ChainNet: Structured Metaphor and Metonymy in WordNet

Rowan Hall Maudslay[1,2,3] Simone Teufel[1] Francis Bond[4] James Pustejovsky[5]
[1]University of Cambridge [2]Magdalene College [3]The Alan Turing Institute
[4]Palacký University [5]Brandeis University
rh635@cam.ac.uk sht25@cam.ac.uk bond@ieee.org jamesp@brandeis.edu

Abstract

The senses of a word exhibit rich internal structure. In a typical lexicon, this structure is overlooked: a word's senses are encoded as a list without inter-sense relations. We present ChainNet, a lexical resource which for the first time explicitly identifies these structures. ChainNet expresses how senses in the Open English Wordnet are derived from one another: every nominal sense of a word is either connected to another sense by metaphor or metonymy, or is disconnected in the case of homonymy. Because WordNet senses are linked to resources which capture information about their meaning, ChainNet represents the first dataset of grounded metaphor and metonymy.

Keywords: WordNet, ChainNet, metaphor, metonymy, polysemy, homonymy, chaining

1. Introduction

A fundamental feature of language is **polysemy**: in different contexts words exhibit different meanings. Consider these invocations of the word *march*:

(1) *March* always has dreadful rainfall!

(2) The infantry performed a militant *march*.

In example (1) *march* refers to a month of the year, while in (2) *march* refers to the act of walking in a regimented way. Because these meanings are semantically unrelated, they are said to exhibit **homonymy**. By contrast, word meanings can also be extended productively:

(3) The *march* approached the town hall.

This meaning of *march* is closely related to the meaning of *march* in (2), but the semantic type (Pustejovsky, 1995) has changed: while in example (2) *march* refers to an act, in (3) it refers to a group of people who are engaged in this act together. This kind of meaning change is known as **metonymy**. Words can also create new meanings according to other processes, such as **metaphor**. A metaphor occurs when, in a creative shift, a word is used in an entirely new setting:

(4) Nothing can stop the *march* of science.

In (4), the imagery of (3) is used metaphorically to describe the steady progress of an academic discipline. Taken together, these examples illustrate how word meaning can be successively extended: (2) was extended to (3), which in turn was extended to (4). This is called **chaining** (Lakoff, 1987). Chaining can cause word meaning to extend into disjointed areas of semantic space (Austin, 1961, p. 72): the academic progress in (4) and the regimented walking in (2) have little in common, even though they are connected in a chain.

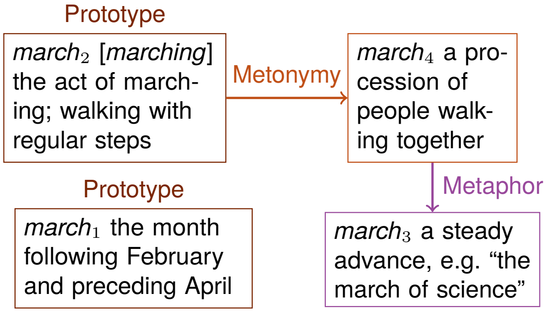

Prototype
march$_2$ [*marching*] the act of marching; walking with regular steps
Metonymy
march$_4$ a procession of people walking together
Metaphor
Prototype
march$_1$ the month following February and preceding April
march$_3$ a steady advance, e.g. "the march of science"

Figure 1: The first four senses of *march* in ChainNet

We present **ChainNet**, a new resource of English metaphor, metonymy, and homonymy. To formalise chaining, we use a lexicon. In a typical lexicon a word is represented by a quasi-ordered list of senses which it can be used to evoke. For example, below are the first four nominal senses of *march* from WordNet 3.0 (Fellbaum, 1998), each corresponding to one of the previous examples:

march$_1$ the month following February

march$_2$ [*marching*] the act of marching; walking with regular steps (especially in a procession of some kind), e.g. "it was a long march"

march$_3$ a steady advance, e.g. "the march of science", "the march of time"

march$_4$ a procession of people walking together, e.g. "the march went up Fifth Avenue"

In ChainNet, the underlying structure of these senses is revealed (Figure 1): every sense is either a prototypical sense, or it is derived from another sense by metaphor or metonymy.

Unlike other metaphor resources, ChainNet provides a representation of the meaning change that is caused by each metaphor. This takes the form of a **feature transformation**. More specifically, features are associated with the sense that a metaphor

LREC-COLING 2024, pages 2984–2996
20-25 May, 2024. © 2024 ELRA Language Resource Association: CC BY-NC 4.0

extends, and the transformation from this sense to a metaphor can then be traced by parallel changes in features. For the "*march* of science" metaphor (*march*$_3$), the feature transformation is:

Features of *march*$_4$		**Features of *march*$_3$**
gradually advances	→	gradually advances
is a group of people	→	~~is a group of people~~

While the first feature of *march*$_4$ is maintained, the second feature is lost by the metaphorical extension. This information can be used in combination with other multimodal resources already linked to WordNet (including images) to capture the meaning transfer of each metaphor in ChainNet.

We produce ChainNet annotations for every nominal sense of 6500 words in WordNet, for a total of $22{,}178$ senses. To collect this annotation, we employed three annotators, who we also used for an agreement study (§4). We establish two baselines for the task of synthesising ChainNet annotations (§5). Finally, we discuss the lessons learned from applying the theory of chaining (Lakoff, 1987) to real-world data (§6).

2. Existing Resources

We are not aware of any computational resource built for research on chaining. Most chaining theories are supported only by toy examples (Wittgenstein, 1953; Austin, 1961; Lakoff, 1987), and to the best of our knowledge no chaining formalism has been applied to large amounts of real-world data.

Metaphor resources are more common. Most consist of sentences in which words are labelled for metaphoricity. The best-known example is the VUAMC (Steen et al., 2010), which was annotated based on the Metaphor Identification Procedure (MIP; Crisp et al., 2007). The VUAMC has been used in two shared tasks to build models that identify metaphorical words (Leong et al., 2018, 2020). However, the main drawback of resources like the VUAMC is that they do not provide any representation of metaphor meaning. As a result, metaphor understanding is understudied.

Instead of analysing tokens, an alternative approach is to analyse senses in a lexicon. It is impossible for a lexicon to enumerate all possible metaphors (Black, 1962, 1977), but many lexica do nevertheless contain examples of **conventional metaphor**. These are metaphors that have been widely adopted by a language community.

WordNet (Miller, 1995) is a popular computational lexicon. WordNet contains senses that correspond to conventional metaphors (e.g. *march*$_3$ in §1), but does not systematically identify which senses these are. Sometimes a sense's gloss mentions that it is metaphorical, but the choice to include this information was left to the discretion of the lexicographers. Consequently, most metaphors are unlabelled, and metonymy is not labelled at all.

WordNet is well positioned for research on chaining and metaphor understanding. This is because WordNet senses are linked to multimodal data that captures their meaning, in the form of images (e.g. ImageNet, Deng et al., 2009) and textual usages (see Petrolito and Bond, 2014). WordNet could also be used for **conceptual metaphor theory** (CMT; Lakoff and Johnson, 1980) research, as has been suggested by Alonge and Castelli (2002) and Lönneker (2003). A conceptual metaphor is a metaphor pattern that affects multiple words (e.g. ARGUMENT IS WAR, affecting words like *attack* and *defend*). Currently the main resource used for CMT research is the Master Metaphor List (Lakoff et al., 1991), but this list is incomplete and is not designed for computational work. An ongoing effort to address this is MetaNet (Petruck, 2018) but this is not openly available. The benefits of using WordNet to study conceptual metaphor could also apply to **logical metonymy**, which is the equivalent term used to describe metonymy patterns (Pustejovsky, 1995).

