On the Importance of Ezafe Construction in Persian Parsing

Alireza Nourian, Mohammad Sadegh Rasooli, Mohsen Imany and Heshaam Faili

Department of Computer Engineering, Iran University of Science and Technology, Tehran, Iran
{nourian,m_imany}@comp.iust.ac.ir

Department of Computer Science, Columbia University, New York, NY, USA
rasooli@cs.columbia.edu

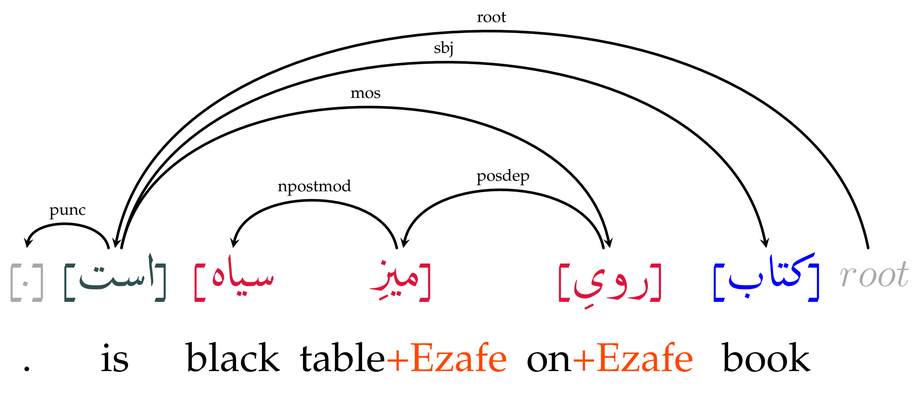
School of Electrical and Computer Engineering, University of Tehran, Tehran, Iran
hfaili@ut.ac.ir

Introduction

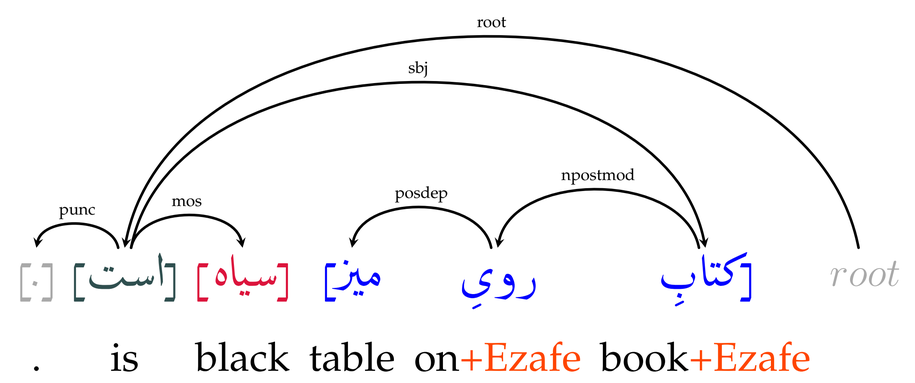
Ezafe is an unstressed vowel -e that occurs at the end of some words (-ye in some specific occasions) that links together elements belonging to a single constituent:

montazer$_{\text{e}}$ Ab
waiting$_{\text{Ezafe}}$ water
waiting for water

This construction is very useful for disambiguating syntactic structures, but Ezafe rarely appears in the written text. This relies on the fact that Persian is written in Perso-Arabic script and vowels are mostly not written. Following figures show two different readings for the same sentence with different Ezafe constructions:



First reading: The book is on the black table.



Second reading: The book on the table is black.

Data Preparation

- We attach the Ezafe indicator to the POS tags and train a sequence tagger on the new tagset.
- We have also manually Ezafe tagged all words in the Persian dependency treebank with 99.6% annotator agreement.
- We define some rules to convert a dependency tree to a shallow phrase structure. Implementation of the these rules is available in the Hazm toolkit:

https://github.com/sobhe/hazm
Python library for digesting Persian text.

Acknowledgements

We thank Computer Research Center of Islamic Sciences (CRCIS) for supporting us on corpus annotation.

Results

- **9%** relative error reduction in **shallow parsing**:

Tagset	Tag Acc.	Precision	Recall	F-Measure
POS	98.71%	89.44%	88.02%	88.72%
POSe	97.33%	**90.42%**	**89.13%**	**89.77%**

Chunking results on the Persian dependency treebank test data with automatic POS tags.

- **4.6%** relative error reduction in **dependency parsing**:

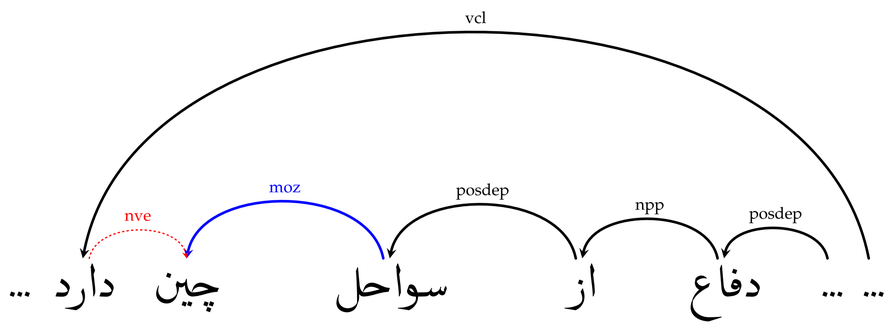
Tagset	Tag Acc.	MaltParser		YaraParser		TurboParser	
		LAS	UAS	LAS	UAS	LAS	UAS
POS	98.71%	85.34%	88.80%	85.90%	89.43%	87.28%	90.59%
POSe	97.33%	**85.74%**	**89.24%**	**86.35%**	**89.86%**	**87.73%**	**91.02%**

Dependency Parsing results on the test data with different automatic tagsets.

Analysis

Effect on the common POS tags Dependency attachment accuracy is improved by 6.5% for adjectives and 6.2% for nouns and for some tags such as determiners the Ezafe construction does not help.

Manual data investigation The main gain is on those sentences where the presence/absence of Ezafe construction is crucial for making correct decisions by the parser:



Effect on the training data size We can leverage Ezafe construction and use only 70% of the training data while reaching the accuracy of the original part of speech tagset trained on the whole data: