Cross-lingual Alignment of Knowledge Graph Triples with Sentences

Swayatta Daw[1]*, Shivprasad Sagare[2]*, Tushar Abhishek[2]*,
Vikram Pudi[1] and Vasudeva Varma[2]
[1]Data Sciences and Analytics Center, IIIT Hyderabad, India
[2]Information Retrieval and Extraction Lab, IIIT Hyderabad, India
{swayatta.daw, shivprasad.sagare, tushar.abhishek}@research.iiit.ac.in
{vikram, vv}@iiit.ac.in

Abstract

The pairing of natural language sentences with knowledge graph triples is essential for many downstream tasks like data-to-text generation, facts extraction from sentences (semantic parsing), knowledge graph completion, etc. Most existing methods solve these downstream tasks using neural-based end-to-end approaches that require a large amount of well-aligned training data, which is difficult and expensive to acquire. Recently various unsupervised techniques have been proposed to alleviate this alignment step by automatically pairing the structured data (knowledge graph triples) with textual data. However, these approaches are not well suited for low resource languages that provide two major challenges: (1) unavailability of pair of triples and native text with the same content distribution and (2) limited Natural language Processing (NLP) resources. In this paper, we address the unsupervised pairing of knowledge graph triples with sentences for low resource languages, selecting Hindi as the low resource language. We propose cross-lingual pairing of English triples with Hindi sentences to mitigate the unavailability of content overlap. We propose two novel approaches: NER-based filtering with Semantic Similarity and Key-phrase Extraction with Relevance Ranking. We use our best method to create a collection of 29224 well-aligned English triples and Hindi sentence pairs. Additionally, we have also curated 350 human-annotated golden test datasets for evaluation. We make the code and dataset publicly available [†] and hope that this will help advance further research in this critical area.

1 Introduction

The pairing of structural data (knowledge graphs, Abstract Meaning Representations (AMRs), tables, databases, etc.) with natural languages sentences has led to the development of many downstream tasks such as Relation extraction (Ji et al., 2017), Knowledge graph population (Vu et al., 2021), dialog generation (Wen et al., 2016), Generation of natural text from structured data (Gardent et al., 2017; Parikh et al., 2020; Mager et al., 2020), etc.

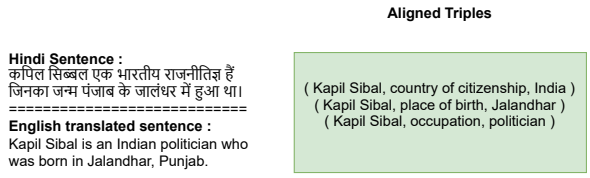

Figure 1: A Cross-lingual English triple and Hindi text Example (with English Translation)

Most existing methods solve above downstream tasks using neural-based end-to-end approaches that require a large amount of well-aligned human-annotated training data. However, the human-annotated dataset is expensive and difficult to obtain as annotators need to understand the structured data and natural text across various domains thoroughly. To overcome the lack of labeled data and difficulty in domain adaptation, unsupervised alignment has recently emerged as an active area of research (Fu et al., 2020; Agarwal et al., 2020; Fan and Gardent, 2020). Most of these unsupervised approaches utilize a large amount of structural and textual data having high content overlap. However, extending these approaches to low resource languages still poses a challenge due to the lack of structured data that has same content distribution as textual data.

In this work, we propose cross-lingual pairing of English triples with native language sentences to mitigate the unavailability of semantic content overlap for low resource languages. We select Hindi as low resource language for evaluating the

[†] https://www.dropbox.com/sh/lrh5q9odadixmqx/AABrTT7YjN6-xVLvviNpqQM6a?dl=0
*Equal Contribution

Proceedings of the 18th International Conference on Natural Language Processing, pages 629–637
Silchar, India. December 16 - 19, 2021.

efficiency of cross-lingual alignment. We explore alignment between the English triples present in Wikidata (Vrandečić and Krötzsch, 2014) with sentences extracted from Hindi Wikipedia articles.

