Developing a Persian Part of Speech Tagger

Karine Megerdoomian
University of California, San Diego
karinem@ling.ucsd.edu

1. Introduction

Assigning grammatical categories to words in a text is an important component of a natural language processing (NLP) system. Corpora tagged with Part of speech (POS) information are often used as a prerequisite for more complex NLP applications such as information extraction, syntactic parsing, machine translation or semantic field annotation. They are also used to help train statistical models.

Prior to tagging, a natural language processing system generally requires modules for segmenting tokens in the text and providing a morphological analysis. The actual annotation scheme used, however, is often motivated by the system application. This paper outlines some of the main challenges that arise in the development of a Persian POS tagger – such as encoding issues, long-distance dependencies in morphology, recognition of complex tokens, word and phrasal boundaries, and analysis of multiword expressions – and proposes approaches to resolving these issues.

2. Symbolic and statistical approaches

There exist two main approaches in the field of computational linguistics: Statistical approaches which employ probabilistic methods for learning from annotated corpora and symbolic methods that take advantage of a knowledge-based system of rules.

Knowledge-based taggers, also known as rule-based taggers, analyze corpus data using a grammatical model. Hence, the information about morphological and grammatical structures is encoded in the program (possibly using a meta language) rather than being "learned" from a training corpus. Rule-based taggers can often correctly analyze complex and long structures, but they are generally unable to provide tags for constructions that have not been recognized.

Statistical taggers use probabilistic algorithms to analyze a document but they need to be trained on a pre-tagged corpus (often tagged manually). Based on this training corpus, statistical taggers build a probability matrix that stores the probability of an individual word belonging to a certain gammatical class or part of speeech, as well as the word's distributional probability. The advantage of probabilistic taggers is that when the tagger encounters an unknown word, it can use the distributional information gathered from the n-grams to determine (or guess) the grammatical class of the unknown word given its nearby context. Statistical taggers can reach high accuracies; however, the results often saturate, at which point the performance of the system can no longer be improved.

Many modern taggers combine both statistical and rule-based methodologies; these systems are known as *hybrid taggers*. Since rule-based taggers can accurately annotate large

grammatical constructions, they are used to analyze and mark up a given corpus. Statistical taggers are then applied to disambiguate the results or to guess the tags for the unknown words. The issues described in this paper are based on such a hybrid model.

3. Tagset design

A *tagset* is the set of all the annotation tags to be used that allows the system to explicitly assign a part of speech or grammatical class to the analyzed tokens. The main objective of the tagset is to define an annotation set that can provide the relevant linguistic information to the user about the sysntactic or semantic properties of the word. However, the design of a tagset depends heavily on the goals of the research and the final application of the NLP system, and therefore there is no single standard tagset for Persian. The most basic tagset will include part of speech information for major grammatical classes. For information retrieval applications, it is often necessary to mark the boundaries of constitutents or to tag noun phrases. Some applications require even more markup such as annotating the semantic information that can be later used in word sense disambiguation. In this section, we will discuss some of the main criteria to be considered when designing a Persian tagset.

Tags are generally short while being able to convey the relevant linguistic information to the user. Examples are AJC for comparative adjective, VBP or V-pres for the present form of a verb, and NPL or Nn-Pl for a plural noun. Depending on the application, some tagsets also include features that will be useful at further stages of processing or that will be needed for predicting the behavior of nearby words. For instance, the superlative adjective in Persian precedes the noun while the base form and the comparative adjective appear following the head noun of a noun phrase. A distinction between these adjective types at the tagger level can therefore facilitate the analysis at the phrasal parsing stage.

In general, since the boundaries of noun phrases are highly ambiguous in Persian, any morphological information pertaining to the presence or absence of the noun phrase boundary would be helpful at later stages of processing. For instance, if a noun appears with the object marker را in Persian text, simply tagging it as Nn would not allow the system to distinguish it from a non-case-marked nominal. Such information would thus be advantageous not only in determining the boundaries of a noun phrase but also in defining its syntactic function.

If the system is to be used in extracting name entities and determining their relations, it may be useful to provide more detailed information on the types of proper names occurring in the text, thus creating distinct tags for PPers (Person), PCnt (Country), or POrg (Organization). Semantic features such as animacy information could also be included in a tag since they can help disambiguate parses based on the verbal agreement.

