Automatic Error Analysis for Morphologically Rich Languages

Ahmed El Kholy and **Nizar Habash**
Center for Computational Learning Systems, Columbia University
475 Riverside Drive New York, NY 10115
{akholy,habash}@ccls.columbia.edu

Abstract

This paper presents AMEANA, an open-source tool for error analysis for natural language processing tasks targeting morphologically rich languages. Unlike standard evaluation metrics such as BLEU or WER, AMEANA automatically provides a detailed error analysis that can help researchers and developers better understand the strengths and weaknesses of their systems. AMEANA is easily adaptable to any language provided the existence of a morphological analyzer. In this paper, we focus on usability in the context of Machine Translation (MT) and demonstrate it specifically for English-to-Arabic MT.

1 Introduction

Error analysis is a central part of the research process in natural language processing (NLP). Through error analysis, researchers and developers can better understand the strengths and weaknesses of their systems. The more detailed the analysis, the more specific the insights can be. Morphologically rich languages, such as Arabic, Turkish or German, are particularly challenging since there is a large space of possible details to explore at the word morphology level. Human evaluation is an attractive solution; however, it typically involves qualitative measures, such as fluency or adequacy, which are very generic and do not capture nor quantify word-level errors. Fine-grained human analysis suffers from low speed, high cost and low consistency due to fatigue and adaptation to machine-generated language. Automating error analysis is a good solution, although simple matching techniques can be too coarse to be helpful.

In this paper, we present AMEANA (Automatic Morphological Error Analysis),[1] an automatic error analysis tool that is designed to identify morphological errors in the output of a given system against a gold reference. AMEANA produces detailed statistics on morphological errors in the output. It also generates an oracularly modified version of the output that can be used to measure the effect of these errors using any evaluation metric. AMEANA is a language independent tool except that a morphological analyzer must be provided for a given language.

Although AMEANA can be used in a variety of NLP tasks involving text generation, we focus here on usability in the context of Machine Translation (MT) and demonstrate it specifically for English-to-Arabic MT. Most published research on MT targets translation into English, a morphologically poor language; however, MT into languages with richer morphology has been receiving increasing attention (Oflazer and Durgar El-Kahlout, 2007; El Kholy and Habash, 2010; Williams and Koehn, 2011).

Sections 2 and 3 present the motivation of this work and related efforts, respectively. Section 4 discusses our approach to building AMEANA. Sections 5 and 6 report on a case study including detailed analysis and verification.

2 Motivation

Most MT automatic evaluation metrics, such as BLEU (Papineni et al., 2002), focus on comparing an MT output against a set of references in order to assign a similarity score. The scores are typically based on exact word matching, a particularly harsh measure especially for morphologically rich languages. This is due in part to two phenomena. First is **Morphological Richness:** words sharing the same core meaning (represented by the lemma or lexeme) can be said to inflect for different morphological features, e.g., gender and number. These features can realize using concatenative (affixes and stems) and/or templatic (root and patterns) morphol-

[1]The first author was funded by a Google research award. Please contact the authors to get a copy of the tool.

ogy. Second is **Morphological Ambiguity:** words with different lemmas can have the same inflected form. As such, a word form can have more than one morphological analysis (represented as a lemma and a set of feature-value pairs). This is especially problematic for languages with reduced orthographies such as Arabic or Hebrew.

Using an abstraction of the word, such as the stem or the lemma, to match output and reference words can address the harshness of exact form matching. Stemming has been shown to be helpful in MT evaluation (Denkowski and Lavie, 2010); but simple stemming is not sufficient when dealing with morphologically rich languages as it suffers from errors of omission and errors of commission (Krovetz, 1993): words with the same core meaning not sharing the same stem, and words with different core meanings sharing the same stem. This is especially problematic for words with templatic morphology, e.g., broken plurals in Arabic.[2] Furthermore, simple stemming does not properly address ambiguity as most shallow stemmers do not provide more than one stem for a given word. A more sophisticated stemming approach using a morphological analyzer can address this limitation. AMEANA can be used with stems, lemmas or even higher abstractions relating different lemmas to each other. In the case study we present on Arabic, we use the lemma representations produced by a morphological analyzer because of the above-mentioned limitations of stemming. We plan to study the use of higher abstractions in the future.

