Error Correction for Arabic Dictionary Lookup

C. Anton Rytting, Paul Rodrigues, Tim Buckwalter, David Zajic, Bridget Hirsch, Jeff Carnes, Nathanael Lynn, Sarah Wayland, Chris Taylor, Jason White, Charles Blake III, Evelyn Browne, Corey Miller, Tristan Purvis

University of Maryland
Center for Advanced Study of Language (CASL)
College Park, MD
E-mail: {crytting,prodrigues}@casl.umd.edu

Abstract

We describe a new Arabic spelling correction system which is intended for use with electronic dictionary search by learners of Arabic. Unlike other spelling correction systems, this system does not depend on a corpus of attested student errors but on student- and teacher-generated ratings of confusable pairs of phonemes or letters. Separate error modules for keyboard mistypings, phonetic confusions, and dialectal confusions are combined to create a weighted finite-state transducer that calculates the likelihood that an input string could correspond to each citation form in a dictionary of Iraqi Arabic. Results are ranked by the estimated likelihood that a citation form could be misheard, mistyped, or mistranscribed for the input given by the user. To evaluate the system, we developed a noisy-channel model trained on students' speech errors and use it to perturb citation forms from a dictionary. We compare our system to a baseline based on Levenshtein distance and find that, when evaluated on single-error queries, our system performs 28% better than the baseline (overall MRR) and is twice as good at returning the correct dictionary form as the top-ranked result. We believe this to be the first spelling correction system designed for a spoken, colloquial dialect of Arabic.

1. Introduction

Non-native learners of Arabic attempting to use electronic lexicons make errors in several categories: they confuse visually similar letters, fail to discern phonemic contrasts, and incorrectly reconstruct the citation forms of inflected words. These problems are magnified when students are faced with input from a colloquial, regional dialect—as they often are in Arabic-language websites, podcasts, and social media. These written and spoken texts diverge in orthography, morphology, and lexical content from the Modern Standard Arabic commonly taught in foreign language classrooms. In addition, learners of Arabic without access to Arabic keyboards may make errors based on an unfamiliar or unintuitive Romanization scheme. Finally, any user may make simple typographical errors, such as hitting a key adjacent to the one intended.

Spelling correction, with a wide range of suggested alternate inputs, can provide a way around these problems by making it easier for the learner to find unfamiliar words in existing lexicons. We have created a spelling corrector for Arabic dictionary lookup which accepts input in the Standard Arabic Technical Transliteration System (SATTS) Romanization[1], verifies whether or not the query matches a citation form in a bilingual Iraqi Arabic to English dictionary, and suggests citation forms that the user may have meant to query. Unlike standard edit-distance approaches, this system incorporates specific knowledge about English-speaking learners of Arabic to detect and correct the most likely errors, rather than relying on strict orthographic similarity.

While our system takes into account errors based on mishearings, transcription mistakes, and typographical errors, we believe that the greatest impact of the system is its ability to guide users in correcting errors based on mishearings of spoken Arabic. Accordingly, we evaluate our system based on a corpus of hearing and pronunciation errors made by English-speaking learners of Arabic (Sethy *et al.*, 2005).

2. Modern Standard and Colloquial Arabic

The Arabic language presents several difficulties for the second language learner, not the least of which is the noted phenomenon of *diglossia*. In the paper 'Diglossia', Ferguson (1959) used Arabic (along with German and Greek) to illustrate how some linguistic communities may be split between what he called a *high prestige* (H) variety and and *low prestige* (L) variety. In the case of German, the contrast was made between Standard German and Swiss German. The Swiss live in a diglossic situation that requires them to be functionally bilingual. Swiss German is the native variety and is needed for communicating locally, but Standard German, which is learned in schools, is needed for communication with the wider German-speaking world and for access to the literary history of German.

The case of Arabic can be considered slightly more complex. Arabic also has a linguistic standard, the so-called Modern Standard Arabic (MSA) that is taught in schools, used in newspapers, in literature and in news broadcasts. MSA also forms the basis for most Arabic foreign language courses both in the West and in the Middle East. This variety, though standard, is not a native language to any Arabs, and it is not based off of any variety of Arabic currently spoken in the Arabic speech community. Instead, MSA is a modern construction, created during the rise of Arab nationalism in the late 19th and early 20th centuries and is based upon the Classical Arabic that was standardized in the eighth century (Haeri, 2003; Owens, 2006; Versteegh, 1997).

