Journal of Mobile Multimedia, Vol. 13, No.3&4 (2017) 222-232
© Rinton Press

COUPLING AN ANNOTATED CORPUS AND A LEXICON FOR AMAZIGH POS TAGGING

SAMIR AMRI and LAHBIB ZENKOUAR
LEC, EMI, Med V University
samiramri@research.emi.ac.ma
Rabat, Morocco

MOHAMED OUTAHAJALA
CESIC, IRCAM
Rabat, Morocco

This paper investigates how to best couple hand-annotated data with information extracted from an external lexical resource to improve part-of-speech tagging performance. Focusing mostly on Amazigh tagging, we introduce a decision tree and Markov model using TreeTagger system. This system gives 92.3 % accuracy on the Amazigh corpus, an error reduction of 15 % (18.45 % on unknown words) over the same tagger without lexical information. We perform a series of experiments that help understanding how this lexical information helps improving tagging accuracy. We also conduct experiments on datasets and lexicons of varying sizes in order to assess the best trade-off between annotating data versus developing a lexicon. We find that the use of a lexicon improves the quality of the tagger at any stage of development of either resource, and that for fixed performance levels the availability of the full lexicon consistently reduces the need for supervised data.

Key words: POS tagging, Amazigh, Treetagger, Machine Learning, NLP, Tagset.

1 Introduction

Part of Speech (POS) Tagging is a very basic and well-known Natural Language Processing (NLP) problem which consists of assigning to each word of a text the proper morphosyntactic tag in its context of appearance. It is very useful for a number of NLP applications: as a preprocessing step to syntactic parsing, in information extraction and retrieval, statistical machine translation, corpus linguistics, etc.

The base of POS tagging is that many words being ambiguous regarding their POS, in most cases they can be completely disambiguated by taking into account an adequate context.

For example in Amazigh language: the word « tazla » on the sentence « day tazla uslmad » (« the teacher run » in English) is disambiguated as a verb because it is preceded by the preverbal particle “day”. Although, in this case the word is disambiguated simply by looking at the preceding tag, it must be taken into account that the preceding word could be ambiguous, or that the necessary context could be much more complicated than merely the preceding word. Furthermore, there are even cases in which the ambiguity is non-resolvable using only morphosyntactic features of the context, and require semantic and/or pragmatic knowledge.

For this purpose, our work involves the construction of dataset and the input pre-processing in order to run the two main modules: training program and tagger itself. For this reason, this work is the part to the still scarce set of tools and resources available for Amazigh automatic processing.

The rest of the paper is organized as follows. Section 2 puts the current article in context by overviewing related work. Section 3 describes Amazigh language particularities. Section 4 presents the used Amazigh tagset and our training corpus. Experimentation results are discussed in Section 5. Finally, we will report our conclusions and eventual future works.

2 Related Work and Motivation

In this section we briefly explore the related works on Amazigh POS tagging and our motivation and goals for this project and writing this paper.

2.1 Related Work

POS tagging is a well-studied problem in NLP, in which the aim, given a natural language text, is to a label each word with a POS tag such as noun, verb, adjective or others.

Different tagging systems can use different sets of tags. Typically a tag describes a word class and some word class specific features, such as number and gender.

Most POS tagger involves two problems:

- Finding the exact tags for each word. This can be easy if the word is in a word tag lexicon, but if the word is unknown, this may be tough to do.
- Choosing between the possible tags. This is called syntactic disambiguation, and it has to be solved for each word that is ambiguous in its POS.

Ambiguous words are very common in most languages. For example the Amazigh word set 'ⵉ ⵍ ⵍ ⵉ ' (illi) can be either a noun (daughter), or a verb (exist). Two factors that determine the tag of a word are its lexical probability and its contextual probability [1, 2].

Moreover, a lot of effort has been devoted to improving the quality of tagging process in terms of accuracy and efficiency. Existing taggers can be classified into three main groups according to the kind of knowledge they use: linguistic, statistic and machine-learning family. Of course some taggers are difficult to classify into these classes and hybrid approaches must be considered.

Within the linguistic approach most systems codify the knowledge involved as a set of rules written by experts. The linguistic models range from a few hundred to several thousand rules, and they usually require years of labor.

