Applied Computing and Intelligence

https://www.aimspress.com/journal/aci

Applied Computing and Intelligence
2(1): 83–98
DOI: 10.3934/aci.2022005
Received: 17 March 2022
Revised: 29 March 2022
Accepted: 30 March 2022
Published: 31 March 2022

Review

Definition modeling: literature review and dataset analysis

Noah Gardner[1], **Hafiz Khan**[2] **and Chih-Cheng Hung**[1,*]

[1] Laboratory of Machine Vision and Security Research, College of Computing and Software Engineering, Kennesaw State University, Marietta GA, USA

[2] Laboratory of Ubiquitous Data Mining, College of Computing and Software Engineering, Kennesaw State University, Marietta GA, USA

* **Correspondence:** chung1@kennesaw.edu

Academic Editor: Pasi Fränti

Abstract: Definition modeling, the task of generating a definition for a given term, is a relatively new area of research applied in evaluating word embeddings. Automatic generation of dictionary quality definitions has many applications in natural language processing, such as sentiment analysis, machine translation, and word sense disambiguation. Additionally, definition modeling is also helpful for evaluating the quality of word embeddings. As more research is done in this field, the need for a summary of different applications, approaches, and obstacles grows apparent. This review provides an overview of the current research in definition modeling and a list of future directions and trends.

Keywords: definition modeling; definition generation; natural language processing; word embeddings

1. Introduction

Definitions are explicit representations of words or phrases that are valuable for exposing the aspects of a given term. In general, definitions are unambiguous and succinct: they should be easy to read and understand. Recent research has allowed the creation of neural language models that can generate useful definitions from embeddings [3, 8, 24]. Word embeddings are vector representations of words that have been employed in a variety of *natural language processing* (NLP) tasks. They are useful for capturing lexical syntax and semantic similarity. Mikolov et al. [19] have shown that basic mathematical operations applied to word embeddings can have meaningful language understanding. However, as continuous representations, the interpretability of word embeddings is limited.

The problem of *definition modeling* was proposed by Noraset et al. [21] to evaluate word embeddings. The task of definition modeling is to generate a definition for a given term. The goal of a model trained on this task is to train on word embedding and definition pairs to learn to generate

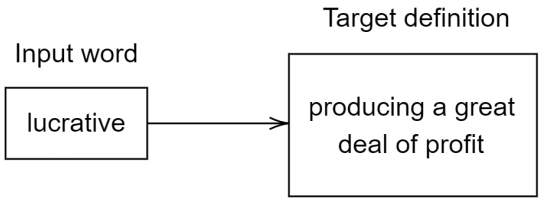

Figure 1. Monoseme example word and definition. A definition model could generate the definition *producing a great deal of profit* for the input word *lucrative*.

a definition for a given word or phrase. An example of a *monoseme* (word with a single definition) is given in Figure 1. Given the input word *lucrative*, a model trained on the task of definition modeling would produce the output definition *producing a great deal of profit*.

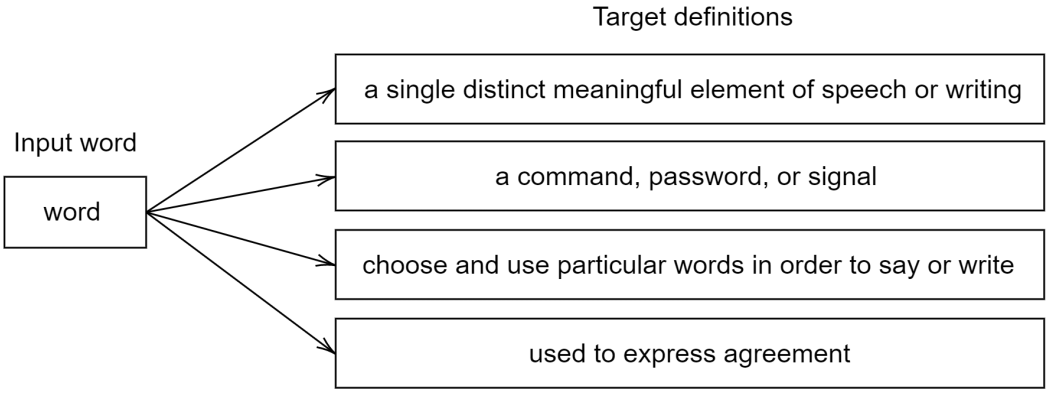

Figure 2. Polyseme example word and definition. A definition model could generate any of the target definitions.

In addition to being a relatively new language modeling task, definition modeling has attracted attention from the literature in a number of areas. First, it was shown that the definition model has poor performance when generating definitions for polysemes: words with multiple definitions [6]. An example polysemous word is shown in Figure 2. Given the input *word*, the goal of a definition model would be to generate one of the target definitions, most ideally the closest definition to the word sense of the input. However, it is difficult to know the word sense given only the input word.

The problem of polysemous words was not addressed in the original work, as only one definition mapped to each word. Once researchers attempted to address this problem, they found that the definition model could not learn the semantics of the polyseme with only the word as an input. Therefore, it was necessary to augment the definition model with additional information, namely, an example sentence that sets the word to be defined (*definiendum*) inside to provide context. This method has been shown to alleviate the problem of generating definitions for polysemes and also improve the performance of the definition model on several measures [2, 6, 17].

