Generationary or: "How We Went beyond Word Sense Inventories and Learned to Gloss"

Michele Bevilacqua[*1], **Marco Maru**[*1,2], and **Roberto Navigli**[1]

[1]Sapienza NLP Group
Department of Computer Science, Sapienza University of Rome
[2]Department of Literature and Modern Cultures, Sapienza University of Rome
firstname.lastname@uniroma1.it

Abstract

Mainstream computational lexical semantics embraces the assumption that word senses can be represented as discrete items of a predefined inventory. In this paper we show this needs not be the case, and propose a unified model that is able to produce contextually appropriate definitions. In our model, Generationary, we employ a novel span-based encoding scheme which we use to fine-tune an English pre-trained Encoder-Decoder system to generate glosses. We show that, even though we drop the need of choosing from a predefined sense inventory, our model can be employed effectively: not only does Generationary outperform previous approaches in the generative task of Definition Modeling in many settings, but it also matches or surpasses the state of the art in discriminative tasks such as Word Sense Disambiguation and Word-in-Context. Finally, we show that Generationary benefits from training on data from multiple inventories, with strong gains on various zero-shot benchmarks, including a novel dataset of definitions for free adjective-noun phrases. The software and reproduction materials are available at http://generationary.org.

1 Introduction

Virtually all modern approaches to Word Sense Disambiguation (WSD), i.e. the task of automatically mapping a word in context to its meaning (Navigli, 2009), use predetermined word senses from a machine lexicon, both in supervised (Huang et al., 2019; Bevilacqua and Navigli, 2020; Scarlini et al., 2020b) and in knowledge-based settings (Tripodi and Navigli, 2019; Scarlini et al., 2020a; Scozzafava et al., 2020). Nevertheless, researchers in Natural Language Processing (NLP), lexical semantics, and lexicography, have long been warning the community about the cognitively inaccurate nature of discrete sense boundaries (Rosch and Mervis, 1975; Kilgarriff, 1997; Tyler and Evans, 2001). As Kilgarriff (2007) argued, different language users have different understandings of words. This fact explains why inter-annotator agreement (ITA) estimates on WSD annotation tasks have never exceeded the figure of 80% (Edmonds and Kilgarriff, 2002; Navigli et al., 2007; Palmer et al., 2007). Moreover, this casts doubt upon the reliability of human-made inventories and "gold standard" evaluation datasets (Ramsey, 2017). Having no indisputable way of determining where one sense of a word ends and another begins, together with the fact that little consensus about how to represent word meaning has hitherto existed (Pustejovsky, 1991; Hanks, 2000; Nosofsky, 2011), are issues lying at the core of what makes WSD hard (Jackson, 2019). Moreover, while English inventories of senses and corpora are widely available the same cannot be said for other languages (Scarlini et al., 2019; Barba et al., 2020; Pasini, 2020), and this limits the scalability of Natural Language Understanding tasks to multiple languages (Navigli, 2018).

In this paper we overcome these limitations by proposing a unified approach to computational lexical semantics that has as its central focus Definition Modeling (DM), i.e. the task of generating a gloss[1] from static or contextual embeddings (Noraset et al., 2017). Generating a meaning description (*definiens*) to define a given term in context (*definiendum*) avoids many of the concerns highlighted above, in that we are not limited to a preexisting list of meanings. We show that we can use a single generation model, i.e. Generationary, not just to compete on the DM benchmarks, but also to achieve strong results on fully-discriminative tasks such as WSD and the recently-proposed Word-in-Context (Pilehvar and Camacho-Collados, 2019,

[*]These authors contributed equally.

[1]To ensure better readability, here we will use the term "gloss" as a synonym of the traditional dictionary "definition".

Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, pages 7207–7221,
November 16–20, 2020.

WiC). This, in turn, results in a more solid assessment of the generation quality, a notorious problem in Natural Language Generation (NLG) evaluation (Gatt and Krahmer, 2018).

In contrast to previous approaches in DM (Gadetsky et al., 2018), we dispense with the requirement of having the *definiendum* represented by a single vector, and we condition gloss generation on a context of which the *definiendum* is an arbitrary span. This allows for the generation of contextual definitions for items that are rarely covered by sense inventories, such as free word combinations (e.g. *clumsy apology* or *nutty complexion*). Finally, the generative formulation makes it possible to train on several lexicographic resources at once, resulting in a versatile model that performs well across inventories, datasets, and tasks.

The main contributions of our approach are as follows:

1. We propose the use of a single conditional generation architecture to perform English DM, WSD and WiC;
2. Our model achieves competitive to state-of-the-art results despite dropping the need of choosing from a predefined sense inventory;
3. Thanks to our encoding scheme, we can represent the *definiendum* as a span in the context, thus enabling definition generation for arbitrary-sized phrases, and seamless usage of BART (Lewis et al., 2019), a pre-trained Encoder-Decoder model;
4. Additionally, we release a new evaluation dataset to rate glosses for adjective-noun phrases.

We envision many possible applications for Generationary, such as aiding text comprehension, especially for second-language learners, or extending the coverage of existing dictionaries.

2 Related Work

Recent years have witnessed the blossoming of research in Definition Modeling (DM), whose original aim was to make static word embeddings intepretable by producing a natural language definition (Noraset et al., 2017).[2] While subsequently released datasets have included usage examples to account for polysemy (Gadetsky et al., 2018; Chang et al., 2018), many of the approaches to "contextual" DM have nevertheless exploited the context merely in order to select a static sense embedding from which to generate the definition (Gadetsky et al., 2018; Chang et al., 2018; Zhu et al., 2019). Such embeddings, however, are non-contextual.

Other works have made a fuller use of the sentence surrounding the target, with the goal of explaining the meaning of a word or phrase as embedded in its local context (Ni and Wang, 2017; Mickus et al., 2019; Ishiwatari et al., 2019). However, these approaches have never explicitly dealt with WSD, and have shown limits regarding the marking of the target in the context encoder, preventing an effective exploitation of the context and making DM overly reliant on static embeddings or surface form information. For example, in the model of Ni and Wang (2017), the encoder is unaware of the contextual target, whereas Mickus et al. (2019) use a marker embedding to represent targets limited to single tokens. Finally, Ishiwatari et al. (2019) replace the target with a placeholder, and the burden of representing it is left to a character-level encoder and to static embeddings. This latter approach is interesting, in that it is the only one that can handle multi-word targets; however, it combines token embeddings via order-invariant sum, and thus it is suboptimal for differentiating instances such as *pet house* and *house pet*.

Recent approaches have explored the use of large-scale pre-trained models to score definitions with respect to a usage context. For example, Chang and Chen (2019) proposed to recast DM as a definition ranking problem. A similar idea was applied in WSD by Huang et al. (2019), leading to state-of-the-art results. However, both of these approaches fall back on the assumption of discrete sense boundaries, and are therefore unable to define targets outside a predefined inventory.

With Generationary, by contrast, we are the first to use a single Encoder-Decoder model to perform diverse lexical-semantic tasks such as DM, WSD and WiC. Moreover, we address the issue of encoding the target in context by using a simple, yet effective, encoding scheme which makes use of special tokens to mark the target *span*, producing a complete and joint encoding of the context without the need for other components. This allows the effective usage of a pre-trained model, which we fine-tune to generate a gloss given the context.

[2] With one single exception (Yang et al., 2020), DM has only been concerned with the English language.