Form abstraction, however, is a double edged sword since it will lead to numerous matching points between the output and reference. To address this concern, AMEANA uses a word matching (alignment) algorithm that minimizes the number of morphological differences and sentence-relative word position.

3 Related Work

Recent efforts reported on improving the quality of MT evaluation using stemming and/or paraphrasing to match output reference translations (Denkowski and Lavie, 2010; Snover et al., 2009). None of these techniques provide detailed error analyses at the morphological level.

Several publications defined different error classifications and typologies for the purpose of evaluation of single systems, or comparison between systems (Flanagan, 1994; Vilar et al., 2006; Farrús et al., 2010). Kirchhoff et al. (2007) developed a framework for semi-automatically analyzing characteristics of input documents to MT systems that determine output performance. The framework heavily depends on human annotation.

To our knowledge, there haven't been many efforts to build publicly available error analysis tools for MT output with focus on rich morphology. Popović and Ney (2006) provided precision and recall measures of MT output for different verbal inflections, but they only focus on Spanish verbs. Their word matching technique is a based on *PER* which may not be sufficient to apply in more general settings (i.e., not just verbs).

Tantug et al. (2008) created a tool which is closely related to our work. They extended the BLEU and METEOR metrics to handle errors in Turkish morphology. Their matching algorithm uses Turkish word roots and a wordnet hierarchy, and it produces oracle score comparable to what AMEANA does.

Stymne (2011) presented a tool for annotation of bilingual segments intended for error analysis of MT. It utilizes a given error typology to annotate translations from an MT system. The tool does not provide detailed morphological error analysis.

4 Approach

In this section, we describe the algorithm used in aligning the output words with their matching reference words. The alignment is then used to produce detailed morphological-error diagnostics and an oracularly modified output to use with MT evaluation metrics. A sample of these diagnostics is shown in Section 6.

4.1 Alignment Algorithm

For every sentence pair of MT output and its reference translation, we apply the following alignment algorithm (see Figure 1):

First: Morphological Analysis We run a morphological analyzer on all output and reference

[2] In broken plurals, the functional number (plural) is inconsistent with the morphological ending (singular suffix) (Alkuhlani and Habash, 2011). Plurality is indicated using a word template realized as a stem that is different from the singular stem.

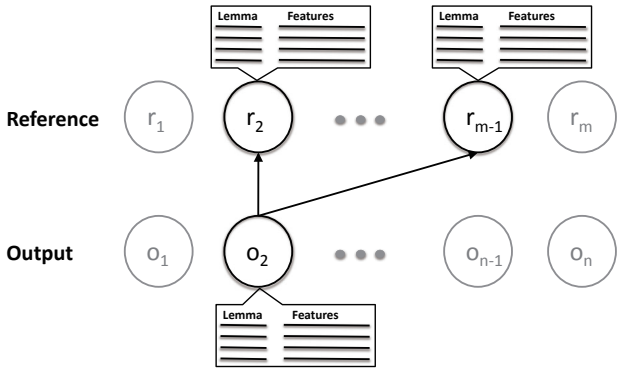

Figure 1: Output word o_2 has at least one common lemma with reference words r_2 and r_{m-1}. Our alignment algorithm selects edges minimizing differences in features (primarily) and relative positions (secondarily), while maximizing the number of paired output-reference words.

words producing a set of lemmas and their associated analyses for each word. A morphological disambiguator or part-of-speech (POS) tagger can be used to limit the choices given to AMEANA, e.g., (Habash and Rambow, 2005). This is not required and perhaps even not desirable given error propagation resulting from running a disambiguator on automatically generated text.