Specifically, through this work, we aim to achieve the following objectives:

1. We introduce solid baselines for the cross-lingual alignment task and propose two novel approaches: NER-based filtering with Semantic Similarity and Key-phrase Extraction with Relevance Ranking. All the approaches mentioned in the paper can be extended to multiple languages, as we do not rely on language-based heuristics.

2. We use our best method to create a collection of 26302 well-aligned English triples and Hindi sentence pairs for training. Similarly, we create validation dataset consisting of 2922 data instances. Additionally, we have also collected 350 human-labeled gold test dataset to evaluate alignment methods.

The remainder of the paper is organized as follows. We discuss related work in Section 2. We discuss the dataset creation details in Section 3. We explain the proposed methods in Section 4. Additionally, we present baselines, experimental settings, results, and analysis in Section 5. Finally, we conclude with a summary of our work and future directions in Section 7.

2 Related work

Recently, there has been a lot of effort in creating automated structured data to text datasets in various domains. (Lebret et al., 2016) introduced a WikiBIO dataset by aligning opening sentences with infoboxes in English Wikipedia articles on person's biographies. Several extensions of this method of aligning Wikipedia text with infoboxes have been proposed to create a dataset in different languages (Nema et al., 2018) and domains (Qader et al., 2018). Datasets created using these methods are constrained to a specific domain. (Fu et al., 2020) alleviates this limitation by aligning knowledge graph triples in Wikidata with opening sentences in Wikipedia. It uses lexical overlap between the name entities present in a sentence, and Wikidata triples for alignment. In addition to using triples available in Wikidata (Wikipedia's Knowledge Graph), (Agarwal et al., 2020) introduced a dataset that also incorporates sub-property information in the form of quadruples. These datasets focus on aligning either knowledge graph triples or infoboxes with sentences present in Wikipedia articles. (Chen et al., 2021) introduced a dataset that combined the structured information residing in Wikidata and infoboxes with a given sentence. To scale alignment of structured data with natural text across various domains (Elsahar et al., 2018; Jin et al., 2020) introduced sequential pipeline strategy consisting of data collection, data filtering, entity linking, and alignment. Additionally, it also suggests incorporating a human-annotated test dataset to evaluate the different alignment methods.

All of the previous approaches depend upon lexical overlap between structured and textual data. These approaches are ineffective for cross-lingual alignment where structured data and textual data are in different languages. Although, we can utilize previously proposed strategies for dataset creation by translating either structured data or textual data to other languages. WebNLG 2020 (Castro Ferreira et al., 2020) shared task presents one such cross-lingual aligned dataset where Shimorina et al. (2019) performs automatic translation and post editing of English sentences to Russian. Final dataset consists of English triples aligned with Russain sentences verbalizing those triples. Such approaches do incur the loss due to automatic translation though. Later, we demonstrate that our proposed approach for cross-lingual alignment achieves comparatively better results.

3 Dataset Creation

3.1 Data collection

We use Wikidata as our Knowledge Graph (KG) for obtaining English triples and Hindi Wikipedia for fetching equivalent sentences. There exists an unambiguous one-to-one mapping between Wikidata entities and Wikipedia articles, which enables us to collect high-quality data for many entities. We initially explored all domains and subdomains of Wikipedia articles. We decided to choose the *person* domain in Hindi Wikipedia as it contains the maximum number of entities within a domain (˜16% of Hindi Wikipedia), allowing us to create a larger dataset. The article text and English triples are fetched and pre-processed for each entity having a Hindi Wikipedia page. Triples with non-useful predicates like external identifiers, URLs, etc., are removed. We extract the first three sen-

tences from each article using sentence tokenization in Hindi. This data acts as the input to our alignment models, which predict a relevant set of triples for each sentence out of that particular entity's entire candidate set of triples. We use our best-proposed approach to create a total of 29224 English triple and sentences pair covering 12429 entities.