3 Generationary

With this work we present a new approach to computational lexical semantics, by means of which we generate glosses for arbitrary-sized phrases in context. Our work has a wider scope than its predecessors, in that we put forward a unified method that overcomes the limits of both DM and WSD. With respect to DM, our full sequence-to-sequence framing of the task enables us to deal with units having different compositional complexity, from single words to compounds and phrases. Thus, Generationary can gloss a *definiendum* that is not found in dictionaries, such as *starry sky*, with the appropriate *definiens*, e.g.: 'The sky as it appears at night, especially when lit by stars'.

As regards WSD, instead, we are no longer bound by the long-standing limits of predefined sense inventories. Thus, it is possible to give (i) a meaningful answer for words that are not in the inventory, and (ii) one that fits the meaning and the *granularity* required by a given context better than any sense in the inventory. Consider the following:

(1) (a) Why cannot we teach our children to read, write and reckon?
(b) Mark or trace on a surface.
(c) To be able to mark coherent letters.

The target word in (1 a) is associated[3] with the gold gloss (1 b) from WordNet (Fellbaum, 1998), the most used sense inventory in WSD. However, Generationary arguably provides a better gloss (1 c). In what follows, we detail our approach.

3.1 Gloss Generation

In this work we address the task of *mapping* an occurrence of a target word or phrase t (in a context c) to its meaning, by reducing it to that of *generating* a textual gloss g which defines $\langle c, t\rangle$. The target t is a span in c, i.e. a pair of indices $\langle i, j\rangle$ corresponding to the first and the last token which make up the target in c. Formally, we propose to apply the standard sequence-to-sequence conditional generation formulation, in which the probability of a gloss, given a context-target pair, is computed by factorising it auto-regressively:

$$P(g|c,t) = \prod_{k=1}^{|g|} P(g_k|g_{0:k-1}, c, t) \qquad (1)$$

where g_k is the kth token of g and g_0 is a special start token. By means of this procedure we can readily perform contextual DM ($t \neq \langle 1, |c|\rangle$), as well as "static" DM, i.e. when the target encompasses the whole context ($t = \langle 1, |c|\rangle$).

To learn the function in Eq. (1) we employ a recent Encoder-Decoder model, i.e. BART (Lewis et al., 2019), which is pre-trained to reconstruct text spans on massive amounts of data. The use of a pre-trained model is particularly important in our case, as successfully generating a gloss for a wide range of different context-target pairs requires a model which can wield vast amounts of semantic and encyclopedic knowledge. BART can be fine-tuned to perform specific kinds of conditional generation by minimizing the cross-entropy loss on new training input-output pairs. In our approach we give as input to BART a $\langle c, t\rangle$ pair, and train to produce the corresponding gold gloss g, with $\langle c, t\rangle$ and g being gathered from various sources (see Section 4.1). We devise a simple encoding scheme that allows us to make the model aware of the target boundaries, without architectural modifications to BART. Particularly, we encode $\langle c, t\rangle$ pairs as sequences of subword tokens in which the boundaries of the t span in c are marked by two special tokens, i.e. `<define>` and `</define>`. For example, the sentence *I felt like the fifth wheel*, with the phrase fifth wheel as the target, will be encoded as `I felt like the <define> fifth wheel </define>`. We fine-tune BART to generate the corresponding gloss g: `(idiomatic, informal) Anything superfluous or unnecessary`.

3.2 Discriminative Sense Scoring

In this section we introduce three distinct techniques by means of which Generationary tackles discriminative tasks without additional training.

3.2.1 Gloss Probability Scoring

With Eq. (1) we are able to compute the probability of a certain gloss g given a pair $\langle c, t\rangle$. Thus, we can perform classification by picking the sense which is associated with the gloss with the highest probability. Formally, we select:

$$\hat{s} = \underset{s \in S_t}{\operatorname{argmax}}\, P(\mathcal{G}(s)|c,t) \qquad (2)$$

where $S_t \subset S$ is the set of applicable senses for target t from the full inventory S, and $\mathcal{G} : S \rightarrow G$ is a function mapping senses to glosses ($\mathcal{G}$, G, S and S_t are determined by the reference dictionary).

[3] According to the human annotators of the Senseval-2 WSD evaluation dataset (Edmonds and Cotton, 2001).

3.2.2 Gloss Similarity Scoring

The usage of model gloss probability does not take into account the definitions that are actually generated. Thus, we adopt a simple best match approach where we compute similarity scores between the system-generated gloss and the glosses associated with the candidates, and we predict the candidate with the highest similarity. We employ a cosine similarity between the gloss vectors produced via the recently introduced Sentence-BERT model (Reimers and Gurevych, 2019, SBERT), and select a predicted sense $\hat{s}$ as follows:

$$\hat{s} = \underset{s \in S_t}{\operatorname{argmax}}\, sim(\hat{g}, \mathcal{G}(s)) \quad (3)$$

where $\hat{g}$ is the most probable output found by beam-search decoding, and sim is the SBERT similarity.

3.2.3 Gloss Similarity Scoring with MBRR

Using just the most probable sequence in the decoding process for the best match search is suboptimal, as more probability mass might be cumulatively assigned to a cluster of very similar outputs. To take this into account we propose the use of a simple approach inspired by Minimum Bayes Risk Re-Ranking (Kumar and Byrne, 2004, MBRR), which considers the mutual (dis)similarity within the set $\hat{G}$ of k generated outputs decoded with beam search. This is done by rescoring each output as the sum of the dissimilarities over all k outputs, weighted by their conditional probability:

$$\hat{g} = \underset{\hat{g}_i \in \hat{G}}{\operatorname{argmin}} \sum_{\hat{g}_j \in \hat{G}} (1 - sim(\hat{g}_i, \hat{g}_j)) P(\hat{g}_j | c, t) \quad (4)$$

The new prediction $\hat{g}$ is then plugged into Eq. (3) as in simple similarity-based scoring.

4 Datasets

4.1 Dictionary Gloss Datasets

We now move on to describe the datasets which we use to train Generationary models by fine-tuning BART. Each dataset includes $\langle c, t, g \rangle$ triples, which are used as our input and output for training.

CHA (Chang and Chen, 2019) is an online dataset[4] of examples and definitions from `oxforddictionaries.com`. It comes with two settings, each with its own train/dev/test splits: in the *Seen* setting (**CHA**$_S$), definitions in the training set are also present in the test set, while the *Unseen* setting (**CHA**$_U$) has a zero-shot test of lemmas not featured in the training set.

[4] `github.com/MiuLab/GenDef`

	instances			unique glosses		
dataset	**train**	**dev**	**test**	**train**	**dev**	**test**
CHA$_S$	555,695	78,550	151,306	78,105	32,953	37,400
CHA$_U$	530,374	70,401	15,959	73,104	29,540	3,958
SEM	333,633	-	-	116,698	-	-
UNI	1,832,302	-	-	947,524	-	-

Table 1: Training, dev and test instances and number of unique glosses in the datasets used.

