

# EVALUATION OF STATISTICAL PART OF SPEECH TAGGING OF PERSIAN TEXT

Samira Tasharofi<sup>a</sup>, Fahimeh Raja<sup>a</sup>, Farhad Oroumchian<sup>a,b</sup>, and Masoud Rahgozar<sup>a</sup>

<sup>a</sup>Electrical and Computer Engineering Department, University of Tehran

{s.tasharofi,f.raja}@ece.ut.ac.ir, Oroumchian@acm.org, and Rahgozar@ut.ac.ir

<sup>b</sup>University of Wollongong in Dubai

FarhadO@uow.edu.au

## ABSTRACT

*Part of Speech (POS) tagging is an essential part of text processing applications. A POS tagger assigns a tag to each word of its input text specifying its grammatical properties. One of the popular POS taggers is TnT tagger which was shown to have high accuracy in English and some other languages. It is always interesting to see how a method in one language performs on another language because it would give us insight into the difference and similarities of the languages. In case of statistical methods such as TnT, this will have an added practical advantages also. This paper presents creation of a POS tagged corpus and evaluation of TnT tagger on Persian text. The results of experiments on Persian text show that TnT provides overall tagging accuracy of 96.64%, specifically, 97.01% on known words and 77.77% on unknown words.*

## 1. INTRODUCTION

Part-of-speech tagging selects the most likely sequence of syntactic categories for the words in a sentence. It determines grammatical characteristics of the words, such as part of speech, grammatical number, gender, person, etc. This task is not trivial since many words are ambiguous: for example, English word "fly" can be a noun (e.g. a fly is a small insect) or a verb (e.g. the birds will fly north in summer). Such phenomenon is known practically in most of the languages.

In recent years, there has been a growing interest in data-driven machine-learning disambiguation methods, which can be used in many situations such as tagging. Among the most promising disambiguation methods are those based on learning decision list [7] which is an ordered list of conjunctive rules. The decision list induction problem is to identify from a training set of examples the decision list that will most accurately classify future examples. Although there are many models and implementations available for the task of tagging, most of them are designed for and tested on English texts; less work has

been done on tagging and tagger evaluation for languages like Persian that have quite different properties and script. There are many different models for tagging which differ on their internal model or the amount of training or intervention of information they need. In this paper we present the evaluation of a statistical part of speech tagger known as TnT tagger[1] on Persian texts. TnT tagger is proposed by Thorsten Brants and in literature its efficiency is reported to be as one of the best and fastest on diverse languages such as German [1], English [1, 2], Slovene [4], and Spanish [5].

The main problem in training statistical taggers is creating an annotated or tagged corpus. We used BijanKhan's tagged corpus [3] for training and testing. However this corpus is built for other purposes and has very fine grained tags which are not suitable for POS tagging experiments. Therefore, we had to make some changes on the corpus as described below in order to be able to use it in our work.

In the rest of this paper, first the TnT tagger is introduced in Section 2. Section 3 describes the test corpus. Section 4 presents the evaluation process involving the corpus file format conversion, obtaining training and test sets from the corpus and tagging of the files. Section 5 depicts the analysis of the results and finally, Section 6 presents conclusion and future works.

## 2. THE TNT TAGGER

Brants's TnT (Trigrams'n'Tags) tagger [1] is a statistical part of speech tagger, trainable on different languages and virtually any tag set. The component for parameter generation is trained on a tagged corpus. The system incorporates several methods of smoothing and of handling unknown words. TnT is not optimized for a particular language; instead, it is optimized for training on a large variety of corpora. The tagger is an implementation of the Viterbi algorithm for second orders Markov models. The main paradigm used for smoothing is linear interpolation; the respective weights are determined by deleted interpolation.

Unknown words are handled by a suffix trie and successive abstraction. Average part-of-speech tagging accuracy reported for various languages is between 96% and 97%, which is at least as good as the state of the art

results found in the literature. The accuracy for known tokens is significantly higher than for unknown tokens. For German newspaper data, when the words seen before (the words in its lexicon) the results are 11% points better than for the words not seen before (97.7% vs. 86.6%). It should be mentioned that the accuracy for known tokens is high even with very small amounts of training data [1].

### 3. THE CORPUS

The corpus which was used in this work is a part of the BijanKhan's tagged corpus [3], which is maintained at the Linguistics laboratory of the University of Tehran.

