Machine Translation for Multilingual Intent Detection and Slots Filling

Maxime De Bruyn, Ehsan Lotfi, Jeska Buhmann, Walter Daelemans
CLiPS Research Center
University of Antwerp, Belgium
maxime.debruyn@uantwerpen.be

Abstract

We expect to interact with home assistants irrespective of our language. However, scaling the Natural Language Understanding pipeline to multiple languages while keeping the same level of accuracy remains a challenge. In this work, we leverage the inherent multilingual aspect of translation models for the task of multilingual intent classification and slot filling. Our experiments reveal that they work equally well with general-purpose multilingual text-to-text models. Furthermore, their accuracy can be further improved by artificially increasing the size of the training set. Unfortunately, increasing the training set also increases the overlap with the test set, leading to overestimating their true capabilities. As a result, we propose two new evaluation methods capable of accounting for an overlap between the training and test set.

1 Introduction

Home assistants are omnipresent in everyday life. We expect to have an assistant at our disposal at any time using our phone, watch, or car — irrespective of our language.

Scaling home assistants to multiple languages brings additional challenges to NLU and ASR components. There are two options: a single model per language or a shared model for all languages. A single model per language works well for resource-rich languages such as English. However, lower resource languages benefit from the cross-lingual knowledge transfer of a single model dealing with all languages (Conneau et al., 2020). This trade-off applies to any multilingual system (Zhang et al., 2022; De Bruyn et al., 2021).

While multilingual intent classification and slot filling datasets exist, their language coverage is limited, except for MASSIVE (FitzGerald et al., 2022), a new dataset focused on multilingual intent detection and slot filling. The authors translated and localized an English-only dataset in 50 topologically diverse languages. MASSIVE provides a good base to scale existing intent detection and slot filling methods to multiple languages.

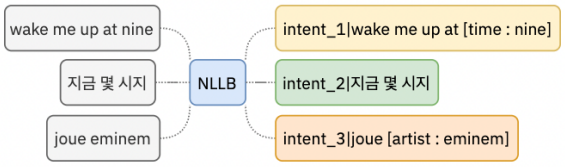

Figure 1: Illustration of our method. We repurpose a translation model for the task of multilingual intent classification and slot filling. We translate from utterances into annotated utterances.

The traditional way to tackle multilingual intents detection and slot filling is to use multilingual models such as XLM-R (Conneau et al., 2020), or mT5 (Xue et al., 2021). These models are similar to their monolingual counterparts (Liu et al., 2019; Raffel et al., 2020) except for the multilingual data used to train them.[1] This approach has been shown to work in multiple studies (FitzGerald et al., 2022; Li et al., 2021). However, MASSIVE has an additional overlooked aspect: utterances are direct translations of one another.

In this work, we approach the task of intent classification and slot filling as a translation task: we translate the original utterance into the annotated utterance. For example, we translate the utterance `what is the temperature in new york?` into the annotated utterance `weather_query|what is the [weather_descriptor : temperature] in [place_name : new york]`.[2]

The typical use of translation models for intent detection and slot filling is to augment the size of an existing dataset (Zheng et al., 2021; Nicosia et al., 2021). However, we believe the inherent multilingual capabilities of these models make them excellent candidates for multilingual intent detec-

[1] They also have larger vocabularies and may have special training tricks for cross-lingual training.

[2] We prepend the slot annotated utterance with the intent.

Proceedings of the Massively Multilingual Natural Language Understanding Workshop (MMNLU-22), pages 69 - 82
December 7, 2022 ©2022 Association for Computational Linguistics

tion and slot filling.

To this end, we leverage the recently released translation model *No Language Left Behind* (NLLB) (NLLB Team et al., 2022) capable of translating between 202 pairs of languages simultaneously using a shared encoder-decoder. We anticipate that the wide range of languages covered by the model will help us deal with lower resources languages present in the MASSIVE dataset.

Better modeling is only half the story. Using more data also helps improve performance. For example, although the MASSIVE dataset displays a large training set of more than 500K training examples, the seed data is only around 10K training examples. Therefore, we used GPT-3 (Brown et al., 2020) to generate additional training data using a dual-model approach. We also leveraged a dataset close to the seed dataset of MASSIVE. As a result, after translating our new training examples to the 50 remaining languages, our training set contains more than 2M training examples — 4x the size of the original training set.

Our experiments reveal that translation models such as NLLB are a good fit for intent classification and slot filling. However, their performance sharply drops in languages that do not use spaces because of tokenization issues.

Unfortunately, the additional training data significantly overlaps with the MASSIVE test set. As a result, we propose two methods capable of dealing with overlaps: weighted exact match and logistic regression.

We conclude this introduction by summarizing our contributions:

- We showed that a translation model such as NLLB can complete the task of intent classification and slot filling
- We demonstrated a method to improve the training data with GPT-3
- We proposed two new evaluation methods taking the training/test set overlap into account

We release our model[3], utterance translation model[4], and generated data[5] on the HuggingFace hub.

[3] maximedb/nllb_massive
[4] maximedb/massive_en_translation
[5] maximedb/massive_generated

2 Related Work

The problem of multilingual intent detection and slot filling is not new. (Razumovskaia et al., 2022) provides an excellent introduction to the subject. We divide our related work section into three parts. We start by reviewing the general problem of task-oriented semantic parsing (i.e., intent detection and slot filling). Next, we review the models commonly used, and lastly, we review the available multilingual datasets.

2.1 Task Oriented Semantic Parsing

Natural Language Understanding (NLU) systems aim to classify an utterance into a predefined set of intents and label the sequence with a predefined ontology of slots (McTear, 2020). Since the release of the ATIS dataset (Price, 1990), this problem has been studied in numerous previous work (Mesnil et al., 2013; Liu and Lane, 2016; Zhu and Yu, 2017). However, it has recently been shown that the flat structure of sequence labeling falls short when a user issues sub-queries, or compositional queries, e.g., `set up a reminder to message mike tonight`[6] Gupta et al. (2018) solves that problem by using hierarchical representations instead.