Some sections of WordNet have been annotated for metaphoricity, but only a few word types are covered: 440 in Mohammad et al. (2016) and 198 in Maudslay and Teufel (2022b). We are not aware of any resource (based on WordNet or otherwise) which systematically identifies metaphor or metonymy through the lexis, and which is also computationally tractable and open. An exception is the Pattern Dictionary of English Verbs (Pustejovsky et al., 2004; Hanks and Pustejovsky, 2005; Hanks, 2013), which identifies semantic type patterns of English verbs but is not connected to WordNet.

3. The ChainNet Schema

ChainNet is a new resource that formalises chaining. It is built as an annotation layer for the Open English Wordnet (McCrae et al., 2019), itself an extension to Princeton WordNet 3.0 (Fellbaum, 1998).

3.1. Three Types of Sense

In ChainNet, every nominal sense of a word is either a prototypical sense or is derived from another sense. We permit only two derivation types: metaphor and metonymy. The idea that metaphor and metonymy are the two axes of meaning extension goes back at least to Jakobson (1956), who saw metaphor as a "vertical" relation based on selection, substitution, and similarity, and saw metonymy as a "horizontal" relation based on combination, contexture, and contiguity. Various authors have proposed further subdivisions to this classification, adding categories for synecdoche (e.g. Bloomfield, 1933; Blank, 1999), folk-etymology

(Ullmann, 1962; Blank, 1999), specialisation and generalisation (Bloomfield, 1933; Blank, 1988), and so on. For the first iteration of ChainNet, however, we choose to empirically test the descriptive power of a simple metaphor–metonymy dichotomy.

The output for each word is therefore a rooted tree (where the prototypical sense is the root), or multiple disjoint trees in the case of homonymous words (as these have multiple roots). A partial example for *march* was shown in Figure 1; in the spirit of Jakobson, we depict metaphor as a vertical relation and metonymy as a horizontal relation.[1]

3.1.1. Prototypes

In ChainNet every word has a prototypical sense, which is a central sense from which other senses expand.[2] Following MIP, we define a prototypical sense as a "basic" sense of a word. For a group of related senses, only one is prototypical. To determine which, we create criteria that annotators can apply. A prototypical sense is usually:

1. **Evoked out of context.** A prototypical sense is usually what comes to mind when a word is heard on its own, without additional context.
2. **Referenced by the unigram.** A prototypical sense is usually referenced using a word on its own, rather than in a multi-word phrase. For example, *apple* can refer to either a fruit (*apple*$_1$) or a tree which bears that fruit (*apple*$_2$). To refer to the fruit, speakers would likely use the word alone, and say e.g. "I ate the *apple*" rather than "I ate the *apple fruit*". The inverse is true for the tree: "I watered the *apple tree*" is more likely than "I watered the *apple*". Because of this, the fruit (*apple*$_1$) is prototypical.
3. **Related to the physical world.** A prototypical sense is usually concrete or imageable.
4. **Related to human experience.** A prototypical sense is often related to embodied life.
5. **Historically older than other senses.** A prototypical sense usually predates other senses in terms of their etymology.

A word can have multiple prototypical senses, but this only occurs when the word exhibits homonymy. An example is *bridge*, which refers to a crossing over a river (*bridge*$_1$) and a card game (*bridge*$_5$):

bridge$_1$ [*span*] a structure that allows people or vehicles to cross an obstacle such as a river

bridge$_5$ any of various card games based on whist

These senses have different etymological origins and are unrelated, so they are both prototypical.

Our five criteria resemble other guidelines (e.g. MIP or Evans, 2005, p. 44). However, a notable point of difference is that we tell annotators explicitly to prioritise a sense which is not historically the oldest (i.e. to violate criteria 5) if this would be the prototypical sense in the minds of contemporary speakers. For example, *train* can refer to a locomotive (*train*$_1$) or a procession of camels (*train*$_3$):

train$_1$ [*railroad train*] public transport provided by a line of railway cars coupled together and drawn by a locomotive, e.g. "express trains don't stop at Princeton Junction"

train$_3$ [*caravan*, *wagon train*] a procession (of wagons or mules or camels) traveling together in single file, e.g. "they joined the wagon train for safety"

The second sense predates the first, but most people today would think of the locomotive, so it is the prototypical sense. Similar logic applies when determining whether two senses are homonymous. Consider a third sense of *bridge*:

bridge$_9$ [*bridge deck*] an upper deck where a ship is steered and the captain stands

This sense originated on paddle steamers which had a bridge across the deck: it is derived from *bridge*$_1$. However, today most speakers would not perceive a relation between *bridge*$_9$ and *bridge*$_1$, so *bridge*$_9$ is also a prototypical sense. In this way, ChainNet focuses explicitly on cognitive versions of linguistic phenomena rather than etymological versions, while other resources conflate the two.

3.1.2. Metaphors

Metaphor occurs when one sense resembles another, often in an abstract way. Consider *neck*:

neck$_1$ [*cervix*] the part of an organism (human or animal) that connects the head to the rest of the body, e.g. "he admired her long graceful neck", "the horse won by a neck"

neck$_2$ a narrow elongated projecting strip of land

neck$_4$ a narrow part of an artifact that resembles a neck in position or form, e.g. "the banjo had a long neck", "the bottle had a wide neck"

The prototypical sense (*neck*$_1$) refers to the part of the body connecting the head to the torso. The other senses *neck*$_2$ (a long strip of land) and *neck*$_4$ (a narrow part of object) are both "like" *neck*$_1$ (they are long and thin), and so they are connected by metaphor. This is shown in Figure 2.

[1] Any words in square brackets at the start of a definition correspond to the synonyms in WordNet synsets.

[2] We adopt the term "prototypical sense" following Lakoff (1987), who himself adapted it from Rosch (1973). Other terms that are sometimes used include "nuclear sense" (Austin, 1961), "core sense" (e.g. Pedersen et al., 2022), "sanctioning sense" (e.g. Evans, 2005), or "basic sense" (e.g. Copestake and Briscoe, 1995).

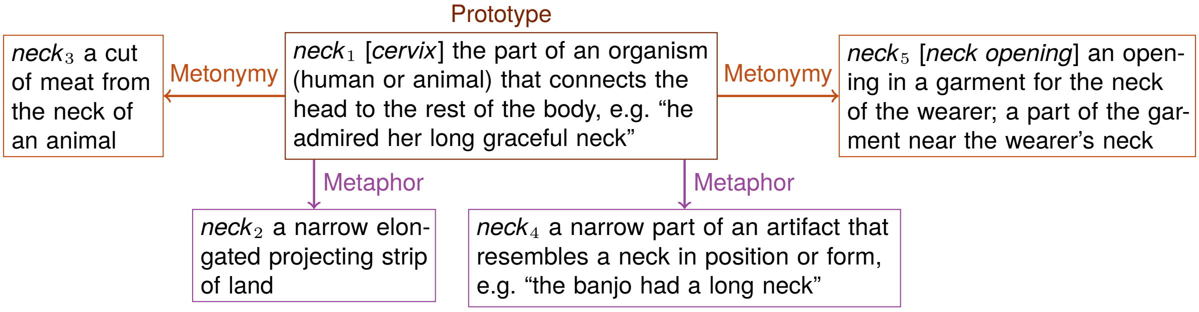

Prototype
neck$_1$ [*cervix*] the part of an organism (human or animal) that connects the head to the rest of the body, e.g. "he admired her long graceful neck"
Metonymy
neck$_3$ a cut of meat from the neck of an animal
Metonymy
neck$_5$ [*neck opening*] an opening in a garment for the neck of the wearer; a part of the garment near the wearer's neck
Metaphor
neck$_2$ a narrow elongated projecting strip of land
Metaphor
neck$_4$ a narrow part of an artifact that resembles a neck in position or form, e.g. "the banjo had a long neck"