MSA sits in contrast with the local varieties of Arabic, generally termed *dialects*. These are the spoken varieties

---

[1] A table of SATTS equivalents can be found at http://en.wikipedia.org/wiki/SATTS

that exist across the Arabic-speaking world, and the dialects spoken in daily life hold a lower prestige with respect to MSA. Because MSA was constructed from a much older form, MSA can differ from the dialects (which often differ from each other) in terms of phonology, lexicon, syntax and morphology. The chart below shows how lexical items and morphological particles or clitics differ between MSA and some Arabic dialects.

	"he said"	"table"	Future Tense
MSA	/qāla/	/tāwila/	/sawfa/
Emirati	/gāl/	/mīz/	/bi-/
Iraqi	/gāl/	/mēz/	/rāḥ-/
Syrian	/ʾāl/	/tāwle/	/rāḥ-/
Egyptian	/ʾāl/	/tarabēza/	/ḥa-/
Yemeni	/qāl/	/māyidih/	/ʕa-/, /ša/
Moroccan	/qāl/	/tābla/	/γād/
Maltese	/ʾāl/	/meyda/	/ha-/, /ād/

Table 1: Examples of variation among various dialects of Arabic and a closely related language (Maltese).

This diglossic situation creates several challenges for second-language learners, the greatest of which is the simple fact that the language they learn in the classroom is not spoken natively by any Arab. When they find themselves conversing with Arabs, learners face great difficulty adapting their learning of the formal MSA variety to the linguistic differences of the dialects, which are spoken in everyday Arab life. In order to become truly proficient in Arabic, a learner must learn not only MSA but also a dialect in addition to having a familiarization with other dialects. Compounding this problem is the fact that many Arabs view the dialects as "corrupt" or "incorrect" Arabic and unworthy of teaching. Consequently, dialect materials for the learner can be sparse.

3. Spelling Correction

3.1 Uses of Spelling Correction

Error correction and normalization generally are useful for a variety of tasks, including optical character recognition, cross-language information retrieval, and the handling of out-of-vocabulary words for machine translation (cf. *e.g.* Habash, 2009). Most of these tasks, however, assume spelling correction for native speakers. We focus here on spelling correction for language learners and non-native language professionals such as translators. The primary purpose of our spelling corrector is to facilitate dictionary look-up of Arabic words in bilingual Arabic-to-English dictionaries.

This requires a different approach than other tasks. For example, unlike post-processing of optical character recognition, normalization of Arabic texts, or out-of-vocabulary handling, we are not so concerned in this work about variation in native spelling patterns (except as they impact lookup in dialect dictionaries of Arabic) as we are of mistakes by English learners. Furthermore, we do not have the luxury of surrounding context to help in disambiguating ambiguous forms or deciding between multiple corrections. On the other hand,

unlike fully automatic processes that must commit to a single correction, we have the luxury of displaying an arbitrary number of corrections to a user (limited only by constraints of the visual user interface and the user's patience in scrolling through alternatives). Hence, we can focus on returning a sensible ranking of alternatives, without a hard constraint that the right answer always be the highest-ranked one. That being said, having the right answer in the top ranked results is important for establishing the user's confidence in the tool.

3.2 Spelling Correction Techniques

A very simple way to create a spell corrector in the absence of any training data is to rank the suggested alternatives to a misspelled word in order of edit distance, or the number of string operations needed to change the query into the suggested alternative. Perhaps the most common formulation of edit distance is Levenshtein distance (Levenshtein, 1965; Wagner, 1974), which considers three string operations (insertion, deletion, or substitution) each with equal cost.

We suspect, however, that ranking alternative strings by Levenshtein distance will be less useful for the task of non-native Arabic dictionary lookup than the task of native English spell correction, for three reasons. First, non-native learners of a language are more likely than native writers to make multiple errors in a word (*cf. e.g.*, Okada, 2004; Mitton & Okada, 2007; Boyd, 2008). Secondly, since Arabic words are typically written without short vowels, the average length of a citation form tends to be shorter than in English, and the lexical space denser (in terms of edit distance). Lastly, Levenshtein distance, taken as a measurement between two broadly transcribed sound tokens, does not accurately reflect the difference in linguistic features between the two sounds, since it treats all symbols equally without regard for linguistic similarity between pairs of phones or letters (Nerbonne & Heeringa, 1997). Hence, Levenshtein distance may provide rankings too coarse-grained for use with non-native errors in Arabic orthography.

