DIBI\(\text{MT}\): A Novel Benchmark for Measuring
Word Sense Disambiguation Biases in Machine Translation

Niccolò Campolungo*  
Sapienza University of Rome  
campolungo@di.uniroma1.it

Federico Martelli*  
Sapienza University of Rome  
martelli@di.uniroma1.it

Francesco Saina  
SSML Carlo Bo, Rome  
f.saina@ssmlcarlobo.it

Roberto Navigli  
Sapienza University of Rome  
navigli@diag.uniroma1.it

Abstract

Lexical ambiguity poses one of the greatest challenges in the field of Machine Translation. Over the last few decades, multiple efforts have been undertaken to investigate incorrect translations caused by the polysemous nature of words. Within this body of research, some studies have posited that models pick up semantic biases existing in the training data, thus producing translation errors. In this paper, we present DIBI\(\text{MT}\), the first entirely manually-curated evaluation benchmark which enables an extensive study of semantic biases in Machine Translation of nominal and verbal words in five different language combinations, namely, English and one or other of the following languages: Chinese, German, Italian, Russian and Spanish. Furthermore, we test state-of-the-art Machine Translation systems, both commercial and non-commercial ones, against our new test bed and provide a thorough statistical and linguistic analysis of the results. We release DIBI\(\text{MT}\) at https://nlp.uniroma1.it/dibimt as a closed benchmark with a public leaderboard.

1 Introduction

The polysemous nature of words poses a long-standing challenge in a wide range of Natural Language Processing (NLP) tasks such as Word Sense Disambiguation (Navigli, 2009; Bevilaqua et al., 2021) (WSD), Information Retrieval (Krovetz and Croft, 1992) (IR) and Machine Translation (Emelin et al., 2020) (MT).

In MT, some research works have addressed the ability of systems to disambiguate polysemous words. For instance, given the sentence He poured a shot of whiskey, the polysemous target word shot unequivocally means a small quantity and therefore a possible translation into Italian could be: Versò un goccio di whiskey. However, some MT systems propose the following translation: Versò uno sparo di whiskey in which the noun sparo means gun-shot. This is one of many examples that seem to encourage a deeper performance analysis in scenarios in which MT systems are required to deal with polysemous words and, specifically, with infrequent meanings of polysemous words. Although state-of-the-art MT systems, both commercial and non-commercial ones, achieve impressive BLEU scores on standard benchmarks, in our work we demonstrate that they still present significant limitations when dealing with infrequent word senses, which standard metrics fail to recognize.

In the last few decades, attempts have been made to investigate the aforementioned phenomena. In fact, recent studies have observed a direct correlation between semantic biases in the training data and semantic errors in translation. However, their findings are limited by the following shortcomings: i) they are not based on entirely manually-curated benchmarks; ii) they rely heavily on automatically-generated resources to determine the correctness of a translation; and iii) they do not cover multiple language combinations.

In this work, we address the aforementioned drawbacks and present DIBI\(\text{MT}\), to the best of our knowledge the first fully manually-curated evaluation benchmark aimed at investigating the impact of semantic biases in MT in five language combinations, covering both nouns and verbs. This benchmark allows the community not only to better explore the described phenomena, but also to devise innovative MT systems which better deal with lexical ambiguity. Specifically, the contributions of the present work are threefold:

- We present DIBI\(\text{MT}\), a novel gold-quality test bed for semantic biases in MT that goes beyond a simple accuracy score, covering five language combinations, namely English and one or other of the following languages: Chinese, German, Italian, Russian and Spanish;
He poured a shot of whiskey.

Figure 1: Example of an annotated dataset item. Target word is shot, in its meaning of a “small drink of liquor”. We expect translations to contain, for example in Italian, goccio (lit. a drop), but not, for example in Spanish, pistolero (a person who shoots).

- We define four novel metrics that better clarify the semantic biases within MT models;
- We provide a thorough statistical and linguistic analysis in which we compare 7 state-of-the-art MT systems, including both commercial and non-commercial ones, against our new benchmark. Furthermore, we extensively discuss the results.

To enable further research, we release DiB1MT as a closed benchmark with a public leaderboard at https://nlp.uniroma1.it/dibimt.

2 Related Work

Over the course of the last few decades, several approaches to the evaluation of the lexical choice in MT have been proposed. To this end, cross-lingual benchmarks were created in which systems were required to provide the translation or a substitute for a given target word in context in a target language (Vickrey et al., 2005; Mihalcea et al., 2010; Lefever and Hoste, 2013).

More recently, Gonzales et al. (2017) put forward ContraWSD, a dataset which includes 7,200 instances of lexical ambiguity for German → English, and 6,700 for German → French. This dataset pairs every reference translation with a set of contrastive examples which contain incorrect translations of a polysemous target word. For each instance, the answer provided by systems is considered correct if the reference translation is scored higher. Based on a denoised version of the ContraWSD dataset and focusing on the language combination German → English, Gonzales et al. (2018) present the Word Sense Disambiguation Test Suite which, unlike ContraWSD, evaluates MT output directly rather than by scoring translations. The suite consists of a collection of 3,249 sentence pairs in which the German source sentences contain one ambiguous target word. As target words, the authors considered only words in German whose translation into English does not cover multiple senses, thus making the evaluation more straightforward. Despite their effectiveness, such benchmarks do not allow systems to be tested in multiple language combinations, and only cover a very limited number of words and senses. To address these limitations, Raganato et al. (2019) proposed MuCoW, an automatically-created test suite covering 16 language pairs, with more than 200,000 sentence pairs derived from word-aligned parallel corpora.

Other research studies investigated the disambiguation capabilities of MT systems by exploring their internal representations (Marvin and Koehn, 2018; Michel et al., 2019), or improving them via context-aware word embeddings (Liu et al., 2018). More recently, Emelin et al. (2020) introduced a statistical method for the identification of disambiguation errors in neural MT (NMT) and demonstrated that models capture data biases within the training corpora, which leads these models to produce incorrect translations. Although the authors expected their approach to be transferable to other language combinations, they only focused on German → English.