Figure 2: The complete ChainNet structure of *neck*

We operationalise metaphor as a relation between a pair of senses, in order to capture how metaphor connects one meaning to another. This is a divergence from existing resources, which assign a binary label (either metaphorical or literal) to individual senses (e.g. Mohammad et al., 2016; Maudslay and Teufel, 2022b) or to individual tokens (e.g. the VUAMC). This approach to annotation often creates problems, because it does not account for chaining. For example, if sense C was a metaphorical extension of B, and B was itself a metaphorical extension of A, then B would be both "metaphorical" (compared to A) and "literal" (compared to C). These cases can be represented in ChainNet, which has no notion of literal senses.

3.1.3. Metonymies

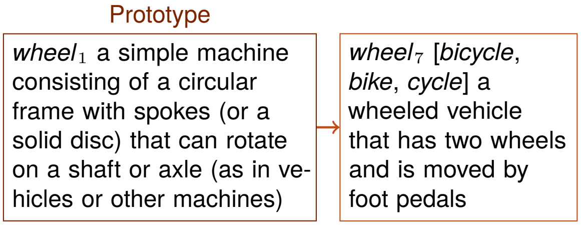
Metonymy occurs when a sense is related to another, but has a different semantic type (Pustejovsky, 1995). Consider two more senses of *neck*:

neck$_3$ a cut of meat from the neck of an animal

neck$_5$ [*neck opening*] an opening in a garment for the neck of the wearer; a part of the garment near the wearer's neck

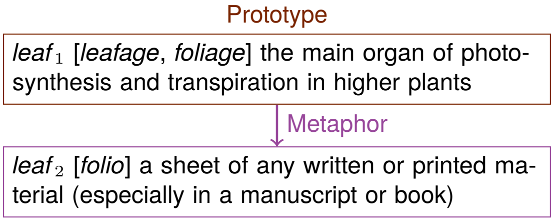
These senses are not "like" *neck*$_1$, but they are related to it in other ways. The sense *neck*$_3$ is "meat from" *neck*$_1$, and *neck*$_5$ is "a hole in clothing for" *neck*$_1$: they are metonyms (see Figure 2).

The full description of metonymy that we give to annotators is broad. It encompasses logical metonymy schemas (Pustejovsky, 1995), such as count–mass alternations (*neck*$_1$ to *neck*$_3$) or figure–ground reversals (*neck*$_1$ to *neck*$_5$). It also encompasses synecdoche, which is when the word for part of something is used to refer to the whole of it, for example when *wheels* refers to a whole vehicle:


Prototype
wheel$_1$ a simple machine consisting of a circular frame with spokes (or a solid disc) that can rotate on a shaft or axle (as in vehicles or other machines)
wheel$_7$ [*bicycle*, *bike*, *cycle*] a wheeled vehicle that has two wheels and is moved by foot pedals

3.2. Recording Metaphor Meanings

ChainNet provides an explanation of how each metaphor modifies the meaning of the sense it extends. The way that explanations are encoded is based on **slippage** (Hofstadter, 1995). Consider *leaf*, which refers to a plant organ (*leaf*$_1$) or a page in a book (*leaf*$_2$):


Prototype
leaf$_1$ [*leafage*, *foliage*] the main organ of photosynthesis and transpiration in higher plants
Metaphor
leaf$_2$ [*folio*] a sheet of any written or printed material (especially in a manuscript or book)

The metaphorical extension *leaf*$_2$ has some features in common with *leaf*$_1$ (e.g. they are both flat), but it is missing an important feature (it is not part of a plant): this feature has "slipped". For every metaphor, ChainNet records features that are maintained, and features that slip.

3.2.1. Procedure

Suppose we have a pair of senses linked by metaphor, which for convenience we refer to here as the literal sense (e.g. *leaf*$_1$) and the metaphorical sense (*leaf*$_2$). Annotators first add features to the literal sense. These features are not intended to be a set of necessary and sufficient conditions, but instead are meant to capture specific nuances of meaning that are relevant to the metaphorical transformation. Features are given in natural language, and are elicited from the annotator with a sentence fragment: `This thing ___`. In the case of *leaf*, features of the literal sense might be `is flat`, `photosynthesises`, and `is part of a plant`.

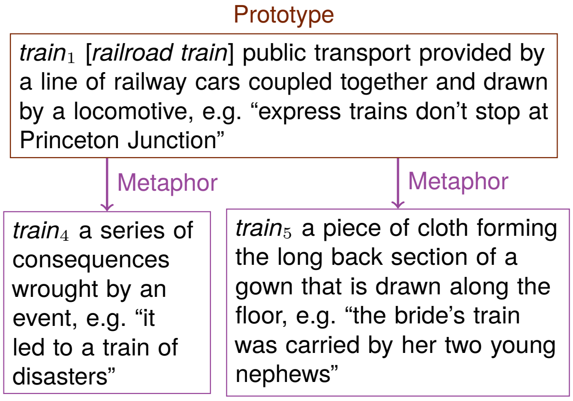
For the metaphorical sense, annotators then decide whether each of those features is kept, lost, or modified. For the *leaf* example, a complete feature transformation (with one kept feature, one lost feature, and one modified feature) could be:

Features of *leaf*$_1$		**Features of *leaf*$_2$**
`is flat`	→	`is flat`
`photosynthesises`	→	~~`photosynthesises`~~
`is part of a plant`	→	`is part of a book`

At minimum, every metaphor has one modified feature, or one kept feature plus one lost feature.

3.2.2. Multiple Metaphorical Extensions

When multiple metaphors extend the same sense, they each transform the same set of features. Consider again *train*, which can refer metaphorically to a series of events (*train*$_4$) or a bride's gown (*train*$_5$):



train$_1$ [*railroad train*] public transport provided by a line of railway cars coupled together and drawn by a locomotive, e.g. "express trains don't stop at Princeton Junction"

train$_4$ a series of consequences wrought by an event, e.g. "it led to a train of disasters"

train$_5$ a piece of cloth forming the long back section of a gown that is drawn along the floor, e.g. "the bride's train was carried by her two young nephews"

Suppose an annotator has given *train*$_1$ the features:

```
is a mode of transport
is a series of carriages
is pulled from the front
```

Each metaphor transforms these features differently. The first metaphor, *train*$_4$ (consequences), builds on the sequentiality of *train*$_1$:

→ ~~is a mode of transport~~
→ is a series of consequences
→ ~~is pulled from the front~~

The second metaphor, however, is explained by a very different slippage. A *train*$_5$ (gown) is pulled from the front by the bride, similar to how a *train*$_1$ is drawn from the front by an engine carriage:

→ ~~is a mode of transport~~
→ ~~is a series of carriages~~
→ is pulled from the front

In this way, slippages distinguish between the different types of metaphor that act on the same word.

3.3. Making Alterations to the Lexicon

In rare cases, errors or omissions in WordNet make it difficult to apply our annotation schema. In these situations, we allow annotators to modify the senses of a word using two operations.

