A BERT-based Approach for Part-of-Speech Tagging in the Low-Resource Context of Sardinian

Salvatore Mario Carta[1,3], Filippo Concas[1], Gianni Fenu[1], Alessandro Giuliani[1], Marco Manolo Manca[1,*], Mirko Marras[1], Piergiorgio Mura[2] and Simone Pisano[2]

[1] *Department of Mathematics and Computer Science, University of Cagliari, Via Ospedale 72, 09124 Cagliari - Italy*
[2] *Department of Humanities, University for Foreigners of Siena, Piazza Carlo Rosselli 27/28, 53100 Siena - Italy*
[3] *VisioScientiae S.r.l., Via Francesco Ciusa 46, 09131 Cagliari - Italy*

Abstract

Natural language processing (NLP) has made significant improvements in recent years, primarily driven by the latest advancements in deep learning technologies and the increasing availability of large-scale linguistic resources. Nevertheless, such advancements have mostly benefited high-resource languages, leaving many minority and underrepresented languages at the margins of computational linguistics research. Sardinian, the native language of the island of Sardinia, exemplifies this disparity. Indeed, despite its cultural and linguistic value, there is a lack of proper resources, annotated corpora, and NLP tools. This work proposes a Part-of-Speech tagging system for Sardinian characterized by methods consistent with its morphological specificity. The system integrates a BERT-based token classifier capable of assigning a grammatical category to each input word in a sentence. The classifier was trained on a balanced, manually-annotated corpus, and its performance was evaluated using standard machine-learning-oriented performance metrics (Accuracy, F1-score, Recall, and Precision). Experiments show that pre-trained architectures such as BERT remain effective even for languages with limited data availability.

Keywords

Low-resource languages, Part-of-speech tagging, Language models.

1. Introduction

Recent scientific advances in language models (LMs) and natural language processing (NLP) have contributed to the development of sophisticated technologies for generating, analyzing, and interpreting the world's major languages. In such a context, large language models (LLMs), such as GPT-4 [1], Llama-3 [2], and Phi-4 [3], have shown strong proficiency across a wide range of language-related tasks [4], including tasks such as sentiment analysis [5, 6], text classification [7, 8], text summarization, and part-of-speech (PoS) tagging [9].

However, despite their increasing effectiveness, LLMs still present limitations in performing several NLP tasks [10]. In particular, they struggle when the task concerns minority and/or low-resource languages, which often exhibit distinctive linguistic features that make them a subject of special interest for linguists. However, linguists rarely have access to automated tools and resources that facilitate in-depth studies, as these minority and/or low-resource languages are often underrepresented in the digital domain and thus inadequately, or even entirely, unknown to most models. Indeed, in this scenario, tools that support linguistic analysis, such as PoS taggers, remain scarce or nonexistent, limiting the ability of linguists to study the features of such tools at scale. More specifically, PoS tagging aims to assign a grammatical label to every word in a sentence to facilitate the study of its grammatical structure. This task is crucial for analyzing the multifaceted nature of a given language.

Sardinian, a Romance language spoken primarily on the island of Sardinia (Italy), stands out as a notable case study of low-resource language. Indeed, its rich morphological structure and its classification as an endangered language have attracted increasing attention in linguistic preservation and digital humanities [11]. In this direction, the present work describes the creation and the evaluation of an automatic Sardinian PoS tagging model. The methodology relies on fine-tuning a BERT-based language model [12] using a corpus manually annotated by linguists specializing in Sardinian. The experimental phase includes the analysis of the hyperparameters and the monitoring of machine-learning-oriented performance metrics. The proposed approach provides a foundational methodology that can be adapted to develop similar tools for other low-resource languages.

The remainder of this paper is structured as follows: Section 2 describes the state of the art; Section 3 provides a mathematical formulation of the problem and a description of the proposed approach; Section 4 illustrates the results; and finally, Section 5 concludes the work.

CLiC-it 2025: Eleventh Italian Conference on Computational Linguistics, September 24 — 26, 2025, Cagliari, Italy

*Corresponding author.

salvatorem.carta@unica.it (S. M. Carta); filippo.concas2@unica.it (F. Concas); fenu@unica.it (G. Fenu); alessandro.giuliani@unica.it (A. Giuliani); marcom.manca@unica.it (M. M. Manca); mirko.marras@unica.it (M. Marras); piergiorgio.mura@unistrasi.it (P. Mura); simone.pisano@unistrasi.it (S. Pisano)

2. Related Work

This section provides an overview of the state of the art in PoS tagging for low-resource languages, followed by a description of the work carried out for the Sardinian language in the context of NLP. The PoS tagger is an NLP tool that assigns a grammatical label to each word in a sentence, thus enabling the identification of the function of each word in that sentence. This tool facilitates syntactic analysis and provides fundamental support for developing any low-resource language, including Sardinian, by automating linguistic analysis in contexts where structured linguistic resources are lacking.