Based on the findings and open research questions raised in the aforementioned works, the present paper aims at investigating not only the presence, but also, most importantly, the nature and properties of semantic biases in MT in multiple language combinations, via a novel entirely manually-curated benchmark called DiB1MT and a thorough performance analysis.

3 Building DiB1MT

The DiB1MT benchmark focuses on detecting Word Sense Disambiguation biases in NMT, i.e., biases of certain words towards some of their more frequent meanings. The creation of such a dataset requires i) a set of unambiguous and grammatically-correct sentences containing a polysemous target word; ii) a set of correct and incorrect translations of each target word into the languages to be covered. Figure 1 depicts an example of a dataset item.

3.1 Preliminaries

BabelNet Similarly to previous studies, we rely on BabelNet¹ (Navigli et al., 2021), a large multilingual...
gual encyclopedic dictionary whose nodes are concepts represented by synsets, i.e., sets of synonyms, containing lexicalizations in multiple languages and coming from various heterogeneous resources, including, inter alia, WordNet (Miller et al., 1990) and Wiktionary. \(^{2}\) Let us define $\mathbb{B}$ as an abstraction used to query the subset of synsets in BabelNet that contain at least one sense\(^3\) from WordNet and one or more senses in languages other than English,\(^4\) while only considering senses coming from high-quality sources, i.e., language-specific wordnets.

**Formal Notation** Given an arbitrary synset $\sigma$, we define $\lambda_L(\sigma)$ as the set of lexicalizations of $\sigma$ in language $L$ contained within $\mathbb{B}$. As an example, let us consider the synset $\tilde{\sigma}$ corresponding to the drink meaning of the word shot. $\tilde{\sigma}$ contains lexicalizations in different languages, including: Shot$_{DE}$, shot$_{EN}$, nip$_{ES}$, chupito$_{ES}$, trago$_{ES}$, bicchierino$_{IT}$ and goccio$_{IT}$. Hence, $\lambda_{EN}(\tilde{\sigma}) = \{\text{shot, nip}\}$, while $\lambda_{ES}(\tilde{\sigma}) = \{\text{chupito, trago}\}$.

Furthermore, let $\lambda_P$ represent a (lemma, part of speech) pair, where $P$ is the part of speech. We denote $\Omega_L(\lambda_P) = \{\sigma_1, \ldots, \sigma_n\}$ as the set of synsets which contain $\lambda_P$ as a lexicalization in language $L$ according to $\mathbb{B}$. Additionally, we define $\delta_L(\lambda_P) = |\Omega_L(\lambda_P)|$ as the polysemy degree, i.e., the number of senses, of $\lambda_P$ in language $L$. For example, given $\lambda_P = \text{shot}_N$, $\Omega_{EN}(\lambda_P)$ would be the set of synsets associated with the nominal term shot (e.g., the act of firing, a photograph and a drink, among others).

### 3.2 Sentence Selection Process

In this section, we detail the creation process of our dataset, i.e., the selection of our sentences as well as the construction and filtering of our items.

**Item Structure and Notation** Before we proceed, let us formally state how each item in the dataset is structured: given a source sentence $s = [w_1, \ldots, w_n]$ as a sequence of words, and a target word $w_i$ in $s$ tagged with some synset $\sigma$, we consider $X = (s, w_i, \sigma)$ as an initial item of the dataset, i.e., an instance composed of an English sentence $s$, a target word $w_i$ and its associated synset $\sigma$; this instance can be annotated for candidate translations of $w_i$ in some language $L$. We also denote $\lambda^L_{w_i}$ as the (lemma, POS) pair of $w_i$.

#### 3.2.1 Starting Sentence Pool

We collect our initial items from two main sources: WordNet and Wiktionary.\(^5\) Specifically, we use the examples from WordNet Tagged Glosses (Langone et al., 2004), where each sentence’s target word was manually associated with its synset, thereby readily providing the first batch of initial items.

As for Wiktionary, instead, we start by obtaining every usage example $s$ and its associated definition $d$ (filtering out archaic usages and slang), then, we automatically extract the target words from the corresponding example.\(^8\) Now, the only step that remains in order to construct an initial item is to associate a synset $\sigma$ with the word $w_i$ used in the example $s$. We perform this association in two phases: first, we try to map the definition $d$ related to the example $s$ to a BabelNet synset by relying on the automatic mappings available in BabelNet 5 between WordNet and Wiktionary, discarding examples for which this association can not be found; second, we manually validate and correct these successful associations to ensure that our initial items are of high quality.

#### 3.2.2 Sentence Filtering

We apply a filtering step to the original sentences in order to select examples that are likely to be more challenging for the models to translate: i) we discard every initial item $X$ for which $\delta_{EN}(\lambda^L_{w_i}) < 3$, i.e., we retain only sentences whose associated (lemma, POS) pair has a polysemy degree of at least 3 in $\mathbb{B}_{EN}$; ii) we retain at most only one sentence per sense per source; iii) differently from previous works, which impose a strict requirement on synsets that are monosemous in the target language, we retain sentences satisfying the following requirement. Let us consider the nominal senses of the word bank: among them, one represents a specific aviation maneuver. In Italian, this synset

\(^2\)https://www.wiktionary.org/
\(^3\)A “sense” is a lexicalization of a specific synset in some language. Henceforth, we will refer to lexicalizations and senses interchangeably.
\(^4\)Specifically, we consider synsets that have lexicalizations in English, Italian, German, Russian, Spanish and Chinese.
\(^5\)For simplicity, we use the term word here, but our work focuses on multi-word expressions as well (both in source and target sentences).

\(^6\)We use the dump of September 2021.
\(^7\)Which we convert from WordNet to BabelNet.
\(^8\)In Wiktionary, target words are marked in bold inside the example sentence.