Splitting Senses Sometimes, a sense conflates a metaphorical and a non-metaphorical sense:

birth$_1$ the time when something begins (especially life), e.g. "they divorced after the birth of the child", "his election signaled the birth of a new age"

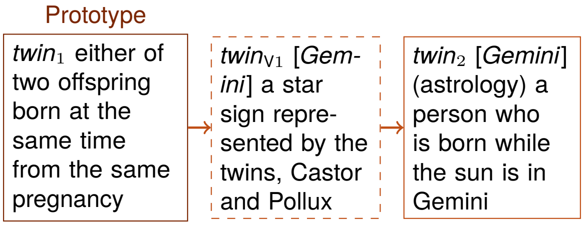
This sense covers both a metaphorical sense of *birth* ("the *birth* of a new age"), and a literal sense ("the *birth* of the child"). To fix this, annotators can split the sense into two separate senses: a non-metaphorical sense *birth*$_{1\mathrm{A}}$, and a metaphorical sense *birth*$_{1\mathrm{B}}$. They then edit the definition in each case to make the distinction clear, and annotate each half in isolation.

Virtual Senses Sometimes a chain exists but cannot be expressed because an intermediate sense is missing. Consider the word *twin*:

twin$_1$ either of two offspring born at the same time from the same pregnancy

twin$_2$ [*Gemini*] (astrology) a person who is born while the sun is in Gemini

The second sense relates to the fact that the Gemini star sign is represented by the twins, Castor and Pollux. However, in WordNet *twin* does not have a Gemini star sign sense. To encode the chain, annotators have the option to add an additional virtual sense, which is a new sense, with a definition provided by the annotator:



twin$_1$ either of two offspring born at the same time from the same pregnancy

twin$_{\mathrm{V}1}$ [*Gemini*] a star sign represented by the twins, Castor and Pollux

twin$_2$ [*Gemini*] (astrology) a person who is born while the sun is in Gemini

4. Collecting Annotations

4.1. Guidelines and Interface

To collect ChainNet annotations, we produced in-depth annotation guidelines and developed a web-based graphical user interface.[3] By default, the interface restricts the types of chain that it is possible to create: metonyms can only extend prototypical senses, and metaphors cannot extend other metaphors. However, this constraint can be overridden by marking a sense as a "conduit", which allows any sense to connect to it. We added this soft constraint as we found in preliminary work that it biased annotators towards shallower trees, which improved agreement. Further details are in App. B.

4.2. Data

To select words for annotation, we first filtered WordNet to only include words with 2–10 nominal senses.

[3] In resources given to annotators, we adapt some of the nomenclature. Metonyms are referred to as "associations", because we found in preliminary work that the similarity in form of "metaphor" and "metonymy" increased annotation errors. Prototypical senses are referred to as "core senses", in order to avoid specialist terminology.

Monosemous words have only one solution (a single prototypical sense), while words with greater than 10 senses were deemed to require too great a cognitive load. This is because the number of possible annotation options increases exponentially with the number of senses, as shown in the middle column of Table 1. We also excluded single-letter wordforms, wordforms which included whitespace or hyphens, and words whose senses were all proper nouns. We then sampled words from the remaining set according to their frequency as nouns, which we estimated by POS-tagging Wikipedia.

4.3. Procedure

We employed three annotators, all undergraduate linguists who are native speakers of British English. They were selected on the basis of a screening task. Annotation consisted of three main phases: training (100 words each), bulk 1 (500 each), and bulk 2 (1000 each). After this, annotators bulk-annotated a further 1900 words between them.

To evaluate the annotation quality, we measure inter- and intra-annotator agreement. Inter-annotator agreement evaluates the degree to which different annotators produce the same annotation. To measure inter-annotator agreement, 100 shared words were distributed throughout the three main phases. Intra-annotator agreement, on the other hand, measures the consistency of annotation within a single mind. Intra-annotator agreement can be measured by asking an annotator to re-annotate material they have annotated already, after a suitable time period: at the end of the main phases, we gave each annotator a set of 100 words which they had annotated 6 weeks prior. It stands to reason that tasks that cannot be held consistent within a single mind are even harder to reliably convey to others, so typically intra-annotator agreement is higher than inter-annotator agreement.

The end result is a dataset with 6500 annotated words, of which 100 are triply annotated, and 300 are doubly annotated by one annotator. WordNet coverage is shown in Table 1: ChainNet covers a high percentage of words with many senses (94% of words with 5–10 senses), but for the long tail of low-frequency words with few senses, coverage decreases.

4.4. Qualitative Findings

We held individual meetings with annotators at the end of each annotation phase to discuss their annotation. The most common source for disagreement was the choice of prototypical sense(s), which could lead to radically different overall annotations. These disagreements usually resulted when WordNet had senses that were so similar that choosing which to make prototypical was arbitrary.

# senses	# options	# words annotated
2	5	2634 / 10259 (26%)
3	49	1502 / 2988 (50%)
4	729	1052 / 1178 (89%)
5	146×10^2	579 / 620 (93%)
6	371×10^3	284 / 306 (93%)
7	114×10^5	204 / 212 (96%)
8	410×10^6	93 / 94 (99%)
9	170×10^8	95 / 97 (98%)
10	794×10^9	57 / 60 (95%)

Table 1: The number of senses of a word; the number of possible annotations per word; the number of annotated words out of the total in WordNet

Another common source of disagreement was idiolectical variation. Consider *host*, which refers to a person (*host*$_1$) or Communion wafer (*host*$_9$):

host$_1$ a person who invites guests to a social event (such as a party in his or her own home) and who is responsible for them while they are there

host$_9$ a technical name for the bread used in the service of Mass or Holy Communion

Two of the annotators labelled both of these senses as prototypical: to them, these senses were homonymous. However, the third annotator had been told at Catholic school that Communion bread 'hosted' the Holy Spirit: to them, *host*$_9$ was a metaphorical extension of *host*$_1$. This alternative interpretation was clearly represented in the feature transformation the annotator recorded.

4.5. Agreement Measures and Results

ChainNet data is complex, and not amenable to standard agreement measures. We use a multi-faceted approach.[4]

Homonymy Agreement If a word has multiple prototypical senses, then its senses are implicitly partitioned into disjointed clusters. We can investigate the degree to which annotators agree on homonymy by investigating whether their partitions align. For each word we compute the Adjusted Rand Index (ARI; Hubert and Arabie, 1985) between the partitionings of each pair of annotators, and take the average over all words and all pairs. ARI is chance-corrected: 0 indicates a random partitioning, while 1 indicates perfect agreement. For inter-annotator agreement, the ARI was $.84$, while for intra-annotator agreement it was $.93$.

[4] For our inter-annotator agreement study, we include the annotation of all three annotators, as well as gold-standard annotation produced by one of the authors. For both inter- and intra-annotator agreement, we excluded wordforms which annotators indicated they did not know.

		Percentage			Kappa		
		All	*AP*	*AC*	*All*	*AP*	*AC*
Inter	Prot.	82			.61		
	Metap.	77	88	89	.51	.78	.79
	Meton.	80	88	89	.51	.72	.74
	Any	70	88	89	.54	.84	.85
Intra	Prot.	88			.75		
	Metap.	86	92	94	.69	.83	.87
	Meton.	87	92	94	.64	.78	.81
	Any	81	92	94	.70	.88	.90

Table 2: Label Agreement

Label Agreement To investigate whether annotators agree on the labels assigned to senses, we compute the mean pairwise percentage agreement, as well as Fleiss' Kappa.[5] Measures for categorical classification such as Kappa require independence between datapoints. This does not hold for our labels, but Kappa is routinely used in the dependency parsing literature where the same issue applies: we report Kappa to allow comparison to that literature. In addition to reporting agreement for all senses ("All"), we additionally report the agreement when we filtered the data to only include a word if the annotators agreed on the choice(s) of prototype (Agree Prototype; "AP"), or alternatively to only include a sense if the annotators agreed on which other sense it was connected to (Agree Connections; "AC"). Results are presented in Table 2.