Definition modeling, especially as a sequence-to-sequence task, is similar to other NLP tasks, such as *word-sense disambiguation*, *word-in-context*, and *definition extraction* [2, 9]. When using context to generate a definition from an input word, the input's word sense must be extracted to select the

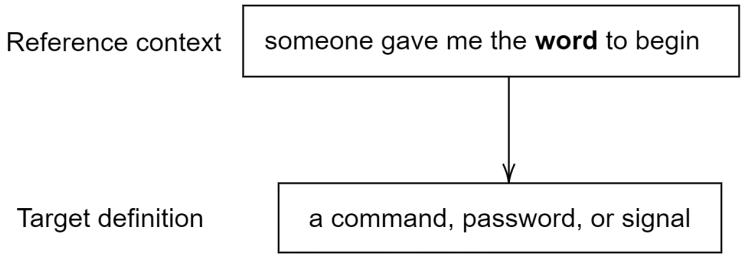

Figure 3. Context-dependent definition task. The word to be defined is marked in bold. In this case, although *word* is a polyseme, there is only one correct target definition due to the contextual information provided.

correct definition. The goal of word-sense disambiguation is similar in that the goal of word-sense disambiguation is to identify the sense of a word used in a sentence. Similarly, definition extraction seeks to extract definitions of terms from an existing corpus [9]. Figure 3 shows an example of definition modeling in a context-dependent situation. In the example, a reference context is given. Inside the reference context, a target word *word* is marked as the word to be defined. The goal of a definition model given this context and marked word would be to generate the target definition *a command, password, or signal*.

Our paper is organized into three sections. Section 2 reviews definition modeling methods as well as word embeddings. Section 3 shares benchmark datasets and statistics that can be used when formulating and evaluating a definition modeling method. Section 4 explores challenges encountered in this research field and gives suggestions for future work.

2. Methodologies explored

We explore recent literature related to definition modeling and present our findings related to explored methodology in this section. Definition generation is a critical task where multiple definitions can be generated for a single target word. Therefore, researchers focus on improving the definition generation task by applying various techniques. Two key technical aspects are observed in the literature: definition generation and word embedding. Definition generation is considered a language modeling task, where we predict the joint probability of a sequence of words, and based on maximum likelihood, the highest probability sequence is returned as a definition of a given target word. Since the output definition mostly depends on the context of the target word, vector representation of such target words is essential to capture context scenarios. Below we discuss both of these aspects, language modeling, and word embedding techniques and related literature.

2.1. Language models and definition generation

A definition model is a language model that is trained on a set of definitions [21]. The goal of a definition model is to learn to generate a definition for a given term. The probability of generating the t-th word in a definition depends on both the previous words in the definition and the word to be defined (Eq 2.1).

$$p(\mathbf{d}|w) = \prod_{t=1}^{T} p(d_t|d_1, ..., d_{t-1}, w) \tag{2.1}$$

where $\mathbf{d}$ is the generated definition as a vector of words ($\mathbf{d} = [d_1, ..., d_T]$) and w is the word or phrase to be defined.

Noraset et al. [21] condition a *recurrent neural network* (RNN) to generate a definition from an input seed word. They modify the model by updating the output of the recurrent unit with an update function inspired by *gated recurrent unit* (GRU) update gate [21]. They apply pretrained word embeddings generated from *Word2Vec* [18]. In later work, it was shown that the definition model does not generate definitions for words with ambiguous word sense, especially polysemantic words [6]. The following context-aware definition model was proposed by Gadetsky et al. [6] to tackle this challenge. To generate a definition, authors use an attention-based skip-gram model to extract dimensions from the embedding which contain the most relevant information [6]. They extend Eq 2.1 by adding a context term which is a contextual phrase or example sentence to be used in the generation of the definition.

$$p(\mathbf{d}|w, \mathbf{c}) = \prod_{t=1}^{T} p(d_t|d_1, ..., d_{t-1}, w, \mathbf{c}) \tag{2.2}$$

where $\mathbf{c}$ is the context phrase ($\mathbf{c} = [c_1, ..., c_T]$).

Researchers apply *sequence-to-sequence* algorithms and represented definitions vectors by formulating language modeling to capture sequence features and context [2, 9, 11, 23, 24]. Among these algorithms, RNN and *long-short-term-memory network* (LSTM) are important as they can capture semantic information across words in a sentence as sequential data. Not all words are equally important in a definition as they have different contributions in the definition generations. Transformer-based techniques help focus on the contribution of particular words in the definition. Therefore, few researchers also focus on transformer networks such as *bidirectional encoder representations from transformers* (BERT) and denoising decoder (BART) [5, 14].

The definition usually contains summarized information about the given target word. Huang et al. [9] focus on generating definition by using extracted self- and correlative definition information of a given term from the web. The authors in [9] extracted sentences containing the target term and then ranked sentences using deployed BERT-based model and extracted self-definitional information (SDI) from Wikipedia. Then, they design a conditional sequence-to-sequence model, BART, and fine-tune parameter with extracted information and general definition for a given term.

Definition modeling works similarly to language models to generate definition sentences and corresponding probabilities. Gadetsky et al. [6] proposed a *conditional RNN* based language model for developing the definition of a given word. First, they created AdaGram based RNN model and conditioned it on adaptive skip-gram vector representation. Their second model focused on an attention-based skip-gram to generate a definition for a corresponding context.