The reasoning for this choice is twofold: on the one hand, oftentimes Wiktionary has multiple examples for the same synset, that differ in only one or two words, thus we skip them to avoid repetitions; on the other hand, we obtain an increase in sense coverage without worsening the annotator load.
includes one lexicalization, avvitamento; although this is not monosemous in Italian (e.g., avvitamento might also refer to a screw thread), neither of the other possible senses of avvitamento has bank as an English lexicalization, which, for Italian, satisfies our third condition. If the same holds true for all languages, the synset passes the test and thus the sentence is retained.

### 3.3 Annotating the Dataset

Once the set of initial items is ready, we can proceed with the annotation phase, which will produce our annotated items.

Specifically, given a language \( L \) and an initial item \( X = (s, w_i, \sigma) \), we associate a set of good (\( G_L \)) and bad (\( B_L \)) translation candidates with \( X \), which represent words that, respectively, we do, and do not, expect to see in a translation of sentence \( s \) in language \( L \). Finally, we refer to \( X_L \) as an annotated item, i.e., the tuple \((s, w_i, \sigma, G_L, B_L)\).

#### 3.3.1 Pre-annotation Item Creation

Before moving forward with the annotation phase, we pre-populate the sets of good (\( G_L \)) and bad (\( B_L \)) lexicalizations for a given initial item \( X \) in language \( L \) extracting them from \( \mathbb{B} \). Formally, we assign \( G_L = \Lambda_L(\sigma) \), i.e., the set of lemmas in language \( L \) of the BabelNet synset associated with \( \sigma \); furthermore, we set \( B_L = \bigcup_{\hat{\sigma} \in \Omega_L} \Lambda_L(\hat{\sigma}) \setminus \{\sigma\} \), i.e., the set of all lemmas in language \( L \) of BabelNet synsets associated with any \( \hat{\sigma} \) excluding \( \sigma \). With this step, we produce an automatically populated version of our annotated items.

#### 3.3.2 Annotation Guidelines

We instruct annotators to update the set of good (\( G_L \)) and bad (\( B_L \)) lexicalizations of \( w_i \) in \( s \) such that each lexicalization contained in the respective set can be considered a good or a bad translation equivalent for the target word in the provided sentential context.\(^{10}\)

We also instruct annotators to discard sentences in which i) the target word \( w_i \) is an idiomatic expression or a proper noun, and ii) the semantic context is not sufficient to properly disambiguate \( w_i \).

Given the expertise required to carry out this task, we rely on three highly qualified translators: one for Italian, German and Russian; one for Spanish and one for Chinese. Our annotators satisfy the following requirements: they are native speakers or hold C2-level certifications and work as professional translators in the given language combinations. The full instructions provided to the annotators can be found in Appendix C.

#### 3.3.3 Resulting Dataset

Our annotators analyzed around 800 sentences, discarding 200 of them, finally obtaining 600 annotated items in 5 languages. Due to a coverage issue of the Russian language in BabelNet, we retain only sentences tagged with nominal or verbal synsets. Dataset statistics are reported in Table 1.

As expected, we note that the lexicalizations found in \( \mathbb{B} \) have been substantially refined by our annotators in all languages, as reported in Table 2. Indeed, across languages, on average, 54% of the good lexicalizations have been added by our annotators, while 42% of the pre-existing lexicalizations have been removed. More importantly, given a language and two sentences containing words referring to the same synset, on average only in 55% of cases do they also share those words’ good lexicalizations, confirming that the assumption that

| All | Nouns | Verbs |
|-----|-------|-------|
| # items | 597 | 314 | 286 |
| # lemmas | 305 | 186 | 147 |
| # synsets | 471 | 254 | 217 |

Table 1: General statistics of our annotated dataset. POS-specific lemmas do not sum to “All” as they can overlap across POS tags (e.g., run).

|           | All (OG) | RG | SL |
|-----------|----------|----|----|
| DE        | 50.9     | 25.0 | 59.7 |
| ES        | 49.6     | 19.5 | 47.7 |
| IT        | 49.1     | 38.2 | 67.1 |
| RU        | 67.4     | 57.3 | 54.4 |
| ZH        | 55.2     | 69.0 | 46.3 |
| Mean      | 54.4     | 41.8 | 55.0 |

Table 2: Annotation Statistics: %OG represents the average percentage of Good lemmas that are Original, i.e., were added by our annotators; %RG represents the average percentage of Good lemmas that were Removed, i.e., lemmas that came from BabelNet and that our annotators deemed incorrect in the context of the given example; %SL represents the average percentage of times two senses Share the same set of Lexicalizations for two different example sentences.

\(^{10}\)Any lexicalization of \( \sigma \) in \( L \) that is removed from \( G_L \) is automatically placed in \( B_L \).
all synonyms of a word are valid replacements can lead to incorrect results.

These statistics lead us to a straightforward, but important, conclusion: only in a limited number of cases is a lexicalization belonging to a given synset to be considered as a suitable translation equivalent for the provided target word and its context. Examined jointly, these metrics suggest that relying on synset lexicalizations from BabelNet alone is prone to producing errors, either due to BabelNet’s intrinsic noise, or due to the lack of different granularity of synsets and contextualized words.

Sentences’ Properties Description As we stated in Section 3.2.1, the sentences we annotate are all usage examples of specific concepts obtained from WordNet or Wiktionary. Such examples are typically short main clauses with no subordinates, featuring on average 9 words (around 50 characters per sentence). All selected sentences include a semantic context which allows the meaning of the target word to be properly identified.

3.4 Analysis Procedure
DiBiMT’s analysis procedure is fairly simple: given an annotated item $X_L = (s, w_i, \sigma, G_L, B_L)$ and a translation model $\mathcal{M}$, we compute $t_L = \mathcal{M}_L(s)$, i.e., the translation of $s$ in language $L$ according to $\mathcal{M}$. Then, we use Stanza (Qi et al., 2020) to perform tokenization, part-of-speech tagging and lemmatization of $t_L$ and, finally, we check if there is any match between the lemmas of the translated sentence and those contained in $G_L$ or $B_L$. In case there is no match, we mark the translation as a MISS; otherwise, we mark it as GOOD or BAD depending on which set matched the lemma.

