A Multi-Agent System for Detecting and Correcting “Hidden” Spelling Errors in Arabic Texts

Chiraz Ben Othmane Zribi, Fériel Ben Fraj and Mohamed Ben Ahmed

RIADI Laboratory, ENSI, La ManoubaUniversity, La Manouba, Tunisia

Abstract. : In this paper, we address the problem of detecting and correcting hidden spelling errors in Arabic texts. Hidden spelling errors are morphologically valid words and therefore they cannot be detected or corrected by conventional spell checking programs. In the work presented here, we investigate this kind of errors as they relate to the Arabic language. We start by proposing a classification of these errors in two main categories: syntactic and semantic, then we present our multi-agent system for hidden spelling errors detection and correction. The multi-agent architecture is justified by the need for collaboration, parallelism and competition, in addition to the need for information exchange between the different analysis phases. Finally, we describe the testing framework used to evaluate the system implemented.

1 Introduction

Hidden errors are spelling errors that occur as valid words. The presence of such a word within an incorrect syntactic or semantic context makes the whole sentence unintelligible. For instance:
Example: تطلع الشّمس علينا من UالشّوقU (the sun shines from desire)
In this example, the writer intended to write "الشّرق"(east) not "الشّوق"(desire) but a typographical error yielded a sentence that does not make sense. Statistics given by Mitton (cited in Verberne, 2002) show that hidden errors count for 40% of all spelling errors. This high number demonstrates the need for studying this kind of errors.
Several researchers have taken an interest in this problem, Golding studied this kind of errors for the English language and proposed multiple correction methods such as the Bayesian method (Golding, 1995), the trigram-based method (Golding and Schabes, 1996) and the Winnow method (Golding and Dan Roth, 1999). Chinese was also studied by Xiaolong and Jianhua (2001). Swedish was the subject of a similar study by Bigert and Knutsson (2002).
Even though Arabic has characteristics that increase the probability of such errors occurring, there is not any research done in the subject of hidden errors for Arabic. In this paper, we describe a multi-agent system that allows the detection and correction of hidden errors, occurring in Arabic texts. Due to the complexity of the problem, we made some assumptions to restrict the scope of our investigation: first, we did not take into account the vowel markings in words and assumed that there is only one hidden error per sentence. Second, we assumed that the error resulted from one ele-

Ben Othmane Zribi C., Ben Fraj F. and Ben Ahmed M. (2005).
A Multi-Agent System for Detecting and Correcting “Hidden” Spelling Errors in Arabic Texts.
In *Proceedings of the 2nd International Workshop on Natural Language Understanding and Cognitive Science*, pages 149-154
DOI: 10.5220/0002556601490154

mentary typographical error such as character insertion, deletion, substitution or transposition.

The remainder of this paper is organized as follows: First, we present the Arabic language characteristics that contribute to increasing the risk of hidden errors. We then present the classification we adopted for these errors. Next, we show the general architecture of our multi-agent system and present a detailed description of the work of each agent in its environment. Finally, we present the method we used to evaluate the efficiency of our system and the results obtained.

2 Difficulties of Arabic Language

In Arabic, the problem of "hidden" spelling errors is much more complicated than in other languages. Indeed, Arabic has numerous writing constraints that can lead to ambiguities. One such constraint is the agglutination of affixes to the simple form in order to obtain composite forms. In addition, Ben Othmane Zribi (1998) notices, "*Arabic words are lexically very close".* According to this author, the average number of forms that are lexically close[1] is **3** for English and **3.5** for French, whereas it is **26.5** for Arabic words without vowel marks. Arabic words are thus much closer to one another than French and English words. Consequently, in Arabic the probability that two words are lexically close is 10 times larger than in English and 14 times larger than in French.

This proximity of Arabic words has a double consequence: First, on error detection, words that are recognized as correct can in fact hide an error. This is the case when, for example, instead of typing the word "كتب" (has written), one types the word "كسب" (has won). Second, on error correction, the number of suggested corrections for an erroneous form can be excessively high. One could estimate that an average of **27** forms can be proposed for correcting each error. These figures illustrate the difficulty of automatic error correction in a language such as Arabic.