3.2 Test Set Annotation

We also collected a human-annotated test set of 460 structured data and text pairs, apart from the unsupervised training and validation set. We sample the 460 instances for annotation from the above-collected data and present them to the user in our specially developed web-based UI. The user can see the sentence and all the candidate triples associated with that entity. Two of the authors independently annotated these instances. The Cohen's Kappa score i.e. inter-annotator agreement for the annotations, was found to be 0.74. Finally, with the help of a language expert, the final test data samples were agreed upon from annotations responses of both the authors. We select 350 data instances as test datasets on which we report the metrics scores of our approaches. The remaining 110 samples are used as internal validation set to tune the hyperparameters like threshold values.

The distribution of sentences and other statistics across different domains can be found in table 1.

Domain	Entity count	Sentence count	Sentence count (in test data)	Avg sentence length (in test data)	Avg fact count (in test data)
Actors	2106	5469	50	14.32	3.60
Cricketers	2316	4694	100	21.19	4.70
Politicians	3906	8916	100	18.64	3.47
Writers	2755	6629	50	15.65	1.78
Singers	739	1944	25	18.04	2.92
Journalists	607	1572	25	17.32	2.12
Total	12429	29224	350	17.52	3.08

Table 1: Table contains entity count and sentence count for final aligned dataset across different domains. It also presents statistics of manually annotated test data for each domain.

4 Unsupervised Cross-lingual Alignment

Our alignment model aims to align the most relevant English triples to Hindi sentences. We introduce two novel approaches for cross-lingual sentence and facts alignment task: 1) NER-based filtering with Semantic Similarity and 2) Key-phrase Extraction with Relevance Ranking.

NER-based filtering with Semantic Similarity incorporates a novel idea for Named Entity Disambiguation. We used Nearest Neighbor-based Search to find the most relevant English words for the given Hindi words in the sentence by projecting Hindi and English words in the same vector space. We use Multilingual Unsupervised and Supervised Embeddings (MUSE) (Lample et al., 2017) to obtain multilingual vector representation and then perform the Nearest Neighbor Search to obtain the top-k candidates. The chosen candidates are further filtered based on semantic similarity, which boosts the precision of the model. We experiment with several state-of-the-art multilingual transformer-based models to find semantic similarities between facts and sentences.

In Key-phrase Extraction with Relevance Ranking, we extract key phrases from a Hindi sentence based on simple POS-tag-based heuristics and then rank extracted key phrases in the sentence to their relevance with its corresponding constituent article. We propose a new multilingual variant of EmbedRank (Bennani-Smires et al., 2018) to obtain rankings. Top-k relevant triples are then selected based on similarity scores with the key phrases of a sentence.

4.1 NER-based filtering with Semantic Similarity

To obtain matching English triples for a given Hindi sentence s, the idea is to filter the triples using named entity recognition before matching them on semantic similarity. Our assumption is based on the observation that if a triple has a Named Entity, then the sentence with which it aligns will also have the same or a variation of that Named Entity. If a triple does not have a Named Entity, we consider it for finding semantic similarity with the sentence.

We concatenate each word in the triple together and then extract named entities from it. Our goal is to find the overlap between the words in the Hindi sentence to the Named Entities identified in the triple. There can be multiple variations in how a Named Entity is written in an Indian Language such as Hindi. So, using a direct translation would not suffice for the alignment objective. Additionally, there might be translation loss associated with it.

To circumvent this problem, we used a pipeline approach consisting of two stages: 1) Filtering of triples based on bucket approach, and 2) Semantic

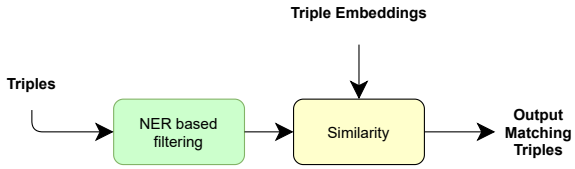

Figure 2: NER-based filtering + Semantic Similarity

similarity approach.

