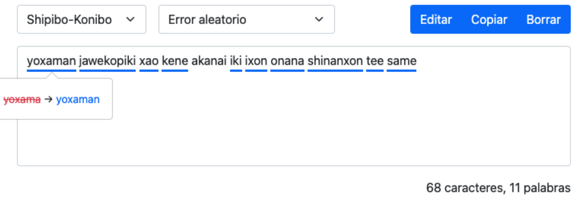
SchAman: Spell-Checking Resources and Benchmark for Endangered Languages from Amazonia

Arturo Oncevay[ϵ,χ] **Gerardo Cardoso**[ρ,α] **Carlo Alva**[ρ,α] **César Lara Ávila**[ρ,α,μ]
Jovita Vásquez Balarezo[η] **Saúl Escobar Rodríguez**[η] **Delio Siticonatzi Camaiteri**[η]
Esaú Zumaeta Rojas[η] **Didier López Francis**[η] **Juan López Bautista**[η] **Nimia Acho Rios**[η]
Remigio Zapata Cesareo[η] **Héctor Erasmo Gómez Montoya**[ρ,α] **Roberto Zariquiey**[ρ,χ]
[ϵ]University of Edinburgh, Scotland [μ]Universidad Nacional de Ingeniería, Peru
[ρ]Pontificia Universidad Católica del Perú ([α]IA-PUCP | [χ]Chana Field Station), Peru
[η]Universidad Católica Sedes Sapientiae – NOPOKI, Peru
a.oncevay@ed.ac.uk,rzariquiey@pucp.edu.pe

Abstract

Spell-checkers are core applications in language learning and normalisation, which may enormously contribute to language revitalisation and language teaching in the context of indigenous communities. Spell-checking as a generation task, however, requires large amount of data, which is not feasible for endangered languages, such as the languages spoken in Peruvian Amazonia. We propose here augmentation methods for various misspelling types as a strategy to train neural spell-checking models and we create an evaluation resource for four indigenous languages of Peru: Shipibo-Konibo, Asháninka, Yánesha, Yine. We focus on special errors that are significant for learning these languages, such as phoneme-to-grapheme ambiguity, grammatical errors (gender, tense, number, among others), accentuation, punctuation and normalisation in contexts where two or more writing traditions co-exist. We found that an ensemble model, trained with augmented data from various types of error achieves overall better scores in most of the error types and languages. Finally, we released our spell-checkers as a web service to be used by indigenous communities and organisations to develop future language materials[1].



yoxaman jawekopiki xao kene akanai iki ixon onana shinanxon tee same

~~yoxama~~ → yoxaman

68 caracteres, 11 palabras

Figure 1: SchAman as a web service.

1 Introduction

In Natural Language Processing (NLP), the normalisation of a language is closely related to automatic spell checking, a process in which a computer program identifies a misspelling and suggests correct or standardised alternatives to the user. Spell-checking, an important step towards grammar checking, can be addressed as a sequence-to-sequence problem with deep neural networks (Junczys-Dowmunt et al., 2018). A common problem with this approach, however, is the large amount of data required. One possible way to deal with this is the generation of synthetic data (Etoori et al., 2018; White and Rozovskaya, 2020), since many of these errors are random, or typographical errors due to close keys.

For low-resource and endangered languages, developing a speller or normalisation tool is an important step for supporting further language revitalisation and documentation efforts, as well as indigenous education programs. This is particularly important in regions like Amazonia, where linguistic diversity is in serious risk (Zariquiey et al., 2019). Although there are rule-based spell-checkers for some languages spoken in Peruvian Amazonia, such as Shipibo-Konibo (Alva and Oncevay, 2017) and pan-Ashaninka (Ortega et al., 2020), their vocabulary coverage is limited and they are not context-sensitive. These are issues that can be assessed by subword and neural-based generation models for sequences of words.

In this study, we propose the implementation of neural spell-checkers for four indigenous languages spoken in Peruvian Amazonia: one Pano language, Shipibo-Konibo (shp), and three Arawak languages, Ashaninka (cni), Yanesha (ame) and Yine (pib). For this purpose, we introduce error augmentation methods to take advantage of the

[1] Data and code are available in https://github.com/iapucp/SchAman, and the code for the web interface and service is in https://github.com/iapucp/SchAman-demo

Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing (Volume 2: Short Papers), pages 411–417
November 20–23, 2022. ©2022 Association for Computational Linguistics

scarce monolingual corpus available (§4), and we create an evaluation resource with a diverse typology of errors: phoneme-to-grapheme ambiguity, grammatical errors (gender, tense, number), accentuation, punctuation and normalisation (§5). We present an initial neural benchmark with a model trained with different types of augmented data (§6), and finally, we release our spell-checkers as a web service (§7), which is ready to deploy and use.

