Comparing Pipelined and Integrated Approaches to Dialectal Arabic Neural Machine Translation

Pamela Shapiro
Johns Hopkins University
pshapiro@jhu.edu

Kevin Duh
Johns Hopkins University
kevinduh@cs.jhu.edu

Abstract

When translating diglossic languages such as Arabic, situations may arise where we would like to translate a text but do not know which dialect it is. A traditional approach to this problem is to design dialect identification systems and dialect-specific machine translation systems. However, under the recent paradigm of neural machine translation, shared multi-dialectal systems have become a natural alternative. Here we explore under which conditions it is beneficial to perform dialect identification for Arabic neural machine translation versus using a general system for all dialects.

1 Introduction

Arabic exhibits a linguistic phenomenon called diglossia—speakers use Modern Standard Arabic (MSA) for formal settings and local dialects for informal settings. There are broad categories of dialects by region, such as Levantine or Maghrebi. However, dialects also vary at a finer-grained level, even within individual countries. An additional complication is that code-switching, i.e. mixing MSA and dialect, is a common occurrence (Elfardy et al., 2014). To put the importance of handling Arabic dialects in perspective, Ethnologue lists Arabic as having the 5th highest number of L1 speakers, spread over 21 regional dialects.[1]

The bulk of work on translating Arabic dialects uses rule-based and statistical machine translation, and much of it is translating between dialects and MSA. Generally, this work builds systems for specific dialects, with substantial amounts of information about the dialects themselves built in (Harrat et al., 2017).

In the meantime, neural machine translation has become the dominant paradigm, and with it multilingual systems have become a more natural possibility (Firat et al., 2016). These systems know nothing about the specific languages involved, but use shared embedding spaces and parameters to yield benefits especially with lower-resource languages. It is a natural extension to apply this to the space of Arabic dialects (Hassan et al., 2017).

There are many possibilities of what exactly a multilingual system might look like, but we focus on one particular decision: Suppose we want to be able to translate a test sentence from an unknown dialect. Is it better to perform dialect identification and then translate with a finely tuned system for that dialect (i.e. a pipelined approach)? Or is it better to throw everything into one integrated, multilingual system[2] which we use for all input regardless of dialect? And how accurate does our dialect identification have to be for the pipeline approach to be useful?

We perform a set of exploratory experiments quantifying this trade-off on LDC data consisting of MSA, Levantine, and Egyptian bitexts, using a standard Transformer architecture (Vaswani et al., 2017). The experimental setup is illustrated in Figure 1 and described in detail in Section 4. To explore the effect of quality of dialect identification, we perform a set of artificial experiments where we add increasing amounts of random noise to reduce language identification accuracy.

Our results show that in some scenarios, depending on the language identification accuracy, there is a cross-over point where the pipelined approach outperforms the integrated, multilingual approach in terms of BLEU scores, and vice versa. We then propose avenues for future work in this direction, based on our initial observations.

[1] https://www.ethnologue.com/statistics/size

[2] In the case of this paper, "multilingual" system refers to a single multi-dialectal system trained on multiple Arabic dialects.

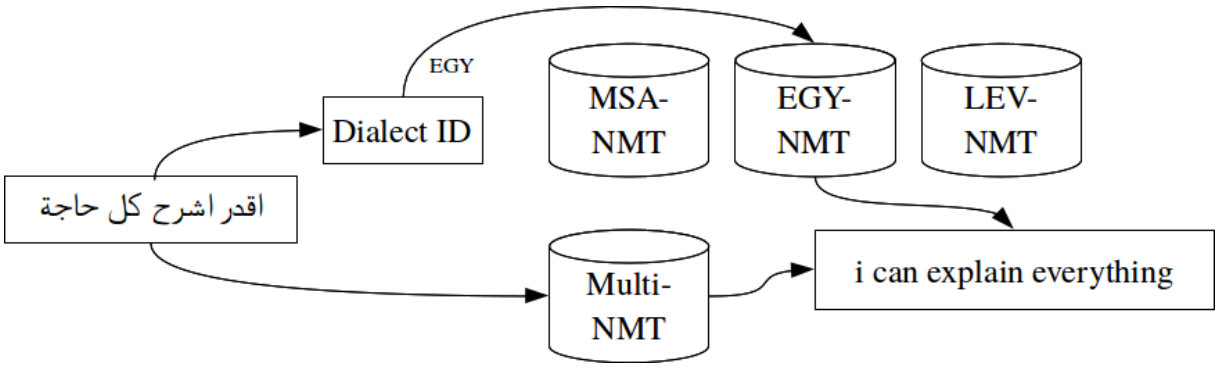


Figure 1: Illustration of our setup. Here, a test sentence of unknown dialect either gets run through a pipeline, where it is classified as Egyptian and then run through an Egyptian-tuned system, or is run through an integrated, multidialectal system.