Other approaches to spell correction, either for native speakers (Church & Gale, 1991; Brill & Moore, 2000) or non-native speakers (Boyd, 2008), assume the existence of a training corpus consisting of pairs of misspelled words and their corresponding intended words. Such corpora are relatively easy to collect for resource-rich languages such as English or Japanese, particularly for native speakers. However, the few extant corpora of spelling errors made by adult learners of Arabic are too small for use in training.[2] Such resources are particularly scarce for the local spoken dialects.

Consequently, we employed a modular approach, developing separate modules for mistypings, phonetic confusions, and other dialectal confusions, each modeled through a weighted finite state transducer (FST).[3] The resulting FSTs are composed with a finite state machine

---

[2] For example, the error corpus described by Abuhakema *et al.* (2008) describes errors at all levels of linguistic production, including syntax and style, yet analyses less than 10,000 word tokens. Less than 50 spelling errors are reported—far too few to learn any reliable statistics from.

[3] We used the AT&T finite state toolkit, available from AT&T at http://www.research.att.com/~fsmtools/fsm.

accepting all strings corresponding to entries in an electronic dictionary. The composed FST calculates the best paths yielding unique, valid strings—*i.e.*, the dictionary entries most likely to have been the intended query given the misheard, mistransliterated, or mistyped input string. The corresponding dictionary entries are then displayed, ranked by likelihood.

To populate the confusion matrices for the phonetic confusion module, we collected data from a college-level Arabic instructor on her students' orthographic and listening comprehension errors. We also collected self-report data from six current or former students of Arabic on errors they believed they made in Arabic. Each participant used a 5-point rating scale to rank the most common insertions, deletions, and substitutions made by native-English students of Arabic. The averaged rankings were used to assign weights to the confusion matrix FSTs mentioned above.

A similar procedure was used to construct the module for transcription-based errors. A list of likely substitutions, insertions, and deletions was developed and ratings obtained from three native speakers of English who were proficient both in Arabic and in the SATTS transliteration system.

The module for handling keyboard-related (typographical) errors uses a constant weight for all pairs of horizontally contiguous neighbors on the QWERTY keyboard (*e.g.*, {C,V}). The module also includes, at a slightly smaller cost, the pair {:,;} which differ only in use of the *shift* key, and (at a greater cost) the pairs {l,:} and {",;} which involve both a horizontal slip and a misuse of the *shift* key. No insertions or deletions are used in this module. Preliminary tests by potential users of this tool indicated that correcting for keyboard-related errors was less useful than corrections from the other two modules; the costs of operations from this module were increased accordingly (making these substitutions less likely to be used).

The modules are combined into a single weighted FST by taking a union of operations found in the three modules. If two modules allow the same operation but assign different weights, the weight corresponding to the minimum cost is used. Transitive operations are assigned the sum of their costs. For example, since emphatic /d^{ʕ}/ (SATTS V) is phonetically confusable with unemphatic /d/ (SATTS D), and V can be mistyped as C (due to keyboard proximity), then the 'double' substitution of D→C would require two errors on the part of the user. The cost of the corresponding correction (C→D) is equal to the summed cost of the corrections (C→V) and (V→D).

4. Evaluation

To our knowledge, there are no spell-checkers or dictionary look-up aids described in the literature specifically designed for native-English adult learners of a colloquial Arabic dialect. Therefore, we evaluated our technique by comparing it with a baseline based on Levenshtein distance.

4.1 Creating an evaluation corpus

In the absence of an established corpus of dictionary look-up errors by native-English adult learners of Iraqi Arabic, we constructed a corpus of pseudo-errors for evaluation purposes. First, we examined a corpus of Iraqi and Lebanese Arabic spoken by native English speakers collected through an "elicited imitation" task, where each participant heard an Arabic sentence and was asked to repeat it from memory. The elicited imitations were recorded and transcribed phonetically. This corpus is further described in Sethy *et al.* (2005).[4] Because this corpus focused on common greetings and other conversational data, rather than a large number of word types suitable for dictionary look-up, we could not use the corpus directly. Instead, we constructed a noisy-channel model (Shannon, 1948) by extracting the probabilities of substitutions, insertions, and deletions the participants made in their elicited speech and applied the probabilities of these errors to a list of correctly spelled words from a dictionary. To construct the noise model, the transcript of the recorded speech is compared to the elicitation transcript for each trigram in the Sethy *et al.* (2005) corpus. For each of these trigrams, we recorded the probability that the participant inserted or deleted the center phone in the trigram, in the context of the left and the right phone. If the center phone was a substitution, the probability was recorded that the original phone would be replaced with the substituted phone, in the context of the left and the right phone.

