Evaluating Solutions for the Rapid Development of State-of-the-Art POS Taggers for Portuguese

António Branco and João Silva
Department of Informatics, University of Lisbon
Faculdade de Ciências, Campo Grande, 1749-016 Lisboa
{ahb,jsilva}@di.fc.ul.pt

Abstract

We report on solutions we adopted for the specific issues that arise when developing new automatic taggers for Portuguese, solutions whose design is general enough, we believe, to be further reused to develop other new taggers for this language, even when using different training data than those we used in our experiments. We report also on the evaluation of tools that make use of such solutions and show that the latter permit to develop POS taggers for Portuguese whose performance matches or surpasses state-of-the-art results obtained for other languages when using the same technology.

Introduction

The application of machine learning techniques to natural language POS tagging has matured to a point where it is now very rapid to develop new, state-of-the-art accuracy taggers (cf. Samuelsson & Voutilainen, 1997; Brill, 1995; Ratnaparkhi, 1996; Brants, 2000 a.o.). Provided that the training data is ready, obtaining a new tagger may be as rapid as a few seconds with some applications. Given the general-purpose of these techniques, this holds true for every language that they have been tried upon even though most of the initial research has been conducted over data from English. Accordingly, and letting aside the time required to accurately annotate the training corpus, the bulk of the time span needed to prepare a new tagger is determined basically by the time needed to prepare tools to handle language-specific issues. Such issues are found in each of the three major steps involved in the automatic tagging *sensu latu* of raw text, namely chunking, tokenizing, and tagging *sensu stricto*.

In the present paper, we report on solutions we arrived at for the specific issues that arise when developing a new automatic tagger for Portuguese and that are generic enough to be further reused to develop other new taggers for this language, possibly from other training data. We focus on the evaluation of tools that make use of such solutions and show that the latter permit to develop POS taggers for Portuguese whose performance match state of the art results obtained for other languages when using the same technology.

Chunker

As in other languages with orthographic conventions similar to those adopted for Portuguese, designated punctuation symbols ('.', '?', '!',...) are used to mark the end of sentences. Most sentence boundaries can then be detected when these terminators precede sentence starters, i.e. designated orthographic clues marking the beginning of a subsequent sentence (viz. word beginning with a capital letter) – the expected abbreviation/period ambiguity of '.' can be addressed by means of the solutions proposed for other languages (Mikheev, 2002).

Conventions for sentence bounding that are specific to Portuguese, or at least not found in other close Romance languages or English under exactly the same format, involve the marking of paragraph (turn taking) and sentence boundaries in written dialogue.

The beginning of the first sentence containing a character's turn is easily handled as this starts with a dash ('–') immediately followed by the usual sentence starters.

```
<s> – Bom dia! </s>
```

Things get convoluted when it comes to narrator's asides. The beginning of a narrator's aside cannot initiate an utterance. It is always indicated by a dash and its ending is indicated by a dash if the aside does not conclude the sentence, or by a period if it is the last part of its sentence:

```
<p><s> – Apetece-me ir ao cinema – anunciou ele. </s></p>
```

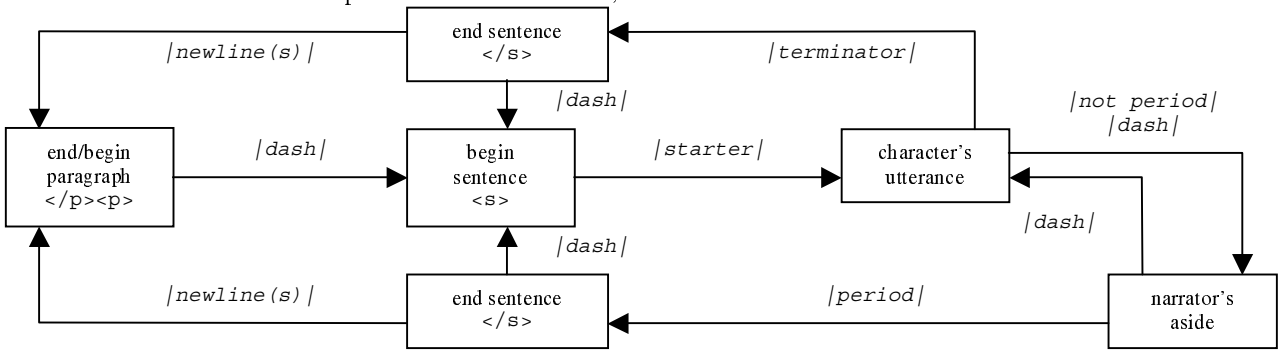
```
<p><s> – Eu cá – disse ela – também quero. </s></p>
```