2 Arabic Dialects

We provide here some background on Arabic dialectal variation for context. Modern Standard Arabic is the formal variety of Arabic, learned in schools and used for formal texts and newscasts. MSA is rooted in the Classical Arabic of the Qur'an, though there have been changes in vocabulary and certain aspects of grammar over time.

However, for most speakers their native language is a regional dialect of Arabic which diverges substantially. One theme among the dialects is the disappearance of certain grammatical attributes of MSA, such as case and the dual form. The dialects also display lexical and phonetic divergences, with some modification of grammatical structures such as tense markers. (Versteegh, 2014; Watson, 2007)

One major challenge of working with dialects in an NLP setting is that they have not been historically written down. However, with the rise of informal texts on the internet and social media, it is more common for dialectal Arabic to appear on the internet, but without having formalized orthography. In fact, it is common on social media to use Latin script including numerals to represent Arabic sounds, dubbed Arabizi (Bies et al., 2014). We work with data which is in the Arabic script only, but Arabizi is an important phenomenon to keep in mind for future work.

The exact regional groupings of regional dialects are not entirely consistent, but here are a few major groupings (including the two which we work with in this paper):

1. **Maghrebi**: spoken in Morocco, Algeria, Tunisia, Libya, Western Sahara, and Mauritania. Maghrebi has French and Berber influences, and not generally mutually intelligible with Eastern dialect groups. (Turki et al., 2016)

2. **Egyptian**: unusual in the amount of media available for NLP, such as Egyptian Arabic Wikipedia. Egypt has produced cinema in Egyptian Arabic that is distributed across the Arab world, increasing the reach of the dialect.

3. **Levantine**: spoken in parts of Lebanon, Jordan, Syria, Palestine, Israel, and Turkey.

4. **Arabian Peninsula**: includes subcategories such as Gulf (spoken along the Persian Gulf) and Hejazi (spoken in parts of Saudi Arabia including Mecca).

5. **Iraqi**: spoken in Iraq and parts of neighboring countries, also called Mesopotamian Arabic.

Zaidan and Callison-Burch (2014) detail in particular the ways in which dialectal varieties might manifest in their written form, from an NLP perspective. For instance, with respect to morphology, they note that the disappearance of grammatical case in dialects mostly only appears in the accusative when a suffix is added, because case in MSA generally are denoted by short vowels which are usually omitted from text. The disappearance of duals and feminine plurals is also noticeable, as well as the addition of more complex cliticization (such as circumfix negation). With respect to syntax, they note that VSO word order is more prevalent in MSA than dialects. Finally, lexical differences are noticeable in text as well.

3 Related Work

3.1 Translating Arabic Dialects

Harrat et al. (2017) provide a survey of machine translation for Arabic dialects. There has been a lot of work translating between dialects and MSA, primarily rule-based (Salloum and Habash, 2012), with some statistical machine translation approaches (Meftouh et al., 2015), which also translates between different dialects. More recently, Erdmann et al. (2017) translate between dialects with statistical MT, additionally modeling morphology and syntax.

Translating between Arabic dialects and other languages has dealt primarily with English as the other language, as we do here. Most work on this has been done with statistical machine translation systems, and generally involves pivoting through MSA or rule-based conversions to MSA. Sawaf (2010) use a hybrid rule-based, SMT system to translate dialectal Arabic. Zbib et al. (2012) explore the effects of different amounts of dialectal bitext versus MSA for SMT and try pivoting through MSA. Sajjad et al. (2013) adapts Egyptian Arabic to look like MSA with character-level transformations and uses SMT with phrase table merging to incorporate MSA-to-English data. We model our data setup after this paper, additionally using the Levantine data from the LDC corpus they use for Egyptian data (LDC2012T09). Meanwhile, Salloum et al. (2014) develop several variants of MSA and DA using SMT, and learn a Naive Bayes Classifier to determine which system would be best suited to translate data of unknown dialect. This is similar to our work in considering the possibility of the dialect being unknown, though we consider Neural Machine Translation (NMT) approaches.