Connection Agreement To investigate whether annotators agreed on the connections between senses, we first compute the overall percentage of unlabelled connections that the annotators agreed on. The direction of connections emerges from the choice of prototype: to avoid cascading errors, we treat connections as undirected. This is therefore equivalent to the undirected unlabelled attachment score (UUAS; see Gómez-Rodríguez et al., 2015). Once again, we additionally compute results when the data is filtered to only include words for which the annotators agreed on the choice(s) of prototype (AP). Finally, we compute the percentage agreement over labels and connections jointly (i.e. the undirected labelled attachment score, ULAS). Results are presented in Table 3.

4.6. Agreement Analysis

In our qualitative findings (§4.4) we observed that differences in prototype selection could lead to radically different annotations, reducing agreement. This observation is borne out in our empirical results. When all datapoints are considered, the inter-annotator agreement for labels is $\kappa = .54$. However, when the data is filtered to only include words where the annotators agreed on the prototypes (AP), this rises to $\kappa = .84$. This same finding is reflected in connection agreement (UUAS 75% for all vs. 91% for AP). For homonymy the inter-annotator agreement is high $(\mathrm{ARI} = .84)$, suggesting that difficulties in prototype selection did not stem from issues in determining whether or not two senses were related, but rather issues in choosing the best prototype from a group of closely related senses.

	UUAS		ULAS	
	All	*AP*	*All*	*AP*
Inter	75	91	65	82
Intra	82	93	76	88

Table 3: Connection Agreement

When we only include senses where the annotators agreed on connections (AC), label agreement is highest ($\kappa = .85$). This is especially true for metaphor labels, where agreement improves from $\kappa = .51$ to $.79$ under AC. This gives credence to our operationalisation of metaphor as a relation: it is difficult to say whether a meaning of a word is metaphorical without knowing which other meaning it is being compared to, but agreement is high when annotators compare the same meanings.

Knowing the prototypical sense is nearly as good as knowing all the connections ($\kappa = .84$ for AP vs. $.85$ for AC). One explanation for the similarity of these results could be that knowing the prototype(s) is sufficient to recover the connections. This explanation is supported by the connection agreement, which improved from 75% UUAS to 91% in AP: when annotators agreed on the prototypical sense(s), they agreed on the connections.

For the intra-annotator agreement study, the results pattern the same as for inter-annotator agreement, but are higher across the board (e.g. $\kappa = .70$ for unfiltered label agreement compared to $.54$ for inter). This is expected, as this setting will negate the effects of idiolectical variation. However, the ceiling for intra-annotator agreement remains limited by the sense inventory we used.

It is difficult to analyse the agreement of feature transformations because they are encoded in natural language. As an informal experiment, we manually aligned the feature transformations for every pair of senses that two of the annotators agreed exhibited metaphor. The result was positive $(93\%$ of metaphors had some form of overlap in their feature transformations), but we see this result as indicative rather than definitive. Agreement values for virtual senses and split senses are also difficult to estimate, since these were very low frequency occurrences: there are only 19 split and 258 virtual senses in the entire dataset.

[5] We give non-prototypical senses the label of their incoming edge (either metaphor or metonymy).

5. Polysemy Parsing

When presenting a new dataset, it is common to also present a baseline model that attempts to reproduce the collected data. We call this task **polysemy parsing**. For a word w, the goal of polysemy parsing is to produce a parse of its senses $\mathcal{S}_w$, where a parse consists of a label for each sense (prototype, metaphor, or metonym), and a connection for each metaphor and metonym. Slippages, virtual senses, and split senses are not treated.

Polysemy parsing models do not have an obvious general application, but they could be used to synthesise annotations for the rest of the English WordNet, or to generate ChainNets for other languages. To investigate this possibility, we implement two polysemy parsing baselines.

5.1. Two Baseline Models

MPD+MST Our first baseline predicts labels using a metaphorical polysemy detection (MPD; Maudslay and Teufel, 2022b) model, and then predicts connections by finding the minimum spanning tree (MST). For each sense $s \in \mathcal{S}_w$ we retrieve a k-dimensional sense-embedding $\mathbf{e}_s \in \mathbb{R}^k$ (from Scarlini et al., 2020), and pass this through a linear layer and softmax to get probabilities for the three labels. Whichever sense has the highest prototype probability is taken as prototypical; the rest are given whichever label has the highest probability.[6] We compute the MST based on the distances between senses in the embedding space.

Biaffine For a second baseline, we adapt a graph-based dependency parsing model, the biaffine parser (Dozat and Manning, 2017). In dependency parsing, the goal is to find dependencies between words in a sentence. Graph-based parsers learn a function that predicts a score for each possible directed edge between a pair of words; a parse is then extracted by finding the maximum spanning tree over those scores using the Chu–Liu/Edmonds's algorithm (McDonald et al., 2005). To score an edge from a head sense $h \in \mathcal{S}_w$ to a dependent sense $d \in \mathcal{S}_w$, our adapted biaffine parser first retrieves the sense embeddings of those senses, $\mathbf{e}_h$ and $\mathbf{e}_d$. Using a pair of multi-layer perceptrons (MLPs), we process these into k'-dimensional head and dependent representations $\mathbf{h}$ and $\mathbf{d}$ respectively:[7]

$$\mathbf{h} = \underset{\text{head}}{\mathrm{MLP}}(\mathbf{e}_h) \qquad \mathbf{d} = \underset{\text{dep}}{\mathrm{MLP}}(\mathbf{e}_d) \tag{1}$$

	Model	LOS	UUAS	ULAS
1 best	Random	35	41	28
	MPD+MST	**51**	52	**43**
	Biaffine	50	**57**	**43**
n best	Random	42	53	36
	MPD+MST	63	68	55
	Biaffine	**65**	**71**	**57**

Table 4: Baseline Results

The score is then computed using a biaffine function, which consists of a matrix U ($\mathbb{R}^{k'} \times \mathbb{R}^{k'}$), a vector $\mathbf{w}$ ($\mathbb{R}^{2k'}$), and a scalar b ($\mathbb{R}$):

$$\mathrm{biaff}(\mathbf{x}, \mathbf{y}) = \mathbf{x}^\top U \mathbf{y} + \mathbf{w}^\top (\mathbf{x} \circ \mathbf{y}) + b \tag{2}$$

$$\mathrm{score}(p \to s) = \mathrm{biaff}(\mathbf{h}, \mathbf{d}) \tag{3}$$

Because the output for polysemy parsing is a forest (not a tree), we add an additional 'root' sense to the input, and treat prototypical senses as being connected to this root. We embed the root as the mean of the sense embeddings of a word. A second biaffine function with its own parameters is used to predict edge labels. This need only distinguish between metaphor or metonymy, since prototypical senses are implicitly those connected to the root.

5.2. Experimental Setup

For evaluation metrics, we use labelling accuracy (label-only score, LOS), as well as UUAS and ULAS. In addition to evaluating the top prediction of each model ("1 best"), we additionally compare models when n alternative high-scoring parses are considered (where n is the number of senses of a word; "n best").[8] We do this to attempt to control against idiolectical variation in the gold standard which is used as evaluation data. Significance values are computed using a two-tailed Monte Carlo permutation test (r=10,000; α=0.01), with Bonferroni correction. Further details are in App. C.

