Rapid Development of Translation Tools

Jan W. Amtrup
Computing Research Laboratory
New Mexico State University
email: jamtrup@crl.nmsu.edu

Karine Megerdoomian
Computing Research Laboratory
New Mexico State University
email: karine@crl.nmsu.edu

Rémi Zajac
Computing Research Laboratory
New Mexico State University
email: zajac@crl.nmsu.edu

Abstract

The Computing Research Laboratory is currently developing technologies that allow rapid deployment of automatic translation capabilities. These technologies are designed to handle low-density languages for which resources, be that human informants or data in electronically readable form, are scarce. All tools are built in an incremental fashion, such that some simple tools (a bilingual dictionary or a glosser) can be delivered early in the development to support initial analysis tasks. More complex applications can be fielded in successive functional versions. The technology we demonstrate has first been applied to Persian-English machine translation within the Shiraz project and is currently extended to cover languages such as Arabic, Japanese, Korean and others.

1 Introduction

One major goal that is pursued at the Computing Research Laboratory (CRL) is to create a machine translation environment that allows the rapid development and deployment of translation systems for low-density languages. The main problem lies with the usually low amount of machine-readable data for the development of knowledge sources (machine readable dictionaries, large corpora) and the difficulty to acquire such knowledge manually. In order to support the creation of machine translation systems for such languages, we developed techniques that allow the incremental installation of successively more complex translation systems and tools. The framework which is used at CRL consists of the following tools, presented here in increasing order of processing complexity (and, consequently, in increasing order of acquisition complexity):

- A bilingual dictionary lookup tool for human translators.
- A dictionary editor which allows modifying existing entries in a dictionary as well as adding new words to a dictionary.
- A preprocessor which combines spell-checking of a source text and augmentation of the underlying dictionary with previously unknown words.
- A glosser that performs dictionary lookup for a complete text and allows viewing of translations in context.
- A translation system that produces automatic translations of source texts on various levels of quality.

In this paper, we will demonstrate several of these components as applied to Persian-English machine translation in the Shiraz project [1]. Knowledge sources and tools for other languages are under development, including Arabic, Japanese, Korean, Russian, Serbo-Croatian and Spanish.

2 The Shiraz System

The aim of the Shiraz project is to develop a machine translation system that translates Persian text into English. We target news material and various miscellaneous texts. The source texts are usually taken from on-line sources (e.g., web pages), although plain text can be handled as well. The dictionary, the primary knowledge source in the system, contains approximately 50,000 entries. It has been built by Per-

[1] http://crl.nmsu.edu/Research/Projects/shiraz

sian lexicographers and consists of single words, compounds and phrasal expressions. A dictionary of common proper names (Onomasticon) was added to extend the coverage for typical newspaper texts.

Dictionary entries consist of information about the orthography, morpho-syntactic category and syntactic properties.

Other knowledge sources (grammars) were developed in a corpus-oriented manner. The basis was a 10MB corpus of Persian on-line texts. We created English translations for the sentences in the corpus and tagged both sides with morpho-syntactic information, rendering a bilingual aligned corpus that we use to test the system.

2.1 Persian-English Dictionary

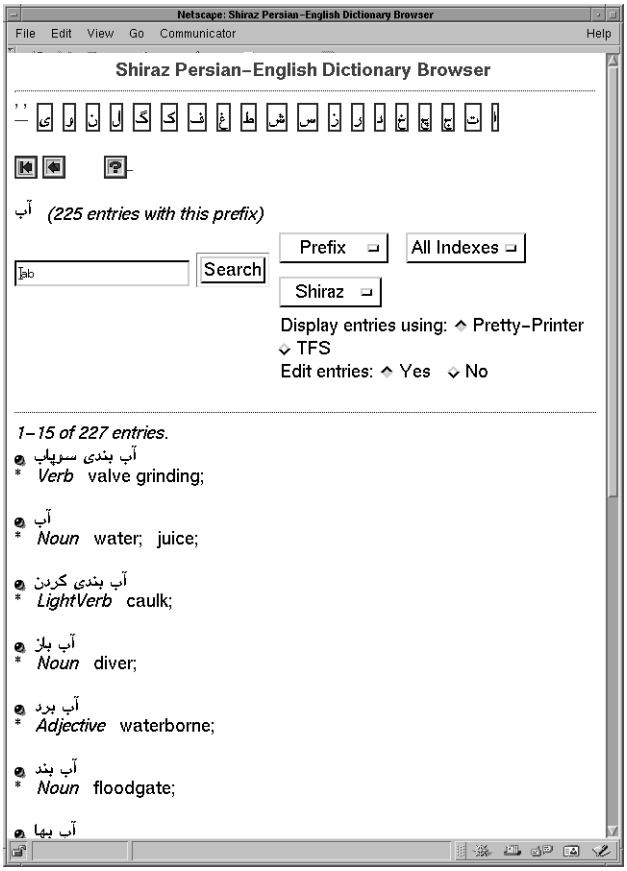

Figure 1: The Shiraz Dictionary browser

The dictionary forms the primary knowledge source for the Shiraz system. It contains approximately 50,000 entries, single words as well as compounds, phrasals and proper names. The dictionary is stored in an SQL database which facilitates several mechanisms of access and modification (Zajac, 1998). This source dictionary is compiled into an internal format to be used for translation applications.