As for using NMT on dialectal Arabic, Guellil et al. (2017) try using NMT on transliterated Algerian data and find that SMT outperforms it. Meanwhile, Hassan et al. (2017) generate synthetic Levantine data using monolingual word embeddings and add that to MSA-English data, briefly considering both multilingual and fine-tuning approaches as we do. While their main focus is the generation of synthetic data with monolingual data, we instead focus on investigating multilingual and fine-tuning approaches and how they interact with dialect identification when the dialect is unknown, additionally exploring the effect of Byte-Pair Encoding (BPE).

3.2 Neural Machine Translation for Dialects and Varieties

While NMT for Arabic dialects has not been extensively explored, there has been some work translating dialects and varieties with NMT recently. Costa-jussà et al. (2018) find that NMT improves over SMT for translating between Brazilian and European Portuguese, though that is a higher resource setting. Lakew et al. (2018b) use a multilingual Transformer for language varieties, as we do. However, their focus is translating into the different varieties rather than from an unknown variety, and they do not work with Arabic.

3.3 Arabic Dialect Identification

There has been a lot of work on Arabic dialect identification. Notably, Zaidan and Callison-Burch (2014) collect crowd-sourced dialect identification annotations and train classifiers to distinguish between MSA, Gulf, Levantine, and Egyptian varieties of Arabic, achieving accuracies ranging from 69.1% to 86.5%. More recently, Salameh and Bouamor (2018) have begun to focus on finer-grained classification, classifying dialects across 25 different cities. They develop a system with fine-grained accuracy of 67.9% for sentences with an average length of 7 words, and more than 90% with 16 words. Here we analyze how NMT is affected by dialect identification only between MSA, Egyptian, and Levantine. However, with the upcoming release of the MADAR corpus (Bouamor et al., 2018), we hope to extend this analysis to the finer-grained case in future work.

3.4 Multilingual NMT

One of the benefits of neural machine translation is the ease of sharing parameters across models, lending itself well to multilingual machine translation (Firat et al., 2016; Johnson et al., 2017; Lee et al., 2017). A multilingual approach uses all of the training data together (possibly up-sampling low-resource languages) to build one model with a single set of parameters.

On the other hand, people have also found transfer learning by simple fine-tuning to work well, especially between related high-resource and low-resource languages (Zoph et al., 2016). The multilingual approach has the benefit of not requiring us to know which dialect we are translating. Meanwhile, with enough training data in the correct dialect, we may be able to do better than

that with the fine-tuning approach. This is the trade-off we explore here. We use a Transformer model (Vaswani et al., 2017), as it has seen to do perform better in general as well as in the multilingual setting (Lakew et al., 2018a).

4 Models

We use a Transformer model for all of our experiments (Vaswani et al., 2017). The Transformer is a recent alternative to recurrent neural sequence-to-sequence models. Instead of just using attention to connect encoder recurrent states to decoder recurrent states, the Transformer expands the function of attention to encompass the main task. It uses self-attention, which is attention applied to two states within the same sequence, as the foundation for sequence representations, rather than an RNN. The Transformer also increases the power of attention with multi-head attention, which performs an attention function several times in parallel, concatenates, and then projects the representation.

In the Transformer, the encoder consists of several layers of multi-head self-attention paired with a feedforward network. The decoder is similar but also has multi-head attention over the encoder output and masks future decoder output tokens. This model has been shown to achieve state-of-the-art in neural machine translation, and we can use it for multilingual or fine-tuning setups the same way we would a sequence-to-sequence model as in Sutskever et al. (2014).

With regards to the different ways we train the Transformer, we describe our setup, illustrated in Figure 1.

