GUIDE: Creating Semantic Domain Dictionaries for Low-Resource Languages

Jonathan Janetzki[a], **Gerard de Melo**[a],
Joshua Nemecek[b], and **Daniel Whitenack**[c*]
[a]Hasso Plattner Institute / University of Potsdam
[b]SIL International, [c]Prediction Guard
jonathan.janetzki@student.hpi.de, gerard.demelo@hpi.de,
joshua_nemecek@sil.org, dan@predictionguard.com

Abstract

Over 7,000 of the world's 7,168 living languages are still low-resourced. This paper aims to narrow the language documentation gap by creating multiparallel dictionaries, clustered by SIL's semantic domains. This task is new for machine learning and has previously been done manually by native speakers. We propose GUIDE, a language-agnostic tool that uses a GNN to create and populate semantic domain dictionaries, using seed dictionaries and Bible translations as a parallel text corpus. Our work sets a new benchmark, achieving an exemplary average precision of 60% in eight zero-shot evaluation languages and predicting an average of 2,400 dictionary entries. We share the code, model, multilingual evaluation data, and new dictionaries with the research community.[1]

1 Introduction

There are 7,168 languages spoken on Earth according to the Ethnologue (Eberhard et al., 2023). Creating dictionaries is the first step toward documenting languages, and it is also one of the most effective ways to preserve languages and cultures (Abah et al., 2018). A successful approach to creating dictionaries for low-resource languages is *rapid word collection* (Boerger, 2017): A team of linguists travels to spend 2–3 weeks with around 60 indigenous people and guides them through a questionnaire. Each question pertains to a particular *semantic domain* (Moe, 2010) that groups words with related meanings. In the following, we call this membership word-*Semantic Domain Question* (SDQ) link. Since this manual collection process involves traveling, it is expensive and sometimes even impossible (e.g., due to the risk of spreading diseases). This work investigates to what extent automated solutions can be an alternative means to procure such dictionary information at a greater speed and lower cost.

(1) What are the parts of a bird?

- ***èfuwu***, ***èkoa***, ***àwàdawo***, ***nusuɖùtɔ***,
  (feathers, gizzard, wings, greedy)

  xèvia, ***àzì***, ***àwàda***,
  (bird, egg, wing)

Figure 1: New dictionary entries: GUIDE linked seven words (bold) in the low-resource language Mina-Gen to an SDQ (top). Five are correct (blue), and two are incorrect (orange and underlined; labeled by a Mina-Gen speaker). Words in parentheses are translations.

The key idea of this paper is to automatically create semantic domain dictionaries for low-resource languages and fill in missing entries using a multilingual parallel text corpus of Bible translations, along with existing semantic domain dictionaries.

Our paper makes the following contributions:

- **Dictionary creation.** We propose the language-agnostic tool *Graph-based Unified Indigenous Dictionary Engine* (GUIDE), which links words in 20 languages and seven language families to their SDQs. It achieves state-of-the-art performance and has an average precision of 65% (see Figure 1). To the best of our knowledge, we propose the first automated approach to address this task.
- **Language flexibility.** To build a dictionary for a language, GUIDE requires only a Bible translation in that language, which is accessible in a verse-aligned format for at least 833 (Åkerman et al., 2023) languages. GUIDE can also be adapted to build dictionaries from any other parallel text.
- **Richer dictionaries.** We have predicted 32,000 new word-SDQ links for twelve languages with existing dictionaries that can en-

[*]Work done while the author was at SIL International.
[1]Repository: https://github.com/janetzki/GUIDE

Proceedings of the 6th Workshop on Research in Computational Linguistic Typology and Multilingual NLP (SIGTYP 2024), pages 10–24
March 22, 2024

rich *FieldWorks Language Explorer* (FLEx)[2] (if verified by a native speaker). Three of these languages are low-resourced.

- **New dictionaries.** We have predicted 19,000 word-SDQ links for eight languages with little to no pre-existing dictionary entries (see Figure 1). While these require further validation by a native speaker, they can be a useful resource, given that seven of the languages are low-resourced.

2 Background

Before describing the task in more detail, we introduce key resources and terms.

2.1 SIL's Semantic Domains

SIL's semantic domains (Moe, 2010) are a language-agnostic, standardized taxonomy to create dictionaries that mirror arbitrary aspects of the world, arranging words in 1,783 semantic domains, which are in turn divided into one or more SDQs. For each SDQ, the dictionaries list corresponding words. Semantic domains, SDQs, and words form a tree-structured graph[3], shown in Appendix A.

Each semantic domain consists of an *identifier* (ID) (e.g., "*5.6.2*"), a name (e.g., "*Bathe*"), a short description, one or more SDQs, and a list of entries (matching words or phrases) for each SDQ. In the following, we use the notation "*5.6.2-4*" as SDQ ID for the 4$^{\text{th}}$ question of semantic domain 5.6.2.

2.2 Defining "Low-Resource Language"

We follow the NLLB Team's (2022) definition of "low-resource languages", assuming that every language that is not listed as one of the 53 high-resource languages in their FLORES-200 dataset (Goyal et al., 2022; Guzmán et al., 2019) is a low-resource language.

3 Related Work

Existing approaches to creating dictionaries address different sets of languages and map words to ontologies or words of other languages.

The *Universal Wordnet* (UWN) is a graph-structured knowledge base for more than 200 languages that de Melo and Weikum (2009) automatically generated. They used several data sources, especially existing bilingual dictionaries and to a limited extent also parallel corpora. As a scaffold, they used *Princeton WordNet* (Fellbaum, 2000), which provides a semantic hierarchy of English terms, and they enriched it with more than 1.5 million new semantic links for more than 800,000 words. Our work has a similar goal, as we investigate how to create and enrich another linguistic resource automatically. We use the semantic domains as a scaffold and focus on low-resource languages, for which we assume only a small amount of parallel text.

Alnajjar et al. (2022) show how to find new translations of words in three endangered Uralic languages. Their key idea is to construct a graph of words in these and other languages, with known translations as edges. An advantage of their approach is that it does not require parallel texts or word alignment. By predicting missing links in this graph, they built new bilingual dictionaries that help preserve these endangered languages. Similarly, we build a graph in which words in different languages are separate nodes. But there are two important differences:

1. GUIDE also builds dictionaries for languages without labeled data.

2. We group words by their SDQs instead of predicting word-to-word translations. This approach allows us to build highly *multiparallel* dictionaries because words often have no 1:1 translations across languages but have different semantic ranges. "*Multiparallel*" means that the dictionaries are not mono- or bilingual but follow the same structure in all languages. SDQs provide for this flexibility.

Based on the reviewed related work on dictionary creation, we can summarize the research gap as follows: There is a need to create highly multiparallel dictionaries for low-resource languages without labeled data. We address this gap by using existing parallel text.

4 Dataset

We next describe the source of our parallel text, the 20 languages that we selected for our dataset, and its size.

[2] FLEx is a tool to document and analyze languages and the most common tool used with Moe's (Moe, 2010) semantic domains. The FieldWorks page provides a more detailed description of FLEx: https://software.sil.org/fieldworks/flex (visited on 2023-10-16).