We applied the noise model to a list of source words to form three conditions, controlling for the maximum number of errors per word (1 Error per word, 1-2 Errors per word, or 1-3 Errors per word – henceforth 1EPW, 1-2EPW, 1-3EPW).

We formed the source word list by choosing 400 citation forms at random from *A Dictionary of Iraqi Arabic* (Woodhead & Beene, 2003). To apply the noise model to the word list, an error point is chosen across the source word at random. At this error point, an operation (insertion, deletion or substitution) is chosen based on the probability of what happened to that phone in the Sethy *et al.* (2005) corpus, in the context of the left and the right phones surrounding that point. If the insertion of an error at the error point was not attested to in the noise model, the error point was abandoned and another one attempted. As the presence of at least one error in each word was desired, attempts were made repeatedly on each word up until the maximum number of errors for that condition was reached. Source words in which no acceptable error could be created were removed from the evaluation set. Multi-word entries were also discarded, leaving 398 forms for evaluation.

Since both the dictionary citation forms and the noisy model are given in terms of pronunciations (strings of phonemes), both the query strings and the original citation forms were converted into orthographic strings (as transliterated in SATTS). Queries containing no errors in the SATTS (*i.e.*, pairs where the query string matched the intended citation form exactly) were discarded, as these would provide no difficulties in lookup even without spell correction. This left 213 queries with orthographic errors for our evaluation in the 1EPW condition, 237 in the 1-2EPW condition, and 247 in the 1-3EPW condition.

4.2 Scoring

Results for each query were scored using reciprocal rank (Voorhees, 2000) given up to 35 suggested strings for

each query. When the spell corrector successfully suggested the intended citation form with rank *i*, it was awarded a score of $1/i$. Otherwise, a score of 0 was given for that query. If the intended citation form was tied in cost (edit distance for Levenshtein, FST path cost for our proposed system) with other forms, then we averaged over the reciprocal rank for each possible ordering of the tied forms. The mean reciprocal rank (MRR) over the set of 398 queries is reported below, along with the MRR over the set of queries with orthographic errors.

5. Results

5.1 Metrics

The reciprocal ranks for the proposed system and the Levenshtein baseline were compared using a paired, two-tailed t-test. The proposed system performed significantly better than the Levenshtein baseline MRR for the 1EPW condition (t = -5.1887, df = 397, $p < 0.0001$), but not for the 1-2EPW condition (t = -1.7344, df = 397, p = 0.0832) or the 1-3EPW condition (t = -1.4753, df = 397, p = 0.1409). Results for the MRR are shown in Table 2.

Further results can be seen in Tables 3-5, which show the number and percentage of items in which the target word appeared within the top *n* suggestions, for *n* = 1, 2, 3 and 5. Table 3 shows this for the 1EPW condition, Table 4 for the 1-2EPW condition, and Table 5 for the 1-3EPW condition.

System	1 Error per word	1-2 Errors per word	1-3 Errors per word
Proposed (all queries)	**0.828**	0.756	0.732
Levenshtein (all queries)	0.748	0.726	0.706
Proposed (errors only)	**0.678**	0.591	0.569
Levenshtein (errors only)	0.529	0.540	0.527

Table 2: Mean reciprocal rank, averaged over all possible ranks in case of ties, for a novel confusion matrix for English learners of Arabic and a baseline based on Levenshtein distance. Values are given for a set of 398 queries and for the subset of these queries where the query string differs from the target string. Bold type indicates results significantly better than baseline Levenshtein.

System	Top 1 (RR=1)	Top 2 (RR≥.5)	Top 3 (RR≥.3)	Top 5 (RR≥.2)
Proposed (all queries)	303 (76.1%)	332 (83.4%)	348 (87.4%)	351 (88.2%)
Levenshtein (all queries)	232 (58.3%)	304 (76.4%)	326 (81.9%)	347 (87.2%)
Proposed (errors only)	118 (55.4%)	147 (69.0%)	163 (76.5%)	166 (77.9%)
Levenshtein (errors only)	47 (22.1%)	119 (55.9%)	141 (66.2%)	162 (76.1%)

Table 3: Numbers and percentages of correct results returned in the top 1, 2, 3, or 5 ranked returns by a novel confusion matrix for English learners of Arabic and a baseline based on Levenshtein distance. The test corpus is constrained to allow one phonological error per word. RR stands for reciprocal rank, averaged over all possible ranks in case of ties.