4.1 Multidialectal Model

One approach to being able to translate sentences of unknown dialect is to train a system in a "multilingual," or here multidialectal fashion. The simplest variant, introduced in Johnson et al. (2017), uses a shared wordpiece vocabulary and trains with data from several languages, adding a tag indicating the language at the beginning of each sentence. We follow this approach, but removing the tag, as in (Lee et al., 2017), and using a Transformer. We use a shared subword vocabulary by applying Byte-Pair Encoding (BPE) to the data for all variants concatenated (Sennrich et al., 2016). However, here we are not dealing with completely different languages, but rather variants of a language.

4.2 Dialect ID and Dialect-Tuned Models

On the other hand, dialect identification is an active area of research, and an alternative approach is to design a dialect-specific model for each dialect. One could simply train a system on dialect data alone. However, since dialects of Arabic are generally far lower-resource than MSA, this is difficult for NMT. To leverage the MSA to benefit the dialect-specific system, we follow the approach of Zoph et al. (2016), simply continuing to train on the low-resource dialects from the model trained on high-resource MSA. Again, we use a shared subword vocabulary trained on all of our training data of all variants, to avoid problems with out-of-vocabulary words.

5 Experiments

We perform experiments comparing multidialectal and dialect-tuned approaches, and then focus on the effect of misclassified dialects with a set of experiments adding synthetic noise to our language classification.

5.1 Data

For MSA training data, we use 5 million sentences of UN Data (Ziemski et al., 2016), in addition to GALE data, LDC2004T17, and LDC2004T18. For MSA dev data, we used NIST OpenMT '08, and for MSA test data, we used NIST OpenMT '09. For Egyptian and Levantine data, we used LDC2012T09, reserving the last 6k sentences of each for dev, test1, and test2 respectively. We only show results for test1 here, and reserve test2 for future use. We normalized the Arabic orthography, tokenized, cleaned, and deduplicated.[3] We applied BPE with 10k, 30k, and 50k merge operations, training on the concatenation of all of the training data. The final counts of sentences for our data are shown in Table 2.

5.2 Implementation

We use the Sockeye (Hieber et al., 2017) implementation of a Transformer (Vaswani et al., 2017) for all of our experiments. We used 6 layers, 512-dimensional embeddings, 8 attention heads, and

[3] By normalize the orthography, we mean that we removed diacritics and tatweels and normalized alefs and yas. For tokenization, we used the Moses tokenizer for English, since it does not have one for Arabic. We did not apply Arabic-specific tokenization that segments clitics as well.

	Multidialectal			Dialect-Tuned		
	10k BPE	30k BPE	50k BPE	10k BPE	30k BPE	50k BPE
MSA	38.23	**38.49**	38.22	36.42	38.04	36.79
EGY	22.44	21.93	21.12	**22.79**	21.64	20.86
LEV	22.31	21.89	21.47	**23.78**	22.68	22.35

Table 1: Multidialectal and dialect-tuned approaches for different BPE sizes. In this experiment, we assume the dialect of the test sentences are known so that the correct Dialect-Tuned models can be applied.

	train	dev	test1	test2
MSA	4,266k	1.4k	1.3k	N/A
EGY	32k	2k	2k	2k
LEV	129k	2k	2k	2k

Table 2: Number of sentences in dataset splits.

2048 hidden units in feed forward layers. We optimize with Adam (Kingma and Ba, 2014), with an initial learning rate of 0.0002, and a learning rate reduce factor of 0.9, applying label smoothing with a smoothing parameter of 0.1. We select the model based on dev BLEU. This is from the sockeye-recipes default medium transformer model[4], which closely but not exactly follows the official AWS sockeye autopilot Transformer model[5]

For our multidialectal experiments, we do not do any up-sampling of the lower-resource data, though this would be another axis to explore in future work.

5.3 Artificially Noised Dialect Identification

With the goal of exploring the importance of dialect identification in this context, we examine how the fine tuning approach suffers as we add artificial noise to to dialect identification. We do this by some percentage of the time randomly choosing one of the other models to decode with. We do this at intervals of 10%. To be precise, we provide pseudocode of the approach below.