[3] The entire hierarchy of semantic domains can be explored at the official page: https://semdom.org/v4/1 (visited on 2023-10-09).

	Language information				Bible translations		Dicts.
Language	ISO	# Speakers	Language family	Res.	Sample	# V.	# Entries
Development							
Bengali	ben	273M	Indo-European	High	আলো হোক (*āelā ehāka*)	31k	0.91k
Chinese (simplified)	cmn	1.14B	Sino-Tiebetan	High	要有光 (*yào yǒu guāng*)	31k	24k
English	eng	1.46B	Indo-European	High	Let there be light	37k	26k
French	fra	310M	Indo-European	High	Que la lumière soit	37k	30k
Hindi	hin	610M	Indo-European	High	उजियाला हो (*ujiyālā ho*)	31k	22k
Indonesian	ind	199M	Austronesian	High	Jadilah terang	11k	11k
Kupang Malay	mkn	350k	Creole (Malay-based)	Low	Musti ada taráng	9.8k	0.33k
Malayalam	mal	37.4M	Dravidian	Low	പ്രകാശം ഉണ്ടാകട്ടെ (*prakāśa uṇṭākaṭṭe*)	31k	25k
Nepali	npi	25.6M	Indo-European	Low	उज्यालो होस् (*ujyālo hos*)	31k	14k
Portuguese	por	260M	Indo-European	High	Que haja luz	31k	21k
Spanish	spa	559M	Indo-European	High	Sea la luz	37k	29k
Swahili	swh	71.6M	Niger-Congo	High	na kuwe nuru	31k	5.2k
Evaluation (zero-shot)							
German	deu	133M	Indo-European	High	Es werde Licht	31k	0
Hiri Motu	hmo	95.0k	Austronesian	Low	Diari ia vara namo	31k	0
Igbo	ibo	30.9M	Niger-Congo	Low	Ka ìhè dị	31k	0
Mina-Gen	gej	620k	Niger-Congo	Low	Kɛ̃klɛ̃ ne va e mè	35k	0
Motu	meu	39.0k	Austronesian	Low	Diari aine vara	31k	0
South Azerbaijani	azb	14.9M	Turkic	Low	Qoy işıq olsun	31k	0
Tok Pisin	tpi	4.13M	Creole (English-based)	Low	Lait i mas kamap	36k	0
Yoruba	yor	45.9M	Niger-Congo	Low	Jẹ́ kí ìmọ́lẹ̀ kí ó wà	31k	0

Table 1: Language information and dataset size: Language name, ISO 639 code (Eberhard et al., 2023), Number of speakers (Eberhard et al., 2023), Language family (Eberhard et al., 2023), and "resourcefulness" for the 20 languages in our dataset (defined in subsection 2.2). "Dicts." means "Semantic domain dictionaries" and "V." means "Verses". The matched number of words refers to the number of dictionary entries that also appear as words in the respective Bible translation. All samples have the same meaning. Text in parentheses shows transliterations of non-Latin scripts. Appendix B lists the Bible translations' source URLs.

4.1 The eBible Corpus

Åkerman et al. (2023) compiled the eBible corpus, which covers 833 languages from 75 language families, including languages that are considered extremely low-resourced. Each Bible translation in the eBible corpus is a text file with one line per verse (i.e., the corpus is verse-aligned).

Our dataset covers 20 languages in total: twelve development (i.e., training) languages and eight zero-shot evaluation languages. The difference between the two is that our dataset also contains semantic domain dictionaries for the development languages, which serve as labels, while there are no labels for the evaluation dataset.

4.2 Selected Languages

Table 1 displays the twelve languages that we use to train our model and the eight zero-shot evaluation languages that we use for testing. We chose languages based on the availability of data, the availability of language speakers for evaluation, and the language family (seeking to cover a broad spectrum).

4.3 Dataset Size

Table 1 further shows the size of our dataset for each of these languages, measured in terms of the number of verses in the Bible translations as a parallel text corpus and the number of semantic domains, which serve as labels. FLEx[4] provides the semantic domain dictionaries.

[4]A list of languages with existing semantic domain dictionaries is on this FieldWorks page: https://software.sil.org/fieldworks/download/localizations/ (visited on 2023-10-16).

5 Dictionary Creation with GUIDE

We now describe the GUIDE technique to induce dictionary entries for semantic domains based on a graph neural network.

5.1 Graph Induction

We transform our dataset into a graph, in which each node is a word in one of the 20 languages. The unique key of each node is its language code and the word itself (e.g., "*eng: grandchild*"). We hence use the term "*node*" as a synonym for "*word*" because each word becomes a node in the *Multilingual Alignment Graph* (MAG) (ImaniGooghari et al., 2022) that we build. The edges are the alignments between these words. We first create a *raw MAG*, which uses absolute word alignment counts from the parallel corpora as edge weights. We then transform it into the *final MAG*, which uses normalized edge weights and contains only a filtered subset of the raw MAG's nodes and edges.

Figure 2 shows the neighborhood of the Mina-Gen word "*màmayɔviwoa*" (grandchild of a female person, according to a Mina-Gen speaker) in the final MAG. Four words from the development languages have a link to an SDQ, while the Mina-Gen (zero-shot evaluation language) word does not.

GUIDE's preprocessing pipeline converts our dataset into the raw MAG and converts the raw MAG into the final MAG. Appendix C visualizes the individual steps. Note that we do not remove stop words.

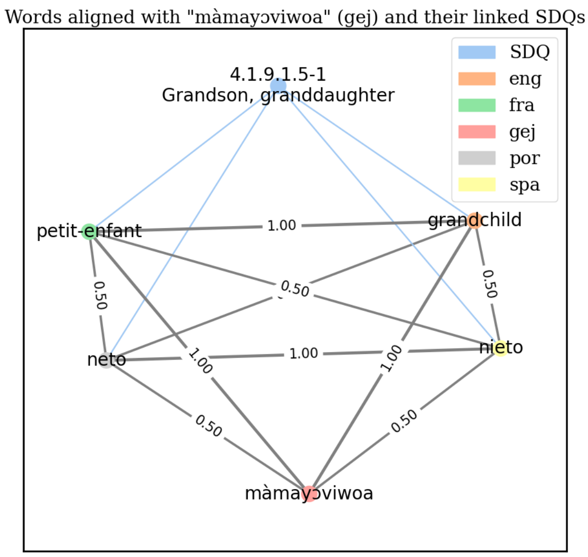

Figure 2: A subgraph from the final MAG showing the 1-hop neighborhood of the Mina-Gen word "*màmayɔviwoa*": The gray edges are word alignments with their normalized strength. Edges with higher strengths are thicker. The blue edges are SDQ links. The shown SDQ 4.1.9.1.5-1 is "*What words refer to the children of your children?*" The SDQ is shown here as a separate node, although it is technically part of the word nodes' feature vectors. To improve readability, the graph excludes some languages.

5.1.1 Tokenization

The first step of our preprocessing pipeline is tokenization. Depending on the language, we use different tokenizers.