Providing a more detailed tagset could ascertain that no useful information is lost at later stages of processing. However, one should keep in mind that the more fine-grained the distinctions made in a tagset, the larger its size would become making it harder to train a statistical model and thus giving rise to more errors. For instance, in most tagger applications there is no need to distinguish the simple past, perfect, imperfect, or compound imperfect tenses and instead they are all marked with the tag V-Past. An average tagset consists of about 40 to 50 tags.

4. Encoding issues

The first component of a NLP system often consists of a tokenization module which segments the document into tokens. The extended Arabic script used in writing Persian texts naturally brings about certain ambiguities since the short vowels are usually not written, yet the system should be flexible enough to be able to detect diacritics when they appear in the text. Furthermore, the inconsistent usage of the whitespace in Persian documents gives rise to problems in detecting word, phrase and sentence boundaries.

Although a number of more recent corpus sites have been converging on the usage of utf8 for most online documents, Persian texts do not follow a single standard encoding. Generally, NLP systems convert the input text into a common encoding, such as Unicode standard which provides a unique number for each character, for easy manipulation in the system. Often, a romanization is used for internal purposes to facilitate the linguistic and lexicographic work.

Encoding issues often occur in processing of Persian text. For instance, besides the range of Unicode characters designed for Persian, online texts sometimes employ Arabic or Ascii characters as well. Hence, the letters *kâf* and *ye* can be expressed by either the Persian encoding (\u06a9 for ک and \u064a for ی) or by the Arabic unicode (\u0643 for ک and \u06cc or \u0649 for ی). Any Persian system should be able to process all of these possible input versions. In addition, a number of control characters, such as the final form marker or directionals, may appear in Persian input text. Since the final form character marker (or zero-width non-joiner, ZWNJ), expressed as \u200c in unicode, indicates boundaries of words or compound parts, it is helpful for the segmenter to treat it as a whitespace and use it to delimit token boundaries. This then allows the system to analyze both forms of a compound, either with intervening space or with an intervening ZWNJ character, in a uniform fashion.

5. Word boundaries

One of the biggest issues in processing of Persian text is the optional nature of the whitespace, which causes distinct words to appear as a single token (e.g., رفتندمردم). In order to analyze these adjoined words, certain systems use a post-segmentation script to separate unrecognized tokens at possible boundary points and to look up the resulting words in the lexicon.

Optionality of the whitespace also raises issues in the analysis of detached morphemes such as فلسطینی ها or می دادند. The inflectional morphemes such as هاي, ها or تـريـن, can appear either as bound to the host, as free affixes separated by a final form character (or ZWNJ), or separated with an intervening space. Any morphological analyzer for Persian should be able to recognize all of these forms and to provide the correct grammatical tags. In Riazati (1997), the detached affixes are treated in the syntactic component while the attached ones are recognized in the morphological analyzer. In the Shiraz system developed at CRL (in New Mexico, USA), a post-tokenization component is used to join the detached morpheme to the stem prior to morphological analysis. Megerdoomian (2004), however, treats these elements as multiword tokens and processes them along with the attached forms in the analyzer component; this method does not require a preprocessing module and there is no need to delay the analysis of the detached morphemes to the syntactic level.

6. Complex Tokens

Complex Tokens refer to multi-element forms, which consist of affixes that represent a separate lexical category or part of speech than the one they attach to. These attached word-like morphemes such as the preposition بـه, the determiner ایـن, the postposition را, or the relativizer کـه, may appear attached to the adjacent word and need to be recognized in the morphological analyzer. Similarly, a number of pronominal or verbal clitic elements may occur on various parts of speech categories. Examples are: بشیوه – اینکار – بهترست .

In certain cases, two distinct syntactic categories may appear without an intervening space even though they are not attached. For instance, the preposition در ends in the character *re* which does not distinguish between a final form and an attached form. Sometimes در appears without a space separating it from the following word (e.g., دردفـتـر)and the tokenizer may not be able to segment the two words; in these instances, the complex tokens need to be recognized in the morphological analyzer by treating the preposition as an affix.