2.2 Translation Models

Previous work tackling multilingual intent detection and slot filling uses multilingual versions of well-known Transformers such XLM-Roberta (Conneau et al., 2020), mT5 (Xue et al., 2021), or mBART (Liu et al., 2020). We diverge from existing research and use machine translation models instead. (Fan et al., 2021) released M2M100, a model capable of translating between pairs of 100 languages using a single shared encoder-decoder model. Instead of mainly going from and to English, the authors use a dataset that covers thousands of language pairs. M2M100 was later improved by the release of No Language Left Behind (NLLB) (NLLB Team et al., 2022), which follows the same architecture as M2M100 but covers 202 languages.

2.3 Cross-Lingual Task Oriented Semantic Parsing

Although the initial dataset for intent classification and slot filling targeted English, the number of non-English datasets is growing rapidly. Non-English

[6] Two intents compose that query: create a reminder and send a message to mike.

datasets fall into two broad categories: non-English monolingual datasets (Meurs et al., 2008; Castellucci et al., 2019; Bellomaria et al., 2019; Zhang et al., 2017; Gong et al., 2019; He et al., 2013; Dao et al., 2021) and multilingual datasets. As we aim to study models capable of handling multiple languages simultaneously, we focus on the latter kind of datasets. We will now cover the existing multilingual datasets in greater detail. Upadhyay et al. (2018) translated an existing English dataset (Price, 1990) into Turkish and Hindi, while Susanto and Lu (2017) translated the same dataset in Vietnamese and Chinese. Schuster et al. (2019) released a multilingual dataset for task-oriented dialogues in English, Spanish, and Thai across three domains. (Li et al., 2021) provides MTOP a new aligned task-oriented dataset in six languages. MASSIVE (FitzGerald et al., 2022) is the largest available dataset, covering 51 languages.

3 Data

There exist multiple alternative datasets to study multilingual intent detection and slot filling. However, in this work, we use the largest one available: the MASSIVE dataset.

3.1 MASSIVE

MASSIVE (FitzGerald et al., 2022) is a dataset assembled by translating and localizing an existing English-only dataset in 50 topologically different languages.

English Seed MASSIVE is a translation of the English-centric SLURP dataset (Bastianelli et al., 2020). SLURP is a dataset of non-compositional queries directed at a home assistant. It covers 18 domains, 60 intents, and 55 slots.

Languages The authors of MASSIVE hired professional translators to translate the SLURP dataset into 50 topologically diverse languages from 29 genera. Furthermore, to complicate the task, the translators sometimes localized the queries instead of simply translating them.

3.2 English Data Augmentation

As the seed data of MASSIVE is limited in scale (10K training examples), we used two methods to increase the training set artificially.

3.2.1 Generated Data

Generator We first fine-tune a GPT-3 (Brown et al., 2020) curie (13B) model on the task of generating an English utterance conditional on the given intent. For example, we train the model to generate `wake me up at nine am` given the prompt `alarm_set`.

Parser Next, we fine-tune a second GPT-3 curie model on intent detection and slot filling tasks. Given an utterance, the model must generate the concatenation of the intent and the annotated utterance. For example, given the prompt `what is the temperature in new york?` must generate `weather_query|what is the [weather_descriptor : temperature] in [place_name : new york]`.

Dataset We generate 30,000 utterances, equally distributed amongst the 60 intents. After removing duplicates and examples where the two models do not agree on the intent, we arrive at a final dataset of 22,276 annotated English utterances.

Intent & Slots Distribution Although we generated an equal amount of utterances per intent, removing duplicates skewed the distribution. However, comparing the entropy of both distributions with MASSIVE reveals that our generated dataset is more equally spread amongst the intents but less equally distributed relative to the slots.[7] See Annex A for a detailed analysis and comparison with the MASSIVE dataset.

3.2.2 Synthetic Data

The SLURP dataset provides a synthetic dataset.[8] It is not part of the official training set, but as it shares the same ontology as MASSIVE, it provides an excellent extension to our training set. We compare the intent and slot distribution with MASSIVE in Annex A.

3.3 Non-English Data Augmentation

We explained in Section 3.2 our method to artificially increase the size of the (English) training set. This section reviews our method to scale this silver training set to the 50 remaining languages in the MASSIVE dataset.

Using commercial translation systems was not an option as this requires aligning the slots in the translated utterances — a complicated task. Instead, we fine-tune a translation model, NLLB (3B), on the task of translating *annotated* utterances directly.

[7] Our generated dataset has an intent distribution entropy of 4.02 and a slot distribution entropy of 3.10 compared to 3.75 and 3.21 for MASSIVE.

[8] https://github.com/pswietojanski/slurp/tree/master/dataset/slurp

Using this method, we translate annotated utterances and reconstruct the utterances by removing the slot annotations from the text. Our translation model is available on the HuggingFace Hub.[9]

4 Model

This work uses a machine translation model for intent detection and slot filling. No Language Left Behind (NLLB) (NLLB Team et al., 2022) is a model specifically targeted at translating between 202 languages using a single encoder-decoder model based on the M2M100 architecture (Fan et al., 2021). It can translate text in 40,602 different directions.

Data NLLB uses FLORES-200 as training data, an extension of FLORES-100 (Goyal et al., 2022). The authors of FLORES-200 used LASER3 (Heffernan et al., 2022) to mine parallel data from the web, resulting in 1.1 billion sentence pairs.

Tokenization NLLB uses a sentencepiece tokenizer (Kudo and Richardson, 2018) with a vocabulary size of 256,000. To ensure low-resource languages are well-represented in the vocabulary, the authors downsample high-resource and upsample low-resource languages.