In recent years, numerous approaches have been extensively investigated, with the aim of developing automatic tagging systems or augmenting training corpora to enable high-accuracy, high-efficiency grammatical annotation at the sentence level. In the context of low-resource languages, where typically scarce data is publicly available, data from more widely known languages similar to the target language is usually employed; one approach following this direction involves the use of Hidden Markov Models (HMMs), in which the PoS tagging task is modelled as a sequence-to-sequence problem [13, 14]. HMMs are first trained on a language with large amounts of annotated data, followed by a model that transfers the learned information to the target language of interest. Different approaches that fill the gap in labeled data are based on adopting unsupervised learning techniques to group words within sentences, annotate them, and then assign a label [15, 16]. Moreover, the problem of PoS tagging is sometimes interpreted as a classification problem. For example, several works proposed to first train fully-connected neural networks (FNNs) and long short-term memory (LSTM) models on annotations projected into English and, subsequently, adapt them to the tags of the target low-resource language [17, 18, 19].

The aforementioned works build upon resources from other languages to create the PoS taggers; alternative methods focus on optimizing the limited availability of data for the target language to achieve equally good results. An example is provided by a model that utilizes translations of parts of the Bible to train PoS taggers by aggregating tags from multiple annotated languages and spreading them through word alignment within the text [20]. Furthermore, different deep learning models have been evaluated to build a PoS tagger for the Albanian language [21], which is a low-resource language as well.

To the best of our knowledge, no prior studies describe a PoS tagger for the Sardinian language. Recent work has introduced a linguistic resource designed to identify semantic relationships between Sardinian words through manual mapping of existing WordNet entries to Sardinian word meanings [22]. However, this resource does not include any tools for automatic linguistic annotation.

3. Methodology

This section describes the methodology followed to build and evaluate the PoS tagger for the Sardinian language. The section is organized as follows: first, the problem is formulated mathematically; subsequently, an overview of the entire methodology is provided; then, an analysis of the data used to build the PoS tagger is conducted; finally, the fine-tuning technique employed is presented.

3.1. Problem Formulation

Mathematically, let $\mathbf{s} \in \mathcal{S}$ be a sentence belonging to a set of sentences; then $\mathbf{s}$ can be identified as a vector whose entries represent the words included in the sentence $\mathbf{s} = [w_1, \dots, w_m]$, with $m \in \mathbb{N}^+$. Therefore, a PoS tagger can be defined as a function f expressed as:

$$\begin{aligned} f : \mathcal{S} &\longrightarrow \mathcal{T} \\ \mathbf{s} &\longmapsto f(\mathbf{s}) = \mathbf{t} = [t_1, \dots, t_m] \end{aligned}$$

where $t_j \in \mathcal{U}$ identifies the tag, i.e., a grammatical label, of the j-th word and is chosen from a specific tagset $\mathcal{U}$, and $\mathcal{T}$ is the set of vectors whose entries contain the tag of each word in a sentence.

In this work, from an application point of view, the problem of estimating the function f defined above is interpreted as a classification problem, and therefore, it is solved by training a specific classifier. Given a dataset $\mathbf{D} = \{\mathbf{s}, \mathbf{t} | \mathbf{s} \in \mathcal{S}, \mathbf{t} \in \mathcal{T}\}$ that includes sentences and their respective tags, the objective is to optimize the parameters of a classifier so that it accurately assigns the correct grammatical tag to each word in a sentence.

3.2. Methodology Overview

Figure 1 illustrates the workflow followed to develop the Sardinian PoS tagger proposed in this study.

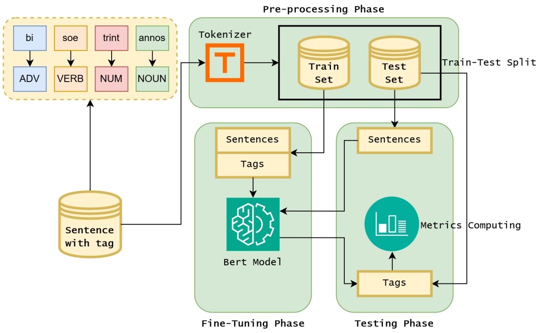


Figure 1: Workflow of the Sardinian PoS tagger.