7. Phonetics and phonological rules

In Persian, the form of the morphological affixes varies based on the ending character of the stem. Hence, if an animate noun ends in a consonant, it receives the plural morpheme *–ân* as in زنـان. If the animate noun ends in a vowel, the glide *ye* is inserted between the stem and the suffix (گـدایـان), and if the word ends in a silent *he* character, the last character of the word is replaced by *gâf* (بـــیگـانگـان). These phonological rules apply across categories and are not limited to the plural formation (e.g., glide insertion before the indefinite morpheme in دانشـــجویـي). In order to recognize these constructions, it is usually more efficient to implement the phonological rules that apply in these cases instead of listing all the possible morphemes as independent affixes.

A problem arises in Persian with characters that may be either vowels or consonants depending on the context and cannot be analyzed correctly simply based on the orthography (e.g., دانشجو vs. گـاو). Thus, any morphological system would need to distinguish the words based on their pronunciation since the phonetic representation of Persian nouns and adjectives plays a crucial role in the type of phonological rule that should apply to morpheme boundaries.

Past morphological analysis systems have often either not captured the pronunciation-orthography discrepancy in Persian thus not constraining the analyses allowed, or they have preclassified the form of the morpheme that can appear on each token. Using phonological rules, along with an attention to word-final pronunciation, can apply across the board at all morpheme boundaries and allows the system to capture important linguistic generalizations.

8. Long distance dependencies

Especially in the Persian verbal paradigm, certain morphological constructions can only be analyzed by looking at dependencies between non-adjacent morphemes. For instance, the two verbal forms مـیگریخـتـند and مـیگریخـته اسـت cannot be distinguished until the person inflection and auxiliary forms have been reached. At the same time, the absence of the مي prefix in these cases produces very different verbal tenses as in گـریخـتـند and گـریخـته اسـت.

Hence, the only way to determine the actual tense of the verb is to take into account the cooccurrence of the prefix and the person inflection.

Accounting for the long-distance dependency between the prefix and the personal inflection in Persian in a linear system, such as a two-level morphology module, leads to very complex paths and continuation class structures in the grammar. Also, using filters to capture long-distance dependencies can sometimes largely increase the size of the transducer. Incorporating some sort of unification process, however, allows the system to accept or reject a non-adjacent affix based on the unification possibilities in the grammar. Hence in the examples discussed, the morphological analyzer would be able to determine that a verbal construction is to be analyzed as Compound Imperfect (مـــي گـريخـته اســت) if the prefix on the verb is مي and includes a present auxiliary . On the other hand, the unification of a null prefix with the present auxiliary will give rise to a Perfect tense (گـريخـته اســت).

9. Multiword expressions

One of the biggest problems in Persian morphological analysis is the presence of a large number of multiword expressions. These include certain compound tenses such as future or modal forms, light verb constructions, or compound nouns. These elements range from lexical units such as بــنابـرايـن to phrasal verbs that can be separated from each other in the sentence as in: اظـهار تأسـف کـردند.

Analysis of the unit-like elements (e.g., بــنابـرايـن) can be accomplished pretty easily by listing them in the lexicon. Certain compound forms can be analyzed in the morphological module by undergoing recursive rules. Thus for verbs, once a participle is formed in the morphological analyzer, it may combine with an auxiliary to create a compound tense; this auxiliary will then follow the same conjugation rules as the original or main verb. For example, the compound imperfect tenses are formed by joining the past participle and the present auxiliary (مـــيگريخـته اســت), but the participle can also combine with the past auxiliary, which could itself be conjugated as in the double compound past (فروخته بوده اند).

The phrasal and productive verbal elements, however, are best analyzed at an intermediary parsing level rather than within the morphological analyzer. The subparts of these elements can be separated from each other in the input text by intervening morphemes (e.g., object pronoun clitics), by modifiers, or by phrasal elements (e.g., noun phrase or preposition phrase). If these elements are simply listed in the lexicon as single units, they would not be recognized by the system. In particular, in more linear systems such as the two-level finite-state transducers these phrasal verbs will not be analyzed unless both their present stem and past stem forms are listed, thus increasing the size of the lexicon. These elements are therefore best analyzed in a parsing module prior to tagging or syntactic analysis. Furthermore, incorporation of unification processes in the analysis of light verb constructions could allow the system to recognize separated phrasal elements as a single verbal unit, and to transfer the relevant information from each subpart (i.e., the personal inflection and tense from the light verb and the semantic information from the preverbal element).

Another instance of phrasal elements are the nominal compounds in which the head of the noun is the first token. As an example, consider the nominals خمر سرخ or مـاشـيـن لـبـاسـشـوئـي, where the plural affix appears on the first subpart as shown: خمرهاي سـرخ and

ماشـينهاي لـباسـشـوئـي. These compounds are also best treated as a phrasal element rather than as a single unit in the lexicon.