Architecture NLLB's architecture is based on the Transformer encoder-decoder (Vaswani et al., 2017). NLLB is trained on several translation directions at once, utilizing the same shared model capacity. This architecture can lead to beneficial cross-lingual transfer between related languages at the risk of increasing interference between unrelated languages. The authors also present a Sparsely Gated Mixture of Experts (MoE) (Almahairi et al., 2016; Bengio et al., 2013). However, we did not experiment with this variant.

Distillation The authors distilled a 54 billion parameter model using MoE into smaller dense models of 1.3 billion and 615 million parameters using online distillation (Hinton et al., 2015). The student model is trained on the training data but with an additional objective: to minimize the cross-entropy to the word-level distribution of the teacher model. We use the distilled 615M parameter model as the base model for intent classification and slot filling.

[9]For anonymity reasons, we will release the URL upon acceptance of this paper.

5 Experiments

This section describes our experiments in applying NLLB to the task of intent classification and slot filling. NLLB is a translation model. While we could repurpose NLLB to the task of intent classification and slot filling directly, we choose to first pre-train it on a translation task.

5.1 Pre-training

As NLLB is, at its core, a translation model, we start by teaching it to translate between the aligned pairs of the MASSIVE dataset. Instead of translating between the utterances of two languages, we translate between the utterance and the annotated utterance. For example, the model must translate "tell me the time in moscow," to the French annotated utterance `datetime_query|donne moi l'heure à [place_name: moscou]`. We take special care in avoiding localized utterances, as this would confuse the model. For example, we avoid predicting `datetime_query|donne moi l'heure à ~~moscou~~ bordeaux.`

5.2 Fine-tuning

In a second step, we fine-tune the model on the task of translating between the utterance and the annotated utterance in the same language. For example, we translate the utterance `what is the temperature in new york?` into the annotated utterance `weather_query|what is the [weather_descriptor : temperature] in [place_name : new york]`.

5.3 Technical Details

We use the NLLB-200 (600M) model for all experiments.[10] We wrap each encoder input according to the following formula: `<s>...</><language_code>`. We prepend each decoder input with the target language code. We train for 50,000 steps during pre-training and fine-tuning with a learning rate of $1e{-}4$ and $1e{-}5$, respectively. We use Pytorch (Paszke et al., 2019), the HuggingFace Trainer (Wolf et al., 2020) and DeepSpeed (Rajbhandari et al., 2020).

6 Results

This section presents a high-level analysis of our results. Table 1 compares our results against the baselines provided by the authors of MASSIVE.

[10]facebook/nllb-200-distilled-600M

Model	Training Set	Intent Acc (%)			Slot F1 (%)			Exact Match (%)		
		High	Low	Avg	High	Low	Avg	High	Low	Avg
XLM-R	M	88.3	77.2	85.1	83.5	63.3	73.6	70.1	55.8	63.7
mT5 Enc.	M	89.0	79.1	86.1	85.7	64.5	75.4	72.3	57.8	65.9
mT5	M	87.9	79.0	85.3	**86.8**	67.6	76.8	73.4	58.3	66.6
NLLB	M+G	89.3	79.2	87.3	85.9	66.3	77.0	74.1	57.8	68.3
NLLB	M+G+S	**94.5**	**84.5**	**93.4**	82.9	**69.6**	**82.9**	**89.2**	**65.0**	**78.5**

Table 1: Modelling results on the MASSIVE test set. NLLB trained on the MASSIVE training set (M), our generated dataset (G) and the synthetic training set from SLURP (S) achieve the highest scores. However, as we show in a later section, this outperformance is due to a large overlap with the MASSIVE test set.

Our experiments reveal that NLLB performs similarly to mT5 on intent detection and slot filling tasks. Furthermore, our two data augmentation strategies improve the results on the MASSIVE test set. First, training with our generated training set improves the locale average exact match from 66.6 to 68.3. Second, training with the generated and synthetic data boosts the exact match as it improves from 68.3 to 78.5. As we show in the next section, this performance boost is mainly due to a large overlap between the training and test set.

7 Training & Test Set Overlap

This section analyses the similarity between the training sets and the MASSIVE. Next, we look for evaluation methods capable of correcting for the overlap between the training and test set.

Exact Duplicates An analysis of the data reveals problematic overlaps between the training sets and the MASSIVE test set. However, this overlap is unequal across the training sets and languages. Table 2 shows the percentage of examples in the MASSIVE test set, which are also present in our three training sets. The English subset of the MASSIVE test set overlaps highly with the synthetic training set described in Section 3.2.2. Localization and translation somewhat reduce the exact match overlap when looking at all languages, although it remains high. The MASSIVE and generated training sets also have a non-zero overlap with the MASSIVE test set.

Close Duplicates Some examples may not be exact duplicates but close duplicates. For example, `call the dentist` and `olly please call the dentist now`. We use character n-grams to measure the similarity between two utterances as similarity metric between two utterances. We search for the most similar training example for each example in the test and record their n-gram similarity.[11] Figure 2 shows the distribution of maximum similarity between the test set and our three training sets for the English subset and across all locales. It is clear from Figure 2 that the English synthetic dataset overlaps significantly with the English MASSIVE test set. However, as for the exact duplicates, the translation and localization process reduces this overlap but does not eliminate it.

Training Set	en-US (%)	All Locales (%)
MASSIVE	0.7	5.9
Generated	5.6	6.4
Synthetic	49.0	12.8

Table 2: Exact duplicate analysis. Percentage of examples in the MASSIVE test set, which are also present in the training set of MASSIVE, our generated training set, and the synthetic training set. Translation reduces the overlap of the synthetic dataset compared to the English-only figures. However, it is the opposite for the MASSIVE test set, where the overlap is higher for all locales compared to English only.

A naive solution would be to remove training examples that overlap with the test set. However, how does one decide what is a close duplicate? Furthermore, as the training set grows, some overlap with the test is inevitable. We argue that the problem is not the training data but the evaluation metric. We need an evaluation metric capable of controlling for the overlap between the test and training sets.