Li et al. [15] proposed *explicit semantic decomposition* (ESD) to decompose the meaning of the word into semantic components and model them with the discrete latent variable for definition generation. This model comprises an *encoder*, *decoder*, and *semantic component predictor*. The encoder consists of two components: word encoder and *bidirectional LSTM* (Bi-LSTM) context encoder. Word encoder creates low-dimensional vectors of the word, whereas the Bi-LSTM context

encoder incorporates context information. Semantic component predictor model approximate posterior using Bi-LSTM model. Finally, LSTM based definition decoder generates a definition from the encoded data.

Bevilacqua et al. [2] propose a span-based encoding model that is used to map occurrences of target words or phrases in a given context and generate a gloss. Using the probability of a gloss for a given context-word pair, their method can perform classification by selecting the gloss with the highest probability. The textual gloss is then applied to define the context and word.

Ishiwatari et al. [10] solve the problem of unknown phrase definition by incorporating local and global context information while defining a word. Local context refers to the sequence of neighboring words of the target word. In contrast, the global context refers to the entire document or even searching the web text to find other occurrences of the expression to understand the meaning. The authors in [10] proposed LSTM based encoder-decoder model where a gated unit deployed reduces the ambiguity of local and global context.

Mickus et al. [17] argue that due to the *distribution hypothesis* (words with similar distribution have similar meaning), the problem of definition modeling should be reformulated as a sequence-to-sequence task, where the input sequence is a sentence with the word to be defined highlighted [17]. The input sequence provides the context necessary to generate the output definition. Zhu et al. [28] study the multi-sense definition modeling task using the Gumble-softmax approach. This approach decomposes word senses from the pre-trained word embeddings and applies LSTM sequence-to-sequence modeling to generate definition sentences.

Reid et al. [23] introduced a variational generative model to produce a definition that directly combines lexical and distributional semantics using the continuous latent variable. Initially, the BERT model is fine-tuned with phrase-context pairs, and in the context, sentence lexeme form is used to reduce the differences in the word or phrase. Once the BERT model encodes the definition, the proposed approach applies a neural definition inference module to compute approximate posterior from the variational distribution of the definition. During definition generation, that is, sequence of word generation task, this model deploys LSTM enabled variational contextual definition modeler to generate a sequence of words as the definition.

Chang et al. [4] explore contextualized embedding for definition modeling - to get contextualized word embedding the authors used the pretrained ELMo and BERT model. The authors in [4] reformulate the problem of definition modeling from text generation to text classification. Instead of mapping the classifier with discrete labels, all ground truth definitions are encoded in the embedding space via learning a mapping function. Then, they generate an embedding for a given word-in-context and apply k-nearest neighbor to predict multiple definitions for a given target word from a corpus of existing definition embeddings. Their results show state-of-the-art performance on the task of definition modeling.

Non-English languages: Most definition modeling methods focus on generating definitions in English for English words. Definition generation was also explored in the non-English language. Since the definition depends on the lexical properties, language syntax, and phrase construction, different languages influence the proposed methodology to capture the definition of a specific word. For example, in parataxis languages (e.g., Chinese), the meaning of a word is based on formation components (morphemes) combined by the formation rule (morphemes are combined to form words).

Zheng et al. [27] utilizes this word meaning formation process in consideration to build a definition

generation model where words decompose into formation features and then use gating techniques to generate definition. In this work, the authors in [27] develop morpheme features using the Bi-LSTM model and concatenate character-level embedding and pre-trained word embedding together. Finally, gated attention-based morpheme features with attention-based context vector to form a feature vector. The definition generator employs a gated LSTM model that generates the definition using the feature vector.

Ni et al. [20] automatically generates explanations for non-standard English expressions using sequence to sequence models. The authors use two encoder approaches: a word-level LSTM-encoder encodes context information, and a character-level encoder encodes target non-standard terms [20]. Kong et al. [13] fine-tune mBERT and XLM cross-lingual model and provide target word and examples sentence as context to produce definition as output. This model can generate definitions in English from various languages (e.g., Chinese to English).

Kabiri et al. [11] proposed context-agnostic multi-sense definition generation model. The proposed RNN based model generates multiple definitions based on a given target word type (polysemous word) and incorporates the char-CNN model to capture affixes knowledge. They associate sense vectors with definitions and create a definition-to-sense and sense-to-definition model. These definition models represented definition by taking the average of the word embeddings of all the words. Their multi-sense model demonstrates the ability to generate multi-sense embeddings across nine languages from various language families.