The process consists of three main steps. In the first phase (*pre-processing*), the available tagged data is transformed and formatted adequately for use in the subsequent steps. Once transformed, the data is split into two parts: one part is used for training the model, and the other for evaluating it. In the second step (*fine-tuning*), the model learns to accurately assign grammatical tags to each word in a sentence based on the training data. Finally, in the third step (*testing*), the fine-tuned model automatically annotates the test data, and standard machine learning metrics are computed to evaluate how well it has learned to assign tags to each word.

Let us note that, as mentioned in the previous section, tags must be chosen from a specific set $\mathcal{U}$. In this work, two different state-of-the-art tag sets will be considered, i.e., the *Universal Tags* [23] (denoted as `tag`), and the tagset, conceived for the Italian language, adopted in the work of Palmero Aprosio & Moretti [24] (denoted as `fineTag`). The latter tagset is compliant with the *EAGLES* standards [25] and also more fine-grained than the former. Consequently, the pipeline depicted in Figure 1 is executed for each tagset separately.

3.3. Data Pre-Processing

In the context of minority languages, particularly the Sardinian language, it is challenging to find or utilize data that enables the training of specific models. In our scenario, to the best of our knowledge, the only available dataset for the Sardinian language that allows us to address a PoS tagging task is proposed by Mura et al. [26]. The dataset consists of $1,472$ sentences in which each word is annotated with both tag sets described in the previous section. The sentences were extracted from transcripts of interviews conducted with 21 native Sardinian emigrants, each speaking a different variety of Sardinian, as part of the *Mannigos* project [27].

Figure 2 illustrates the distribution of the number of words per sentence in the dataset. It is worth pointing out that the term *word* in this context refers to any part of the sentence, including punctuation. It can be observed that most sentences contain a limited number of words, with a significant portion not exceeding 100 words. Another key aspect is the distribution of tags within the dataset. Ensuring a balanced representation of grammatical categories allows the model to effectively learn each tag from the two defined tag sets. Figure 3 illustrates this distribution and highlights the overall balance level. Even though the dataset appears to be heavily imbalanced due to the natural linguistic structures that are common in any language, it is noteworthy that *all* tag labels in the considered tag sets are represented in the dataset.

The development of the PoS tagger in this work is based on fine-tuning the BERT language model. This choice requires a careful data pre-processing phase,

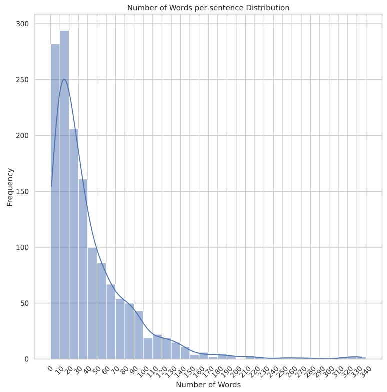


Figure 2: Distribution of the number of words per sentence.

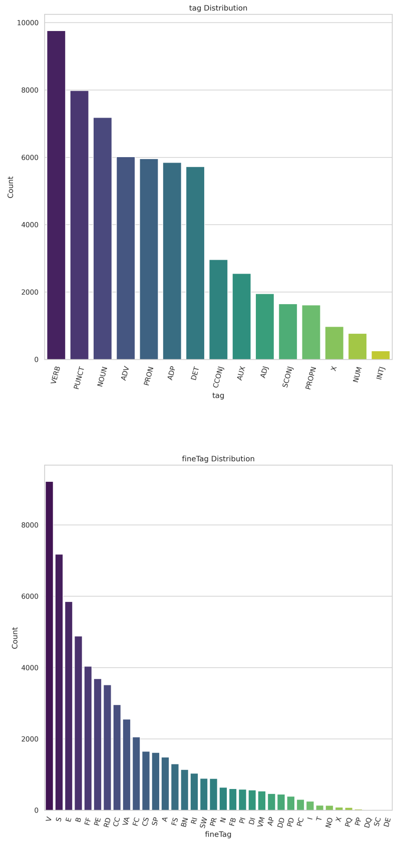


Figure 3: Distribution of labels from the `tag` (above) and the `fineTag` (below) sets in the dataset used in this study.

Table 1
Size of train and test sets for the two considered tagsets.

Tagset	Train set size	Test set size
`tag` [23]	1,172	293
`fineTag` [24]	1,177	295

where the text is appropriately tokenized (i.e., divided into smaller units called tokens, which may consist of words, sub-words, or characters). For consistency, this process was performed using the BERT tokenizer, which employs the *WordPiece* technique. This latter breaks unknown words into more common sub-word units, ensuring that each token aligns with an entry in the BERT vocabulary. Finally, each token is transformed into a numerical identifier that BERT can process. To streamline processing, each sentence was standardized to a length of 512 tokens by appending padding tokens as needed.