Second: Graph Construction We build a graph where each word is represented by a node. We draw an edge for each output-reference word pair if there is at least one common lemma between them. Each edge receives a weight based on the following equation:

$$W \;=\; \min\left(D_{ab}\right) + \frac{\left(\left|\frac{P_a}{S_a} - \frac{P_b}{S_b}\right|\right)}{2}$$

We define a and b as words in output and reference. For each pair of morphological analyses for a and b sharing the same lemma, we compute the count of features with different values. We define D_{ab} as the set of all feature-difference counts. Consequently, $\min\left(D_{ab}\right)$ is the minimum feature difference possible between words a and b. We define P_a and P_b as the position of words a and b in their respective sentences. We also define S_a and S_b as the lengths of the sentences in which a and b appear, respectively. The absolute difference in relative word position $\left|\frac{P_a}{S_a} - \frac{P_b}{S_b}\right|$ is used as a tie breaker. It is divided by 2 to account for the extreme case of matching words at opposite ends of their respective sentences. The smaller the value W, the closer the two words a and b are to each other. In this equation, we give more weight to feature differences by giving a whole point for each mismatching feature, while word position distance is used as a tie breaker.

Third: Bipartite Matching Once the graph is constructed, the search space for the alignment is defined as a maximum bipartite matching problem constrained on the weights of the edges. We use a modified version of the Ford-Fulkerson algorithm (Ford and Fulkerson, 1956) to solve the matching problem and select a number of edges that maximizes the number of aligned output-reference words.

After alignment, each output word receives a matching category based on the reference word it is paired with. If the output and reference words have same form, the category is *Exact Match*, otherwise, it is *Lemma Match*. Unpaired output words receive the category *Unmatchable*.

4.2 Morphological Diagnostics

We sum over all the feature differences in the *Lemma Match* category words. In cases with multiple analyses with the same lemma and same minimum feature-difference count, we assign equal partial error to each analysis so that they sum up to 1 instead of choosing among them. The partial errors are aggregated for each feature difference in all analyses. We will generically refer to feature differences as *errors* with respect to the reference, although some may not actually be erroneous (albeit not directly matching).

AMEANA produces general statistics such as the number and percentage of *Exact Match*, *Lemma Match* and *Unmatchable* words; the average number of errors per sentence; and the number of sentences with a certain number of errors. Detailed statistics are produced for errors in *Lemma Match* words including the number and percentage of errors for all features, feature-value pairs, and their combination. Additionally, AMEANA produces precision, recall and F-scores for correctly generating the various features. See Section 6 for some examples of these statistics.

4.3 Use for MT Evaluation

One of the side benefits provided by AMEANA is the production of an oracularly modified MT out-

put where output words with a *Lemma Match* are replaced with the reference words they are aligned to. The modified output can be run through any evaluation metric such as BLEU or METEOR to get the upper limit of improvement the system can reach by just making better morphological choices. AMEANA also gives the user the option of restricting the oracle generation such that certain morphological features, in addition to the lemma, must correctly match the reference.

4.4 AMEANA Language Independence

As mentioned above, AMEANA is a language-independent error-analysis tool. To use AMEANA for a particular language, the user must specify the following parameters in a simple and easy to use configuration file:

- The output of a morphological analyzer run on the MT output and the reference.
- The tag marking the lemma in the morphological analyzer output and the list of morphological features to consider in the error analysis.
- A list of features to focus on in oracle generation, if desired.

5 Case Study: AMEANA for Arabic

In this section and the following section, we work with an English-to-Arabic Statistical MT system as a case study to show the different error-analysis outputs of AMEANA and to verify its performance. Since Arabic is the target language of the MT system we use, we first discuss relevant aspects of Arabic morphology, and how we adapt AMEANA to work with Arabic.

