
Terminology Semantic Sememe Tree Knowledge Graph

Journal Title
XX(X):1–11
©The Author(s) 2026
Reprints and permission:
sagepub.co.uk/journalsPermissions.nav
DOI: 10.1177/ToBeAssigned
www.sagepub.com/

SAGE

Peiyan Wang¹, Feiyan Jiang², Yuyang Wang³ and Sijia Shen⁴

Abstract

Knowledge graphs are foundational language resources and research tools derived from corpus studies. Existing graphs mostly focus on factual or event-based descriptions. Semantic category graphs for domain-specific terminology are scarce. Existing terminology repositories mostly remain at the level where signifiers and signifieds are not fully separated, lacking independent modeling and semantic linking. The Terminology Semantic Sememe Tree Knowledge Graph proposes a representation method for terminological concepts, which separately encodes signifiers (terms) and signifieds (concepts), based on a sememe tree structure. Our knowledge graph consists of three parts: the sememe system repository, the term record repository, and the relation repository. It covers core elements including terms, concepts, concept relations, sememes, and dynamic roles. Unlike other terminological semantic knowledge graphs that integrate terms and definitions into unified descriptions, this knowledge graph embeds such relationships within the sememe tree structures, which facilitates the implementation of semantic computation.

Keywords

Terminology, Sememe tree, Knowledge graph, Signifier, Signified, Semantic computation

Introduction

Currently, mainstream terminology databases worldwide (such as AGROVOC [Food and Agriculture Organization of the United Nations \(2025\)](#), SNOMED CT [International Health Terminology Standards Development Organisation \(2025\)](#), IATE [European Union \(2025\)](#), EuroTermBank [EuroTermBank Consortium \(2025\)](#) and MeSH U.S. [National Library of Medicine \(2025\)](#)) generally adopt the term-concept model. By conducting standardized definitions and semantic annotations of the concepts denoted by terms, they establish structured knowledge organization systems. However, from a semiotic perspective, the construction methods of such terminology databases mostly remain at the level where signifiers and signifieds are not fully separated: terms act as both linguistic symbols and direct references to their denotative concepts, lacking

independent modeling and semantic linking between signifiers and signifieds.

In our dataset, the semantics of a term are expressed in a structured way by multiple sememes and dynamic roles, which called a sememe tree. Sememe is the smallest, indivisible, and unambiguous semantic unit in the field of linguistics [Bloomfield \(1926\)](#). Semantic relations are annotated between different sememes for a single term. These relations are called dynamic roles. The sememe tree provides semantic compositional relations from sememes to terms. It breaks through lexical barriers and opens

¹ ² ³ ⁴School of Computer Science, Shenyang Aerospace University, Shenyang, China

Corresponding author:

Peiyan Wang, Shenyang Aerospace University
No.37 Daoyi South Avenue, Shenbei New Area
Shenyang 110136, Liaoning Province, P.R.China.

Email: wangpy@sau.edu.cn

the semantic black box, enabling people to deeply understand the rich semantics behind words. It also gives semantics a degree of computability. This helps computers perform semantic computation on vocabulary. A term sememe tree, as shown in Figure 1. Connected to the root node is the first sememe, which serves as the most core morpheme for expressing concepts. Dynamic roles, which link individual sememes together, express the relationships between sememes.

As illustrated in the figure, the concept of the term Wing is subsumed by that of the term Airplane; this is reflected in the tree structure by the nesting of the conceptual structure of the term Airplane within that of the term Wing. Similarly, the term adaptive wing is conceptually an extension of the term wing, a feature that is also embodied in the tree structure. The concepts of the terms in our dataset are represented by sememe tree structures, which can clearly illustrate the relationships between terms. Unlike traditional methods that rely on explicit definitions of term relationships, we embed such relationships implicitly within the tree structures—a design that facilitates the implementation of semantic computation between terms.