The most extended approach nowadays is the statistical family (obviously due to the limited amount of human effort involved). Basically, it consists of building a statistical model of the language and using this model to disambiguate a word sequence. The language model is coded as a set of co-occurrence frequencies for different kinds of linguistic phenomena.

This statistical acquisition is usually found in the form of n-gram collection, that is, the probability of a certain sequence of length n is estimated from its occurrences in the training corpus.

In the case of POS tagging, usual models consist of tag bi-grams and tri-grams (possible sequences of two or three consecutive tags, respectively). Once the n-gram probabilities have been estimated, new examples can be tagged by selecting the tag sequence with highest probability. This is roughly the technique followed by the widespread Hidden Markov Model taggers.

Stochastic methods, more than rule-based methods, have used annotated corpora for POS tagging. Two of the well understood and used stochastic methods were discussed: Markov models and Decision tree methods. These approaches and many others have performed with accuracies ranging from 96 %

to 97 %. It is believed that this is the level of accuracy that can be attained with the present annotated corpora due to annotation inconsistencies.

In area of POS tagging, many studies have been made. It reached excellent levels of performance through the use of discriminative models such as maximum entropy models [MaxEnt] [3, 4], Support Vector Machines [SVM] [5, 6] or Markov Conditional Fields [CRF] [7, 8], and TNT [9] which uses stochastic trigram HMM tagger and a suffix analysis technique to estimate lexical probabilities for unknown tokens based on properties of the words in the training corpus sharing same suffixes. Then, decision trees have been used for POS tagging and parsing as in [10]. Decision tree induced from tagged corpora was used for POS disambiguation [11].

For Amazigh POS tagging, Outahajala et al. built a POS tagger for Amazigh [12], as an under-resourced language. The data used to accomplish the work was manually collected and annotated. To help increasing the performance of their tagger, they used machine learning techniques (SVM and CRF) and other resources and tools, such as dictionaries and word segmentation tools to process the text and extract features' sets consisting of lexical context and character n-grams. The corpus contained 20,000 tokens and was used to train their POS tagger models [12].

2.2 Motivation and Goals

There is a pressing necessity to develop an automatic POS tagger for Amazigh. With this motivation, we identify the major goals of this work:

- We wish to investigate different machine learning algorithms to develop a POS tagger for Amazigh.
- This work also includes the development of a reasonably good amount of annotated corpora for Amazigh, which will directly facilitate several NLP applications.
- Amazigh is a morphological rich language. We wish to use the morphological features of a word to enable us to develop a POS tagger with limited resources.
- Finally, we aim to explore the appropriateness of different machine learning techniques by a set of experiments and also a comparative study of the accuracies obtained by working with different POS tagging methods.

3 Amazigh Language Particularities

3.1 Amazigh Language

Amazigh, also called Berber, belongs to the Hamito-Semitic "Afro-Asiatic" languages [13]. Amazigh is spoken in Morocco, Algeria, Tunisia, Libya, and Siwa (an Egyptian Oasis); it is also spoken by many other communities in parts of Niger and Mali. It is used by tens of millions of people in North Africa mainly for oral communication and has been introduced in mass media and in the educational system in collaboration with several ministries in Morocco.

Amazigh is a difficult morphological language; it uses different dialects in its standardization (Tachelhiyt, Tarifiyt and Tamazight the three used in Morocco).

Amazigh, like most of the languages which have only recently started being investigated for NLP, still suffers from the scarcity of language processing tools and resources. In this sense, Amazigh language presents interesting challenges for NLP researchers. Therefore, POS tagging is an important and basic step in the processing of any given language.

3.2 Amazigh script

Amazigh is one of the languages with complex and challenging pre-processing tasks. Its writing system poses three main difficulties:

- Writing forms' variation that requires a transliterator to convert all writing prescriptions into the standard form 'Tifinaghe – Unicode'. This process is confronted with spelling variation related to regional varieties, and transcription systems, especially when Latin or Arabic alphabet is used.
- The standard form adopted 'Tifinaghe – Unicode' requires special consideration even in simple applications. Most of the existed applications were developed for Latin script.
- Different prescriptions differ in the style of writing words using or elimination of spaces within or between words.

