3.1 Neural Semantic Parsing

We approach our semantic parsing task using a SEQ2SEQ architecture *Transformer* encoder-decoder network (Vaswani et al., 2017). The encoder computes a contextual representation for each input token through *multi-head self-attention* by combining parallel dot-product attention weightings, or "heads", over the input sequence. The decoder repeats this self-attention across the output sequence and incorporates the source sequence through multi-head attention over the encoder output. A Transformer layer maps input $X = \{x_i\}_{i=0}^{N}$, where $x_i \in \mathbb{R}^{d_x}$, to output $Y = \{y_i\}_{i=0}^{N}$ using attention components of Query $\mathbf{Q}$, Key $\mathbf{K}$ and Value $\mathbf{V}$ in H attention heads:

$$\mathbf{e}_i^{(h)} = \frac{\mathbf{Q}W_Q^{(h)}\left(\mathbf{K}W_K^{(h)}\right)^T}{\sqrt{d_x/H}};\ \mathbf{s}_i^{(h)} = \text{softmax}\left(\mathbf{e}_i^{(h)}\right) \quad (1)$$

$$\mathbf{z}_i^{(h)} = \mathbf{s}_i^{(h)}\left(\mathbf{V}W_V^{(h)}\right);\ \ \mathbf{z}_i = \text{concat}\{\mathbf{z}_i^{(h)}\}_{h=1}^{H} \quad (2)$$

$$\hat{\mathbf{y}}_i = \text{LayerNorm}\left(X + \mathbf{z}_i\right) \quad (3)$$

$$\mathbf{y}_i = \text{LayerNorm}\left(\hat{\mathbf{y}}_i + \text{FC}\left(\text{ReLU}\left(\text{FC}\left(\hat{\mathbf{y}}_i\right)\right)\right)\right) \quad (4)$$

Following Wang et al. (2019), Equation 1 describes attention scores between Query (Q) and Key (K), $\mathbf{z}_i^h$ is the h^{th} attention head, applying scores $\mathbf{s}_i^{(h)}$ to value (V) into the multi-head attention function $\mathbf{z}_i$ with $W_{\{Q,K,V\}}^{(h)} \in \mathbb{R}^{d_x \times (d_x/H)}$. Output prediction $\mathbf{y}_i$ combines $\mathbf{z}_i$ with a residual connection and two fully-connected (FC) layers, ReLU nonlinearity, and layer normalization (Ba et al., 2016). The encoder computes self-attention through query, key, and value all equal to the input, $\{\mathbf{Q},\mathbf{K},\mathbf{V}\} = X$. Decoder layers use self-attention over output sequence, $\{\mathbf{Q},\mathbf{K},\mathbf{V}\} = Y_{out}$, followed by attention over the encoder output E ($\mathbf{Q} = Y_{out}$ and $\{\mathbf{K},\mathbf{V}\} = E$) to incorporate the input encoding into decoding.

3.2 Crosslingual Modeling

Consider a parser, $\text{SP}(x)$, which transforms utterances in language x_{L}, to some executable logical form, y. We express a dataset in some language L as $\mathcal{D}^{\text{L}} = \left(\{x_n^{\text{L}}, y_n, d_n\}_{n=1}^{N}, KB\right)$, for N examples where x^{L} is an utterance in language L, y is the corresponding logical form and d is a denotation from knowledge base, $d = KB(y)$. The MT approximation of language L is described as J; using MT from English, $x^J = \text{MT}\left(x^{\text{EN}}\right)$. Our hypothesis is that $J \approx \text{L}$ such that prediction $\hat{y} = \text{SP}\left(x^{\text{L}}\right)$ for test example x^{L} approaches gold logical form, y_{gold},

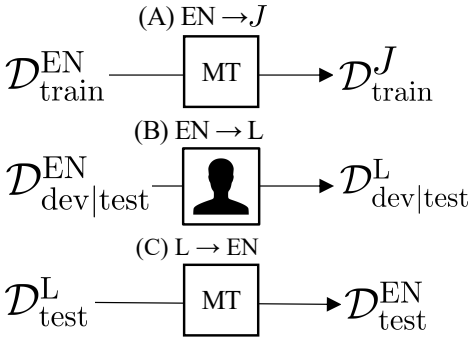

Figure 1: (A) Machine Translation (MT) from English into some language, L, for training data. J is the MT approximation of this language to be parsed. (B) Human translation of the development and test sets from English into language L. (C) Translation from language L into English using MT. Any system parsing language L must perform above this "back-translation" baseline to justify development.

conditioned upon the quality of MT. An ideal parser will output non-spurious prediction, $\hat{y}$, executing to return an equal denotation to $KB\left(y_{\text{gold}}\right) = d_{\text{gold}}$. The proportion of predicted queries which retrieve the correct denotation defines the *denotation accuracy*. Generalization performance is always measured on real queries from native speakers e.g. $\mathcal{D}^J = \{\mathcal{D}^J_{\text{train}}, \mathcal{D}^{\text{L}}_{\text{dev}}, \mathcal{D}^{\text{L}}_{\text{test}}\}$ and $\mathcal{D}^J_{\text{dev|test}} = \emptyset$.