Stanza tokenizer. A *Stanza* (Qi et al., 2020) tokenizer exists for eight of the 20 languages in our dataset: Chinese (simplified), English, French, Hindi, Indonesian, Portuguese, Spanish, and German. All of them are high-resource languages.

SentencePiece. If the Stanza toolkit does not provide a tokenizer, we use a language-agnostic tokenizer. For six agglutinative languages (Bengali, Malayalam, Nepali, Swahili, South Azerbaijani, and Igbo), we invoke *SentencePiece* (Kudo and Richardson, 2018) to identify subwords. We train the SentencePiece tokenizer for each of these six languages with a vocabulary size of 10,000.

Punctuation mark splitting. If we cannot use Stanza, and the language is not agglutinative, we resort to simply splitting at punctuation marks (including whitespace). Specifically, we use such punctuation mark splitting for Kupang Malay, Hiri Motu, Mina-Gen, Motu, Tok Pisin, and Yoruba.

5.1.2 Term Normalization

Multi-Word Terms. For each language covered by the Stanza toolkit (Qi et al., 2020), we perform additional preprocessing steps: *Part-of-Speech* (POS)-Tagging, *Multi-Word Token* (MWT) expansion (only for French, Indonesian, Portuguese, Spanish, and German), and lemmatization. MWT expansion merges common combinations of tokens. It produces, for example, "*arc-en-ciel*" (rainbow) in French and "*guarda-costa*" (coastguard) in Portuguese.

Case Normalization. We normalize the words in all languages with Latin script by lowercasing them.

5.1.3 Edge Induction

Word-SDQ Matching. For all languages in our development dataset, we assign all matching SDQs to each word. We perform this matching by simply looking for exact matches in the semantic domain dictionary for the respective language.

Word Alignment. The core assumption of this paper is that words with similar meanings would be aligned. Similar to Imani Googhari et al. (2022), we use the Eflomal statistical word aligner (Östling and Tiedemann, 2016) to generate bilingual alignments for each language pair in our dataset, except for pairs of two zero-shot evaluation languages because both have no labels. We post-process the unidirectional alignments of Eflomal with atools[5] and the *grow-diag-final-and* (GDFA) heuristic (Koehn et al., 2005) to obtain symmetric bilingual alignments. We also aggregate all alignments by word, resulting in the raw MAG.

5.1.4 Graph Refinement

Three processing steps convert our raw MAG to the final MAG.

Edge Weight Normalization. In the raw MAG, each edge between two word nodes u and v has a weight $w_{\text{raw}}(u, v) \in \mathbb{N}^+$ that we convert to a normalized weight $w_{\text{norm}}(u, v) \in (0, 1]$:

$$w_{\text{norm}}(u, v) = 2\,\frac{w_{\text{raw}}(u, v)}{S_{L(v)}(u) + S_{L(u)}(v)}$$

where $S_{L(v)}(u)$ is the strength of node u concerning the language of v, specifically the sum of the edge weights of all edges from word u to a word in language v.

Edge Weight Filtering. To reduce noisy alignments, we remove all edges (u, v) with a weight $w_{\text{norm}}(u, v) < 0.2$.

Isolated Node Removal. As the final preprocessing step, we remove all words from the graph that have no edge to a word in the development dataset, including words from such development languages. We call such words *isolated* even though they may have neighbors in a zero-shot evaluation language. This process reduces the number of nodes in the MAG by 52% – from 414,964 to 199,605, which is the final number of nodes in the MAG.

5.2 Graph Neural Network

GUIDE uses a *Graph Neural Network* (GNN) (Scarselli et al., 2009) to perform a massively multi-class multi-label classification. Each class is one of 7,425 SDQs.

5.2.1 Node Features

We train the GNN by representing each node with a set of features, using two main types of node features (Duong et al., 2019): graph structural features and word meaning features.

Graph structural features. Inspired by Imani et al. (2022), we incorporate *node degree* and *weighted node degree* (i.e., the sum of adjacent weights) as additional graph structural information. These two features are continuous numbers.

Word meaning features. We further incorporate *SDQ count* and *SDQ link* features. While the SDQ count is an integer (stored as a continuous number), the SDQ links are a multi-hot vector with 7,425 dimensions (i.e., these links are categorical features). In total, each node/word receives a vector with 7,428 feature values.

5.2.2 Model Architecture

The GNN adopts a *Graph Convolutional Network* (GCN) (Kipf and Welling, 2017) architecture, as implemented in *PyTorch Geometric* (Fey and Lenssen, 2019). Appendix C visualizes its fairly simple architecture. After adding the node features to the final MAG, the single-layer GCN (a *GCNConv*[6] layer) aggregates the features of each node's neighbors. The results are 7,425 scores per node, one for each SDQ. We normalize these scores with *sigmoid* as a non-linear output activation function. Finally, we apply a threshold, accepting only word-SDQ links with a score ≥ 0.999. The GCNConv layer has 55,160,325 parameters in total.

Modified Identity Matrix Initialization. After initializing the weight matrix and bias vector of our model's GCN layer with small random weights, we overwrite parts of it. Our initialization strategy is similar to an identity initialization, which uses an identity matrix as a weight matrix.

Our weight matrix has the shape 7,428 $\times$ 7,425 (see Section 5.2.1). Of the 7,428 input features, 7,425 are a multi-hot vector that encodes the SDQ links. We modify the identity initialization by overwriting the diagonal of this 7,425 $\times$ 7,425 submatrix with large weights (50.0). We also initialize the entire bias vector with low weights (-5.0). Thus,

[5] fast-align repository: https://github.com/clab/fast_align (visited on 2023-10-20)

[6] PyTorch Geometric documentation: https://pytorch-geometric.readthedocs.io/en/latest/generated/torch_geometric.nn.conv.GCNConv.html (visited on 2023-10-10)

during optimization, the learning process starts at the point that a word can e.g. belong to the SDQ "*What words refer to the sun?*" only if at least one neighbor does.

Soft F_1 Loss. As the loss function, we use the soft F_1 loss[7]. The soft F_1 loss uses continuous ("*soft*") instead of discrete values.

6 Experimental Setup

This section provides details about the environment in which we executed GUIDE and how we evaluated it.

6.1 Configuration

We split our development data with a random 80%/10%/10% node split. We train the model ten times in a range of 30 to 40 epochs on the development dataset with a batch size of 6,000 and a learning rate of 0.05 using *Adam* optimization (Kingma and Ba, 2014). We use early stopping after five epochs with a warm-up time of 30 epochs.

Hardware. We run all experiments on an *ASUS ESC8000 G4* with 500 GB of RAM, two *AMD Intel Xeon Silver 4214 processors* with twelve cores and 2.20 GHz, and eight *NVIDIA Quadro RTX A6000* (each having 48 GB VRAM) GPUs. We train the model on a single GPU. The entire training process takes less than 30 minutes and the inference time is approximately ten milliseconds per word.

6.2 Evaluation Setup

We use two evaluation methods: evaluation on the incomplete semantic domain dictionaries (dataset-based) and manual evaluation (questionnaire-based).

Dataset-based Evaluation. In the calculation of soft F_1 loss as well as precision, recall, and F_1 score, we ignore "empty" SDQs. An empty SDQ is an SDQ that has no assigned words in the dataset in a specific language. Ignoring empty questions allows us to evaluate our model even using incomplete semantic domain dictionaries.