Following tokenization, the dataset was divided into two train sets and two test sets, one for each tagset, selecting 80% of the sentences for the first set and 20% for the second. These pre-processing steps, along with the removal of sentences containing missing or incorrect tags, led to the data splits described in Table 1.

3.4. Model Fine-Tuning

The next step is to choose the appropriate model for the fine-tuning phase. As a result of extensive, preliminary empirical evaluations, the pre-trained BERT model in its *large-cased* version was selected [12]. In more detail, BERT is a deep learning model based on the Transformer architecture developed by Google. Its special feature is its ability to process context bidirectionally, i.e., by simultaneously considering both the context to the left and the right of a word, significantly improving performance in the context of this work. It should be noted that, in this study, BERT was implemented for token classification, and the same architecture is used for both tagsets. For token classification, BERT follows this structure:

- *Input Embedding*: Each token is transformed into a vector representation that combines *token embeddings*, i.e., the token representation, *segment embeddings*, i.e., the sentence the token belongs to, and *positional embeddings*, the position of the token in the sentence.
- *Transformer Layers*: The network comprises 24 layers of this type, each using multi-head attention mechanisms to model the relationships between tokens.
- *Output Layer*: BERT returns a probability distribution over all possible classes for each token. The final output is a sequence of logits, with one prediction for each token.
- *Token Alignment*: Since some tokens are split into sub-tokens, it is necessary to realign the predictions to assign a single label to the original word.

Even though the BERT model is multilingual, it does not recognize minority languages like Sardinian. However, the pre-trained BERT model has learned the morphosyntactic behaviors of languages similar to Sardinian, such as Italian or Spanish. Consequently, a fine-tuning phase in which the BERT model identifies the primary characteristics of the Sardinian language can lead to a high-performance PoS tagger for the Sardinian language.

Given that the PoS tagging problem can be interpreted as a classification problem, the tuning phase of a token classification model can be interpreted as a supervised training phase, in which the model sees which tags are assigned to each part of speech. In this phase, it is therefore essential to choose the appropriate loss function to minimize during the tuning phase and the hyperparameters to be input to the trainer to allow optimal learning. As for the former, the Cross-Entropy Loss function was chosen, which, with the padding approach, takes the form:

$$\mathcal{L} = -\frac{1}{\sum_{i=1}^{N} m_i} \sum_{i=1}^{N} m_i \cdot \log(p_{i,y_i}) \tag{1}$$

where:

- N is the total number of tokens;
- $m_i \in \{0, 1\}$ is the mask that is 1 if the token i is valid (not padding), 0 otherwise;
- $y_i \in \mathcal{U}$ is the true class of the token i;
- p_{i,y_i} is the probability the model predicts for the correct class y_i.

Note that the same loss function was used for models trained on both the `tag` and `fineTag` sets.

Figure 4 shows the evolution of training loss, validation loss, and validation F1 score over epochs for both models during the tuning phase. These graphs were instrumental in determining the optimal number of fine-tuning epochs and in choosing other hyperparameters. Although all three metrics were considered, particular attention was paid to the validation F1 score, as it most directly reflects the model's ability to generalize on the classification task.

Table 2
Hyperparameters used for fine-tuning the BERT model.

Hyperparameter	`tag` set	`fineTag` set
Batch Size	16	16
Epochs	30	40
Weight Decay	0.01	0.01
Learning Rate	2e-5	2e-5
Optimizer	*AdamW*	*AdamW*

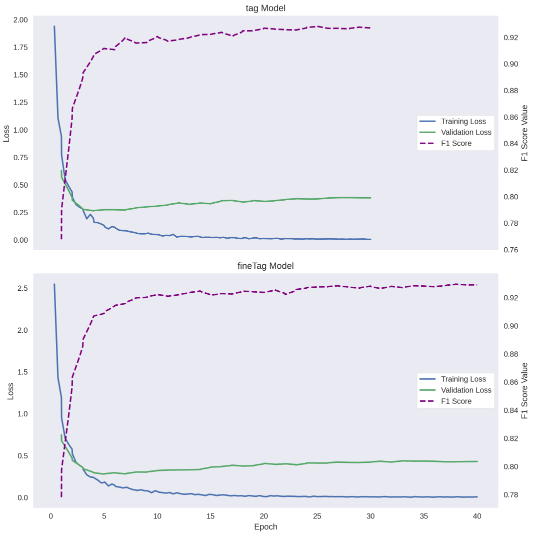


Figure 4: Train loss, validation loss, and validation F1 score over epochs tracked during fine-tuning.

All experiments were conducted on an NVIDIA RTX A6000 GPU with 48GB of vRAM.

