correctly due to problems in representing complex semantics. Therefore, the development of a novel model has been motivated by the following aspects:

(1) Computer-human-understandable representation: providing information understandable by both computers and humans, realizing the accurate interpretation of sentences in the human-computer messaging cycle of humans and computers without ambiguity.
(2) Accurate semantic representation among computing applications: applying computer-human-understandable information in computing applications and enabling information to be semantically interoperable.
(3) Automated multilingual information processing by software agents: allowing multilingual information to be automatically processed across domains and contexts.

Thus, this research proposes a new multilingual semantic representation parser for sentence-based text or documents that enhances textual representation and reduces multilingual ambiguity. Based on our previous conceptual work [16], we propose a novel Machine Natural Language Parser (Mparser) to realize universal representations between computers and users unambiguously. The explainable MParser parses a simple English sentence, resolving complex concepts towards a bag of universal concepts sentence-readable and -understandable for any heterogeneous information, and mediates contextual human natural languages collaboratively, as shown in Figure 2. The universal concepts sentence shares a common concept at both the syntactic and semantic levels between users and computers.

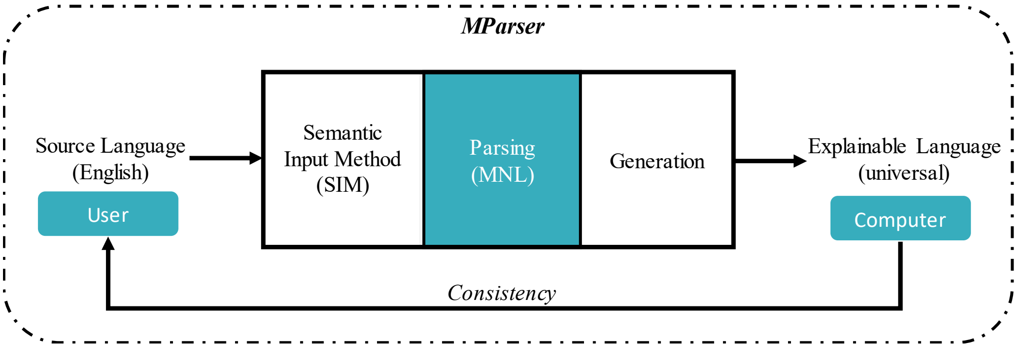

MParser
Source Language (English)
User
Semantic Input Method (SIM)
Parsing (MNL)
Generation
Explainable Language (universal)
Computer
Consistency

Figure 2. A general MParser process.

To achieve consistency and universal representation, *MParser* designs from human input and sentence generation:

(1) In the human input, each unique concept is collaboratively edited with SIM [15] based on a common dictionary (CoDic) [17] for eliminating atomic concept ambiguity and morphological features. Thus, a simple English sentence can be converted to a sequence of unique concepts across conversational contexts.
(2) To maintain complex semantic concept consistency between computers and users, an MParser for English sentences parses the semantic roles between English words and represents them for deriving a unique concept that can be accurately represented and understood by computers through case grammar [16]. The cases are used to label words, which are aligned from local language perspectives. The proposed parser utilizes powerful linguistic tools such as Stanford Parser and universal dependency relations.
(3) Evaluate the proposed MParser through annotator agreement between the expert's case labeling and MParser's outputs. Additionally, 154 non-expert participants investigated judgments of semantic expressiveness.

The rest of this paper is organized as follows. Section 2 compares the proposed approach with related work. Section 3 introduces the general process and methodology of

MParser. Section 4 introduces the activity of human semantic input. Section 5 introduces the activity of sentence computerization. Section 6 and 7 implement and evaluate MParser. Finally, a conclusion is given.

2. Related Work

Semantic representation presents the meaning of sentences, and the process should be reliable and computational [18]. The alternative approaches to semantic representation can be divided into two categories: document representation [1,19] and meaning representation.

Document Representation: Currently, document information exchange mainly has three approaches: (1) Standardization approaches define a semantic document by combining a set of standardized document compositions: for example, EDI-based (http://www.edibasics.com/ediresources/document-standards/), XML-based (ebXML. Available: http://www.ebxml.org) and Web service-based (http://www.edibasics.com/ediresources/document-standards/). The problem with this approach is that documents are only interoperable on representation syntax and templates, and these standards are heterogeneous and incompatible with each other. (2) Ontology modeling [20,21] approaches define a semantic document in a certain domain (e.g., RDF [22], RDFS [23] and OWL (http://www.edibasics.com/ediresources/document-standards/)). They are usually used to solve the problem of semantic interoperability and realize collaboration. Generally, an ontology clearly describes the relationships of entities [18] and can be employed for knowledge representation. However, if computers in different contexts participate in user-computer interaction, it will not be easy to achieve a consistent understanding, because an ontology is domain-dependent, preventing it from being understood between heterogeneous document descriptions. (3) Collaborative approaches [17,24] allow participants from different contexts to construct document terms and solve the cross-domain problem, but the document is constrained by a template and lacks flexibility. One issue is that the user still needs a user template to construct the document.

Current subjects of research on document representations are rule format [25,26], ontology [20,24], XML+Ontology [21], tree/graph [27], and collaborative approach [15,17,28]. First, it is not easy to embed and extract meanings to/from a document automatically. For example, it is not easy for a document written in natural languages to be automatically converted to a machine-processable format (e.g., RuleML [25,26]). Second, constructing semantic documents needs intensive work. For example, [5] proposes a semantic disambiguation solution by using a machine-readable semantic network (e.g., WordNet) as a common knowledge base. However, it is time-consuming and sometimes unnecessary because it also disambiguates unambiguous terms. To acquire accurate semantic concept representation for a document, [20] requires learning a concept border from a particular document collection based on a particular ontology in the same domain. However, there is a heavy workload and enormous data redundancy to construct and store concept borders for different domains. Third, it is not easy to maintain semantic consistency between heterogeneous document systems. For example, [24] claims accurate mapping between different ontologies' entities, and [20] requires the similarity computation between keywords in a received document and equivalent terms in a domain-wide ontology. Both approaches hardly reach a trade-off between low computational demand and semantic interoperability.

In short, these approaches rely on the homogeneity of concept in multilingual text or domain semantics, and sentence-based documents or complex concepts may cause semantic loss among different contexts through the above state-of-art approaches.

Semantic Representation: It defines the annotation to construct syntactic structure such as FrameNet [29] and Semlink [30], but focuses on argument out of other relations. In this context, there are several available semantic representation approaches. For instance, universal networking language (UNL) proposes independent language representation so that sentences inputted in any language can be translated into any other natural language. Abstract meaning representation (AMR) [9] proposes a relatively more straightforward sentence-level semantic parser to cover semantic role broad predictions. AMR manually annotates sentences and utilizes PropBank frames [31] to represent the semantic relationship between words. However, AMR faces difficulties across translation because the syntactical similarity is not suitable cross-linguistically [32]. Therefore, new multilayered solutions such as universal concept cognitive annotation (UCCA) [10] and universal decompositional semantics (UDS) [33] are applied in cross text for semantic annotation and word senses by BabelNet [34] and Open Multilingual Wordnet (http://compling.hss.ntu.edu.sg/omw/). They constructed substantial multilingual semantic nets to achieve universality by connecting resources such as WordNet and Wikipedia. The method adapts linguistic theory to build a manual and multilingual scheme. However, UCCA annotates short sentences (e.g., multiword expressions) where the same multiword or entity is annotated in many different sentences. Groningen Meaning Bank (GMB) is a new solution to integrate language phenomena into a single formalism instead of covering single phenomena in an isolated way. Additionally, universal dependencies (UD) [12] build cross text dependency-based annotations for multilingual sentences.

Most of the semantic representation methods use simple concepts such as UCCA, but some other methods adapt concepts such as WordNet synsets for UNL and PropBank frames for AMR. Furthermore, UNL has its relationships set while AMR uses PropBank relationships. UNL, UCCA, and AMR are fully manual annotated, but GMB produces meaning representations automatically and can be corrected by experts. However, such approaches (e.g., AMR, UCCA, GMB, and UNL) focus on lexical-semantic or multilingual words rather than on sentence semantics and cannot guarantee sentence-based semantic representation to be universal and unambiguous across languages. Most of the proposed semantic representation methods do not consider the morphological and syntactic characteristics of the language in the construction of sentence-level semantic labeling. Contributions made in the semantic representation of any language text will utilize the translated English resources, which may negatively affect the performance of other semantic representation methods. In our research work, MParser propose a universal semantic representation to extract semantic relationships from local language text using local language tools and resources, such as Stanford Parser. In addition, the proposed parser takes into account the syntactic and morphological features of a given sentence. It is worth noting that the proposed MParser model uses various tools, resources, and text features to reduce the negative impact of resource quality on semantic representation. Moreover, MParser achieves a universal representation and semantic consistency across languages.

3. MParser

3.1. Overview

MParser comprises two processes: (1) human semantic input (HSI) and (2) sentence computerization (SC), as shown in Figure 3. First, human semantic input is the process of converting human natural language (HNL$_i$) (here, i indicates English) through an editor typing from CoDic into a sequence of machine-readable sentences SiS_{ci}, which comprises sequentially converting sets of literals to a list of the symbolic signs. The editor (i.e., human user) inputs the HNL$_i$ by SIM from CoDic to constrain sentence creation based on strict rules. Second, sentence computerization (f_c) is a process of converting a sentence SiS_{hi} to a sentence SiS_m that is universally readable and understandable by a computer in MParser, denoted by $f_c := SiS_m \leftarrow SiS_{hi}$. In particular, this involves a sequence of activities: sentence analysis (i.e., parsing a local language sentence based on the local grammatical rules through robust Stanford Parser and universal dependency), case generation (i.e., appending a case on each sign to represent its grammatical functions and properties), and machine representation (i.e., representing a sentence that is computer-readable and -understandable). Thus, sentence $SiS_m \subset MParser$ only readable and understandable by computers can be converted back to a human-readable and -understandable SiS_{hj} (here, j indicates other languages), such that $f_r := SiS_{hj} \leftarrow SiS_m$ to rebuild human-readable sentences based on SiS_m.

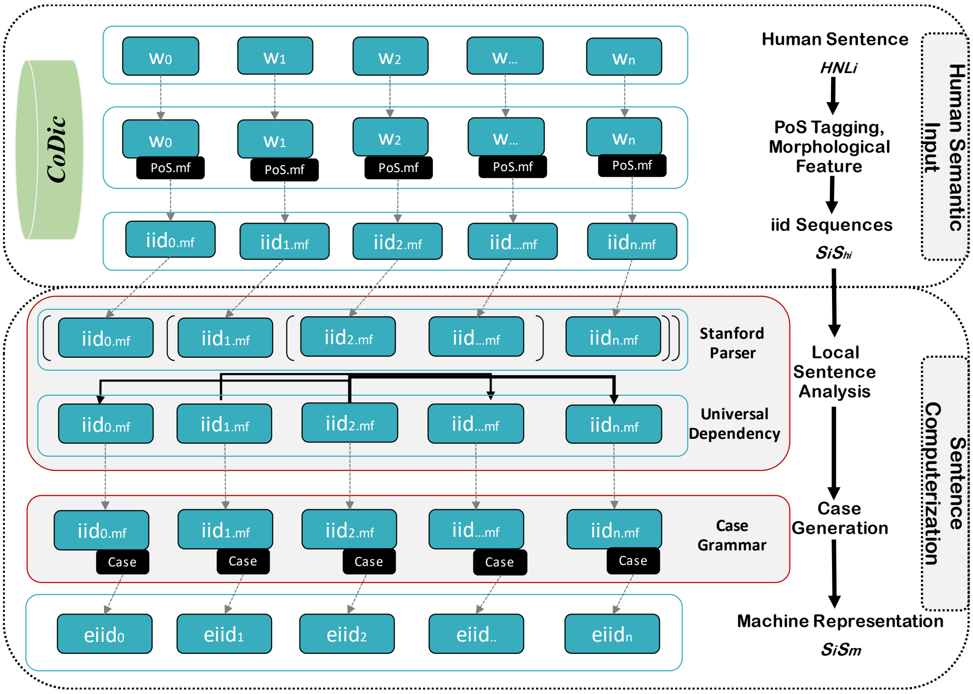


Figure 3. The process of MParser.

3.2. Methodology

The theoretical foundation of MParser comes from the sign description framework (SDF) [26], as shown in Figure 4. It is a language for representing signs in computing systems and is particularly intended to represent the interpreted meanings or ideas of all objects in reality, such as appearing in dictionaries, texts, software, and web pages. A sign: = (*sign*, *denoter*, *reifier*, *denotation*, *connotation*) is modeled by a bi-tree, consisting of three relationships of a *denotation*, a *connotation* and a *reification* between signs.

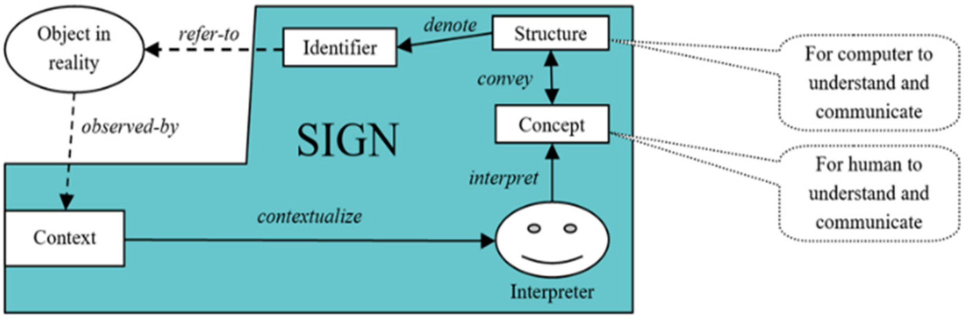


Figure 4. An SDF data model.