5.1 Arabic Morphology

Arabic is a morphologically rich language with a large set of morphological features such as person, number, gender, voice, aspect, mood, case, and state. Arabic features are realized using both concatenative (affixes and stems) and templatic (root and patterns) morphology with a variety of morphological, phonological and orthographic adjustments. One aspect of Arabic that contributes to its richness is the various attachable clitics. There are three degrees of cliticization that apply in a strict order to a word base: `[cnj+ [prt+ [det+ BASE +pro]]]`. At the deepest level, the BASE can have either the determiner (+ال *Al*+[3] 'the') or a member of the class of pronominal enclitics, `+pro`, (e.g., +هم *hm*+ 'their/them'). Next comes the class of particle proclitics (prt+), e.g., +لـ *l*+ 'to/for'. At the shallowest level of attachment we find the conjunction proclitic (cnj+), e.g., +و *w*+ 'and'. All these clitics are part of the word morphology and they are included in our error analysis. The Penn Arabic Treebank (PATB) tokenization scheme (Maamouri et al., 2004) which we use in this paper separates all clitics except for the determiner *Al*+.

Arabic morphological richness leads to thousands of inflected forms per lemma and a high degree of ambiguity: about 12 analyses per word, typically corresponding to two lemmas on average (Habash, 2010).

5.2 Adapting AMEANA to work with Arabic

In order to make AMEANA work for Arabic, we have to modify the configuration file to specify the parameters mentioned in Section 4.4. For the morphological analyzer, we use ALMORGEANA (ALMOR) (Habash, 2007). ALMOR is a morphological analysis and generation system for Arabic. It provides analyses of a given word based on the lemma-and-features level of representation which is what we want as an input for AMEANA.

6 Results

In this section, we present two sets of results: a demonstration of the use of AMEANA for MT error analysis and a study verifying its behavior.

6.1 Machine Translation Error Analysis

We ran AMEANA on the output of three SMT systems based on previous work on English-Arabic SMT (El Kholy and Habash, 2010). We present next the experimental settings of the MT systems. Then we present four sets of results produced by AMEANA to demonstrate its usability.

6.1.1 MT Experimental Settings

All systems share the following settings. They use the Moses phrase-based SMT decoder (Koehn et al.,

[3] All Arabic transliterations are presented in the HSB scheme (Habash et al., 2007)

English	Erdogan states Turkey to reject any pressures to urge it to recognize Cyprus.
Reference	أردوغان يؤكد بأن تركيا سترفض أي ضغوطات لحثها على الاعتراف بقبرص . *Ârdwγ An yŵkd bÂn trkyA strfD Ây DγwTAt lHθhA ς lý AlAςtrAf bqbrS .*
MT Output	اردوجان أن تركيا دولة ترفض أي ضغط لدفعها للاعتراف بقبرص . *ArdwjAn Ân trkyA dwlħ trfD Ây DγT ldfςhA llAςtrAf bqbrS .*
Modified MT	اردوجان بأن تركيا دولة سترفض أي ضغوطات لدفعها الاعتراف بقبرص . *ArdwjAn bÂn trkyA dwlħ strfD Ây DγwTAt ldfςhA AlAςtrAf bqbrS .*

MT Output	ArdwjAn	Ân	trkyA	dwlħ	trfD	Ây	DγT	ldfςhA	llAςtrAf	bqbrS	.
Modified MT	ArdwjAn	bÂn	trkyA	dwlħ	strfD	Ây	DγwTAt	ldfςhA	AlAςtrAf	bqbrS	.
Match Category	UM	LEM	Exact	UM	LEM	Exact	LEM	UM	LEM	Exact	Exact
MT Features		Part:ϕ			Part:ϕ		Gen:M,Num:S		Part:*li+*		
Reference Features		Part:*bi+*			Part:*sa+*		Gen:F,Num:P		Part:ϕ		

Table 1: AMEANA word-by-word error analysis. The first four rows specify the English input, Arabic reference, MT output and oracularly modified MT output, respectively. The second half of the table lists every word in the MT output (column 1) with the reference word used to modify it (column 2). Column 3 specifies the reference-match category: exact match indicates the MT output word appears in the reference; unmatchable indicates no match is found; and lemma match indicates a lemma-level match. For lemma match cases, the differences in MT output and reference morphological features are specified in columns 4 and 5.