5.3. Results

Results are presented in Table 4. Both baselines perform significantly better than random in all settings. For the single-output comparison, under LOS and ULAS the two baselines were statistically indistinguishable, but for UUAS, the biaffine baseline performed significantly better than MPD+MST (57 compared to 52). The lower performance of the biaffine parser under LOS and ULAS reflects a weakness in its labelling component, which is trained separately.

[6] This ensures that at least one sense is a prototype.

[7] Because our input is a set of senses (not a list of words), we skip the contextualisation step of the standard biaffine parser, which is typically achieved using an LSTM (Hochreiter and Schmidhuber, 1997).

[8] To find each alternative parse, we removed an edge from the predicted parse, and recomputed the MST without that edge, then evaluated whichever parse from the options scored the highest under UUAS.

When considering n alternative predictions, every result is significantly higher than it was previously. In this setting, the MPD+MST and biaffine baselines are again significantly better than random, and the margin of their advantage is increased. For instance, the MPD+MST baseline gains 12 points on LOS, while a random baseline improves by 7. This is most pronounced for the biaffine parser, which attains 71 points UUAS when n alternative parses are considered, up from 57 in the single-output setting. While the biaffine parser had difficulties predicting a single parse, it performed better when predicting several likely parses.

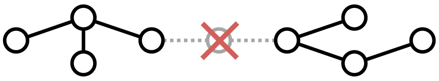
The performance of the biaffine parser may have been limited because it was trained using the out-of-the-box hyperparameters recommended by Dozat and Manning (2017), or because of the alterations we made to adapt it for our task, specifically the lack of a contextualisation step and the way we compute the root embedding. Nevertheless, these results represent a proof-of-concept, and suggest that future work may be able to synthesise ChainNet annotations for the rest of WordNet.

6. Reflections on Chaining

A side effect of ChainNet's construction is that we have brought the theory of chaining (Lakoff, 1987) into contact with a large amount of real-world data. In this section, we comment on two points of interest that have arisen through this process.

Chains Evidence Multistability A word's senses can be connected into chains in multiple valid ways that are sometimes radically different. This is analogous to syntactic ambiguity (e.g. "we saw her duck"), except that for chaining these situations are the rule, rather than the exception. Evidence for this comes from multiple sources. In our inter-annotator agreement study (§4.6), we found that differences in the choice of prototypical sense(s) reduced agreement. In our qualitative analysis (§4.4), we found that disagreements were sometimes caused by idiolectical variation between annotators (e.g. for the word *host*), and that the different annotation versions that resulted were usually internally-consistent. In our evaluation of polysemy parsing baselines (§5), we found that the performance of the biaffine parser improved substantially in a top-n decode setting: the model could identify multiple high-scoring parses, but had difficulties in selecting a single "best" parse. To address the effects of multistability, future ChainNet iterations could provide multiple valid parses for each word. Alternatively, chains could be operationalised as cyclic graphs rather than trees, in which a nexus of senses serves as the prototype as opposed to a single sense.

Cognitive Chains ≠ Etymological Chains We told annotators to prioritise their perception of words, and to override historical reality if necessary (§3.1.1). For an indication of the effect this had, we can compare homonymy in ChainNet to etymological data. More specifically, we can retrieve clusters of interconnected senses from ChainNet, and compare these to clusters that are based on etymological derivation (from Maudslay and Teufel, 2022a). For 17% of words, ChainNet's clusters are different from etymological clusters. In a large majority of these cases (95%), words exhibit homonymy in ChainNet where they do not in the etymological data. That is to say, ChainNet's homonyms are more fine-grained than etymological homonyms: the mean number of clusters per word in ChainNet is 1.23, but in etymological data it is 1.03. One possible cause for this could be chaining. Suppose that a word has a chain of senses that have emerged over time. If an intermediate sense falls out of use, the senses of the word could form two disconnected chain fragments. This situation is illustrated below:



The result is cognitive homonymy, without etymological homonymy. A real-world example of this is *bridge*: the sense *bridge*$_9$ (a deck of a ship) is derived from *bridge*$_1$ (a structure to cross an obstacle), but this relation is no longer commonly perceived (see §3.1.1). We argue that future lexical-semantic resources should clearly distinguish between cognitive and etymological versions of phenomena, since the two do not always align.

7. Conclusion

We present ChainNet, a new dataset of metaphor and metonymy. Rather than identifying instances of these phenomena in text, ChainNet instead reveals the sub-surface complexity of senses in WordNet. In addition to identifying inter-sense relations, ChainNet also provides a natural language explanation for each metaphor, in the form of a feature transformation. This information can be used in tandem with other grounded information from WordNet to study metaphor and metonymy meaning.

ChainNet opens up new modelling possibilities. Multimodal data from WordNet could be used to study metaphor meaning. One direction would be to build cognitive models of metaphor generation, for example by adapting the task formulation of Sun et al. (2021). Another direction would be to investigate whether ChainNet evidences patterns of conceptual metaphor or logical metonymy, for example using clustering algorithms. Future annotation work could extend ChainNet to other languages, or could refine ChainNet with additional relation types.

Data Availability Statement

ChainNet and other accompanying material are freely available at https://github.com/rowanhm/ChainNet.

Ethics Statement

The annotation collection component of this work involved working with participants; for this, we obtained approval from the Ethics Committee of the Department of Computer Science and Technology at the University of Cambridge. Annotators were paid £14.72 per hour of annotation work.

Acknowledgements

The first author would like to thank the three students who carried out the annotation work, although ethical approval requires they remain anonymous. He would also like to thank Magdalene College, who supported him for the duration of this project, and the Alan Turing Institute and Isaac Newton Trust, who together provided the funding that enabled data collection.

Bibliographical References

Antonietta Alonge and Margherita Castelli. 2002. Which way should we go? Metaphoric expressions in lexical resources. In *Proceedings of the Third International Conference on Language Resources and Evaluation (LREC'02)*, Las Palmas, Canary Islands - Spain. European Language Resources Association (ELRA).

John L. Austin. 1961. The meaning of a word. *Journal of Symbolic Logic*, 35(4):23–43.

Max Black. 1962. *Models and Metaphor*. Cornell University Press.

Max Black. 1977. More about metaphor. *Dialectica*, pages 431–457.

Andreas Blank. 1999. Why do new meanings occur? A cognitive typology of the motivations for lexical semantic change. In Andreas Blank and Peter Koch, editors, *Historical Semantics and Cognition*, pages 61–90. De Gruyter Mouton, Berlin, Boston.

Glenn David Blank. 1988. Metaphors in the lexicon. *Metaphor and Symbol*, 3(3):21–36.

Leonard Bloomfield. 1933. *Language*. Allen & Unwin, New York.

Carlo Bonferroni. 1936. Teoria statistica delle classi e calcolo delle probabilita. *Pubblicazioni del R Istituto Superiore di Scienze Economiche e Commericiali di Firenze*, 8:3–62.

Arthur Cayley. 1889. A theorem on trees. *Quarterly Journal of Mathematic's*, 23:376–378.

Yoeng-Jin Chu and Tseng-Hong Liu. 1965. On the shortest arborescence of a directed graph. *Scientia Sinica*, 14:1396–1400.

Ann Copestake and Ted Briscoe. 1995. Semi-productive polysemy and sense extension. *Journal of semantics*, 12(1):15–67.