3.3 The richness of Amazigh morphology

Amazigh language has a complex morphology and the process of its standardization is performed via different dialects [14, 15]. Amazigh NLP presents many challenges for researchers. Its major features are:

- Amazigh has its own script: the Tifinagh, which is written from left to right. The transliteration into Latin alphabet is used in all the examples in this article.
- It does not contain uppercase.
- It presents for NLP ambiguities in grammar classes, named entities, meaning, etc. For example, grammatically the word «ⵜⴰⵣⵍⴰ» (tazla) can function as verb «ⴰⵔⵜⴰⵣⵍⴰ» meaning (over it) or as name (race), etc.
- As most languages whose research in NLP is new, the Amazigh is not endowed with linguistic resources and NLP tools.

Amazigh language is a morphological rich language. The most used grammatical classes are Noun, Verb, Adjective or Adverb. Practically speaking, nouns and verbs are the base of the Amazigh morphology and the more important categories to focus on, as others can be derived from them. We will present below these two grammatical Amazigh categories:

Noun: we will expose the morphological structure of noun that is in Amazigh characterized by gender, number, and status. The noun is either masculine or feminine. It is plural or singular: plural starts from two. The noun is free or annexed.

The masculine noun: the majority begins by one of the vowels (a, i, u), example: «ⴰⵔⴳⴰⵣ» « argaz » (which means man) in free status or «ⵓⵔⴳⴰⵣ» « urgaz » in annexed status, «ⵉⵣⵎ» « izm » (lion), «ⵓⴷⵎ» « udm » (face). However, there are masculine words that begin with a consonant, example : «ⴼⴰⴷ» (fad) (thirst in English), «ⵍⴰⵣ» (laz) (famine in English).

The feminine noun: it usually starts with (ta, ti, tu). In sometimes it is generally obtained by adding to masculine noun the discontinuous affix (t: t). Exp: «ⵜⴰⵡⴰⴷⴰ» « tawada » (going), «ⵎⵍⵙⵉⵡⵜ» « mlsiwt » (garment).

The plural nouns of the form (i: an), (i: en) (i: awen) (i: iwen) or nouns that change vowel pattern. The initial vowel (a) is transformed in (i), when the vowel is (i = u), it remains unchanged. Examples: (ⵉⵣⵍⵉ | ⵉⵣⵍⴰⵏ) (izli | izlan), (ⴰⴼⵓⵙ | ⵉⴼⴰⵙⵏ) (afus | ifasn).

Verb: the morphological aspect of the verb in Amazigh depends primarily on the affixation and composition. Some verbs are derivations by affixation (prefixes, suffixes) and other verbs are necessarily derived from nouns, either from a verb and a noun or either from two verbs.

Examples: «ⴷⴷⵓ» (ddu) (go) and «ⵉⵜⵛⵛⴰ» (itcha) (eat).

Particle: is a function word that is not assignable to noun neither to verb. It contains pronouns, conjunctions, prepositions, aspectual, orientation and negative particles, adverbs, and subordinates. Generally, particles are uninflected words. However in Amazigh, some of these particles are flectional, such as the possessive and demonstrative pronouns.

4 Tagset and Corpus

4.1. Used Tagset

A POS tagset is a collection of labels which represent word classes. A coarse-grained tagset might only distinguish main word classes such as adjectives or verbs, while more fine-grained tagsets also make distinctions within the broad word classes, e.g. distinguishing between verbs in past and future tense. This is an important step for a lexical labeling work to be based on the word classes of language and will reflect all morphosyntactic relationships words of Amazigh corpus (Table 1):

N°	**TAG**	**Designation**
1	NN	Commun noun
2	NNK	Kinship noun
3	NNP	Proper noun
4	VB	Verb,base form
5	VBP	Verb,participle
6	ADJ	Adjective
7	ADV	Adverb
8	C	Conjunction
9	DT	Determiner
10	FOC	Focalizer
11	IN	Interjection
12	NEG	Particle, negative
13	VOC	Vocative
14	PRED	Particle,predicate
15	PROR	Particle,orientation
16	PRPR	Particle,preverbal
17	PROT	Particle,other
18	PDEM	Demonstrative pronoun
19	PP	Personal pronoun
20	PPOS	Possessive pronoun
21	INT	Interrogative
22	REL	Relative
23	S	Preposition
24	FW	Foreign word
25	NUM	Numeral
26	DATE	Date
27	ROT	Residual,other
28	PUNC	Punctuation

Table 1: Amazigh Tagset

4.2. Corpus

A corpus is a collection of language data that are selected and organized according to explicit linguistic criteria to serve as a sample of jobs determined a language. Generally, a corpus contains up few millions of words and can be lemmatised and annotated with information about the parts of speech.