The fact that the preceding sentence has been concluded with a narrator's aside or not determines the way the beginning of the next utterance is marked. A character's sentence other than the first one in the current turn starts also with a dash if and only if it follows a sentence ending with a narrator's aside.

```
<p><s> – Vamos ao jardim. </s><s> Está um lindo dia. </s></p>
```

```
<p><s> – Não – replicou ela. </s><s> – Eu não vou. </s></p>
```

As for termination symbols of character's utterances, only those that are different from a period can appear before the beginning of a narrator's aside.



```
<s> — Bom dia! — exclamou ela. </s>
<s> — Mau dia — retorquiu ele, azedo. </s>
```

A perspicuous way of compiling and displaying the conventions related to written dialog orthographic format is by mean of a finite state automaton (FSA). The states represent concepts such as "character's utterance" or "narrator's aside" and the transitions between these states are triggered by the occurrence of specific sequences of symbols in the input. For example, the FSA above indicates that, when running over text, if the chunker is in a character's utterance, the occurrence of a sequence of two tokens separated by blank(s), where the first token is not a period and the second is a dash, will be taken as the start of the narrator's aside; on the other hand, if the chunker is in a narrator's aside, the occurrence of a dash will be taken as the end of that aside and the continuation of the character's utterance that preceded the aside.

Based on the above procedure, we developed the LX-Chunker, a handcrafted rule-based tools written in Flex. It scored a recall of 99.95% and precision of 99.92% when tested on a 12 042 sentence (230 Ktoken) corpus accurately hand tagged with respect both to morphosyntactic categories of the tokens and to sentence and paragraph boundaries.[1] This corpus (LX-Corpus) is made of news texts and novels with dialogue parts (17.90% of the sentences).[2]

We have made experiments with some other sentence chunkers that either are specific for Portuguese or that can be trained for any language, including Portuguese. A Perl module, by Simões, is available at the CPAN site (http://www.cpan.org). When this handcrafted rule-based chunker is run over the LX-Corpus, it evaluated to 98.21% of recall and 93.41% of precision. We also experimented with the MXTerminator, a maximum entropy-based tool to train sentence chunkers (Reynar & Ratnaparkhi, 1997). We used it to train a chunker over the same corpus used for evaluating the previously referred chunkers. Even when evaluated in biasing conditions favouring a higher score, namely when evaluated over the training corpus, it scored 85.88% of recall and 97.09% of precision.

As expected, these experiments seem to indicate that, for texts including dialogue-specific orthographic conventions, our proposal makes a relevant contribution to enhance chunking quality.

Tokenizer

For most tokens in a raw text, tokenization is a trivial procedure, consisting in detaching punctuation marks and taking advantage of the whitespace as a delimiter symbol. There are, however, a few non-trivial cases (complete list In Branco & Silva, 2003a) that involve tokenization-ambiguous strings, i.e. strings that can be tokenized in more than one way: `deste` → `|deste|` or `deste` → `|de|este|`