2007) trained on an English-Arabic parallel corpus of about 135k sentences (4 million words). Phrase-table maximum phrase length is 8. Word alignment is done using GIZA++ (Och and Ney, 2003) run on the lemma level of representation. Lemmatization as well as tokenization (discussed below) is done using the MADA+TOKAN toolkit (Habash and Rambow, 2005). A 5-gram language model is based on 200M words from the Arabic Gigaword together and the Arabic side of the training data (Stolcke, 2002). Decoding weight optimization (Och, 2003) is done using 300 sentences from the 2004 NIST MT evaluation test set (MT04). Systems are compared on their performance on the 2005 NIST MT evaluation set (MT05). This Arabic-English test set has four English references. We invert it by selecting the first English reference to be our input and use the Arabic side as the only reference.

The three systems vary as follows: the D0 system uses no morphological tokenization whatsoever; the TB system uses the PATB tokenization scheme (Maamouri et al., 2004); and the LEM system uses PATB tokenization and keeps the main word in lemma form. TB has been previously shown to outperform D0; and LEM is the lemmatized version of TB used in TB's word alignment (El Kholy and Habash, 2010). We expect LEM to under perform compared to the other two systems. The first three columns of Table 4 show automatic evaluation scores in three metrics for all systems.

6.1.2 Overall Lemma Match Statistics

Table 1 shows the AMEANA output of one sentence from the TB system. The first four lines are the English input sentence, Arabic translation reference, MT output, and the AMEANA modified MT output, respectively. Following that is word-by-word analysis in the following format. The first row is the original MT output words in sequence and the second row is the modified MT words. Third row is the matching category while the fourth and fifth rows are the morphological features differences between the MT output word and its reference translation word when the matching category is *Lemma Match*.

Table 2 shows the numbers and percentage of words in each matching category for the three systems. *Exact Match* is the simplest statistics that can be obtained using any MT evaluation metric, e.g., it is a sub-score used in BLEU. AMEANA allows us to distinguish a subset of no matches that can be matched at the lemma level. This allows to quantify the percentage of words that have no lexical translation problems (since they have matching lemmas) and identify the subset that has feature problems even though the lemma is correct. Such distinction may be useful for techniques involving post-editing or word-repair. The D0 and TB systems have simi-

lar *Exact Match* percentages. In both systems about one-third of the *non-Exact Match* cases have matchable lemmas. LEM has a much lower *Exact Match* but also a much higher *Lemma Match*. LEM overall has the highest *Any Match* (includes *Exact Match* & *Lemma Match*), which suggests it has the highest lexical translation quality despite its low fluency.

	D0	TB	LEM
Output Word Count	28,126	28,816	28,759
Exact Match (%)	58.0	59.0	38.7
Lemma Match (%)	13.9	13.3	33.8
Any Match (%)	72.0	72.3	72.6
Unmatchable (%)	28.0	27.7	27.4

Table 2: Unigram analysis of three English-to-Arabic SMT systems

6.1.3 Lemma Match Error Distribution

Table 3(a) presents the percentage of matching errors among the *Lemma Match* words, which are only about a seventh (D0, TB) or a third (LEM) of all words. The errors are classified by feature, e.g., conjunction, determiner, or gender; and by two feature-classes: *PATB clitics* and *other features*. This table allows us to study the distribution of various error types per system. Comparing across systems must take into account the size of the *Lemma Match* set of words. For example, although LEM has a lower percentage of pronominal clitics than TB or D0, it actually has 40% more instances of errors. Overall, these numbers show that the determiner is the biggest single feature error across systems. Non-PATB clitic errors collectively constitute a smaller proportion of matching errors than other word features, although the difference between the two sets gets smaller in our best performer, TB. The PATB clitics together with determiner, gender and number are biggest culprits overall. This analysis suggests targeting them may be most beneficial. Some features have low counts because they are associated with specific POS which are less frequent, e.g., verbal mood, voice and aspect.