The sememes in our knowledge graph adopt the sememe system of *HowNet* Dong et al. (2010), which enables domain-specific terminology in our dataset and general-domain vocabulary in *HowNet* to share a unified semantic space. It plays an important role in multiple areas of natural language processing. For example, word similarity calculation Zhu and Sun (2013); Nie et al. (2020); Zhang et al. (2014); Wu et al. (2012), semantic composition Qi et al. (2019), word sense disambiguation Zhang et al. (2005); Wang et al. (2008); Jiang et al. (2014); Hou et al. (2020); Zhou et al. (2019), word vector learning Niu et al. (2017), relation extraction Li et al. (2019), text classification Zhang (2013b), event extraction Ding et al. (2019), sentiment analysis Zeng et al. (2018); Xianghua et al. (2013); Fu et al. (2011); Yan et al. (2007); Liu et al. (2008), and text adversarial attack Zang et al. (2020).

Our knowledge graph consists of three major components: the sememe system repository, the term record repository, and the relation repository.

The sememe system repository stores the classification system of relationships between sememes and dynamic roles, and also includes the synonymous and antonymous relations among sememes. All sememes are divided into four major categories:

(1) Event Taxonomy: There are a total of 801 event sememes in our dataset. Event sememes describe the states of things and the changes that things undergo. For example, sememe {event | 事件} can be categorized into sememe {static | 静态} and sememe {act | 行动}; {static | 静态} are further divided into {relation | 关系}, and {relation | 关系} can be further subdivided into {possession | 领属关系} and {inclusive | 蕴涵关系}.

(2) Entity Taxonomy: There are 151 entity sememes in our dataset. Entity sememes are divided into four categories: thing, parts, time, and space. Parts sememes are classified as "part"({part | 部件}) and "fitting"({fitting | 配件}). Parts are necessary components of the whole, while fittings are not necessary. For example, the sole is a component of the shoe, and the shoelace is its accessory. Space sememes are divided into direction (such as east, west) and location (such as address, point, interior). The time sememes include {TimeSect | 时间段}, {year | 年}, {season | 季}.

(3) Attribute Taxonomy: There are 244 attribute sememes in our dataset. Under specific conditions, attributes are more important than the host itself. Attributes are divided into seven categories: {Appearance | 外观}, {Measurement | 量度}, {Property | 特性}, {Relationship | 关系}, {Situation | 状况}, {QuantityProperty | 数量特性}, {Quantity | 数量}. These seven categories can be subdivided into detailed taxonomy, such as {Form | 形状}, {Beauty | 美丑}.

(4) Attribute-Value Taxonomy: There are 886 attribute-value sememes in our dataset. Attribute values and attributes reference each other. It is also divided into seven categories, corresponding one-to-one with attribute sememes. An attribute value

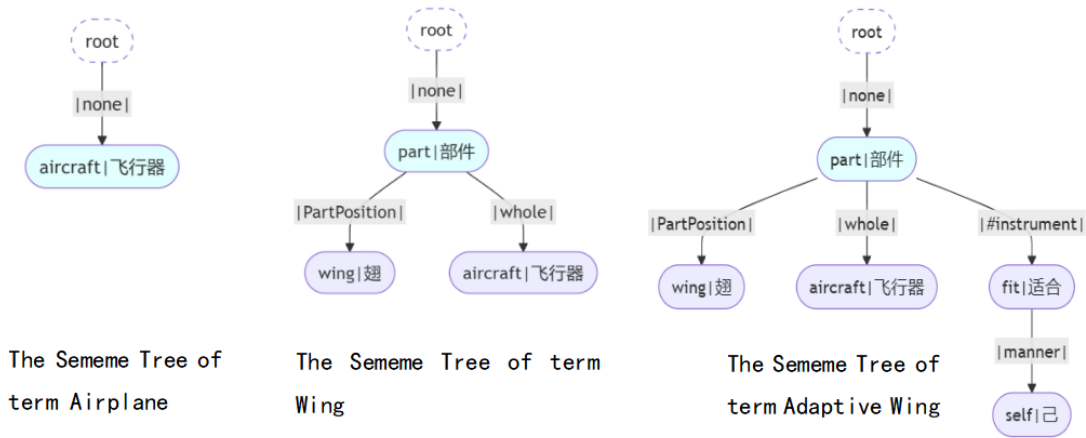


Figure 1. Conceptual Representation of the Sememe Tree

may also have corresponding attribute values or attributes. For example, “{AppearanceValue | 外观 值}” may possess the attribute value “{ColorValue | 颜色 值}”, and “{ColorValue | 颜色 值}” may in turn possess certain attributes representing color.