Human Evaluation using Questionnaires. For each language, we built one questionnaire to evaluate 100 – 120 random and shuffled predicted word-SDQ links. We recruited human annotators who speak the respective languages, in part by seeking out language-specific online fora and communities. The human annotators who answered the questionnaires could select only "*yes*" or "*no*" for each pair. We always consider only the first 100 answers in the evaluation.

Appendix D lists the URLs to the 20 completed questionnaires. To clarify the SDQs, we also provided a list of valid English answers for each SDQ (except in the English questionnaire). The 14 languages for which a tokenizer or lemmatizer could change a word's spelling (see Section 5.1.1 and Section 5.1.2) also included this note: *"Please also answer "yes" if there is a typo but you still recognize a matching word."* An example of a preprocessing-related "*typo*" is the German word "*Hüfte*" (hip), which became "*huft*". This word does not exist because the Stanza lemmatizer applied stemming.

7 Evaluation

This section evaluates GUIDE's performance, shows the results of an ablation study, and discusses the findings.

7.1 Results

Table 2 shows the evaluation results. We include a random baseline, as we are not aware of any other approach to automatically link words from low-resource languages to SDQs. For each word-SDQ pair, the random classifier predicts an existing word-SDQ link with a probability of 50%. There are $N = 199,605$ words in the MAG with 81,632 links to SDQs. Therefore, the random classifier predicts 741 million ($N \times 7,425/2$) word-SDQ links, of which 40,816 are correct. This ratio leads to a precision of 0.00006. The recall is 0.5, and the F_1 score is 0.0001.

7.2 Ablation Study

Table 3 shows how GUIDE's (dataset-based) performance changes when components are removed.

Interestingly, four components harm the model's F_1 score: the isolated node removal and all features of the node feature vector, but the SDQ link feature.

7.3 Discussion of the Results

GUIDE predicted 71,094 word-SDQ links in total, of which 19,166 (37%) belong to zero-shot evaluation languages. 31,873 (62%) of the links predicted for the development languages are new. Because the total number of matched words in the MAG is 199,605, the model predicts one word-SDQ link

[7]Our implementation is inspired by this GitHub page: https://gist.github.com/SuperShinyEyes/dcc68a08ff8b615442e3bc6a9b55a354 (visited on 2023-10-16).

Language	Evaluation with dataset Precision	Recall	F_1	Manual evaluation Precision	# Predicted links
Random baseline	0.00	**0.500**	0.000	n/a	741,033,563
Development					
Bengali	0.22 ± 0.11	0.002 ± 0.001	0.004 ± 0.003	0.56	2,809 (2,770)
Chinese (simplified)	0.17 ± 0.02	0.014 ± 0.002	0.026 ± 0.004	0.34	5,752 (**5,036**)
English	**0.63** ± 0.02	**0.125** ± 0.006	**0.208** ± 0.009	**0.86**	7,119 (2,314)
French	0.59 ± 0.03	0.097 ± 0.005	0.167 ± 0.008	0.78	6,993 (2,527)
Hindi	0.25 ± 0.02	0.029 ± 0.003	0.051 ± 0.006	0.78	3,914 (2,835)
Indonesian	0.34 ± 0.05	0.035 ± 0.005	0.064 ± 0.009	0.77	1,799 (1,068)
Kupang Malay	0.14 ± 0.05	0.013 ± 0.005	0.024 ± 0.009	0.79	1,440 (1,351)
Malayalam	0.10 ± 0.03	0.015 ± 0.004	0.026 ± 0.007	0.45	2,768 (2,480)
Nepali	0.20 ± 0.01	0.022 ± 0.002	0.039 ± 0.004	0.38	2,641 (2,156)
Portuguese	0.43 ± 0.02	0.088 ± 0.006	0.146 ± 0.009	**0.86**	6,759 (3,737)
Spanish	0.59 ± 0.02	0.090 ± 0.005	0.155 ± 0.008	0.84	**7,614** (3,579)
Swahili	0.33 ± 0.04	0.018 ± 0.003	0.033 ± 0.005	0.75	2,320 (2,020)
Evaluation (zero-shot)					
German	n/a	n/a	n/a	0.67	**5,022**
Hiri Motu	n/a	n/a	n/a	0.62	1,190
Igbo	n/a	n/a	n/a	0.45	1,405
Mina-Gen	n/a	n/a	n/a	**0.80**	3,063
Motu	n/a	n/a	n/a	0.32	2,731
South Azerbaijani	n/a	n/a	n/a	0.58	2,238
Tok Pisin	n/a	n/a	n/a	0.69	880
Yoruba	n/a	n/a	n/a	0.63	2,637
Averages					
Development set	0.33 ± 0.04	0.046 ± 0.004	0.079 ± 0.007	0.68 ± 0.19	4,327 ± 2,338
Zero-shot evaluation set	n/a	n/a	n/a	0.60 ± 0.15	2,396 ± 1,324
Stanza	**0.43** ± 0.02	**0.068** ± 0.005	**0.117** ± 0.008	**0.74** ± 0.17	**5,622** ± 1,975
SentencePiece	0.21 ± 0.05	0.014 ± 0.003	0.026 ± 0.005	0.53 ± 0.13	2,364 ± 524
Punctuation mark split	0.14 ± 0.05	0.013 ± 0.005	0.024 ± 0.009	0.64 ± 0.18	1,990 ± 927
Total	0.33 ± 0.04	0.046 ± 0.004	0.079 ± 0.007	0.65 ± 0.18	3,555 ± 2,180

Table 2: Evaluation results: For each development language, cells with "±" show the average value of ten runs and the standard deviation. In the six bottom rows, "±" shows the average and the respective standard deviation. The six "average" rows show the average values for the development set, zero-shot evaluation set, and the languages tokenized with Stanza, SentencePiece, and punctuation mark splitting, respectively (see Section 5.1.1), as well as the average of all languages. The number of predicted word-SDQ links in the rightmost column is only from the run that we used to create the questionnaires. The number in parentheses is the number of new links. The highest values in each category are bolded.

per 2.8 words. Taking the model's precision of 0.65 into account, it predicts one correct word-SDQ link per 4.4 words. This number demonstrates GUIDE's few-shot learning capabilities.