7.1 Logistic Regression

Instead of looking at the simple exact match accuracy, we want to express the exact match accuracy as a function of the test/train similarity. One potential solution is to use logistic regression with similarity as the independent variable and exact match as the dependent variable.

[11] We do this search on a per-language basis.

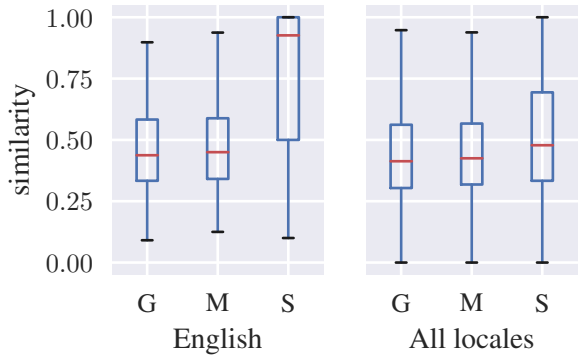

Figure 2: Box plot of the maximum similarity between examples in the MASSIVE test set with the training set of MASSIVE (M), Generated (G) and Synthetic (S), for the English part and the entire dataset (all locales). The English synthetic (S) training set overlaps highly with the MASSIVE test set. Translation and localization reduces this overlap in the all-locales dataset.

Training S.	β_0	β_1	R^2
M+G	-0.96±0.03	3.31±0.06	0.07
M+G+S	-0.69±0.03	3.14±0.06	0.08

Table 3: We report the logistic regression results for two NLLB models fine-tuned on the training set of MASSIVE (M), generated (G), and synthetic (S). We report the point estimate and the 95% confidence interval for each parameter. After correcting for any overlap between the training and test set, the second is statistically better than the first.

$$p(x) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 x)}} \quad (1)$$

Where $p(x)$ represents the probability of an exact match, β_0 represents the intercept and β_1 the slope. Using this method, we can compare both models at the same level of similarity.

Results Table 3 presents a summary of the logistic regression results. We report the point estimate and confidence interval for both β_0, β_1 and the pseudo R^2 given by `statsmodels` (Seabold and Perktold, 2010). Using Equation 1, we can estimate the performance of both models at multiple levels of similarity, as shown in Figure 3.

According to Table 3 and Figure 3, the model trained on the three training datasets is better than the one trained only on two — taking the overlap into account. However, these numbers also indicate that both models struggle with utterances dissimilar to the training set. Moreover, they achieve an exact match accuracy lower than random chance on dissimilar utterances — casting doubt on their abilities to generalize to unseen utterances.

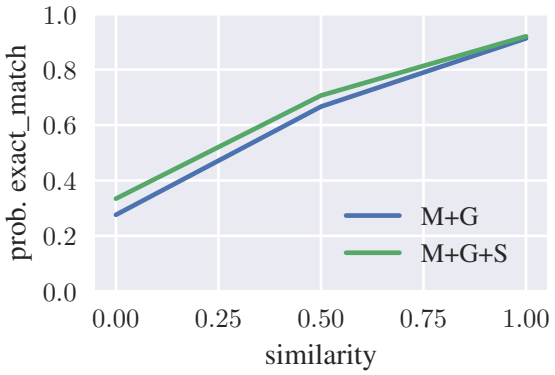

Figure 3: Exact match probability at three levels of similarity: 0, 0.5, and 1.0. We used Equation 1 with the estimated parameters from Table 3. Model two is better than model one on dissimilar utterances. However, the difference diminishes when the similarity increases.

Training Set	**Weighted Average (%)**
M+G	59.2
M+G+S	67.2

Table 4: We report the weighted average results for two NLLB models fine-tuned on the training set of MASSIVE (M), generated (G), and synthetic (S). The second model is better than the first even after correcting for its high overlap with the training set.

7.2 Weighted Average

Another possibility is to give less importance to test examples similar to the training set.

$$\sum_{i=1}^{n} \frac{w_i * exact\_match_i}{\sum_{i=1}^{n} w_i} \quad (2)$$

where $w_i = 1 - sim_i$.

Results Table 4 displays the results according to the weighted average metric. According to this metric, the second model outperforms the first one. This metric is easy to understand. However, it does not tell us anything about the performance of dissimilar queries.

7.3 Summary

According to our overlap-aware evaluation metrics, the model trained on the synthetic datasets is the most performant, even after correcting for its high overlap with the test.

Language	Intercept	Num. Token Split	R-Squared
ja-JP	0.85*	-0.16*	0.013
zh-CN	0.58*	-0.15*	0.006
zh-TW	0.11	-0.03	0.000

Table 5: Logistic regression of exact match accuracy explained by the number of split token. The number of split token negatively influence the capability of the token to correctly parse the slots for ja-JA and zh-CN. The coefficient are not significantly different than zero for zh-TW. Starred numbers (*) are statistically different than zero with a p-value of 0.05

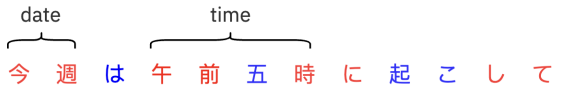

Figure 4: Our method does not scale well to non-space delimited languages. For example, in the utterance above, the time slot ends in the middle of a token. To correctly parse the utterance, the model must replace token 20202 (時に) by tokens 249229 (時) and 5954 (に).

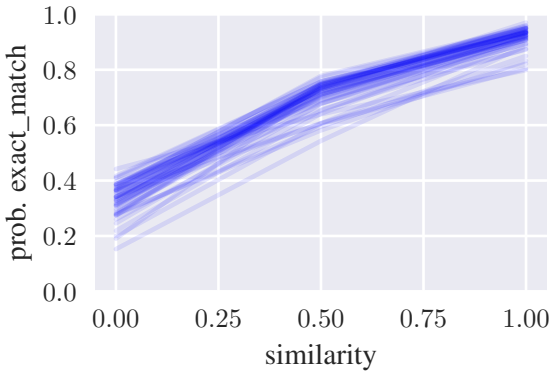

Figure 5: Exact match probability at three levels of similarity: 0, 0.5, and 1.0. We used Equation 1 with the estimated parameters from Table 3. Model two is better than model one on dissimilar utterances. However, the difference diminishes when the similarity increases.