A *denotation* is an internal relationship between a sign and its denoter, such that the denoter denotes the properties of a sign. We can understand a denoter as a feature container, containing all features of a sign. For a natural language, these features consist of the form (e.g., iid, term, and pronunciation), sense (i.e., meaning), part of speech (e.g., noun), tense (e.g., past), aspect (e.g., perfective), gender (e.g., male), number (e.g., single), and context (e.g., English). In essence, denotation provides a way to define a sign in the context of a sentence by a set of properties provided by a denoter.

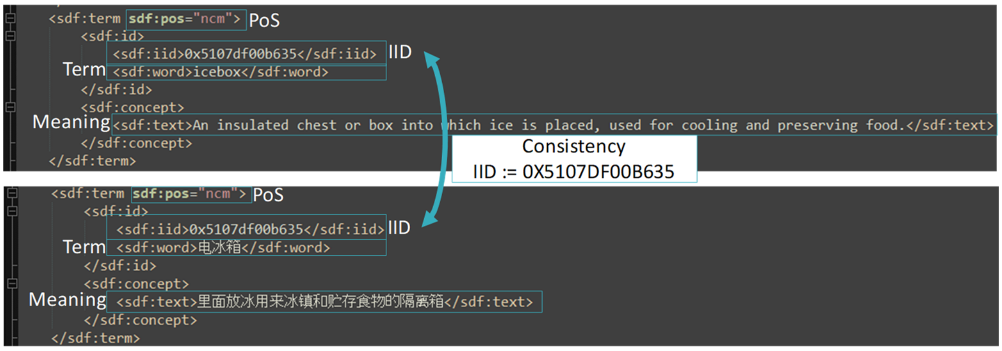
A *connotation* is an external relationship between signs, such that a sign is connoted by a set of signs, which builds a parse tree of a set of signs. For instance, when a set of signs constructs a sentence as a sign in language, it can be parsed through connotation in grammatical cases. For example, we replace the sign of a sentence, and connotation can then parse the sentence sign into many atomic signs.

A *reification* is an instantiation relationship between a reifier (often a particular sign) and a specific denoter (often an abstract sign). For instance, given a denoter denoting the sign of "color", then "white" is a reifier, and between "white" and "color", there is a reification relationship. Or the sign is INT datatype, and 1234 is the reifier.

By generalizing these represented concepts of objects into structured signs, SDF represents all objects in reality, such as objects of abstract and concrete, physical and virtual, and real and fictitious.

CoDic (CoDic http://www.cis.umac.mo/~jzguo/pages/codic/, accessed on 30 August 2021) [17] is a common dictionary and an application of the SDF consisting of 93,546 English words, 20,446 Chinese words, and 190,001 word senses. In CoDic, a concept is a basic element in a sentence and consists of words and phrases. Each concept has already been collaboratively edited without semantic ambiguity. Any dictionary term in CoDic (called a sign) is identified as a unique and internal identifier $iid \in IID$, which is neutral and independent of any natural language and can refer to any term of a natural language. PoS plays a very important role and includes 16 kinds of signs, which are: *Noun(n):= {Common (ncm), Pronoun(npr), Proper Organization (nop), Proper Geography(ngp), Pronoun(npr)}, Verb(v):= {Intransitive (vit), Transitive(vtr), Ditransitive(vdi), Copulative(cop)}, Adjective (adj), Adverb(adv), Preposition (prep), Conjunction(cnj), Interjection (int), Onomatopoeia (ono) and Particle (par)*. For the detailed description of PoS in CoDic, please see Appendix A. Given a simple sign *s = (t, iid) = (icebox, 5107df00b635) = (common noun, "An insulated chest or box into which ice is placed, used for cooling and preserving food.")* as shown in Figure 5. Specifically, the form of the sign is presented as follows:

- **IID**: = **POS**+Y+ID: indicates the universal sign representational form. For instance, iid = 5107df00b635, in which 1 after 5 refers to common noun, 7df refers to year 2015, and 00b635 is ID.
- **Term** indicates literal representational form for a sign, e.g., "*icebox*" is the literal representation of the sign 5107df00b635 in English context.
- **Meaning** is the sense of a sign, e.g., "*An insulated chest or box into which ice is placed, used for cooling and preserving food*" is the sense of 5107df00b635.



```
<sdf:term sdf:pos="ncm">
    <sdf:id>
        <sdf:iid>0x5107df00b635</sdf:iid>
        <sdf:word>icebox</sdf:word>
    </sdf:id>
    <sdf:concept>
        <sdf:text>An insulated chest or box into which ice is placed, used for cooling and preserving food.</sdf:text>
    </sdf:concept>
</sdf:term>
```

```
<sdf:term sdf:pos="ncm">
    <sdf:id>
        <sdf:iid>0x5107df00b635</sdf:iid>
        <sdf:word>电冰箱</sdf:word>
    </sdf:id>
    <sdf:concept>
        <sdf:text>里面放冰用来冰镇和贮存食物的隔离箱</sdf:text>
    </sdf:concept>
</sdf:term>
```

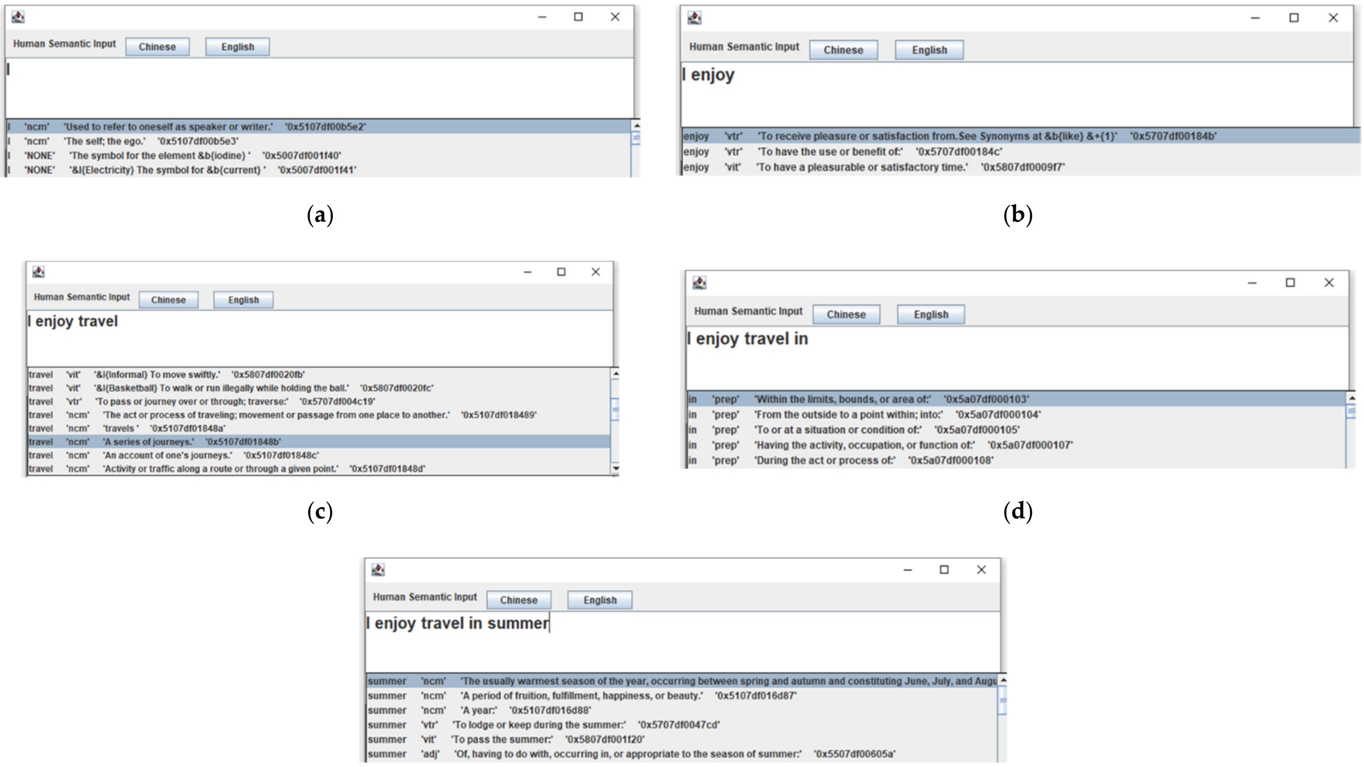
Figure 5. CoDic.

Thus, the meaning of *iid* is: *5107df00b635* = "*icebox*" = "アイスボックス" = "电冰箱" though they are in heterogeneous contexts.

4. Human Semantic Input (HSI)

In human semantic input, the user's initial intention is essential when they try to translate the transmitted concepts into unique semantic representations. If semantics are insufficient for a clear and accurate representation, in that case, the same literal words in users' minds may be different from different contexts between computers and users; it is possible to fail the information interaction because of ad hoc user input. Therefore, HSI tries to solve ad hoc input through a supervised sentence input that cannot casually input the words and phrases in users' minds.

In MParser, all written sentences are constrained by HSI, which is a supervised human-readable sentence via CoDic. We developed an editor to input any term by selecting PoS and the exact meaning, which has a unique identifier (*iid*), to point to the same meaning regardless of contexts. We use a simple English sentence "*I enjoy travel in summer*." to illustrate HSI. First of all, a user types words one by one by selecting terms as shown in Figure 6: the terms "*I*" (*ncm,0* × *5107df00b5e2*), "*enjoy*" (*vtr, 0x5707df00184b*), "*travel*" (*ncm, 0x5107df01848b*), "*in*" (*prep, 0x5a07df000103*), and "*summer*" (*ncm, 0x5107df016d86*).



```
I
```

```
I    'ncm'    'Used to refer to oneself as speaker or writer.'    '0x5107df00b5e2'
I    'ncm'    'The self; the ego.'    '0x5107df00b5e3'
I    'NONE'    'The symbol for the element &b{iodine} '    '0x5007df001f40'
I    'NONE'    '&I{Electricity} The symbol for &b{current} '    '0x5007df001f41'
```

(**a**)

```
I enjoy
```

```
enjoy    'vtr'    'To receive pleasure or satisfaction from.See Synonyms at &b{like} &+{1}'    '0x5707df00184b'
enjoy    'vtr'    'To have the use or benefit of:'    '0x5707df00184c'
enjoy    'vit'    'To have a pleasurable or satisfactory time.'    '0x5807df0009f7'
```

(**b**)

```
I enjoy travel
```

```
travel    'vit'    '&I{Informal} To move swiftly.'    '0x5807df0020fb'
travel    'vit'    '&I{Basketball} To walk or run illegally while holding the ball.'    '0x5807df0020fc'
travel    'vtr'    'To pass or journey over or through; traverse:'    '0x5707df004c19'
travel    'ncm'    'The act or process of traveling; movement or passage from one place to another.'    '0x5107df018489'
travel    'ncm'    'travels '    '0x5107df01848a'
travel    'ncm'    'A series of journeys.'    '0x5107df01848b'
travel    'ncm'    'An account of one's journeys.'    '0x5107df01848c'
travel    'ncm'    'Activity or traffic along a route or through a given point.'    '0x5107df01848d'
```

(**c**)

```
I enjoy travel in
```

```
in    'prep'    'Within the limits, bounds, or area of:'    '0x5a07df000103'
in    'prep'    'From the outside to a point within; into:'    '0x5a07df000104'
in    'prep'    'To or at a situation or condition of:'    '0x5a07df000105'
in    'prep'    'Having the activity, occupation, or function of:'    '0x5a07df000107'
in    'prep'    'During the act or process of:'    '0x5a07df000108'
```

(**d**)

```
I enjoy travel in summer
```

```
summer    'ncm'    'The usually warmest season of the year, occurring between spring and autumn and constituting June, July, and Augu
summer    'ncm'    'A period of fruition, fulfillment, happiness, or beauty.'    '0x5107df016d87'
summer    'ncm'    'A year:'    '0x5107df016d88'
summer    'vtr'    'To lodge or keep during the summer:'    '0x5707df0047cd'
summer    'vit'    'To pass the summer:'    '0x5807df001f20'
summer    'adj'    'Of, having to do with, occurring in, or appropriate to the season of summer:'    '0x5507df00605a'
```

(**e**)

Figure 6. HSI for English sentence "I enjoy travel in summer." (**a**) term "I"; (**b**) term "enjoy"; (**c**) term "travel"; (**d**) term "in"; (**e**) term "summer".

CoDic resources are all on the level of lemmas, and the "term" can be seen as word senses in CoDic, which cannot realize different morphological forms for a word. For instance, in English, the lemma "*enjoy*" yields morphological features: *enjoys*, *enjoyed*, *enjoying*. Thus, the morphological feature (*mf*) for each lemma of CoDic is designed and lists the forms needed in each language. The morphologic feature (*mf*) has the *gender* (*G*) and n*umber* (*N*) features for nouns and the features of *tense* (*T*), *aspect* (*A*) and *voice* (*V*) for verbs. The morphological feature (*mf*) can be different in each language (for details of morphological features, please see Appendix B). The morphological features (*mf*) are parsed according to the local grammar rule because different languages have different morphological phenomena, which are language-dependent for each language. Actually, populating the morphological feature is an engineering effort of its own. In HSI, users manually select the correct feature for each term in the CoDic. Thus, when a user inputs nouns or verbs, he/she needs a second selection for words, including morphologic features (*mf*), as shown in Figure 7.