2.2. Word embedding

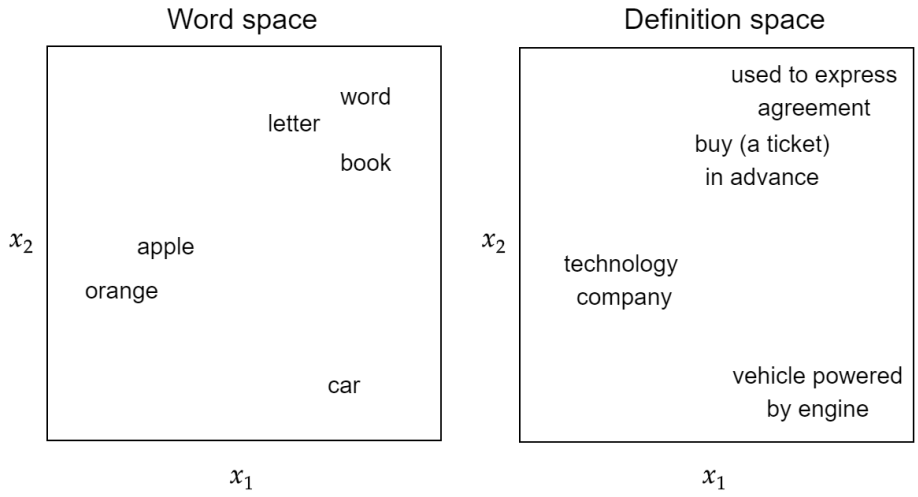

Figure 4. An example of two vector spaces: word space and definition space. As vector representations, semantically similar words and definitions are closer together. Although word embedding vectors can be quite large in practice, we represent them with two attributes for simplicity.

We can transform text or words into vector representations to analyze words effectively. Figure 4 represents word space (2-D) by attaching several numerical attributes to the words (x_1 and x_2). Word

embeddings are fixed-length vectors representing words in a vector space such that similar words meanings have similar vector space representations. In Word2Vec, a popular word embedding model, surrounding words are predicted from a given target word [18]. For an example, we will use the sentence *the height of Mount Everest is 29029 feet*. Given a target word *Mount*, we apply a context window of ±3. The model will attempt to predict the 3 words preceding the target word (*the height of*). The model will also attempt to predict the 3 words succeeding the target word (*Everest is 29029*). In the prediction process, the model simultaneously learns the vector representation of words and maximizes the prediction probability of the context window words.

The vector space representation is useful to measure the distance between words and do vector space calculations [19]. In definition modeling, the definition is also represented in a vector space so that the candidate definition of a target word can easily be found from the vector space. An example is shown in Figure 4, where each definition is represented using two attributes: x_1 and x_2. In the definition modeling problem, word to vector representation is key in modeling definitions for a given term.

Bosc et al. [3] exploited dictionary recursivity into consideration and proposed an autoencoder-based word embedding algorithm, and generated a single embedding per word—the proposed auto-encoder model comprises of an LSTM encoder and decoder. The authors in [3] introduced three embeddings: definition embeddings produced by the proposed definition encoder, input embeddings for the encoder, and output embeddings. While modeling these embedding models, A consistency penalty is applied as soft weight in their cost function to enforce input embedding and definition embedding closer [3].

Washio et al. [24], the authors consider lexical-semantic relations between the defined word and defining words using unsupervised methods to propose definition modeling. To learn word embeddings, the authors proposed LSTM-based encoder and decoder with an additional cost function to learn word-pair embeddings in the decoder and capture lexical-semantic relations. Dictionary embeddings often follow a genus and differentia structure for a dictionary definition. Noraset et al. [21] capture hypernyms embedding following proper genus database WebIsA containing hypernym relations. In addition, the authors in [21] incorporate char-CNN to capture affixes to model gated-RNN based definition modeling.

Word embeddings are learned from large corpora. Therefore, it may consist of biases such as gender, race, and religion. On the other hand, word dictionaries contain unbiased, concise definitions. To overcome these biases by utilizing pre-trained word-embedding, Kaneko et al. [12] apply learned embedding from existing input word embeddings using encoder-decoder architecture by defining a decoder cost function that considers dictionary agreement as a constraint and decodes the debiased embedding.

Zhang et al. [25] propose a novel framework by formulating definition modeling and word-embedding as multitask learning problems. The authors in [25] presented two types of multitasking models to combine usage and definition modeling. First, the authors in [25] used the GRU-based context encoder model as a semantic generative network to generate word embedding. This approach encodes context sequences into continuous vectors and generates a fixed-size sentence embedding. After that, self-attention is applied to consider the target word sense. Then, this model learns context-sensitive word embedding by fine-tuning ELMo models. Finally, the authors in [25] formulated multitask sequence-to-sequence modeling for usage modeling to generate definition and example sentences.

2.3. Evaluation criteria

Table 1. Evaluation criteria used in definition modeling.

Criteria	Methods
BLEU	[2, 6, 9–11, 15, 21, 23, 24]
Perplexity	[2, 6, 17, 21, 24]
ROUGE-L	[2, 4, 9]
METEOR	[2, 9, 15]
BERTScore	[2, 9, 23]
Human	[10, 15, 23]
Precision	[4]
Cosine similarity	[4]

A variety of evaluation criteria are used to evaluate generated definitions. Table 1 lists the evaluation criteria used in the definition modeling task. The evaluation takes the reference and candidate definitions as input and outputs a score on how well the candidate matches the reference. The reference definition is the correct definition of the source word or phrase, typically provided by a dictionary. The candidate definition is the machine-generated definition. We provide brief descriptions of the evaluation criteria used.

BLEU: *Bilingual evaluation understudy* (BLEU) is a standard algorithm used to evaluate machine translations [22]. BLEU score is calculated as n-gram precision, or the ratio of correct n-grams to the total number of output n-grams. A drawback of the BLEU score is that it matches correct n-grams and thus may not give a good score to an acceptable generated definition.