In a general setup like ours, where one counts on a tagger trained over previously annotated data, this type of difficulties inevitably introduces some circularity: Although tagging decisions require that a previous tokenization process has been completed, the tokenization of these ambiguous strings requires previous knowledge of the POS tag of the token(s) corresponding to the string. For instance, in the example above, `deste` would be tokenized as one token if and only if it had been tagged as a Verb, but for it to be tagged as a Verb it should have already been tokenized as one token. In order to dissolve this circularity and correctly handle type-ambiguous strings, we used a two-level approach to tokenization where tagging is interpolated into the tokenization process, proceeding now in two stages, one before and another after the tagger has been applied. Accordingly, (i) a pre-tagging tokenizer definitely identifies every token except those related to ambiguous strings, which are provisionally identified as one token. (ii) Subsequently, the tagger assigns a composite or a simple tag to every ambiguous string depending on it being a contracted or a non-contracted form, respectively: The tagger has been trained over a corpus where ambiguous strings are always tokenized as a single token and annotated with single or composite tags. For instance, the string `deste` is tagged either a `deste_V` or as `deste_PREPDEM`. (iii) Finally, a post-tagging tokenizer handles only ambiguous strings, breaking those that are tagged with a composite tag into two tokens and corresponding tags.[3] In order to implement the two-level tokenization approach just described, we used Ratnaparkhi's MXPoST system (Ratnaparkhi, 1996) to train a tagger for Portuguese. This system offers a state-of-the-art level of performance, having permitted to develop a tagger with 97.08% of precision (with every token being assigned a single tag). It was trained over the LX-Corpus, where the ambiguous strings amount to 2% of the tokens. This approach permitted to successfully resolve 99.4% of these ambiguous cases, against a baseline of 78.2% of success. This baseline is obtained with the rough and ready heuristic of tokenizing every ambiguous string into two tokens, a heuristic straightforwardly suggested by the fact that 78.2% of the ambiguous strings are contractions in the test corpus.

Tagger

With suitable solutions for the Portuguese-specific issues concerning chunking and tokenization in place, the last step in the task of tagging raw text is the tagging procedure *sensu stricto*. That is, given that sentential and lexical tokens have been identified, the step yet to accomplished is to assign the POS tag to each lexical token, possibly taking into account the neighbouring boundaries of the containing sentence or paragraph.

When using machine-learning tools out of the shelf to develop a new tagger, the remaining critical issues dwell around the gathering of appropriate training data. For the sake of the focusing on the language-critical issues involved in developing a new tagger, let us assume that one can rely on a previously annotated corpus as a starting point. Let us further assume that the consistency and accuracy of the annotation of such a general-purpose training corpus is ensured. The remaining concern is then directed towards manipulating and relabeling the training data in accordance with the tagset that needs to be opted

[1] Precision = tp/(tp+fp) and Recall = tp/(tp+fn), with tp=true positives (chunked correctly), fp=false positives (chunked erroneously), fn=false negatives (did not chunk when it should).

[2] This corpus was prepared from a corpus kindly granted by CLUL-Centro de Linguística da Universidade de Lisboa (Nascimento et al., 2000). We are very grateful to Fernanda Nascimento e Amália Mendes for their help

[3] For a detailed rendering of this, see (Branco & Silva, 2003).

for. The design of the tagset turns out thus to be the non-trivial, language-specific aspect that calls to be addressed.

In this respect, given that statistically based applications will be used, one finds the usual tension between increasing the discriminative power of the tagger — by using more tags — and minimizing the data sparseness — by using fewer tags. The search for the best performance of a POS tagger supported by a suitably tuned balance of these two attractors cannot be reduced, however, to arbitrarily playing around with the number and the assignment of tags. Syntactic categorization encodes basic linguistic generalizations about the distribution of lexemes, which by their own nature, are to be empirically uncovered, not superimposed in view of stipulative convenience.