Among the corpus, there is the British National Corpus (100 million words) [16] and the American National Corpus (20 million words) [17].

A balanced corpus would provide a wide selection of different types of texts and from various sources such as newspapers, books, encyclopedias or the web.

For the Moroccan Amazigh language, it was difficult to find ready-made resources. We can just mention the manually annotated corpus of Outahajala et al. [18]. This corpus contains 20k words using a tagset described in Table 1 that is why we decided to build our own corpus. In order to have a vocabulary sufficiently large, we took texts from tawiza website, texts from IRCAM website and from primary school textbooks … etc. We have collected these different resources; after that, we have cleaned them and convert them to UTF-8 Unicode format. Table 2 provides source statistics of our corpus which includes 3625 sentences (approximately 40,200 words):

Source	**%**
Online newspapers and periodicals	22.7
Primary school textbooks	15
Texts from websites of organizations	10.4
Texts from government websites	8.6
Miscellany	16.5
Blog	15
Texts from website of IRCAM	12.8

Table 2: Constituents of Amazigh corpus

4.3. Annotation of the corpus

The morpho-syntactic annotation of our raw corpus is doing on two steps: an automatic assignment of labels by the existing tagger and then a revision thereof by a human annotator. We find this way to precede the construction of the Penn Treebank corpus [19].

For this, to annotate our raw Amazigh corpus we used the Amazigh language model developed with probabilistic tagger CRF++ [20]. This tagger assigns the proper grammatical classes, defined on the tagset presented in Section 4. This tagger is based on a supervised learning model.

From the reference corpus previously tagged manually [18], this tagger learns a language model that allows it to label our raw Amazigh corpus. So we established our reference corpus, labeled, corrected and segmented it.

We created, using a Perl program, a glossary of words included in the corpus. This program assigns for each word its different possible morphosyntactic classes and their number occurrences. We also created, for each word in the corpus, a lexicon trigram that contains triplets: word, tag and lemma. This lexicon contains words' morphosyntactic classes and their lemmas. It allows inferring the morphosyntactic class for unknown words and establishing a connection diagram between each word, its POS class and the words of its entourage.

Moreover, in order to make Amazigh corpus easy to use, we produced a CSV format which contains one word per line associated with its morphosyntactic information. We also used symbols to facilitate reading the corpus as follow:

- The '/' symbol to separate between Amazigh script and its transliteration on Latin.
- The '|' symbol to separate between the word in Latin transliteration and its POS tag.

To illustrate this, Figure 1 shows an example in our corpus for the sentence: «ⵜⵉⵡⵓⵔⵣⴰ ⵓⵔⴷ ⴰⵖⵏⵏⴰ ⵏⵥⵔⵔⴰ ⴰⴷⴷⴰⵣ ⵜⵣⵏⴷ, ⵛⴰⴽⴰ ⵜⵉⵡⵓⵔⵣⴰ ⴷ ⵉⵙⴽⴽⵉⵏ ⵏⵏⴰ ⴰⴽ

ⵉⵜⵜⴰⵎⵎⴰⵏ ⴰⵜⴳⵏⴷ.» (Which means in English: dreams are not what you see when you sleep, dreams are those things that keep you from sleeping.)

ⵜⵉⵡⵓⵔⴳⴰ /tiwurga|NN ⵓ ⵓr|NEG ⴷ /d|S ⴰⵢⵏⵏⴰ /aynna|REL
ⵏⵥⵔⵔⴰ /nZRRa|VB ⴰⴷⴷⴰⴳ /addag|ADV ⵜⴳⵏⴷ /tgnd|VB ,|,
ⵎⴰⴽⴰ /maka|PRED ⵜⵉⵡⵓⵔⴳⴰ /tiwurga|NN ⴷ /d|CC ⵉⵙⴽⴽⵉⵏ /iskkinn|NN
ⵏⵏⴰ /nna|PROT ⴰⴽ /ak|PRPR ⵉⵜⵜⴰⵎⵎⴰⵏ /ittamman|VB ⴰⵜⴳⵏⴷ /atgnd|VB .|.