This produces an analyzed item, which for simplicity we denote as $X^M_L = (X_L, t_L, R, \omega_L)$, where $R$ is one of GOOD, BAD or MISS and $\omega_L$ represents the matched lemma in case there was a match (GOOD or BAD), $\epsilon$ otherwise.

4 Results and Discussion
We now: i) use DiBiMT to carry out an evaluation of 7 different machine translation systems; ii) report the obtained results, including a thorough statistical and linguistic evaluation; iii) extensively discuss our findings, providing multiple measures of semantic bias; and iv) offer some insights into the causes of such biases. In Appendix D we include a model-specific breakdown of the various scores and metrics reported throughout this section.

4.1 Comparison Systems
We test a wide range of models, both commercial and non-commercial ones, and report their performances on DiBiMT’s evaluation metrics:

- **DeepL Translator**12, a state-of-the-art commercial NMT system.
- **Google Translate**13, arguably the most popular commercial NMT system.
- **OPUS** (Tiedemann and Thottingal, 2020), the smallest state-of-the-art NMT model available to date, a base Transformer (each model has approximately 74M parameters) trained on a single language pair on large amounts of data.
- **MBart50** (Tang et al., 2021), multilingual BART fine-tuned on the translation task for 50 languages (610M parameters). We refer to MBart50 as the English-to-many model, and to MBart50_MTM as the many-to-many model.
- **M2M100** (Fan et al., 2021), a multilingual model able to translate from/to 100 languages. We test both versions of the model, the 418M parameter one (which we dub M2M100) and the 1.2B parameter one (dubbed M2M100_LG).

4.2 Discussion of MISS
Figure 2 reports general results of the analysis per (model, language) pair. Given the high percentage of analyzed items classified as MISS, we asked our annotators to perform an inspection on a random sample of 70 items per language in order to unearth the reasons, with varying results. We identified multiple causes, namely: i) word omission in the translation (around 19% of items, mostly in Chinese and Italian); ii) issues with Stanza’s tokenization (around 11%, mostly Chinese and Russian) and lemmatization (around 12%, mostly Italian and German); iii) words translated as themselves (approximately 5%, often in multilingual neural models); iv) translations which have nothing to

\[12\text{https://deepl.com/}
\[13\text{https://translate.google.com/}

Table 3: General results: accuracy on DiBiMT across models and languages. Higher is better.

|       | DeepL | Google | M2M100 | M2M100<sub>LG</sub> | MBart50 | MBart50<sub>MTM</sub> | OPUS  | Mean |
|-------|-------|--------|--------|----------------------|---------|------------------------|-------|------|
| DE    | 74.60 | 21.90  | 22.19  | 26.96                | 28.73   | 28.65                  | 27.99 | 33.00|
| ES    | 57.87 | 22.54  | 25.51  | 30.00                | 33.89   | 32.66                  | 36.66 | 34.16|
| IT    | 53.49 | 18.04  | 21.83  | 25.14                | 29.34   | 30.54                  | 29.76 | 29.76|
| RU    | 71.58 | 22.89  | 26.22  | 35.19                | 36.06   | 33.33                  | 41.07 | 38.05|
| ZH    | 46.00 | 15.04  | 16.99  | 22.35                | 31.21   | 34.15                  | 35.07 | 27.64|

Mean: 60.71 20.08 22.55 27.93 31.85 31.87 32.68 32.52

Table 4: Semantic Biases: SFII, i.e., Sense Frequency Index Influence, represents the average percentage of errors at varying levels of $\mu_{X_P}(\sigma)$. SPDI, i.e., Sense Polysemy Degree Importance, instead, represents the average percentage of errors at varying level of $\delta_L(\lambda_P)$. Lower is better.

|       | DeepL | Google | M2M100 | M2M100<sub>LG</sub> | MBart50 | MBart50<sub>MTM</sub> | OPUS  | Mean |
|-------|-------|--------|--------|----------------------|---------|------------------------|-------|------|
| SFII  | SFII  | SFII   | SFII   | SFII                 | SFII    | SFII                   | SFII  | SFII |
| DE    | 34.78 | 28.30  | 86.61  | 79.54                | 82.00   | 78.15                  | 76.15 | 75.89|
| ES    | 56.04 | 46.14  | 83.84  | 78.41                | 83.08   | 77.95                  | 73.84 | 76.27|
| IT    | 57.71 | 49.01  | 85.47  | 80.62                | 80.22   | 78.69                  | 76.10 | 77.25|
| RU    | 41.97 | 33.64  | 84.01  | 79.85                | 83.49   | 79.85                  | 74.72 | 71.06|
| ZH    | 64.97 | 59.58  | 91.97  | 87.98                | 91.81   | 88.79                  | 82.17 | 69.48|

Mean: 51.10 43.33 86.38 82.01 83.39 79.24 80.19 76.75

An example is the sentence *he is a crack shot*, where the word *shot* is translated by MBart50 into Italian as “schianto”, which can be interpreted in this case as “someone very good looking”.

4.3 General Results

Table 3 reports accuracy for non-MISS analyzed items (i.e., #GOOD/#GOOD+/#BAD). With the sole exception of DeepL, which greatly outperforms every other competitor, models achieve extremely low scores, in the range of 20%-33%. Surprisingly, Google Translate performs worst across languages.

4.4 Analyzing the Semantic Biases

In addition to accuracy, DiBiMT analyzes the semantic biases of a translation model via four novel metrics, which we define in detail in what follows.

**Sense Frequency Index Influence (SFII)** We study the sensitivity of models to disambiguating senses with respect to their frequency. To do this, we define $\mu_{X_P}(\sigma)$ as the index of synset $\sigma$ in $\Omega_{EN}(\lambda_P)$ ordered according to WordNet’s sense frequency, as computed from SemCor. That is, index $k$ means that synset $\sigma$ is the $k$-th most frequent meaning for $\lambda_P$.