In addition, our dataset also contains 116 dynamic roles. The semantics of complex concepts is expressed by sememes and the dynamic roles between them. The dynamic roles express the relationships between sememes, such as {According to}, {HostOf}, and other multiple relations.

The term record repository stores 30,806 term records. Each record contains five components: the Chinese term, the English term, definition, source, and concept. The concept is represented in three formats: KDML(Knowledge Database Markup Language), JSON, and Mermaid(visualized). The source is specified with details including book title, chapter, section, and page number. Figure 2 shows the example of term in our knowledge graph.

The relation repository stores pairs of semantic relations between terms, including Hypernym-Hyponym, Whole-Part(Fitting), Host-Attribute, as well as Synonym-Antonym relations. In our dataset, the relationships among concepts such as sememes, dynamic roles, and sememe trees are intertwined and complex. To quantify such semantic relationships and also facilitate management, we transform these concepts into a traditional triple-based knowledge graph structure. The connections between terms, between terms and sememe tree structures, and between

sememe tree structures and sememes together form a complex semantic knowledge graph. On this basis, we employ the KG2E graph embedding algorithm to quantify the edge relationships between nodes and convert them into vector representations, which facilitates tasks such as semantic computation. Our relation repository stores the representation vectors, along with the top 50 nearest neighbor terms for each term, which are calculated based on the Euclidean distance between vectors.

This dataset is built based on the FAIR [Wilkinson et al. \(2016\)](#) principles. It enables users to query corresponding term definitions and obtain implicit relationships between terms. The practical value of this dataset lies in its inclusion of term classifications and relational associations. All sememes, concepts, and term definitions possess semantic relationships. Therefore, term entries can be utilized by semantic reasoning tools for analysis through their classification hierarchies. The dataset also provides a paradigm for constructing domain-specific terminology systems, expanding and redefining the scope of retrieval. It offers references for adding new terms.

An additional advantage of this dataset is that it is shareable, extensible, and reusable. Knowledge graphs can enhance semantic search capabilities for domain-specific terms, improve user understanding of implicit relationships among terms, provide large language models with explicit reasoning paths, and assist in cross-domain semantic borrowing. We

ability to process natural language Zhao et al. (2023). Based on this, we propose a method for generating term sememe trees using large language models. It constructs dynamic prompt templates and uses a chain-of-thought approach to guide large language models to generate term sememe trees. The specific steps are as follows.

Prompt template

This dataset designs four different representation formats of the concept of a term: KDML, list, JSON, and five-tuple. To ensure the accuracy of data generated by large language models, we allow the model to generate only structured five-tuples corresponding to terms based on definite sememes and dynamic roles. In the data post-processing stage, a program converts the five-tuple structures into corresponding KDML and JSON formats. The five-tuple structure is a highly structured method for representing term concepts. Its structure should be: $\langle odeep, osememe, role, fdeep, fsememe \rangle$. *Odeep* is an integer type, which represents the depth of the first sememe in the term sememe tree. *Osememe* is of character type and represents the tail sememe. Role is a dynamic role relationship. *Fdeep* is of integer type and represents the depth of the first sememe in the term sememe tree. *Fsememe* is of character type and represents the first sememe. For example, $\langle 1, \text{burn} | \text{焚烧}, \text{RelateTo}, 0, \text{place} | \text{地方} \rangle$ represents that layer 1 is the sememe $\{\text{burn} | \text{焚烧}\}$, and the relationship between $\{\text{burn} | \text{焚烧}\}$ and $\{\text{place} | \text{地方}\}$ is the dynamic role "RelateTo". The prompt template constructed according to the above rules is shown in Figure 3 and Figure 4.

Neighbor semantic structure

Relying solely on prompt rules is insufficient. It is necessary to derive the neighbor semantic structure of terms and obtain examples of the neighbor semantic structure for given terms to enhance the in-context learning capability of large language models. Many terms do not exist in isolation, they are interrelated, residing within a hierarchical relational network Richardson et al. (1998). Neighbor semantic structure refers to a set of terms with similar semantic

content. These terms achieve semantic association by preserving core meaning and metaphorical extension. Using such neighbor terms as examples to guide the large language model can significantly improve the accuracy in generating term five-tuples. We use the NSAC-TGM Shen et al. (2026) model to generate term vectors for retrieval, computing cosine similarity and then select the top-five terms with the highest frequency as neighbor semantic structure prompt examples.