Perplexity: *Perplexity* is related to entropy, which is a measurement of the uncertainty of a probability distribution and is normalized by sentence length. The perplexity is a measure of the difficulty of generating a sentence. The lower the perplexity, the more natural the sentence is for the model.

ROUGE-L: *Recall-oriented understudy for gisting evaluation* (ROUGE) measures the matching n-grams between the reference and candidate definitions [16]. ROUGE-L is a modified version of ROUGE that uses the *longest common subsequence* to measure the similarity between the two definitions. An advantage of ROUGE-L is that it automatically determines the longest in-sequence common n-grams.

METEOR: *Metric for evaluation of translation with explicit ordering* (METEOR) is a metric that is based on unigram matching between the reference and candidate translations [1]. It computes a score based on the harmonic mean of precision and recall.

BERTScore: *Bidirectional encoder representations from transformers* (BERTScore) is a metric that computes a similarity score of the candidate and reference definitions based on the pre-trained contextual embeddings from BERT [26]. In addition, BERTScore computes precision, recall, and F1 measure.

Cosine similarity: *Cosine similarity* is a measure of the similarity between two vectors. It is simply calculated as the dot product of the two vectors divided by the product of their magnitudes.

Precision: *Precision* is a measure of the proportion of correctly identified words in a sentence.

In principle, any other string similarity measure could be applied for this task [7]. Human-based evaluation scores would be ideal due to expert linguistic knowledge. However, in practice, collecting

expert evaluation is costly. As BERTScore takes advantage of semantic information, it correlates better with human judgments and may be most useful for evaluating generated definitions [26].

3. Datasets and analysis

Table 2. Datasets used in definition modeling.

Dataset	Methods
Oxford	[2, 4, 6, 10, 15, 17, 23, 24]
WordNet	[2, 10, 11, 15, 17, 21, 24]
Urban Dictionary	[10, 20, 23]
Wikipedia	[9, 23]
Wiktionary	[2, 11]
OmegaWiki	[11]
Hei++	[2]

Various benchmark datasets have been proposed to train and evaluate definition models. Table 2 lists datasets applied in different definition modeling methods. In this section, we provide brief descriptions of each dataset and provide an analysis of various characteristics of the datasets.

Oxford Dictionary: *The Oxford Dictionary of English*[*] is a free dictionary of English words and phrases. Collected by Gadetsky et al. [6], this dataset features contextual information for each word along with the definition. This dataset is useful for evaluating the ability of a model to generate definitions for polysemous words.

GCIDE/WordNet: *The GNU Collaborative International Dictionary of English*[†] (GCIDE) is a free dictionary supplemented with some definitions from WordNet[‡]. GCIDE is a useful corpus for dictionary definitions for general words. This dataset was modified by Noraset et al. [21] for their original definition model. Kabiri et al. [11] also provide a modified dataset for their method.

Urban Dictionary: *The Urban Dictionary*[§] is a free dictionary of slang words and phrases where definitions are crowd-sourced by users. Proposed by Ni et al. [20], the Urban Dictionary dataset is useful for idioms and rarely-used phrases which are not contained in other dictionary datasets due to only containing slang definitions.

Wikipedia: *The English Wikipedia*[¶] is a free online encyclopedia. Collected by Ishiwatari et al. [10], it combines the useful tasks of WordNet, Oxford Dictionary, and Urban Dictionary, since it contains descriptions of many concepts along with context to be used in context-aware models.

Wiktionary: *Wiktionary*[‖] is a free online dictionary from the same parent organization as Wikipedia (Wikimedia Foundation). It is useful as it provides a definitions for a large number of languages which can allow for multi-lingual definition modeling. We share statistics for the English version of Wiktionary, since most definition modeling methods focus on English.

[*] https://languages.oup.com/
[†] https://gcide.gnu.org.ua/
[‡] https://wordnet.princeton.edu/
[§] https://www.urbandictionary.com/
[¶] https://en.wikipedia.org/
[‖] https://en.wiktionary.org/

OmegaWiki: Similar to Wiktionary, *OmegaWiki* is a multi-lingual dictionary. The goal of OmegaWiki is to create a lexical resource with all definitions of all words in every language. Kabiri et al. [11] use this resource due to the availability of a variety of languages.
Hei++: *Hei++*[**] is a unique evaluation dataset proposed by Bevilacqua et al. [2]. Rather than contain singular words or phrases to define as the other dictionary-based resources, this dataset is comprised of adjective-noun phrases. An example phrase, *starry sky*, can be defined as 'The sky as it appears at night, especially when lit by stars.' This is a hand-made dataset created with an expert lexicographer's assistance. As a result, this dataset is small and should be used in model evaluation rather than training. Our dataset analysis shows no overlap of this dataset with the other benchmark datasets, implying this dataset can also be used to evaluate the ability of a model to generalize on never-before-seen data.

3.1. Definition statistics

In our analysis of the datasets above, to distinguish the benchmark datasets provided by the correlating authors, we use the notations listed in Table 3.

Table 3. Dataset notations.