3.5. Model Testing

Several metrics were used to evaluate model performance. Given the classification nature, the four performance metrics used in this study are Accuracy, Recall, Precision, and F1 score. Note that the last three metrics mentioned were calculated in their macro version, considering the presence of more than two classes to be evaluated.

These metrics allow us to assess how accurately the PoS tagging models classify the various words in the sentence. In particular, they allow us to analyze both the model's ability to identify all relevant classes (Recall) and its accuracy in avoiding false assignments (Precision), providing an overall measure of the balance between these two properties (F1 score). The following formulas define the metrics in detail.

$$\text{Accuracy} = \frac{\sum_{i=1}^{N} m_i \cdot \mathbb{1}(y_i = \hat{y}_i)}{\sum_{i=1}^{N} m_i}$$

$$\text{Precision}_{\text{macro}} = \frac{1}{C} \sum_{c=0}^{C-1} \frac{TP_c}{TP_c + FP_c}$$

$$\text{Recall}_{\text{macro}} = \frac{1}{C} \sum_{c=0}^{C-1} \frac{TP_c}{TP_c + FN_c}$$

$$\text{F1}_{\text{macro}} = \frac{1}{C} \sum_{c=0}^{C-1} \frac{2 \cdot \text{Precision}_c \cdot \text{Recall}_c}{\text{Precision}_c + \text{Recall}_c}$$

Table 3
Evaluation of the fine-tuned models for the two tagsets.

Metric	`tag` set	`fineTag` set
Accuracy	0.9418	0.9362
Precision$_{\text{Macro}}$	0.9250	0.9274
Recall$_{\text{Macro}}$	0.9347	0.9308
F1$_{\text{Macro}}$	0.9298	0.9291

with:

- $TP_c = \sum_{i=1}^{N} m_i \cdot \mathbb{1}(y_i = c) \cdot \mathbb{1}(\hat{y}_i = c)$;
- $FP_c = \sum_{i=1}^{N} m_i \cdot \mathbb{1}(y_i \neq c) \cdot \mathbb{1}(\hat{y}_i = c)$;
- $FN_c = \sum_{i=1}^{N} m_i \cdot \mathbb{1}(y_i = c) \cdot \mathbb{1}(\hat{y}_i \neq c)$.

in which N, m_i, and y_i are the same as defined in Formula 1; while $\hat{y}_i$ is the tag predicted by the model, C is the size of the set $\mathcal{U}$ (i.e. the number of all possible tags), and $\mathbb{1}(A)$ is the indicator function, equal to 1 if condition A is true, 0 otherwise.

It is important to note that all metrics introduced vary within a range between 0 and 1, with values closer to 1 indicating better performance.

4. Experimental Results

This section is organized into two main parts. In the first part, we present the quantitative analysis of the models, reporting and comparing their performance on the test sets. These results allow us to evaluate the overall effectiveness of each model in a rigorous and reproducible manner. The second part is dedicated to a brief qualitative analysis, in which we examine selected examples unobserved during the fine-tuning and testing phases. This analysis aims to illustrate the models' predictions in practice, thus complementing the information obtained from the quantitative evaluation.

4.1. Quantitative Analysis

Table 3 shows the performance of the two fine-tuned BERT-based models on the test sets[1]. The first model, fine-tuned on the coarser-granularity tagset (`tag`), achieves an accuracy of 0.9418 and a macro F1 score of 0.9298, with recall and precision scores of 0.9347 and 0.9250, respectively. The second model, fine-tuned on the more detailed tagset (`fineTag`), produces slightly lower but still good results, with an accuracy of 0.9362, a macro F1 of 0.9291,

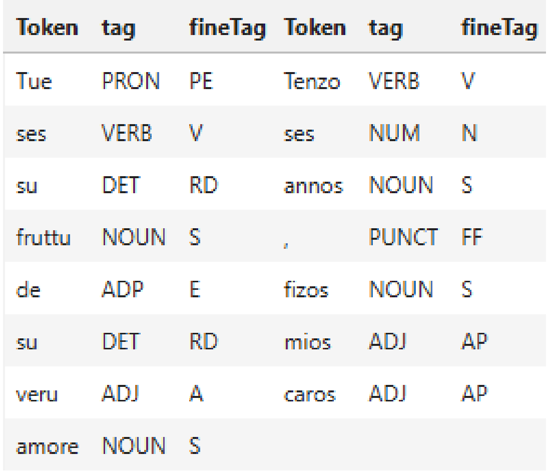
[1] While per-tag evaluation metrics could in principle offer additional insights, given also the large size of the tagset, we chose to focus on overall metrics to maintain a clear and coherent narrative aligned with the primary research questions. We consider a detailed per-tag analysis an important direction for future work, particularly in application-specific settings where tag-level behavior is critical.

a recall of 0.9308, and a precision of 0.9274. These results indicate that both models generalize to the test data well. It is important to note that high performance is still achieved even in the `fineTag` setting, which involves a classification task with 36 PoS classes (the `tag` set included 15 PoS classes). This observation highlights the robustness of the fine-tuned models, demonstrating their ability to handle more complex and fine-grained label distributions without substantial performance loss. Notably, these results are achieved despite the linguistic variability within the dataset, which includes multiple Sardinian language varieties with differing morphological features. Nevertheless, the models successfully capture the core structural patterns of each variety, demonstrating strong generalization across intra-language variation.