In Figure 3(a), we plot the number and percentage of errors made on average by the models, grouping items by $\mu_{X_P}(\sigma)$. Figure 3(a) reports the results on all items. As expected, we observe that higher polysemy leads to more errors, confirming that models still struggle with very polysemous words. Similarly to SFII, SPDI is defined as the average percentage of
We try to analyze whether MT models translate the target word into a lexicalization belonging to the Most Frequent Sense associated with $\lambda_P$. MFS+, instead, checks whether the wrong translation belongs to any synset that is more frequent than the target one. Lower is better.

Then, we check the frequency of the surface form. We take the most frequent according to $\mu_{\lambda_{P}}^X$, and this needs to be further investigated in future research.

### 4.5 Are verbs harder than nouns?

The existing literature in WSD points to the fact that verbs are generally harder than nouns, mostly due to their highly polysemous nature (Barba et al., 2021b). We try to analyze whether MT models...
are affected by the same phenomenon: in Table 6, we report the average results obtained by running DiBiMT on all its sentences (column ALL) and the subset of sentences whose target word was either a NOUN or a VERB. In general, we observe an average drop of accuracy of 4 points, as well as an astounding difference of 18 percentage points in MISS handling, which we will investigate more thoroughly in future work. Interestingly, MT models are much more inclined to translate nouns into their most frequent sense; we attribute this difference to the generally higher polysemy of verbs compared to nouns, which increases the size of the space of possible translations for a given verb, thus decreasing the chance that it gets translated into the MFS. Aside from this, we draw the same conclusion as that drawn by previous works in the field of WSD, with nouns being generally easier to translate than verbs.

### 4.6 Is the encoder disambiguating?

We try to assess to what extent, in a multilingual encoder-decoder architecture, the encoder is determining the implicit disambiguation of the source sentence before generating the translation. For instance, we ask ourselves this question: given an ambiguous word \( w_i \) in the source sentence \( s \), how often does the model translate it into a lexicalization representing the same sense, if prompted to translate \( s \) into different languages? Intuitively, if the encoder was the sole contributor to the implicit disambiguation performed by the model, we would expect to see the meaning to always be the same, regardless of the target language.

To measure this, we perform the following experiment: given a model \( \mathcal{M} \),\(^{16}\) two languages \( L_1 \) and \( L_2 \) and an initial item \( X \), we take \( \mathcal{M} \)’s analyzed items \( X_{L_1}^M \) and \( X_{L_2}^M \)\(^{17}\) and check if translations in \( L_1 \) and \( L_2 \) have a synset in common, i.e., \( |\Omega_{L_1}(\omega_{L_1}) \cap \Omega_{L_2}(\omega_{L_2})| > 0 \). The results of this experiment are reported in Figure 4.

We observe that, on average, this phenomenon occurs around 70% of the time. Hence, it is safe to assume that, while the encoder certainly plays an important role in the disambiguation of the input sentence, the decoder is also contributing significantly. Another interesting observation is that the alphabet of the target language does not seem languages. We also disregard DeepL and Google Translate as their architecture is proprietary.

\(^{16}\)We disregard OPUS here as it is a set of bilingual models, rather than a single model capable of translating into multiple languages.

\(^{17}\)We skip item \( X \) if either \( X_{L_1}^M \) or \( X_{L_2}^M \) is a MISS.
to have any influence, as language pairs involving Russian display scores that are very similar to those of the other three European languages. We attribute lower scores in Chinese to coverage issues in BabelNet, which would hinder a correct fulfillment of the condition defined for this experiment.

4.7 How challenging is DiBiMT?

Given the low performances achieved by MT models, we test a WSD system on the English sentences within DiBiMT, both to assess the toughness of our system and to establish an additional baseline. We use ESCHER (Barba et al., 2021a), a state-of-the-art model on English WSD. Interestingly, ESCHER achieves an overall accuracy score of 66.33, almost 15 points lower than the results on the standard WSD benchmark (80.7 on ALL, Raganato et al., 2017), therefore confirming the challenging nature of DiBiMT. Furthermore, in order to estimate the difference in disambiguation capability between NMT models and a dedicated WSD system, we compute ESCHER’s performances on the set of English sentences of non-MISS analyzed items for each (model, language) pair. We report these results in Table 7, whose accuracy scores can be directly compared to those in Table 3.

As expected, the average MT accuracy is significantly lower than ESCHER’s, with the sole exception of DeepL, which manages to surpass it on German and Russian. These results clearly demonstrate that current NMT models are still not on par with dedicated WSD systems, and thus that they might benefit from the inclusion of such WSD systems within the NMT ecosystem.

4.8 Is this a decoding issue?

As a final experiment, we assess whether the semantic biases are caused by search errors (i.e., failures of the decoding algorithm), or model errors (i.e., the models deemed their translations the best possible). For each (model $M$, language $L$) pair, we sample a BAD translation ($t_{BAD}$), pair it with a GOOD translation ($t_{GOOD}$) produced by another model (prioritizing DeepL), and ask annotators to check their correctness and apply corrections where needed. Then compute the perplexities according to $M$ with the corresponding English sentence $s$, i.e., $p_{GOOD} = p_M(t_{GOOD}|s)$ and $p_{BAD} = p_M(t_{BAD}|s)$. We repeat this sampling 50 times per ($M$, $L$) pair and check how often $p_{BAD} > p_{GOOD}$. Table 8 shows that, on average, this happens in 93% of cases, thus confirming that most semantic biases are embedded within models and are not caused by the decoding strategy.

5 Conclusions

In this work, we presented DiBiMT, a novel benchmark for measuring and understanding semantic biases in NMT, which goes beyond simple accuracy and provides novel metrics that summarize how biased NMT models are. We tested DiBiMT on 7 widely adopted NMT systems, extensively discussing their performances and providing novel insights into the possible causes and relations of semantic biases within NMT models.

Furthermore, statistics of our annotations suggest that, when dealing with translations, synsets’ lexicalizations cannot be used interchangeably, as their choice depends heavily on the context.