There are two main steps involved in accessing the dictionary: Browsing and editing. Both tasks are performed through a web-based interface. First, the user searches for entries in the dictionary. This can be done by entering a word (or parts of it) in one of several ASCII-based transliteration schemata for Persian (see Figure 1). Alternatively, a letter tree of Persian characters (a level of which is displayed in the top part of Figure 1) may be used to search for words. The browser then shows a list of Persian words with their respective syntactic categories and English translations.

Selecting an entry (by clicking on it) starts the second phase of dictionary work, namely editing an entry (see Figure 2). Here, the various fields that constitute an entry can be modified. A head word may have variants and several different parts of speech attached to it. Consequently, the content of the editing page is parameterized which allows for different schemata for different dictionaries (enabling the use for more than one language).

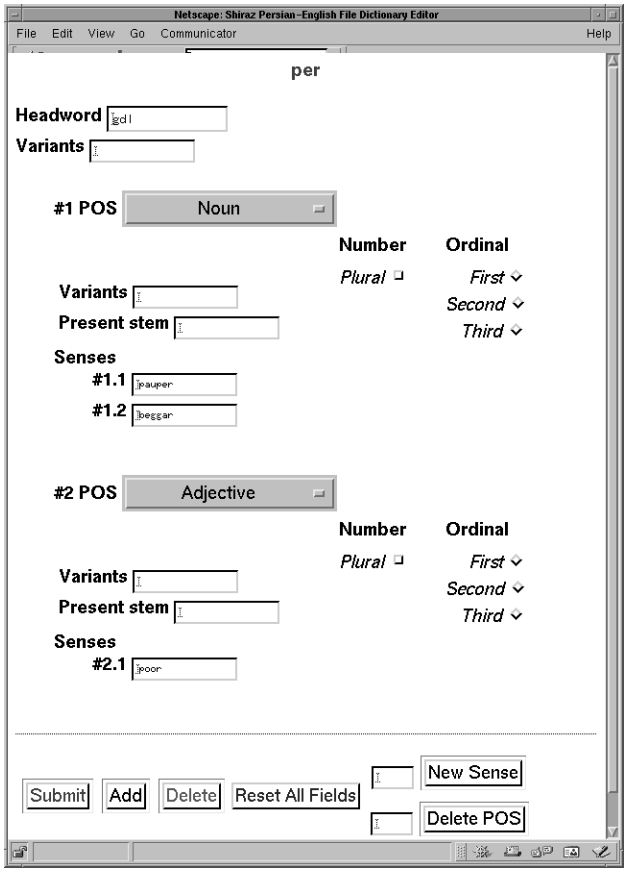

Figure 2: The Shiraz Dictionary editor

Usually, editing of entries is performed directly on the contents of the database. However, for certain special tasks (such as adding new entries based on corpus work or the results of spell-checking, see below) we de-

signed a file-based editing mode that iterates over the contents of a file. After completing modifications on all entries in a file, it can be uploaded into the main dictionary database.

2.2 Spell-Checker

The purpose of the spell-checker employed for Shiraz is twofold: It can be used to correct genuine spelling errors in a Persian text, and it prepares a file of so-far unknown words which are scheduled for addition in the dictionary (see Figure 3).