4.2. Qualitative Analysis

In addition to quantitative evaluation, we conducted a qualitative analysis to gain a deeper understanding of the behavior of the two fine-tuned PoS taggers in real-world scenarios. This section presents two illustrative examples, shown in Figure 5, where we compare the results of the two PoS taggers on sentences that are not part of the dataset but are completely external. These examples allow us to examine how the models handle both simple and ambiguous linguistic structures and to evaluate their ability to assign the correct PoS tags in context. The model outputs are easily readable and clearly aligned with each token in the sentence, making it straightforward to assess the tagging quality and spot possible inconsistencies visually.

One notable aspect is the accurate tagging of simpler elements, such as punctuation marks, which both models consistently identify. More interestingly, the models also demonstrate a strong ability to disambiguate words based on context. For instance, both sentences in Figure 5 include the word *ses*, which in Sardinian can function either as a numeral (meaning "six") or as a verb (a form of the verb to be). Despite the ambiguity, both models correctly assign different PoS tags to *ses* depending on their usage in each sentence, showing that they have learned to exploit contextual cues to resolve such lexical ambiguity. This suggests that the models are not merely memorizing patterns, but rather capturing meaningful linguistic distinctions across a morphologically rich and internally diverse language.



Token	tag	fineTag	Token	tag	fineTag
Tue	PRON	PE	Tenzo	VERB	V
ses	VERB	V	ses	NUM	N
su	DET	RD	annos	NOUN	S
fruttu	NOUN	S	,	PUNCT	FF
de	ADP	E	fizos	NOUN	S
su	DET	RD	mios	ADJ	AP
veru	ADJ	A	caros	ADJ	AP
amore	NOUN	S			

Figure 5: Example output of the two PoS taggers on external sentences. Each row corresponds to a token, with associated tags produced by the two models: the *tag* column shows the output of the model fine-tuned on the *Universal Tagset* tagset, while the *fineTag* column displays the output of the model fine-tuned on the more fine-grained tagset of [24].

5. Conclusions

This work has introduced an automatic PoS tagging model for Sardinian, a minority and morphologically complex language, using a BERT fine-tuning approach. Starting from a heterogeneous, manually annotated corpus composed of sentences taken from interviews with native speakers of different varieties of Sardinian, we implemented and tested two distinct models, each trained on a different set of grammatical classes, from [23] and [24] respectively. The results obtained, both in terms of quantitative and qualitative analyses, highlighted the effectiveness and robustness of the proposed model, which is capable of generalizing even in the presence of language variability. This contribution is part of a broader effort to promote and preserve low-resource languages, offering methodologies that can be replicated and extended to other similar linguistic contexts [28].

Looking ahead, this work can be extended in several directions. On the one hand, it will be possible to further refine the models by expanding the corpus, integrating parallel resources (e.g., annotated translations), or using semi-supervised learning techniques or transfer learning from other languages. A particularly relevant aspect that will need to be addressed concerns the transparency, understandability, and interpretability of the models, an issue not explored in this study but increasingly important across many application domains where intelligent methods support human activity, such as medicine [29, 30], finance [31, 32, 33], safety [34], data analysis [35], and industry [36, 37, 38], to name a few. Furthermore, it will be essential to annotate the dataset with explicit information about Sardinian varieties and assess the PoS taggers' performance across these varieties, in order to better understand their generalization capabilities and potential linguistic biases. Finally, an important development could be the creation of an accessible user interface that would make the PoS tagger usable by linguists, scholars, and citizens not experts in computer science. Such a tool could be integrated into digital platforms for teaching, documentation, and linguistic research on Sardinian, contributing to greater digitization and visibility of the language.

Acknowledgments

We acknowledge financial support under the National Recovery and Resilience Plan (NRRP), Mission 4 Component 2 Investment 1.5 - Call for tender No.3277 published on December 30, 2021 by the Italian Ministry of University and Research (MUR) funded by the European Union – NextGenerationEU. Project Code ECS0000038 – Project Title eINS Ecosystem of Innovation for Next Generation Sardinia – CUP F53C22000430001- Grant Assignment Decree No. 1056 adopted on June 23, 2022 by the Italian Ministry of University and Research (MUR).