Peter Crisp, Raymond Gibbs, Alice Deignan, Low, Graham, Gerard Steen, Cameron, Lynne, Elena Semino, Joe Grady, Alan Cienki, and Zoltan Kövecses. 2007. MIP: A method for identifying metaphorically used words in discourse. *Metaphor and Symbol*, 22(1):1–39.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. ImageNet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255.

Timothy Dozat and Christopher D. Manning. 2017. Deep biaffine attention for neural dependency parsing. In *International Conference on Learning Representations*.

Jack Edmonds. 1967. Optimum branchings. *Journal of Research of the national Bureau of Standards B*, 71(4):233–240.

Vyvyan Evans. 2005. The meaning of time: Polysemy, the lexicon and conceptual structure. *Journal of linguistics*, 41(1):33–75.

Joseph L Fleiss. 1971. Measuring nominal scale agreement among many raters. *Psychological bulletin*, 76(5):378.

Carlos Gómez-Rodríguez, Daniel Fernández-González, and Víctor Manuel Darriba Bilbao. 2015. Undirected dependency parsing. *Computational Intelligence*, 31(2):348–384.

Patrick Hanks. 2013. *Lexical analysis: Norms and exploitations*. MIT Press.

Patrick Hanks and James Pustejovsky. 2005. A pattern dictionary for natural language processing. *Revue Française de linguistique appliquée*, 10(2):63–082.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Comput.*, 9(8):1735–1780.

Douglas R. Hofstadter. 1995. *Fluid concepts and creative analogies: Computer models of the fundamental mechanisms of thought.* Basic Books.

Lawrence Hubert and Phipps Arabie. 1985. Comparing partitions. *Journal of classification*, 2(1):193–218.

Roman Jakobson. 1956. The metaphoric and metonymic poles. In Roman Jakobson and Morris Halle, editors, *Fundamentals of Language*. Mouton de Gruyter.

George Lakoff. 1987. *Women, fire, and dangerous things: What categories reveal about the mind.* University of Chicago Press.

George Lakoff, Jane Espenson, and Alan Schwartz. 1991. *Master Metaphor List (Second Draft)*. University of California.

George Lakoff and Mark Johnson. 1980. *Metaphors We Live By*. University of Chicago Press.

Chee Wee (Ben) Leong, Beata Beigman Klebanov, Chris Hamill, Egon Stemle, Rutuja Ubale, and Xianyang Chen. 2020. A report on the 2020 VUA and TOEFL metaphor detection shared task. In *Proceedings of the Second Workshop on Figurative Language Processing*, pages 18–29, Online. Association for Computational Linguistics.

Chee Wee (Ben) Leong, Beata Beigman Klebanov, and Ekaterina Shutova. 2018. A report on the 2018 VUA metaphor detection shared task. In *Proceedings of the Workshop on Figurative Language Processing*, pages 56–66, New Orleans, Louisiana. Association for Computational Linguistics.

Birte Lönneker. 2003. Is there a way to represent metaphors in WordNets? Insights from the Hamburg Metaphor Database. In *Proceedings of the ACL 2003 Workshop on the Lexicon and Figurative Language*, pages 18–27, Sapporo, Japan. Association for Computational Linguistics.

Ilya Loshchilov and Frank Hutter. 2017. Fixing weight decay regularization in Adam. *CoRR*, abs/1711.05101.

Rowan Hall Maudslay and Simone Teufel. 2022a. Homonymy information for English WordNet. In *Proceedings of Globalex Workshop on Linked Lexicography within the 13th Language Resources and Evaluation Conference*, pages 90–98, Marseille, France. European Language Resources Association.

Rowan Hall Maudslay and Simone Teufel. 2022b. Metaphorical polysemy detection: Conventional metaphor meets word sense disambiguation. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 65–77, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

John P. McCrae, Alexandre Rademaker, Francis Bond, Ewa Rudnicka, and Christiane Fellbaum. 2019. English WordNet 2019 – an open-source WordNet for English. In *Proceedings of the 10th Global Wordnet Conference*, pages 245–252, Wroclaw, Poland. Global Wordnet Association.

Ryan McDonald, Fernando Pereira, Kiril Ribarov, and Jan Hajič. 2005. Non-projective dependency parsing using spanning tree algorithms. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, pages 523–530, Vancouver, British Columbia, Canada. Association for Computational Linguistics.

Saif Mohammad, Ekaterina Shutova, and Peter Turney. 2016. Metaphor as a medium for emotion: An empirical study. In *Proceedings of the Fifth Joint Conference on Lexical and Computational Semantics*, pages 23–33, Berlin, Germany. Association for Computational Linguistics.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. PyTorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc.

Bolette Pedersen, Nathalie Carmen Hau Sørensen, Sanni Nimb, Ida Flørke, Sussi Olsen, and Thomas Troelsgård. 2022. Compiling a suitable level of sense granularity in a lexicon for AI purposes: The open source COR lexicon. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 51–60, Marseille, France. European Language Resources Association.

Tommaso Petrolito and Francis Bond. 2014. A survey of WordNet annotated corpora. In *Proceedings of the Seventh Global Wordnet Conference*, pages 236–245, Tartu, Estonia. University of Tartu Press.

Miriam Petruck. 2018. *MetaNet*. John Benjamins, Amsterdam.

James Pustejovsky. 1995. *The Generative Lexicon*. MIT Press.

James Pustejovsky, Patrick Hanks, and Anna Rumshisky. 2004. Automated induction of sense in context. In *COLING 2004: Proceedings of the 20th International Conference on Computational Linguistics*, pages 924–930, Geneva, Switzerland. COLING.

Eleanor H. Rosch. 1973. Natural categories. *Cognitive Psychology*, 4(3):328–350.

Bianca Scarlini, Tommaso Pasini, and Roberto Navigli. 2020. SensEmBERT: Context-enhanced sense embeddings for multilingual word sense disambiguation. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05):8758–8765.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(56):1929–1958.

Zhewei Sun, Richard Zemel, and Yang Xu. 2021. A computational framework for slang generation. *Transactions of the Association for Computational Linguistics*, 9:462–478.

Stephen Ullmann. 1962. *Semantics: An introduction to the science of meaning*. Blackwell, Oxford.

Ludwig Wittgenstein. 1953. *Philosophical investigations*. Macmillan.

Language Resource References

Christine Fellbaum, editor. 1998. *WordNet: An Electronic Lexical Database*. MIT Press.

John McCrae. 2020. *Open English Wordnet 2020*. GitHub.

George A. Miller. 1995. WordNet: A lexical database for English. *Commun. ACM*, 38(11):39–41.

Gerard J Steen, Aletta G Dorst, J Berenike Herrmann, Anna A Kaal, and Tina Krennmayr. 2010. Metaphor in usage. *Cognitive Linguistics*, 21(4):765–796.

A. Calculating the Number of Possible Annotations

Here we show how we compute the number of possible annotations of a word w that has n senses (shown in Table 1), excluding features, split senses, and virtual senses. We denote this quantity a_n. This quantity corresponds to the number of possible forests of rooted trees with labelled edges.

Assuming that only one sense can be prototypical (i.e. ignoring homonymy), we can take the number of possible annotations as:

$$a_n^1 = \underbrace{n^{n-2}}_{(1)} \times \underbrace{k^{n-1}}_{(2)} \times \underbrace{n}_{(3)} \tag{4}$$

where the terms correspond to:

(1) The number of possible undirected trees with n elements, given by Cayley's formula.