We evaluate parsing on two languages to compare transfer learning from English into varied locales. We investigate German, a similar Germanic language, and Mandarin Chinese, a dissimilar Sino-Tibetan language, due to the purported quality of existing MT systems (Wu et al., 2016) and availability of native speakers to verify or rewrite crowdsourced annotation. Similar to Conneau et al. (2018), we implement a "back-translate into English" baseline wherein the test set in ZH/DE is machine translated into English and a semantic parser trained on the source English dataset predicts logical forms. Figure 1 indicates how each dataset is generated. To maintain a commercial motivation for developing an in-language parser, any proposed system must perform above this baseline. Note that we do not claim to be investigating semantic parsing for low-resource languages since, by virtue, we require adequate MT into each language of interest. We use Google Translate (Wu et al., 2016) as our primary MT system and complement this with systems from other global providers. The selection and use of MT is further discussed in Appendix C.

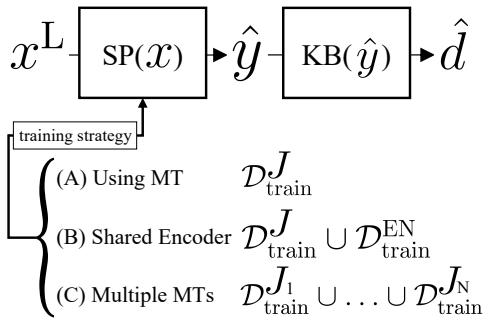

Figure 2: The semantic parser (SP) predicts a logical form, $\hat{y}$, from an utterance in language L, x^{L}. A knowledge base (KB) executes the logical form to predict a denotation, $\hat{d}$. Approaches to crosslingual modeling involve: (A) using machine translation (MT) to approximate training data in language L; (B) training SP on both MT data and source English data; (C) using multiple MT systems to improve the approximation of L.

3.3 Feature Augmentation

Beyond using MT for in-language training data, we now describe our approach to further improve parsing using external resources and transfer learning. These approaches are described in Figure 2.

Pre-trained Representations Motivated by the success of contextual word representations for semantic parsing of English by Shaw et al. (2019), we extend this technique to Chinese and German using implementations of BERT from Wolf et al. (2019). Rather than learning embeddings for the source language *tabula rasa*, we experiment with using pre-trained 768-dimensional inputs from BERT-base in English, Chinese and German[2], as well as the multilingual model trained on 104 languages. To account for rare entities which may be absent from pre-trained vocabularies, we append these representations to learnable embeddings. Representations for logical form tokens are trained from a random initialisation, as we lack a BERT-style pre-trained model for meaning representations (i.e., λ−DCS or SQL queries). Early experiments considering multilingual word representations (Conneau et al., 2017; Song et al., 2018) yielded no significant improvement and these results are omitted for brevity.

Multilingual "Shared" Encoder Following Duong et al. (2017) and Susanto and Lu (2017a), we experiment with an encoder trained with batches from multiple languages as input. Errors in the MT data are purportedly mitigated through the model observing an equivalent English utterance for the same logical form. The joint training dataset is described as $\mathcal{D}^{\text{EN}+J}_{\text{train}} = \mathcal{D}^{\text{EN}}_{\text{train}} \cup \mathcal{D}^{J}_{\text{train}}$ for $J = \{\text{ZH}, \text{DE}\}$. Consistent with Section 3.2, we measure validation and test performance using only utterances from native speakers, $\mathcal{D}^{\text{L}}_{\text{dev|test}}$, and ignore performance for English. This is similar to the `All` model from Duong et al. (2017), however, our objective is biased to maximize performance on one language rather than a balanced multilingual objective.

Machine Translation as Paraphrasing Paraphrasing is a common augmentation for semantic parsers to improve generalization to unseen utterances (Berant and Liang, 2014; Dong et al., 2017; Iyer et al., 2017; Su and Yan, 2017; Utama et al., 2018). While there has been some study of multilingual paraphrase systems (Ganitkevitch and Callison-Burch, 2014), we instead use MT as a paraphrase resource, similar to Mallinson et al. (2017). Each MT system will have have different outputs from different language models and therefore we hypothesize that an ensemble of multiple systems, $(J_1, \ldots J_N)$, will provide greater linguistic diversity to better approximate L. Whereas prior work uses back-translation or beam search, a developer in our scenario lacks the resources to train a NMT system for such techniques. As a shortcut, we input the same English sentence into m public APIs for MT to retrieve a set of candidate paraphrases in the language of interest (we use three APIs in experiments).