SEM is a dataset built by exploiting the SemCor corpus (Miller et al., 1993) – which is manually tagged with WordNet senses – to associate sentence-level contexts with definitions. We filter out NER-like sense annotations (e.g. those mapping proper names such as *Frank Lloyd Wright* to the general sense of *person*). Moreover, to improve coverage, since not all WordNet senses appear in SemCor, we use synonymy information to build additional contexts, e.g. `<define> separate, part, split </define>` → `go one's own way; move apart.`

UNI is the concatenation of the train splits of SEM and CHA, plus the following: (i) a cleaned-up January 2020 dump of Wiktionary, from which circular definitions (e.g. starting with *synonym of*) have been filtered out, and (ii) the training split containing data from the GNU Collaborative International Dictionary of English (GCIDE), included in the dataset of Noraset et al. (2017).

We use CHA and SEM since they were employed by state-of-the-art approaches to DM (Chang and Chen, 2019) and WSD (Huang et al., 2019). With UNI, instead, we bring together diverse sense inventories to create a dataset that is less dependent on the idiosyncrasies of each of its sources. We report statistics in Table 1.

4.2 The Hei++ Evaluation Dataset

As of now, there is no publicly available dataset enabling the assessment of definition generation quality on free phrases (e.g. *exotic cuisine*), which are not commonly found in traditional dictionaries and benchmarks. Thus, we present Hei++, a dataset which associates human-made definitions with adjective-noun phrases. With Hei++ we can test the ability of Generationary to generate glosses, in a zero-shot setting, for items which are not featured in the training set. We encourage the community to use it for future evaluations.

As a first step in building Hei++ we retrieve the

test split of the HeiPLAS dataset (Hartung, 2016),[5] which we choose as our starting point since it contains commonly used adjective-noun phrases. After removing duplicates and discarding ill-formed phrases, we ask an expert lexicographer to write a single definition for each adjective-noun pair. At the end of the annotation process we obtain a dataset made up of 713 adjective-noun phrases with their definitions to be used as a gold standard.

5 Quantitative Experiments

We first perform a threefold automatic evaluation to test the strengths of Generationary in different settings. On the one hand, we assess its ability to produce suitable definitions by testing the generation quality on the DM setting (Section 5.1). On the other, we aim to further appraise how well the generated outputs describe the contextual meaning, by evaluating the performance they bring about on the discriminative benchmarks of WSD (Section 5.2) and WiC (Section 5.3).[6]

5.1 Definition Modeling

In this experiment we use different NLG measures to automatically assess how well generated definitions match gold glosses. We evaluate on the *Seen* (CHA_S) and *Unseen* (CHA_U) test splits of CHA, which is the largest contextual DM benchmark released so far. Moreover, we report results on our Hei++ (HEI) dataset of adjective-noun phrases. We do not include results on the datasets of Noraset et al. (2017) and Gadetsky et al. (2018), as the former only includes targets with no surrounding context, and the latter is largely included in CHA.[7]

5.1.1 Systems

For each evaluation dataset D we test two Generationary models: one trained on the corresponding train split (Gen-D), and one trained on UNI (Gen-UNI).[8] We compare against (i) a random baseline which predicts, for each test item, a random definition taken from the same test set; (ii) the model of Ishiwatari et al. (2019), which we have re-trained on the same data as Generationary (Ishiwatari-D), and (iii) the state-of-the-art approach of Chang and Chen (2019, Chang). On HEI, which has no training split, we only evaluate Gen-UNI and the random baseline, since Ishiwatari-UNI generates strings consisting of mostly unknown word placeholders (<unk>), and Chang and Chen (2019) cannot handle multi-word targets.

5.1.2 Measures

Previous approaches have employed both perplexity (PPL) and string-matching measures (e.g. BLEU) for scoring DM systems. PPL is very appropriate when, as in DM, there are many possible "good" answers.[9] PPL, however, produces a score just on the basis of a pre-existing gold definition, by collecting teacher forcing probabilities without taking into account any actual output generated through beam-search decoding, and thus not assessing the quality of the generation. To evaluate this quality, BLEU and ROUGE-L (Lin, 2004) are also reported. Note, however, that these two measures are based on simple string matches which, in many cases, are not good indicators of output quality. To counteract this problem, we also report results with METEOR (Banerjee and Lavie, 2005) – which uses stemming and WordNet synonyms – and BERTScore (Zhang et al., 2019), which uses vector-based contextual similarities.[10] Finally, to present a complete comparison against the ranking-based approach of Chang and Chen (2019), we report results (precision@k) on their retrieval task of recovering the correct definition, for the target in context, from the whole inventory of 79,030 unique glosses in their dataset. We rank definitions by applying the MBRR plus cosine similarity strategy described in Section 3.2.3.

5.1.3 Results

As shown in Table 2, Generationary models outperform competitors in every setting. On CHA_S, our specialized model (Gen-CHA_S) shows much better results than Gen-UNI, because NLG measures give high scores to glosses which are lexically similar to the gold ones, while multi-inventory training will, instead, expose the model to many other, differently formulated, but equally valid definitions. Note, moreover, that our Gen-CHA_S model outperforms both Ishiwatari et al. (2019) and Chang and Chen (2019), even though the latter, being a ranking model, is obviously at an advantage, since it gets a perfect score when it ranks the gold definition first. In CHA_U we observe that the Gen-UNI model

[5] www.cl.uni-heidelberg.de/~hartung/data

[6] Hyperparameters are documented in Appendix B.

[7] Results on these datasets are reported in Appendix C.

[8] To ensure a fair comparison when evaluating on the *Unseen* setting of CHA, we have removed lemmas appearing in the CHA_U test set from the UNI training set.

[9] See Appendix D for details on perplexity computation.

[10] See Appendix E for configuration details.

	model	ppl↓	BL↑	R-L↑	MT↑	BS↑
CHA_S	Random	-	0.2	10.8	3.2	68.1
	Chang	-	74.7	78.3	-	-
	Ishiwatari-CHA_S*	-	6.2	28.2	11.1	74.2
	Ishiwatari-UNI*	-	3.0	23.2	8.2	72.6
	Gen-CHA_S	**1.2**	**76.2**	**78.9**	**54.8**	**93.0**
	Gen-UNI	1.4	66.9	72.0	47.0	90.7
CHA_U	Random	-	0.3	11.0	3.2	68.2
	Chang	-	7.1	19.3	-	-
	Ishiwatari-CHA_U*	-	2.1	19.9	7.1	71.7
	Ishiwatari-UNI*	-	2.1	19.7	6.7	71.5
	Gen-CHA_U	20.3	8.1	28.7	12.7	76.7
	Gen-UNI	**15.4**	**8.8**	**29.4**	**13.5**	**76.8**
HEI	Random	-	1.6	12.7	0.4	73.4
	Gen-UNI	**16.0**	**6.3**	**26.3**	**15.1**	**78.9**

Table 2: DM evaluation results. Columns: perplexity, BLEU, Rouge-L, METEOR, BERTScore (ppl/BL/R-L/MT/BS). Row groups are mutually comparable (**bold** = best). ↑/↓: higher/lower is better. *: re-trained.

attains higher performances than Gen-CHA_U, indicating that, when 'overfitting' on the inventory is factored out, multi-inventory training enables the model to generalize better in a zero-shot setting. Furthermore, figures for HEI are in the same ballpark as those on CHA_U, demonstrating that Generationary can easily deal, not only with unseen lemmas, but also with entirely different kinds of target.