System	Top 1 (RR=1)	Top 2 (RR≥.5)	Top 3 (RR≥.3)	Top 5 (RR≥.2)
Proposed (all queries)	275 (69.1%)	301 (75.6%)	317 (79.6%)	322 (80.9%)
Levenshtein (all queries)	222 (55.8%)	296 (74.4%)	319 (80.2%)	338 (84.9%)
Proposed (errors only)	114 (48.1%)	140 (59.1%)	156 (65.8%)	161 (67.9%)
Levenshtein (errors only)	61 (25.7%)	135 (57.0%)	158 (66.7%)	177 (74.7%)

Table 4: Numbers and percentages of correct results returned in the top 1, 2, 3, or 5 ranked returns by a novel confusion matrix for English learners of Arabic and a baseline based on Levenshtein distance. The test corpus is constrained to allow 1 or 2 phonological errors per word. RR stands for reciprocal rank, averaged over all possible ranks in case of ties.

System	Top 1 (RR=1)	Top 2 (RR≥.5)	Top 3 (RR≥.3)	Top 5 (RR≥.2)
Proposed (all queries)	263 (66.1%)	290 (72.9%)	308 (77.4%)	316 (79.4%)
Levenshtein (all queries)	213 (53.5%)	284 (71.4%)	310 (77.9%)	337 (84.7%)
Proposed (errors only)	112 (45.3%)	139 (56.3%)	157 (63.6%)	165 (66.8%)
Levenshtein (errors only)	62 (25.1%)	133 (53.8%)	159 (64.4%)	186 (75.3%)

Table 5: Numbers and percentages of correct results returned in the top 1, 2, 3, or 5 ranked returns by a novel confusion matrix for English learners of Arabic and a baseline based on Levenshtein distance. The test corpus is constrained to allow 1-3 phonological errors per word. RR stands for reciprocal rank, averaged over all possible ranks in case of ties.

5.2 Discussion and Error Analysis

In the 1EPW condition, the proposed system outperforms the Levenshtein baseline by all measures. It also outperforms the baseline in all conditions with respect to the percentage of items where the top-ranked return corresponds to the target word. However, in the 1-2EPW and 1-3EPW conditions, the proposed system performs under the baseline in returning the correct result within the top 5 returns.

This was a somewhat surprising result, but in retrospect it is a useful caution for limitations of the method of creating the confusion matrices and the evaluation proposed. While the confusions mentioned and ranked by teachers and students cover the most common cases, they do not cover every possible error. In particular, the confusion matrices currently used do not take into account errors dependent on surrounding phones or letters, such as incorrectly doubling a letter in the spelling (*e.g.*, to represent a geminate consonant, as in SNND سنند for Modern Standard Arabic SND سنّد /sannada/ '*support*').[5] However, the error generation model used in evaluation generates a large number of these letter doublings – 20, 42, and 52 in the 1EPW, 1-2EPW, and 1-3EPW conditions, respectively. Whether these are likely errors for students writing queries or is an artefact of the type and sparsity of data available for creating the error generation model is not known for certain, though the latter seems more likely. While enhancing the finite-state system to allow for context-dependent letter doublings is fairly straightforward, it is not clear that it is worthwhile. It is hoped that more direct evaluations of errors made in tasks more relevant to dictionary queries (such as student transcriptions of orally dictated words, or student use of the query system itself) will shed light on whether this expansion is needed.

6. Future Work

While our constructed corpus of generated orthographic error data derived from the speech errors observed in the Sethy *et al.* (2005) corpus provides some indication of the system's performance on sound-based confusions Arabic learners may make, we are interested in extending our evaluation using naturally occurring error data. Spelling and typing tests using auditory presentation may be utilized to capture students' Arabic confusions on carefully constructed word lists. Additionally, we plan to collect corpora of errors using on-line survey techniques. Research we may pursue incorporate the use of online typing games designed to elicit errors on chosen stimuli, or software designed to discover unknown errors in free-form web text. Additionally, we wish to compare the kinds of errors made by native Arabic speakers to non-native Arabic learners in these experiments.