Post constraint

Although the prompt template restricts the model in terms of data structure, the model may deviate from expectations during generation due to multiple factors, including model randomness and term complexity. Therefore, post-generation constraints must be applied to the generated five-tuples:

(1) Content completeness constraint. The generated sememe tree description must conform to the specified output format. Any sub-list, subtree, or five-tuple that does not conform to the rules will be excluded. Figure 5 shows the specific process for content integrity when the description form is a five-tuple. Input the concept of the term definition d_{um} . First, define the valid semantic element set K , which provides an explicit basis for content constraint verification. Then, process each terminological concept definition a in the input set d_{um} one by one, ensuring that all definitions undergo verification. After all terminological concept definitions have completed element verification and cleaning, the processed set of definitions is output as $d_{filter1}$.

(2) Sememe consistency constraint. The Chinese and English version of sememes must align with the sememe system. If there is one error between the two versions, it should be replaced with the correct sememe from the sememe system. However, if both are incorrect, these sememes remain unchanged because no corresponding correct sememe can be found. Figure 6 shows the specific process of sememe consistency constraints when the description form is a five-tuple. First define the sememe dictionary,

Prompt Template:

You are a term sememe tree generator. Input a term and its definition. Generate a five-tuple representing the concept of the term. **[Rule]**

You must follow these steps to generate a five-tuple in which a term represents a concept. **[COT]**

Example: **[Example]**

Requirements: Enter terms and definitions, providing only the five-tuple structure representing the concept each term represents. **[Input]**

Rule

The five-tuple structure is a set of $\langle odeep, osememe, role, fdeep, fsememe \rangle$. It must contain exactly five elements. *odeep* is of integer type and represents the depth of the first sememe in the term sememe tree structure. *osememe* is of character type and represents the tail sememe. *role* is the dynamic role. *fdeep* is of integer type and represents the depth of the first sememe in the term sememe tree structure. *fsememe* is of character type and represents the first sememe.

CoT

First, generate the first sememe.

Second, generate all sememes.

Third, generate all dynamic role relations.

Finally, generate the five-tuple structure: fill the generated sememes and dynamic role relations into the five-tuple structure.

Example:

Term: blade row

Definition: A blade row consists of multiple blades arranged at equal spacing distances and equal installation angles.

Thinking steps:

Step 1: Generate the first sememe: ['part|部件'].

Step 2: Generate all sememes: ['root', 'part|部件', 'machine|机器', 'part|部件'].

Step 3: Generate all dynamic role relations: ['none', 'whole', 'whole'].

Step 4: Finally generate five-tuple structures: $\langle 0, \text{part|部件}, \text{none}, -1, \text{root} \rangle$, $\langle 1, \text{part|部件}, \text{whole}, 0, \text{part|部件} \rangle$, $\langle 2, \text{machine|机器}, \text{whole}, 1, \text{part|部件} \rangle$.

Figure 3. Prompt Template and Input-Output Examples

where C is the set of all Chinese sememes and S is the set of all sememes. Then, input the filtered result $d_{filter1}$ (generated by the algorithm of content completeness constraint), process each terminological concept definition a . Iterate through each contained element: If the Chinese sememe c of the element exists in the Chinese sememe set, update the element to the corresponding sememe s to ensure consistent Chinese-English sememe matching. After all definitions have been verified and updated, the processed set is output as d .

Data processing

At this stage, we have obtained the validated five-tuples. Next, we only need to process them according to simple rules. We construct the sememes and dynamic roles based on relevant information such as their depth in the sememe tree, which is explicitly kept in the five-tuple. This yields the concept in the format of KDML, JSON, and Mermaid representations corresponding to the terms. Eventually, we developed a corresponding terminology semantic knowledge graph system and integrated all data into JSON format. This constitutes our dataset.

Example:**Term:** blade profile**Definition:** The cross-sectional shape of a blade designed according to specific aerodynamic requirements. It is the basic unit that constitutes the blade.**Thinking steps:**

Step 1, generate the first sememe: ['part|部件'].

Step 2, generate all sememes: ['root', 'part|部件', 'machine|机器', 'part|部件'].