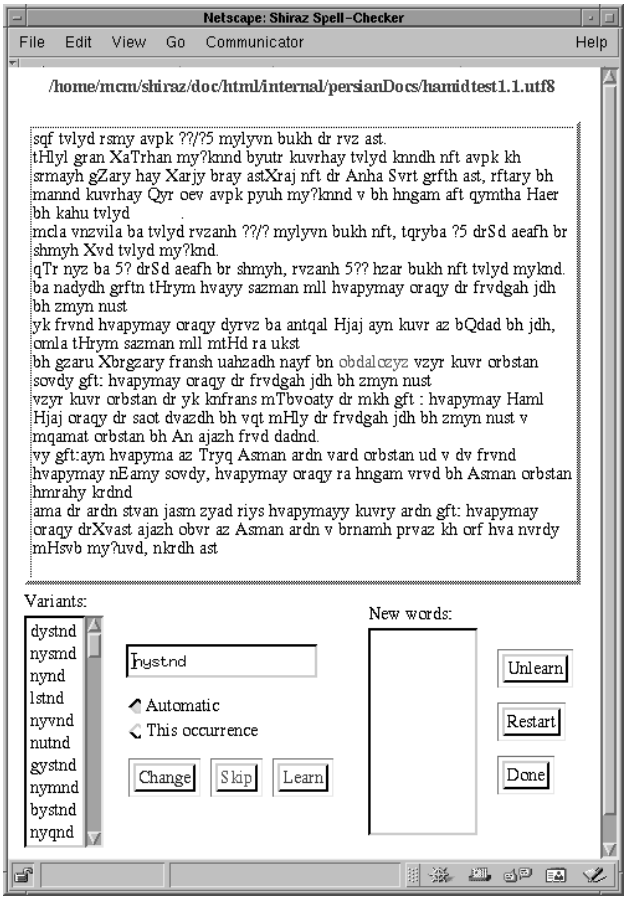

Figure 3: The Spell-Checker interface

In order to produce a list of correct candidates for a possibly misspelled word, we perform a shallow morphological analysis (stemming). The citation form thus derived is checked against the dictionary. Certain common problems, like the attachment of two words without an intervening space, are taken into account. If the word could not be found, a list of candidates is created from words that are similar to the input (we allow errors with an edit distance of one). Then, the user can choose to select one of the candidates presented; if the word was not misspelled, but appears to be a new word, it may be marked for addition into the dictionary.

2.3 Glosser

The glosser performs a dictionary lookup for all words in a text and displays the original words together with their possible translations, showing the context of the whole text (see Figure 4). This can either be used to get a rough impression of the contents of a text, or to augment dictionary entries by including adequate translations.

Figure 4: The Glosser interface

The input text is divided into individual tokens first. Each word is then subject to morphological analysis to compute the citation form of inflected words. After dictionary lookup the original Persian words are displayed with all possible English translations. Compounds and Persian light verb constructions are treated using separate components. To further analyze the way a Persian word was processed, a second window can be opened that contains the morphological information obtained. If an unknown word was detected, we simply take the transliterated form as translation.

2.4 Translator

The translator is used to automatically create translations on various levels of quality. The lowest of these levels is given by a word-for-word translation similar to the output of the glosser. In contrast to the glosser, however, morphological transfer and generation are performed. The next higher level of quality is achieved by taking into account syntactic knowledge encoded in a grammar. Together with an English generation module that is able to reorder translations of sub-phrases if needed, this gives an overall better translation quality (cf. Vanni and Zajac, 1997).

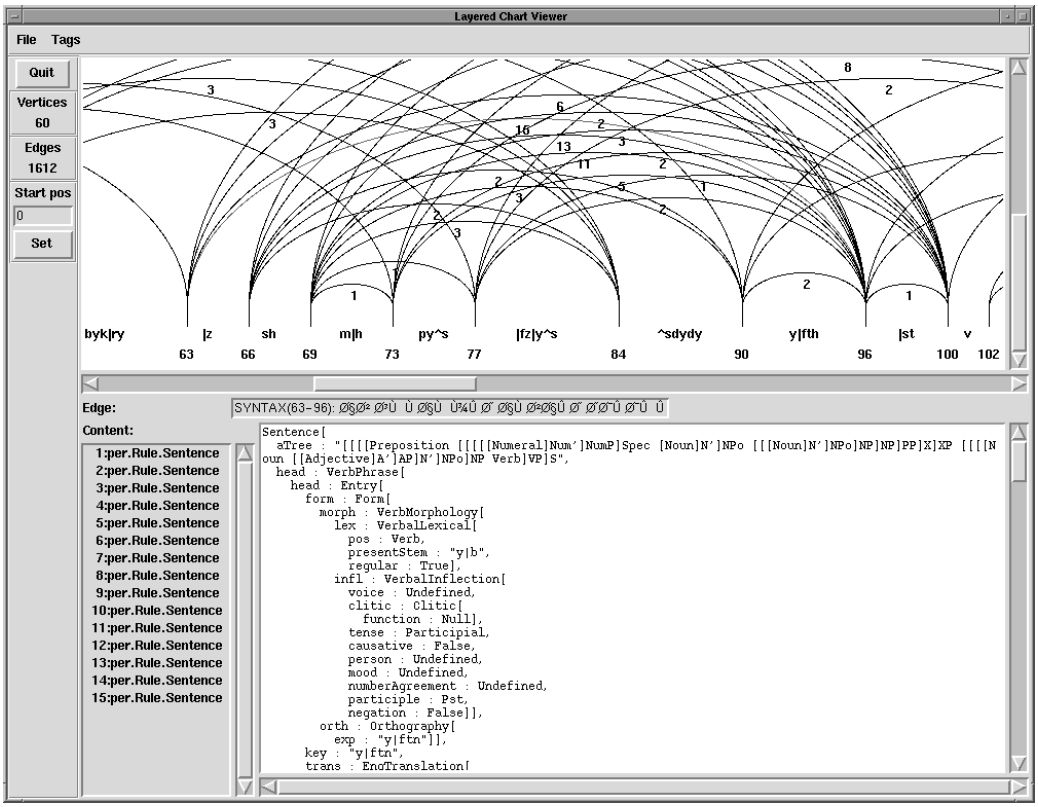

Figure 5: Viewing a complex analysis

Furthermore, we are able to use an English language model to choose the best among several possible translations of a sentence. A graphical interface (see Figure 5) can be used to trace the way a particular sentence was analyzed by the system.