Additionally, in Table 3 we report the results of the precision@k evaluation when macro-averaging on lemmas (left) and senses (right). Figures on the two different splits of CHA show very different trends. On the CHA_S setting, the base model from Chang and Chen (2019) achieves, in most cases, the highest recovery rate. However, with $k = 1$, which is the most realistic case, Gen-CHA_S outperforms the competitor by 4.6 points when macro-averaging on senses, i.e. items with the same gold definition. On the more challenging zero-shot CHA_U setting, both Generationary models strongly outperform Chang (large), more than doubling the performance on $k = 1$ and showing an improvement of more than 75% on $k = 10$. Gen-UNI, which was underperforming Gen-CHA_S in the *Seen* setting, now achieves better results across the board, since it can exploit the supervision of a wide array of different glosses from multiple inventories.

5.2 WSD Evaluation

We now move to the assessment of Generationary in a traditional WSD setting. Even though our approach goes beyond fixed sense inventories, here we want to show that this degree of freedom does not come at the expense of performance when presented with the task of choosing a sense from a finite predefined list.

	model	P@k (lemmas) 1	5	10	P@k (senses) 1	5	10
CHA_S	Chang (base)	**74.8**	**83.3**	**85.5**	63.3	**74.0**	**77.1**
	Chang (large)	73.9	82.6	84.9	62.4	73.2	76.3
	Gen-CHA_S	73.0	77.7	79.4	**67.9**	72.9	74.7
	Gen-UNI	63.0	70.2	72.7	55.5	63.1	65.8
CHA_U	Chang (base)	3.3	9.6	14.4	2.3	7.4	11.4
	Chang (large)	3.5	10.5	15.6	2.5	8.2	12.4
	Gen-CHA_U	7.8	19.9	25.5	6.5	16.8	22.0
	Gen-UNI	**9.3**	**21.3**	**27.7**	**7.4**	**18.0**	**23.8**

Table 3: Macro precision@k (lemmas and senses) on the retrieval task of Chang and Chen (2019). Row groups are mutually comparable (**bold** = best).

We test on the five datasets collected in the evaluation framework of Raganato et al. (2017), namely: Senseval-2 (Edmonds and Cotton, 2001), Senseval-3 (Snyder and Palmer, 2004), SemEval-2007 (Pradhan et al., 2007), SemEval-2013 (Navigli et al., 2013), SemEval-2015 (Moro and Navigli, 2015), which are all annotated with WordNet 3.0 senses (or converted to its inventory). We denote with ALL and ALL^- the concatenation of all evaluation datasets, including or excluding, respectively, SemEval-2007, which is our development set for this experiment. Moreover, we test on the subset of ALL^- containing instances whose lemmas are not covered in SemCor (0-shot).

5.2.1 Systems

To choose a possible sense from WordNet and perform WSD, we evaluate the techniques presented in Section 3.2, i.e. probability scoring (Prob.), simple similarity scoring (Sim.), and similarity scoring with MBRR. We evaluate our Gen-SEM, which is trained on examples specifically tagged according to the WordNet inventory, and Gen-UNI, which includes definitions from many different inventories. We compare against recent WSD approaches which make use of gloss knowledge, i.e. LMMS (Loureiro and Jorge, 2019) and the state-of-the-art approach of GlossBERT (Huang et al., 2019).

5.2.2 Results

We report the results of the WSD evaluation in Table 4. The MBRR scoring strategy proves to be the most versatile, with Gen-SEM (MBRR) achieving a higher F1 than Gen-SEM (Prob.) on almost every dataset, and outperforming Gen-SEM (Sim.) on

model	S2	S3	S7	S13	S15	ALL	ALL$^-$	0-shot	N	V	A	R
LMMS$_{2348}$	76.3	75.6	68.1	75.1	77.0	75.4	75.9*	66.3*	78.0*	64.0*	**80.7***	83.5*
GlossBERT	77.7	75.9	**72.1**	76.8	**79.3**	**77.0**	**77.2***	68.7*	79.7*	**66.5***	79.3*	85.5*
Gen-SEM (Prob.)	76.9	73.7	69.2	74.6	78.2	75.3	75.7	60.6	77.5	65.0	78.4	**87.6**
Gen-SEM (Sim.)	77.5	**76.4**	71.6	76.8	77.4	76.7	77.0	63.3	**80.1**	64.8	79.1	85.0
Gen-SEM (MBRR)	**78.0**	75.4	71.9	77.0	77.6	76.7	77.0	65.0	79.9	64.8	79.2	86.4
Gen-UNI (MBRR)	77.8	73.7	68.8	**78.3**	77.6	76.3	76.8	**73.0**	79.8	63.3	80.1	84.7

Table 4: Results on the WSD evaluation. Row groups: (1) previous approaches; (2) Generationary. Columns: datasets in the evaluation framework (S2 to S15), ALL w/ and w/o the dev set (ALL/ALL$^-$), zero-shot set (0-shot), and results by PoS on ALL (N/V/A/R). F1 is reported. **Bold**: best. *: re-computed with the original code.

the 0-shot set. As both Sim. and MBRR outscore Prob., it is clear that generating a gloss and ranking candidates with similarity is a better strategy than directly ranking with model probability, which leaves room for further improvement as better similarity measures are developed.

On another note, Gen-SEM (MBRR) achieves performances which are overall comparable with those of the state of the art (GlossBERT) without having been explicitly trained to perform WSD. Compared to Gen-SEM (MBRR), Gen-UNI (MBRR) sacrifices 0.4 and 0.2 points on, respectively, ALL and ALL$^-$, but obtains 8 points more on the zero-shot set, also improving over GlossBERT by 4.3 points. This demonstrates that, when using Generationary with data from multiple inventories, (i) performances remain in the same ballpark as those of a state-of-the-art system, and (ii) much improved generalizability is achieved, as shown by the state-of-the-art results on the zero-shot setting.

5.3 Word-in-Context

In the task of Word-in-Context (WiC) (Pilehvar and Camacho-Collados, 2019), predefined sense inventories are not required and meaning identification is reduced to a binary problem in which, given two contexts, both featuring an occurrence of the same lemma, a model has to predict whether the two targets have the same meaning. We compare against Chang and Chen (2019), which is the only DM approach reporting results for WiC, following their setting in which no task-specific training is performed and the training set for the task is used for testing. Results are reported for both Gen-CHA$_S$, which is trained on the same data as Chang and Chen (2019), and Gen-UNI.[11]

To perform the task, for each pair in the WiC dataset we generate two sets, γ and γ', each of 10 glosses, for the two respective sentences in the pair. Then, for each generated gloss $\hat{g} \in \gamma$, we compute the score $z_{\hat{g}}$ as the mean SBERT similarity between $\hat{g}$ and the 10 generated glosses in γ'. Analogously, we compute $z_{\hat{g}'}$ as the mean similarity between $\hat{g}' \in \gamma'$ and the glosses in γ. For each gloss g we normalize z_g by subtracting an approximate mean similarity of g with random glosses, computed as the mean similarity between g and all other unrelated glosses in the batch. If the mean score $(\sum_{\hat{g}\in\gamma} z_{\hat{g}} + \sum_{\hat{g}'\in\gamma'} z_{\hat{g}'})/20$ exceeds a threshold t (tuned on the dev set), we predict that a WiC pair shares the same sense.

Gen-CHA$_S$, with an accuracy of 69.2, outperforms Chang and Chen (2019), which achieves 68.6, demonstrating the strength of our approach in this setting. Moreover, Gen-UNI, which attains a result of 71.1, outscores both Gen-CHA$_S$ and the competitor, once again bearing witness to the versatility of training on multiple inventories.