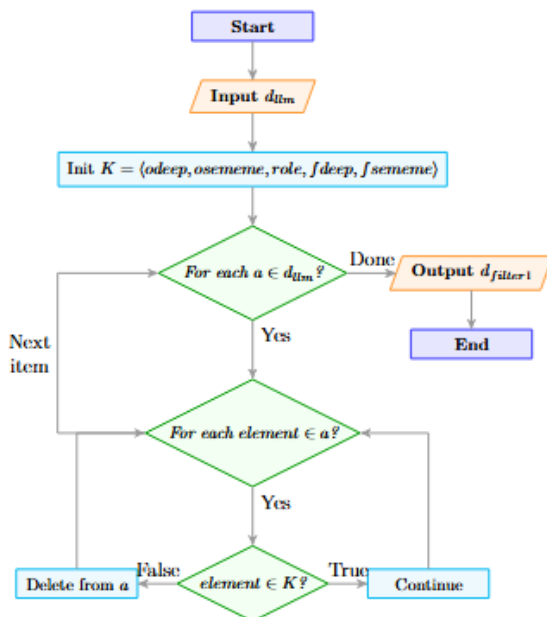
Step 3, generate all dynamic role relations: ['none', 'whole', 'whole'].

Step 4, generate the five-tuple structures: $\langle 0, \text{part|部件}, \text{none}, -1, \text{root} \rangle$, $\langle 1, \text{part|部件}, \text{whole}, 0, \text{part|部件} \rangle$, $\langle 2, \text{machine|机器}, \text{whole}, 1, \text{part|部件} \rangle$.**Term:** rectifier blade**Definition:** Blades fixed on the casing to increase airflow static pressure and straighten the flow.**Thinking steps:**

Step 1, generate the first sememe: ['part|部件'].

Step 2, generate all sememes: ['root', 'part|部件'].

Step 3, generate all dynamic role relations: ['none', 'whole'].

Step 4, generate the five-tuple structures: $\langle 0, \text{part|部件}, \text{none}, -1, \text{root} \rangle$, $\langle 1, \text{machine|机器}, \text{whole}, 0, \text{part|部件} \rangle$.**Input:****Term:** Blade**Definition:** Blade elements formed by spatial stacking according to a specific pattern, or by direct aerodynamic shaping to create spatial curved surfaces, exchange and convert energy with airflow through them.**Output:** $\langle 0, \text{part|部件}, \text{none}, -1, \text{root} \rangle$ 、 $\langle 1, \text{machine|机器}, \text{content}, 0, \text{part|部件} \rangle$ **Figure 4.** Prompt Templates and Input-Output Examples (continued)**Figure 5.** Algorithm of Content Completeness Constraint**Data record**

The dataset is published on Github and includes supporting files:

- README.md – a dataset description file,
- LICENSE.txt – a license for all files,
- tskgs_at – our dataset file folder, which includes the source code for data processing and resource folder. In the resource folder, we have:
 - taxonomy – It includes different taxonomy of sememes, such as antonymy and converse sememe, written in English-Chinese.
 - tskg – This folder holds the key components of our dataset: the vector-form (generated by KG2E algorithm) calculation methods for explicit and implicit inter-term relationships, and the most important file tskg.json—

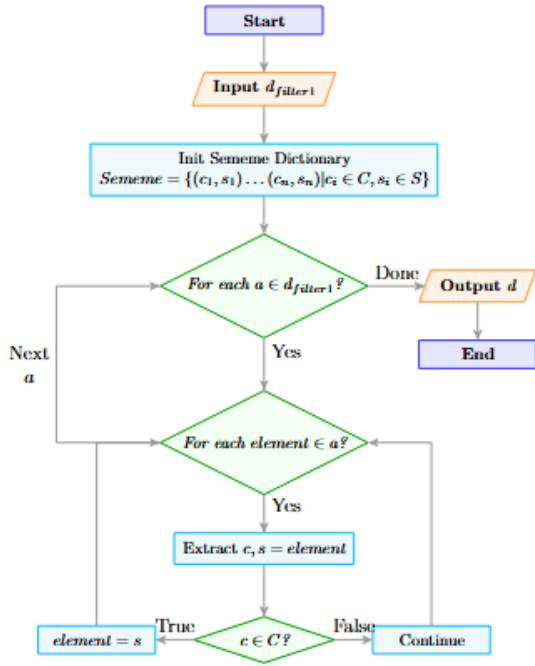


Figure 6. Algorithm of Sememe Consistency Constraint

our dataset file, in which all term concept descriptions are stored in JSON format.