Filtering of triples based on bucket approach creates a bucket of English words by retrieving top-k nearest neighbor English words for each word present in the given Hindi sentence s from the common multilingual vector space created using MUSE (Lample et al., 2017). Then we calculate the intersection of the named entities identified in triple with the previously created bucket of English words for that Hindi sentence s. Finally, we obtain a score for each triple by dividing the amount of intersection by the total number of words present across all the named entities. We retain facts having score above a certain threshold, then proceed with semantic similarity in the next stage.

Semantic similarity approach further refines the triples obtained from the previous stage by calculating the inner product between the Hindi sentence representation and fact representation. Both the sentence level representation and fact level representation are obtained from multilingual transformer models as discussed in Section 5. Finally, we retain triples above a certain threshold (different threshold from the previous stage). We have illustrated the pipeline approach in Figure 2.

4.2 Key-phrase Extraction with Relevance Ranking

We extract the Hindi key phrases from the Hindi Wikipedia article based on simple POS-tag-based heuristics for this method. We define a phrase as a key phrase if it contains at least zero or more Adjectives followed by one or more Nouns. These obtained key phrases are ranked on how semantically similar they are to the input Hindi Wikipedia article. We call this process Key-phrase Extraction with Relevance Ranking. The ranking mechanism follows a multilingual variant of the EmbedRank (Bennani-Smires et al., 2018) method. The intuition behind EmbedRank is to embed candidate phrases and the corresponding article in the same high-dimensional vector space. Then, the key phrases are ranked based on closeness with

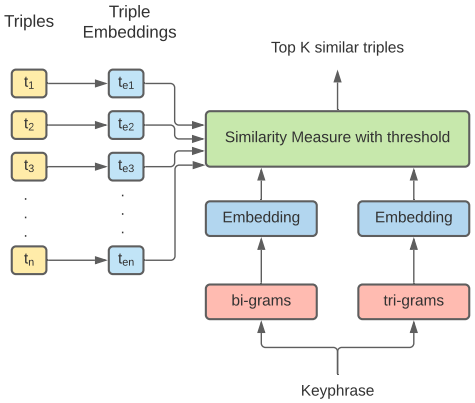

Figure 3: Method to return Top K triples from key phrases

the article in the same vector space. Our variant is explained in Algorithm 1.

Algorithm 1: Ranking key phrases with respect to Article Relevance

1. Let N = { set of all key phrases in article A}.
2. Concatenate all the key phrases in N and let $Nv \leftarrow$ vector representation of the concatenated key phrases.
3. For a sentence s in the article A, $M \leftarrow$ set of all extracted key phrases from s. So, $M \subseteq N$.
4. For each key phrase K in M, let $Kv \leftarrow$ vector representation of K.
5. Assign a $score$ to K , where $score$ = similarity between Kv and Nv.
6. Rank all the key phrases in M based on the $score$.

The process of obtaining similar triples from ranked key phrases is explained in Figure 3. After the key phrases are ranked for an article A, we extract n-grams for each key phrase. We find the vector embeddings for each n-gram and each triple. Now, each n-gram is compared with each triple, and a semantic similarity score is obtained. We keep the best matching triple for each n-gram. Then, we obtain the most similar triples per n-gram for a key phrase. Among them, we select top-k triples. These top-k triples form the most relevant triples for a key phrase. We combine the results from all key phrases in a Hindi sentence to obtain sentence-level matches.

5 Experiments

5.1 Baselines

We experimented with the following baselines:

Multilingual Universal Sentence Encoder (Yang et al., 2019a) is a general-purpose sentence embedding model for transfer learning and semantic text retrieval tasks. It relies on a standard dual-encoder neural framework with shared weights, trained in a multi-task setting with an additional translation bridging task. We use the same strategy to filter out the fact triples as mentioned for mBERT.