Dataset	Year	Reference
WordNet-A	2016	[21]
Urban	2017	[20]
Oxford	2018	[6]
WordNet-B	2019	[10]
Wikipedia	2019	[10]
Wiktionary	2020	[11]
WordNet-C	2020	[11]
Omega	2020	[11]
Hei++	2020	[2]

First, in Table 4, we provide some analysis of the definition statistics of the datasets. We evaluate all splits (train, test, and validate) for each dataset by combining all the words and corresponding definitions. The table shows the number of unique words and a total number of definitions for each dataset. Of note, some datasets provide the same definition for the same word, meant to be used in a context-aware model. In this analysis, we ignore the context phrases and treat these duplicate definitions as independent. We also show the mean length of the definitions, the standard deviation of the lengths, and the definitions per word.

Next, in Table 5, we show the number of polysemous words in each dataset. As with the definition statistics, we treat exact duplicate definitions independently because they have different contexts. The number of polysemes is calculated as the number of words or phrases with more than one definition in the dataset. Finally, we show the ratio of the number of polysemous words to the total number of words in the dataset as a percentage. It is important to evaluate the polysemous data due to the difficulty of predicting definitions for polysemous words.

Our following analysis is on the overlap present across the benchmark datasets. We show the number of words in each dataset which are present in all other datasets as a percentage. This allows

[**]http://generationary.org/

Table 4. Definition statistics.

Dataset	Words	Definitions	Definitions per word	Mean length	Standard deviation length
Wikipedia	168,753	988,690	5.86	5.99	4.53
Urban	240,334	507,504	2.11	12.11	7.71
WordNet-A	22,554	162,925	7.22	6.60	5.73
Oxford	36,767	122,319	3.33	11.07	7.01
Wiktionary	17,000	29,426	1.73	7.65	6.92
WordNet-C	20,000	28,814	1.44	10.96	7.28
Omega	17,000	22,735	1.34	14.61	9.83
WordNet-B	9,937	17,410	1.75	6.64	3.78
Hei++	713	713	1.00	9.44	2.80

Table 5. Polyseme statistics.

Dataset	Words	Polysemes	Polysemes (%)
WordNet-A	22,554	22,171	98
Oxford	36,767	20,563	56
Wikipedia	168,753	77,278	46
WordNet-B	9,937	4,221	42
Urban	240,334	74,620	31
Wiktionary	17,000	4,634	27
Omega	17,000	3,412	20
WordNet-C	20,000	3,649	18
Hei++	713	0	0

us to identify the most similar and most unique datasets. The overlap of each dataset is calculated as the words that are present in each other datasets. The overlap of each dataset is shown in Table 6. The values in the table represent the percent of the words in the row dataset that are present in the column dataset. For example, 25% of the words in the OmegaWiki dataset are present in the Oxford dataset. We also show the uniqueness of each dataset, calculated as the percentage of words in a dataset that are not present in any other dataset. The plot of dataset uniqueness is shown in Figure 5. The uniqueness of the Hei++ dataset is due to two factors: its relatively small size and focuses on adjective-noun phrases.

In most of the datasets, some definitions consist only of a single word. Single-word definitions may cause evaluation criteria such as BLEU to be challenging to improve. We show the percentage of definitions in each dataset which consists of only a single word. We also show the number of single-word definitions in each dataset which are considered to be a synonym of the word or phrase being defined. We used WordNet synsets to identify synonymous words. The single word definition analysis is shown in Figure 6.

Across every benchmark dataset, there does not exist a word that is present in each dataset. However, there is a word that exists in 8 out of 9 datasets: the word *movement*. We show selected definitions for this word in Table 7. Several different word senses can be seen across the dataset, such as movement as something moving, a specific album, bowel movement, and even the illusion of something moving.

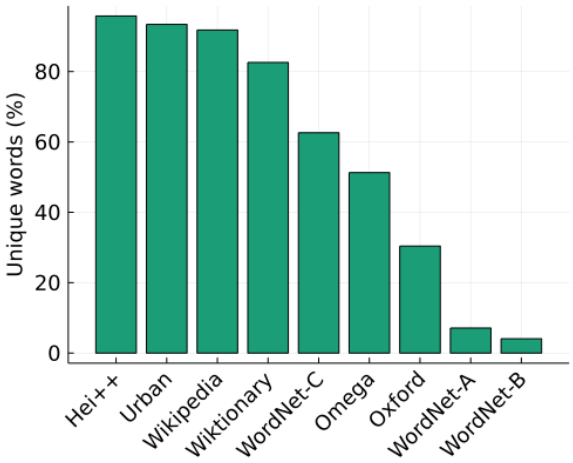

Figure 5. Plot of dataset uniqueness.

Table 6. Individual dataset overlap.