The architecture of the system is extremely modular and can be parameterized to a great extent. The system is transfer-based and works on an augmented chart structure, a layered chart (Amtrup, 1997). Using these properties, the translation system can be viewed as modular and robust. It produces translations using the highest level of description if possible. If such a deep analysis fails, fragments of the input are translated on a word-by-word basis.

During development, the translator is usually called directly from the command line to perform some task. However, we implemented a web-based interface which can be used to translate headlines and parts of the body of newspaper text. The interface is shown in Figure 6.

2.5 Implementation

The underlying formalism for all knowledge sources and the representation of linguistic objects during runtime is typed complex feature structures. The architecture of the translation system is based on a simplified version of layered charts and a strong component concept.

The dictionary browser and editor are servlets implemented in Java. The spelling-checker is also implemented using Java. The glosser and translation system are implemented using C++. Throughout all components and tools we use Unicode to encode text strings internally. Externally, a wide range of encodings can be used. Using Unicode to represent character data allows a wide variety of different languages and scripts to be processed.

3 Application to other Languages

Besides Persian, the method and tools presented in this paper are applied to several other languages to a varying degree. These languages include: Arabic, Japanese, Korean, Russian, Serbo-Croatian and Spanish.

4 Conclusion

The development of automatic translation capability for low-level density languages in a short amount of time is difficult. In this paper, we used the example of Persian-English machine translation to demonstrate what tools might be helpful in the process of deploying an MT system.

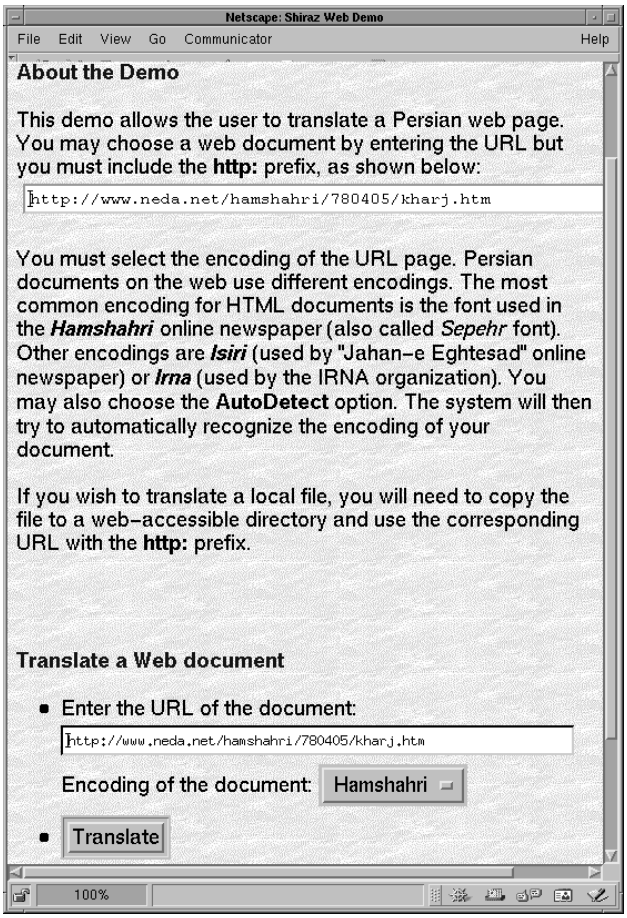

Figure 6: The Web translator interface

Acknowledgments

This research has been funded in part by DoD, Maryland Procurement Office, MDA904-96-C-1040.

References

Amtrup J. (1997). "Layered Charts for Speech Translation". In Proceedings of the Seventh International Conference on Theoretical and Methodological Issues in Machine Translation, TMI '97, Santa Fe, NM, July 1997.

Vanni M. and Zajac R. (1997). "Glossary-Based MT Engines in a Multilingual Analyst's Workstation Architecture". Machine Translation 12, Special Issue on New Tools for Human Translators. pp 131–157.

Zajac R. (1998). "The Habanera Lexical Database Management System". In First International Conference Language Resources and Evaluation. Granada, Spain, pp 28–30, May 1998.