$\mathcal{D} = \{MSA, LEV, EGY\}$
for test sentence s with true dialect $d \in \mathcal{D}$ **do**
  With probability p, switch model dialect $\hat{d}$
  if Switching **then**
    Sample $\hat{d}$ uniformly from $\mathcal{D} \setminus d$
  else
    $\hat{d} = d$
  Translation t =decode($model_{\hat{d}}, s$)

6 Results

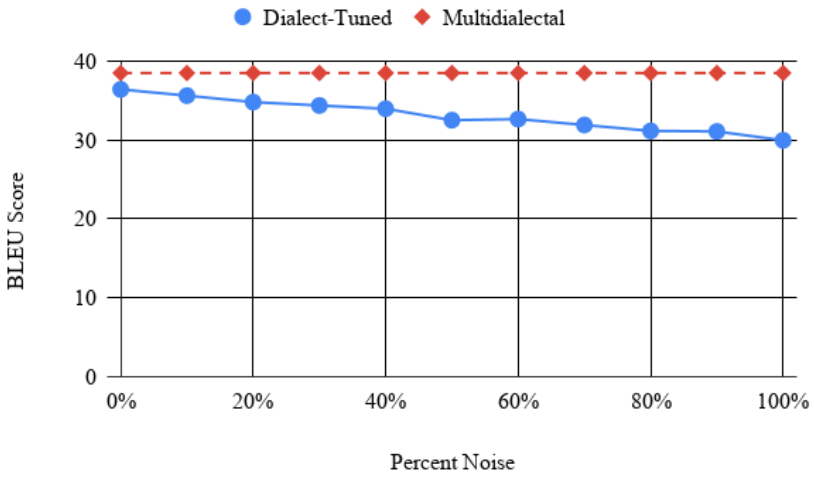
As an initial experiment, we compare the two approaches in the case where the dialect of the test sentence is known. The test1 BLEU scores of the multidialectal and dialect-tuned approaches for different BPE sizes are in Table 1. We can see that scores are pretty consistent across BPE sizes, with 10k being best for EGY and LEV while 30k is best for MSA. As we'd expect, with complete information about dialects, the fine tuning approach for EGY and LEV achieves the highest scores. With LEV, which has more available training data, this trend is clearer across BPE sizes. With EGY, which has a much smaller amount of training data, this gain is only achieved in the best BPE size for EGY of 10k. Interestingly, the multidialectal model does best for MSA, rather than the model trained only on MSA. It is possible that the comparatively small amount of dialectal data provides useful regularization for the MSA model, or that it is benefiting from the shared aspects of the dialects.

Our main results are shown in Figure 2. Here, we plot the BLEU score of each test set, MSA, EGY, and LEV, as the amount of noise we've added to dialect identification increases. It is interesting to see that in all cases, it consistently degrades as more noise is added, but ultimately doesn't reach a terribly low score even at 100% noise. We include the multidialectal system's performance as a horizontal dotted line. Where the lines intersect in the EGY and LEV case represents at what level of noise we have lost the benefit of dialect identification. So, by 20% error in both cases, we might as well be using the multidialectal system.

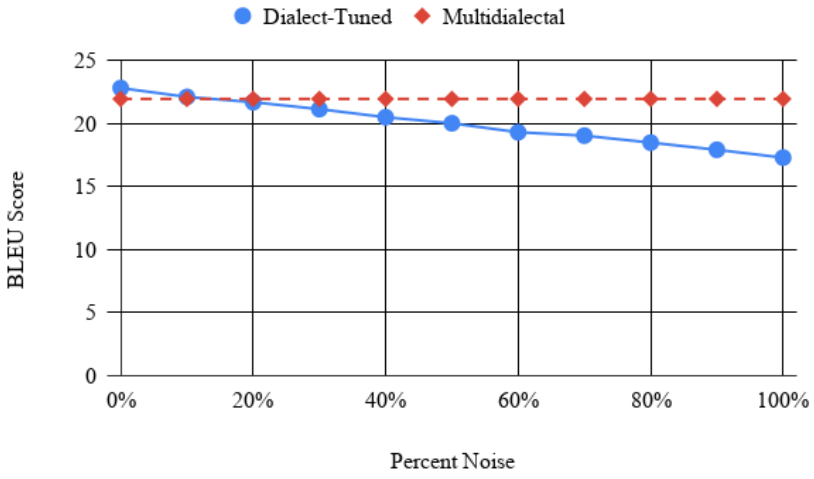
We note that this result (crossover at 20%) should be interpreted in light of the training data and the models we used. The cross-over point in