Dataset	Hei++	Omega	Oxford	Urban	Wiki	Wiktionary	Word Net-A	Word Net-B	Word Net-C
Hei++	-	0%	0%	2%	2%	0%	0%	0%	1%
Omega	0%	-	25%	13%	12%	2%	19%	8%	6%
Oxford	0%	5%	-	7%	6%	1%	14%	5%	4%
Urban	0%	1%	2%	-	1%	0%	1%	1%	0%
Wikipedia	0%	0%	1%	1%	-	0%	0%	0%	0%
Wiktionary	0%	1%	5%	4%	2%	-	3%	1%	1%
WordNet-A	0%	3%	10%	4%	3%	1%	-	6%	2%
WordNet-B	0%	11%	35%	15%	11%	2%	53%	-	8%
WordNet-C	0%	5%	16%	7%	7%	1%	11%	5%	-

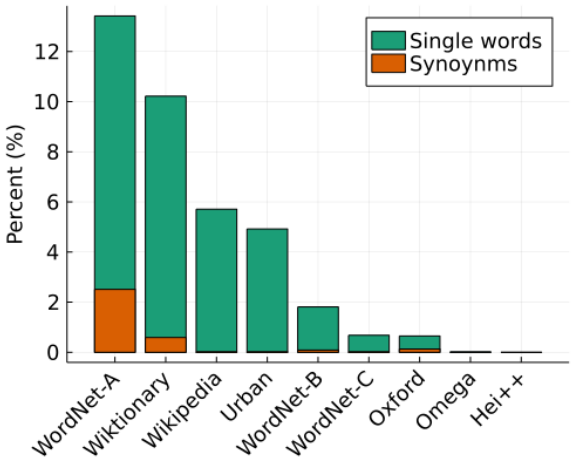

Figure 6. Plot of single word definitions.

Table 7. Definitions for the term *movement*.

Dataset	Definition
WordNet-A	a natural event that involves a change in the position or location of something
Oxford	a group of people with a common ideology who try together to achieve certain general goals
WordNet-B	a major self-contained part of a symphony or sonata
Wikipedia	album by new order
Urban	[pot credit] slang, to hit on a woman
WordNet-C	an optical illusion of motion produced by viewing a rapid succession of still pictures of a moving object
Wiktionary	the deviation of a pitch from ballistic flight
Omega	what a dogs body releases from time to time as a little pile of waste remaining from digestion , after it has been collected in the colon.

4. Challenges and future directions

Definition modeling faces several challenges, allowing new opportunities for future research to be developed.

Polysemes: The basic definition model cannot be used to generate definitions for *polysemes*, words with multiple definitions. As a significant challenge for the original definition model, many researchers have proposed methods to tackle this problem. However, many of the proposed approaches require the context of the *definiendum* to be provided to the model. Methods that provide appropriate definitions for polysemes without context may be valuable in tasks with limited language resources.

Technical terms: It is challenging to generate definitions for technical terms which require expert knowledge of the field [9]. It may be necessary to provide specific context to generate definitions for technical terms appropriately. However, obtaining the context requires scraping and parsing web resources outside of the standard datasets available. Therefore, it may be necessary to generate definitions for technical terms to augment dictionary datasets properly.

Word combinations: Complex word combinations, including proverbs and sayings, are rarely covered by sense inventories [2]. In word combinations, multiple words are used in series to create a new phrase that may be interpreted as a single word for the case of definition modeling. Since the resulting definition of word combinations may or may not depend on the words used, context may be necessary to parse these word combinations and generate useful definitions. Still, more research is needed to determine if this is the case. Additionally, word combinations may be absent from the standard dictionary-based datasets.

Non-English words: As many of the datasets developed for defintion modeling thus far take information from English dictionaries, most methods also are only applied to English words. In addition, as there exist several lexical resources in other languages, it should be possible to generate definitions for non-English words. To evaluate the quality of word embeddings for non-English words within definition modeling, it is necessary to develop a method to generate definitions for non-English words. There is some work in Chinese definition modeling [27], and in French definition modeling [23]. However, more research is needed to determine the best method for generating

definitions for non-English words, especially for a model that can generalize across multiple languages. **Evaluation criteria:** Definition models have been evaluated on a number of metrics, including precision, perplexity, BLEU, and ROUGE. However, as definition modeling aims to improve the interpretability of word embeddings, it is important to select the evaluation criteria correctly. Many definitions consist of a single word, which can interfere with evaluation metrics such as BLEU and ROUGE scores [17]. Human evaluation of generated definitions can be useful but difficult for researchers to obtain.

5. Conclusions

The problem of definition modeling is challenging to solve. Specifically, a major challenge is generating definitions for polysemous words. Since the formulation of the task, researchers have been working on various approaches to generate definitions for creating NLP corpora and the evaluation of word embeddings. In this paper, we provide an overview of definition modeling methods and word embeddings applied to the definition modeling task. We share some benchmark datasets and analyses. Our analysis highlights unique points available in each benchmark dataset, including definition statistics, polyseme statistics, and the overlap across all datasets. Finally, we share the collected datasets in a public GitHub repository. [††]

Acknowledgments

We would like to thank the constructive feedback provided by the reviewers.

Conflict of interest

All authors declare no conflicts of interest in this paper.

References

1. S. Banerjee, A. Lavie, METEOR: An automatic metric for MT evaluation with improved correlation with human judgments, in *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, (2005), 65–72.
2. M. Bevilacqua, M. Maru, R. Navigli, Generationary or "How We Went beyond Word Sense Inventories and Learned to Gloss", in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, (2020), 7207–7221. https://doi.org/10.18653/v1/2020.emnlp-main.585
3. T. Bosc, P. Vincent, Auto-Encoding Dictionary Definitions into Consistent Word Embeddings, in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, (2018), 1522–1532. https://doi.org/10.18653/v1/D18-1181
4. T.-Y. Chang, Y.-N. Chen, What Does This Word Mean? Explaining Contextualized Embeddings with Natural Language Definition, in *Proceedings of the 2019 Conference on Empirical Methods*