8 Error Analysis

8.1 Tokenization

Our formatting of input and output consists of surrounding slots with brackets along with the slot name (e.g., `[place_name : new york]`. This method implies that slots' boundaries align with tokenization. Otherwise, the model cannot correctly place the opening or closing bracket — unless it uses a different token than the ones in the source utterance. See Figure for an example.

We identified three languages for which this problem occurs: ja-JP in 66% of the test set, zh-CN in 66% of the test set, and zh-TW in 69% of the test set. These are three languages that do not use spaces between words.

Similar to Section 7.1, we ran a logistic regression to explain the exact match performance by the number of split tokens. Table 5 shows the results. We identified a statistically significant relationship between the number of split tokens and the exact match performance for ja-JP and zh-CN. The performance of zh-TW is low regardless of the number of split tokens.

8.2 Generalization

Section 7.1 demonstrated that models struggle to generalize to utterances dissimilar to the training set. In this section, we decompose this conclusion by languages. Figure 5 decomposes Figure 3 by languages. It shows the probability of an exact match on the test set by increasing levels of similarity to the training set. Figure 5 shows a wide distribution of probabilities for low similarity utterances (6% standard deviation), while the distribution for highly similar utterances is more concentrated (3% standard deviation). Some languages do better than others. For example, km-KH achieves an exact match probability of 44% at a similarity of 0.0 while vi-VN only achieves a an exatch match probability of 15%. We list the full details of Figure 5 in Appendix B.

9 Future Work

In this work, we estimated the similarity between two utterances using character n-grams. However, while this captures lexically similar utterances, it fails to capture utterances semantically similar but lexically different. For example, these two utterances are highly similar, although they only share a single common token: `what time is it?` and `tell me the time`. Future work can tackle this by using multilingual sentence encoders such as LASER3 (Heffernan et al., 2022), Multilingual Universal Sentence Encoder (Yang et al., 2020), or multilingual models on Sentence Transformers

(Reimers and Gurevych, 2020).

This work did not explicitly address cross-lingual training and instead relied on the cross-lingual pre-training of the translation model. Future work could combine a translation model with cross-lingual training methods such as xTune (Zheng et al., 2021), or X-Mixup (Yang et al., 2022).

Section 8.1 showed the limitation of subword tokenization methods. Future work could explore methods which do not uses subword tokenization such as byT5 (Xue et al., 2022).

10 Conclusion

In this work, we showed that a translation model such as NLLB can perform the task of intent classification and slot filling. Because of tokenization issues, it is, however, suboptimal with non-spaced languages.

Moreover, we showed that artificially increasing the training sets' size leads to improved performance. Unfortunately, we also show that this added data can overlap with the existing test set, distorting the true evaluation of these models. The normal way to overcome this problem is to remove the overlap from the training set. However, deciding on what constitutes an overlap remains an open question. Therefore, we argued that the data overlap is not the problem — the evaluation metric is. As a result, we proposed two evaluation metrics that control the training/test overlap. Both metrics reveal that the model trained on overlapped data improves the results on non-overlapped data. However, our analysis also reveals that these models struggle to beat random chance when evaluated on utterances dissimilar to the training set.

Acknowledgement

We thank the reviewers for their helpful feedback. This research received funding from the Flemish Government under the *Onderzoeksprogramma Artificiële Intelligentie (AI) Vlaanderen* programme.

References

Amjad Almahairi, Nicolas Ballas, Tim Cooijmans, Yin Zheng, Hugo Larochelle, and Aaron Courville. 2016. Dynamic capacity networks. In *Proceedings of the 33rd International Conference on International Conference on Machine Learning - Volume 48*, ICML'16, page 2091–2100. JMLR.org.

Emanuele Bastianelli, Andrea Vanzo, Pawel Swietojanski, and Verena Rieser. 2020. SLURP: A spoken language understanding resource package. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7252–7262, Online. Association for Computational Linguistics.

Valentina Bellomaria, Giuseppe Castellucci, Andrea Favalli, and Raniero Romagnoli. 2019. Almawave-slu: A new dataset for slu in italian. *arXiv*, abs/1907.07526.

Yoshua Bengio, Nicholas Léonard, and Aaron C. Courville. 2013. Estimating or propagating gradients through stochastic neurons for conditional computation. *CoRR*, abs/1308.3432.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

Giuseppe Castellucci, Valentina Bellomaria, Andrea Favalli, and Raniero Romagnoli. 2019. Multi-lingual intent detection and slot filling in a joint bert-based model. *arXiv*, abs/1907.02884.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.

Mai Hoang Dao, Thinh Hung Truong, and Dat Quoc Nguyen. 2021. Intent detection and slot filling for vietnamese. *arXiv*, abs/2104.02021.

Maxime De Bruyn, Ehsan Lotfi, Jeska Buhmann, and Walter Daelemans. 2021. MFAQ: a multilingual FAQ dataset. In *Proceedings of the 3rd Workshop on Machine Reading for Question Answering*, pages 1–13, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Angela Fan, Shruti Bhosale, Holger Schwenk, Zhiyi Ma, Ahmed El-Kishky, Siddharth Goyal, Mandeep Baines, Onur Celebi, Guillaume Wenzek, Vishrav Chaudhary, Naman Goyal, Tom Birch, Vitaliy Liptchinsky, Sergey Edunov, Edouard Grave, Michael Auli, and Armand Joulin. 2021. Beyond english-centric multilingual machine translation. *J. Mach. Learn. Res.*, 22(1).

