WERd: Using Social Text Spelling Variants for Evaluating Dialectal Speech Recognition

WERD: USING SOCIAL TEXT SPELLING VARIANTS FOR EVALUATING DIALECTAL SPEECH RECOGNITION

Ahmed Ali[1,2], Preslav Nakov[1], Peter Bell[2], Steve Renals[2]

[1]Qatar Computing Research Institute, HBKU, Doha, Qatar
[2]Centre for Speech Technology Research, University of Edinburgh, UK

{amali, pnakov}@qf.org.qa, {peter.bell, s.renals}@ed.ac.uk

ABSTRACT

We study the problem of evaluating automatic speech recognition (ASR) systems that target dialectal speech input. A major challenge in this case is that the orthography of dialects is typically not standardized. From an ASR evaluation perspective, this means that there is no clear gold standard for the expected output, and several possible outputs could be considered correct according to different human annotators, which makes standard word error rate (WER) inadequate as an evaluation metric. Such a situation is typical for machine translation (MT), and thus we borrow ideas from an MT evaluation metric, namely TERp, an extension of translation error rate which is closely-related to WER. In particular, in the process of comparing a hypothesis to a reference, we make use of spelling variants for words and phrases, which we mine from Twitter in an unsupervised fashion. Our experiments with evaluating ASR output for Egyptian Arabic, and further manual analysis, show that the resulting WERd (i.e., *WER for dialects*) metric, a variant of TERp, is more adequate than WER for evaluating dialectal ASR.

Index Terms— Automatic speech recognition, dialectal ASR, ASR evaluation, word error rate, multi-reference WER

1. INTRODUCTION

Automatic Speech Recognition (ASR) has shown fast progress recently, thanks to advancements in deep learning. As a result, the best systems for English have achieved a single-digit word error rate (WER) for some conversational tasks [1]. However, this is different for dialectal ASR, for which the WER can easily go over 40%[2].

In a standardized language such as English, we know that *enough* is a correct spelling, while *enuf* is not. However, we cannot be sure about the correct spellings of dialectal words; at best, we would know what a preferred or a dominant spelling is. This is because dialects typically do not have an official status and thus their spelling is not regulated, which opens widely the door to orthographic variation.[1]

[1]Note that here we target primarily intra-dialectal variation. Yet, there is also inter-dialect variation, e.g., between the different dialects of Arabic.

English Gloss	Spelling Variants	Buckwalter
He was not	ماكانش	mAkAn$
	ماكنش	mAkn$
	ما كانش	mA kAn$
	مكنش	mkn$
I told him	قولتله	qwltlh
	قولت له	qwlt lh
	قلتله	qltlh
	قلت له	qlt lh
By the morning	على الصبح	ElY AlSbH
	علي الصبح	Ely AlSbH
	ع الصبح	E AlSbH
	عالصبح	EAlSbH
	عصّبح	ESbH

Table 1: Egyptian phrases with multiple spelling variants: shown in Arabic script and in Buckwalter transliteration.

Table 1 shows some examples of spelling variation in Dialectal Arabic (DA). We can see that clitics (pronouns and negations) can be written concatenated or separated from the verb, the definite article can undergo different spelling variations due to coarticulation with the following word, long vowels can become short, and thus be dropped as they are typically not written in Arabic, etc. While some variations can happen in standardized languages such as English, e.g., *healthcare* vs. *health care*, or *organize* vs. *organise*, this is much less common, and in ASR it is easily handled with simple rules, e.g., the Global Mapping file[2] in SCLITE [3, 4].

The above examples partially explain the high WER for dialects. While they suffer from the lack of training resources, the main problem is their informal status, which means that their spelling is rarely regulated. This makes training an ASR system for dialects much harder as there is no single gold standard towards which to optimize at training time.

[2]The global mapping file can help for handcrafted variants like *color/colour* and *ten/10* in English. However, it is not applicable to dialectal Arabic, where multiple spelling variants are acceptable; we use 11M pairs.

More importantly, it is hard to evaluate such a system and to measure progress as multiple possible text outputs for the same speech signal could be considered correct by different people. Thus, there is need for an evaluation measure that would allow for common spelling variations. In this work, we propose to mine such variations from dialectal Arabic tweets and to incorporate them as spelling variants as part of a more adequate ASR evaluation measure for dialects.

Previously, the problem was addressed using the multi-reference word error rate (MR-WER) [5], which is similar to the multi-reference BLEU score [6] used to evaluate Machine Translation (MT). However, obtaining multiple references is expensive. Moreover, it could take many human annotators to get good coverage of the possible orthographic variants of the transcription of a speech recording. Thus, we propose to use a single reference, but to perform matching using spelling variants that could capture some of the variation.