2 Related work

Ghosh and Kristensson (2017) proposed the first deep learning model for spelling and completing text in keyboard decoding for English as a sequence to sequence task. This inspires further work such as in Sakaguchi et al. (2017). They presented a word recognition model based on a semi-character level recurrent neural network, which is inspired in the robust word recognition mechanism known as the "Cmabrigde Uinervtisy" effect. Regarding augmentation methods for spelling, Etoori et al. (2018) assessed a low-resource spell-checking case for Indic languages, where they generated synthetic data with random noise and linguistic information. Also, Li et al. (2018) used a nested recurred neural network to detect spelling errors for English, and augmented the dataset with misspelling words with similar pronunciation. Likewise, grammar-checking is addressed as a sequence to sequence task by Junczys-Dowmunt et al. (2018) and Choe et al. (2019). The latter generated erroneous versions of large corpus without annotations using a real noise function, which are feed to a large model and then fine-tuned (domain and style adaptation).

Finally, for the languages spoken in Amazonia, there are only spell-checkers for Shipibo-Konibo (Alva and Oncevay, 2017) and Ashaninka (Ortega et al., 2020). The former is a rule-driven approach with graphs and syllabic information, whereas the latter is a finite-state-transducer or FST. However, they work at word-level, meaning that they lack context and are at disadvantage when words are joined or split by mistake.

3 Languages and Data

The four languages in the focus of this paper are highly agglutinating and synthetic, meaning that they can compress a large amount of information in a single word composed of several bound morphemes, often with more or less clear-cut morphological boundaries. In addition, they do not have a long writing tradition, but they include more than one competing orthographic tradition, one promoted by the Summer Institute of Linguistics (SIL)[2] and another one promoted by the Ministry of Education of Peru and considered official. Official orthographies do not have more than 20 years in any case. The context opens a real world challenge for normalisation. More details are included in the Appendix.

Language	# sentences	|V|
Shipibo-Konibo (shp)	22,032	22,904
Asháninka (cni)	12,629	23,721
Yanesha (ame)	13,241	23,626
Yine (pib)	7,658	14,142

Table 1: Number of sentences and vocabulary size of the monolingual corpora used for augmentation, extracted from Bustamante et al. (2020).

Approach	Lang.	original	modified
RANDOM	shp	jaweratorin	jawerat**ro**in
PROXKEY	cni	kitaiteri	k**k**taiteri
P2GAMB	ame	sewayanon	se**hu**ayanon
SYLSIM	pib	katuyma	ka**tul**yma
DENORM	ame	phokwe'	**p̃**hokwe'

Table 2: Examples of the error augmentation approaches at word-level (a sentence is given as input).

Monolingual texts There is almost no web data available for these languages, but we make use of the monolingual corpora extracted from educational and language learning PDF material by Bustamante et al. (2020), which is already parsed and cleaned. Table 1 shows the data used, where we only considered sentences with fewer than 50 characters. The decision is pragmatic: to assess the impact of the augmentation type for spelling, and not to stress long-term dependencies in the model.

4 Error augmentation approaches

We create different augmented training sets (same size) with each type of error described as follows.

Noisy baseline (RANDOM) We generate errors at character-level with insertion, replacement and deletion operations. We also consider the whitespace into these random operations, as it is a common error for speakers with poor background of the standard writing.

[2] SIL International (https://www.sil.org/)

	General errors			Normalisation		
Lang.	# sentences	Vocabulary size		# sentences	Vocabulary size	
		w/o errors	w/ errors		w/o errors	w/ errors
Shipibo-Konibo	2,936	10,710	13,336	916	3,931	4,279
Asháninka	3,544	11,385	13,209	796	3,124	3,291
Yanesha	3,490	10,146	12,793	754	1,781	1,825
Yine	2,078	6,131	6,710	702	1,667	1,710

Table 3: Corpora size and vocabulary of General errors and Normalisation

Proximity keys (ProxKey) It is based on the keyboard layout, when a user misstypes a neighbour key. We consider the QWERTY layout of Spanish Latinamerican, which is the predominant layout used for the speakers of the target languages.

Phoneme-to-grapheme ambiguity (p2gAmb) Similar to Li et al. (2018), we consider the correspondence between graphemes and phonemes as a source for augmenting more linguistically-informed errors. The difference with English, is that the Amazonian languages have stronger correspondence of phonemes-graphemes (known as a transparent orthography (Borgwaldt et al., 2005)), given their recent writing standardisation. Nevertheless, there are still phonemes that have a very similar pronunciation, and can confuse the listener at spelling time (e.g. *w*→*hu*).