In the future, we plan to improve DiBiMT by introducing better heuristics to recognize and handle MISS cases, especially covering the linguistic phenomena we described (see Section 4.2); we also aim at widening language coverage and increasing the number of sentences in the benchmark, consequently improving word and sense coverage. To enable further research, we release DiBiMT as a closed benchmark with a public leaderboard at: https://nlp.uniroma1.it/dibimt.

---

18The publicly available version trained on SemCor data only.
Acknowledgements

The authors gratefully acknowledge the support of the ELEXIS project No. 731015 under the European Union’s Horizon 2020 research and innovation programme and the PerLIR project (Personal Linguistic resources in Information Retrieval) funded by the MIUR Progetti di ricerca di Rilevante Interesse Nazionale programme (PRIN 2017). This work was also partially supported by the MIUR under the grant “Dipartimenti di eccellenza 2018-2022” of the Department of Computer Science of Sapienza University.

References

Edoardo Barba, Tommaso Pasini, and Roberto Navigli. 2021a. Esc: Redesigning WSD with extractive sense comprehension. In Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 4661–4672.

Edoardo Barba, Luigi Procopio, and Roberto Navigli. 2021b. ConSeC: Word Sense Disambiguation as continuous sense comprehension. In Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pages 1492–1503, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Michele Bevilacqua, Tommaso Pasini, Alessandro Raganato, Roberto Navigli, et al. 2021. Recent trends in word sense disambiguation: A survey. In Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21. International Joint Conference on Artificial Intelligence, Inc.

Denis Emelin, Ivan Titov, and Rico Sennrich. 2020. Detecting word sense disambiguation biases in machine translation for model-agnostic adversarial attacks. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 7635–7653, Online. Association for Computational Linguistics.

Angela Fan, Shruti Bhosale, Holger Schwenk, Zhiyi Ma, Ahmed El-Kishky, Siddharth Goyal, Mandeep Baines, Onur Celebi, Guillaume Wenzek, Vishrav Chaudhary, et al. 2021. Beyond English-Centric Multilingual machine translation. Journal of Machine Learning Research, 22(107):1–48.

Annette Rios Gonzales, Laura Mascarell, and Rico Sennrich. 2017. Improving Word Sense Disambiguation in Neural Machine Translation with Sense Embeddings. In Proceedings of the Second Conference on Machine Translation, pages 11–19.

Annette Rios Gonzales, Mathias Müller, and Rico Sennrich. 2018. The Word Sense Disambiguation Test Suite at WMT18. In Proceedings of the Third Conference on Machine Translation: Shared Task Papers, pages 588–596.

Robert Krovetz and W Bruce Croft. 1992. Lexical ambiguity and information retrieval. ACM Transactions on Information Systems (TOIS), 10(2):115–141.

Helen Langone, Benjamin R Haskell, and George A Miller. 2004. Annotating WordNet. In Proceedings of the Workshop Frontiers in Corpus Annotation at HLT-NAACL 2004, pages 63–69.

Els Lefever and Véronique Hoste. 2013. Semeval-2013 task 10: Cross-lingual 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 158–166.

Frederick Liu, Han Lu, and Graham Neubig. 2018. Handling Homographs in Neural Machine Translation. In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), pages 1336–1345, New Orleans, Louisiana. Association for Computational Linguistics.

Rebecca Marvin and Philipp Koehn. 2018. Exploring Word Sense Disambiguation Abilities of Neural Machine Translation Systems. In Proceedings of the 13th Conference of the Association for Machine Translation in the Americas (Volume 1: Research Track), pages 125–131.

Paul Michel, Xian Li, Graham Neubig, and Juan Pino. 2019. On Evaluation of Adversarial Perturbations for Sequence-to-Sequence Models. 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), pages 3103–3114, Minneapolis, Minnesota. Association for Computational Linguistics.

Rada Mihalcea, Ravi Sinha, and Diana McCarthy. 2010. Semeval-2010 task 2: Cross-Lingual Lexical Substitution. In Proceedings of the 5th international workshop on semantic evaluation, pages 9–14.

George A Miller, Richard Beckwith, Christiane Fellbaum, Derek Gross, and Katherine J Miller. 1990. Introduction to wordnet: An on-line lexical database. International journal of lexicography, 3(4):235–244.

Roberto Navigli. 2009. Word Sense Disambiguation: A Survey. ACM computing surveys (CSUR), 41(2):1–69.

Roberto Navigli, Michele Bevilacqua, Simone Conia, Dario Montagnini, and Francesco Cocco. 2021. Ten Years of Babelnet: A Survey. In Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21, pages 4559–4567.
Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D. Manning. 2020. Stanza: A Python Natural Language Processing Toolkit for Many Human Languages. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations.

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. Association for Computational Linguistics.

Alessandro Raganato, Yves Scherrer, and Jörg Tiedemann. 2019. The MuCoW test suite at WMT 2019: Automatically harvested multilingual contrastive word sense disambiguation test sets for machine translation. In Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1), pages 470–480.

Yuqing Tang, Chau Tran, Xian Li, Peng-Jen Chen, Naman Goyal, Vishrav Chaudhary, Jiatao Gu, and Angela Fan. 2021. Multilingual Translation from Denoising Pre-Training. In Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021, pages 3450–3466, Online. Association for Computational Linguistics.

Jörg Tiedemann and Santhosh Thottingal. 2020. OPUS-MT — Building open translation services for the World. In Proceedings of the 22nd Annual Conference of the European Association for Machine Translation (EAMT), Lisbon, Portugal.

David Vickrey, Luke Biewald, Marc Teyssier, and Daphne Koller. 2005. Word-Sense Disambiguation for Machine Translation. In Proceedings of human language technology conference and conference on empirical methods in natural language processing, pages 771–778.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtoicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. Transformers: State-of-the-Art Natural Language Processing. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, pages 38–45, Online. Association for Computational Linguistics.
A Analysis Procedure Details

Our analysis procedure, which we described in Section 3.4, involves steps that go beyond simple lemma matching. For instance, in case of multi-word expressions, we allowed annotators to specify a wildcard, i.e., any number of tokens (including zero) were allowed to expand and still trigger a match. Additionally, since Stanza has multi-word expansion tokenization for some of the languages in our list, when available, we try to perform matching on both the list of words (alongside the list of tokens) in the translated sentence. Finally, in case no match is produced by the aforementioned steps, we apply a surface-level string matching heuristic which, especially in Chinese, helps us increase coverage.