This was applied to MT, e.g., for parameter optimization [7], where additional synthetic references are generated for tuning purposes, or for phrase-based SMT, where paraphrasing is applied to the source side of the phrase table [8], of the training bi-text [9], or both [10, 11, 12, 13]. Paraphrasing has been also used for evaluating text summarization [14].

More relevant to the present work, in MT evaluation, paraphrasing was applied to the output of an MT system [15]. It was also incorporated in measures such as TERp [16], which is a translation edit rate metric with paraphrases. Indeed, here we borrow ideas from TERp for dialectal ASR, with a paraphrase table (in our case, a spelling variants table), which we mine automatically from a huge collection of tweets in an unsupervised fashion. Our experiments and our manual analysis show that this is a very promising idea.

Our contributions are as follows: (*i*) We propose a method for automatically collecting spelling and tokenization variations for dialectical Arabic (and, presumably, other languages and language variants) from Twitter data; (*ii*) We further incorporate these spelling variants in an evaluation metric, WERd, which is variation of TERp, and we demonstrate its utility for dialectal Arabic ASR. We release the code for that metric, as well as the spelling variants we mined and used in the metric:[3] eleven million pairs, which we extracted from a seven-billion words corpus of dialectal Arabic tweets.

2. METHOD

We propose a method for evaluating dialectal ASR, which consists of two steps: (*i*) collecting a large number of spelling variants, which we mine from social media in an unsupervised manner, and (*ii*) using these spelling variants, with associated probabilities, into an MT-inspired evaluation measure (together with standard unit-cost word insertions, deletions, and substitutions).

[3] https://github.com/qcri/werd

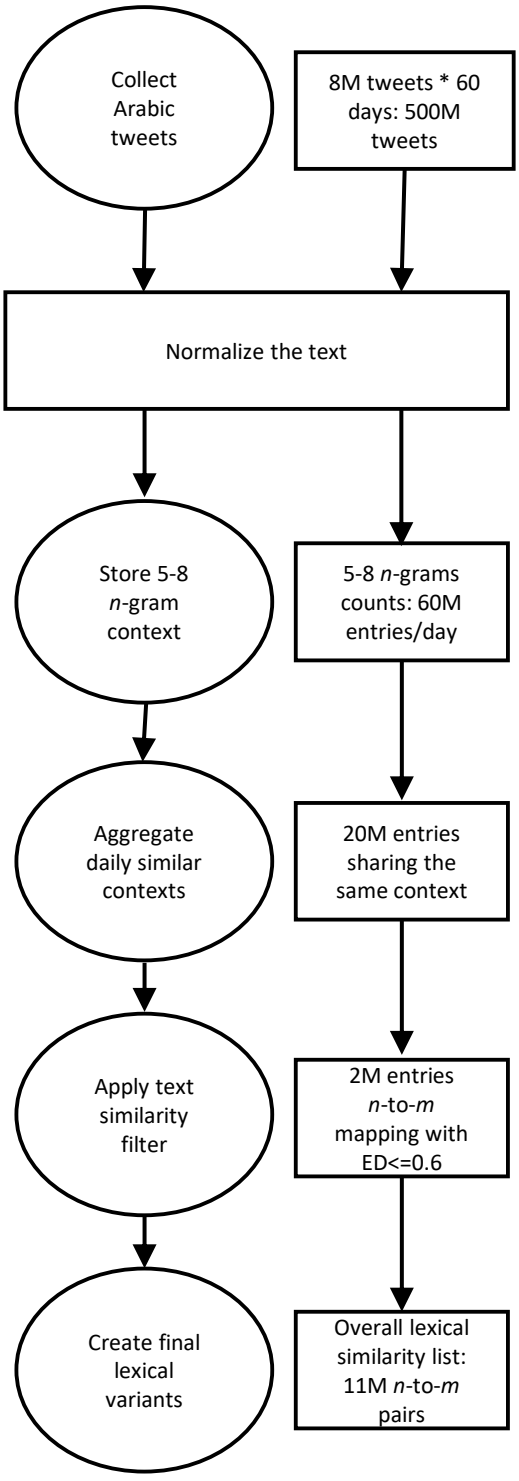

Fig. 1. Diagram of our pipeline for extracting dialectal spelling variants from Twitter.

2.1. Mining Spelling Variants from Social Media

We use social media to mine dialectal spelling variants from a collection of half a billion dialectal Arabic tweets. Our approach is language-independent, scalable, and unsupervised, as it assumes no prior knowledge about the language, its dialects, or the data.

We build a list of pairs of spelling variants with probabilities using the following steps (as shown in Figure 1):

First, we collect Arabic tweets. Then, we normalize hashtags, URLs, emoticons. We further drop Arabic diacritics and elongation, and we reduce letter repetitions to maximum three. Our pipeline is an extension to the previous work done in Arabic language processing for microblogs [17].