Jack FitzGerald, Christopher Hench, Charith Peris, Scott Mackie, Kay Rottmann, Ana Sanchez, Aaron Nash, Liam Urbach, Vishesh Kakarala, Richa Singh, Swetha Ranganath, Laurie Crist, Misha Britan, Wouter Leeuwis, Gokhan Tur, and Prem Natarajan. 2022. Massive: A 1m-example multilingual natural language understanding dataset with 51 typologically-diverse languages. *arXiv*, abs/2204.08582.

Yu Gong, Xusheng Luo, Yu Zhu, Wenwu Ou, Zhao Li, Muhua Zhu, Kenny Q. Zhu, Lu Duan, and Xi Chen. 2019. Deep cascade multi-task learning for slot filling in online shopping assistant. In *Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence and Thirty-First Innovative Applications of Artificial Intelligence Conference and Ninth AAAI Symposium on Educational Advances in Artificial Intelligence*, AAAI'19/IAAI'19/EAAI'19. AAAI Press.

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.

Sonal Gupta, Rushin Shah, Mrinal Mohit, Anuj Kumar, and Mike Lewis. 2018. Semantic parsing for task oriented dialog using hierarchical representations. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2787–2792, Brussels, Belgium. Association for Computational Linguistics.

Xiaodong He, Li Deng, Dilek Hakkani-Tur, and Gokhan Tur. 2013. Multi-style adaptive training for robust cross-lingual spoken language understanding. In *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 8342–8346.

Kevin Heffernan, Onur Çelebi, and Holger Schwenk. 2022. Bitext mining using distilled sentence representations for low-resource languages. *arXiv*, arxiv.2205.12654.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. 2015. Distilling the knowledge in a neural network. *arXiv*, abs/1503.02531.

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.

Haoran Li, Abhinav Arora, Shuohui Chen, Anchit Gupta, Sonal Gupta, and Yashar Mehdad. 2021. MTOP: A comprehensive multilingual task-oriented semantic parsing benchmark. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 2950–2962, Online. Association for Computational Linguistics.

Bing Liu and Ian Lane. 2016. Attention-based recurrent neural network models for joint intent detection and slot filling. *Interspeech 2016*, pages 685–689.

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. 2020. Multilingual denoising pre-training for neural machine translation. *arXiv*, abs/2001.08210.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv*, abs/1907.11692.

Michael McTear. 2020. Conversational ai: dialogue systems, conversational agents, and chatbots. *Synthesis Lectures on Human Language Technologies*, 13(3):1–251.

Grégoire Mesnil, Xiaodong He, Li Deng, and Yoshua Bengio. 2013. Investigation of recurrent-neural-network architectures and learning methods for spoken language understanding. In *INTERSPEECH*.

Marie-Jean Meurs, Frédéric Duvert, Frédéric Béchet, Fabrice Lefèvre, and Renato de Mori. 2008. Semantic frame annotation on the French MEDIA corpus. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*, Marrakech, Morocco. European Language Resources Association (ELRA).

Massimo Nicosia, Zhongdi Qu, and Yasemin Altun. 2021. Translate & Fill: Improving zero-shot multilingual semantic parsing with synthetic data. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 3272–3284, Punta Cana, Dominican Republic. Association for Computational Linguistics.

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*, abs/2207.04672.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca

Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc.

P. J. Price. 1990. Evaluation of spoken language systems: the ATIS domain. In *Speech and Natural Language: Proceedings of a Workshop Held at Hidden Valley, Pennsylvania, June 24-27,1990*.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Samyam Rajbhandari, Jeff Rasley, Olatunji Ruwase, and Yuxiong He. 2020. Zero: Memory optimizations toward training trillion parameter models. In *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, SC '20. IEEE Press.

Evgeniia Razumovskaia, Goran Glavas, Olga Majewska, Edoardo M Ponti, Anna Korhonen, and Ivan Vulic. 2022. Crossing the conversational chasm: A primer on natural language processing for multilingual task-oriented dialogue systems. *Journal of Artificial Intelligence Research*, 74:1351–1402.

Nils Reimers and Iryna Gurevych. 2020. Making monolingual sentence embeddings multilingual using knowledge distillation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4512–4525, Online. Association for Computational Linguistics.

Sebastian Schuster, Sonal Gupta, Rushin Shah, and Mike Lewis. 2019. Cross-lingual transfer learning for multilingual task oriented dialog. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3795–3805, Minneapolis, Minnesota. Association for Computational Linguistics.

Skipper Seabold and Josef Perktold. 2010. statsmodels: Econometric and statistical modeling with python. In *9th Python in Science Conference*.

Raymond Hendy Susanto and Wei Lu. 2017. Neural architectures for multilingual semantic parsing. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 38–44, Vancouver, Canada. Association for Computational Linguistics.

Shyam Upadhyay, Manaal Faruqui, Gokhan Tür, Hakkani-Tür Dilek, and Larry Heck. 2018. (almost) zero-shot cross-lingual spoken language understanding. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6034–6038.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi 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 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.

Linting Xue, Aditya Barua, Noah Constant, Rami Al-Rfou, Sharan Narang, Mihir Kale, Adam Roberts, and Colin Raffel. 2022. ByT5: Towards a token-free future with pre-trained byte-to-byte models. *Transactions of the Association for Computational Linguistics*, 10:291–306.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2021. mT5: A massively multilingual pre-trained text-to-text transformer. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 483–498, Online. Association for Computational Linguistics.

Huiyun Yang, Huadong Chen, Hao Zhou, and Lei Li. 2022. Enhancing cross-lingual transfer by manifold mixup. *arXiv*, abs/2205.04182.

Yinfei Yang, Daniel Cer, Amin Ahmad, Mandy Guo, Jax Law, Noah Constant, Gustavo Hernandez Abrego, Steve Yuan, Chris Tar, Yun-hsuan Sung, Brian Strope, and Ray Kurzweil. 2020. Multilingual universal sentence encoder for semantic retrieval. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 87–94, Online. Association for Computational Linguistics.