References

[1] J. Achiam, S. Adler, S. Agarwal, L. Ahmad, I. Akkaya, F. L. Aleman, D. Almeida, J. Altenschmidt, S. Altman, S. Anadkat, et al., Gpt-4 technical report, arXiv preprint arXiv:2303.08774 (2023).

[2] A. Grattafiori, A. Dubey, A. Jauhri, A. Pandey, A. Kadian, A. Al-Dahle, A. Letman, A. Mathur, A. Schelten, A. Vaughan, et al., The llama 3 herd of models, arXiv preprint arXiv:2407.21783 (2024).

[3] M. Abdin, J. Aneja, H. Behl, S. Bubeck, R. Eldan, S. Gunasekar, M. Harrison, R. J. Hewett, M. Javaheripi, P. Kauffmann, et al., Phi-4 technical report, arXiv preprint arXiv:2412.08905 (2024).

[4] Y. Chang, X. Wang, J. Wang, Y. Wu, L. Yang, K. Zhu, H. Chen, X. Yi, C. Wang, Y. Wang, et al., A survey on evaluation of large language models, ACM transactions on intelligent systems and technology 15 (2024) 1–45.

[5] D. Dessì, G. Fenu, M. Marras, D. Reforgiato Recupero, Leveraging cognitive computing for multiclass classification of e-learning videos, in: European Semantic Web Conference, Springer, 2017, pp. 21–25.

[6] D. Dessí, M. Dragoni, G. Fenu, M. Marras, D. Reforgiato Recupero, Deep learning adaptation with word embeddings for sentiment analysis on online course reviews, in: Deep learning-based approaches for sentiment analysis, Springer, 2020, pp. 57–83.

[7] S. Carta, A. Giuliani, M. M. Manca, L. Piano, L. Pompianu, S. G. Tiddia, Towards knowledge graph refinement: Misdirected triple identification, in: Adjunct Proceedings of the 32nd ACM Conference on User Modeling, Adaptation and Personalization, 2024, pp. 460–466.

[8] A. Pisu, L. Pompianu, A. Salatino, F. Osborne, D. Riboni, E. Motta, D. Reforgiato Recupero, et al., Leveraging language models for generating ontologies of research topics, in: CEUR WORKSHOP PROCEEDINGS, volume 3747, CEUR-WS, 2024, p. 11.

[9] A. Benlahbib, A. Boumhidi, A. Fahfouh, H. Alami, Comparative analysis of traditional and modern nlp techniques on the cola dataset: From pos tagging to large language models, IEEE Open Journal of the Computer Society (2025).

[10] S. Jadhav, A. Shanbhag, A. Thakurdesai, R. Sinare, R. Joshi, On limitations of llm as annotator for low resource languages, arXiv preprint arXiv:2411.17637 (2024).

[11] G. Mensching, The internet as a rescue tool of endangered languages: Sardinian, in: Proceeding Conference Multilinguae: multimedia and minority languages. San Sebastian: The Association of Electronics and Information Technology Industries, 2000.

[12] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, in: Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers), 2019, pp. 4171–4186.

[13] J. Buys, J. A. Botha, Cross-lingual morphological tagging for low-resource languages, in: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), 2016, pp. 1954–1964.

[14] R. Cardenas, Y. Lin, H. Ji, J. May, A grounded unsupervised universal part-of-speech tagger for low-resource languages, in: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), 2019, pp. 2428–2439.

[15] C. Christodoulopoulos, S. Goldwater, M. Steedman, Two decades of unsupervised pos induction: How far have we come?, in: Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing, 2010, pp. 575–584.

[16] P. F. Brown, V. J. Della Pietra, P. V. Desouza, J. C. Lai, R. L. Mercer, Class-based n-gram models of natural language, Computational linguistics 18 (1992) 467–480.

[17] L. Duong, T. Cohn, K. Verspoor, S. Bird, P. Cook, What can we get from 1000 tokens? a case study of multilingual pos tagging for resource-poor languages, in: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), 2014, pp. 886–897.

[18] M. Fang, T. Cohn, Learning when to trust distant supervision: An application to low-resource pos tagging using cross-lingual projection, arXiv preprint arXiv:1607.01133 (2016).

[19] M. Fang, T. Cohn, Model transfer for tagging low-resource languages using a bilingual dictionary, arXiv preprint arXiv:1705.00424 (2017).

[20] Ž. Agić, D. Hovy, A. Søgaard, If all you have is a bit of the Bible: Learning POS taggers for truly low-resource languages, in: C. Zong, M. Strube (Eds.), Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers), Association for Computational Linguistics, Beijing, China, 2015, pp. 268–272.