Next, we extract all n-grams of lengths 5–8. In each n-gram, we consider the first two and the last two words as a context, and the 1–4 words in the middle as a *target* for this context. For example, for a 5-gram we will have $< L_1, L_2, t_1, R_1, R_2 >$, while for an 8-gram we will have $< L_1, L_2, t_1, t_2, t_3, t_4, R_1, R_2 >$, where L_i and R_i represent the left and the right context words, and t_j are the target words in the middle ($1 \leq i \leq 2$, $1 \leq j \leq 4$).

Next, we generate pairs of potential spelling variants for targets that share the same contexts. This is subject to the constraint that the normalized Levenshtein distance between the targets is less than t, measured in characters. We tried values between 0.1 and 0.6 for t, and we manually inspected the resulting pairs of spelling variants. Ultimately, we set $t = 0.6$. With normalization in mind, we further impose a constraint that in each pair of spelling variants, one of the targets is extracted in the same contexts at least N times more frequently than the other one (we set N to 3). Finally, with each pair of spelling variants, we associate a score: the average of the two Levenshtein distances. The resulting scored pairs of spelling variants form a spelling variant table for WERd.

Here are two examples from this final table of *n*-to-*m* spelling variant pairs with corresponding frequencies and normalized edit distance (shown in Buckwalter):

mAfy	mAAfy	752	75	0.25
lwny w DAEt	lwny wDAEt	32	8	0.1

The first column (yellow) contains the frequent form, which is the target **mAfy**. The second column (green) contains the source **mAAfy**, which is a less frequent term. The next column is the frequency of the target, e.g., the word **mAfy** occurred 752 times. The following column is the frequency of the source in the same context, e.g., **mAAfy** occurred 75 times. Finally comes the normalized edit distance.

Related approaches for paraphrase extraction have used random walks [18], pairwise similarity [19], and continuous representations [20, 21]. Unlike that work, we mine pairs of spelling variants for ASR evaluation, not for modeling; we further allow many-to-many mappings, and we do not target canonical gold normalization.

2.2. Using the Spelling Variants for Evaluation: WERd

We borrow ideas from an evaluation measure for MT evaluation, namely *Translation Edit Rate Plus* or *TERp* [22]. TERp allows block alignment of words, called *shifts* within the hypothesis as a low cost edit, a cost of 1, the same as the cost for inserting, deleting or substituting a word. TERp uses a greedy search and shift constraints to both reduce the computational complexity and to model the quality of translation better. The metric further supports tuned weights for the edit operations, a paraphrase table, synonym/hypernym-based matching using WordNet, etc.

The main motivation for using paraphrases in TERp for MT evaluation is to capture some lexical variation, e.g., *(controversy over, polemic about), (by using power, by force), (brief, short), (response, reaction)*. In contrast, we focus on capturing spelling variation in a dialect as shown in Table 1.

In this work, we only use the paraphrasing capability of TERp. We restrict the matching to monotonic, i.e., no reorderings and no shifts. The only additional operation that we allow, compared to WER, is mapping between the hypothesis and the reference using a pair of spelling variants from our spelling variants table, which can span up to four words on either side of the pair of spelling variants as we have explained above. This monotonic version of TERp, with no reordering but with spelling variant matching capabilities gives rise to our metric for dialectal ASR evaluation, which we will call WERd (or *WER for dialects*).

3. EXPERIMENTS AND EVALUATION

3.1. Dialectal Data

Speech Data. We collected two hours of Egyptian Arabic Broadcast news [23] speech data, which we split into 1,217 segments, each 3-10 seconds long. It can be argued that Egyptian Arabic, which is one of the Arabic dialects, is a language with no established orthographic rules. This makes it difficult to develop standard transcription guidelines covering orthography. Therefore, we decided to have multiple transcriptions, but to let transcribers write the transcripts as they deemed correct, while trying to be as verbatim as possible. All the transcribers are native speakers of the chosen dialect with no linguistic background.[4] Table 2 shows the overlap between the annotators, at the segment level, for their original transcription and after applying surface normalization for *alef*, *yah* and *hah*, which is standard for Arabic. In Table 2, the first number is for the original text, and the second number is for the normalized text. We can see that even after normalization,[5] there are about 15% differences between most of the annotators.

[4]The transcribers were asked to follow these transcription guidelines: `http://alt.qcri.org/resources/MGB-3/Arabic_Transcription%20_Guidelines_20170330.pdf`

[5]Below, we will report results after normalization only.