6 Qualitative Experiment

Given that the ability of Generationary to produce fluent and meaningful definitions is its key asset, in addition to the automatic evaluation reported in Section 5 we devised a qualitative experiment on two distinct datasets we constructed. While our previous experiments shed light upon the quality of Generationary in comparison with other automatic systems, here we employ human annotators to compare definitions produced with our approach against glosses written by human lexicographers.

The datasets that we use in this experiment are (i) our Hei++ dataset of definitions for adjective-nouns phrases (Section 4.2) and (ii) SamplEval, i.e. a sample of 1,000 random instances made up of 200 items[12] for each of the five WSD datasets included in ALL (see Section 5.2), with at most one

[11] In this experiment we have excluded Wiktionary, which was used to build the WiC dataset, from the UNI training set.

[12] We do not sample instances annotated with many senses.

dataset	gold	Gen.	≥
Hei++	4.43	3.58	29.9
SamplEval	3.75	3.62	51.3

Table 5: Qualitative evaluation results. Columns: dataset, average Likert for gold and Generationary, % of Generationary scores equal or better than gold (≥).

total instance per sense. With Hei++ we assess the ability of Generationary to accurately gloss complex expressions, such as free phrases (e.g. *wrong medicine* or *hot forehead*), that are rarely covered by traditional dictionaries. With SamplEval, instead, we test whether generated glosses can improve over gold definitions associated with gold senses in WordNet.

6.1 Annotators and Annotation Scheme

For each context-target pair in Hei++ and SamplEval we have two definitions: a gold one, written by a lexicographer, and one generated by Gen-UNI, which is not tied to any specific inventory and has proven the most versatile model across tasks. We hired three annotators with Master's Degrees in Linguistics and effective operational proficiency in English and, in a similar fashion to Erk and McCarthy (2009), we asked them to assign a graded value to the definitions based on their pertinence to describing the target t in c, according to a five-level Likert scale (see Appendix F).[13] The annotators received a wage in line with the standards of their country of residence, and worked an overall amount of 90 person-hours (30 per annotator). The ITA was substantial, with an average pairwise Cohen's κ of 0.69 (SamplEval) and 0.67 (Hei++).

6.2 Results

As can be seen in Table 5, the quality of Generationary glosses in the SamplEval dataset is comparable to those drawn from WordNet. Note that, although it would be expected for gold annotations to come close to the top of the scale, this is not the case, as they received an average score of 3.75 out of 5, demonstrating the suboptimal nature of "ready-made" meaning distinctions. We report comparable scores on the Hei++ dataset. The gap with respect to gold definitions here is wider, probably because (i) Generationary is not specifically trained on complex expressions, and (ii) the gold score is higher since phrases are less ambiguous than single words. Interestingly, the annotators rated Generationary glosses at least as high as their gold counterparts on 51.3% and on 29.9% of the total cases on SamplEval and Hei++, respectively: this result provides evidence for the reliability of Generationary definitions as valid alternatives to glosses taken from established inventories of discrete word senses.

[13] We presented glosses for each sentence in random order.

c_1	[...] I scooted them into the dog run.
$\hat{g}_1$	Cause to move along by pushing.
g_1	Run or move very quickly or hastily.
c_2	Exotic cuisine.
$\hat{g}_2$	A style of cooking that is out of the ordinary and unusual (as if from another country).
g_2	Cuisine involving unfamiliar foods.
c_3	He was never the same after the accident.
$\hat{g}_3$	Indicates that a person has lost the good qualities that were present before the accident.
c_4	Sam is in a better place now.
$\hat{g}_4$	A phrase used to express that one has learned about another's death.
c_5	Yesterday I had to undergo a beardectomy.
$\hat{g}_5$	The surgical removal of the beard.
c_6	You've got a hard coconut to smash here, Dr. Yang!
$\hat{g}_6$	Something difficult to deal with.
c_7	The mind is haunted by the ghosts of the past.
$\hat{g}_7$	People's memories of the past are still present in their mind, even after they have ceased to exist.
c_8	The fault, dear Brutus, is not in our stars, but in ourselves.
$\hat{g}_8$	The responsibility for a problem lies with the people who cannot see it themselves.

Table 6: Sample of Generationary definitions ($\hat{g}$) for several targets in context (c). g: gold definition.

7 Generation Examples

In Table 6 we show a sample of definitions generated via our Gen-UNI model for various spans in context.[14] As can be seen, the glosses $\hat{g}_1$ and $\hat{g}_2$ (extracted from SamplEval and Hei++, respectively) demonstrate that Generationary can indeed provide better, more specific definitions than gold standard ones. The following reported examples show the strength of our model on contexts which do not resemble those it is trained on: Generationary is proficient at (i) handling fixed or semi-fixed

[14] See Appendix A for further samples of generated glosses.

idioms of different lengths ($\hat{g}_3$, $\hat{g}_4$) and (ii) defining non-conventional words and phrases ($\hat{g}_5$, $\hat{g}_6$); interestingly, Generationary is also able to (iii) provide high-level explanations for whole figurative contexts ($\hat{g}_7$, $\hat{g}_8$), which goes well beyond what is commonly referred to as *glossing*. This might result in interesting applications beyond the scope of this work, e.g. for paraphrase generation and metaphor interpretation (Rai and Chakraverty, 2020).

8 Error Analysis

To have a broader picture of the quality of the outputs produced by means of Generationary, we perform behavioural testing for our Gen-UNI model, in the spirit of Ribeiro et al. (2020). As a result, we can identify two main trends behind failures to generate an appropriate contextual definition, which we refer to as *disambiguation errors* and *hallucinations*, respectively.

Disambiguation errors When the model predicts a perfectly good definition for the target, but one that fits another common context of occurrence, a disambiguation error arises. For instance, given the $\langle c, t \rangle$ pair in (2 a), with the word pupil as the target, the model fails to identify the "aperture in the iris of the eye" sense, and instead produces an output gloss which is compatible with the meaning of the homograph (2 b):

(2) (a) The teacher stared into the pupils of her pupil.

(b) A person receiving instruction, especially in a school.

Hallucinations Other errors stem from the fact that the model can only rely on the knowledge about possible *definienda* that it is able to store in the parameters during the pre-training and training stages. Thus, if the contextual knowledge is not sufficient to extrapolate a definition, the model – which is required to always generate an output – will hallucinate an answer on the basis of contextual clues, incurring the risk of introducing non-factualities. This particularly concerns named entities and domain-specific concepts, but the clearest examples can be seen with targets that do not correspond to any existing, fictional or non-fictional entity. For example, given the input sentence (3):

(3) Squeaky McDuck wasn't happy about it,

the model outputs the following:

(4) The title character in the "Squeaky Squeakety-Squeakiness" cartoon series.

In this case, the model picked the cue of the *cartoonish* Squeaky McDuck character, and hallucinated the name of a plausible cartoon series for it. Note that neither Squeaky McDuck nor the cartoon series actually exist.

9 Conclusion

With this work, we showed that generating a definition can be a viable, suitable alternative to the traditional use of sense inventories in computational lexical semantics, and one that better reflects the non-discrete nature of word meaning. We introduced Generationary, an approach to automatic definition generation which, thanks to a flexible encoding scheme, can (i) encode targets of arbitrary length (including unseen multi-word expressions), and (ii) exploit the vast amount of knowledge encoded in the BART pre-trained Encoder-Decoder, through fine-tuning.