Taking the preceding considerations into account, there are possible "candidate" categories or subcategories that should not to be included in the tagset used to annotate the corpus over which the tagger is to be trained. In the first place, different tags not justified by different distribution are to be excluded. This is the case, for instance, of tags indicating the degree of an adjective (example: `alto_ADJNORM`, `altíssimo_ADJSUP`). Tough conveying some distribution-related information, there may be tags that can be unequivocally inferred from the form of the token at stake. In view of decreasing the data sparseness, such tags should be avoided. For example, this is the case of tags indicating the polarity of an adverb (example: `sim_ADVPOS`, `nem_ADVNEG`), or tags indicating inflectional features, which can be subsequently determined from suffixes by a lemmatizer (example: `alto_ADJMascSing`, `altas_ADJFemPlu`). Also, when considering tagsets proposed in grammar textbooks of a more traditional, philological-oriented persuasion, it is not unusual to find categories aimed at indicating the constituency status of the phrase containing the relevant token. Such different tags encode information about whether the token at stake is a constituent of an elided or of a non-elided phrase but not an actual difference with respect to the syntactic distribution of that token. One example of this is the category "indefinite pronoun" versus some other category of closed classes. This category has been proposed for tagging articles, demonstratives or other pronominal items in headless Noun Phrases. For instance, according to such traditional views, the demonstrative `aquele` would receive `DEM` in the non-elliptical NP in `li [aquele_DEM livro]`$_{NP}$ but it would receive `INDPRON` in the corresponding elliptical NP in `li [aquele_INDPRON Ø]`$_{NP}$). Given that no difference with respect to syntactic distribution of items like `aquele` is at stake, and in view of taming the data sparseness effect, the tags indicating the elliptical status of the containing phrase have no place in our tagset. Returning to the specific examples above, `aquele` receives the same tag on both cases and the last example is tagged as: `li [aquele_DEM Ø]`$_{NP}$ Under more traditional approaches, single-word NPs like `tudo` are also proposed to receive the "indefinite pronoun" tag or a similar one. It is understood that a tag like `IN` (Indefinite Nominals) should be included to cover these cases.

It is of note that the rationale discussed above and followed to circumscribe the tagset, not only helps to exclude possible candidate tags, but also to isolate and include categories that are usually not taken into account in a more traditional perspective.

Though being verbal forms, gerund, past participle and infinitive forms each have a distribution of its own because they are the main predicators of subordinate clauses with specific distribution. Moreover, infinitival forms support nominative constituents (e.g. `[ouvir_INF música]`$_{NP}$ `diminui o stress`) and past participle can be used with adjectival force (e.g. `o candidato eleito_PTP não chegou a tomar posse`). The tags `GER`, `PTP` and `INF` are thus included in the tagset to enhance the discriminative power of the tag.

Other "non-canonical" tags are also included. These may be less interesting from a general linguistic point of view but they are important to improve also the contribution of the tagger for subsequent processing stages, e.g. named entity recognition. They cover dialogue particles (`adeus`, `olá`) social titles (`Pres.`, `Drª.`), part of addresses (`Rua`, `Av.`,), email addresses, months (`Janeiro`, `jan.`), days of the week (`Terça-feira`, `ter.`), measurement units (`km`, `kg`) as distinct syntactic classes. Our tagset includes also specific tags for digits, roman numerals, denominators of fractions (`meio`, `terço`), orders of magnitude (`centenas`, `biliões`), symbols (`/`, `#`) and letters.

Finally, in order to tag multi-word expressions from closed classes, a special tagging scheme is used where each component word receives the same tag prefixed by `L`, and followed by the corresponding index number. For example: `apesar_LPREP1 de_LPREP2`.

With the full tagset for the training data (Branco & Silva, 2003b) defined under the above guidelines, we prepared the LX-Corpus, over which we trained different taggers using different algorithms: Brill's TBL (Transformation-based), Brants's TnT and Tufis & Mason's QTag (HMM), Ratnaparkhi's MXPoST (Maximum-entropy). Accuracy measurement for the taggers was obtained averaging 10 test runs. Each run over a held out evaluation corpus with 10% of consecutive lines not used for training:[4]