Social Media Data. We further collected dialectal Arabic tweets in order to extract spelling variants. In particular, we issued queries using `lang:ar` against the Twitter API[6]. Note that we did not try to control the location where the tweets originated from, but only the language they were written in. We collected two months of tweets (from December 2015 and January 2016), with about eight million tweets per day on average, which yielded a total of half a billion tweets containing over seven billion word tokens.

3.2. ASR System

For our experiments, we used the speech-to-text transcription system built as part of QCRI's submission to the 2016 Arabic Multi-Dialect Broadcast Media Recognition (MGB-2) Challenge[24]. Here are some key features of that system:

Data: The training data consisted of 1,200 hours of transcribed broadcast speech data collected from the Aljazeera news channel. In addition, we had ten hours of development data [25]. We used data augmentation techniques such as speed and volume perturbation, which increased the size of the training data to three times the original size [26].

Speech lexicon: We used a grapheme-based lexicon [27] with 900k entries, which we constructed using the words that occurred more than twice in the training transcripts.

Speech features: To train the acoustic models, we used 40-dimensional high-resolution Mel Frequency Cepstral Coefficients (MFCC_hires), extracted for each speech frame, which we concatenated with 100-dimensional i-Vectors per speaker in order to facilitate speaker adaptation [28].

Acoustic models: We experimented with three acoustic models: Time-Delayed Neural Networks (TDNNs) [29], Long Short-Term Memory Recurrent Neural Networks (LSTMs), and Bi-directional LSTMs (BiLSTMs) [30]. The latter acoustic model outperforms the other two models in terms of WER. We trained all models using Lattice-Free Maximum Mutual Information (LF-MMI) [31] using the Kaldi toolkit [32].

Language model: We trained two n-gram language models (LMs). The first one, a tri-gram LM (KN3) used the spoken utterance transcripts for the 1,200 hours. We used this LM in order to generate decoding lattices. We then rescored these lattices using a four-gram LM (KN4), which we trained on the in-domain data and on some extra text. We used interpolated Kneser-Ney smoothing for both LMs, which we built using the SRILM toolkit [33]. We further trained a Recurrent Neural Network Language Model with MaxEnt Connections (RNNME) using the RNNLM toolkit [34]

Overall ASR system: We combined the three aforementioned acoustic models, and for the second pass we additionally used the four-gram and the RNNLM for re-scoring the decoded speech lattices. The overall performance was 14.7% WER on the MGB-2 tasks, and this was the best result achieved at the challenge.

[6] `http://dev.twitter.com/`

	ref2	ref3	ref4	ref5
ref1	77/86	80/84	78/86	80/87
ref2	—	74/83	71/85	72/85
ref3	—	—	77/84	78/84
ref4	—	—	—	91/93

Table 2: Pairwise overlap of the five human references before/after normalization (in %).

	WER	**TER**	**WERd**	**MR-WER**
ref1	46.2	37.4	34.3	—
ref2	42.9	38.7	35.7	—
ref3	48.9	41.9	38.3	—
ref4	46.2	39.0	35.6	—
ref5	46.0	38.3	34.9	—
ALL refs	—	—	—	25.3

Table 3: WER vs. TER vs. WERd vs. MR-WER, after normalization (in %).

3.3. Experimental Results

We first evaluated the ASR system on our two-hour dialectal Arabic test dataset using WER with respect to each of the five references. The results are shown in Table 3. We can see that the WER is much higher on our dialectal Arabic dataset, ranging in 40–50%.

We further calculated MR-WER for our ASR system using all five references, achieving a score of 25.3%. This number is much lower than when evaluating with respect to any individual reference, which is to be expected, as we allow more matching options.

Table 3 reports TER and WERd scores calculated with respect to each of the five references and shows for both metrics a strong correlation with WER. We can also see that the scores for WERd are halfway between WER and MR-WER (e.g., for ref1, it is 34.3 vs. 46.2 and 25.3, respectively), but without the need for additional human references.

There are two reasons for MR-WER to be considerably lower compared to the other metrics. First, the way foreign words and code-switching are handled by different annotators, e.g., words like *BBC* can be written in either Arabic (بي بي سي *by by sy*) or Latin characters. Some annotators would use Arabic while other would prefer English, which would allow matching either of them when evaluating the ASR output with multiple references. Second, in dialectal Arabic, there are many filler words such as يعني *yEny*, اصل *ASl* and زي *zy*, which some annotators would skip and some would keep.

	No Variants	ED $\leq$ 0.1	ED $\leq$ 0.2	ED $\leq$ 0.3	ED $\leq$ 0.4	ED $\leq$ 0.5	ED $\leq$ 0.6
ref1	37.4	37.1	36.6	35.5	34.8	34.6	34.3
ref2	38.7	38.4	37.9	36.9	36.3	36.1	35.7
ref3	41.9	41.5	40.9	39.7	38.9	38.7	38.3
ref4	39.0	38.6	38.1	36.9	36.2	36.0	35.6
ref5	38.3	37.9	37.3	36.2	35.6	35.3	34.9

Table 4: WERd using pairs of spelling variants extracted using different maximum edit distances (ED).