All of the above content is freely available at https://github.com/Yuyang6458/Tskgs_at

Technical Validation

We invited four domain experts to validate 200 term entries, selected via uniform random sampling, for terminological conceptual validation in this dataset and then computed consistency. Consistency verification uses the F1 [Hripcsak and Rothschild \(2005\)](#) score for evaluation. The specific method treats the verification result of one expert (A1) as the ground truth and computes the F1 score of the verification result of another expert (A2) (calculation method as in Equation 3). The average validation consistency of the four experts is 95.25%. After expert validation, the quality of this dataset, namely the sememe tree terminology semantic knowledge graph, passes inspection.

$$P = \frac{A1 + A2}{A1} \quad (1)$$

$$R = \frac{A1 + A2}{A2} \quad (2)$$

$$F_1 = \frac{2 \times P \times R}{P + R} \quad (3)$$

Potential applications

Unlike existing term banks that uniformly represent concepts and their references through "terms + definitions", our knowledge graph innovatively achieves the separated representation of terms and concepts. Specifically, it independently establishes a formal representation system for concepts and clarifies the core concepts denoted by terms through "term-concept" mappings. This architecture overcomes the traditional mode of "binding term symbols with conceptual connotations" and more accurately captures complex linguistic phenomena such as "one concept mapping to multiple terms (e.g., synonymous terms)" and "one term denoting to multiple concepts (e.g., ambiguous terms)". Our dataset can support domain-specific language processing tasks, and its application value is primarily reflected in the following aspects :

- Enhancing performance in fundamental tasks: Through explicit "term-concept" associations and formal concept representations, the dataset can significantly enhance the precision of information retrieval, strengthen the reasoning capacity of intelligent question-answering systems, and improve the domain-specific adaptability of text classification. This offers a new perspective from a semantic dimension for traditional tasks.
- Expanding application scenarios in industry and scientific research: Technological advancements directly enhance the user experience of aviation-domain intelligent customer service and intelligent recommendation systems. More importantly, the structured conceptual system provides a unified semantic framework for interdisciplinary knowledge integration, accelerating domain knowledge sharing and collaborative innovation, and thus facilitating the technological upgrading of the aviation industry.
- Supporting for the Standardization of Terminology: The dataset provides core Chinese-English bilingual term pairs to support the manual or automatic construction of multilingual term dictionaries. This aids in reducing

the manual annotation costs associated with multilingual term alignment, thereby facilitating the efficient development of standardized cross-lingual term resources in the aviation domain. For the term relevance calculation, it supports the design of evaluation samples based on term attributes. Such samples can effectively evaluate models' ability to capture semantic associations among terms. Additionally, the dataset provides aviation term fine-tuning data for general language models, enhancing their ability to perceive domain-specific semantic nuances. Furthermore, it supplements contextual information for terms, mitigating models' comprehension biases toward polysemous aviation terms.

Data Availability

Our dataset of knowledge graph, definitions of terms and the corresponding sememes with dynamic roles are freely available at https://github.com/Yuyang6458/Tskgs_at.

Code Availability

The source code for data processing and the interface functions are freely available at https://github.com/Yuyang6458/Tskgs_at

Author Contribution

P.W. was responsible for the knowledge base architecture, implementation of knowledge base application code, and review and revision of the initial draft of the paper. F.J. and S.S. took charge of the knowledge base algorithm implementation and knowledge base construction. Y.W. was in charge of the implementation of knowledge base application code and paper writing. All authors reviewed the manuscript.

Acknowledgements

This class file was developed by Sunrise Setting Ltd, Brixham, Devon, UK.