6.1.4 Morphological Feature Correctness

Table 3(b) presents the F-measure (balanced harmonic mean of the precision and recall) of words matching between the output and the reference for a variety of matching criteria of morphological features. The last two rows are for *Exact Match* and *Any Match*. These two can be interpreted as the lowest and highest limits on matching given the space of morphological errors. While *Exact Match* requires the lemma and all features to match, *Any Match* only requires the lemma to match – of course, in *Exact Match*, the lemma matches by definition. The rest of the rows are for matching subsets that include the lemma together with a particular feature, such as conjunction or determiner. These numbers are not oracle scores, they reflect the correctness of the text on different morphological features even if the final word form is not matchable.

	(a) Error Type %			(b) Match F-score %		
	D0	TB	LEM	D0	TB	LEM
PATB Clitics	52.9	53.6	24.3	63.9	65.3	64.4
Conjunction	20.2	18.6	7.4	68.4	69.9	70.1
Particle	23.1	24.3	10.5	68.0	69.2	69.0
Pronoun	15.1	15.7	8.7	69.1	70.3	69.6
Other Features	61.1	57.8	84.6	62.8	64.7	43.9
Determiner	31.0	29.7	60.0	66.9	68.4	52.3
Gender	14.3	12.8	20.5	69.2	70.7	65.7
Number	11.8	10.8	14.0	69.6	71.0	67.9
Person	4.0	4.0	3.4	70.7	71.9	71.4
Stem	3.6	4.1	3.9	70.7	71.9	71.3
Case	3.5	3.0	2.4	70.7	72.0	71.8
Aspect	2.2	2.1	3.1	70.9	72.1	71.5
State	1.0	0.8	0.8	71.1	72.3	72.3
Mood	0.8	0.7	0.4	71.1	72.3	72.5
Voice	0.2	0.2	0.4	71.2	72.4	72.5
Exact (Lemma+Feature) Match				57.4	59.1	38.7
Any (Lemma) Match				71.2	72.4	72.6

Table 3: (a) Comparison between the different morphological errors in the MT output in terms of their percentage of the total number of morphological errors and the percentage of total words in the given document. (b) Comparison of F-scores of words matching between the output and the reference for a list of morphological features.

Across all features, TB outperforms D0. This is consistent with their overall BLEU scores; however, it is interesting to see that the improvement in features other than PATB clitics is actually more than in PATB clitics overall (by 1.9% compared to 1.4%). The main area LEM is suffering compared to TB and D0 is in non-PATB clitics. This is expected given the lack of inflections in the output of LEM. Lemma plus determiner matching yields the lowest single F-score over than *Exact Match*. That said, it is about 70% of the way between *Exact Match* and

Any Match (for D0 and TB) (and 40% for LEM).

6.1.5 Oracle Generation for MT Evaluation

We evaluate the oracularly modified output using several MT evaluation metrics. Table 4 shows the difference in scores between the original MT output and the modified one. There are ≈7, 1.8 and 10.5, points difference in BLEU (Papineni et al., 2002), NIST (Doddington, 2002) and METEOR (Denkowski and Lavie, 2010) scores, respectively. These differences are the upper limits that a system can reach by making better morphological choices on the unigram level. It is important to keep in mind that some of these improvements are very hard to achieve and some are incorrect linguistically although they maximize the reference matching.

	Basic MT Output			**Oracle MT Output**		
	D0	TB	LEM	D0	TB	LEM
BLEU	25.47	29.48	10.20	33.38	35.57	35.69
NIST	6.8797	7.2681	3.9786	8.8220	8.9231	8.9266
METEOR	42.23	45.67	22.57	53.58	55.42	55.41

Table 4: Comparison between the original and the modified MT output in BLEU, NIST and METEOR metrics. METEOR is used in language-independent mode.