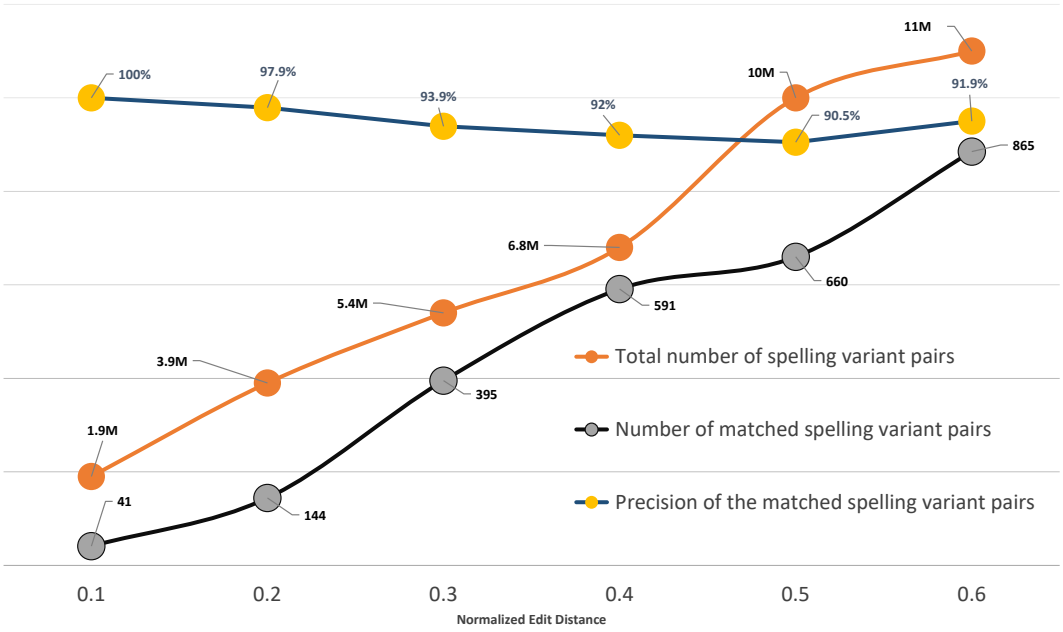

Fig. 2. Analysis of the total number of spelling variants, the number of matched variants, and the precision for different thresholds on the edit distance (with respect to ref1). Note that the y axis shows different units for each of the three curves.

4. DISCUSSION

WERd for different thresholds. Table 4 shows the performance of WERd when using pairs of spelling variants with different maximum edit distances: 0.1–0.6. As the threshold increases, WERd decreases, e.g., for ref1, it goes from 37.4 to 34.3, or 8% relative reduction. The difference is due to the number of matched spelling variant pairs, e.g., 865 for ref1.

Analysis of the pairs of spelling variant matches. Next, we study the relationship between the threshold on the maximum edit distance vs. the spelling variant table size, the number of spelling variants matches, and the accuracy of these matches. This is shown in Figure 3.3, where we focus the best reference, ref1 (according to native speakers of Egyptian Arabic who have a linguistic background). We can see that the threshold has a major impact on the spelling variant table size: going from 0.1 to 0.6 yields a six times larger table. It also yields a 21 times larger number of spelling variant matches on the test dataset: from 41 to 865.

Of course, this comes at a cost: while all 41 spelling variant matches at threshold of 0.1 are correct, there are 8% errors among the 865 matches at threshold of 0.6. We believe this is a relatively small price to pay, given the advantage of being able to identify 791 additional correct matches, which we capture without the need for having multiple references.[7]

Pearson correlation. We further measured the correlation between WER/MR-WER vs. TER/WERd. We first calculated the scores for WER/MR-WER/TER/WERd for each of the 1,217 test utterances in isolation, and then we calculated the Pearson correlation using the corresponding lists of utterance-level scores. The results are shown in Table 5. We can see that WERd correlates better than TER with both WER and also with MR-WER. Overall, we can conclude that WERd is a promising measure for evaluating ASR systems that target dialectal speech input.

[7] We were unable to measure the recall as it requires manual evaluation of all the possible candidates for spelling variants in the references.

Metrics Compared	Correlation
WER vs. TER	0.44
WER vs. WERd	0.47
MR-WER vs. TER	0.36
MR-WER vs. WERd	0.39

Table 5: Pearson correlations.