[4] https://github.com/kevinduh/sockeye-recipes/blob/master/hpm/tm1.hpm-template

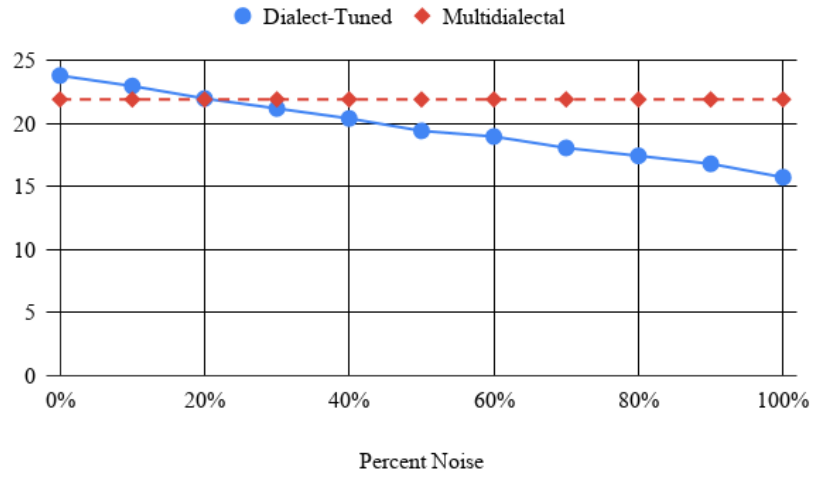
[5] https://github.com/awslabs/sockeye/blob/master/sockeye_contrib/autopilot/models.py



(a) Modern Standard Arabic Base Model



(b) Egyptian-Tuned Model



(c) Levantine-Tuned Model

Figure 2: Effect of noise on fine tuning models with 10k BPE. The dialect-tuned models are models fined-tuned on specific dialects, applied with noisy dialect identification. We provide the multidialectal model performances for comparison.

BLEU score between multidialectal and dialect-tuned models will depend on the relative strengths of each model. Further, dialect identification errors may be correlated (e.g. higher confusion between two close-by dialects, compared to two far-away ones). In other words, we imagine for different datasets and models, it will be important to re-run this analysis. We also hypothesize the the results may vary by sentence length, as well, which influences language identification accuracy.

To understand which combinations of test sentences and models are least and most compatible, we also present a matrix of all combinations of model and test set in 3. We can see that EGY and LEV test sets are much more harmed by the MSA model than the LEV- and EGY- tuned models respectively. It is possible that there is some shared vocabulary between EGY and LEV that it is learning, or that the EGY and LEV training sets are just a closer domain to each other than to MSA.

Finally, we use a very simple baseline for dialect ID to see how it performs. We train a model for language ID with `langid.py` (Lui and Baldwin, 2012), which uses naive Bayes classifier with a multinomial event model. Training `langid.py` on our data does not work well for dialect ID—in particular, the system is very sensitive to data size. It would probably be better to provide larger quantities of monolingual data for this if avaiable. However, we report results here to give a sense of how a very basic language ID system might perform. We try training it in two ways: (1) with the data proportions left as-is and (2) upsampling the EGY and LEV data sizes to match the MSA data size. (1) results in predicting almost all sentences as MSA, and (2) results in predicting almost all sentences as EGY. As you can see in Table 4, this results in (1) performing well only on MSA and (2) performing well only on EGY, with the other results being heavily degraded.

7 Discussion

While adding random noise is not necessarily reflective of the cases in which dialect identification systems would be likely to make errors, it does help us get an idea of how useful it is to tune an NMT model to a specific Arabic dialect, in light of faulty knowledge about which dialect it is.

Our mutidialectal approach performs competitively with the tuned approaches, but at a well-chosen BPE size and with less than 20% error, it does seem beneficial to tune to the dialect. A couple factors seem to contribute to whether it is useful, beyond error rate of dialect system:

1. **BPE Merge Hyperparameter**: The dialects seem to perform best at the lowest BPE merge hyperparameter that we tried. This is the lower range of BPE settings usually used, but it would be worthwhile to explore this with even lower settings. As the merge hyperparameter decreases, we are getting closer to character-level, which may be able to handle the shared subwords across dialects better in light of varied morphological inflections.