Syllable similarity (SylSim) Given the regular and transparent orthography of the languages, we focus on syllables. For instance, for Shipibo-Konibo, Alva and Oncevay (2017) used a syllable-based graph to identify a misspelled word: if you cannot split the word in syllables, there could be a misspelling or it could be a loanword. We use the syllabification method for Shipibo-Konibo and developed the rules for the other three languages. To apply the syllabile similarity error, we split a word into their syllables, and then look for a similar syllable (edit distance) to replace one or more.

De-normalisation (DeNorm) We map an old and the most recent writing standard in all languages, and develop a method to apply a denormalisation noise given a sentence.

We present examples of each augmented-error approach in Table 2. For the language-dependent methods (SylSim, p2gAmb, DeNorm), which require more specialised knowledge about the writing and speech systems, we collected the information needed in collaboration with field linguists, language grammars and standardisation norms.

	shp	**cni**	**ame**	**pib**
Phonetic	2,132	1,354	5,540	1,347
Gender	142	282	-	1
Tense	96	66	-	-
Number	51	111	9	2
Punctuation	47	43	327	-
Accentuation	39	-	238	-
Syntactic	3,622	1,272	330	3,916
Semantic	517	93	-	-

Table 4: Number of errors per type in the General errors dataset per language.

5 Evaluation corpora

With the support of language teachers, we defined an error typology of the most common mistakes of their students: phoneme ambiguity, grammar mistakes (gender, tense, number), punctuation, accentuation, syntactic, semantic and normalisation. After that, we provide an annotation protocol to create a parallel corpus of corrected written sentences aligned with misspelled ones, with an annotation of the type of errors included in each sentence (it could be more than one):

- Two teachers per language receive a word list.
- For each word, they first write a sentence that includes that word (or a similar one, e.g. inflected) without any misspelling.
- From the created sentence, they inject one or more of the errors from the defined typology, and label the error type.

We define two corpora: General errors and Normalisation. We consider that normalisation requires a differentiated corpus, given its relevance in the standardisation of their writing systems. Table 3 shows the amount of sentences and the vocabulary of the new corpora, while Table 4 shows more details about the General dataset.

	General				Normalisation			
	shp	**cni**	**ame**	**pib**	**shp**	**cni**	**ame**	**pib**
Random	85.3 (5.4)	88.5 (0.2)	75.2 (4.0)	85.6 (6.6)	88.9 (2.1)	75.7 (1.9)	64.6 (3.4)	72.6 (1.0)
ProxKey	85.8 (5.9)	89.2 (0.8)	76.5 (5.4)	85.2 (6.2)	88.4 (1.6)	74.0 (0.1)	64.8 (2.7)	73.4 (1.8)
p2gAmb	88.4 (8.5)	89.1 (0.8)	77.0 (5.9)	-	91.3 (4.5)	78.9 (5.1)	71.0 (8.9)	-
SimSyl	84.1 (4.2)	87.8 (-0.5)	75.2 (4.0)	84.8 (5.9)	87.9 (1.2)	75.4 (1.6)	63.6 (1.4)	70.9 (-0.7)
DeNorm	88.5 (8.6)	89.6 (1.3)	76.9 (5.7)	**86.4 (7.5)**	**92.4 (5.6)**	**80.4 (6.6)**	**72.3 (10.2)**	**80.4 (8.7)**
All	84.7 (4.9)	86.6 (-1.7)	74.7 (3.6)	83.9 (4.9)	88.6 (1.9)	76.8 (3.0)	68.2 (6.1)	75.7 (4.0)
Ensemble	**88.7 (8.8)**	**89.8 (1.4)**	**77.4 (6.3)**	86.2 (7.3)	91.7 (5.0)	78.0 (4.2)	67.9 (5.8)	76.6 (5.0)

Table 5: chrF (and ΔchrF) scores on the General and Normalisation test set for all languages.