The human evaluation using questionnaires reveals that GUIDE's precision is in fact almost twice as high as suggested by the dataset-based evaluation (0.65 instead of 0.34). The precision of 0.65 and the (dataset-based) recall of 0.046 show that the model predicts mostly correct word-SDQ links, but it creates only fractions of complete semantic domain dictionaries. Nevertheless, the recall is likely to be higher in practice because the evaluation with the incomplete dataset fails to recognize true positive predictions. While GUIDE cannot replace linguists who compile semantic domain dic-

	Δ Precision	Δ Recall	ΔF_1
GUIDE (reference values)	0.33 $\pm$ 0.04	0.046 $\pm$ 0.004	0.079 $\pm$ 0.007
Preprocessing			
¬ Stanza pipeline	−0.01 $\pm$ 0.04	−0.017 $\pm$ 0.005	−0.027 $\pm$ 0.008
¬ MWT expansion	+0.02 $\pm$ 0.03	−0.001 $\pm$ 0.003	−0.001 $\pm$ 0.006
¬ Lemmatization	+0.00 $\pm$ 0.03	−0.016 $\pm$ 0.003	−0.025 $\pm$ 0.006
¬ SentencePiece tokenization	−0.00 $\pm$ 0.04	−0.005 $\pm$ 0.003	−0.008 $\pm$ 0.006
¬ Lowercasing	+0.01 $\pm$ 0.05	−0.001 $\pm$ 0.005	−0.002 $\pm$ 0.008
¬ Isolated node removal	−0.02 $\pm$ 0.03	+0.013 $\pm$ 0.006	+0.019 $\pm$ 0.009
Node features			
¬ Degree	−0.00 $\pm$ 0.04	+0.012 $\pm$ 0.005	+0.016 $\pm$ 0.007
¬ Weighted degree	+0.01 $\pm$ 0.04	+0.007 $\pm$ 0.004	+0.010 $\pm$ 0.007
¬ SDQ count	+0.02 $\pm$ 0.04	+0.001 $\pm$ 0.004	+0.002 $\pm$ 0.007
¬ SDQ links	−0.33 $\pm$ 0.00	−0.045 $\pm$ 0.000	−0.077 $\pm$ 0.001
Other			
¬ Modified identity matrix initialization	−0.05 $\pm$ 0.07	−0.038 $\pm$ 0.001	−0.063 $\pm$ 0.003

Table 3: Changes in GUIDE's performance for eleven ablations: Cells with "$\pm$" show the average value of three runs and the standard deviation. Deactivating the Stanza pipeline and SentencePiece tokenization means that we used tokenization by punctuation mark split instead (see Figure 4).

tionaries, it can provide an initial dictionary with thousands of entries, of which a significant percentage is correct.

8 Conclusion

This paper presents the language-agnostic tool GUIDE, which creates and fills up multiparallel semantic domain dictionaries in 20 languages from seven language families. The model achieves state-of-the-art performance in linking words to their SDQs and supports 833 languages. Although GUIDE has a recall of only 0.046, we show that it has a precision of 0.60 even in languages for which it has no training data, probably due to language similarity and the model's multilingual nature.

We propose 32,000 new word-SDQ links for twelve existing dictionaries and 19,000 word-SDQ links for eight new dictionaries. Ten out of these 20 languages are low-resource languages.

Limitations

We discuss the limitations of our approach across multiple components.

Computational Limitations

The node feature matrix is a memory bottleneck. It is saved as a dense vector of size $N \times 7,428$, where N is the number of nodes/words. The model allocates approximately 3.5 GB of VRAM per language. Therefore, 48 GB of VRAM (see Section 6.1) limit us to loading approximately 13 languages. Therefore, we cannot load the entire MAG of 20 languages at once but load a subgraph of the twelve development languages plus only a single zero-shot evaluation language. This approach does not affect the quality of the results because the evaluation languages do not have labeled data and cannot learn word-SDQ links from each other.

Dataset

The used Bible translations and semantic domain dictionaries cause various limitations that we discuss in the following.

Bible translations. The general challenge of using only the Bible as parallel data is the narrow domain (Ebrahimi and Kann, 2021). The Bible does not include all the words used in today's world, particularly those related to technology, science, and modern culture, such as "*computer*". The language in the Bible is often of a high register and does not reflect the way people talk in everyday life (e.g., with slang and idioms). Although different Bible translations convey the same meaning, they differ in their proximity to the original text (in Hebrew, Aramaic, and Greek). While some are literal translations, others paraphrase a lot to be understandable to a modern audience. These different approaches to Bible translation cause noise in the word alignments.

Semantic domain dictionaries. The semantic domain dictionaries are incomplete. They cover a part of the languages' vocabularies and are also missing SDQ links for the words they cover. This limitation is the nature of language data because living languages are constantly evolving. They receive new words and new meanings for existing words. However, we found only a handful of incorrect SDQ links in our training data, listed in Appendix E.

Preprocessing Pipeline

The preprocessing pipeline produces a MAG that contains misleading edges (leading to false positives) and lacks useful edges and nodes (leading to false negatives). We now discuss three reasons for these limitations.

Ambiguity. Words are often ambiguous (e.g., "*date*") and thus align to different words in another language. The preprocessing pipeline treats them as if they are the same word, which confuses semantic patterns in the MAG, leading to misclassifications.

Noisy alignments. There is a lot of noise in word alignments because we train Eflomal on a small corpus that contains many words only once. We mitigate this noise by aggregating all alignments from all languages.

Collocations. We ignore most word groups (so-called collocations (Smadja et al., 1996), e.g., "*harvest moon*") in the semantic domain dictionaries unless Stanza provides an MWT expansion model for the language.

Node Features

Although three node features turned out to harm the model's performance, it could also ignore other potentially useful node properties.

Ethics Statement

We are not aware of any adverse effects on any individual or group resulting from the study we have conducted. However, we acknowledge that the limitations raised above may lead to dictionaries of inferior quality compared to manual language documentation. Thus, automated techniques cannot be taken as a reason to forgo traditional language documentation.

Acknowledgments

We would like to thank Lukas Ehrig and Jonathan Schneider for their kind and thoughtful feedback, as well as the anonymous SIGTYP reviewers. We also would like to express our sincere gratitude to everyone who completed a language questionnaire.

References

Cosmas Julius Abah, Jane Wong Kong Ling, and Anantha Govindasamy. 2018. Root-Oriented Words Generation: An Easier Way Towards Dictionary Making for the Dusunic Family of Languages. *Malaysian Journal of Social Sciences and Humanities (MJSSH)*, 3(2):95 – 112.

Khalid Alnajjar, Mika Hämäläinen, Niko Tapio Partanen, and Jack Rueter. 2022. Using graph-based methods to augment online dictionaries of endangered languages. In *Proceedings of the Fifth Workshop on the Use of Computational Methods in the Study of Endangered Languages*, pages 139–148, Dublin, Ireland. Association for Computational Linguistics.

Brenda H. Boerger. 2017. Rapid Word Collection, dictionary production, and community well-being. In *International Conference on Language Documentation & Conservation*.

Gerard de Melo and Gerhard Weikum. 2009. Towards a Universal Wordnet by Learning from Combined Evidence. In *Proceedings of the 18th ACM Conference on Information and Knowledge Management (CIKM)*, pages 513–522, New York, NY, USA. ACM Press.

Chi Thang Duong, Thanh Dat Hoang, Haikun Dang, Quoc Viet Hung Nguyen, and K. Aberer. 2019. On Node Features for Graph Neural Networks. *arXiv*.

David M. Eberhard, Gary F. Simons, and Charles D. Fennig. 2023. *Ethnologue: Languages of the World*, twenty-sixth edition. SIL International.

Abteen Ebrahimi and Katharina Kann. 2021. How to adapt your pretrained multilingual model to 1600 languages. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4555–4567, Online. Association for Computational Linguistics.