2. **Amount of Training Data**: There does seem to be a difference in performance of tuned models between EGY and LEV which lines up with data size. There is much less EGY training data, and the fine-tuning process converges very quickly on the data. On the other hand, LEV has a decent amount of training data and shows more consistent improvements over the multidialectal model.

One trend we observed that is worth noting, is that the average sentence length differs substantially from MSA to EGY and LEV in our test sets, which may make sense given the more formal content of MSA. This might have some implications for NMT and dialect identification. Dialect identification is known to be harder on shorter sentences. Meanwhile, NMT can sometimes be hard on very long sentences. It is worth looking into these subtleties for future work understanding how to optimize NMT translation of unknown dialects of Arabic.

8 Future Work

One area for future work would be further exploring how this setup interacts with existing dialect identification systems to determine their usefulness for Arabic NMT of unknown dialects.

Additionally, the role of morphology in this setup with BPE would be useful to explore. It is possible that models that incorporate characters would be more useful at capturing shared information between MSA and dialects.

Finally, it would be great to test this on more dialects. We hope to do experimentation on larger dialectal corpora in the future, such as the soon-to-be-released MADAR corpus (Bouamor et al., 2018).

		Test Set		
		MSA	EGY	LEV
Model	MSA	**36.42**	27.38	25.39
	EGY	10.33	**22.79**	19.81
	LEV	8.24	15.70	**23.78**

Table 3: How each model performs on each test set.

	Test Set		
	MSA	EGY	LEV
No up-sampling	36.24	10.35	8.25
Up-sampling	27.56	22.79	15.71

Table 4: How well the pipelined approach does with `langid.py` as dialect ID.

9 Conclusion

We have done a set of preliminary experiments exploring a couple different approaches to translating Arabic of unknown dialect. An integrated, multi-dialectal model proved to be beneficial for MSA. Meanwhile, with a dialect identification error rate less than 20% and with a small enough BPE size and large enough training data, using a pipelined approach with a dialect-tuned model proves to be beneficial. We hope that this can be beneficial for determining future directions translating Arabic dialects.

10 Acknowledgments

This work is supported in part by the Office of the Director of National Intelligence (ODNI), Intelligence Advanced Research Projects Activity (IARPA), via contract FA8650-17-C-9115. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of ODNI, IARPA, or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for governmental purposes notwithstanding any copyright annotation therein.

References

Ann Bies, Zhiyi Song, Mohamed Maamouri, Stephen Grimes, Haejoong Lee, Jonathan Wright, Stephanie Strassel, Nizar Habash, Ramy Eskander, and Owen Rambow. 2014. Transliteration of arabizi into arabic orthography: Developing a parallel annotated arabizi-arabic script sms/chat corpus. In *Proceedings of the EMNLP 2014 Workshop on Arabic Natural Language Processing (ANLP)*, pages 93–103.

Houda Bouamor, Nizar Habash, Mohammad Salameh, Wajdi Zaghouani, Owen Rambow, Dana Abdulrahim, Ossama Obeid, Salam Khalifa, Fadhl Eryani, Alexander Erdmann, et al. 2018. The madar arabic dialect corpus and lexicon. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC-2018)*.

Marta R Costa-jussà, Marcos Zampieri, and Santanu Pal. 2018. A neural approach to language variety translation. In *Proceedings of the Fifth Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial 2018)*, pages 275–282.

Heba Elfardy, Mohamed Al-Badrashiny, and Mona Diab. 2014. Aida: Identifying code switching in informal arabic text. In *Proceedings of The First Workshop on Computational Approaches to Code Switching*, pages 94–101.

Alexander Erdmann, Nizar Habash, Dima Taji, and Houda Bouamor. 2017. Low resourced machine translation via morpho-syntactic modeling: the case of dialectal arabic. *arXiv preprint arXiv:1712.06273*.

Orhan Firat, Kyunghyun Cho, and Yoshua Bengio. 2016. Multi-way, multilingual neural machine translation with a shared attention mechanism. In *Proceedings of NAACL-HLT*, pages 866–875.

Imane Guellil, Faiçal Azouaou, and Mourad Abbas. 2017. Comparison between neural and statistical translation after transliteration of algerian arabic dialect. *WiNLP: Women & Underrepresented Minorities in Natural Language Processing (co-located with ACL 2017)*, pages 1–5.