Word Overlap uses K Nearest neighbor search to choose K-most relevant English words for each Hindi word present in the Hindi sentence. The word search happens in a multilingual vector space created using MUSE (Lample et al., 2017). We keep all these top-K English words in a bucket. Then, we calculate the overlap between words in the triple and the English words in the bucket for each sentence. If the overlap is above a certain threshold, we classify that triple as aligned with that sentence.

Static Sentence Similarity use MUSE (Lample et al., 2017) to obtain multilingual word embeddings. We find the average of these word embeddings to create sentence representation for a Hindi sentence. We average all the word embeddings in a triple to obtain fact-level representation for that triple. Finally, we find the cosine similarity between sentence level and fact level representation and retain triples above a certain threshold for a given Hindi sentence.

mBERT (Devlin et al., 2018) (multilingual Bidirectional Encoder Representations from Transformers) encodes both the Hindi sentence and list of associated facts. Facts are verbalized by concatenating the subject, predicate, and object. We obtain the vector representations by taking the average of sub-word representation from the last layer of mBERT (mean pooling). Then, we find the cosine similarity score between the sentence and fact-level representation. Finally, we retain fact triples whose similarity score is greater than a certain threshold.

MuRIL (Khanuja et al., 2021) (Multilingual Representations for Indian Languages) is pre-trained on a significantly large amount of Indian text corpora with an extensive vocabulary for Indian languages. With MuRIL, we use the same strategy to filter out the fact triples as mentioned for mBERT.

LaBSE (Feng et al., 2020)(Language-Agnostic BERT Sentence Embedding) is a multilingual embedding model that encodes text from different languages into a shared embedding space pre-trained using the Masked Language Modeling and Translation Language Modeling objectives. With LaBSE, we use the same strategy to filter out the fact triples as mentioned for mBERT.

XLM-R (STS) and XLM-R (Paraphrase) are sentence transformers that fine-tune XLM-Roberta (Conneau et al., 2019) on semantic text similarity (STS) (Cer et al., 2017) and on multilingual paraphrase dataset (Yang et al., 2019b) respectively.

5.2 Experimental Settings

For the Word Overlap approach, we set the threshold value to 1 and fixed k=5 in k nearest neighbor retrieval. We translate the words which are out of the vocabulary. For all multilingual transformer-based methods: mBERT, MuRIL, LaBSE, multilingual universal sentence encoder, XLM-R, we use the base model available (consists 12 layers) on Huggingface (Wolf et al., 2020).

Threshold value	F1-Score
0.35	0.48
0.45	**0.55**
0.55	0.52
0.65	0.38

Table 2: Threshold values for sentence-triple semantic similarity on internal validation set for XLM-R (base)

The threshold value is set to 0.45 for cosine similarity after hyperparameter tuning on our internal validation dataset. We tried various pooling strategies like [CLS] token representation, sum pooling, and mean pooling for sentence-level representation. We found that mean pooling consistently performs the best.

K	F1-Score
3	0.65
4	0.72
5	**0.74**
6	0.66
7	0.67
8	0.68
9	0.66
10	0.63

Table 3: K value for K-Nearest neighbor for NER-based filtering with Semantic Similarity method (tested on internal validation set)

Figure 4: The first example is a predicted sample from the Key-phrase Extraction with Relevance Ranking approach. The second example is a predicted sample for the NER based filtering with Semantic Similarity approach. The prediction by each model is highlighted in bold in the candidate triples.

We determine the optimal K for K-Nearest neighbors and the optimal similarity threshold by tuning these hyperparameters on the internal validation set consists of 110 instances. We provide the detailed results of this hyperparameter search in Table 2 and Table 3. We obtain the optimal value for K in K-Nearest Neighbors as 5. Similarly, we observe the optimal value for the similarity threshold to be 0.45. We use XLM-R (base) as the reference transformer-based model as it is the best performing baseline.