6.2 AMEANA Verification

To evaluate the performance of AMEANA, we conducted a manual verification of the alignment and error analysis produced by the tool. We looked at 20 sentences (509 words) from the TB MT system output discussed above. We found that 100% of the *Lemma Match* words are aligned correctly (100% precision). However, we found five cases where the words were unmatchable although there were good candidates in the reference translation. For instance, the word نبدأ *nbdÂ* 'we start' in the MT output should have been matched with the word ببدء *bbd'* 'with the start of' in the reference but it was categorized as unmatchable. AMEANA fails in these cases because the two words have different POS and there is no single common lemma between them. This failure in design is one of the issues that we would like to deal with in future work.

Another issue that we encountered is that even after a successful alignment, the errors detected may not be accurate. For example, the word صحف *SHf* 'newspapers' (a feminine broken plural with masculine singular surface morphology) was aligned to the word صحيفة *SHyfħ* 'newspaper' (feminine singular). The alignment is correct but the morphological errors detected are not accurate because they are based on surface morphology not functional morphology features (Smrž, 2007; Alkuhlani and Habash, 2011). In the 509 plus words we studied, we found only one case. Still of course, this is a particular limitation of the analyzer used. A different analyzer that addresses such issues can be used with AMEANA without a problem in the future, e.g., ElixirFM (Smrž, 2007).

7 Conclusion and Future Work

We presented AMEANA, a language independent tool for error analysis. It is a simple and elegant tool that could help developers and researchers better understand the strengths and weaknesses of their systems especially if they are targeting morphologically rich languages. The tool can work with any language that has a morphological analyzer.

In the future, we plan to make AMEANA work with multiple references instead of just a single reference. We also plan to work on the alignment component of the tool to be able to deal with words that have different POS or lemmas. We also plan to study the utility of our tool for the task of automatic evaluation.

References

Sarah Alkuhlani and Nizar Habash. 2011. A Corpus for Modeling Morpho-Syntactic Agreement in Arabic: Gender, Number and Rationality. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics (ACL'11)*, Portland, Oregon, USA.

Michael Denkowski and Alon Lavie. 2010. METEOR-NEXT and the METEOR Paraphrase Tables: Improved Evaluation Support For Five Target Languages. In *Proceedings of the ACL 2010 Joint Workshop on Statistical Machine Translation and Metrics MATR*.

George Doddington. 2002. Automatic evaluation of machine translation quality using n-gram co-occurrence statistics. In *Human Language Technology*, pages 128–132, San Diego.

Ahmed El Kholy and Nizar Habash. 2010. Orthographic and morphological processing for english-arabic statistical machine translation. In *Proceedings of Traite-*

ment Automatique du Langage Naturel (TALN-10). Montréal, Canada.

Mireia Farrús, Marta R. Costa-jussà, José B. Mariño, and José A. R. Fonollosa. 2010. Linguistic-based evaluation criteria to identify statistical machine translation errors. In *Proceedings of EAMT*, pages 52–57, Saint Raphael, France.

Mary Flanagan. 1994. Error classification for mt evaluation. In *Proceedings of AMTA*, pages 65–72, Columbia, Maryland, USA.

L. R. Ford and D. R. Fulkerson. 1956. Maximal Flow through a Network. *Canadian Journal of Mathematics*, 8:399–404.

Nizar Habash and Owen Rambow. 2005. Arabic Tokenization, Part-of-Speech Tagging and Morphological Disambiguation in One Fell Swoop. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 573–580, Ann Arbor, Michigan.

Nizar Habash, Abdelhadi Soudi, and Tim Buckwalter. 2007. On Arabic Transliteration. In A. van den Bosch and A. Soudi, editors, *Arabic Computational Morphology: Knowledge-based and Empirical Methods*. Springer.