Fig 1: An annotated sentence from Amazigh corpus

5 Experiment Settings and Results

5.1. Methods and tools

We choose TreeTagger system (hencefore TT) which is a basic Markov Model tagger and makes use of decision trees to get more reliable estimates for contextual parameters.
TT assumes trigram transition probabilities. To deal with data sparseness, the trigram probabilities are estimated by growing a decision tree.

5.1.1. Decision Trees

Decision trees recently used in many NLP tasks, such as automatic speech recognition, POS tagging, parsing, disambiguation sense and information retrieval. TT estimates the transition probabilities with a binary decision tree [21]. The initial step of constructing the decision tree happens during the training phase. It will parse through the text and analyses trigrams, inserting each unigram into the tree. For a given node in the tree, the probability of which tag to use is obtained from the two previous nodes (trigram). Once the tree is created, its nodes are pruned. If the information gain of a particular node is determined below a defined threshold, its children nodes are removed. Figure 2 represents simplified version of a decision tree for Amazigh language.

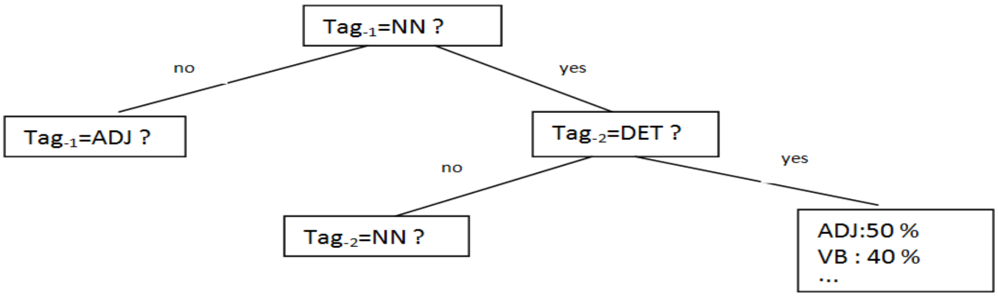

Figure 2: A simplified decision tree for Amazigh

5.1.2. Hidden Markov Models (HMM)

HMM is a generative statistical model of a Markov process with hidden states [22, 23]. The intuition behind HMM and all stochastic taggers is a simple generalization of the "pick the most likely tag for this word" approach. The unigram tagger only considers the probability of a word for a given tag t; the surrounding context of that word is not considered. On the other hand, for a given sentence or word sequence, HMM taggers choose the tag sequence that maximizes the following formula: P (word | tag) * P (tag | previous n tags)

To illustrate POS tagging via HMM we take the sentence: « itcha yan urgaz aghrum » (a man ate the bread) (Figure 3):

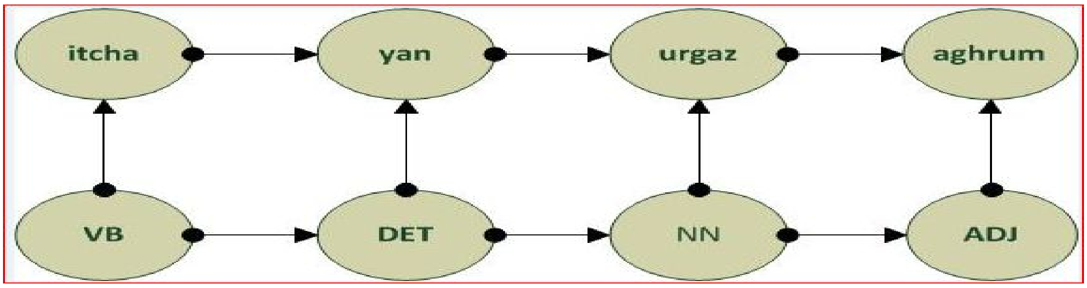

Figure 3: Graphic illustration of Hidden Markov Model

5.2. Results and Discussion

We recall that our corpus training was performed using the tagger described in the previous Section and we set its contents after a long adjustment and manual checking of about 40000 words. To evaluate our work, we used precision which means the proportion of correct tags from the tagging set. To perform this evaluation we used the tools included in TT.