B Neural Models Implementation

We use HuggingFace’s Transformers library (Wolf et al., 2020) for all neural models. As per standard practice, we generate translations using beam search as decoding algorithm with beam size 5.

C Instructions for Dataset Annotation

In this work, we investigate semantic biases in Machine Translation across languages. You are provided with a spreadsheet containing 300 instances, each including the following information: a lemma, its part of speech, a definition and some good and bad translation candidates derived from BabelNet. Your task is to manually verify the correctness of the good candidates and add new good candidates if deemed necessary. Furthermore, you are asked to verify that all bad candidates are wrong.

From a translation perspective, a good candidate is a word which correctly translates the English target word in the given context. Instead, a bad candidate is a wrong translation of the English target word in the given context.

Please adopt the following guidelines while annotating:

- Do not annotate idioms.
- Do not annotate instances in which the semantic context does not allow us to unequivocally determine the meaning of the target word.
- Do not annotate proper names, e.g., “Run” in the sentence The military campaign near that creek was known as “The battle of Bull Run”.
- You are allowed to include cross-PoS candidates (that is, candidates whose PoS is different from that of the target word), in this case please include the candidate in square brackets like this: [candidate_with_different_pos|Px], where x represents the part-of-speech tag of the translated word. Do this for multi-word expressions as well.

Mark with the tag “DISCUSS” difficult instances which you would like to discuss.

D Model-specific Analyses

We include model-specific analyses with per-language breakdown of the scores achieved on our benchmark. The column named ESCHER provides the scores of the WSD system on the subset of sentences of the specified model and language, and should be treated as an additional baseline to compare with the accuracy achieved by the system. Details can be found in Section 4.

- DeepL
- Google
- OPUS
- M2M100
- M2M100\_LG
- MBart50
- MBart50\_MTM
DeepL

Back to Model-specific Analyses.

|     | %MISS | Accuracy | MFS   | MFS+ | SPDI | SFII | ESCHER |
|-----|-------|----------|-------|------|------|------|--------|
| DE  | 36.07 | 74.60    | 53.68 | 84.21| 28.30| 34.78| 66.86  |
| ES  | 25.26 | 57.87    | 59.89 | 87.91| 46.14| 56.04| 67.89  |
| IT  | 20.21 | 53.49    | 68.08 | 86.38| 49.01| 57.71| 66.67  |
| RU  | 35.04 | 71.58    | 50.00 | 83.33| 33.64| 41.97| 66.76  |
| ZH  | 31.86 | 46.00    | 49.07 | 88.89| 59.58| 64.97| 68.42  |
| Mean| 29.69 | 60.71    | 56.14 | 86.15| 43.33| 51.10| 67.32  |

Figure 5: Evaluation on DeepL

(a) Sense Frequency Index

(b) Sense Polysemy Degree
Google

Back to Model-specific Analyses.

| %MISS | Accuracy | MFS  | MFS+ | SPDI | SFII | ESCHER |
|-------|----------|------|------|------|------|---------|
| DE    | 35.87    | 21.90| 56.76| 86.82| 79.54| 86.61   |
| ES    | 23.29    | 22.54| 61.96| 89.05| 78.41| 83.84   |
| IT    | 23.12    | 18.04| 61.96| 87.23| 80.62| 85.47   |
| RU    | 35.37    | 22.89| 48.12| 83.28| 83.49| 84.01   |
| ZH    | 32.49    | 15.04| 56.05| 88.20| 87.98| 91.97   |
| Mean  | 30.03    | 20.08| 56.97| 86.92| 82.01| 86.38   |

Figure 6: Evaluation on Google

(a) Sense Frequency Index

(b) Sense Polysemy Degree
OPUS

Back to Model-specific Analyses.

|     | %MISS | Accuracy | MFS  | MFS+ | SPDI | SFII | ESCHER |
|-----|-------|----------|------|------|------|------|---------|
| DE  | 37.84 | 27.99    | 56.98| 87.92| 76.25| 79.85| 66.95   |
| ES  | 25.69 | 36.66    | 64.47| 91.21| 69.12| 74.85| 66.83   |
| IT  | 29.11 | 29.95    | 64.48| 89.66| 72.02| 80.59| 65.82   |
| RU  | 45.84 | 41.07    | 48.40| 84.04| 69.27| 68.49| 69.21   |
| ZH  | 38.31 | 27.75    | 51.71| 87.45| 75.66| 79.96| 69.88   |
| Mean| 35.36 | 32.68    | 57.21| 88.06| 72.46| 76.75| 67.74   |

Figure 7: Evaluation on OPUS

(a) Sense Frequency Index

(b) Sense Polysemy Degree
M2M100

Back to Model-specific Analyses.

|        | %MISS | Accuracy | MFS   | MFS+  | SPDI  | SFII  | ESCHER |
|--------|-------|----------|-------|-------|-------|-------|--------|
| DE     | 49.41 | 22.19    | 61.28 | 87.23 | 76.15 | 82.00 | 65.85  |
| ES     | 41.91 | 25.51    | 61.81 | 89.37 | 77.95 | 83.08 | 66.77  |
| IT     | 42.44 | 21.83    | 60.75 | 86.79 | 76.58 | 80.22 | 66.35  |
| RU     | 51.77 | 26.22    | 47.87 | 83.41 | 78.34 | 79.85 | 66.42  |
| ZH     | 48.66 | 16.99    | 59.06 | 91.34 | 87.18 | 91.81 | 69.26  |
| Mean   | 46.84 | 22.55    | 58.15 | 87.63 | 79.24 | 83.39 | 66.93  |

Figure 8: Evaluation on M2M100

(a) Sense Frequency Index

(b) Sense Polysemy Degree

Figure 9: Overall Language Cooccurrence Heatmap for M2M100
M2M100\textsubscript{LG}