Closer look at the spelling variants used in test. Finally, we had a closer look at the 865 spelling variant pairs that were matched and used when calculating WERd for the test set of 1,217 segments, when using edit distance of 0.6. Our analysis shows three types of word-level changes:

1. *Word splitting*: 3% of the pairs
   e.g., مفيش (mfy$) → ما فيش (mA fy$).

2. *Word merging*: 16% of the pairs
   e.g., زي ما حنا (zy mA HnA) → زي ماحنا (zy mAHnA).

3. *Word substitution*: 81% of the pairs
   e.g., الامريكان (AlAmrykAn) → الاميركان (AlAmyrkAn).

These statistics show that we learn many useful spelling variants, i.e., more than 80%, rather than just splitting and merging words. Moreover, note that these word-level substitutions are actually small character-level transformations inside words. Tables 6 and 7 show some examples of correct and wrong spelling variant pairs that were matched when calculating WERd for our Dialectal Arabic test set.

Table 8 further shows how spelling variant pairs affect hypothesis scoring for an example test sentence. There are three spelling variant pairs that match the input ASR hypothesis ما فيش → مفيش, الامريكيه → الاميركيه, and finally علشان → عشان. The former involves word splitting, while the latter two are about substitution. We can see that the WER after using spelling variant substitutions would go down to 30%, (the actual WERd score would be slightly higher as it needs to take the cost of the spelling variant substitutions into account), while the initial WER was 61.5%.

English Gloss	Spelling Variants		Operation
Netanyahu	نتانياهو	ntAnyAhw	word substitution
	نتنياهو	ntnyAhw	
as we are	زي ما حنا	zy mA HnA	word merging
	زي ماحنا	zy mAHnA	
talking	بتتكلم	bttklm	word substitution
	تتكلم	ttklm	
like (as if)	يعني	yEny	word splitting
	يعني ب	yEny b	

Table 6: Correctly accepted spelling variants in test.

English Gloss	Spelling Variants		Operation
some	بعد	bEd	word substitution
after	بعض	bED	
principal	رئيسي	r}ysy	word substitution
president	رئيس	r}ys	

Table 7: Wrongly accepted spelling variants in test.

Hypothesis (before): مفيش هم من مصر من الولايات المتحده الاميركيه عشان
mfy$ hm mn mSr mn AlwlAyAt AlmtHdh AlAmyrkyh E$An
WER: 61.54 [8/13; 0 insertions, 4 deletions, 4 substitutions]

Hypothesis (after): ما فيش هم من مصر من الولايات المتحده الامريكيه علشان
mA fy$ hm mn mSr mn AlwlAyAt AlmtHdh AlAmrykyh El$An
WER: 30.77 [4/13; 0 insertions, 3 deletions, 1 substitutions]

Reference: ما فيش زيهم جم من مصر وجم من كل الولايات المتحده الامريكيه علشان
mA fy$ zyhm jm mn mSr wjm mn kl AlwlAyAt AlmtHdh AlAmrykyh El$An

Table 8: Extra word matches due to using spelling variants. Shown is an ASR hypothesis for a test utterance, and the impact of hypothesis matching on the number of insertions, deletions and substitutions, as well as on the overall WER score.

5. CONCLUSION AND FUTURE WORK

We have addressed the evaluation of ASR systems that target dialectal speech input, where a major problem is the natural variation in spelling due to the unofficial status and the lack of standardization of the orthography. We have proposed a new metric, WERd (or *WER for dialects*), a variation of TERp, for which multiple text outputs for the same speech signal can be acceptable given a single reference transcript. Our implementation of WERd was based on mining 11M pairs of spelling variants from a huge dialectal Arabic tweet collection. Our automatic experiments, as well as manual analysis, have shown that this is a highly promising metric that addresses the problems of WER for dialectal speech, and approaches the performance of multi-reference WER.

In future work, we plan experiments with other dialects and non-standardized language varieties. We also want to incorporate word embeddings in the process of computation, e.g., character-based, which can naturally tolerate some spelling variation [35]. We further want to explore using weighted finite state transducers as an alternative way to allow using multiple spelling variants for both references and hypotheses.

Last but not least, we have made publicly available our code together with our data and the spelling variants we have mined. We hope that this will enable further research in ASR evaluation for languages with non-standardized orthography.

6. ACKNOWLEDGEMENTS

This work was partially supported by the EU H2020 project SUMMA, under grant agreement 688139.

7. REFERENCES

[1] George Saon, Gakuto Kurata, Tom Sercu, Kartik Audhkhasi, Samuel Thomas, Dimitrios Dimitriadis, Xiaodong Cui, Bhuvana Ramabhadran, Michael Picheny, Lynn-Li Lim, Bergul Roomi, and Phil Hall, "English conversational telephone speech recognition by humans and machines," *arXiv preprint arXiv:1703.02136*, 2017.