Christiane Fellbaum. 2000. WordNet: An Electronic Lexical Database. *Language*, 76:706.

Matthias Fey and Jan E. Lenssen. 2019. Fast Graph Representation Learning with PyTorch Geometric. In *ICLR Workshop on Representation Learning on Graphs and Manifolds*.

Naman Goyal, Cynthia Gao, Vishrav Chaudhary, Peng-Jen Chen, Guillaume Wenzek, Da Ju, Sanjana Krishnan, Marc'Aurelio Ranzato, Francisco Guzmán, and Angela Fan. 2022. The Flores-101 evaluation benchmark for low-resource and multilingual machine translation. *Transactions of the Association for Computational Linguistics*, 10:522–538.

Francisco Guzmán, Peng-Jen Chen, Myle Ott, Juan Pino, Guillaume Lample, Philipp Koehn, Vishrav Chaudhary, and Marc'Aurelio Ranzato. 2019. The FLORES evaluation datasets for low-resource machine translation: Nepali–English and Sinhala–English. 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 6098–6111, Hong Kong, China. Association for Computational Linguistics.

Ayyoob Imani, Lütfi Kerem Senel, Masoud Jalili Sabet, François Yvon, and Hinrich Schuetze. 2022. Graph neural networks for multiparallel word alignment. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 1384–1396, Dublin, Ireland. Association for Computational Linguistics.

Ayyoob Imani Googhari, Lütfi Kerem Şenel, Masoud Jalili Sabet, François Yvon, and Hinrich Schütze. 2022. Graph Neural Networks for Multiparallel Word Alignment. In *arXiv*.

Ayyoob ImaniGooghari, Silvia Severini, Masoud Jalili Sabet, François Yvon, and Hinrich Schütze. 2022. Graph-based multilingual label propagation for low-resource part-of-speech tagging. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 1577–1589, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Diederik P. Kingma and Jimmy Ba. 2014. Adam: A Method for Stochastic Optimization. *arXiv*.

Thomas N. Kipf and Max Welling. 2017. Semi-Supervised Classification with Graph Convolutional Networks. *arXiv*, pages 215–223.

Philipp Koehn, Amittai Axelrod, Alexandra Birch Mayne, Chris Callison-Burch, Miles Osborne, and David Talbot. 2005. Edinburgh system description for the 2005 IWSLT speech translation evaluation. In *Proceedings of the Second International Workshop on Spoken Language Translation*, Pittsburgh, Pennsylvania, USA.

Taku Kudo and John Richardson. 2018. SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 66–71, Brussels, Belgium. Association for Computational Linguistics.

Ronald Moe. 2010. Compiling Dictionaries Using Semantic Domains. *Lexikos*, 13.

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*, pages 101–108, Online. Association for Computational Linguistics.

Franco Scarselli, Marco Gori, Ah Chung Tsoi, Markus Hagenbuchner, and Gabriele Monfardini. 2009. The Graph Neural Network Model. *IEEE Transactions on Neural Networks and Learning Systems*, 20(1):61–80.

Frank Smadja, Kathleen R. McKeown, and Vasileios Hatzivassiloglou. 1996. Translating collocations for bilingual lexicons: A statistical approach. *Computational Linguistics*, 22(1):1–38.

NLLB Team, Marta R. Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, Anna Sun, Skyler Wang, Guillaume Wenzek, Al Youngblood, Bapi Akula, Loic Barrault, Gabriel Mejia Gonzalez, Prangthip Hansanti, John Hoffman, Semarley Jarrett, Kaushik Ram Sadagopan, Dirk Rowe, Shannon Spruit, Chau Tran, Pierre Andrews, Necip Fazil Ayan, Shruti Bhosale, Sergey Edunov, Angela Fan, Cynthia Gao, Vedanuj Goswami, Francisco Guzmán, Philipp Koehn, Alexandre Mourachko, Christophe Ropers, Safiyyah Saleem, Holger Schwenk, and Jeff Wang. 2022. No Language Left Behind: Scaling Human-Centered Machine Translation. *arXiv*.

Vesa Åkerman, David Baines, Damien Daspit, Ulf Hermjakob, Taeho Jang, Michael Martin, Joel Mathew, and Marcus Schwarting. 2023. The eBible Corpus: Data and Model Benchmarks for Bible Translation for Low-Resource Languages. *arXiv*.

Robert Östling and Jörg Tiedemann. 2016. Efficient Word Alignment with Markov Chain Monte Carlo. *The Prague Bulletin of Mathematical Linguistics*, 106(1):125–146.

A Appendix A. Semantic Domain Hierarchy

Figure 3 visualizes the semantic domain hierarchy.

B Appendix B. Bible Translation Sources

Table 4 shows the web source of the Bible translations we used.

C Appendix C. Pipeline and Model Visualization

Figure 4 and Figure 5 visualize our preprocessing pipeline. Figure 6 shows the model architecture.

D Appendix D. Questionnaires

Table 5 provides the links to the questionnaires that we used to manually evaluate GUIDE's performance.

E Appendix E. Incorrect Semantic Domain Dictionary Entries

Table 6 shows incorrect entries that we discovered in the development dataset.

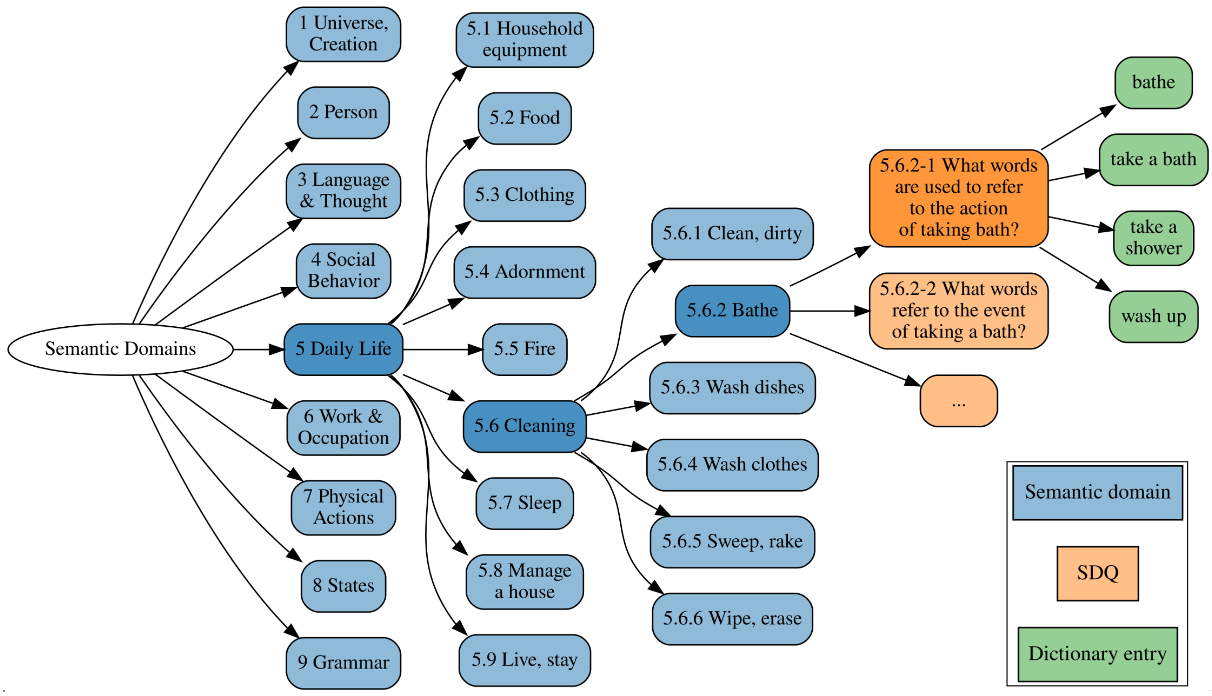

Figure 3: A drill-down into the tree-structured hierarchy of semantic domains: Nodes with expanded children are highlighted.