Website: <http://www.sunrise-setting.co.uk>

References

- Kopka H and Daly PW (2003) *A Guide to L^AT_EX*, 4th edn. Addison-Wesley.
- Lamport L (1994) *L^AT_EX: a Document Preparation System*, 2nd edn. Addison-Wesley.
- Mittelbach F and Goossens M (2004) *The L^AT_EX Companion*, 2nd edn. Addison-Wesley.
- Food and Agriculture Organization of the United Nations (2025) *AGROVOC [Linked Open Data set]*. <https://www.fao.org/agrovoc/>.
- International Health Terminology Standards Development Organisation (2025) *SNOMED CT [Clinical terminology dataset]*. <https://www.snomed.org/snomed-ct>.
- European Union (2025) *IATE (Interactive Terminology for Europe) [Multilingual terminology database]*. <https://iate.europa.eu/home>.
- EuroTermBank Consortium (2025) *EuroTermBank [Multilingual European terminology bank]*. <https://www.eurotermbank.com>.
- U.S. National Library of Medicine (2025) *MeSH [Medical Subject Headings database]*. <https://www.nlm.nih.gov/mesh/meshhome.html>.
- Bloomfield L (1926) *A set of postulates for the science of language. Language*, **2**:153–164.
- Dong Z, Dong Q, Hao C (2010) *HowNet and its computation of meaning. Coling 2010: Demonstrations*, 53–56 (Beijing: Coling 2010 Organizing Committee, August 2010).
- Zhu Z Y, Sun J H (2013) *Improved vocabulary semantic similarity calculation based on HowNet. J. Comput. Appl.*, **33**(8):2276–2279.
- Nie H, Zhou J, Wang H, et al. (2020) *Word similarity computing based on HowNet and synonymy thesaurus*. In: *Intelligent Systems and Applications: Proceedings of the 2019 Intelligent Systems Conference (IntelliSys) Volume 2*, 292–305 (Springer International Publishing).
- Zhang P, Zhang Z, Zhang W, et al. (2014) *Semantic similarity computation based on multi-feature combination using HowNet. J. Softw.*, **9**(9):2461–2466.
- Wu B, Yang J, He L (2012) *Chinese HowNet-based multi-factor word similarity algorithm integrated of result modification*. In: *Neural Information Processing: 19th International Conference, ICONIP 2012, Doha, Qatar, November 12–15, 2012, Proceedings, Part V 19*, 256–266 (Springer Berlin Heidelberg).