For recognizing named entities, we use a BERT-CRF tagger trained on the OntoNotes dataset (Weischedel et al., 2017). We use AllenNLP (Gardner et al., 2017) for our NER implementation.

For Key-phrase Extraction with Relevance Ranking, we set n-gram values $\in [2, 3]$ and use Stanford coreNLP (Manning et al., 2014) to detect POS-tags. We used XLM-R (Paraphrase) as the multilingual transformer encoder with a similarity threshold of 0.45.

5.3 Evaluation Metric and Results

We use micro-average *Precision*, *Recall* and *F1-Score* as our evaluation metrics.From the results in Table 4, it is evident that MuRIL performs better than mBERT as it is solely pre-trained on Indian languages with extensive vocabulary size. Surprisingly, a simple approach like word overlap has higher recall than MuRIL. The reason is that it searches k-nearest neighbors in a multilingual vector space, as explained in section 5.1. So, this process captures more word variations while retrieving the facts. XLM-R (paraphrase) model in baselines performs better than other multilingual transformers as it is fine-tuned on the downstream tasks specific to text similarity. LaBSE is pre-trained on the translation language modeling loss. So, it effectively captures the semantic similarity between facts and sentences of different languages.

We observe that Key-phrase Extraction with Relevance Ranking has high precision. As the process captures the relevance of each key phrase with its article, it ensures to keep only those key phrases that are highly relevant to the article. The matches are refined further by n-gram matching with triples.

Surprisingly, NER-based filtering with Semantic Similarity gives the highest performance in terms of both precision and recall. This result shows that the most relevant fact triples are significantly biased towards having named entities as the primary factual information. Therefore, even though our Key-phrase Ranking method considers the entire context of an article to obtain relevant phrases, the NER-based model still performs better.

6 Error Analysis

Key-phrase Extraction with Relevance Ranking: As per the ranking mechanism, we keep only the most relevant top ranked triples. However, we notice that in some cases, especially where there are multiple triples which convey similar information, the model misses to capture all the relevant triples. Only the highest rank triples are considered, which leads to similar triples being missed out due to a slightly lower rank. In Figure 4, the first example is a predicted sample by the Key-phrase extraction model. We observe that occupation:author and oc-

S.no	Approaches	Precision	Recall	F1-Score
1	mBERT (mean pooling)	0.37	0.31	0.33
2	Static Sentence Similarity	0.38	0.48	0.42
3	Multilingual Universal Sentence Encoder	0.62	0.38	0.47
4	Word Overlap	0.50	0.52	0.51
5	LaBSE (mean pooling)	0.49	0.56	0.52
6	XLM-R (STS)	0.57	0.48	0.52
7	MuRIL (mean pooling)	0.55	0.51	0.53
8	XLM-R (paraphrase)	0.52	0.58	0.55
9	Key-phrase Extraction with Relevance Ranking	0.78	0.72	0.75
10	NER based filtering with Semantic similarity	**0.79**	**0.83**	**0.81**

Table 4: Precision, Recall and F1-score across different approaches.

cupation:novelist are missed out by the model, due to the ranking mechanism.

NER based filtering with Semantic Similarity: We notice that sometimes fact triples without named entities are being missed by the model. The second example in Figure 4 is a predicted sample by the NER-based model. We observe that occupation: politician has been ignored by the model, as "politician" is not a named entity.

7 Conclusion

We investigate the unexplored problem of cross-lingual alignment of English triples with sentences for low-resource languages like Hindi. This paper demonstrates the result over several baselines ranging from simple techniques like word overlap to more complex approaches that use pre-trained language models. Finally, we propose two novel methods of NER-based filtering with Semantic Similarity and Key-phrase Extraction with Relevance Ranking. We show through our experiments that these approaches perform better than the baselines on the human-annotated gold dataset, which we have created as a part of this project. We created a large dataset of English triples mapped with Hindi sentences using our best-performing model, making it publicly available for further research.