Qingyu Zhang, Xiaoyu Shen, Ernie Chang, Jidong Ge, and Pengke Chen. 2022. Mdia: A benchmark for multilingual dialogue generation in 46 languages. *arXiv*, abs/2208.13078.

Weinan Zhang, Zhigang Chen, Wanxiang Che, Guoping Hu, and Ting Liu. 2017. The first evaluation of chinese human-computer dialogue technology. *CoRR*, abs/1709.10217.

Bo Zheng, Li Dong, Shaohan Huang, Wenhui Wang, Zewen Chi, Saksham Singhal, Wanxiang Che, Ting Liu, Xia Song, and Furu Wei. 2021. Consistency regularization for cross-lingual fine-tuning. 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 3403–3417, Online. Association for Computational Linguistics.

Su Zhu and Kai Yu. 2017. Encoder-decoder with focus-mechanism for sequence labelling based spoken language understanding. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5675–5679. IEEE.

A Distribution of Intents & Slots

We list in Table 6 the distribution of intents across the three datasets. Table 7 shows the distribution of slots across the three datasets.

B Logistic Regression by Languages

We list the results of the logistic regression by language in Table 8.

Intent	MASSIVE	Generated	Synthetic
calendar_set	7,0%	2,6%	3,6%
play_music	5,5%	2,2%	3,3%
weather_query	5,0%	2,0%	3,0%
calendar_query	4,9%	2,2%	2,4%
general_quirky	4,8%	1,7%	5,2%
qa_factoid	4,7%	2,0%	8,3%
news_query	4,4%	2,1%	2,5%
email_query	3,6%	2,2%	12,0%
email_sendemail	3,1%	2,4%	11,1%
datetime_query	3,0%	1,5%	1,4%
calendar_remove	2,7%	2,1%	1,1%
play_radio	2,5%	2,1%	1,5%
social_post	2,5%	2,4%	8,7%
qa_definition	2,3%	2,2%	3,7%
transport_query	2,0%	2,3%	1,1%
cooking_recipe	1,8%	2,2%	1,2%
lists_query	1,7%	1,5%	1,0%
play_podcasts	1,7%	1,5%	1,0%
recommendation_events	1,7%	2,0%	0,9%
alarm_set	1,6%	1,8%	0,6%
lists_createoradd	1,5%	1,7%	0,6%
recommendation_locations	1,5%	2,3%	0,9%
lists_remove	1,4%	1,7%	0,9%
music_query	1,3%	1,3%	0,6%
iot_hue_lightoff	1,3%	1,3%	0,6%
qa_stock	1,3%	2,5%	2,7%
play_audiobook	1,3%	2,0%	0,3%
qa_currency	1,2%	2,2%	3,3%
takeaway_order	1,2%	2,1%	0,4%
alarm_query	1,1%	1,3%	0,2%
email_querycontact	1,1%	2,0%	3,3%
transport_ticket	1,1%	1,8%	0,6%
iot_hue_lightchange	1,1%	2,1%	0,7%
iot_coffee	1,1%	1,2%	0,5%
takeaway_query	1,1%	1,8%	0,5%
transport_traffic	1,0%	1,8%	0,4%
music_likeness	1,0%	1,5%	0,5%
play_game	1,0%	1,7%	0,7%
audio_volume_up	1,0%	1,2%	0,1%
audio_volume_mute	1,0%	1,5%	0,3%
social_query	0,9%	2,0%	2,8%
transport_taxi	0,9%	1,9%	0,5%
iot_cleaning	0,8%	1,4%	0,4%
alarm_remove	0,7%	1,8%	0,2%
qa_maths	0,7%	1,7%	0,8%
iot_hue_lightup	0,7%	1,3%	0,4%
iot_hue_lightdim	0,7%	1,4%	0,4%
general_joke	0,6%	1,3%	0,3%
recommendation_movies	0,6%	2,0%	0,4%
email_addcontact	0,5%	1,3%	1,4%
iot_wemo_off	0,5%	0,8%	0,2%
datetime_convert	0,5%	1,6%	0,2%
audio_volume_down	0,5%	1,1%	0,1%
music_settings	0,4%	0,9%	0,2%
iot_wemo_on	0,4%	1,0%	0,2%
general_greet	0,2%	0,2%	
iot_hue_lighton	0,2%	1,0%	0,1%
audio_volume_other	0,2%	0,6%	0,0%
music_dislikeness	0,1%	0,9%	0,1%
cooking_query	0,0%	0,0%	0,0%

Table 6: Distribution of intents across the three datasets. Generated represents the utterances generated by GPT-3, while synthetic represents the synthetic training set of SLURP.