The corpus is gathered from daily news and common texts. It was tagged with a rich set of tags consisting of 550 different tags. The tags are organized in a tree structure. This vast amount of tags are used to achieve a fine grained part-of-speech tagging, i.e. a tagging that discriminates the subcategories in a general category. Considering this large size of tags makes any machine learning process impracticable. So we decided to reduce the number of tags as described in the following.

#### 3.1. Selecting the Suitable Tags

Most of the tools for part-of-speech tagging do not work with a large set of tags. In order to make the tagging process more feasible, we decided to reduce the size of our tag set. The process of tag selection included a statistical analysis on the corpus [6] (e.g. the number of times that each tag appeared in the corpus) and only the tags that appear enough number of times were kept.

BijanKhan's corpus uses a good representation for tags; each tag in the tag set follows a hierarchical structure. Each tag name includes the names of its parent tags. Each name starts with the name of the most general tag and follows by names of the subcategories until it reaches to the name of the leaf tag. For example, the tag "N\_PL\_LOC" contains three levels; "N" at the beginning stands for noun; the second part, "PL" shows the plurality of the tag, and the last part, "LOC", illustrates that the tag is about locations. For another example, the tag "N\_PL\_DAY" demonstrates a noun that is plural and describes a date.

The tag set reduction was done according to the following four steps:

1. In the first step, we reduced the depth of the hierarchy as follows. We considered all the tags with three or more levels in hierarchy and changed them to two-level ones. Hence, both of the above examples will reduce to a two-level tag, namely "N\_PL". The new tag shows that they are plural nouns. After rewriting all the tags in the corpus in this manner, the corpus contained only 81 different tags.
2. Among the 81 remaining tags in the corpus, there were a number of tags that described numerical entities. After close examination of these tags, it was realized that many of them are not correct and are product of the mistakes in the tagging process. In order to prevent

decreasing the accuracy of our part-of-speech tagger, all these tags were renamed to "DEFAULT" tag. So, the number of tags in the tag set reduced to 72 tags in this step.

3. In the third step, some of the two-level tags were also reduced to one-level tags. Those were tags that appeared in the corpus rarely but were unnecessarily too specific. Examples of these are conjunctions, morphemes, prepositions, pronouns, prepositional phrases, noun phrases, conditional prepositions, objective adjectives, adverbs that describe locations, repetitions and wishes, quantifiers and mathematical signatures. By this modification, the number of tags reduced to 42.
4. In this step we reduced the tags that appeared rarely in the corpus. These are noun (N) and short infinitive verbs (V\_SNFL). We consider the semantic relationship between these tags and their corresponding words. For example, since the words with tag "N" are single words, we replace "N" with "N\_SING". Also because the meaning of the "V\_SNFL" tag is not similar to any other tags in the corpus, we simply removed it from the corpus. After this stage, 40 tags remained in our final tag set.

#### 3.2. Statistical Analysis of the Corpus

Table 1 shows the tags and their corresponding frequencies in the corpus.

Studying the table carefully reveals that the tag "N\_SING" has the most number of appearances in the corpus. On the other hand, the "NN" tag has the minimum occurrences (two times) in the corpus.

## 4. EXPERIMENTAL PROCESS

In order to do our experiments, some steps must be followed. These steps include preparing the corpus files for TnT (Format conversion), providing test and training sets from the corpus and finally tagging the files. In this section, we will describe these steps.

#### 4.1. Format Conversion

The untagged input files for TnT tagger tool should have only one column of tokens of the text. If the line contains a space, all characters after the first space character are ignored.

The format of tagged files required for TnT training set has only two columns with the same order as our corpus; it is similar to that of the untagged files but it extends the format by a second column: the first column is the token, and the second column is the tag. Everything after the second column is ignored.

The token in training and test files occupies all characters from the beginning of the line up to the first space and must not contain spaces. As some tokens in Persian have some spaces between their characters such as "بر می گردم" or "BAR MI GARDAM", a conversion program is

implemented to remove these spaces from the tokens. It is clear that removing these spaces does not affect the accuracy of TnT. The tokens can be encoded using all characters with the ASCII codes from 31 to 255.