System	TBL	TnT	MXPOST	QTag
Accuracy	97.09%	96.87%	97.08%	89.97%

Table 1: Taggers accuracy for Portuguese

These systems were originally applied to languages other than Portuguese. They were used to develop taggers over labelled corpora with different tagsets and length wrt LX-Corpus. It is nevertheless instructive to compare the evaluation results obtained now for Portuguese with the results obtained with the same development tools for other languages, even if the evaluation methodologies do not completely coincide. Brill's TBL tagger for English was developed with the help of 1.1 Mtokens of the Penn-Treebank/Wall Street Journal (PT-WSJ) corpus labelled with a 45 category tagset. It was trained over 950 Ktokens and evaluated over the remaining 150 Ktokens. It scores 96.50% of accuracy (Brill, 1995). TnT tagger for English was developed over approximately the same labelled corpus (1.2 Mtokens of PT-WSJ). It was evaluated with the same methodology used for the Portuguese taggers, scoring 96.70% (Brants, 2000). MXPoST tagger for English was also developed over the PT-WSJ. It was trained over 90% of the corpus and tested in one run over the remainder 10%. It scored 96.60% accuracy (Ratnaparkhi, 1996). Finally, QTag tagger for Romanian was trained over 250 Ktokens labelled with a tagset with 89 categories. The evaluation used "several" runs of 90%

[4] See an online demo at http://lx-suite.di.fc.ul.pt

training vs. 10% test and scored 96.22% (Tufis & Mason, 1998). Given these values, it worth noting that the Portuguese taggers, though developed over a shorter corpus, performed slightly better than the taggers for English with corresponding learning algorithm. As for QTag, its better performance is because the HMM tagger for Romanian was combined with rule-based tag guesser for unknown words taking into account word endings.

Focusing now on taggers developed specifically for Portuguese, it is possible to evaluate or to collect evaluation results for the systems that are available at the time of writing this paper. Though these taggers may have been built over corpus of quite different length and genre, and using tagsets with different size, for the completeness sake, it is certainly interesting to collect a brief overview of the current state-of-the-art in this respect. The Tycho Brahe tagger is based on Brill's TBL system (Finger, 2000). Trained on 130 Ktokens of the Tycho Brahe corpus, made of historic Portuguese texts labelled with 33 tag tagset, and evaluated over 45 Ktokens, it scored 88.24% of accuracy. Additional refinement modules raise that score to 95.43% accuracy. In (Aires, 2000), three taggers are evaluated and compared. All use the same 105 Ktoken mixed-type corpus labelled with a 42 tag tagset. Training is done over 80% of the corpus. TreeTagger, which is based on a decision tree procedure, scored 86.47%. Brill's TBL did better, scoring 88.76% of accuracy. MXPoST in turn scored 89.66%. Exhibiting the best result, MXPoST was then trained over 90% of the corpus and evaluated over the remaining 10%, achieving 90.25% accuracy. In (Aluísio *et al.*, 2003) three taggers were trained over 80% of the MAC-MORPHO corpus of news texts. The remaining 20% were used for evaluation. Brill's TBL scored 90.74%, TreeTagger scored 94.54% and MXPoST scored 95.92%. Palavroso/MARv is a morphological analyser coupled with an ambiguity resolver (Ribeiro *et al.*, 2003). It was trained over 230 Ktokens and evaluated over 60 Ktokens, of a corpus labelled with a tagset of 54 tags. This tagger scored 94.23% accuracy. Under the same training and evaluation conditions, a tagger obtained with Brill's TBL system scored 95.17% accuracy. A tagger for Portuguese has been developed by CEPRIL[5]. It uses QTag as the underlying system. It was trained on a 500 Ktoken corpus of news text labelled with a tagset of 15 tags. It scored, on average, 93% accuracy. The EMS tagger uses Brill's TBL tagger coupled with the JSpell morphological analyser (Reis & Almeida, 1998). Trained on a 10 Ktoken corpus labelled with a 200 tag tagset it scored 96% of accuracy. [6]

[5] Online demo at: http://lael.pucsp.br/corpora/etiquetagem

[6] For the sake of completeness, it is worth referring handcrafted ruled-based taggers, even tough they lie outside the scope of this paper, concerned with solutions for rapid development of taggers for Portuguese: PosiTagger (Aires, 2000), a symbolic tagger that uses Brill's TBL coupled with handcrafted rules, scored 82.65% of accuracy when tested over 10% of a 105 Ktoken mixed-type corpus labelled with a 42 tag tagset. The PALAVRAS tagger (Bick, 2000) uses a tagset with about 90 tags and subtags. It is reported to achieve between 98.8% and 99.7% accuracy over five very small test corpora in the range of 1,8-4,8 Ktokens, mostly made of news text. The small size of test data may justify the contrast with the score of mature taggers for English following the same constraint grammar approach (Samuelsson & Voutilainen, 1999).