	Shipibo-Konibo		**Asháninka**		**Yanesha**		**Yine**	
	DeNorm	Ensemble	DeNorm	Ensemble	DeNorm	Ensemble	DeNorm	Ensemble
Phonetic	97.3 (2.5)	97.6 (2.8)	97.4 (0.7)	97.1 (0.5)	95.8 (1.7)	96.1 (1.9)	94.8 (1.8)	94.8 (1.8)
Gender	97.7 (3.5)	97.7 (3.6)	95.8 (1.3)	95.2 (0.6)	-	-	100.0 (2.1)	100.0 (2.1)
Tense	97.5 (3.4)	97.5 (3.5)	97.0 (1.6)	96.3 (1.0)	-	-	-	-
Number	97.2 (3.3)	97.0 (3.0)	96.9 (1.2)	96.2 (0.4)	93.7 (3.6)	92.4 (2.3)	100.0 (7.6)	100.0 (7.6)
Punctuation	96.6 (2.8)	97.0 (3.2)	96.9 (0.7)	95.9 (-0.3)	89.9 (3.5)	90.4 (3.9)	-	-
Accentuation	96.7 (2.6)	96.9 (2.8)	-	-	89.7 (3.2)	90.9 (4.3)	-	-
Syntactic	97.0 (2.3)	97.2 (2.5)	97.8 (0.7)	97.6 (0.5)	90.4 (3.3)	91.3 (4.3)	96.3 (3.0)	96.3 (3.0)
Semantic	96.7 (2.5)	96.8 (2.6)	97.1 (1.2)	96.5 (0.6)	-	-	-	-

Table 6: chrF (and ΔchrF) scores for each error type in the General test set, using DeNorm and Ensemble.

6 Benchmark

Model architecture and training We use Pruthi et al. (2019)'s model for word recognition to deal with adversarial misspellings. This is a semi-character recurrent neural network based on Sakaguchi et al. (2017). The model receives as input a sentence with misspellings, and generates a corrected one. The hyper-parameters are included in the Appendix. Besides, we train the model using a single Tesla T4 GPU from Google Colab.

Evaluation metric As we are dealing with a sequence-to-sequence problem, we use chrF (Popović, 2015) as our metric. This is important to assess whether our model is modifying the input more than expected. We also include a ΔchrF value, which is the difference between the chrF score of original correct-error reference pair, minus the score obtained by the correct-output one.

Models and evaluation The goal of the benchmark is to determine which augmentation approach can generalise better to real errors annotated by the language teachers (General, Normalisation). For the experiment, we double the original corpus using each augmentation approach[3]. We also train a model using all the augmented data (All), and set up an ensemble model by majority vote. To aid the training process, we split the General set in 500-500 sentences for test and validation, and the rest as complement for training in all settings. We did not do the same for the Normalisation set, which is smaller.

6.1 Results and Discussion

Table 5 shows the results for all the models in both General and Normalisation test sets, where ΔchrF is positive in most cases, indicating that the output sentences are closer to the reference than the misspelled ones. We clearly observe that DeNorm and Ensemble models achieved the first and second best scores consistently in most scenarios. Besides, p2gAmb has a robust performance in the Normalisation dataset, despite not being trained on the same data distribution (as in DeNorm for instance). However, this is consistent with the standardisation efforts of the writing system, as they try to make the orthography more transparent (e.g. avoiding characters with similar correspondent sounds, as with *c* and *k*).

To analyse the performance per error type, we simplified the test set entries and kept only one

[3] Further experiments with 3x, 4x or more augmented data did not provide significant difference in the overall results.

error per sentence[4]. Table 6 shows the results for DENORM and Ensemble in all languages, where we observe that ΔchrF is positive in almost all settings, indicating a consistent improvement over the misspelled sentences. We also observe that Asháninka is the language that obtains the smallest improvements (measured in ΔcharF). One potential reason is the different but very close dialects that are merged in the initial monolingual corpus of Asháninka. Besides that, we do not observe a signicant advantage of the Ensemble model over DENORM in almost any type. We recall that both models are fed with part of the annotated corpus for training, indicating that DENORM is a robust approach for generalisation.

7 Web service

We implement an API and a web service that includes all the models presented in the previous section. The web interface includes the following features: (1) the user can select the language and model of preference, (2) the system highlights which words are updated, and what is the modification, (3) the user can modify the output and provide feedback. Figure 1 shows an example. Finally, we open-source our demo code in: https://github.com/iapucp/SchAman-demo.

8 How to scale up to new languages

For new languages from Amazonia, the first step is to obtain a monolingual corpus as seed text. According to the results, it is more significant to augment training data with the DENORM and P2GAMB approaches, which require a short involvement of an expert or the study of language grammars. This is less expensive than to develop an FST-based tool for spell-checking[5]. The creation of the evaluation resource is the most costly (in terms of expert hours), however, our methodology can be reproduced easily.

9 Conclusions and Future Work

We develop spell-checking resources (for training and evaluation) and define an initial benchmark for four endangered languages of the Amazonia region of Peru. Experiments showed that DENORM and Ensemble models achieve overall better results in most error types and languages, and they have a positive impact when dealing with new vocabulary.