3 Classification of Hidden Spelling Errors

Detecting hidden errors cannot be done by a morphological analysis since these errors generate morphologically valid forms that are however erroneous on the syntactic or the semantic level. Consequently, a sentence containing a syntactic error is lexically correct but the structuring of its words is incorrect. On the other hand, a sentence containing a semantic error is not clear because of the presence of a hidden error within its context.

- **Syntactic Errors :** There are different types of grammatical anomalies. We have classified them as follows: errors of agreement, errors related to verb transitivity and errors of grammatical structure.

[1] Two words are lexically close if they differ from one another by one single editing error (substitution, addition, deletion and inversion).

- **Semantic Errors :** Semantic errors can also be divided into two sub-classes: semantic incompatibility and semantic omissions.

4 Suggested Approach

The complexity of the problem, as well as the hierarchy of the hidden errors point out the need for interaction between the various phases of analysis. Indeed, the detection and the correction of the syntactic errors may require the contribution of semantic knowledge. Similarly, the treatment of hidden semantic errors requires syntactic backtracking for a better detection and correction.

An added constraint for Natural Language Processing (NLP) systems is that they must respond quickly to the user. Therefore, one of our objectives was to reduce the response time by the use of parallel processing for various parts of the system.

Consequently, we chose a multi-agent architecture where different agents work in collaboration, competition, coordination and parallelism, in order to achieve the whole goal of the system. Each agent contributes to the final solution, and they all share a common environment where they can pass information and cooperate. Moreover, a multi-agent architecture offers flexibility since it easily accepts the addition of new agents.

5 General Architecture of the Detection-Correction System

For more efficiency, an error checking system must have various linguistic information about the texts to be analysed. For that purpose, a morpho-syntactic analysis of the input text is performed by our system.

5.1 Syntactic Group of Agents

This group of agents is made up of four agents: the Agreement agent, the Transitivity agent, the Grammatical checker agent and the Supervisor agent. The Supervisor receives the text to be checked and sends it, sentence by sentence, to its colleagues in the same group.

- **The Agreement agent:** checks the validity of agreement constraints using a set of 840 agreement rules;
- **The Transitivity agent:** tries to detect anomalies between verbs and their object complements by checking the transitivity rules;
- **The Grammatical checker agent:** checks the order of the parts of speech of the agglutinative forms (HyperCGs) in the sentence by considering ternary sequences of HyperCGs. It uses for this, a third dimension matrix that shows all licit ternary sequences of hyperCGs.

The Supervisor controls the work of these three agents. If one agent detects an anomaly, it informs the others agents to stop their work and lets the supervisor know about the error. This starts the process of correction.

5.2 Semantic Group of Agents

The Semantic group of agents consists of four agents: the Supervisor sends the text, sentence by sentence, to the other agents of the same group. The other three agents are: the Co-occurrence agent, the Repetition agent and the Coordinator agent.

- **The Co-occurrence Agent:** This agent checks that each word in the sentence has semantic affinities with its context. It proceeds in two ways: First, the agent searches for collocations between the target word and the surrounding words. Collocations, if they are found, should consolidate each word in its context. In addition to collocations, the Co-occurrence agent searches for ordinary co-occurrences between each target word and its context.
- **The Repetition Agent:** This agent checks whether the lemma of the textual form to check repeats itself in the text. It is based on the assumption that “Words (or more precisely lemmas of words) of a text tend to repeat themselves in this text”. Indeed, according to research carried out by Ben Othmane Zribi and Ben Ahmed (2003) on an Arabic textual corpus, it seems that a textual form can appear 5.6 times on average, whereas a lemma can appear 6.3 times on average in the same text.
- **The Coordinator Agent:** This agent combines the results obtained by the two agents: Co-occurrence and Repetition in the following formula.The final result of semantic checking is sent to the Supervisor in order to start the process of correction.

5.3 The Correction Agent

Finally, the Correction agent starts to correct the errors detected by the syntactic and semantic checkers. It proceeds by generating all the forms close to the error. These forms are obtained through one editing error. They are then all added to a list, which contains the candidates for the correction. As previously cited, the number of these candidates can be excessively high and one could estimate that an average of 27 forms will be suggested for the correction of each error. In extreme cases, this number can reach 185 forms (Ben Othmane Zribi, 1998).