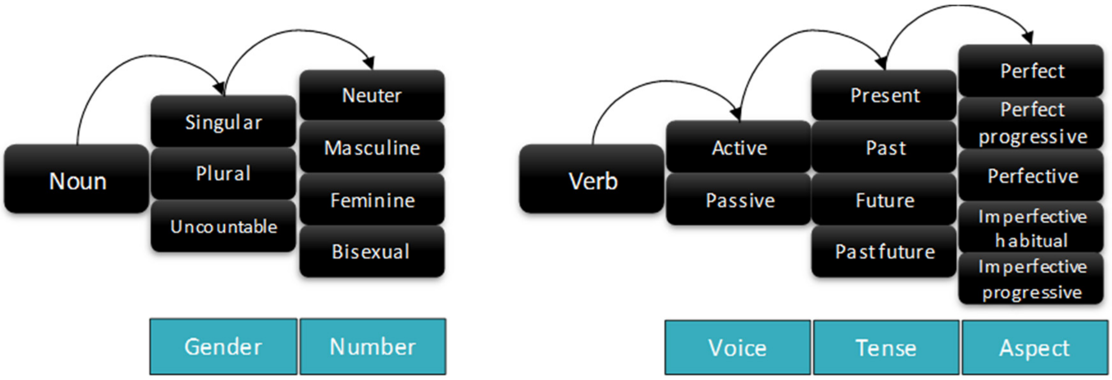


Figure 7. Morphological feature choice in MParser.

Thus, in the example sentence: [('I', 'ncm'), ('enjoy', 'vtr'), ('travel', 'ncm'), ('in', 'prep'), ('summer', 'ncm')], terms "*I*", "*travel*" and "*summer*" choose *singular* and *neuter* (actually, no gender attribute in English, the default is *neuter*), and the hex is 0 for the noun. The term "*enjoy*" chooses *active present imperfective habitual*, and the hex is 03 for the verb. The morphologic feature identification algorithm is presented in Table 1. Table 2 shows the tenses of a sentence in English and *HSI* through a basic example ("*she go home*"). Following interesting observations from Table 2, it can be observed that helping verbs (***Bold font***) have been removed during the human sentence input for all tenses of verbs. MParser uses only the root form of the verb. These helping verbs, such as "*is, am, be, being, has, had*", are represented by a hex of morphologic feature (*mf*). Thus, the human input sentence is universal for all languages.

Table 1. Morphologic feature identification algorithm.

```
Function (Input words)
Input
String ← Input word
if (String.pos= "ncm" or "npp" or "ntp") then
Gender(G): = n | m | f | b                         /* Select noun's gender */
Number(N): = s | p | u                             /* Select noun's number */
return ← noun morphological feature (mf)
if (String.pos= "vtr" or "vid" or "vit") then
Tense(T) = present | past | future | past future   /* Select a verb's tense */
Aspect(A) = f | g | w | h | p                      /* Select verb's aspect */
Voice(V): = active | passive                       /* Select verb's voice */
return ← verb morphological feature (mf)
```

Table 2. Human semantic input of tenses in English.

Tense of Sentence	English Sentence	HSI
Past perfect	She ***had gone*** home.	She.$_{mf}$ ***go***.$_{mf}$ home. (***mf*** refers to defined Hex)
Future perfect	She ***will have gone*** home.	
Present perfect continuous	She ***has been going*** home.	
Past perfect continuous	She ***had been going*** home.	
Future perfect continuous	She ***will have been going*** home.	
Simple present	She ***goes*** home.	
Simple past	She ***went*** home.	
Simple future	She ***will go*** home.	
Present continuous	She ***is going*** home.	
Past continuous	She ***was going*** home.	
Future continuous	She ***will be going*** home.	

HSI converted a sequence of human-readable literals HNL$_i$ to a sequence of signs SiS_{ci} that a computer program can understand without semantic ambiguity. Formally, the concepts are defined below.

Definition 1. *(Human Simple Sentence "SiS_{hi}"): Given a well-formed sequence of literal words $(w_1, \ldots, w_k, \ldots, w_m)$ in HNL$_i$, input by a human user, in i context (i is English user), then:*

$$SiS_{hi} := (w_{0.mf}, w_{1.mf}, \ldots, w_{k.mf}, \ldots, w_{m.mf}) = \sum_{k=0}^{m} w_k \tag{1}$$

where w_0 is an automatically generated leading word signifying the beginning of a sentence, $0 < k \leq m$ is the word sequence number of the word w_k in SiS_{hi}.

Definition 2. *(Computer Sentence "SiS_{ci}"): Given SiS_{hi}: = $(w_{0.mf}, w_{1.mf}, \ldots, w_{k.mf}, \ldots, w_{m.mf})$ $=\sum_{k=0}^{m} w_k$, then SiS_{hi} is generated into iid sentence, called computer sentence "SiS_{ci}", such that:*

$$SiS_{ci} := (iid_{0.mf}, iid_{1.mf}, \ldots, iid_{k.mf}, \ldots, iid_{m.mf}) = \sum_{k=0}^{m} iid_k \tag{2}$$

where iid_0 is an automatically generated leading word signifying the beginning of a sentence, $0 < k \leq m$ is the iid sequence number of the iid_k in SiS_{ci}. For the result of human semantic input, SiS_{ci} is a supervised computer-readable sentence.

5. Sentence Computerization (SC)

Sentence computerization (SC) transforms a human sentence into a computer sentence. It consists of three main activities: (1) Analyze the constituency structure and universal dependency relationship from the outputting words in the HSI step. (2) Adapt the Stanford parser tool to extract potential relationships between outputting words. (3) Apply predefined case grammar rules to label semantic roles outputting words and generate a universal sentence. The activities involve local sentence analysis described in Section 5.1, case generation described in Section 5.2, and machine representation described in Section 5.3.

5.1. Local Sentence Analysis

Each word is tagged into the PoS and morphological feature (*mf*) in the sentence from the step of HSI. Local sentence analysis identifies the relationship between different words that constitute the English sentence. We adapted the Stanford parser tool [35], which provides full syntactic analysis, minimally a constituency (bracketed sentences) parse of local English sentences between different PoS. Constituency parse describes what the constituents are and how the words are put together. For instance, a sentence: "*the quick brown fox jumps over the lazy dog*" can transform into:

```
[('ROOT', [('NP', [('NP', [('DT', ['the']), ('JJ', ['quick']), ('JJ', ['brown']), ('NN', ['fox'])]), ('NP',
[('NP', [('NNS', ['jumps'])]), ('PP', [('IN', ['over']), ('NP', [('DT', ['the']), ('JJ', ['lazy']), ('NN',
['dog'])])])])])])]
```

The bracketed sentence represents grammatical functions, such as NP, VP, and PP based on English grammar. However, Stanford parser adapts the Penn PoS tagger rather than the CoDic PoS tagger, such that it is impossible to parse the sentence directly. Thus, we built a mapping between Penn PoS tagger and our CoDic PoS tagger, and the PoS mapping algorithm is shown in Table 3. In particular, particle words, which are only for the local English language, have no common *iid* to map other languages and will not appear in the final machine representation.

Table 3. PoS mapping algorithm.

```
if (isCoDicPos)
if (CoDicpos =par and iid= "xxx") {
Stanfordpos = "xxx";
} else if (CoDicpos = noun or verb and mf= "xxx") {
Stanfordpos = "xxx" or insert words and Stanfordpos = "xxx";
} else if (CoDicpos = other PoS) {
Stanfordpos = "xxx";
} else
print ="error"
end if; }
```

Stanford parser presents and parses a word's relationship by a pure constituency, but ignores their semantic role. For example, SVO (subject-verb-object) structure is presented as S → NP VP NP by the Stanford parser, and it is impossible to parse subject, object, and other semantic roles in a sentence. Nivre et al. [12] proposed a universal dependency (UD) that uses dependency labels and PoS tags to parse sentences for different languages. The UD annotation defines a classification of around 40 relations as the universal dependency label sets (https://universaldependencies.org/#language-tagset, accessed on 30 August 2021), such as *nsubj*: nominal subject, *amod*: adjectival modifier. Thus, when the UD appeared, it immediately became interesting to see its relationship with the Stanford parser. For instance, the sentence "*the quick brown fox jumps over the lazy dog*" can transform into:

```
[[((u'jumps', u'VBZ'), u'nsubj', (u'fox', u'NN')), ((u'fox', u'NN'), u'det', (u'The', u'DT')), ((u'fox',
u'NN'), u'amod', (u'quick', u'JJ')), ((u'fox', u'NN'), u'amod', (u'brown', u'JJ')), ((u'jumps',
u'VBZ'), u'nmod', (u'dog', u'NN')), ((u'dog', u'NN'), u'case', (u'over', u'IN')), ((u'dog', u'NN'),
u'det', (u'the', u'DT')), ((u'dog', u'NN'), u'amod', (u'lazy', u'JJ'))]]
```

Finally, through Stanford Parser and UD, the local English sentence becomes a segmented sentence with dependency relationships for each word, as shown in Definition 3.

Definition 3. (*Segmented Simple Sentence "*$SiS^q{}_{ci}$*"*): *Given* SiS_{ci}: = $(iid_{0.mf}, iid_{1.mf}, \ldots, iid_{k.mf}, \ldots, iid_{m.mf}) = \sum_{k=0}^{m} iid_k$, *then* SiS_{ci} *is segmented into* $q + 1$ *subsequences, called q-subsequences* $SiS^q{}_{ci}$. *Each subsequence has p number of iid, such that:*

$$Segment:\ (iid_{0.mf},\ (iid_{1.mf},\ \ldots,\ iid_{p.mf})_1,\ \ldots,\ (iid_{1.mf},\ \ldots,\ iid_{p.mf})_i,\ \ldots,\ (iid_{1.mf},\ \ldots,\ iid_{p.mf})_q \leftarrow \sum_{k=0}^{m} iid_k \tag{3}$$

$$SiS^q{}_{ci} = Segment\ (iid_{0.mf}, iid_{1.mf}, \ldots, iid_{k.mf}, \ldots, iid_{m.mf}) = (iid_{0.mf}, (iid_{1.mf}, \ldots, iid_{p.mf})_1, \ldots, (iid_{1.mf}, \ldots, iid_{p.mf})_i, \ldots, (iid_{1.mf}, \ldots, iid_{p.mf})_q) \tag{4}$$

$$SiS^q{}_{ci} = iid \sum_{i=1}^{q} \Big(\sum_{j=1}^{p} iid_j\Big)_i \tag{5}$$

where the length of i-th subsequence $(iid_{1.mf}, \ldots, iid_{p.mf})_i = \sum_{j=1}^{p} iid_{j.mf}$ *is* p $(1 \leq p \in \mathbb{N})$.

5.2. Case Generation

MParser grammar is a set of machine natural language grammars such as universal grammar (UG) and case grammar (CG), originating from Fillmore's case study [36,37]. MParser grammar specifies various sequences of signs, forming a general natural language commonly read and understood both by humans and computer systems. It consists of morphological features (intrinsic) (discussed in HSI) and case grammar components (extrinsic). The morphological component varies from one language to another regarding the sets of morphological features, which are inflection forms themselves, but uses common naming conventions. Each case label either presents a syntactic, semantic, or computational function or marks a grammatical function in general and abstracts a particular grammatical phenomenon pertaining to a group of words, phrases, sentences, or others that appeared in natural languages.

In our previous work [16], we proposed a case grammar representing a universal and deep case (or semantic roles) that reflects in a sentence as the central means of explaining both the syntactic structure as well as the meaning of sentences. The case grammar component displays a common representation of syntactic structures and structural words and can be used as a resource for language processing tasks, such as translation, multilingual generation, and machine inference. The novel available cases are defined as follows:

- **Nominative Case** (NOM): denotes a semantic category of entities that initiate actions, trigger events, or give states. Nominative case often associates with the agentive properties of volition, sentience, instigation, and motion.
- **Predicative case** (PRE): denotes a semantic category of process in terms of action, event, or state. The process starts from a sign in the semantic category of the nominative.
- **Accusative case** (ACC): denotes a semantic class of patients who are the participants affected by the semantic class of agents marked by agentive case, which is the direct object of an agentive action.
- **Dative case** (DAT): denotes a semantic class of indirect participants relevant to an action or event. The objective participant marked by dative is called recipient or beneficiary of an action.
- **Genitive case** (GEN): denotes a semantic category of attributes that belong to things. It describes an attributive relationship of one thing to another thing.
- **Linking case** (LIN): denotes the thing that corresponds to the theme of thematic nominatives, such as attributes, classification, or identification of a theme.
- **Adverbial case** (ADV): denotes a semantic category of constraints belonging to predicative signs (i.e., a verb). It corresponds to the adverbial syntactic case.
- **Complementary case** (COM): denotes additional attributes of an entity, an action, an event, or a state, such as means, location, movement, time, causality, extent, and range. Under the PRE structure, COM is shown in COMv form. Under the NOM/ACC/DAT structure, COM is shown in COMn form. For other situations, it just shows COM form.

In this paper, cases are labels or tags that mark signs' syntactic, semantic, and computational functions in the marked forms such as marked words, phrases, and sentences within a natural language's text. For example, in the sentence "*earth moves around sun*", the behavior "*move*" is performed by the entity "*earth*" and the behavioral method is "*around the sun*". A case is used to label the functionality of a word or a phrase in the sentence, such as "*NOM.earth PRE.moves COMv.around NOM.sun*". The universal case grammar provides a common grammar transformable to the grammar of any existing natural language.

Tree Generation

Case generation converts a sequence of single concepts (i.e., atomic signs) into complex concepts (i.e., a compound sign), that is self-described. MParser builds a sentence-based case concept associated with an *iid* defining how an *iid* grammatically functions and combines with other *iids* by the case grammar. It does not need to consider the order of the sentence, which is a bag of concepts. The key of case generation to a sign lies in two facts:

(1) There is a known PoS already associated with the term (HSI);
(2) The term has a clear grammatical relationship with other terms in a sentence (local sentence analysis).