- Qi F, Huang J, Yang C, et al. (2019) *Modeling semantic compositionality with sememe knowledge*. *Proc. 57th Annu. Meet. Assoc. Comput. Linguist.*, 5706–5715 (Florence, Italy: Assoc. Comput. Linguist., July 28–August 2, 2019). <https://doi.org/10.18653/v1/P19--1571>.
- Zhang Y, Gong L, Wang Y (2005) *Chinese word sense disambiguation using HowNet*. In: *Advances in Natural Computation: First International Conference, ICNC 2005, Changsha, China, August 27–29, 2005, Proceedings, Part I 1*, 925–932 (Springer Berlin Heidelberg).
- Wang J, Liu J, Zhang P (2008) *Chinese word sense disambiguation with PageRank and HowNet*. *Third International Joint Conference on Natural Language Processing (IJCNLP)*, 39–44 (Hyderabad, India: Assoc. Comput. Linguist.).
- Jiang X, Qiu L, Li Y (2014) *A Tibetan word sense disambiguation method based on HowNet and Chinese-Tibetan parallel corpora*. *Trustworthy Computing and Services, International Conference (ISCTCS)*, 152–159 (Beijing, China: Springer).
- Hou B, Qi F, Zang Y, et al. (2020) *Try to substitute: An unsupervised Chinese word sense disambiguation method based on HowNet*. *Proc. 28th Int. Conf. Comput. Linguist.*, 1752–1757.
- Zhou Q, Yue G, Meng Y (2019) *Incorporating HowNet-based semantic relatedness into Chinese word sense disambiguation*. *Workshop on Chinese Lexical Semantics*, 359–370.
- Niu Y, Xie R, Liu Z, et al. (2017) *Improved word representation learning with sememes*. *Proc. 55th Annu. Meet. Assoc. Comput. Linguist. (Vol. 1: Long Papers)*, 2049–2058 (Vancouver, Canada: Assoc. Comput. Linguist., July 30–August 4, 2017). <https://doi.org/10.18653/v1/P17-1187>.
- Li Z, Ding N, Liu Z, et al. (2019) *Chinese relation extraction with multi-grained information and external linguistic knowledge*. *Proc. 57th Annu. Meet. Assoc. Comput. Linguist.*, 4377–4386 (Florence, Italy: Assoc. Comput. Linguist., July 28–August 2, 2019). <https://doi.org/10.18653/v1/P19-1437>.
- Zhang P Y (2013) *A HowNet-based semantic relatedness kernel for text classification*. *TELKOMNIKA Indones. J. Electr. Eng.*, **11**(4):1909–1915.
- Ding N, Li Z, Liu Z, et al. (2019) *Event detection with trigger-aware lattice neural network*. *Proc. 2019 Conf. Empirical Methods Nat. Lang. Process. 9th Int. Joint Conf. Nat. Lang. Process. (EMNLP-IJCNLP)*, 347–356 (Hong Kong, China, November 3–7, 2019). <https://doi.org/10.18653/v1/D19-1034>.
- Zeng X, Yang C, Tu C, et al. (2018) *Chinese LIWC lexicon expansion via hierarchical classification of word embeddings with sememe attention*. *Proc. AAAI Conf. Artif. Intell.*, **32**(1) (New Orleans, LA, USA: AAAI, February 2–7, 2018). <https://doi.org/10.1609/aaai.v32i1.11646>.
- Xianghua F, Guo L, Yanyan G, et al. (2013) *Multi-aspect sentiment analysis for Chinese online social reviews based on topic modeling and HowNet lexicon*. *Knowledge-Based Systems*, **37**:186–195.
- Fu X, Liu G, Guo Y, et al. (2011) *Multi-aspect blog sentiment analysis based on LDA topic model and HowNet lexicon*. In: *International Conference on Web Information Systems and Mining*, 131–138 (Berlin, Heidelberg: Springer Berlin Heidelberg).
- Yan J, Bracewell D B, Ren F, et al. (2007) *Semi-automatic construction of an emotion ontology using HowNet*. *Artificial Intelligence and Pattern Recognition*, 17–21.
- Liu B, Ren F, Wang C (2008) *The building of Chinese emotion thesaurus using HowNet based on the main sememe*. *2008 Fourth International Conference on Natural Computation*, **6**:91–95 (IEEE).
- Zang Y, Qi F, Yang C, et al. (2020) *Word-level textual adversarial attacking as combinatorial optimization*. *Proc. 58th Annu. Meet. Assoc. Comput. Linguist.*, 6066–6080 (Seattle, WA, USA: Assoc. Comput. Linguist., July 5–10, 2020). <https://doi.org/10.18653/v1/2020.acl-main.540>.
- Wilkinson M D, et al. (2016) *The FAIR guiding principles for scientific data management and stewardship*. *Scientific Data*, **3**:1–9. <https://doi.org/10.1038/sdata.2016.18>.
- (2003) *Terms in aviation science and technology*. *Chinese Sci. Technol. Terms J.*, (4):6–9.
- (2009) *Atmospheric science terms*. *China Terminology*, (2):16–18.
- (2003) *Chinese terms in mechanical engineering (Second Edition)*. *Chinese Sci. Technol. Terms J.*, (2):30–33.
- (2019) *Chinese terms in computer science and technology*. *China Terminology*, (2):10.

- (2008) *Chinese terms in electric power*. *China Terminology*, (6):8–11.
- Zhao W X, Zhou K, Li J, et al. (2023) *A survey of large language models*. *arXiv Preprint*, arXiv:2303.18223.
- Richardson S D, Dolan W B, Vanderwende L (1998) *Mind-Net: acquiring and structuring semantic information from text*. *COLING 1998 Vol. 2: 17th Int. Conf. Comput. Linguist.*.
- Shen S, Wang P, Qian H, Li Z (2026) *Adaptive nearest neighbor sememe tree generation model*. In: *Neural Information Processing (ICONIP 2024), Lect. Notes Comput. Sci.*, **15295** (Mahmud M, Doborjeh M, Wong K, Leung A C S, Doborjeh Z, Tanveer M (eds); Springer, Singapore). https://doi.org/10.1007/978-981-96-6603-4_15.
- JSONLint (2026) *JSONLint: a JSON validation and formatting tool*. *JSONLint Official Website*. <https://jsonlint.com>.
- JSON Schema (2026) *JSON Schema: official specification and documentation*. *JSON Schema Official Website*. <https://json-schema.org>.
- Hripcsak G, Rothschild A (2005) *Agreement, the F-measure, and reliability in information retrieval*. *J. Am. Med. Inform. Assoc.*, **12**(3):296–298.