We are developing a separate system for use in searching a dictionary for words encountered in written text. This will incorporate confusion matrices for visual similarity among Arabic script letters and letter proximity in the most common Arabic keyboard layouts.

7. Conclusion

Our system provides interactive spell correction in the context of dictionary lookup for a spoken Arabic dialect. To our knowledge, it is the first such resource described for a spoken, colloquial dialect of Arabic. It performs significantly better than the baseline approach, which assumes no language-specific knowledge. Yet the knowledge required to build this was easily obtained by a few interviews with teachers and students of the language. More generally, our evaluation gives an indication of what can be achieved for a language even with very few resources other than a primary source dictionary, and particularly without the use of training corpora such as those required by Church and Gale (1991), Brill and Moore (2000), and Boyd (2008).

8. Acknowledgement of Support

9. References

Abuhakema, Ghazi, Faraj, Reem, Feldman, Anna and Fitzpatrick, Eileen. (2008). Annotating an Arabic Learner Corpus for Error. In *Proceedings of the Sixth International Language Resources and Evaluation (LREC'08)*, Nicoletta Calzolari, Khalid Choukri, Bente Maegaard, Joseph Mariani, Jan Odjik, Stelios Piperidis, and Daniel Tapias (Eds.). Marrakech, Morocco: European Language Resources Association, pp. 1347-1350.

Boyd, Adriane. (2008). *Pronunciation Modeling in Spelling Correction for Writers of English as a Foreign Language.* M.S. Thesis, The Ohio State University.

Brill, Eric, and Moore, Robert C. (2000). An Improved Error Model for Noisy Channel Spelling Correction. In *Proceedings of the 38th Annual Meeting on Association for Computational Linguistics*, pp. 286-293.

Church, Kenneth W. and Gale, William A. (1991). Probability scoring for spelling correction. *Statistics and Computing*, 1, pp.93-103.

Ferguson, Charles A. (1959). Diglossia. *Word*, 5, pp. 325-340.

Habash, Nizar Y. (2009). REMOOV: A Tool for Online Handling of Out-of-Vocabulary Words in Machine Translation. In *Proceedings of the 2nd International Conference on Arabic Language Resources and Tools (MEDAR)*, pp. 217-220.

Haeri, Niloofar. (2003). *Sacred Language, Ordinary People*. New York: Palgrave Macmillan.

---

[5] While an early implementation of the system included a back-off to Levenshtein distance (with a high weight), this impacted performance speed, and so the back-off was removed to keep the system's response time as fast as possible.

Levenshtein, Vladimir. (1965). Binary codes capable of correcting deletions, insertions and reversals. *Doklady Akademii Nauk SSSR*, 163(4), pp. 845–848.

Mitton, Roger and Okada, Takeshi (2007). The adaptation of an English spellchecker for Japanese writers. Presented at *Symposium on Second Language Writing*, 15-17 Sept. 2007, Nagoya, Japan.

Nerbonne, John and Heeringa, Wilbert. (1997). Measuring Dialect Distance Phonetically. In *Workshop on Computational Phonology. Special Interest Group of the Association for Computational Linguistics*. Madrid: Association for Computational Linguistics, pp. 11-18.

Okada, Takeshi (2004). A corpus analysis of spelling errors made by Japanese EFL writers. *Yamagata English Studies*, 9, pp. 17–36.

Owens, Jonathan. (2006). *A Linguistic History of Arabic*. Oxford: Oxford University Press.

Sethy, Abhinav, Mote, Nicolaus, Narayanan, Shrikanth, and Johnson, W. Lewis. (2005). Modeling and Automating Detection of Errors in Arabic Language Learner Speech. In *INTERSPEECH-2005*, pp. 177-180.

Shannon, Claude. (1948). A Mathematical Theory of Communication. In *The Bell System Technical Journal*, 27, pp. 379-423, 623-656.

Versteegh, Kees. (1997). *The Arabic Language*. New York: Columbia University Press.

Voorhees, E.M. (2000). The TREC-8 question answering track report. In *The Eighth Text REtrieval Conference (TREC-8)*, Spec Pub 500-246. Washington DC: NIST, pp. 77-82.

Wagner, R.A. & Fischer, M.J. (1974). The string-to-string correction problem. *Journal of the Association for Computing Machinery*, 21(1), pp. 168-73.

Woodhead, D.R. & Beene, Wayne (Eds.). (2003). *A Dictionary of Iraqi Arabic: Arabic – English*. Washington, D.C.: Georgetown UP.