Before presenting the results of our assessment, we describe our work corpus. We have carried out our assessment using 9 training corpora. Each training corpus is a subset of our global dataset: the first one represents 10% (4000) of the 40000 and the second one is constructed of 20% tokens (8000) until reach the ninetieth corpus which its size is 90% (36000) of the main corpus. For these 9 taggers we used the rest of the reference corpus as test corpus.

Analysis of the precision rate (Figure 4) of our tagger indicates that the best one, 92.37%, is achieved when the text size reaches 36000 tokens. In this situation, the number of unknown words is less than 20%.

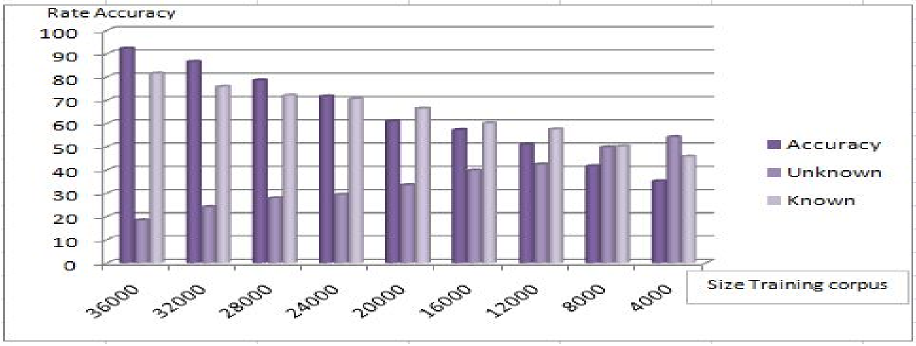

Figure 4: Rate accuracy of Amazigh POS tagging

Our scores are low at first sight compared to the precision rate of 97.5% achieved by TT on German corpus [21]. The significant difference of the performance between Amazigh and German is due mainly in the size of training corpus and in the morphological characteristics specific to each language.

We believe that for a first testing and evaluation of POS tagging of a less resourced language as Amazigh, TT is highly efficient. Other parameters must be taken into account to evaluate the tagging of an Amazigh corpus with this tagger like the size and the quality of the corpus.
We also checked the percentage of unknown and known words in every phase of our evaluation. This information is summarized in the Table 3:

Phase	Size	Accuracy	Unknown	Known
9	36000	**92.3**	**18.45**	**81.55**
8	32000	86.57	24.12	75.88
7	28000	78.59	27.96	72.04
6	24000	71.75	29.45	70.55
5	20000	61.02	33.56	66.44
4	16000	57.25	39.86	60.14
3	12000	51.01	42.47	57.53
2	8000	41.7	49.8	50.2
1	4000	35.24	54.2	45.8

Table 3: Summary of the evaluations

Outahajala et al. used SVMs and CRF for experimentation of Amazigh POS tagging [12]. However, CRFs outperformed SVMs on the 10 folds average level.

Comparing our results got with those of [12] (88.66% for SVM and 88.27% for CRF), we can deduce that these results are encouraging, and it is desirable to integrate other morphological features to improve the accuracy, considering that we have used corpus of only ~40k tokens with a tag set of 28 tags.

5.3 Error analysis

The most common types of errors are the confusion between proper noun and common noun and the confusion between adjective and common noun. These results from the fact that most of the proper nouns can be used as common nouns and most of the adjectives can be used as common nouns in Amazigh.
Almost all the confusions are wrong assignment due to less number of instances in the training corpora, including errors due to long distance phenomena.

6 Conclusion

In this work we have presented and evaluated a machine-learning based algorithm for obtaining statistical language models oriented to Amazigh POS tagging. We have directly applied the acquired models in a simple and fast tree-based tagger obtaining fairly good results. We also have combined the model with an Amazigh lexical resource to improve the accuracy.

We are also especially interested in extending the experiments involving combinations of more than two taggers in a double direction: first, to obtain less noisy corpora for the retraining steps in bootstrapping processes; and second, to construct ensembles of classifiers to increase global tagging accuracy. We plan to apply these techniques to develop taggers and annotated corpora for Amazigh language in the near future.