From two points of view, Generationary represents a unified approach: first, it exploits multiple inventories simultaneously, hence going beyond the quirks of each one; second, it is able to tackle both generative (Definition Modeling) and discriminative tasks (Word Sense Disambiguation and Word-in-Context), obtaining competitive to state-of-the-art results, with particularly strong performances on zero-shot settings. Finally, human evaluation showed that Generationary is often able to provide a definition that is on a par with or better than one written by a lexicographer.

We make the software and reproduction materials, along with a new evaluation dataset of definitions for adjective-noun phrases (Hei++), available at `http://generationary.org`.

Acknowledgments

The authors gratefully acknowledge the support of the ERC Consolidator Grant MOUSSE No. 726487 under the European Union's Horizon 2020 research and innovation programme.

This work was supported in part by the MIUR under the grant "Dipartimenti di eccellenza 2018-2022" of the Department of Computer Science of Sapienza University.

References

Satanjeev Banerjee and Alon Lavie. 2005. 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*, pages 65–72, Ann Arbor, MI, USA.

Edoardo Barba, Luigi Procopio, Niccolò Campolungo, Tommaso Pasini, and Roberto Navigli. 2020. MuLaN: Multilingual Label propagatioN for word sense disambiguation. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20*, pages 3837–3844.

Michele Bevilacqua and Roberto Navigli. 2020. Breaking through the 80% glass ceiling: Raising the state of the art in word sense disambiguation by incorporating knowledge graph information. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2854–2864, Online.

Ting-Yun Chang and Yun-Nung Chen. 2019. What does this word mean? Explaining contextualized embeddings with natural language definition. 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)*, pages 6064–6070, Hong Kong, China.

Ting-Yun Chang, Ta-Chung Chi, Shang-Chi Tsai, and Yun-Nung Chen. 2018. xSense: Learning sense-separated sparse representations and textual definitions for explainable word sense networks. *arXiv preprint arXiv:1809.03348*.

Philip Edmonds and Scott Cotton. 2001. SENSEVAL-2: Overview. In *Proceedings of SENSEVAL-2 Second International Workshop on Evaluating Word Sense Disambiguation Systems*, pages 1–5, Toulouse, France.

Philip Edmonds and Adam Kilgarriff. 2002. Introduction to the special issue on evaluating word sense disambiguation systems. *Natural Language Engineering*, 8(4):279–291.

Katrin Erk and Diana McCarthy. 2009. Graded word sense assignment. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 440–449, Singapore.

Christiane Fellbaum, editor. 1998. *WordNet: An Electronic Lexical Database*. MIT Press, Cambridge, MA, USA.

Artyom Gadetsky, Ilya Yakubovskiy, and Dmitry Vetrov. 2018. Conditional generators of words definitions. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 266–271, Melbourne, Australia.

Albert Gatt and Emiel Krahmer. 2018. Survey of the state of the art in natural language generation: Core tasks, applications and evaluation. *Journal of Artificial Intelligence Research*, 61:65–170.

Patrick Hanks. 2000. Do word meanings exist? *Computers and the Humanities*, 34(1–2):205–215.

Matthias Hartung. 2016. *Distributional Semantic Models of Attribute Meaning in Adjectives and Nouns*. Ph.D. thesis, Institut für Computerlinguistik Ruprecht-Karls-Universität Heidelberg.

Luyao Huang, Chi Sun, Xipeng Qiu, and Xuanjing Huang. 2019. GlossBERT: BERT for word sense disambiguation with gloss knowledge. 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)*, pages 3509–3514, Hong Kong, China.

Shonosuke Ishiwatari, Hiroaki Hayashi, Naoki Yoshinaga, Graham Neubig, Shoetsu Sato, Masashi Toyoda, and Masaru Kitsuregawa. 2019. Learning to describe unknown phrases with local and global contexts. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pages 3467–3476, Minneapolis, MN, USA.

Philip C. Jr. Jackson. 2019. I do believe in word senses. *Proceedings ACS*, 321:340.

Adam Kilgarriff. 1997. I don't believe in word senses. *Computers and the Humanities*, 31(2):91–113.

Adam Kilgarriff. 2007. Word senses. In Eneko Agirre and Phillip Edmonds, editors, *Word Sense Disambiguation*, pages 29–46. Springer, Dordrecht.

Shankar Kumar and William Byrne. 2004. Minimum Bayes-risk decoding for statistical machine translation. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (HLT-NAACL)*, pages 169–176, Boston, MA, USA.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. 2019. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain.

Edward Loper and Steven Bird. 2002. NLTK: The Natural Language Toolkit. In *Proceedings of the ACL-02 Workshop on Effective Tools and Methodologies for Teaching Natural Language Processing and Computational Linguistics - Volume 1*, ETMTNLP '02, pages 63–70, Stroudsburg, PA, USA.

Daniel Loureiro and Alípio Jorge. 2019. Language modelling makes sense: Propagating representations through WordNet for full-coverage word sense disambiguation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5682–5691, Florence, Italy.

Timothee Mickus, Denis Paperno, and Matthieu Constant. 2019. 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*, pages 1–11, Turku, Finland.

George A. Miller, Claudia Leacock, Randee Tengi, and Ross T. Bunker. 1993. A semantic concordance. In *Human Language Technology: Proceedings of a Workshop Held at Plainsboro, New Jersey, March 21-24, 1993*.

Andrea Moro and Roberto Navigli. 2015. SemEval-2015 task 13: Multilingual all-words sense disambiguation and entity linking. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 288–297, Denver, CO, USA.

Roberto Navigli. 2009. Word sense disambiguation: A survey. *ACM Computing Surveys*, 41(2):1–69.

Roberto Navigli. 2018. Natural language understanding: Instructions for (present and future) use. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI-18*, pages 5697–5702.

Roberto Navigli, David Jurgens, and Daniele Vannella. 2013. SemEval-2013 task 12: Multilingual word sense disambiguation. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 222–231, Atlanta, GA, USA.

Roberto Navigli, Kenneth C. Litkowski, and Orin Hargraves. 2007. SemEval-2007 task 07: Coarse-grained English all-words task. In *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)*, pages 30–35, Prague, Czech Republic.

Ke Ni and William Yang Wang. 2017. Learning to explain non-standard English words and phrases. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (IJCNLP)*, pages 413–417, Taipei, Taiwan.

Thanapon Noraset, Chen Liang, Larry Birnbaum, and Doug Downey. 2017. Definition modeling: Learning to define word embeddings in natural language. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, pages 3259–3266, San Francisco, CA, USA.

Robert M Nosofsky. 2011. The generalized context model: An exemplar model of classification. In Emmanuel M. Pothos and Andy J. Wills, editors, *Formal Approaches in Categorization*, pages 18–39. Cambridge University Press, Cambridge.

Martha Palmer, Hoa Trang Dang, and Christiane Fellbaum. 2007. Making fine-grained and coarse-grained sense distinctions, both manually and automatically. *Natural Language Engineering*, 13(2):137–163.

Tommaso Pasini. 2020. The Knowledge Acquisition Bottleneck Problem in Multilingual Word Sense Disambiguation. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20*, pages 4936–4942.

Mohammad Taher Pilehvar and Jose Camacho-Collados. 2019. WiC: the Word-in-Context dataset for evaluating context-sensitive meaning representations. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pages 1267–1273, Minneapolis, MN, USA.

Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Brussels, Belgium.