We plan to use the cross-lingual aligned dataset for various NLP tasks like text generation, KB population, and concept extraction for future work. Also, we are planning to extend this work to other Indian languages. We strongly believe that our alignment models and dataset will enhance the research undertaken for low-resource languages in the scientific community.

References

Oshin Agarwal, Heming Ge, Siamak Shakeri, and Rami Al-Rfou. 2020. Knowledge graph based synthetic corpus generation for knowledge-enhanced language model pre-training. *arXiv preprint arXiv:2010.12688*.

Kamil Bennani-Smires, Claudiu Musat, Andreea Hossmann, Michael Baeriswyl, and Martin Jaggi. 2018. Simple unsupervised keyphrase extraction using sentence embeddings. *arXiv preprint arXiv:1801.04470*.

Thiago Castro Ferreira, Claire Gardent, Nikolai Ilinykh, Chris van der Lee, Simon Mille, Diego Moussallem, and Anastasia Shimorina. 2020. The 2020 bilingual, bi-directional WebNLG+ shared task: Overview and evaluation results (WebNLG+ 2020). In *Proceedings of the 3rd International Workshop on Natural Language Generation from the Semantic Web (WebNLG+)*, pages 55–76, Dublin, Ireland (Virtual). Association for Computational Linguistics.

Daniel Cer, Mona Diab, Eneko Agirre, Inigo Lopez-Gazpio, and Lucia Specia. 2017. Semeval-2017 task 1: Semantic textual similarity-multilingual and cross-lingual focused evaluation. *arXiv preprint arXiv:1708.00055*.

Mingda Chen, Sam Wiseman, and Kevin Gimpel. 2021. Wikitablet: A large-scale data-to-text dataset for generating wikipedia article sections. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 193–209.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Unsupervised cross-lingual representation learning at scale. *arXiv preprint arXiv:1911.02116*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Hady Elsahar, Pavlos Vougiouklis, Arslen Remaci, Christophe Gravier, Jonathon Hare, Frederique Laforest, and Elena Simperl. 2018. T-rex: A large scale alignment of natural language with knowledge base triples. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*.

Angela Fan and Claire Gardent. 2020. Multilingual amr-to-text generation. *arXiv preprint arXiv:2011.05443*.

Fangxiaoyu Feng, Yinfei Yang, Daniel Cer, Naveen Arivazhagan, and Wei Wang. 2020. Language-agnostic bert sentence embedding. *arXiv preprint arXiv:2007.01852*.

Zihao Fu, Bei Shi, Wai Lam, Lidong Bing, and Zhiyuan Liu. 2020. Partially-aligned data-to-text generation with distant supervision. *arXiv preprint arXiv:2010.01268*.

Claire Gardent, Anastasia Shimorina, Shashi Narayan, and Laura Perez-Beltrachini. 2017. Creating training corpora for NLG micro-planners. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 179–188, Vancouver, Canada. Association for Computational Linguistics.

Matt Gardner, Joel Grus, Mark Neumann, Oyvind Tafjord, Pradeep Dasigi, Nelson F. Liu, Matthew Peters, Michael Schmitz, and Luke S. Zettlemoyer. 2017. Allennlp: A deep semantic natural language processing platform.

Guoliang Ji, Kang Liu, Shizhu He, and Jun Zhao. 2017. Distant supervision for relation extraction with sentence-level attention and entity descriptions. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 31.

Zhijing Jin, Qipeng Guo, Xipeng Qiu, and Zheng Zhang. 2020. Genwiki: A dataset of 1.3 million content-sharing text and graphs for unsupervised graph-to-text generation. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 2398–2409.