Concluding remarks

In this paper, we presented solutions for language-specific issues in the development of taggers for Portuguese. These solutions are generic enough to be reused and thus to further reduce the time span required to develop taggers for Portuguese. We also presented evaluation results showing that, when coupled to shape a tagger for raw text, these solutions do not degrade overall accuracy and efficiency, keeping up or even surpassing state-of-the-art results obtained for other languages when using similar technology.

References

Aires, R., 2000, "Implementação, Adaptação, Combinação e Avaliação de Etiquetadores para o Português do Brasil". MsC Thesis, Univ. São Paulo, Brazil.

Aluísio, S., G. Pinheiro, M. Finger, M. Nunes and S. Tagnin, 2003, "The Lacio-Web Project: Overview and Issues in Brazilian Portuguese Corpora Creation". *Proc. of Corpus Linguistics 2003*, 14–21.

Bick, E., 2000, "The Parsing System PALAVRAS". PhD Thesis, University of Århus, Denmark.

Branco, A. and J. Silva, 2003a, "Contractions: breaking the tokenization-tagging circularity". In Mamede et al. (eds.) *Computational Processing of the Portuguese Language*, Berlin: Springer, LNAI 2721, 167—170.

Branco, A. and J. Silva, 2003b, "Swift Development of State-of-the-Art Taggers for Portuguese". In Branco *et al.* (eds.) *Proc. Workshop on Tagging and Shallow Processing of Portuguese*, Lisbon, Univ. of Lisbon, Dep. Informatics, TR28-03, 7–10.

Brants, T., 2000, "TnT - A Statistical Part-of-speech Tagger". *Proc. Applied Natural Language Proc.*, ACL, 224—231.

Brill, E., 1995, "Transformation-based Error-driven Learning and Natural Language Processing". *Computational Linguistics*, 21, 543—565

Finger, M., 2000, "Técnicas de Otimização da Precisão Empregadas no Etiquetador Tycho Brahe". *Proc. of V Encontro para o Processamento Computacional da Língua Portuguesa Escrita e Falada* .

Mikheev, A., 2002, "Periods, Capitalized Words, etc." *Computational Linguistics* 28(3), 289—318.

Nascimento, F., L. Pereira and J. Saramago, 2000, "Portuguese Corpora at CLUL" *Proc. 2nd Int. Conf. on Language Resources and Evaluation*, Paris: ELRA, 1603—1607.

Palmer, D. and M. Hearst, 1994, "Adaptive Sentence Boundary Disambiguation". *Proc. of the Conf. Applied Natural Language Processing.*

Ratnaparkhi, A., 1996, "A Maximum Entropy Part-of-speech Tagger". *Proc. of the Empirical Methods on Natural Language Processing*, ACL, 133—142.

Reis, R. and J. Almeida, 1998, "Etiquetador morfo-sintático para o Português". *Proc. XIII Congresso da Associação Portuguesa de Linguística*(2), 209–221.

Reynar, J.. and A.t Ratnaparkhi, 1997, "A Maximum Entropy Approach to Identifying Sentence Boundaries". *Proc. 5th Conference on Applied Natural Language Processing.*

Ribeiro, R., L. Oliveira and I. Trancoso, 2003, "Using Morphossyntactic Information in TTS Systems". In Mamede et al. (eds.) *Computational Processing of the Portuguese Language*, Berlin: Springer, LNAI 2721, 143–150.

Samuelsson, C. and A. Voutilainen, 1997, "Comparing a Linguistic and a Stochastic Tagger". *Proc. of Annual Meeting of ACL,* 246—253.

Tufis, D. and O. Mason, 1998, "Tagging Romanian texts". 1st *Int Conf on Language Resources and Evaluation.*