Language	Bible translation URL
Development	
Bengali	`https://github.com/BibleNLP/ebible/blob/main/corpus/ben-ben2017.txt`
Chinese	`https://github.com/BibleNLP/ebible/blob/main/corpus/cmn-cmn-cu89s.txt`
English	`https://github.com/BibleNLP/ebible/blob/main/corpus/eng-eng-web.txt`
French	`https://github.com/BibleNLP/ebible/blob/main/corpus/fra-frasbl.txt`
Hindi	`https://github.com/BibleNLP/ebible/blob/main/corpus/hin-hin2017.txt`
Indonesian	`https://github.com/BibleNLP/ebible/blob/main/corpus/ind-ind.txt`
Kupang Malay	`https://github.com/BibleNLP/ebible/blob/main/corpus/mkn-mkn.txt`
Malayalam	`https://github.com/BibleNLP/ebible/blob/main/corpus/mal-mal.txt`
Nepali	`https://github.com/BibleNLP/ebible/blob/main/corpus/npi-npiulb.txt`
Portuguese	`https://github.com/BibleNLP/ebible/blob/main/corpus/por-porbrbsl.txt`
Spanish	`https://github.com/BibleNLP/ebible/blob/main/corpus/spa-spablm.txt`
Swahili	`https://github.com/BibleNLP/ebible/blob/main/corpus/swh-swhulb.txt`
Evaluation (zero-shot)	
German	`https://github.com/BibleNLP/ebible/blob/main/corpus/deu-deu1951.txt`
Hiri Motu	`https://github.com/BibleNLP/ebible/blob/main/corpus/hmo-hmo.txt`
Igbo	`https://ebible.org/details.php?id=ibo`
Mina-Gen	`https://www.bible.com/sl/versions/2236-gen-gegbe-biblia-2014`
Motu	`https://github.com/BibleNLP/ebible/blob/main/corpus/meu-meu.txt`
South Azerbaijani	`https://github.com/BibleNLP/ebible/blob/main/corpus/azb-azb.txt`
Tok Pisin	`https://github.com/BibleNLP/ebible/blob/main/corpus/tpi-tpi.txt`
Yoruba	`https://github.com/BibleNLP/ebible/blob/main/corpus/yor-yor.txt`

Table 4: The links show the source of the Bible translations: All translations are from ebible.org, except for the Mina-Gen Bible, which was provided by a language expert. We downloaded the Igbo Bible from ebible.org because it is not in the eBible corpus (i.e., on GitHub). All URLs were visited on 2023-10-21.

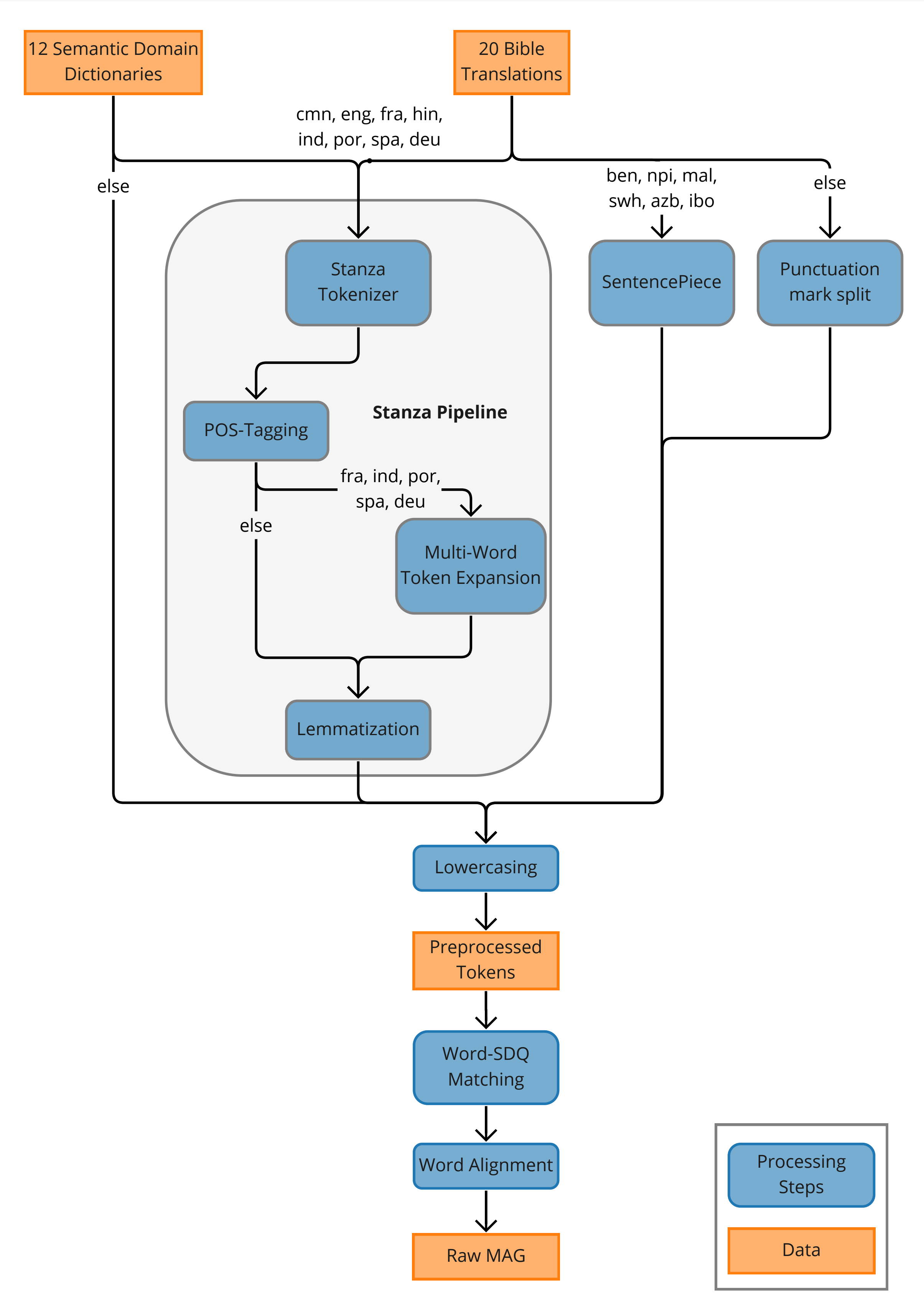

Figure 4: The first part of the preprocessing pipeline (graph creation).