(2) The number of possible edge labellings for a tree with n elements (and hence $n-1$ edges), and k possible edge labels (in our case $k = 2$, for metaphor and metonymy).

(3) The number of possible roots of a tree of size n (each element could be the prototype).

We need to adapt this to accommodate homonymy (multiple prototypes). Let $[n]$ denote a set with n elements, containing all the natural numbers up to and including n, i.e. $[n] = \{1, 2, ..., n\}$. These numbers correspond to the indices of the senses of w. Furthermore, let B_s be a set containing all possible partitions of a set s. For example, for the set $[2] = \{1, 2\}$, $B_{[2]} = \{\{\{1\}, \{2\}\}, \{\{1, 2\}\}\}$. Each partition corresponds to a particular way of splitting the senses of a word into disjoint sets, i.e. homonyms. We use this to calculate a_n:

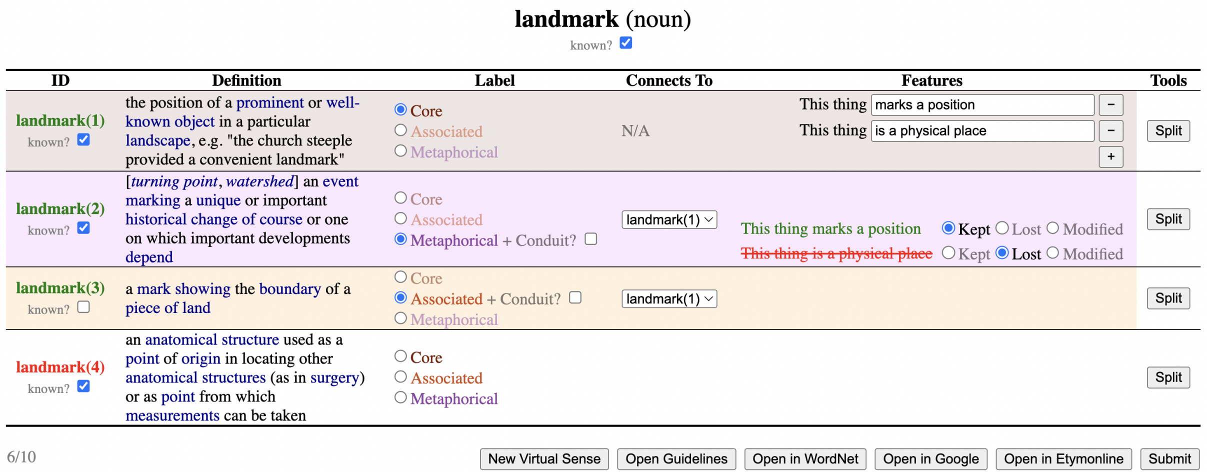
$$a_n = \sum_{\mathcal{P} \in B_{[n]}} \prod_{\mathcal{S} \in \mathcal{P}} a^1_{|\mathcal{S}|} \tag{5}$$

The inner term gives us the total number of possible annotations for a particular partitioning. The outer summation covers all possible partitionings.

B. The Annotation Interface

The annotation for a word is entered by completing a table, where each row of the table contains one of the word's senses. This interface is shown in Figure 3. At the bottom of the table, there are buttons which can be used to add virtual senses, to open the annotation guidelines, to search for the word in various other resources, and finally to submit the annotation. Annotation can only be submitted if it is complete, otherwise the system will not accept the submission. From left to right, the columns are:

ID This column contains the ID of each sense, as well as a check box which annotators can use to mark whether they know the sense (by default it is checked). Whole words can be labelled as unknown by deselecting the check box below the word at the top of the screen; annotators are required to annotate every word and sense, including those they did not previously know. A red sense ID indicates that a sense has incomplete annotation, whereas green indicates completion.



landmark (noun)

known? ☑

ID	Definition	Label	Connects To	Features	Tools
landmark(1) known? ☑	the position of a prominent or well-known object in a particular landscape, e.g. "the church steeple provided a convenient landmark"	● Core ○ Associated ○ Metaphorical	N/A	This thing [marks a position] − This thing [is a physical place] − +	Split
landmark(2) known? ☑	[*turning point*, *watershed*] an event marking a unique or important historical change of course or one on which important developments depend	○ Core ○ Associated ● Metaphorical + Conduit? □	landmark(1)	This thing marks a position ● Kept ○ Lost ○ Modified ~~This thing is a physical place~~ ○ Kept ● Lost ○ Modified	Split
landmark(3) known? □	a mark showing the boundary of a piece of land	○ Core ● Associated + Conduit? □ ○ Metaphorical	landmark(1)		Split
landmark(4) known? ☑	an anatomical structure used as a point of origin in locating other anatomical structures (as in surgery) or as point from which measurements can be taken	○ Core ○ Associated ○ Metaphorical			Split

Figure 3: Screen capture of the annotation tool, with incomplete annotation for *landmark*

Definition This column contains the WordNet definition for each sense. Sometimes these definitions are vague. If this is the case, annotators can hover their mouse over blue words in the definition. This causes a popup bubble to appear which reveals the definition of the word in question.

Label Annotators use radio buttons to decide which label to apply to each sense. If a sense is labelled as a metaphor or a metonym, the option appears to mark the sense as a conduit. Conduits can be extended by any other sense (see §4.1).

Connects To If the annotator labels a sense as a metaphor or metonym, then a drop-down list will appear in this column. Annotators use this list to choose which sense is being extended. Options that are invalid (such as the same sense that is being labelled) are greyed out.

Features If a sense is extended by a metaphor (e.g. *landmark*$_1$ in Figure 3), then a button appears in this column to add features. Clicking the button causes a feature to appear. Each feature has an empty text box that can be filled in. Meanwhile, if a sense is a metaphor (*landmark*$_2$ in Figure 3), this column will contain a copy of each feature of the sense it extends. These copies are each accompanied by radio buttons to indicate whether the feature is kept, lost, or modified. If a feature is marked as modified then a text box appears containing the same text as the feature. This text box is used to record the modification.

Tools This column contains buttons to perform additional operations. For most senses the button gives annotators the option to split the sense. If clicked, that row of the column will split in two, the definitions will become editable, and the button will change to give annotators the option to re-merge the senses. If a virtual sense is added, a new row is added to the table; in that case, the button gives annotators the option to delete the virtual sense.

C. Baseline Training Details

Training Data Polysemy parsing does not treat virtual senses or split senses. To train models, we process ChainNet to remove these features. More specifically, we remove each split sense by re-merging it and giving the combined sense whichever annotation belonged to its non-metaphorical component. We remove each virtual sense by connecting all of its children to its parent. We then randomly divide the resultant dataset into train, development, and test partitions (an $80{:}10{:}10$ split of the 6500 words in ChainNet).

Training Procedure We implement models in PyTorch (Paszke et al., 2019), and optimise the cross entropy loss in each case using AdamW (Loshchilov and Hutter, 2017). Every epoch, we compute the loss on the development set. When the loss fails to decline for 8 epochs we recover the best model. The first time this happens we divide the learning rate by 10 and continue training; the second time we terminate model training.

Hyperparameters We trained in batches of 32 words, with a learning rate of $\alpha = 5{\times}10^{-5}$, and β_1 and β_2 for AdamW both set to 0.9. For the biaffine parser the dropout (Srivastava et al., 2014) was 0.33; k' was 100 for the label predictor and 2048 for the edge predictor (to match the dimensionality of the embeddings, $k = 2048$ from Scarlini et al., 2020). These parameters were chosen following Dozat and Manning (2017).