To reduce the number of candidates, the Correction agent substitutes the erroneous word with each suggested correction and forms a set of candidate sentences. These sentences are processed once more by the detection part of the system and sentences containing syntactic or semantic anomalies are eliminated from the list. The remaining sentences are then sorted

6 Testing and results

At this stage of the project, we have implemented the syntactic group of agents and integrated the Correction agent previously developed by Ben Othmane Zribi (1998).

In order to assess the system realized, we needed a textual corpus containing hidden errors. However, for lack of a corpus containing this kind of errors in their natural form, we had to manually create our own corpus. We generated among the forms that

exist in the corpus a list of artificial hidden errors based on the restrictive assumptions of our study.
This corpus, which constitutes the data to our system, contains approximately 720 not vowel marked textual forms. It was segmented in 100 sentences, into which we introduced 100 hidden errors of the syntactic type. These errors are of various types: 43 errors of agreement, 50 syntactic structure errors and the remainder errors relate to verb transitivity.

6.1 Evaluation of the Detection Component

The system for the detection of hidden errors gave very satisfactory results with a rate of 80% of accuracy (number of good detections / total number of detections). However, the system had some shortcomings, which caused a silence rate (number of not detected errors / total numbers of errors) of 23% mainly due to:

- The width of the range of checking: Some of the detection agents gave better results with short sentences than with long ones. In spite of the phase of segmentation into sentences, the number of words per sentence remains large.
- The competition between agents: When a detecting agent finds an error, it stops the others without knowing if this error is a real one.

6.2 Evaluation of the Correction Component

This evaluation was performed in two phases: Phase 1: when the correction component returned a list of candidate correction. Phase 2: after the reduction of the list using the detection system. The results are illustrated in the table below:

Table 1. Evaluation of the Corrector agent

	Coverage	**Accuracy**	**Ambiguity**	**Proposal**	**Rank**
Initially	100%	100%	100%	82.5	8.7
After reducing	93.3%	86.6%	86	18.4%	2.8

7 Conclusion and Future Work

The part of the system that has been implemented gave satisfactory results. The choices that were initially made enabled us to reach our goals. However, we estimate that the results obtained can still be improved upon by updating the linguistic rules used and by taking into account the semantic information. Therefore, our next step is to implement the semantic group of agents.

References

1. Ben Othmane Zribi C. *De la synthèse lexicographique à la détection et à la correction des graphies fautives arabes*. Thèse de doctorat, Université de Paris XI, Orsay, 1998.
2. Ben Othmane Zribi C. and Ben Ahmed M. *Le contexte au service de la correction des graphies fautives arabes*. TALN'03, Nantes, 11-13 Juin 2003.
3. Bigert J. and Knutsson O. *Robust Error Detection: A Hybrid Approach Combining Unsupervised Error Detection and Linguistic Knowledge*. In Proceedings of Robust Methods in Analysis of Natural Language Data (ROMAND'02), Frascati, Italie, 2002.
4. Golding A. R. *A bayesian hybrid method for context- sensitive spelling correction*. In *Proceedings* of the Third Workshop on Very Large Corpora, Cambridge, Massachusetts, USA, pages 39-53, 1995.
5. Golding A. R. et Dan Roth. *Applying winnow to context-sensitive spelling correction*. In Lorenza Saitta (ed.) Machine Learning: Proceedings of the 13th International Conference. Bari, Italie, pp. 182-190, 1996.
6. Golding A. R. et Dan Roth. *A winnow-based approach to context-sensitive spelling correction*. Machine Learning, 34(1-3), 107-130, 1999.
7. Verberne S. *Context sensitive spell checking based on word trigram probavilities*. Mémoire de Mastère, Université de Nijmegen, 2002.
8. Xiaolong W., Jianhua L. *Combine trigram and automatic weight distribution in Chinese spelling error correction*. Journal of computer Science and Technology, Volume 17 Issue 6, Province, China, 2001.