Sameer Pradhan, Edward Loper, Dmitriy Dligach, and Martha Palmer. 2007. SemEval-2007 task-17: English lexical sample, SRL and all words. In *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)*, pages 87–92, Prague, Czech Republic.

James Pustejovsky. 1991. The generative lexicon. *Computational Linguistics*, 17(4).

Alessandro Raganato, Jose Camacho-Collados, and Roberto Navigli. 2017. Word sense disambiguation: A unified evaluation framework and empirical comparison. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 99–110, Valencia, Spain.

Sunny Rai and Shampa Chakraverty. 2020. A survey on computational metaphor processing. *ACM Computing Surveys (CSUR)*, 53(2):1–37.

Rachel Ramsey. 2017. *An Exemplar-Theoretic Account of Word Senses*. Ph.D. thesis, Northumbria University.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. 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)*, pages 3982–3992, Hong Kong, China.

Marco Tulio Ribeiro, Tongshuang Wu, Carlos Guestrin, and Sameer Singh. 2020. Beyond accuracy: Behavioral testing of NLP models with CheckList. In

Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 4902–4912, Online.

Eleanor Rosch and Carolyn B Mervis. 1975. Family resemblances: Studies in the internal structure of categories. *Cognitive psychology*, 7(4):573–605.

Bianca Scarlini, Tommaso Pasini, and Roberto Navigli. 2019. Just "OneSeC" for producing multilingual sense-annotated data. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 699–709, Florence, Italy.

Bianca Scarlini, Tommaso Pasini, and Roberto Navigli. 2020a. SensEmBERT: Context-enhanced sense embeddings for multilingual word sense disambiguation. In *Thirty-Fourth AAAI Conference on Artificial Intelligence*, New York, NY, USA.

Bianca Scarlini, Tommaso Pasini, and Roberto Navigli. 2020b. With more contexts comes better performance: Contextualized sense embeddings for all-round Word Sense Disambiguation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, online.

Federico Scozzafava, Marco Maru, Fabrizio Brignone, Giovanni Torrisi, and Roberto Navigli. 2020. Personalized PageRank with syntagmatic information for multilingual word sense disambiguation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 37–46, Online.

Benjamin Snyder and Martha Palmer. 2004. The English all-words task. In *Proceedings of SENSEVAL-3, the Third International Workshop on the Evaluation of Systems for the Semantic Analysis of Text*, pages 41–43, Barcelona, Spain.

Rocco Tripodi and Roberto Navigli. 2019. Game theory meets embeddings: A unified framework for word sense disambiguation. 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)*, pages 88–99, Hong Kong, China.

Andrea Tyler and Vyvyan Evans. 2001. Reconsidering prepositional polysemy networks: The case of over. *Language*, 77(4):724–765.

Liner Yang, Cunliang Kong, Yun Chen, Yang Liu, Qinan Fan, and Erhong Yang. 2020. Incorporating sememes into Chinese definition modeling. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. BERTScore: Evaluating text generation with BERT. *arXiv preprint arXiv:1904.09675*.

Ruimin Zhu, Thanapon Noraset, Alisa Liu, Wenxin Jiang, and Doug Downey. 2019. Multi-sense definition modeling using word sense decompositions. *arXiv preprint arXiv:1909.09483*.

c_1: Good news. $\hat{g}_1$: (New Testament) The gospel as revealed by Jesus to the apostles. g_1: Any news that arouses feelings of joy or eases anxiety.
c_2: Uneven margin. $\hat{g}_2$: A margin that is not uniform. g_2: A margin that is not perfectly leveled.
c_3: Early diagnosis. $\hat{g}_3$: The diagnosis of a condition before symptoms appear. g_3: A diagnosis that is made at an initial stage of a disease.
c_4: Sincere friendship. $\hat{g}_4$: A friendship that is not based on deceit or hypocrisy. g_4: Friendship marked by genuine feelings of benevolence.
c_5: Painful performance. $\hat{g}_5$: A performance of a piece of music that is difficult to play. g_5: A performance that is exceptionally bad.
c_6: Courageous heart. $\hat{g}_6$: A heart that is strong enough to endure adversity. g_6: The feelings of a person that is not afraid of getting hurt.
c_7: Inaccurate thermometer. $\hat{g}_7$: A thermometer that is inaccurate in measuring temperature. g_7: A thermometer that indicates the wrong temperature.
c_8: New friend. $\hat{g}_8$: A friend who has recently come into one's life. g_8: A recently made friend.
c_9: Familiar guest. $\hat{g}_9$: A person who is a regular customer or client of a hotel, restaurant, etc. g_9: A well known guest.
c_{10}: Vivacious hostess. $\hat{g}_{10}$: A woman who entertains guests at their home and makes them feel welcome. g_{10}: A woman host who shows liveliness.

Table 7: Random sample of Generationary definitions ($\hat{g}$) for Hei++ contexts (c). g: gold definition.

A Generation Examples

In the evaluation of NLG systems, human qualitative assessment is very important. Therefore, we choose to report a fair number of non-cherrypicked, zero-shot generation examples, produced by means of our GEN-UNI model.

c_1: Hipster guys #2 & #3. $\hat{g}_1$: Members of the hipster subculture.
c_2: Because you have no estrogen... you could not possibly understand what chocolate means to a woman. $\hat{g}_2$: The absence of estrogen in the body.
c_3: Mozilla Firebird bookmark toolbar doesn't hide command click to open link in new tab. $\hat{g}_3$: A toolbar in the Mozilla browser.
c_4: Guy #2: On that table over there. $\hat{g}_4$: The (usually specified) table on which something is located.
c_5: Man, why do chip manufacturers always put the crumbs at the bottom of the bag? $\hat{g}_5$: A company that manufactures potato chips.
c_6: Install new theme without an internet connection. $\hat{g}_6$: A connection to the internet.
c_7: It's a true story. $\hat{g}_7$: A fictional story based on real events.
c_8: Seeking female friend, 40s, attractive 5 ft 5+. $\hat{g}_8$: A friend who is not a man.
c_9: Clicking Get New Extensions in Options Dialog opens a new browser... $\hat{g}_9$: A dialog box in a web browser, allowing the user to select from a number of options.
c_{10}: And he goes to my other roommate for advice. $\hat{g}_{10}$: The other person with whom one shares a room.

Table 8: Generationary definitions ($\hat{g}$) for random targets and contexts (c) excerpted from `webtext`.

In Table 7 we show Generationary outputs and gold definitions for 10 randomly sampled phrases in the Hei++ dataset. In addition, in Table 8 we report gloss generation examples for random words and noun phrases taken from the `webtext` corpus included in the NLTK suite (Loper and Bird, 2002). We exclude swear words, slurs, numbers, and noun phrases consisting entirely of named entities. Moreover, every sampled item whose target was featured in our training set was filtered out.

B Reproducibility Details

To train our models we employ the `fairseq` library. Generationary has the same number of parameters as BART (Lewis et al., 2019), i.e. ca. 458M. For fine-tuning, we use the same hyperparameters used in Lewis et al. (2019) for summarization,[15] except that:

[15] github.com/pytorch/fairseq/blob/master/examples/bart/README.summarization.md

- the learning rate is set to 5×10^{-5} on the basis of preliminary experiments;
- due to memory concerns, we feed the input in batches of 1,024 tokens, updating every 16 iterations;
- we use inverse square root learning rate scheduling, which does not require to set a maximum number of iterations a priori;
- we double the number of warmup steps to 1,000.