A sentence is defined as a sequence of signs, each marked with a functionality label defined as a case. Each sign in a sentence can describe its case grammar relationship with other signs; that is the compound sign, called *SignX*, which is

Sentence :: = $\mathbf{SignX_1 \ldots SignX_i \ldots SignX_n}$

$\mathbf{SignX = IID.C_1 \ldots IID.C_i \ldots IID.C_n}$

For example:

[(‘**NOM**’, [(‘**GEN**’, [‘the’]), (‘**GEN**’, [‘quick’]), (‘**GEN**’, [‘brown’]), (‘**NOM**’, [‘fox’])]), (‘**PRE**’,[(‘PRE’, [‘jump’])]), (‘**COMv**’, [(‘**COMv**’, [‘over’]), (‘**ACC**’, [(‘**GEN**’, [‘the’]), (‘**GEN**’, [‘lazy’]), (‘**ACC**’, [‘dog’])])])])]

NOM and ACC cases are appending for nouns such as the words “*fox*” and “*dog*”, PRE case is appending for verbs such as the word “*jump*”. Thus, the case generation (fox_NOM (jump_PRE)) yielding the English “*fox jump*” can be turned into Chinese by just changing the lexical item: (狐狸 _NOM (跳_PRE)) yielding “狐狸跳”. The case is appending NOM and PRE to form correct sentences in both languages. Meanwhile, the morphology feature (*mf*) builds inflection features for nouns and verbs in both languages.

Based on sign theory [28], every concept (e.g., *fox, jump)* is a meaning group, which appends a single case (e.g., NOM, PRE) to modify a larger meaning group in a tree of concepts. If each concept in a sequence is unique, then the sequence is also unique. The tree is defined as $T = (N, E)$, where N indicates a group of nodes, and E indicates a group of edges, where $E \subseteq N \times N$. The path in a tree is a sequence of nodes $n_1, n_2, \ldots, n_{k\text{-}1}, n_k$, where each pair (n_1, n_2) has $e(n_1, n_2) \in E$. A cycle is a path $n_1, n_2, \ldots, n_{k\text{-}1}, n_k$ $(k > 2)$ that consists of distinct nodes, except $n_1 = n_k$. In our tree generation, we present a sentence in a tree-based *SignX* representation as T_{SignX}. Nodes N contains two main types: *iid* node N_{iid} and case node N_c. Formally, the node-set is:

$$N = \{N_{iid}, N_c \mid iid \in IID, c \in C\}$$

where IID is a group of all words’ *iids* in the sentence, and each *iid* in the local sentence is represented as a node in the T_{SignX}. C is a group of predefined case concepts, including NOM, PRE, and so on. Additionally, edges E link any two nodes in a tree, where:

$$E \subseteq \{n_f, n_c \mid f, c \in N\}$$

An important principle is designed in sentence construction, which is the *father–child* relationship. Each edge $e(n_f, n_c)$, where $n_f, n_c \in N$ is connected with a father–child relationship that represents the structure relationship between its two connected nodes $\mathrm{n_f}$ and $\mathrm{n_c}$ —whether a father code (*f*) is modified by another child node (*c*) or not, while the father node proceeds. A *father node* is a key sentence constituent. Differently, a *child node* is always dependent and belongs to a father node. This correspondence can be illustrated in Figure 8. Applying this principle, we can always construct a sequence of sentences in different order of atomic concepts but still ensure structural equivalence.

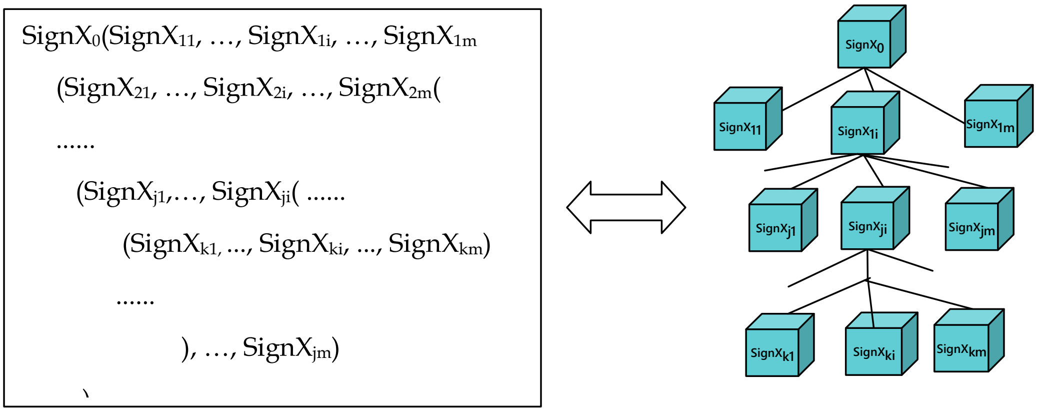


Figure 8. MParser SignX Tree T_{SignX}.

The case generation is converted into T_{SignX} using the tree generation algorithm. T_{SignX} provides a phrase-based structure, such as SVO, OVS sequences, and case labeling, and is a non-redundant representation. The T_{SignX} Tree algorithm is derived as follows:

1. *Linearize input to a term sequence S.*
2. *Connect each term in S to its smallest subtree in* T_{SignX}.

3. *Append one case in each node of T_{SignX} based on case grammar rules.*
4. *Parse the universal dependency labels at each branching node N of the T_{SignX}.*
5. *Find the dependency relationship in the node of each word:*
    a. *If exist corresponding dependency label, then replace the current case using dependency mapping rules;*
    b. *If no dependency relationship, keep the current case.*

The proposed $\mathrm{T_{SignX}}$ model can represent different sentences with the same tree if they have the same semantics. Because the order of the words does not affect its representation, it reduces the influence of language, which has the property of flexible order. A sentence becomes a case sentence through the case generation, which appends a case concept for each word, as shown in Definition 4.

Definition 4. (*Case Sentence "SiS_c"*): *Given* $SiS^q{}_{ci} = (iid_{0.mf}, (iid_{1.mf}, \ldots, iid_{p.mf})_1, \ldots, (iid_{1.mf}, \ldots, iid_{p.mf})_i, \ldots, (iid_{1.mf}, \ldots, iid_{p.mf})_q)$, *then* $SiS^q{}_{ci}$ *is appended cases for the sentence, called Case Sentence* SiS_c. *Each word has one case, such that*:

$$SiS_c = (iid_{0.mf.\mathrm{C}}, (iid_{1.mf.\mathrm{C}}, \ldots, iid_{p.mf.\mathrm{C}})_1, \ldots, (iid_{1.mf.\mathrm{C}}, \ldots, iid_{p.mf.\mathrm{C}})_i, \ldots, (iid_{1.mf.\ .\mathrm{C}}, \ldots, iid_{p.mf.\ .\mathrm{C}})_q) \tag{6}$$

where the length of i-th subsequence $(iid_{1.mf.\mathrm{C}}, \ldots, iid_{p.mf.\mathrm{C}})_i = \sum_{j=1}^{p} iid_j$*is p* $(1 \leq p \in \mathbb{N})$, *and C is appended case.*

5.3. Machine Representation

After attaching a case to a word, a machine universal language representation shows a computer-readable and -understandable sentence without huge extra data to process it.

Definition 5. (*Computer-Understandable Simple Sentence "SiS_m"*): *Given a sign-based sentence* $SiS_c = (iid_{0.mf.\mathrm{C}}, (iid_{1.mf.\mathrm{C}}, \ldots, iid_{p.mf.\mathrm{C}})_1, \ldots, (iid_{1.mf.\mathrm{C}}, \ldots, iid_{p.mf.\mathrm{C}})_i, \ldots, (iid_{1.mf.\ \mathrm{C}}, \ldots, iid_{p.mf.\mathrm{C}})_q)$, SiS_m *is a set of extend iid, called eiid, such that*:

$$SiS_m = (S, eiid_1, \ldots, eiid_k, \ldots, eiid_n) \tag{7}$$

where an extended iid (eiid):
eiid : = Term.iid.mf.Case.F.C

(***term*** and "***iid***" refers to a sense in CoDic, "*PoS*" is already defined in *iid*, ***mf*** refers to morphological feature, ***F*** is the index of the higher level father sign in MParser tree, and index of the lower level child node "***C***" in MParser tree). Additionally, the machine representation referring to PoS is defined:

(1) If *PoS* is noun, *eiid* = Term.IID.mf.Case.F.C, in which mf refers to the morphological feature of the noun.
(2) If *PoS* is verb, *eiid* = Term.IID.mf.Case.F.C, in which mf refers to the morphological feature of the verb.
(3) If *PoS* is adjective | adverb | prep | conjunction | . . . , *eiid* = Term. IID.Case.F.C;
(4) If *PoS* is a particle, delete the node. (Unlike a noun or a verb, a particle is localized and meaningless in a sense for other languages, only confers a local grammatical meaning, and it is not possible to map it to other languages.)

Finally, through the machine representation activity, a sentence becomes a bag of semantic concepts without considering the sequence of the sentence through term index and can be self-described for understanding by computers.

6. Implementation

The MParser is implemented in Python and Java under macOS version 11.0.1 system, and runs under python 3.7 and JDK 1.8. CoDic is represented in XML format for

English and Chinese. In addition, Stanford Parser and universal dependency APIs are called by MParser. In the implementation, several sentences are processed and analyzed to describe how to represent a sentence and maintain semantic consistency from English sentences. In MParser, the user first types words one by one by selecting terms and additional morphological features such as *"I enjoy travel in summer"* in the HSI step. By calling the constructInfo function in MParser, the sentence is generated into:

constructInfo [(‘I’, ‘ncm’, ‘0x5107df00b5e2′, ‘0′), (‘enjoy’, ‘vtr’, ‘0x5707df00184b’, ‘03′), (‘travel’, ‘ncm’, ‘0x5107df032b53′, ‘0′), (‘in’, ‘prep’, ‘0x5a07df000103′, “), (‘summer’, ‘ncm’, ‘0x5107df016d86′, ‘0′)]

The step of ***constructInfo*** constructs the information for each typed word in HSI, such as term, PoS, iid and morphological features for nouns and verbs. Next, the sentence goes to the *Sentence Computerization* step, which is an automated analysis without user participation. ***ParserList*** function of MParser calls the Stanford Parser API to construct a phrase-based structure sentence based on predefined PoS tagger mapping rules between Stanford Parser and CoDic:

paserList [‘(ROOT’, ‘ (S’, ‘ ((ncm I))’, ‘ ((vtr enjoy)’, ‘ ((ncm travel))’, ‘ ((prep in)’, ‘ ((ncm summer))))’, ‘ (. .)))’]

Meanwhile, a universal dependency is parsed by calling the ***dependency_parse*** function in MParser, and finding each word dependency relationship by the ***everyWordDep*** function in the sentence:

dependency_parse [(‘ROOT’, 0, 2), (‘nsubj’, 2, 1), (‘dobj’, 2, 3), (‘case’, 5, 4), (‘nmod’, 2, 5), (‘punct’, 2, 6)]
everyWordDep {‘I’: ‘nsubj’, ‘travel’: ‘dobj’, ‘in’: ‘case’, ‘summer’: ‘nmod’, ‘.’: ‘punct’}

After the local sentence analysis, the English sentence includes phrase-based structure and dependency semantic roles. Then, the sentence is analyzed based on case rules:

(1) This sentence begins from an S, which is a *declarative* sentence.
(2) The noun (*ncm*) *we* is case *NOM* [*I-ncm-NOM*] if it is before a verb such at *ncm-NOM* ← *vtr-PRE* (except GEN, ADV and others).
(3) The verb (*v*) *enjoy* is case *PRE* [*enjoy-vtr-P*] where *transitive verb* (*vtr*) follows only one *noun* structure, such that *vtr-PRE* → *vtr-PRE noun* [*supplementary*: *vit-PRE*; *vdi-PRE* → *vdi-PRE* $noun_1$ $noun_2$.]
(4) The noun (*ncm*) *travel* is case *ACC* [*travel-ncm-ACC*] if it is before a vtr verb such at *vtr-PRE* ← *ncm-ACC*.
(5) The preposition (*prep*) *in* is case *COMv* [*in-prep-COMv*] under PRE structure.
(6) The noun (*ncm*) *summer* is case *NOM* [*summer-ncm-NOM*], such that *in-prep-COMv*← *summer-ncm-NOM.*

We applied our case grammar rules to generate the MParser tree. The tree visualizations are presented by NLTK API (NLTK API: http://nltk.org). Figure 9 shows the structure and tree screenshot from MParser.

Finally, the machine representation generated a universal sentence:

S.0.0(I.0x5107df00b5e2.0.NOM.0.1(enjoy.0x5707df00184b.03.PRE.0.1(travel.0x5107df01848b.0.ACC.1.2(in.0x5a07df000103.COMv.1.2(summer.0x5107df016d86.0.NOM.2.3)))))

The universal sentence presents a sequence of extracted meaningful concepts related to each other using cases and syntactical relationships. The sentence also can map into Chinese words for Chinese CoDic via unique *iid*. An illustration shows a transformation from local English HNL (*i*) to a universal sentence, then Chinese HNL (*j*) in Table 4.

```
(ROOT
  (S
    (NOM (ncm I))
    (PRE (vtr enjoy)
      (ACC (ncm travel))
      (COMv (prep in)
        (NOM (ncm summer))))
  (. .)))
```

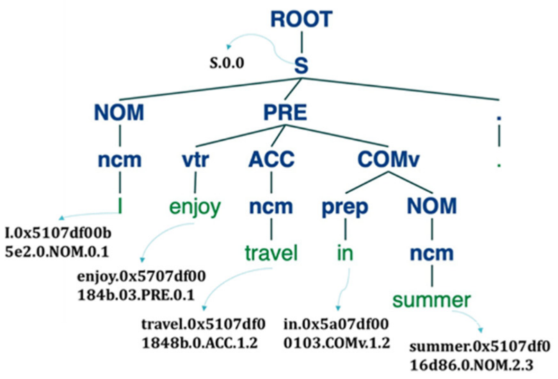


Figure 9. Structure and MParser tree for English sentence "I enjoy travel in summer".

Table 4. Transformation from English to Chinese in MParser.