Nizar Habash. 2007. Arabic Morphological Representations for Machine Translation. In A. van den Bosch and A. Soudi, editors, *Arabic Computational Morphology: Knowledge-based and Empirical Methods*. Springer.

Nizar Habash. 2010. *Introduction to Arabic Natural Language Processing*. Morgan & Claypool Publishers.

Katrin Kirchhoff, Owen Rambow, Nizar Habash, and Mona Diab. 2007. Semi-automatic error analysis for large-scale statistical machine translation systems. In *Proceedings of the Machine Translation Summit (MT-Summit)*, Copenhagen, Denmark.

P. Koehn, H. Hoang, A. Birch, C. Callison-Burch, M. Federico, N. Bertoldi, B. Cowan, W. Shen, C. Moran, R. Zens, C. Dyer, O. Bojar, A. Constantin, and E. Herbst. 2007. Moses: open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions*, pages 177–180, Prague, Czech Republic.

Robert Krovetz. 1993. Viewing morphology as an inference process. In *SIGIR'93*, pages 191–202.

Mohamed Maamouri, Ann Bies, Tim Buckwalter, and Wigdan Mekki. 2004. The Penn Arabic Treebank: Building a Large-Scale Annotated Arabic Corpus. In *NEMLAR Conference on Arabic Language Resources and Tools*, pages 102–109, Cairo, Egypt.

Franz Josef Och and Hermann Ney. 2003. A Systematic Comparison of Various Statistical Alignment Models. *Computational Linguistics*, 29(1):19–52.

Franz Josef Och. 2003. Minimum Error Rate Training for Statistical Machine Translation. In *Proceedings of the 41st Annual Conference of the Association for Computational Linguistics*, pages 160–167, Sapporo, Japan.

Kemal Oflazer and Ilknur Durgar El-Kahlout. 2007. Exploring different representational units in English-to-Turkish statistical machine translation. In *Proceedings of the Second Workshop on Statistical Machine Translation*, pages 25–32, Prague, Czech Republic, June. Association for Computational Linguistics.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a Method for Automatic Evaluation of Machine Translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, PA.

Maja Popović and Herman Ney. 2006. Error analysis of verb inflections in spanish translation output. In *TC-Star Workshop on Speech-to-Speech Translation*, pages 99–103, Barcelona, Spain, June.

Otakar Smrž. 2007. *Functional Arabic Morphology. Formal System and Implementation*. Ph.D. thesis, Charles University in Prague, Prague, Czech Republic.

Matthew Snover, Nitin Madnani, Bonnie Dorr, and Richard Schwartz. 2009. Fluency, adequacy, or hter? exploring different human judgments with a tunable mt metric. In *Proceedings of the Fourth Workshop on Statistical Machine Translation at the 12th Meeting of the European Chapter of the Association for Computational Linguistics (EACL-2009)*, Athens, Greece, March.

Andreas Stolcke. 2002. SRILM - an Extensible Language Modeling Toolkit. In *Proceedings of the International Conference on Spoken Language Processing (ICSLP)*, volume 2, pages 901–904, Denver, CO.

Sara Stymne. 2011. Blast: A tool for error analysis of machine translation output. In *ACL 2011 demonstration session*, Portland, Oregon.

A. Cuneyd Tantug, Kemal Oflazer, and Ilknur Durgar El-Kahlout. 2008. BLEU+: a Tool for Fine-Grained BLEU Computation. In *Proceedings of the Sixth International Language Resources and Evaluation (LREC'08)*, Marrakech, Morocco.

David Vilar, Jia Xu, Luis Fernando D'Haro, and Hermann Ney. 2006. Error analysis of machine translation output. In *Proceedings of LREC*, pages 697–702, Genoa, Italy.

Philip Williams and Philipp Koehn. 2011. Agreement constraints for statistical machine translation into german. In *Proceedings of the Sixth Workshop on Statistical Machine Translation*, pages 217–226, Edinburgh, Scotland, July. Association for Computational Linguistics.