We experiment with two approaches to utilising these pseudo-paraphrases. The first, MT-Paraphrase, aims to learn a single, robust language model for L by uniformly sampling one paraphrase from $(J_1, \ldots J_N)$ as input to the model during each epoch of training. The second approach, MT-Ensemble, is an ensemble architecture similar to Garmash and Monz (2016) and Firat et al. (2016) combining attention over each paraphrase in a single decoder. For N paraphrases, we train N parallel encoder models, $\{e_n\}_{n=1}^{N}$, and ensemble across each paraphrase by combining N sets of encoder-decoder attention heads. For each encoder output, $E_n = e_n(X_n)$, we compute multi-head attention, $\mathbf{z}_i$ in Equation 2, with the decoder state, D, as the query and E_n as the key and value (Equation 5). Attention heads are combined through a combination function (Equation 6) and output $\mathbf{m}_{i\epsilon}$ replaces $\mathbf{z}_i$ in Equation 3.

[2] `deepset.ai/german-bert`

We compare ensemble strategies using two combination functions: the mean of heads (Equation 7a) and a gating network (Garmash and Monz 2016; Equation 7b) with gating function $\mathbf{g}$ (Equation 8) where $W_{\mathbf{g}} \in R^{N\times|V|}, W_h \in R^{|V|\times N|V|}$. We experimentally found the gating approach to be superior and we report results using only this method.

$$\mathbf{m}_n = \text{MultiHeadAttention}(D, E_n, E_n) \quad (5)$$

$$\mathbf{m}_{i\varepsilon} = \text{comb}(\mathbf{m}_1, \ldots \mathbf{m}_N) \quad (6)$$

$$\text{comb} = \begin{cases} \frac{1}{N}\sum_n^N \mathbf{m}_n & \text{(a)} \\ \sum_n^N \mathbf{g}_n \mathbf{m}_n & \text{(b)} \end{cases} \quad (7)$$

$$\mathbf{g} = \text{softmax}(W_{\mathbf{g}}\tanh(W_h[\mathbf{m}_n, \ldots \mathbf{m}_N])) \quad (8)$$

Each expert submodel uses a shared embedding space to exploit similarity between paraphrases. During training, each encoder learns a language model specific to an individual MT source, yielding diversity among experts in the final system. However, in order to improve robustness of each encoder to translation variability, inputs to each encoder are shuffled by some tuned probability p_{shuffle}. During prediction, the test utterance is input to all N models in parallel. In initial experiments, we found negligible difference in MT-Paraphrase using random sampling or round-robin selection of each paraphrase. Therefore, we assume that both methods use all available paraphrases over training. Our two approaches differ in that MT-Paraphrase uses all paraphrases sequentially whereas MT-Ensemble uses paraphrases in parallel. Previous LSTM-based ensemble approaches propose training full parallel networks and ensemble at the final decoding step. However, we found this was too expensive given the non-recurrent Transformer model. Our hybrid mechanism permits the decoder to attend to every paraphrased input and maintains a tractable model size with a single decoder.

4 Data

We consider two datasets in this work. Firstly, we evaluate our hypothesis that MT is an adequate proxy for "real" utterances using ATIS (Dahl et al., 1994). This *single-domain* dataset contains 5,418 utterances paired with SQL queries pertaining to a US flights database. ATIS was previously translated into Chinese by Susanto and Lu (2017a) for semantic parsing into λ-calculus, whereas we present these Chinese utterances aligned with SQL queries from Iyer et al. (2017). In addition, we translate ATIS into German following the methodology described below. We use the split of 4,473/497/448 examples for train/validation/test from Kwiatkowski et al. (2011).

We also examine the *multi-domain* Overnight dataset (Wang et al., 2015), which contains 13,682 English questions paired with λ−DCS logical forms executable in SEMPRE (Berant et al., 2013). Overnight is 2.5× larger than ATIS, so a complete translation of this dataset would be uneconomical for our case study. As a compromise, we collect human translations in German and Chinese only for the test and validation partitions of Overnight. We argue that having access to limited translation data better represents the crosslingual transfer required in localizing a parser. We define a fixed development partition of a stratified 20% of the training set for a final split of 8,754/2,188/2,740 for training/validation/testing. Note we consider only Simplified Mandarin Chinese for both datasets.

Crowdsourcing Translations The ATIS and Overnight datasets were translated to German and Chinese using Amazon Mechanical Turk, following best practices in related work (Callison-Burch, 2009; Zaidan and Callison-Burch, 2011; Behnke et al., 2018; Sosoni et al., 2018).

We initially collected three translations per source sentence. Submissions were restricted to Turkers from Germany, Austria, and Switzerland for German and China, USA, or Singapore for Chinese. Our AMT interface barred empty submissions and copying or pasting anywhere within the page. Any attempts to bypass these controls triggered a warning message that using MT is prohibited. Submissions were rejected if they were $>80\%$ similar (by BLEU) to references from Google Translate (Wu et al., 2016), as were nonsensical or irrelevant submissions.

In a second stage, workers cross-checked translations by rating the best translation from each candidate set, including an MT reference, with a rewrite option if no candidate was satisfactory. We collected three judgements per set to extract the best candidate translation. Turkers unanimously agreed on a single candidate in 87.8% of the time (across datasets). Finally, as a third quality filter, we recruited bilingual native speakers to verify, rewrite, and break ties between all top candidates. Annota-