Figure 5: The second part of the preprocessing pipeline (graph refinement).

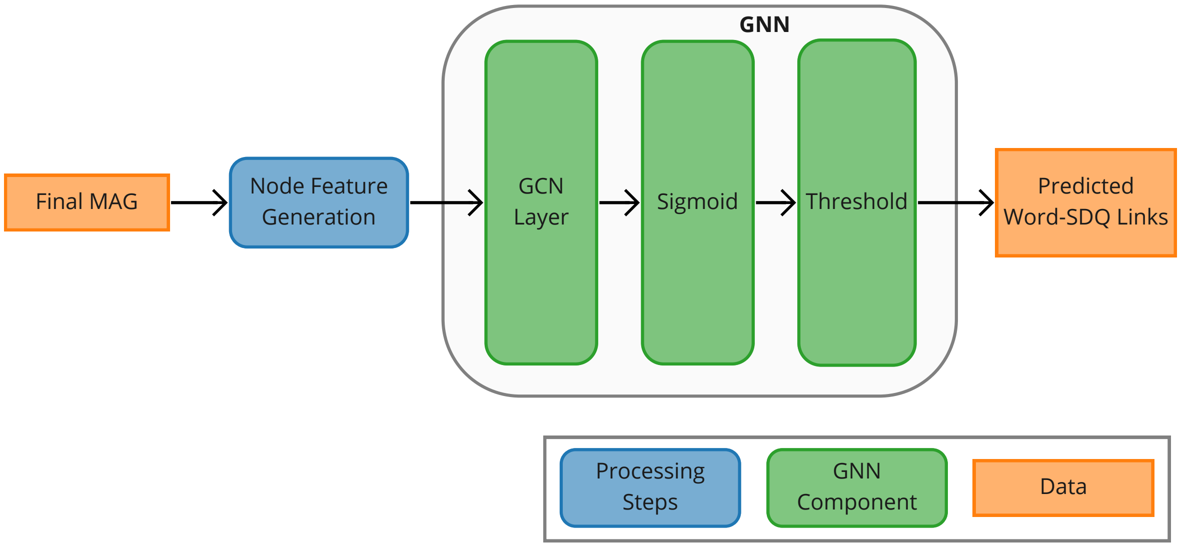

Figure 6: Model architecture: GUIDE takes the MAG with node features and predicts SDQ-links for these nodes.

Language	Questionnaire URL
Development	
Bengali	https://docs.google.com/spreadsheets/d/1_qoYnswufDY0gVZuebcoQ1DD9BLqSVD8NozWzqiGWR8
Chinese (simplified)	https://docs.google.com/spreadsheets/d/1sppwKhC5Ev3frbQ8Mq_MoQGc5ehym6QdQSPcjjdWNPg
English	https://docs.google.com/spreadsheets/d/1zt_3gqNrbSYsIOzjwm3BxewOaulFY1laVXshCZIYGLU
French	https://docs.google.com/spreadsheets/d/1eWkOK5T9ttWx-9ZmETc-fUY1Q7HzihbTr6mK8irZ83g
Hindi	https://docs.google.com/spreadsheets/d/14D6pGKgQtoHG5LWORaU9Ko5XUn_wDh0-x4Hnxj2nHag
Indonesian	https://docs.google.com/spreadsheets/d/13iVFF0xxwpQ_pXf-zKFW2jebA3TZnIiSL9rFpD-dWPY
Malayalam	https://docs.google.com/spreadsheets/d/1-DFjBkS1wjCahowBjg-iGLBV-moZww-J8lKpO0HN44Y
Nepali	https://docs.google.com/spreadsheets/d/1n-f9LbF0vYfO4gtu1YmD6LZB1_Gyo-VxV35WaYBN9_Q
Portuguese	https://docs.google.com/spreadsheets/d/1_WKQmj5KHDE6p8MsCFawvQoxOcLn3MYPWb-4aYpgV6U
Kupang Malay	https://docs.google.com/spreadsheets/d/1EP1ctJ7yl5QYFdY6eV6KYDQg1_mz90j-J8jDGwT9yJY
Spanish	https://docs.google.com/spreadsheets/d/1-2ZwbunnsqOYBW_beI9Rax3XW1Zjpacl5GrrzlPtfF0
Swahili	https://docs.google.com/spreadsheets/d/1H9RVi1mCkL9WmcH2zXYuOwwj73CAMg1My6P-jAAWYgI
Evaluation (zero-shot)	
German	https://docs.google.com/spreadsheets/d/1mPtzuD3_NFWOhLBXUElRNeAGtmiiXcKx_7n7Er3kZsc
Hiri Motu	https://docs.google.com/spreadsheets/d/1gTiNxhvRV9UtUq84Q0E3itJ2nYS8Gv4ElpIp3mEEHAE
Igbo	https://docs.google.com/spreadsheets/d/1yU8FCS19KRIWkbqm1aQBUCBEjVA4zoC60TuM0fTQN8Q
Mina-Gen	https://docs.google.com/spreadsheets/d/1Ib-xD6-1FuBLQ9M3F2UVbocnnbK7NBgv62Lg9h0p_6o
Motu	https://docs.google.com/spreadsheets/d/1e45Hw000K6OrluBQxe-8ifAR3h_Dz725sg3ZBlVnXRM
South Azerbaijani	https://docs.google.com/spreadsheets/d/1q8WfBhZDlOzRsihUbjH-wFyruudA11Chosotgf71rx0
Tok Pisin	https://docs.google.com/spreadsheets/d/1EENt0FJpTdDHpm2P1i-MQ56ZkdQKkBi5GnUDw30gx_o
Yoruba	https://docs.google.com/spreadsheets/d/11LBgUSHSnUFOP3Zp2ikgTp8vjGB6xR8cQkcdZeKNaSQ

Table 5: The completed questionnaires on Google Sheets for each of the 20 languages: We instructed the participants to answer 100 – 120 questions (see Section 6.2).

Language	Word	Translation	SDQ ID	SDQ
English	stock		3.2.5.1-1	What words refer to believing that something is true?
Hindi	राल (*rāl*)	resin	1.2.2.4-2	What types of minerals are there?
Portuguese	estoque	stock	3.2.5.1-1	What words refer to believing that something is true?
Portuguese	rebelião	rebellion	4.5.4.6-10	What do the authorities do to stop a rebellion?
Portuguese	estoque	stock	4.7.7.3-7	What means are used to restrain prisoners?
Portuguese	deter	detain	3.4.2.1.2-1	What words refer to feeling hateful?
Portuguese	carmesim	crimson	8.3.3.3.4-7	What are the shades of blue?
Spanish	rebelión	rebellion	4.5.4.6-10	What do the authorities do to stop a rebellion?
Spanish	sedición	sedition	4.5.4.6-10	What do the authorities do to stop a rebellion?

Table 6: Nine incorrect entries in the semantic domain dictionaries that we discovered, verified by native speakers: The incorrect word-SDQ links in the dataset are rare. The text in parentheses shows a transliteration.