More detailed research should be done in order to establish quantitative conclusions to compare tagger performances. The cross evaluation of the main state-of-the-arts taggers in a range of operating conditions is a work we plan to start in the short run. It is also necessary to establish a standard benchmark for the evaluation of POS taggers, to reliably evaluate the results of future research in this field.

References

1. Voutilainen, A. Part-of-speech tagging. The Oxford hand book of computational linguistics,2003, (pp. 219–232).
2. Sun, G. , Lang, F. and Qiao, P. Chinese part-of-speech tagging based on fusion model. In Proceedings of the 11th joint conference on information sciences,2008, Amsterdam: Atlantis Press.
3. Ratnaparkhi, A. a Maximum Entropy Model for Part-Of-Speech Tagging. In Proceedings of EMNLP, Philadelphia, USA 1996
4. Toutanova, K. and Manning, C. Enriching the knowledge sources used in a maximum entropy part-of speech tagger. In EMNLP/VLC 1999, pages 63–71.
5. Giménez, J. and L. Màrquez , L. SVMTool: A General POS Tagger Generator Based on Support Vector Machines. In Proceedings of the 4th International Conference on Language Resources and Evaluation, Lisbon, Portugal, 26–28 May 2004, pp. 43--46.
6. Kudo, T. and Matsumoto, Y. Use of Support Vector Learning for Chunk Identification. In: Proc.of CoNLL-2000 and LLL-2000.
7. Lafferty, J., McCallum, A. and Pereira, F. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. In: Proc. of ICML-01,2001, pp. 282-289.
8. Tsuruoka, Y., Tsujii, J. and Ananiadou, S. Fast full parsing by linear-chain conditional random fields. In Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics (EACL 2009), p. 790–798.
9. Brants , T. 2000. TnT – A statistical part-of-sppech tagger. In Proceedings of the 6th Applied NLP Conference. 224-231.
10. Black , E., Jelinek, F., Lafferty, J. , Mercer, R. and S. Roukos, S.1992. Decision tree models applied to the labeling of text with parts-of-speech. In Proceedings of the DARPA workshop on Speech and Natural Language, Harriman, New York.
11. Màrquez, L. and Rodríguez, H. 1998. Part of Speech Tagging Using Decision Trees. Lecture Notes in AI 1398-C. Nédellec & C. Rouveirol (Eds.). Proceedings of the 10th European Conference on Machine Learning, ECML'98. Chemnitz, German
12. Outahajala, M., Benajiba, Y., Rosso, P. and Zenkouar, L. POS Tagging In Amazigh Using Support Vector Machines And Conditional Random Fields. In Natural Language to Information Systems, LNCS (6716), Springer-Verlag, pp, 238—241, 2011
13. Cohen, D. Chamito-sémitiques (langues). In Encyclopædia Universalis 2007.
14. Chafiq, M. (1991).[Forty four lessons in Amazigh]. éd. Arabo-africaines
15. Chaker, S. Textes en linguistique berbère -introduction au domaine berbère, éditions du CNRS,1984, pp 232-242
16. BURNARD, L. The British National Corpus,1998
17. IDE, N. and MACLEOD, C. The american national corpus : A standardized resource of American english. In Proceedings of Corpus Linguistics 2001, volume 3.
18. Outahajala, M., .Zenkouar, L. and Rosso, P. Building an annotated corpus for Amazigh. In Proceedings of 4th International Conference on Amazigh and ICT, 2011, Rabat, Morocco.
19. Marcus, M. P., Marcinkiewicz, M. A. and Santorini, B. Building a Large Annotated Corpus of English: The Penn Treebank. Computational Linguistics, 19(2), 313-330, (1993).

20. Outahajala, M. and Rosso, P. Using a Small Lexicon with CRFs Confidence Measure to Improve POS Tagging Accuracy, Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016), portoroz, Slovenia.
21. Schmid, H. Probabilistic part-of-speech tagging using decision trees. In International Conference on New Methods in Language Processing, Manchester, UK, 1994, pages 44-49
22. Manning, C. and Schütze, H. Foundations of Statistical Natural Language Processing. The MIT Press,1999.
23. Toutanova, K. Dan, K. Manning, C. and Yoram, S..Feature-Rich Part-of Speech Tagging with a Cyclic Dependency Network. In Proceedings of HLT-NAACL 2003 pages 252-259.