[††]https://github.com/DefinitionModeling/DefModelDatasets.jl

in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), (2019), 6064–6070. https://doi.org/10.18653/v1/D19-1627

5. J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, (2019), 4171–4186.
6. A. Gadetsky, I. Yakubovskiy, D. Vetrov, Conditional Generators of Words Definitions, in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, (2018), 266–271. https://doi.org/10.18653/v1/P18-2043
7. N. Gali, R. Mariescu-Istodor, D. Hostettler, P. Fränti, Framework for syntactic string similarity measures, *Expert Syst. Appl.*, **129** (2019), 169–185. https://doi.org/10.1016/j.eswa.2019.03.048
8. F. Hill, K. Cho, A. Korhonen, Y. Bengio, Learning to Understand Phrases by Embedding the Dictionary, *Transactions of the Association for Computational Linguistics*, **4** (2016), 17–30. https://doi.org/10.1162/tacl_a_00080
9. J. Huang, H. Shao, K. C.-C. Chang, CDM: Combining Extraction and Generation for Definition Modeling, *arXiv:2111.07267 [cs]*.
10. S. Ishiwatari, H. Hayashi, N. Yoshinaga, G. Neubig, S. Sato, M. Toyoda, M. Kitsuregawa, Learning to Describe Unknown Phrases with Local and Global Contexts, in *Proceedings of the 2019 Conference of the North*, (2019), 3467–3476. https://doi.org/10.18653/v1/N19-1350
11. A. Kabiri, P. Cook, Evaluating a Multi-sense Definition Generation Model for Multiple Languages, in *Text, Speech, and Dialogue* (eds. P. Sojka, I. Kopeček, K. Pala and A. Horák), **12284** (2020), 153–161. https://doi.org/10.1007/978-3-030-58323-1_16
12. M. Kaneko, D. Bollegala, Dictionary-based Debiasing of Pre-trained Word Embeddings, in *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, (2021), 212–223. https://doi.org/10.18653/v1/2021.eacl-main.16
13. C. Kong, L. Yang, T. Zhang, Q. Fan, Z. Liu, Y. Chen, E. Yang, Toward Cross-Lingual Definition Generation for Language Learners, *arXiv:2010.05533 [cs]*.
14. M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, L. Zettlemoyer, BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension, in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, (2020), 7871–7880. https://doi.org/10.18653/v1/2020.acl-main.703
15. J. Li, Y. Bao, S. Huang, X. Dai, J. Chen, Explicit Semantic Decomposition for Definition Generation, in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, (2020), 708–717. https://doi.org/10.18653/v1/2020.acl-main.65
16. C.-Y. Lin, ROUGE: A package for automatic evaluation of summaries, in *Text Summarization Branches Out*, (2004), 74–81.
17. T. Mickus, D. Paperno, M. Constant, Mark my word: A sequence-to-sequence approach to definition modeling, in *Proceedings of the First NLPL Workshop on Deep Learning for Natural Language Processing*, (2019), 1–11.

18. T. Mikolov, K. Chen, G. Corrado, J. Dean, Efficient Estimation of Word Representations in Vector Space, *arXiv:1301.3781 [cs]*.

19. T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, J. Dean, Distributed representations of words and phrases and their compositionality, in *NIPS*, 2013.

20. K. Ni, W. Y. Wang, Learning to explain non-standard English words and phrases, in *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, (2017), 413–417.

21. T. Noraset, C. Liang, L. Birnbaum, D. Downey, Definition Modeling: Learning to define word embeddings in natural language, *arXiv:1612.00394 [cs]*.

22. K. Papineni, S. Roukos, T. Ward, W.-J. Zhu, Bleu: a method for automatic evaluation of machine translation, in *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, (2002), 311–318. https://doi.org/10.3115/1073083.1073135

23. M. Reid, E. Marrese-Taylor, Y. Matsuo, VCDM: Leveraging Variational Bi-encoding and Deep Contextualized Word Representations for Improved Definition Modeling, in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, (2020), 6331–6344. https://doi.org/10.18653/v1/2020.emnlp-main.513

24. K. Washio, S. Sekine, T. Kato, Bridging the Defined and the Defining: Exploiting Implicit Lexical Semantic Relations in Definition Modeling, in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, (2019), 3521–3527. https://doi.org/10.18653/v1/D19-1357

25. H. Zhang, Y. Du, J. Sun, Q. Li, Improving interpretability of word embeddings by generating definition and usage, *Expert Syst. Appl.*, **160** (2020), 113633. https://doi.org/10.1016/j.eswa.2020.113633

26. T. Zhang, V. Kishore, F. Wu, K. Q. Weinberger, Y. Artzi, BERTScore: Evaluating Text Generation with BERT, *arXiv:1904.09675 [cs]*.

27. H. Zheng, D. Dai, L. Li, T. Liu, Z. Sui, B. Chang, Y. Liu, Decompose, Fuse and Generate: A Formation-Informed Method for Chinese Definition Generation, in *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, (2021), 5524–5531. https://doi.org/10.18653/v1/2021.naacl-main.437

28. R. Zhu, T. Noraset, A. Liu, W. Jiang, D. Downey, Multi-sense Definition Modeling using Word Sense Decompositions, *arXiv:1909.09483 [cs]*.

AIMS Press