Back to Model-specific Analyses.

\begin{table}[h]
\centering
\begin{tabular}{|c|c|c|c|c|c|c|c|}
\hline
 & \%MISS & Accuracy & MFS & MFS+ & SPDI & SFII & ESCHER \\
\hline
DE & 42.02 & 26.96 & 59.13 & 87.30 & 74.71 & 78.90 & 67.18 \\
ES & 36.75 & 30.00 & 61.78 & 88.03 & 73.84 & 79.87 & 66.86 \\
IT & 37.07 & 25.14 & 62.82 & 88.81 & 76.10 & 78.69 & 68.50 \\
RU & 42.50 & 35.19 & 45.25 & 84.16 & 69.69 & 74.72 & 67.69 \\
ZH & 39.73 & 22.35 & 59.35 & 92.45 & 82.17 & 88.79 & 69.82 \\
\hline
Mean & 39.61 & 27.93 & 57.66 & 88.15 & 75.30 & 80.19 & 68.01 \\
\hline
\end{tabular}
\caption{Evaluation on M2M100\textsubscript{LG}}
\end{table}

Figure 10: Evaluation on M2M100\textsubscript{LG}

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{fig10}
\caption{(a) Sense Frequency Index}
\end{figure}

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{fig11}
\caption{(b) Sense Polysemy Degree}
\end{figure}

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{fig11}
\caption{Overall Language Cooccurrence Heatmap for M2M100\textsubscript{LG}}
\end{figure}
MBart50

Back to Model-specific Analyses.

|     | %MISS | Accuracy | MFS | MFS+ | SPDI | SFII | ESCHER |
|-----|-------|----------|-----|------|------|------|---------|
| DE  | 40.24 | 28.73    | 58.89 | 89.72 | 73.86 | 84.10 | 66.87   |
| ES  | 39.29 | 33.89    | 60.17 | 91.10 | 71.06 | 77.13 | 65.37   |
| IT  | 40.10 | 29.34    | 62.90 | 87.50 | 71.51 | 78.67 | 64.33   |
| RU  | 44.16 | 36.06    | 47.39 | 87.20 | 70.11 | 73.86 | 66.35   |
| ZH  | 44.35 | 31.21    | 50.66 | 89.87 | 73.14 | 80.39 | 68.93   |
| Mean| 41.63 | 31.85    | 56.00 | 89.08 | 71.94 | 78.83 | 66.37   |

Figure 12: Evaluation on MBart50

(a) Sense Frequency Index
(b) Sense Polysemy Degree

Figure 13: Overall Language Cooccurrence Heatmap for MBart50
MBart50<sub>MTM</sub>

Back to Model-specific Analyses.

|     | %MISS | Accuracy | MFS  | MFS+ | SPDI | SFII | ESCHER |
|-----|-------|----------|------|------|------|------|--------|
| DE  | 41.25 | 28.65    | 55.82| 89.56| 74.24| 84.95| 67.77  |
| ES  | 41.06 | 32.66    | 63.09| 91.85| 71.57| 79.06| 67.18  |
| IT  | 43.29 | 30.54    | 68.97| 91.81| 69.48| 79.41| 65.81  |
| RU  | 45.18 | 33.33    | 44.91| 87.96| 72.87| 78.58| 64.29  |
| ZH  | 44.59 | 34.15    | 54.17| 90.28| 71.50| 76.59| 69.58  |
| Mean| 43.07 | 31.87    | 57.39| 90.29| 71.93| 79.72| 66.93  |

Figure 14: Evaluation on MBart50<sub>MTM</sub>

![Graph](image1.png)

File 15: Overall Language Cooccurrence Heatmap for MBart50<sub>MTM</sub>

![Graph](image2.png)
Figure 16: Full page version of Figure 2.
He poured a **shot** of whiskey.

| German | ✓ | Schlückchen | ✗ | Schlag |
|        |   | Schuss      |   | Injektion |
| Spanish | ✓ | Trago | ✗ | Pistolero |
|        |   | Chupito    |   | Tiro   |
| Italian | ✓ | Goccio | ✗ | Iniezione |
|        |   | Bicchierino |   | Sparo |
| Russian | ✓ | шот | ✗ | стрелок |
|        |   | рюмка |   | выстрел |
| Chinese | ✓ | 杯 | ✗ | 枪手 |
|        |   | 小杯 |   | 本垒打 |

Table 1: Example of item annotated in all languages. First row is the example, target word is in bold, second row is the definition of the synset associated with the word in the example.

They tracked him back toward the **head** of the stream.

| German | ✓ | Flussursprung | ✗ | Kopf |
|        |   | Quelle |   | Kommando |
| Spanish | ✓ | Fuente | ✗ | Cabeza |
|        |   | Manantial |   | Jefe |
| Italian | ✓ | Fonte | ✗ | Testa |
|        |   | Sorgente |   | Capo |
| Russian | ✓ | исток | ✗ | проход |
|        |   | вопрос |   |  |
| Chinese | ✓ | 源头 | ✗ | 头 |
|        |   | 族长 |   |  |

Table 2: Example of item annotated in all languages. First row is the example, target word is in bold, second row is the definition of the synset associated with the word in the example.
If you **take off** for Thanksgiving you must work Christmas and vice versa.

To absent oneself from work or other responsibility, especially with permission.

| Language | ✓/✗ | Sich eine Auszeit nehmen | Sich freinehmen | Losgehen | Starten |
|---------|------|--------------------------|-----------------|----------|---------|
| German  | ✓    | ✓                        | ✗               |          |         |
| Spanish | ✓    | ✓                        | ✗               | Salir    | Llevar  |
| Italian | ✓    | ✓                        | ✗               | Togliersi| Decollare|
| Russian | ✓    | ✓                        | ✗               | вычесть | убить   |
| Chinese | ✓    | ✓                        | ✗               | 离开     | 减      |

Table 3: Example of item annotated in all languages. First row is the example, target word is in bold, second row is the definition of the synset associated with the word in the example.