10. Phrasal boundaries

The Persian noun phrase (NP) is highly ambiguous and thus causes immense difficulties for automatic parsing of written text. Numerous factors contribute to the ambiguity of the Persian NP structure: short vowels are not written which produces lexical ambiguities, there are very few overt morphemes in the language to mark boundaries of noun phrases, there are often no particles in written text linking the constituents of a noun phrase since the *ezafe* morpheme is usually an unwritten short vowel. Furthermore, since the basic word order in Persian is Subject-Object-Verb, the lack of overt morphology for marking boundaries makes it very difficult to determine where the subject ends and the object begins. All of these factors, coupled with very long sentences, a relatively free word order and the optionality of the subject, make the Persian noun phrase extremely ambiguous for an analysis of written text. Depending on the application and goals of the system, it may be useful to incorporate within the tagging module as much of the NP boundary information as can be obtained from morphological analysis.

Two of the most consistent lexical items that delimit the boundary of a noun phrase in written text are the Pronoun and the Proper Name. In general, the possessor element demarcates the end of the noun phrase in Persian as in وزيـــر خـارجـه آيـنده آمـريـكـا . The suffix را always marks the boundary of an object noun phrase or a topicalized phrase. In addition, a number of affixes such as the pronominal clitic (تان/شان), the indefinite article(ی), and the relativizing affix (ی), all indicate that the end of a noun phrase has been reached. In all of these cases, it may be possible to create a tag that marks the boundary of the noun phrase (e.g., +NPB).

On the other hand, the presence of an *ezafe* morpheme indicates that the boundary of the noun phrase has not been reached and the nominal or adjectival element needs to be joined with the constituent that follows it. Hence, if the ezafe is present at the end of a word ending in a vowel, a tag could be added to the analyzed token to indicate the lack of boundary for the noun phrase (e.g., +NONPB). This is shown in the analysis of the noun phrase below:

	انفجارهای اخیر عراق
Morphological analysis:	انفجار+Noun+Pl+NoNPB اخیر+Adj+Sg عراق+Prop+Cntry
Tagger analysis:	انفجار [Nn-Pl-NoNPB] اخیر [Adj] عراق [Prop-Cnt]

Note that the lack of an ezafe morpheme can also be used in detecting the boundary of a noun phrase. Hence, if a noun or adjective ends in a vowel and is not followed by the ezafe affix as in انفجارها , it can be marked with the +NPB tag.

11. Conclusion

This paper presents an overview of the main challenges encountered in the development of a POS tagger for Persian. The paper describes problems arising from encoding issues, detached inflectional morphemes, as well as attached word-like elements forming complex tokens, the

discrepancy between orthography and phonetics in application of phonological rules, the interdependency between non-adjacent morphemes in a word, and the recognition of phrasal boundaries. In addition, the paper introduces certain criteria to be considered in designing an annotation set for POS tagging. By contrasting various approaches in the field, possible methods are proposed for resolving these computational and linguistic issues.

References

Assi, S. Mostafa and M. Haji Abdolhosseini. 2000. Grammatical Tagging of a Persian Corpus. *International Journal of Corpus Linguistics* 5(1):69-82.

Bateni, Mohammad-Reza. 1995. توصیف ساختمان دستوری زبان فارسی. Amir Kabir publishers, Tehran, Iran.

Beesley, Kenneth R. and Karttunen, Lauri. 2003. *Finite-State Morphology: Xerox Tools and Techniques*. CSLI publications, Palo Alto.

Dehdari, Jonathan and Deryle Lonsdale. 2004. An Integrated System for Processing Persian. Unpublished manuscript (submitted to Coling 2004).

Mahootian, Shahrzad. 1997. *Persian*. Routledge.

Megerdoomian, Karine. 2000. Unification-Based Persian Morphology. In *Proceedings of CICLing 2000*. Alexander Gelbukh, ed. Centro de Investigación en Computación-IPN, Mexico.

Megerdoomian, Karine. 2004. Finite-State Morphological Analysis of Persian. Unpublished manuscript (submitted to Coling 2004).

Riazati, Dariush. 1997. Computational Analysis of Persian Morphology. MSc thesis, Department of Computer Science, RMIT.