Salima Harrat, Karima Meftouh, and Kamel Smaili. 2017. Machine translation for arabic dialects (survey). *Information Processing & Management*.

Hany Hassan, Mostafa Elaraby, and Ahmed Tawfik. 2017. Synthetic data for neural machine translation of spoken-dialects. *arXiv preprint arXiv:1707.00079*.

Felix Hieber, Tobias Domhan, Michael Denkowski, David Vilar, Artem Sokolov, Ann Clifton, and Matt Post. 2017. Sockeye: A toolkit for neural machine translation. *arXiv preprint arXiv:1712.05690*.

Melvin Johnson, Mike Schuster, Quoc V Le, Maxim Krikun, Yonghui Wu, Zhifeng Chen, Nikhil Thorat, Fernanda Viégas, Martin Wattenberg, Greg Corrado, et al. 2017. Googles multilingual neural machine translation system: Enabling zero-shot translation. *Transactions of the Association for Computational Linguistics*, 5:339–351.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Surafel Melaku Lakew, Mauro Cettolo, and Marcello Federico. 2018a. A comparison of transformer and recurrent neural networks on multilingual neural machine translation. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 641–652.

Surafel Melaku Lakew, Aliia Erofeeva, and Marcello Federico. 2018b. Neural machine translation into language varieties. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 156–164.

Jason Lee, Kyunghyun Cho, and Thomas Hofmann. 2017. Fully character-level neural machine translation without explicit segmentation. *Transactions of the Association for Computational Linguistics*, 5:365–378.

Marco Lui and Timothy Baldwin. 2012. langid. py: An off-the-shelf language identification tool. In *Proceedings of the ACL 2012 system demonstrations*, pages 25–30. Association for Computational Linguistics.

Karima Meftouh, Salima Harrat, Salma Jamoussi, Mourad Abbas, and Kamel Smaili. 2015. Machine translation experiments on padic: A parallel arabic dialect corpus. In *Proceedings of the 29th Pacific Asia Conference on Language, Information and Computation*, pages 26–34.

Hassan Sajjad, Kareem Darwish, and Yonatan Belinkov. 2013. Translating dialectal arabic to english. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, volume 2, pages 1–6.

Mohammad Salameh and Houda Bouamor. 2018. Fine-grained arabic dialect identification. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1332–1344.

Wael Salloum, Heba Elfardy, Linda Alamir-Salloum, Nizar Habash, and Mona Diab. 2014. Sentence level dialect identification for machine translation system selection. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, volume 2, pages 772–778.

Wael Salloum and Nizar Habash. 2012. Elissa: A dialectal to standard arabic machine translation system. *Proceedings of COLING 2012: Demonstration Papers*, pages 385–392.

Hassan Sawaf. 2010. Arabic dialect handling in hybrid machine translation. In *Proceedings of the conference of the association for machine translation in the americas (amta), denver, colorado*.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 1715–1725.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112.

Houcemeddine Turki, Emad Adel, Tariq Daouda, and Nassim Regragui. 2016. A conventional orthography for maghrebi arabic. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC)*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 5998–6008.

Kees Versteegh. 2014. *Arabic Language*. Edinburgh University Press.

Janet CE Watson. 2007. *The Phonology and Morphology of Arabic*. OUP Oxford.

Omar F Zaidan and Chris Callison-Burch. 2014. Arabic dialect identification. *Computational Linguistics*, 40(1):171–202.

Rabih Zbib, Erika Malchiodi, Jacob Devlin, David Stallard, Spyros Matsoukas, Richard Schwartz, John Makhoul, Omar F Zaidan, and Chris Callison-Burch. 2012. Machine translation of arabic dialects. In *Proceedings of the 2012 conference of the north american chapter of the association for computational linguistics: Human language technologies*, pages 49–59. Association for Computational Linguistics.

Michal Ziemski, Marcin Junczys-Dowmunt, and Bruno Pouliquen. 2016. The united nations parallel corpus v1. 0. In *Lrec*.

Barret Zoph, Deniz Yuret, Jonathan May, and Kevin Knight. 2016. Transfer learning for low-resource neural machine translation. *arXiv preprint arXiv:1604.02201*.