Training is performed for at most 50 epochs. We employ a single NVIDIA GeForce RTX 2080 Ti GPU to perform all the reported experiments, with average runtimes per epoch of BART fine-tuning ranging from ca. 50 minutes (Gen-SEM) to >120 minutes (Gen-UNI).

On the DM task, we evaluate on the best epoch, i.e. the one with the lowest cross-entropy loss on the dev set, with no hyperparameter tuning.

On the WSD task, instead, we perform minimal hyperparameter tuning, with search trials just on beam size (testing with values of 1, 10, 25, and 50), choosing as the best configuration the one with the highest F1 on our dev set, SemEval-2007; with simple similarity scoring, the best Gen-SEM has a beam size of 10, while, with MBRR similarity scoring, the best Gen-SEM has a beam size of 25. We use only MBRR with Gen-UNI, with a beam size of 10, resulting in the best performance on the development set.

On the WiC task we only perform tuning of the threshold on the dev set, by trying every value in range between the lowest and the highest z score, with a minimum step of 0.025. We compute similarities in batches of 125 pairs.

For training and prediction of the models of Ishiwatari et al. (2019), we use the code provided by the authors.[16] We use the same hyperparameters, except that we increase the vocabulary size to 39,000, which results in much improved performances on our benchmarks.

C Additional Results on DM

In Table 9 we report results, for the DM evaluation described in Section 5.1, on two additional datasets.

NOR (Noraset et al., 2017) includes data from the GCIDE and WordNet. It features only "static" pairs, in which the context coincides with the word to be defined. Nonetheless, each lemma can be associated with multiple definitions.

	model	ppl↓	BL↑	R-L↑	MT↑	BS↑
NOR	Random	-	0.2	6.3	1.9	69.0
	Noraset et al. (2017)	48.2	-	-	-	-
	Ishiwatari-NOR*	-	1.9	15.7	5.0	**72.9**
	Gen-NOR	**28.6**	**3.8**	**17.7**	**8.1**	**72.9**
GAD	Random	-	0.2	8.7	2.8	68.6
	Gadetsky et al. (2018)	43.5	-	-	-	-
	Mickus et al. (2019)	34.0	-	-	-	-
	Ishiwatari-GAD*	-	2.5	18.7	7.0	72.8
	Gen-GAD	**12.3**	**9.9**	**28.9**	**12.8**	**77.9**

Table 9: DM evaluation results. Columns: perplexity, BLEU, Rouge-L, METEOR, BERTScore (ppl/BL/R-L/MT/BS). Row groups are mutually comparable (**bold** = best). ↑/↓: higher/lower is better. *: re-trained.

GAD (Gadetsky et al., 2018) collects context-target pairs and definitions from `oxforddictionaries.com`. The target lemma is not present in all contexts, so in these cases we prepend the lemma according to the following template: '*lemma*: *context*'.[17]

D Perplexity

Perplexity captures the confidence of the model in outputting a certain sequence. In approaches with word-level tokenization, evaluated at word-level, perplexity can be computed by exponentiating the negative log-likelihood that is used for training:

$$PPL_w^w = exp(-\sum_{w \in V} P(w|c,t,\bar{h}) \ln \hat{P}(w|c,t,\bar{h})) \quad (5)$$

$$= exp(-\ln \hat{P}(\bar{w}|c,t,\bar{h})) \quad (6)$$

where c is the context, t is the target, V is the vocabulary, $\bar{w}$ is the gold word, and $\bar{h}$ is the gold history of previous tokens. Generationary employs subword-level tokenization, but we can still obtain the word-level probabilities by applying the chain rule of conditional probability:

$$PPL_w^s = exp(-\ln \prod_{i=1}^{|\bar{w}^\star|} \hat{P}(\bar{w}_i^\star|c,t,\bar{h},\bar{w}_{1:i-1}^\star)) \quad (7)$$

where $\bar{w}^*$ is the n-ple that is the subword split of $\bar{w}$, e.g. ⟨ `token`, `##ization` ⟩ for

[16] `github.com/shonosuke/ishiwatari-naacl2019`

[17] The train/dev/test splits of NOR and GAD are disjoint in the lemma of the target words.

tokenization. Do we maintain full comparability? There are two issues here. The first stems from the fact that, thanks to the application of the chain rule, the vocabulary is open, i.e. the support of the subword model is the set of possible words, so that every item receives non-zero probability.

In contrast, a word-level model without some kind of backoff strategy has a closed vocabulary. If the evaluation set includes a word outside V, the closed vocabulary model has a special <unk> token, on which it is trained to concentrate all the probability mass that the open vocabulary model, instead, would spread over all the possible words which are not in V. This entails an unfavorable advantage of the closed vocabulary model over the open vocabulary. Moreover, there is an additional complication arising from the fact that, while the subword tokenizers are usually deterministic, i.e. any word is always split in the same way, there might be multiple legal subword splits depending on the vocabulary, and to obtain the probability of the word we would need to marginalize over all splits. In other words, we would need to marginalize by summing the probability of ⟨ token, ##ization ⟩, ⟨ token, ##iz, ##ation ⟩, ⟨ to, ##ken, ##ization ⟩ and so on. This is very burdensome, and in practice we only consider the deterministic split produced by the tokenizer. In doing this, we underestimate the probability of the word and, thus, overestimate the perplexity of the subword-level model.

E NLG Measures Details

In order to ensure comparability, here we report the BLEU, ROUGE, METEOR, and BERTScore configurations that we used. A scorer is available as part of the provided software.

BLEU We employ the reference implementation of corpus BLEU provided in the sacrebleu package (Post, 2018, https://github.com/mjpost/sacreBLEU). We use default parameters. Signature:
BLEU+case.mixed+numrefs.1+smooth.exp+tok.13a+version.1.3.6.

ROUGE We have employed the Python rouge library (https://github.com/pltrdy/rouge).

METEOR We have employed the Java meteor library (https://www.cs.cmu.edu/~alavie/METEOR), version 1.5. METEOR is calculated using the -norm and -noPunct flags. Signature:
meteor-1.5-wo-en-norm_nopunct-0.85_0.2_0.6_0.75-ex_st_sy_pa-1.0_0.6_0.8_0.6

BERTScore We evaluate using the Python BERTScore (https://github.com/Tiiiger/bert_score) library, with the roberta-large-mnli model and default parameters. Signature:
roberta-large-mnli_L19_no-idf_version=0.3.0(hug_trans=2.8.0)

F Likert Scale

We employ a five-level Likert scale to rank glosses in both the annotation experiments on SamplEval and Hei++ (see Section 6.1). In Table 10 we show one of the annotation examples that were provided to the annotators to be used as guidelines.

	Was he going to be saddled with a creep for a bar-buddy?
1	*Wrong gloss. May refer to a homonym of the target.* A heating element in an electric fire.
2	*Wrong gloss. Captures the domain of the target.* A counter where you can obtain food or drink.
3	*Correct gloss. Utterly vague and generic.* A person with whom you are acquainted.
4	*Correct gloss. Fits the context, but misses some details.* A close friend who accompanies his buddies in their activities.
5	*Correct gloss. Perfectly describes the target in its context.* A friend who you frequent bars with.

Table 10: Annotation guidelines excerpt. Rows: Likert score, *explanation* and example definition for target.