I	Enjoy	Travel	In	Summer	English(HNL$_i$)
0x5107df00b5e2	*0x5707df00184b*	*0x5107df01848b*	*0x5a07df000103*	*0x5107df016d86*	***iid***
0x5107df00b5e 2.0.NOM.0.1	***0x5707df00184 b.03.PRE.0.1***	***0x5107df01848 b.0.ACC.1.2***	***0x5a07df00010 3.COMv.1.2***	***0x5107df016d 86.0.NOM.2.3***	***eiid***
我	享受	旅程	在	夏天	***Chinese (HNL$_j$)***

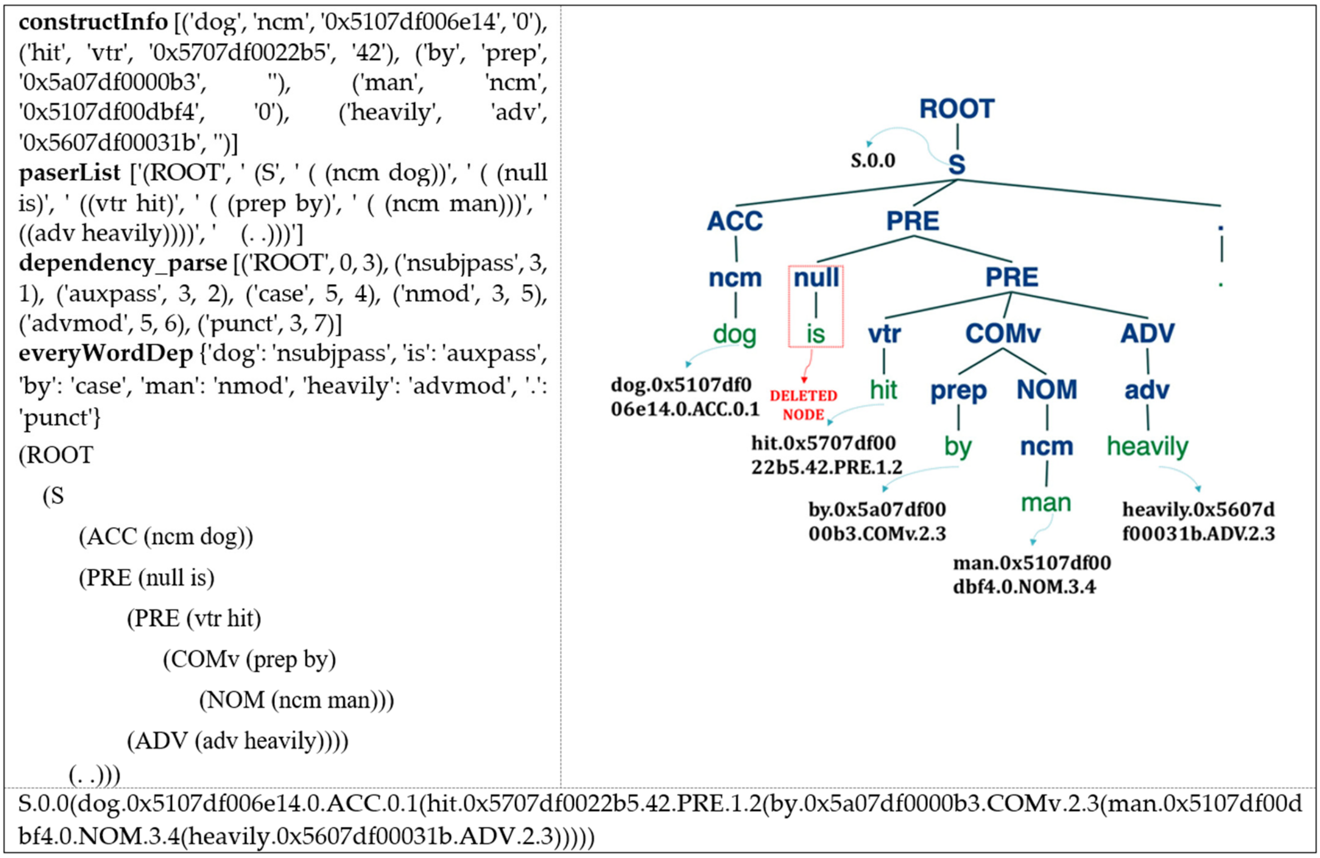
First, the English sentence is converted to machine-readable *iid* sequences from English CoDic. Then, through case generation and machine representation steps, the English computer-understandable sentence is converted into a universal computer-readable and -understandable *eiid* sentence that is a bag of unique concepts. Finally, the *eiid* sentence can be translated into another language such as Chinese based on local rules. MParser ensures that any sentence in an HNL$_i$ can be transformed into HNL$_j$ without any semantic loss.

We also tested a passive sentence in English to illustrate the difference between NOM and ACC from the semantic role, which is "*dog is hit by man heavily*.", as shown in Figure 10.

From the example, we found that "*dog*" is ACC, and "*man*" is NOM in a passive sentence, and they meet the standard semantic role for a passive sentence. The PoS of the word "*is*" is null since it is inserted during local sentence analysis, not from CoDic, and it does not appear in final machine representation. We illustrate from tenses of three English sentences, shown in Table 5.

Table 5. Tense test of MParser in English.

HSI	English Sentence Analysis	Machine Representation
I.0 go.00 home.0 (*I **have gone** home.*)	**I**/NN (have/VBP) **go**/VBN **home**/NN.	S.0.0(I.0x5107df00b5e2.0.NOM.0.1(go.0x5707df00203d.00.PRE.1.2(home.0x5107df00afcc.0.ACC.2.3)))
I.0 go.40 home.0. (*I **have been going** home.*)	**I**/NN (have/VBP been/VBN) **go**/VBN **home**/NN.	S.0.0(I.0x5107df00b5e2.0.NOM.0.1(go.0x5707df00203d.40.PRE.1.2(home.0x5107df00afcc.0.ACC.2.3)))
I.0 go.04 home.0 (*I **am going** home.*)	**I**/NN (is/VBP) **go**/VBG **home**/NN.	S.0.0(I.0x5107df00b5e2.0.NOM.0.1(go.0x5707df00203d.04.PRE.1.2(home.0x5107df00afcc.0.ACC.2.3)))



```
constructInfo [('dog', 'ncm', '0x5107df006e14', '0'), ('hit', 'vtr', '0x5707df0022b5', '42'), ('by', 'prep', '0x5a07df0000b3', ''), ('man', 'ncm', '0x5107df00dbf4', '0'), ('heavily', 'adv', '0x5607df00031b', '')]
paserList ['(ROOT', ' (S', ' ( (ncm dog))', ' ( (null is)', ' ((vtr hit)', ' ( (prep by)', ' ( (ncm man)))', ' ((adv heavily))))', '    (. .)))']
dependency_parse [('ROOT', 0, 3), ('nsubjpass', 3, 1), ('auxpass', 3, 2), ('case', 5, 4), ('nmod', 3, 5), ('advmod', 5, 6), ('punct', 3, 7)]
everyWordDep {'dog': 'nsubjpass', 'is': 'auxpass', 'by': 'case', 'man': 'nmod', 'heavily': 'advmod', '.': 'punct'}
(ROOT
  (S
      (ACC (ncm dog))
      (PRE (null is)
          (PRE (vtr hit)
              (COMv (prep by)
                  (NOM (ncm man)))
          (ADV (adv heavily))))
    (. .)))
S.0.0(dog.0x5107df006e14.0.ACC.0.1(hit.0x5707df0022b5.42.PRE.1.2(by.0x5a07df0000b3.COMv.2.3(man.0x5107df00dbf4.0.NOM.3.4(heavily.0x5607df00031b.ADV.2.3)))))
```

Figure 10. MParser for English sentence "dog is hit by man heavily".

7. Evaluation

Human manual evaluation is the crucial and ultimate criterion for validating semantic case labeling given our definition of semantics as a meaning as it is understood by a language speaker [38]. In this research, MParser was evaluated using intrinsic and extrinsic evaluation. Intrinsic evaluation (reader-focused) aimed to evaluate the properties of MParser output by asking participants about the degree of semantic expressiveness of the output in a questionnaire. The extrinsic (expert-focused) evaluation aimed to evaluate the agreement rate of case labeling between MParser outputs and experts.

7.1. Dataset

In our experiment, we randomly selected 100 sentences from a dataset (https://www.kaggle.com/c/billion-word-imputation/data, accessed on 30 August 2021)[39],which is a large corpus of English language sentences, to manually input each word for each sentence in MParser, and finally output 75 retained sentences (*N* = *75*) (please see Appendix C for 75 automatic sentence outputs from MParser) because we removed some unrecognizable words from CoDic and unparseable sentences. Taking into account the validity of the questionnaire, we divided the 75 sentences (*N* = *75*) into 5 groups (each with 15 sentences (*N* = *15*)), which were Group A, B, C, D, and E. Table 6 shows our test dataset, which were 50 short sentences with less than 8 words and 25 long sentences with more than 8 words.

Table 6. Number of MParser outputs.

Sentence Type	Number of Words					Total
	Group A (N = 15)	*Group B (N = 15)*	*Group C (N = 15)*	*Group D (N = 15)*	*Group E (N = 15)*	
Short Sentence *(length <= 8, N = 50)*	*46 (N = 10)*	*59 (N = 10)*	*50 (N = 10)*	*54 (N = 10)*	*60 (N = 10)*	***269***
Long Sentence *(length > 8, N = 25)*	*45 (N = 5)*	*44 (N = 5)*	*45 (N = 5)*	*51 (N = 5)*	*51 (N = 5)*	***236***
Total	***91***	***103***	***95***	***105***	***111***	***505***

7.2. Experiment Settings

Intrinsic: An intrinsic (reader-focused) design usually requires a larger sample of (non-expert) participants. In order to investigate judgments of the semantic expressiveness of MParser outputs, we used 154 valid participants to judge the degree of semantic expressiveness for 75 generated sentences through a questionnaire [40]. The semantic expressiveness criterion was: *"how clear is it to understand what is being described"* or *"how clear it would be to identify the case label from the description"*. We adapted the 5-point Likert scale of semantic expressiveness, as follows:

1. Very unclear 2. Unclear 3. Acceptable 4. Clear 5. Perfectly clear

Readers were from cohorts of undergraduate and graduate students pursuing English-related degrees. Before completing the questionnaire, they were expected to understand the attributes of each MNL case label; each group required at least 25 readers to complete.

Extrinsic: In the semantic case labeling evaluation, ideally, by asking the annotator to make some semantic prediction or annotation based on pre-specified criteria and comparing it with the case extracted from the proposed method, the degree of agreement between the proposed method and the expert's annotation could be determined. Thus, a small number of expert annotators were recruited to label cases of the MParser [41]. We used three experts, two Ph.D. students majoring in an English linguistics-related research area, and one university English lecturer to label the 75 sentences. Before labeling, they were required to fully understand the description of attributes of each MNL case through learning case grammars. Additionally, five groups of sentences (each with 15 sentences) required three experts to be completed. This meant that every expert needed to label 75 sentences. To facilitate labeling by the experts and compare it to test data of MParser, we split each word of each sentence, and the experts only needed to select the case for each word. We measured pairwise agreement of extrinsic evaluation among experts and MParser outputs using the kappa coefficient (κ), which is widely used in computational linguistics for measuring agreement in category judgments [42]. It is defined as

$$K = \frac{P(A) - P(E)}{1 - P(E)} \tag{8}$$

where *P(A)* is the observed agreement rate of case labeling for one annotator such as expert 1, and *P(E)* is the expected agreement rate for another expert 2. The simple Kappa coefficient adapts binary classification. Thus, case labeling was achieved by a binary classification where each case has *Yes* (1) or *No* (0). For example, a NOM case label might be NOM case (1) or non-NOM case (0) in one word for annotators. We calculated κ from two aspects: inter-annotator agreement and intra-annotator agreement. Inter-annotator agreement was calculated for 75 sentences, which were annotated by two experts. Intra-annotator agreement followed a similar process but was calculated for 75 sentences that were annotated between expert and MParser outputs. The interpretation standard of Kappa varied (−1 to 1) according to Landis and Koch [43]: ***<0 Poor | 0—0.2 Slight | 0.2—0.4 Fair | 0.4—0.6 Moderate | 0.6—0.8 Substantial | 0.8–1 Perfect.***

7.3. Results

From Table 7 and Figure 11, the judgments of semantic expressiveness indicated that MParser had better results since *Clear* and *Perfectly clear* had the largest percentage overall. Additionally, the *Perfectly clear* percentage between short sentences (N = *50*) and long sentences (N = *25*), at 44% and 23%, respectively, indicated that performance with short sentences was more significantly clear in semantic expressiveness.

Table 7. The judgements of semantic expressiveness in intrinsic evaluation.