**Table 1.** The tags distribution

Tag Name	Frequency in Corpus	Probability
ADJ	22	8.46826E-06
ADJ_CMPR	7443	0.002864966
ADJ_INO	27196	0.010468306
ADJ_ORD	6592	0.002537398
ADJ_SIM	231151	0.088974829
ADJ_SUP	7343	0.002826473
ADV	1515	0.000583155
ADV_EXM	3191	0.001228282
ADV_I	2094	0.000806024
ADV_NEGG	1668	0.000642048
ADV_NI	21900	0.008429766
ADV_TIME	8427	0.003243728
AR	3493	0.001344528
CON	210292	0.080945766
DEFAULT	80	3.07937E-05
DELM	256595	0.098768754
DET	45898	0.017667095
IF	3122	0.001201723
INT	113	4.34961E-05
MORP	3027	0.001165155
MQUA	361	0.000138956
MS	261	0.000100464
N_PL	160419	0.061748611
N_SING	967546	0.372428585
NN	2	7.69842E-07
NP	52	2.00159E-05
OH	283	0.000108933
OHH	20	7.69842E-06
P	319858	0.123119999
PP	880	0.00033873
PRO	61859	0.023810816
PS	333	0.000128179
QUA	15418	0.005934709
SPEC	3809	0.001466163
V_AUX	15870	0.006108693
V_IMP	1157	0.000445353
V_PA	80594	0.031022307
V_PRE	42495	0.01635721
V_PRS	51738	0.019915033
V_SUB	33820	0.013018022
Max	967546	0.372428585
Min	2	7.69842E-07
Sum	2597937	1

## 4.2. Providing Test and Training Sets

In the majority of the part of speech tagging approaches, the sample is often subdivided into "training" and "test" sets. The training set is generally used for learning, i.e. fitting the parameters of the tagger. The test set is for assessing the performance of the tagger.

In our experiments, we repeated the experiments five times and each time we used a random sample of files, 123 files from 814 files, as the test set and used the rest of the files for the training. Table 2 shows the number of tokens and their percentages in the training and test sets respectively.

**Table 2.** Number of tokens in training and test sets

Run	Training Tokens/Percent	Test Tokens/Percent	Total
1	2196166 / 84.52	402050 / 15.47	2598216
2	2235558 / 86.04	362658 / 13.96	2598216
3	2192411 / 84.38	405805 / 15.61	2598216
4	2178963 / 83.86	419253 / 16.13	2598216
5	2186811 / 84.16	411405 / 15.83	2598216
Avg.	2197982 / 84.59	400234.2 / 15.40	

## 4.3. The Training Process

Before tagging, the parameters of the model must be learnt from a tagged corpus. The parameter generation requires a tagged training corpus in the format described in section 4.1. The program generates lexical and contextual frequencies from the training corpus and stores them in two files. The tagging process requires these two files containing the model parameters for lexical and contextual frequencies and an untagged (raw) input file in the format described in section 4.1.

## 5. EXPERIMENTAL RESULTS

For the evaluation purpose, the tagged file was compared with the original manually tagged test file and the differences were recorded.

Considering the tagging accuracy as the percentage of correctly assigned tags, we have evaluated the performance of the TnT tagger from two different aspects: (1) the overall accuracy (taking into account all tokens in the test corpus) and (2) the accuracy for known and unknown words, respectively. The latter is interesting since after training the tagger, it could be used on other text than the training text. It is interesting to know how it would cope with words that did not appear in its training.

Tables 3, 4 and 5 depict the results of the experiments. For each run, Table 3 shows the percentage of seen words (words that exist in training set), number of tokens in the test set, number of tokens correctly tagged and the percentage of accuracy for that run. Similarly, Table 4 shows the same for words that are new for the tagger. Table 5 shows the overall result for each run and its average. In general:

1. The overall part-of-speech tagging accuracy is around 96.64%.
2. The accuracy for known tokens is significantly higher than that for unknown tokens (97.01% vs. 77.77%). It shows 19.24% points accuracy difference between the words seen before and those not seen before.