Simran Khanuja, Diksha Bansal, Sarvesh Mehtani, Savya Khosla, Atreyee Dey, Balaji Gopalan, Dilip Kumar Margam, Pooja Aggarwal, Rajiv Teja Nagipogu, Shachi Dave, et al. 2021. Muril: Multilingual representations for indian languages. *arXiv preprint arXiv:2103.10730*.

Guillaume Lample, Alexis Conneau, Ludovic Denoyer, and Marc'Aurelio Ranzato. 2017. Unsupervised machine translation using monolingual corpora only. *arXiv preprint arXiv:1711.00043*.

Rémi Lebret, David Grangier, and Michael Auli. 2016. Neural text generation from structured data with application to the biography domain. *arXiv preprint arXiv:1603.07771*.

Manuel Mager, Ramón Fernandez Astudillo, Tahira Naseem, Md Arafat Sultan, Young-Suk Lee, Radu Florian, and Salim Roukos. 2020. GPT-too: A language-model-first approach for AMR-to-text generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1846–1852, Online. Association for Computational Linguistics.

Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven J. Bethard, and David McClosky. 2014. The Stanford CoreNLP natural language processing toolkit. In *Association for Computational Linguistics (ACL) System Demonstrations*, pages 55–60.

Preksha Nema, Shreyas Shetty, Parag Jain, Anirban Laha, Karthik Sankaranarayanan, and Mitesh M Khapra. 2018. Generating descriptions from structured data using a bifocal attention mechanism and gated orthogonalization. *arXiv preprint arXiv:1804.07789*.

Ankur P Parikh, Xuezhi Wang, Sebastian Gehrmann, Manaal Faruqui, Bhuwan Dhingra, Diyi Yang, and Dipanjan Das. 2020. Totto: A controlled table-to-text generation dataset. *arXiv preprint arXiv:2004.14373*.

Raheel Qader, Khoder Jneid, François Portet, and Cyril Labbé. 2018. Generation of company descriptions using concept-to-text and text-to-text deep models: dataset collection and systems evaluation. In *Proceedings of the 11th International Conference on Natural Language Generation*, pages 254–263. Association for Computational Linguistics.

Anastasia Shimorina, Elena Khasanova, and Claire Gardent. 2019. Creating a corpus for Russian data-to-text generation using neural machine translation and post-editing. In *Proceedings of the 7th Workshop on Balto-Slavic Natural Language Processing*, pages 44–49, Florence, Italy. Association for Computational Linguistics.

Denny Vrandečić and Markus Krötzsch. 2014. Wikidata: a free collaborative knowledgebase. *Communications of the ACM*, 57(10):78–85.

Binh Vu, Craig A Knoblock, Pedro Szekely, Minh Pham, and Jay Pujara. 2021. A graph-based approach for inferring semantic descriptions of wikipedia tables. In *International Semantic Web Conference*, pages 304–320. Springer.

Ralph M. Weischedel, Eduard H. Hovy, Mitchell P. Marcus, and Martha Palmer. 2017. Ontonotes : A large training corpus for enhanced processing.

Tsung-Hsien Wen, Milica Gasic, Nikola Mrksic, Lina M Rojas-Barahona, Pei-Hao Su, David Vandyke, and Steve Young. 2016. Multi-domain neural network language generation for spoken dialogue systems. *arXiv preprint arXiv:1603.01232*.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, 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.

Yinfei Yang, Daniel Cer, Amin Ahmad, Mandy Guo, Jax Law, Noah Constant, Gustavo Hernandez Abrego, Steve Yuan, Chris Tar, Yun-Hsuan Sung, et al. 2019a. Multilingual universal sentence encoder for semantic retrieval. *arXiv preprint arXiv:1907.04307*.

Yinfei Yang, Yuan Zhang, Chris Tar, and Jason Baldridge. 2019b. PAWS-X: A Cross-lingual Adversarial Dataset for Paraphrase Identification. In *Proc. of EMNLP*.