[2] Ahmed Ali, Stephan Vogel, and Steve Renals, "Speech Recognition Challenge in the Wild: Arabic MGB-3," in *Proceedings of the IEEE Workshop on Automatic Speech Recognition and Understanding*, Okinawa, Japan, 2017, ASRU '17.

[3] Ronald Rosenfeld and Philip Clarkson, "CMU-Cambridge statistical language modeling toolkit v2," 1997.

[4] Jonathan Fiscus, "A post-processing system to yield reduced word error rates: Recognizer output voting error reduction (ROVER)," in *Proceedings of the IEEE Workshop on Automatic Speech Recognition and Understanding*, Santa Barbara, California, USA, 1997, ASRU '97, pp. 347–354.

[5] Ahmed Ali, Walid Magdy, Peter Bell, and Steve Renals, "Multi-reference WER for evaluating ASR for languages with no orthographic rules," in *Proceedings of the IEEE Workshop on Automatic Speech Recognition and Understanding*, Scottsdale, Arizona, USA, 2015, ASRU '15, pp. 576–580.

[6] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu, "BLEU: A method for automatic evaluation of machine translation," in *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, Philadelphia, Pennsylvania, USA, 2002, ACL '02, pp. 311–318.

[7] Nitin Madnani, Necip Fazil Ayan, Philip Resnik, and Bonnie Dorr, "Using paraphrases for parameter tuning in statistical machine translation," in *Proceedings of the Second Workshop on Statistical Machine Translation*, Prague, Czech Republic, 2007, WMT '07, pp. 120–127.

[8] Chris Callison-Burch, Philipp Koehn, and Miles Osborne, "Improved statistical machine translation using paraphrases," in *Proceedings of the Conference on Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, New York, New York, USA, 2006, NAACL-HLT '06, pp. 17–24.

[9] Preslav Nakov, "Improving English-Spanish statistical machine translation: Experiments in domain adaptation, sentence paraphrasing, tokenization, and recasing," in *Proceedings of the Third Workshop on Statistical Machine Translation*, Columbus, Ohio, USA, 2008, WMT '08, pp. 147–150.

[10] Preslav Nakov, "Improved statistical machine translation using monolingual paraphrases," in *Proceedings of the 18th European Conference on Artificial Intelligence*, Patras, Greece, 2008, ECAI '08, pp. 338–342.

[11] Preslav Nakov and Hwee Tou Ng, "Translating from morphologically complex languages: A paraphrase-based approach," in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, Portland, Oregon, USA, 2011, ACL '11, pp. 1298–1307.

[12] Pidong Wang, Preslav Nakov, and Hwee Tou Ng, "Source language adaptation for resource-poor machine translation," in *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, Jeju Island, Korea, 2012, EMNLP-CoNLL '12, pp. 286–296.

[13] Pidong Wang, Preslav Nakov, and Hwee Tou Ng, "Source language adaptation approaches for resource-poor machine translation," *Comput. Linguist.*, vol. 42, no. 2, pp. 277–306, June 2016.

[14] Liang Zhou, Chin-Yew Lin, and Eduard Hovy, "Re-evaluating machine translation results with paraphrase support," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, Sydney, Australia, 2006, EMNLP '06, pp. 77–84.

[15] David Kauchak and Regina Barzilay, "Paraphrasing for automatic evaluation," in *Proceedings of the Conference on Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, New York, New York, USA, 2006, NAACL-HLT '06, pp. 455–462.

[16] Matthew G. Snover, Nitin Madnani, Bonnie Dorr, and Richard Schwartz, "TER-Plus: Paraphrase, semantic, and alignment enhancements to translation edit rate," *Machine Translation*, pp. 117–127, 2009.

[17] Kareem Darwish, Walid Magdy, and Ahmed Mourad, "Language processing for Arabic microblog retrieval," in *Proceedings of the 21st ACM International Conference on Information and Knowledge Management*, Maui, Hawaii, USA, 2012, CIKM '12, pp. 2427–2430.

[18] Hany Hassan and Arul Menezes, "Social text normalization using contextual graph random walks," in *Proceedings of the Annual Meeting on Association for Com-*

putational Linguistics, Sofia, Bulgaria, 2013, ACL '13, pp. 1577–1586.

[19] Bo Han, Paul Cook, and Timothy Baldwin, "Automatically constructing a normalisation dictionary for microblogs," in *Proceedings of the Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, Jeju Island, Korea, 2012, EMNLP-CoNLL '12, pp. 421–432.

[20] Vivek Kumar Rangarajan Sridhar, "Unsupervised text normalization using distributed representations of words and phrases," in *Proceedings of the 1st Workshop on Vector Space Modeling for Natural Language Processing*, Denver, Colorado, USA, 2015, pp. 8–16.