	Perfectly Clear	*Clear*	*Acceptable*	*Unclear*	*Very Unclear*	*Total*
Group A	119	152	128	21	15	435
Group B	130	192	98	9	6	435
Group C	143	223	109	4	1	480
Group D	127	166	90	3	4	390
Group E	139	226	101	11	3	480
	30%	**43%**	**24%**	**2%**	**1%**	**2220**

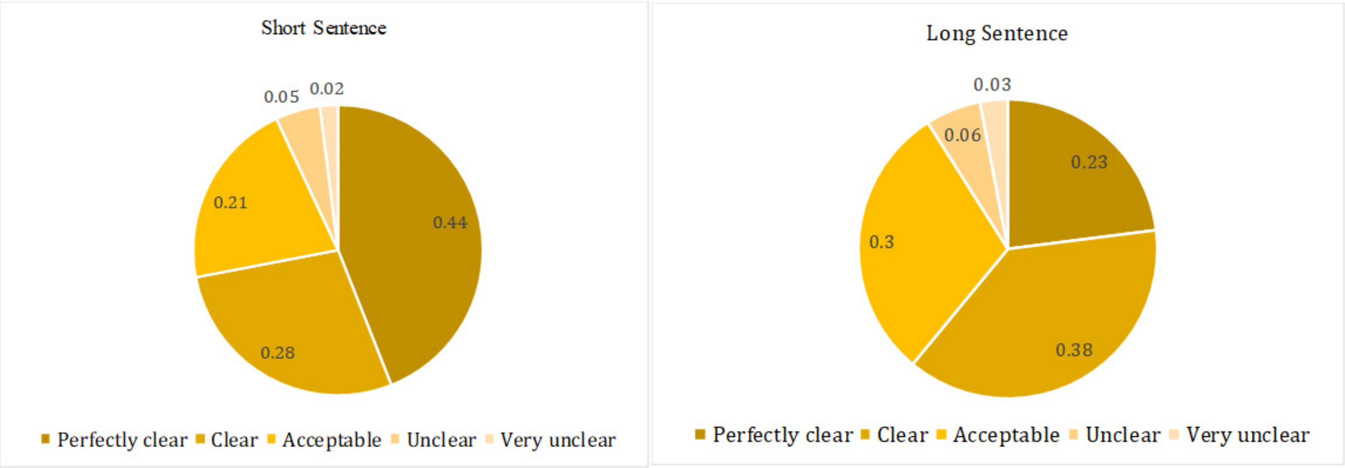


Figure 11. The percentage of semantic expressiveness for short and long sentences in intrinsic evaluation.

Table 8 shows the experimental results using MParser and human expert labeling. The average κ values were 0.693 for inter-annotator agreement and 0.717 for intra-annotator agreement. As $0.6 < \kappa < 0.8$ indicates substantial agreement, the empirical results showed good consistency between the predictions generated by our approach and those of experts. The analysis of the κ values between three experts found that the agreement κ values for experts 2 and 3 were relatively higher. Experts 1 and 2, 3 had a slight gap, but the κ values were still within the range $0.6 < \kappa < 0.8$. Table 8 found that experts 2 and 3 had higher average κ values than expert 1 in intra-annotator agreement. In addition, we calculated average κ values for intra-annotator agreement between short sentences and long sentences, as shown in Table 9. The average κ value for long sentences was significantly lower than that for short sentences. This result is consistent with the trend for our intrinsic evaluation, which showed that the higher complexity of a sentence was more likely to cause disagreement in case grammar labeling. In summary, comparing expert and MParser outputs, inter-annotator and intra-annotator agreement presented substantial results, and there was no major disagreement between our MParser results and those of the experts.

Table 8. Kappa agreement between experts and Mparser.

Group	Inter-Annotator (Expert, Expert)	κ *	$\kappa_{avg.}$	Intra-Annotator (Expert, MParser)	κ *	$\kappa_{avg.}$
Group **A** (*N* = *15*)	*(Expert 1, Expert 2)* *(Expert 2, Expert 3)* *(Expert 1, Expert 3)*	*0.688* ***0.831*** *0.703*	*0.741*	*(Expert 1, MParser)* *(Expert 2, MParser)* *(Expert 3, MParser)*	*0.637* ***0.853*** *0.768*	*0.753*
Group **B** (*N* = *15*)	*(Expert 1, Expert 2)* *(Expert 2, Expert 3)* *(Expert 1, Expert 3)*	*0.597* ***0.766*** *0.627*	*0.663*	*(Expert 1, MParser)* *(Expert 2, MParser)* *(Expert 3, MParser)*	*0.537* *0.685* ***0.781***	*0.668*
Group **C** (*N* = *15*)	*(Expert 1, Expert 2)* *(Expert 2, Expert 3)* *(Expert 1, Expert 3)*	*0.648* ***0.673*** *0.605*	*0.642*	*(Expert 1, MParser)* *(Expert 2, MParser)* *(Expert 3, MParser)*	*0.603* *0.779* ***0.821***	*0.734*
Group **D** (*N* = *15*)	*(Expert 1, Expert 2)* *(Expert 2, Expert 3)* *(Expert 1, Expert 3)*	*0.694* ***0.775*** *0.593*	*0.687*	*(Expert 1, MParser)* *(Expert 2, MParser)* *(Expert 3, MParser)*	*0.613* ***0.835*** *0.724*	*0.724*
Group **E** (*N* = *15*)	*(Expert 1, Expert 2)* *(Expert 2, Expert 3)* *(Expert 1, Expert 3)*	*0.686* ***0.837*** *0.668*	*0.730*	*(Expert 1, MParser)* *(Expert 2, MParser)* *(Expert 3, MParser)*	*0.616* *0.749* ***0.753***	*0.706*
Avg.	***Substantial***		***0.693***	***Substantial***		***0.717***

* *p* value < 0.001.

Table 9. Kappa intra-annotator agreement between short and long sentences.

Sentence Type	Inter-Annotator (Expert, MParser)	$\kappa_{avg.}$
All Sentences *(N = 75)*	*(Expert 1, MParser)* *(Expert 2, MParser)* *(Expert 3, MParser)*	*0.601* ***0.780*** *0.769*
Short Sentence *(length <= 8)* *(N = 50)*	*(Expert 1, MParser)* *(Expert 2, MParser)* *(Expert 3, MParser)*	*0.728* ***0.834*** *0.819*
Long Sentence *(length > 8)* *(N = 25)*	*(Expert 1, MParser)* *(Expert 2, MParser)* *(Expert 3, MParser)*	*0.505* ***0.726*** *0.719*

7.4. Discussion

7.4.1. Case Labeling

From the experimental results in 7.3, we can see that our MParser had better results. We also calculated each case match rate (MR) for all words (*N* = *505*) between experts and MParser outputs as the ratio of *MatchedCase* to *TotalCase*.

From the results shown in Table 10, we found that PRE and GEN cases had extremely high MRs, which were 0.986 and 0.959, respectively. ADV, ACC, and LIN cases came next. The MR of DAT was relatively low because of the differences in the judgment of the infinitive. To our surprise, the MR of the NOM case was relatively low. Through one-to-one analysis of sentences, we found that when nouns were under the COM (COM$_n$/COM$_v$) structure, some experts still labeled the COM case for nouns, and our MParser identified the nouns as NOM case. For COM, COM$_n$, and COM$_v$ cases, the MR was not very high because the experts had different labels on which COM case to use for prepositions. However, if the COM case was considered a general COM case, COM$_{all}$, the average of the MR achieved a very high score, which was 0.920, indicating a consensus on the COM case.

Table 10. Case Match Rate (MR) between Experts and MParser outputs.

Intra-Annotator **(Expert, MParser)**	**MR (N = 505)**											
	NOM	***PRE***	***ACC***	***DAT***	***GEN***	***LIN***	***ADV***	***COM***	***COMv***	***COMn***	***COM_{all}***	***Avg.***
(Expert 1, MParser)	*0.684*	*0.979*	*0.804*	*0.647*	*0.958*	*0.756*	*0.840*	*0.682*	*0.649*	*0.690*	*0.916*	***0.782***
(Expert 2, MParser)	*0.706*	*1*	*0.847*	*0.684*	*0.973*	*0.807*	*0.891*	*0.639*	*0.711*	*0.687*	*0.907*	***0.805***
(Expert 3, MParser)	*0.715*	*0.979*	*0.828*	*0.749*	*0.947*	*0.784*	*0.874*	*0.662*	*0.684*	*0.648*	*0.938*	***0.801***
Avg.	***0.702***	***0.986***	***0.826***	***0.693***	***0.959***	***0.782***	***0.868***	***0.661***	***0.681***	***0.675***	***0.920***	

7.4.2. Semantic Consistency

Here, we discuss the multilingual semantic consistency of MParser between English and Chinese. In MParser, a sentence is a concept tree, consisting of simple sentences defined by a sequential list *SiS*, where each atomic concept *iid* is a *low-level concept llc*∈LLC in the step of human semantic input (HSI), and compound concept *eiid* ∈EIID is a *high-level concept hlc*∈HLC generated in MParser, acting as a sentence constituent in the step of sentence computerization (SC). Given two sentences, SiS_i, which is an English sentence, and $SiS_{j,}$ which is a Chinese sentence, if low-level concept equivalence and high-level concept equivalence are equal such that $SiS_i = {}_m\ SiS_j$ ($=_m$ indicates semantic equivalence), then they are semantically consistent. As low-level concept equivalence is semantic consistency of terms, or word-based, high-level concept equivalence is sentence-based semantic consistency.

1. Low-level concept equivalence: *SiS_i and SiS_j are equivalent if and only if:*

(1) $\forall\ LLC_i \subset IID_i \subset CoDic$

(2) $\forall\ LLC_j \subset IID_j \subset CoDic$

(3) *Mapping relationship:* $LLC_i \leftrightarrow LLC_j$

This guarantees that two heterogeneous single concepts are semantically consistent, as two sentences share a common $iid \in CoDic$.

2. High-level concept equivalence: *SiS_i and SiS_j are equivalent if and only if:*

(1) $\forall\ HLC_i \subset EIID_i$

(2) $\forall\ HLC_j \subset EIID_j$

(3) *Mapping relationship:* $HLC_i \leftrightarrow HLC_j$

HLC achieves complex concept consistency by converging all heterogeneous structures onto an isomorphic grammatical structure through MParser.

3. LLC ⇔ HLC: *LLC and HLC are equivalent if and only if:*

(1) *Mapping relationship: IID ↔ EIID, which is iid in Def. 4 mapped to eiid in Def. 5*

Thus, if and only if the following mapping path exists for semantic equivalence:

SiS_i (local concept) ↔ LLC_i (local concept, $IID_{i,}$) ↔ Map (IID_i, Common concept) ↔ HLC_i (Common concept, $EIID_i$) ↔ HLC_j ($EIID_j$, Common concept) ↔ Map (Common concept, IID_j) ↔ LLC_j (IID_j, local concept) ↔ SiS_j (local concept)

It is obvious that if all three conditions are met, then $SiS_i =_m SiS_j$. Figure 12 illustrates that languages *i* and *j* are semantically consistent as they share common tree concepts in cross languages through the unique *iid* and *eiid* in MParser.

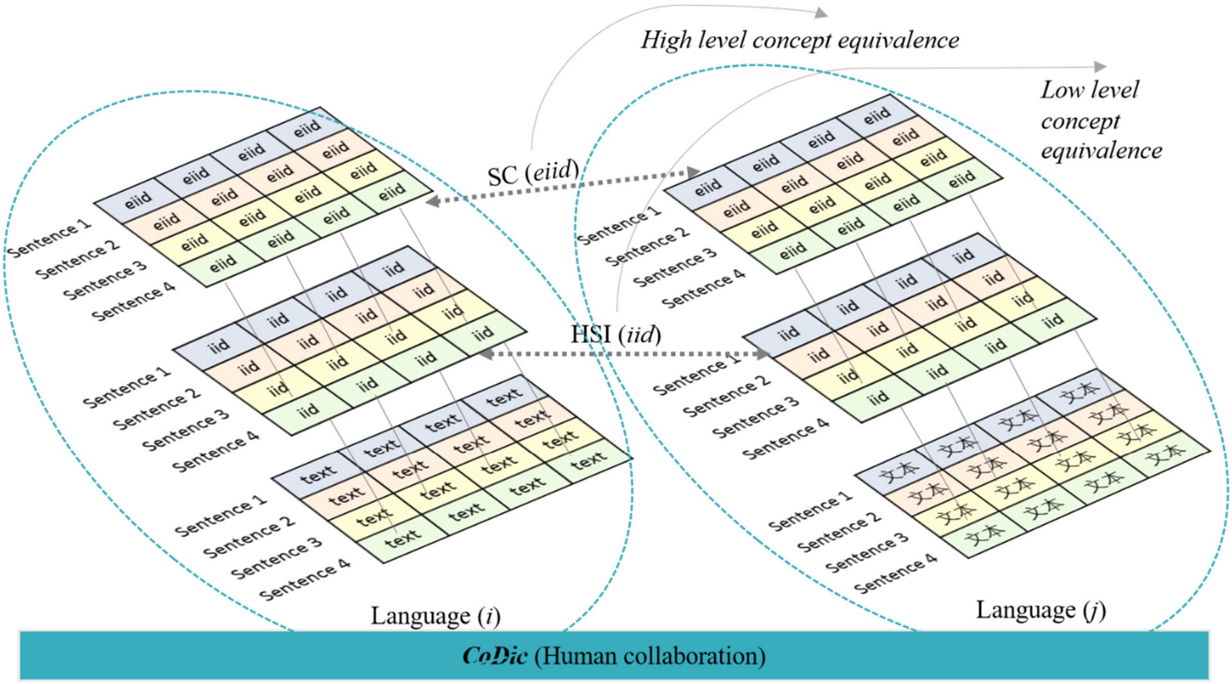


Figure 12. An illustration on semantic consistency.

8. Conclusions and Future Work

Creating a common semantic representation for multilingual languages is an essential goal of the NLP community. To facilitate multilingual sentence representation and semantic interoperability, this research presented an MParser for parsing local language sentences and providing a common understanding across the heterogeneous sentence. MParser converts complex concepts into a computer-readable and -understandable universal sentence for any simple multilingual sentence. This approach has provided a universal grammatical feature such that any sentence can be processed as a bag of concepts and refer to any term of a natural language. Additionally, it has laid a theoretical foundation for enabling humans and computers to understand sentences semantically through unique *iid* and *eiid*.

In the future, we plan to apply the approach to more real-world applications. For example, we will conduct research on how to achieve content persistence during construction of the Metaverse [44] by proposing a content-level persistence maintenance model since the ambiguity of the language, the use of synonyms to express a single idea, creates problems. In the blockchain, we will explore the question of how to achieve semantic interoperability between IoT devices and users [45]. In the field of smart contracts, we will study the cross-context issues of smart contracts between unknown business partners such as developers or anybody who even comes from different backgrounds or languages. Since language barriers prevent cross-language searches, most users do not have easy access to most of this [46]. Moreover, it also will be necessary to extend the research, including semantic inference on extracted meaning. We hope that our novel method will inspire the community to integrate various functions into our work.