Intent	MASSIVE	Generated	Synthetic
date	16,0%	10,8%	10,7%
place_name	9,6%	10,6%	8,0%
event_name	8,8%	4,3%	5,5%
person	7,6%	5,4%	17,2%
time	7,0%	5,8%	4,1%
media_type	4,2%	5,4%	9,5%
business_name	3,4%	5,7%	7,6%
weather_descriptor	2,8%	1,1%	1,5%
transport_type	2,8%	5,0%	1,2%
food_type	2,6%	4,2%	1,4%
relation	2,2%	2,3%	4,8%
timeofday	2,1%	2,0%	1,3%
artist_name	2,0%	0,8%	1,2%
device_type	2,0%	3,4%	1,1%
definition_word	2,0%	2,0%	3,5%
currency_name	1,9%	3,8%	5,7%
house_place	1,7%	3,8%	0,8%
list_name	1,7%	1,8%	0,9%
business_type	1,7%	2,8%	0,8%
news_topic	1,6%	0,7%	1,1%
music_genre	1,6%	0,9%	1,0%
player_setting	1,4%	2,1%	0,5%
radio_name	1,2%	1,1%	0,9%
song_name	1,1%	0,3%	0,7%
order_type	0,9%	1,6%	0,3%
color_type	0,9%	1,7%	0,4%
game_name	0,8%	1,3%	0,6%
general_frequency	0,7%	0,3%	0,4%
personal_info	0,7%	1,2%	2,0%
audiobook_name	0,6%	0,9%	0,2%
podcast_descriptor	0,6%	0,6%	0,3%
meal_type	0,6%	0,4%	0,4%
playlist_name	0,5%	0,1%	0,3%
podcast_name	0,5%	0,4%	0,3%
time_zone	0,5%	1,1%	0,2%
app_name	0,4%	0,3%	0,1%
change_amount	0,4%	0,9%	0,1%
music_descriptor	0,4%	0,2%	0,2%
joke_type	0,3%	0,8%	0,2%
email_folder	0,3%	0,2%	0,9%
email_address	0,3%	0,4%	1,4%
transport_agency	0,3%	0,5%	0,2%
coffee_type	0,2%	0,2%	0,1%
ingredient	0,2%	0,1%	0,1%
cooking_type	0,1%	0,1%	0,1%
movie_name	0,1%	0,1%	0,1%
movie_type	0,1%	0,2%	0,0%
transport_name	0,1%	0,1%	0,1%
drink_type	0,1%	0,1%	0,0%
alarm_type	0,1%	0,1%	0,0%
transport_descriptor	0,1%	0,0%	0,0%
audiobook_author	0,1%	0,2%	0,0%
sport_type	0,0%	0,0%	0,0%
music_album	0,0%		0,0%
game_type	0,0%	0,0%	0,0%

Table 7: Distribution of slots across the three datasets. Generated represents the utterances generated by GPT-3, while synthetic represents the synthetic training set of SLURP.

language	β_0	β_1	R_2	$f(x=0)$	$f(x=0.5)$	$f(x=1)$
all	-0.69	3.14	0.08	0.33	0.71	0.92
af-ZA	-0.98	4.01	0.11	0.27	0.74	0.95
am-ET	-0.46	3.09	0.06	0.39	0.75	0.93
ar-SA	-0.58	3.01	0.07	0.36	0.72	0.92
az-AZ	-0.55	3.24	0.08	0.37	0.75	0.94
bn-BD	-1.27	3.71	0.10	0.22	0.64	0.92
cy-GB	-0.66	3.37	0.08	0.34	0.74	0.94
da-DK	-0.95	4.13	0.12	0.28	0.75	0.96
de-DE	-0.65	3.58	0.09	0.34	0.76	0.95
el-GR	-0.92	3.64	0.09	0.28	0.71	0.94
en-US	-1.45	4.93	0.21	0.19	0.73	0.97
es-ES	-0.60	2.99	0.07	0.36	0.71	0.92
fa-IR	-0.96	2.70	0.06	0.28	0.60	0.85
fi-FI	-0.86	3.80	0.10	0.30	0.74	0.95
fr-FR	-0.37	2.65	0.05	0.41	0.72	0.91
he-IL	-0.72	3.44	0.08	0.33	0.73	0.94
hi-IN	-0.76	3.10	0.08	0.32	0.69	0.91
hu-HU	-0.55	3.25	0.08	0.37	0.75	0.94
hy-AM	-1.05	3.35	0.08	0.26	0.65	0.91
id-ID	-0.67	3.33	0.08	0.34	0.73	0.93
is-IS	-0.56	3.19	0.07	0.36	0.74	0.93
it-IT	-0.46	2.82	0.06	0.39	0.72	0.91
ja-JP	-0.48	2.77	0.06	0.38	0.71	0.91
jv-ID	-0.34	2.95	0.06	0.42	0.76	0.93
ka-GE	-0.46	2.59	0.06	0.39	0.70	0.89
km-KH	-0.23	1.62	0.03	0.44	0.64	0.80
kn-IN	-0.94	2.55	0.05	0.28	0.58	0.83
ko-KR	-0.49	3.42	0.08	0.38	0.77	0.95
lv-LV	-0.81	3.62	0.09	0.31	0.73	0.94
ml-IN	-1.39	3.64	0.10	0.20	0.61	0.90
mn-MN	-0.79	3.32	0.07	0.31	0.70	0.93
ms-MY	-0.77	3.55	0.08	0.32	0.73	0.94
my-MM	-0.97	4.12	0.08	0.27	0.75	0.96
nb-NO	-0.72	3.65	0.09	0.33	0.75	0.95
nl-NL	-0.80	3.71	0.10	0.31	0.74	0.95
pl-PL	-0.52	2.65	0.06	0.37	0.69	0.89
pt-PT	-0.56	3.05	0.07	0.36	0.72	0.92
ro-RO	-0.36	3.00	0.06	0.41	0.76	0.93
ru-RU	-0.47	3.12	0.07	0.38	0.75	0.93
sl-SL	-0.63	3.25	0.08	0.35	0.73	0.93
sq-AL	-0.54	3.04	0.07	0.37	0.73	0.92
sv-SE	-0.51	3.53	0.09	0.37	0.78	0.95
sw-KE	-0.89	3.26	0.08	0.29	0.68	0.91
ta-IN	-0.70	3.20	0.07	0.33	0.71	0.92
te-IN	-0.65	2.18	0.04	0.34	0.61	0.82
th-TH	-0.66	2.61	0.06	0.34	0.66	0.88
tl-PH	-1.12	3.72	0.09	0.25	0.68	0.93
tr-TR	-0.71	3.53	0.09	0.33	0.74	0.94
ur-PK	-0.80	3.30	0.08	0.31	0.70	0.92
vi-VN	-1.72	3.78	0.10	0.15	0.54	0.89
zh-CN	-0.42	2.35	0.06	0.40	0.68	0.87
zh-TW	-0.56	1.97	0.05	0.36	0.61	0.80

Table 8: Logistic regression results by language