[21] Richard Sproat and Navdeep Jaitly, "RNN approaches to text normalization: A challenge," *arXiv preprint arXiv:1611.00068*, 2016.

[22] Matthew Snover, Bonnie Dorr, Richard Schwartz, Linnea Micciulla, and John Makhoul, "A study of translation edit rate with targeted human annotation," in *Proceedings of Association for Machine Translation in the Americas*, Cambridge, Massachusetts, USA, 2006, AMTA '06.

[23] Samantha Wray and Ahmed Ali, "Crowdsource a little to label a lot: Labeling a speech corpus of dialectal Arabic," in *Proceedings of the 16th Annual Conference of the International Speech Communication Association*, Dresden, Germany, 2015, INTERSPEECH '15, pp. 2824–2828.

[24] Sameer Khurana and Ahmed Ali, "QCRI advanced transcription system (QATS) for the Arabic multi-dialect broadcast media recognition: MGB-2 challenge," in *Proceedings of the IEEE Workshop on Spoken Language Technology*, San Diego, California, USA, 2016, SLT '16, pp. 292–298.

[25] Ahmed Ali, Peter Bell, James Glass, Yacine Messaoui, Hamdy Mubarak, Steve Renals, and Yifan Zhang, "The MGB-2 challenge: Arabic multi-dialect broadcast media recognition," in *Proceedings of the IEEE Workshop on Spoken Language Technology*, San Diego, California, USA, 2016, SLT '2016, pp. 279–284.

[26] Tom Ko, Vijayaditya Peddinti, Daniel Povey, and Sanjeev Khudanpur, "Audio augmentation for speech recognition," in *Proceedings of the 16th Annual Conference of the International Speech Communication Association*, Dresden, Germany, 2015, INTERSPEECH '15, pp. 3586–3589.

[27] Mirjam Killer, Sebastian Stüker, and Tanja Schultz, "Grapheme based speech recognition," in *Proceedings of the 8th European Conference on Speech Communication and Technology*, Geneva, Switzerland, 2003, EUROSPEECH–INTERSPEECH '03.

[28] George Saon, Hagen Soltau, David Nahamoo, and Michael Picheny, "Speaker adaptation of neural network acoustic models using i-vectors," in *Proceedings of the IEEE Workshop on Automatic Speech Recognition and Understanding*, Olomouc, Czech Republic, 2013, ASRU '13, pp. 55–59.

[29] Vijayaditya Peddinti, Daniel Povey, and Sanjeev Khudanpur, "A time delay neural network architecture for efficient modeling of long temporal contexts," in *Proceedings of the 16th Annual Conference of the International Speech Communication Association*, Dresden, Germany, 2015, INTERSPEECH '15, pp. 3214–3218.

[30] Haşim Sak, Andrew Senior, and Françoise Beaufays, "Long short-term memory recurrent neural network architectures for large scale acoustic modeling," in *Proceedings of the 15th Annual Conference of the International Speech Communication Association*, Singapore, 2014, INTERSPEECH '14, pp. 338–342.

[31] Daniel Povey, Vijayaditya Peddinti, Daniel Galvez, Pegah Ghahremani, Vimal Manohar, Xingyu Na, Yiming Wang, and Sanjeev Khudanpur, "Purely sequence-trained neural networks for ASR based on lattice-free MMI," in *Proceedings of the 17th Annual Conference of the International Speech Communication Association*, San Francisco, California, USA, 2016, INTERSPEECH '16, pp. 2751–2755.

[32] Daniel Povey, Arnab Ghoshal, Gilles Boulianne, Lukas Burget, Ondrej Glembek, Nagendra Goel, Mirko Hannemann, Petr Motlicek, Yanmin Qian, Petr Schwarz, et al., "The Kaldi speech recognition toolkit," in *Proceedings of the IEEE Workshop on Automatic Speech Recognition and Understanding*, Waikoloa, Hawaii, USA, 2011, ASRU '11.

[33] Andreas Stolcke et al., "SRILM - an extensible language modeling toolkit," in *Proceedings of the 7th International Conference on Spoken Language Processing*, Denver, Colorado, USA, ICSLP-INTERSPEECH '02.

[34] Tomas Mikolov, Stefan Kombrink, Anoop Deoras, Lukar Burget, and Jan Cernocky, "RNNLM - recurrent neural network language modeling toolkit," in *Proceedings of the IEEE Workshop on Automatic Speech Recognition and Understanding*, Big Island, Hawaii, USA, 2011, ASRU '11, pp. 196–201.

[35] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov, "Enriching word vectors with subword information," *TACL*, vol. 5, pp. 135–146, 2017.