**Table 3.** Known tokens results

Run	Percent	Tokens	Correct	Accuracy
1	98.07	394290	382211	96.94%
2	98.16	345913	345913	97.18%
3	98.04	397849	343894	96.96%
4	98.02	410970	398487	96.96%
5	98.07	403460	391475	97.03%
Avg.	98.072	390496.4	372396	97.01%

**Table 4.** Unknown tokens results

Run	Percent	Tokens	Correct	Accuracy
1	1.93	7760	5829	75.12%
2	1.84	6689	5357	80.09%
3	1.96	7956	6153	77.34%
4	1.98	8283	6435	77.69%
5	1.93	7945	6246	78.62%
Avg.	1.928	7726.6	6004	77.77%

**Table 5.** Overall results

Run	Tokens	Correct	Accuracy
1	402050	388040	96.52%
2	362658	351270	96.86%
3	405805	391890	96.57%
4	419253	404922	96.58%
5	411405	397721	96.67%
Avg.	400234.2	386768.6	96.64%

In Table 6 the overall part-of-speech tagging accuracy is compared to the performance of TnT tagger for English, German and Spanish as reported in the literature. As depicted in the table the overall accuracy for Persian is less than but close to that of German and English [1], and higher than Spanish.

**Table 6.** Overall results

Language	Tokens Unknown	Known accuracy	Unknown accuracy
English	2.9%	97.0%	85.5%
Germany	11.9%	97.7%	89.0%
Spanish	14.4%	96.5%	79.8%
Persian	1.894%	97.002%	77.454%

## 6. CONCLUSION AND FUTURE WORKS

An evaluation of a statistical part of speech tagger known as TnT on Persian has been presented. In this work, a test collection for POS tagging was produced by reducing the tag set of a manually tagged corpus. The experiments were repeated several times in which the training and test sets were selected randomly from 85% and 15% of the

collection respectively. The results show that the overall accuracy of the tagger is about 96.59% and the accuracy for known words is much higher than unknown words (about 24%).

In comparison with other languages, the accuracy of TnT for Persian, is less than but near to its accuracy for English and Germany and higher than its accuracy for Spanish. It should be noted that the results of using TnT on different languages show that the decisions made in TnT yield good results on a large variety of corpora.

This shows that with the statistical part of speech tagging without prior linguistic knowledge, we can generate a reasonable POS tagger for Persian language. Even though, Persian has a different script than English or other Latin script based languages.

In future developments of this work, it is intended to compare the TnT tagger with other tagging models on the Persian texts. Moreover, investigating the effect of other approaches of selecting training and test sets with varying sizes on the performance of the tagger is envisioned

## ACKNOWLEDGEMENTS

Many thanks go to Thorsten Brants for his attention to our e-mails and giving us his very efficient and user friendly tool. We would like to thank Dr. Faili for his helps in gathering and preparing the tagged corpus and Dr. BijanKhan for his valuable work in tagging the Persian texts and providing us with his tagged corpus. We also thank Iran Telecommunication and Research Center (ITRC) for their grants on this project.

## REFERENCES

- [1] T. Brants, "TnT – a Statistical Part-of-Speech Tagger," in *Proc. sixth conference on applied natural language processing (ANLP-2000)*, Seattle, WA, 2000.
- [2] R. Mihalcea, "Performance Analysis of a Part of Speech Tagging Task," in *Proc. Computational Linguistics and Intelligent Text Processing*, Gelbukh A. Editor, Centro de Investigaci3n en Computaci3n IPN, M3xico, 2003.
- [3] M. BijanKhan, "The Role of the Corpus in Writing a Grammar: An Introduction to a Software," *Iranian Journal of Linguistics*, vol. 19, no. 2, fall and winter 2004.
- [4] S. Dzeroski, T. Erjavec, and J. Zavrel, "Morphosyntactic Tagging of Slovene: Evaluating Taggers and Tagsets," in *Proc. LREC 2000*, Athens, 2000.
- [5] R.M. Carrasco, and A. Gelbukh, "Evaluation of TnT Tagger for Spanish," in *Proc. Fourth Mexican International Conference on Computer Science (ENC'03)*, 2003.
- [6] F. Oroumchian, S. Tasharofi, H. Amiri, H. Hojjat, and F. Raja, "Creating a Feasible Corpus for Persian POS Tagging," *Technical Report*, no. TR3/06, University of Wollongong (Dubai Campus), May 2006.
- [7] R.L. Rivest, "Learning Decision Lists," *Machine Learning Journal*, vol. 2, no. 3, pp. 229-246, 1987.