Author Contributions: Conceptualization, P.Q. and J.G.; methodology, P.Q., J.G., W.T. and B.S.; software, P.Q.; writing—original draft preparation, P.Q.; writing—review and editing, W.T., B.S. and Q.T.; supervision, J.G.; project administration, J.G.; funding acquisition, J.G., Q.T. All authors have read and agreed to the published version of the manuscript.

Funding: This research is partially supported by the University of Macau Research Grant No. MYRG2017-00091-FST, MYRG2019-00024-FST and FDCT-NSFC Grant No. 0004/2019/AFJ.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A. Parts of Speech (PoS)

Table 1. PoS in CoDic.

PoS		Abbr.	Definition
Noun (n)	Common	*Ncm*	A term class denoting a common entity.
	Proper Person	*npp*	A term class denoting a proper person entity.
	Proper Organization	*nop*	A term class denoting a proper organizational entity.
	Proper Geography	*ngp*	A term class denoting a proper geographical entity.
	Pronoun	*npr*	A term class substituting a noun or a noun phrase.
Verb (v)	Intransitive	*vit*	A term class denoting an action, an event, or a state without following any entity.
	Transitive	*vtr*	A term class denoting an action, an event, or a state following only one entity.
	Ditransitive	*vdi*	A term class denoting an action, an event, or a state without following only two entities.
	Copulative	*cop*	A term class denoting a linkage between an entity and a *copulated component* (coc) that expresses a state of being. Adopting "coc" is to avoid the confusion of current use of "predicative expression".
Adjective		*adj*	A term class describing the attributes of an entity.
Adverb		*adv*	A term class describing the attributes of an action, an event, or a state.
Preposition		*prep*	A term class denoting a relation to other noun-formed term(s) before, in the middle, or after.
Conjunction		*conj*	A term class connecting terms, phrases and clauses, such as *and*, *or,* and *if*.
Interjection		*int*	A term class expressing a spontaneous feeling or reaction.
Onomatopoeia		*ono*	A term class imitating, resembling, or suggesting a sound.
Particle		*par*	A term class indicating a case encompassed by it.

B. Grammatical Features

In MParser, the gender and number features are only attributed to nouns. The features of tense, aspect, and voice are only attributed to verbs. For the grammatical aspects, we have the following definitions:

- *Perfect* (*prf*): a verb form that indicates that an action or circumstance occurred earlier than the time under consideration, often focusing attention on the resulting state rather than on the occurrence itself. E.g., "I have made dinner".
- *Perfect Progressive* (*pfg*): a verb form that indicates that an action was progressive and finished at a time. E.g., "I had been doing homework until 6 PM yesterday".
- *Perfective* (*pfv*): a grammatical aspect that describes an action viewed as a simple whole, i.e., a unit without interior composition. Sometimes called the aoristic aspect, which is a verb form to usually refer to past events. For example, "I came".
- *Imperfective* (*ipfv*): a grammatical aspect used to describe a situation viewed with interior composition. The imperfective is used to describe ongoing, habitual, repeated, or similar semantic roles, whether that situation occurs in the past, present, or future. Although many languages have a general imperfective, others have distinct aspects for one or more of its various roles, such as progressive, habitual, and iterative aspects.

1. *Imperfective habitual* (*iph*): describes habitual and repeated actions. For example, "I read". "The rain beat down continuously through the night".
2. *Imperfective progressive* (*ipp*): describes ongoing actions or events. For example, "The rain was beating down".

Thus, we now have the feature combinations for noun and verb as shown in Tables 2 and 3.

Table 2. Grammatical features of noun on morphological change.

Number	Gender	Binary Postfix	Hex Postfix
Countable singular	Neuter	0000	0
	Masculine	0001	1
	Feminine	0010	2
	Bisexual	0011	3
Countable plural	Neuter	0100	4
	Masculine	0101	5
	Feminine	0110	6
	Bisexual	0111	7
Uncountable	Neuter	1000	8
	Masculine	1001	9
	Feminine	1010	A
	Bisexual	1011	B

Table 3. Grammatical features of verb on morphological change.

Voice	Tense	Aspect	Binary Postfix	Hex Postfix
active	Present	Perfect	0000 0000	00
		Perfect progressive	0000 0001	01
		Perfective	0000 0010	02
		Imperfective habitual	0000 0011	03
		Imperfective progressive	0000 0100	04
	Past	Perfect	0001 0000	10
		Perfect progressive	0001 0001	11
		Perfective	0001 0010	12
		Imperfective habitual	0001 0011	13
		Imperfective progressive	0001 0100	14
	Future	Perfect	0010 0000	20
		Perfect progressive	0010 0001	21
		Perfective	0010 0010	22
		Imperfective habitual	0010 0011	23
		Imperfective progressive	0010 0100	24
	Past future	Perfect	0011 0000	30
		Perfect progressive	0011 0001	31
		Perfective	0011 0010	32
		Imperfective habitual	0011 0011	33

C. MParser Output—75 Sentences

In MParser, we manually input 75 valid sentences and automatically output parsed results for each sentence, as shown in Table 4.

Table 4. 75 sentences from Mparser output.

1.	*I like apples.* *(I.NOM like.PRE apple.ACC)*
2.	*I miss those times and cherish them often.* *(I.NOM miss.PRE those.GEN time.ACC cherish.PRE them.ACC often.ADV)*
3.	*She has been found.* *(She.NOM find.PRE)*
4.	*Nobody can understand.* *(Nobody.NOM can.PRE understand.PRE)*
5.	*His method was strange but impressive.* *(His.GEN method.NOM was.PRE strange.LIN impressive.LIN)*
6.	*She said she is waiting until night.* *(she.NOM said.PRE she.NOM. wait.PRE. until.COMv night.NOM)*

Table 4. *Cont.*

7.	*We need to speed into perspective.* *(we.NOM. need.PRE speed.PRE into.COMv perspective.NOM)*
8.	*The size of sample will change user behavior.* *(size.NOM of sample.NOM change.PRE user.ACC behavior.ACC)*
9.	*The car was sold with a three warranty.* *(Car.ACC sell.PRE with.COMv three.NOM warranty.NOM)*
10.	*The crash occurred in our province.* *(Crash.NOM occurr.PRE in.COMv our.GEN province.NOM)*
11.	*Russia remains hostage oil and gas prices.* *(Russia.NOM remain.PRE hostage.ACC oil.ACC gas.ACC price.ACC)*
12.	*Previous appointees stayed the role until their deaths.* *(Previous.GEN appointee.NOM stay.PRE role.ACC until.COMv their.GEN death.NOM)*
13.	*Everyone has been for their particular skill.* *(Everyone.NOM is.PRE for.LIN their.GEN particular.GEN skill.NOM)*
14.	*They have their cake and eat it too.* *(They.NOM have.PRE their.GEN cake.ACC eat.PRE it.ACC too.ADV)*
15.	*It was experiencing some hard moments.* *(It.NOM experience.PRE some.GEN hard.GEN moment.ACC)*
16.	*I ′m going to join the club.* *(I.NOM go.PRE join.PRE club.ACC)*
17.	*This dispute with the legal is just beginning.* *(This.GEN dispute.NOM with.COMn legal.NOM is.LIN just.ADV beginning.COMn)*
18.	*She said the outage started in the afternoon.* *(She.NOM said.PRE outage.NOM started.PRE in.COMv afternoon.NOM)*
19.	*Our teacher's appearance looks bad and dirty.* *(Our.GEN teacher.NOM appearance.NOM look.LIN bad.COM dirty.COM)*
20.	*The quick brown fox jumped over the lazy dog.* *(Quick.GEN brown.GEN fox.NOM jump.PRE over.COMv lazy.GEN dog.NOM)*
21.	*I wish you are lucky too.* *(I.NOM wish.PRE you.NOM are.PRE lucky.LIN too.ADV)*
22.	*I spoke to my mum at last night.* *(I.NOM spoke.PRE my.GEN mum.ACC at.COMv last.GEN night.NOM)*
23.	*Everybody wants to their mark.* *(Everybody.NOM want.PRE their.GEN mark.ACC)*
24.	*The dog is hit by the man heavily.* *(Dog.ACC hit.PRE by.COMv man.NOM heavily.ADV)*
25.	*The day finally dawned.* *(Day.NOM finally.ADV dawn.PRE)*
26.	*They are just excited about the honor.* *(They.NOM are.PRE just.ADV excited.LIN about.COMv honor.NOM)*
27.	*She detailed the highs and lows.* *(She.NOM detail.PRE high.ACC low.ACC)*
28.	*Two of the soldiers were catching ride.* *(Two.NOM soldier.NOM catch.PRE ride.ACC)*
29.	*The students also track the men's progress.* *(Student.NOM also.ADV track.PRE man. ACC progress.ACC)*
30.	*He is popular in all of the House.* *(He.NOM is.PRE popular.LIN in.COMv all.GEN House.NOM)*
31.	*Fame released in UK cinemas.* *(Fame.NOM release.PRE in.COMv UK.NOM cinema.NOM)*
32.	*I enjoy travel in summer.* *(I.NOM enjoy.PRE travel.ACC in.COMv summer.NOM)*
33.	*We relied on the integrity of truth.* *(We.NOM rely.PRE integrity.ACC truth.ACC)*
34.	*His sense of taste is returning.* *(His.GEN sense.NOM taste.NOM return.PRE)*

Table 4. *Cont.*

35.	*Home builders also jumped most financials.* *(Home.NOM builder.NOM also.ADV jump.PRE most.GEN financial.ACC)*
36.	*They were taxed income when we earned them.* *(They.NOM tax.PRE income.ACC we.NOM earn.PRE them.ACC)*
37.	*She joined a sport during primary school.* *(She.NOM join.PRE sport.ACC during.COMv primary.NOM school.NOM)*
38.	*Your friends are good men.* *(Your.GEN friend.NOM are.LIN good.GEN man.ACC)*
39.	*You will find links to this news.* *(You.NOM find.PRE link.ACC to.COMv this.GEN news.NOM)*
40.	*Some radio channels will move new position.* *(Some.GEN radio.NOM channel.NOM move.PRE new.GEN position.ACC)*
41.	*She has also worked with battery hens.* *(She.NOM also.ADV work.PRE with.COMv battery.NOM hen.NOM)*
42.	*The group now owns venues across the country.* *(Group.NOM now.NOM own.PRE venue.ACC across.COMv country.NOM)*
43.	*The student finished their season in one hour.* *(Student.NOM finish.PRE their.GEN season.ACC in.COMv one.NOM hour.NOM)*
44.	*It sets the two on collision courses.* *(It.NOM set.PRE two.ACC on.COMv collision.NOM course.NOM)*
45.	*The two people were taking in the class.* *(Two.GEN people.NOM talk.PRE in.COMv class.NOM)*
46.	*The financial crisis has many of those bets.* *(Financial.GEN crisis.NOM has.PRE many.GEN those.GEN bet.ACC)*
47.	*The party is at a new location.* *(Party.NOM is.PRE at.LIN new.GEN location.NOM)*
48.	*This is great place to start the trip.* *(This.NOM is.PRE great.COM place.LIN start.PRE trip.ACC)*
49.	*I want to pick something else really.* *(I.NOM want.PRE pick.PRE something.ACC else.GEN really.ADV)*
50.	*You should find a similar thing like sport.* *(You.NOM should.ADV find.PRE similar.GEN thing.ACC like.COMv sport.NOM)*
51.	*The violence was some of the worst ethnic in China for decades.* *(Violence.NOM is.PRE some.GEN worst.GEN ethnic.ACC in.COMn China.NOM for.COMv decade.NOM)*
52.	*The market is mired in scandals and has not recovered good.* *(Market.NOM mired.PRE in.COMv scandals.NOM not.ADV recover.PRE good.COM)*
53.	*The insurgents often attack police and sometimes city officials at night.* *(Insurgent.NOM often.ADV attack.PRE police.ACC sometimes.ADV city.ACC official.ACC at.COMv night.NOM.)*
54.	*The cake is made by the shop after months slowly.* *(Cake.ACC made.PRE by.COMv shop.NOM after.COMv month.NOM slowly.ADV)*
55.	*His detention began in this week when he was trying to leave the city on a false passport.* *(His.GEN detention.NOM begin.PRE in.COMv this.GEN week.NOM he.NOM try.PRE leave.PRE city.ACC on.COMv false.GEN passport.NOM)*
56.	*I want to thank every member of congress who stood tonight with courage.* *(I.NOM want.PRE thank.PRE every.GEN member.ACC congress.ACC stand.PRE tonight.ADV with.COMv courage.NOM)*
57.	*It was his job to fight the war and make an assessment when the time came.* *(It.NOM is.PRE his.GEN job.ACC fight.PRE war.ACC make.PRE assessment.ACC time.NOM come.PRE)*

Table 4. *Cont.*

58.	*The Justice Department scheduled a news conference Tuesday afternoon to announce the indictment.* *(Justice.NOM Department.NOM schedule.PRE news.ACC conference.ACC in.COMv afternoon.NOM announce.PRE indictment.ACC)*
59.	*The president had been scheduled to leave for the trip on Sunday.* *(President.NOM schedule.PRE leave.PRE for.COMv trip.NOM on.COM Sunday.NOM)*
60.	*A sale has been hit after a robbery in a store.* *(Sale.ACC hit.PRE after.COMv robbery.NOM in.COMv store.NOM)*
61.	*I have won this race twice and it would be great to win it again.* *(I.NOM win.PRE this.GEN race.ACC twice.ADV it.NOM is.LIN great.COM win.PRE it.ACC again.ADV)*
62.	*We 've got great commanders on the ground in leadership.* *(We.NOM get.PRE great.GEN commander.ACC on.COMv ground.NOM in.COMv leadership.NOM)*
63.	*He intends to return to the company within next year.* *(He.NOM intend.PRE return.PRE company.ACC within.COMv next.GEN year.NOM)*
64.	*Providing sensitive information to strangers by phone is dangerous.* *(Providing.PRE sensitive.GEN information.ACC to.COMv stranger.NOM by.COMn phone.NOM is.LIN dangerous.COM)*
65.	*She heard the noise and thought someone must have been making it for the event.* *(She.NOM hear.PRE noise.ACC think.PRE someone.NOM must.ADV make.PRE it.ACC for.COMv event.NOM)*
66.	*He had been banned over fears that raised the chances of contamination.* *(He.ACC ban.PRE over.COMv fear.NOM raise.PRE chance.NOM contamination.NOM)*
67.	*Readers who want local color in their mysteries usually seek exotic foreign.* *(Reader.NOM want.PRE local.GEN color.ACC in.COMv their.GEN mystery.NOM usually.ADV seek.PRE exotic.GEN foreign.ACC)*
68.	*He said he will develop a new investment strategy for several months.* *(He.NOM said.PRE he.NOM develop.PRE new.GEN investment.NOM strategy.NOM for.COMv several.GEN month.NOM)*
69.	*The emerging legislation is at his economic recovery program for further years.* *(Emerging.GEN legislation.NOM is.PRE at.LIN his.GEN economic.NOM recovery.NOM program.NOM for.COMv further.GEN year.NOM)*
70.	*All the records were always at hand if we must call about something.* *(All.GEN record.NOM are.LIN always.ADV at.COM hand.NOM we.NOM must.ADV call.PRE about.COMv something.NOM)*
71.	*The TV series has become a big hit among viewers who find empathy with characters in the drama.* *(TV.NOM series.NOM become.PRE big.GEN hit.NOM among.COMv viewer.NOM find.PRE empathy.ACC with.COMv character.NOM in.COMv drama.NOM)*
72.	*The chain of workers involved in real estate deals has grown over the years.* *(Chain.NOM worker.NOM involved.PRE in.COMv real.GEN estate.NOM deal.NOM grow.PRE over.COMv year.NOM)*
73.	*Rival studios have come together to push consumers to rent more movies on their cable boxes.* *(Rival.NOM studio.NOM come.PRE together.ADV push.PRE consumer.ACC rent.PRE more.GEN movie.ACC on.COMv their.GEN cable.NOM boxe.NOM)*
74.	*He fled to a neighboring town where he took a family hostage.* *(he.NOM fled.PRE neighbour.GEN town.ACC he.NOM take.PRE family.NOM hostage.NOM)*
75.	*Everyone was expecting France teams to make the finals competition.* *(Everyone.NOM expect.PRE France.ACC team.ACC make.PRE final.GEN competition.ACC)*

References

1. Zou, Y.; Lu, W. Learning Cross-lingual Distributed Logical Representations for Semantic Parsing. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), Melbourne, Australia, 15–20 July 2018.
2. Balahur, A.; Perea-Ortega, J.M. Sentiment analysis system adaptation for multilingual processing: The case of tweets. *Inf. Process. Manag.* **2015**, *51*, 547–556. [CrossRef]
3. Noraset, T.; Lowphansirikul, L.; Tuarob, S. WabiQA: A Wikipedia-Based Thai Question-Answering System. *Inf. Process. Manag.* **2021**, *58*, 102431. [CrossRef]
4. Zheng, J.; Li, Q.; Liao, J. Heterogeneous type-specific entity representation learning for recommendations in e-commerce network. *Inf. Process. Manag.* **2021**, *58*, 102629. [CrossRef]
5. Etaiwi, W.; Awajan, A. Graph-based Arabic text semantic representation. *Inf. Process. Manag.* **2020**, *57*, 102183. [CrossRef]
6. Liang, P. Learning executable semantic parsers for natural language understanding. *Commun. ACM* **2016**, *59*, 68–76. [CrossRef]
7. Liang, P.; Potts, C. Bringing Machine Learning and Compositional Semantics Together. *Annu. Rev. Linguistics* **2015**, *1*, 355–376. [CrossRef]
8. Bos, J.; Basile, V.; Evang, K.; Venhuizen, N.J.; Bjerva, J. The Groningen Meaning Bank. In *Handbook of Linguistic Annotation*; Springer: Berlin/Heidelberg, Germany, 2017; pp. 463–496.
9. Banarescu, L.; Bonial, C.; Cai, S.; Georgescu, M.; Griffitt, K.; Hermjakob, U.; Knight, K.; Koehn, P.; Palmer, M.; Schneider, N. Abstract meaning representation for sembanking. In Proceedings of the LAW, Sofia, Bulgaria, 8–9 August 2013; pp. 178–186.
10. Abend, O.; Dvir, D.; Hershcovich, D.; Prange, J.; Schneider, N. Cross-lingual Semantic Representation for NLP with UCCA. In Proceedings of the 28th International Conference on Computational Linguistics: Tutorial Abstracts, Barcelona, Spain, 8–13 December 2020; pp. 1–9.
11. Boguslavsky, I.; Frid, N.; Iomdin, L.; Kreidlin, L.; Sagalova, I.; Sizov, V. Creating a Universal Networking Language module within an advanced NLP system. In Proceedings of the 18th Conference on Computational Linguistics, Saarbrücken, Germany, 31 July–4 August 2000; Volume 1, pp. 83–89.
12. Nivre, J.; Marneffe, M.-C.D.; Ginter, F.; Goldberg, Y.; Hajic, J.; Manning, C.D.; McDonald, R.; Petrov, S.; Pyysalo, S.; Silveira, N.; et al. Universal dependencies v1: A multi-lingual treebank collection. In Proceedings of the of LREC, Portorož, Slovenia, 23–28 May 2016; pp. 1659–1666.
13. Xiao, G.; Guo, J.; Da Xu, L.; Gong, Z. User Interoperability with Heterogeneous IoT Devices Through Transformation. *IEEE Trans. Ind. Inform.* **2014**, *10*, 1486–1496. [CrossRef]
14. Nikiforov, D.; Korchagin, A.B.; Sivakov, R.L. An Ontology-Driven Approach to Electronic Document Structure Design. In *Communications in Computer and Information Science*; Springer: Berlin/Heidelberg, Germany, 2017; pp. 3–16.
15. Xiao, G.; Guo, J.; Gong, Z.; Li, R. Semantic input method of Chinese word senses for semantic document exchange in e-business. *J. Ind. Inf. Integr.* **2016**, *3*, 31–36. [CrossRef]
16. Qin, P.; Guo, J. A novel machine natural language mediation for semantic document exchange in smart city. *Futur. Gener. Comput. Syst.* **2020**, *102*, 810–826. [CrossRef]
17. Guo, J. Collaborative conceptualisation: Towards a conceptual foundation of interoperable electronic product catalogue system design. *Enterp. Inf. Syst.* **2009**, *3*, 59–94. [CrossRef]
18. Li, W.; Suzuki, E. Adaptive and hybrid context-aware fine-grained word sense disambiguation in topic modeling based document representation. *Inf. Process. Manag.* **2021**, *58*, 102592. [CrossRef]
19. Medjahed, B.; Benatallah, B.; Bouguettaya, A.; Ngu, A.H.; Elmagarmid, A.K. Busi-ness-to-business interactions: Issues and enabling technologies. *VLDB J.* **2003**, *12*, 59–85. [CrossRef]
20. Bing, L.; Jiang, S.; Lam, W.; Zhang, Y.; Jameel, S. Adaptive concept resolution for document repre-sentation and its applications in text mining. *Knowl.-Based Syst.* **2015**, *74*, 1–13. [CrossRef]
21. Tekli, J. An Overview on XML Semantic Disambiguation from Unstructured Text to Semi-Structured Data: Background, Applications, and Ongoing Challenges. *IEEE Trans. Knowl. Data Eng.* **2016**, *28*, 1383–1407. [CrossRef]
22. Decker, S.; Melnik, S.; Van Harmelen, F.; Fensel, D.; Klein, M.; Broekstra, J.; Erdmann, M.; Horrocks, I. The Semantic Web: The roles of XML and RDF. *IEEE Internet Comput.* **2000**, *4*, 63–73. [CrossRef]
23. Wang, T.D.; Parsia, B.; Hendler, J. A survey of the web ontology landscape. In Proceedings of the International Semantic Web Conference, Athens, GA, USA, 5–9 November 2006.
24. Rico, M.; Taverna, M.L.; Caliusco, M.L.; Chiotti, O.; Galli, M.R. Adding Semantics to Electronic Business Documents Exchanged in Collaborative Commerce Relations. *J. Theor. Appl. Electron. Commer. Res.* **2009**, *4*, 72–90. [CrossRef]
25. Governatori, G. REPRESENTING BUSINESS CONTRACTS IN RuleML. *Int. J. Cooperative Inf. Syst.* **2005**, *14*, 181–216. [CrossRef]
26. Tsadiras, A.; Bassiliades, N. RuleML representation and simulation of Fuzzy Cognitive Maps. *Expert Syst. Appl.* **2013**, *40*, 1413–1426. [CrossRef]
27. Marneffe, M.; Maccartney, B.; Manning, C. Generating Typed Dependency Parses from Phrase Structure Parses. In Proceedings of the LREC'06, Genoa, Italy, 22–28 May 2006; pp. 449–454.
28. Guo, J. SDF: A Sign Description Framework for Cross-context Information Resource Representation and Inter-change. In Proceedings of the 2nd Int'l Conference on Enterprise Systems (ICES 2014), Shanghai, China, 2–3 August 2014.
29. Ruppenhofer, J.; Ellsworth, M.; Schwarzer-Petruck, M.; Johnson, C.R.; Baker, C.F.; Scheffczyk, J. *FrameNet II: Extended Theory and Practice*; International Computer Science Institute: Berkeley, CA, USA, 2006.

30. Loper, E.; Yi, S.-T.; Palmer, M. Combining lexical resources: Mapping between PropBank and VerbNet. In Proceedings of the 7th International Workshop on Computational Linguistics, Syktyvkar, Russia, 23–25 September 2007.
31. Palmer, M.; Gildea, D.; Kingsbury, P. The Proposition Bank: An Annotated Corpus of Semantic Roles. *Comput. Linguist.* **2005**, *31*, 71–106. [CrossRef]
32. Xue, N.; Bojar, O.; Hajic, J.; Palmer, M.; Uresova, Z.; Zhang, X. Not an intelingua, but close: Comparison of English AMRs to Chinese and Czech. In Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14), Reykjavik, Iceland, 26–31 May 2014; pp. 1765–1772.
33. White, A.S.; Reisinger, D.; Sakaguchi, K.; Vieira, T.; Zhang, S.; Rudinger, R.; Rawlins, K.; Van Durme, B. Universal Decompositional Semantics on Universal Dependencies. In Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP), Austin, TX, USA, 1–5 November 2016; pp. 1713–1723.
34. Ehrmann, M.; Cecconi, F.; Vannella, D.; McCrae, J.P.; Cimiano, P.; Navigli, R. Representing multilingual data as linked data: The case of babelnet 2.0. In Proceedings of the LREC, Reykjavik, Iceland, 26–31 May 2014; pp. 401–408.
35. Klein, D.; Manning, C.D. Accurate unlexicalized parsing. In Proceedings of the Proceedings of the 41st Annual Meeting on Association for Computational Linguistics—ACL '03, Sapporo, Japan, 7–12 July 2003; pp. 423–430.
36. Cook, V.J. Chomsky's universal grammar and second language learning. *Appl. Linguist.* **1985**, *6*, 2–18. [CrossRef]
37. Starosta, S.; Anderson, J.M. *On Case Grammar: Prolegomena to a Theory of Grammatical Relations*; Routledge: Abingdon, UK, 2018.
38. Gkatzia, D.; Mahamood, S. A Snapshot of NLG Evaluation Practices 2005–2014. In Proceedings of the Proceedings of the 15th European Workshop on Natural Language Generation (ENLG), Brighton, UK, 10–11 September 2015; pp. 57–60.
39. Chelba, C.; Mikolov, T.; Schuster, M.; Ge, Q.; Brants, T.; Koehn, P.; Robinson, T. One billion word benchmark for measuring progress in statistical language modeling. *arXiv* **2013**, arXiv:1312.3005.
40. Brysbaert, M. How Many Participants Do We Have to Include in Properly Powered Experiments? A Tutorial of Power Analysis with Reference Tables. *J. Cogn.* **2019**, *2*, 16. [CrossRef] [PubMed]
41. Shrotryia, V.K.; Dhanda, U. Content Validity of Assessment Instrument for Employee Engagement. *SAGE Open* **2019**, *9*, 2158244018821751. [CrossRef]
42. Carletta, J. Assessing agreement on classification tasks: The kappa statistic. *Comput. Linguist.* **1996**, *22*, 249–254.
43. Landis, J.R.; Koch, G.G. The Measurement of Observer Agreement for Categorical Data. *Biometrics* **1977**, *33*, 159–174. [CrossRef]
44. Shen, B.; Tan, W.; Guo, J.; Zhao, L.; Qin, P. How to Promote User Purchase in Metaverse? A Systematic Literature Review on Consumer Behavior Research and Virtual Commerce Application Design. *Appl. Sci.* **2021**, *11*, 11087. [CrossRef]
45. Shen, B.; Guo, J.; Yang, Y. MedChain: Efficient Healthcare Data Sharing via Blockchain. *Appl. Sci.* **2019**, *9*, 1207. [CrossRef]
46. Qin, P.; Tan, W.; Guo, J.; Shen, B. Intelligible Description Language Contract (IDLC)—A Novel Smart Contract Model. *Inf. Syst. Front.* **2021**, 1–18. [CrossRef]