[21] E. Fetahi, M. Hamiti, A. Susuri, B. Selimi, D. I. Saiti, Neural network and transformer-based pos tagger for low resource languages, in: 2024 International Conference on Information Technologies (InfoTech), 2024, pp. 1–4.

[22] M. Angioni, F. Tuveri, M. Virdis, L. L. Lai, M. E. Maltesi, Sardanet: A linguistic resource for sardinian language, in: Proceedings of the 9th Global Wordnet Conference, 2018, pp. 412–419.

[23] Universal pos tags, 2014-2024. URL: https://universaldependencies.org/u/pos/.

[24] A. Palmero Aprosio, G. Moretti, Tint 2.0: an all-inclusive suite for nlp in italian, in: Proceedings of the Fifth Italian Conference on Computational Linguistics (CLiC-it 2018), 2018.

[25] Eagles part-of-speech (pos) tag set, 2014-2024. URL: https://www.ilc.cnr.it/EAGLES96/home.html.

[26] P. Mura, S. Pisano, S. Carta, A. Giuliani, M. Manca, The corpus of Sardinian emigrants:a tool for a quantitative approach to contact phenomena, MiLES: Minority Languages in European Societies - International Conference-Turin / Bard - BOOK OF ABSTRACTS, July 3-6, 2024.

[27] S. Pisano, V. Piunno, V. Ganfi, Appunti per un corpus di sardo multimediale, in: M. V. D. Marzo, S. Pisano (Ed.), Per una pianificazione del plurilinguismo in Sardegna, Condaghes, 2022, pp. 147–164.

[28] S. M. Carta, S. Chessa, G. Contu, A. Corriga, A. Deidda, G. Fenu, L. Frigau, A. Giuliani, L. Grassi, M. M. Manca, et al., Limba: An open-source framework for the preservation and valorization of low-resource languages using generative models, arXiv preprint arXiv:2411.13453 (2024).

[29] A. S. Podda, R. Balia, M. M. Manca, J. Martellucci, L. Pompianu, A deep learning strategy for the 3d segmentation of colorectal tumors from ultrasound imaging, Image and Vision Computing (2025) 105668.

[30] R. Saia, S. Carta, G. Fenu, L. Pompianu, Influencing brain waves by evoked potentials as biometric approach: taking stock of the last six years of research, Neural Computing and Applications 35 (2023) 11625–11651.

[31] A. Giuliani, R. Savona, S. Carta, G. Addari, A. S. Podda, Corporate risk stratification through an interpretable autoencoder-based model, Computers & Operations Research 174 (2025) 106884.

[32] M. Nallakaruppan, B. Balusamy, M. L. Shri, V. Malathi, S. Bhattacharyya, An explainable ai framework for credit evaluation and analysis, Applied Soft Computing 153 (2024) 111307.

[33] S. Carta, A. S. Podda, D. Reforgiato Recupero, M. M. Stanciu, Explainable ai for financial forecasting, in: International Conference on Machine Learning, Optimization, and Data Science, Springer, 2021, pp. 51–69.

[34] A. Pisu, N. Elia, L. Pompianu, F. Barchi, A. Acquaviva, S. Carta, Enhancing workplace safety: A flexible approach for personal protective equipment monitoring, Expert Systems with Applications 238 (2024) 122285.

[35] G. Armano, A. Giuliani, A two-tiered 2d visual tool for assessing classifier performance, Information Sciences 463-464 (2018) 323–343.

[36] A. S. Podda, R. Balia, L. Pompianu, S. Carta, G. Fenu, R. Saia, Cargram: Cnn-based accident recognition from road sounds through intensity-projected spectrogram analysis, Digital Signal Processing 147 (2024) 104431.

[37] A. Mana, A. Allouhi, A. Hamrani, S. Rehman, I. El Jamaoui, K. Jayachandran, Sustainable ai-based production agriculture: Exploring ai applications and implications in agricultural practices, Smart Agricultural Technology 7 (2024) 100416.

[38] Y. Rong, Z. Xu, J. Liu, H. Liu, J. Ding, X. Liu, W. Luo, C. Zhang, J. Gao, Du-bus: a realtime bus waiting time estimation system based on multi-source data, IEEE Transactions on Intelligent Transportation Systems 23 (2022) 24524–24539.

Declaration on Generative AI

During the preparation of this work, the author(s) used ChatGPT (OpenAI), Grammarly, and DeepL Write / DeepL Translate in order to: Text translation, Grammar and spelling check, and Formatting assistance. After using these tool(s)/service(s), the author(s) reviewed and edited the content as needed and take(s) full responsibility for the publication’s content.