The spell-checking models are available as an API and web service, and it was made available to language teachers and students. As future work, we plan to develop multilingual models (three of the four targeted languages are from the same language family), and to deploy a more explainable spelling application (e.g. indicating which type of error has been corrected).

Acknowledgements

We are grateful to Dr Gian Battista Bolis and Juan Rubén Ruiz Zevallos, from Universidad Católica Sedes Sapientiae, for enabling and supporting the collaboration with the bilingual teachers from NOPOKI. Moreover, we appreciate the support of other researchers from PUCP in an early stage of this research: John E. Miller, Adriano Ingunza, Candy Angulo, Gema Silva, Jaime Montoya, Kervy Rivas, Gina Bustamante and Patrick Figueroa.

Finally, the last author acknowledges the Max Planck Institute for Evolutionary Anthropology, Department of Linguistic and Cultural Evolution, for its support to the development of the Chana Field Station in the Amazonian region of Peru, and the support of CONCYTEC-ProCiencia, Peru, under the contract 183-2018-FONDECYT-BM-IADT-MU from the funding call E041-2018-01-BM.

References

Carlo Alva and Arturo Oncevay. 2017. Spell-checking based on syllabification and character-level graphs for a Peruvian agglutinative language. In *Proceedings of the First Workshop on Subword and Character Level Models in NLP*, pages 109–116, Copenhagen, Denmark. Association for Computational Linguistics.

Susanne R Borgwaldt, Frauke M Hellwig, and Annette MB De Groot. 2005. Onset entropy matters–letter-to-phoneme mappings in seven languages. *Reading and Writing*, 18(3):211–229.

Gina Bustamante, Arturo Oncevay, and Roberto Zariquiey. 2020. No data to crawl? monolingual corpus creation from PDF files of truly low-resource languages in Peru. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 2914–2923, Marseille, France. European Language Resources Association.

[4] This process makes the input and output sentence very similar, resulting in higher chrF scores than in Table 5.

[5] Moreover, in preliminary experiments, we compared the performance of our baseline models with the FST-based tools of Alva and Oncevay (2017) and Ortega et al. (2020) for Shipibo-Konibo and Ashaninka, respectively, and we found that the rule-based systems could not overcome the data-driven ones for synthetically generated errors in input sentences.

Yo Joong Choe, Jiyeon Ham, Kyubyong Park, and Yeoil Yoon. 2019. A neural grammatical error correction system built on better pre-training and sequential transfer learning.

Pravallika Etoori, Manoj Chinnakotla, and Radhika Mamidi. 2018. Automatic spelling correction for resource-scarce languages using deep learning. In *Proceedings of ACL 2018, Student Research Workshop*, pages 146–152, Melbourne, Australia. Association for Computational Linguistics.

Shaona Ghosh and Per Ola Kristensson. 2017. Neural networks for text correction and completion in keyboard decoding.

Marcin Junczys-Dowmunt, Roman Grundkiewicz, Shubha Guha, and Kenneth Heafield. 2018. Approaching neural grammatical error correction as a low-resource machine translation task.

Hao Li, Yang Wang, Xinyu Liu, Zhichao Sheng, and Si Wei. 2018. Spelling error correction using a nested rnn model and pseudo training data.

John Ortega, Richard Alexander Castro-Mamani, and Jaime Rafael Montoya Samame. 2020. Overcoming resistance: The normalization of an Amazonian tribal language. In *Proceedings of the 3rd Workshop on Technologies for MT of Low Resource Languages*, pages 1–13, Suzhou, China. Association for Computational Linguistics.

Maja Popović. 2015. chrF: character n-gram F-score for automatic MT evaluation. In *Proceedings of the Tenth Workshop on Statistical Machine Translation*, pages 392–395, Lisbon, Portugal. Association for Computational Linguistics.

Danish Pruthi, Bhuwan Dhingra, and Zachary C. Lipton. 2019. Combating Adversarial Misspellings with Robust Word Recognition. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5582–5591, Florence, Italy. Association for Computational Linguistics.

Keisuke Sakaguchi, Kevin Duh, Matt Post, and Benjamin Van Durme. 2017. Robsut wrod reocginiton via semi-character recurrent neural network.

Max White and Alla Rozovskaya. 2020. A comparative study of synthetic data generation methods for grammatical error correction. In *Proceedings of the Fifteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 198–208, Seattle, WA, USA → Online. Association for Computational Linguistics.

Roberto Zariquiey, Harald Hammarström, Mónica Arakaki, Arturo Oncevay, John Miller, Aracelli García, and Adriano Ingunza. 2019. Obsolescencia lingüística, descripción gramatical y documentación de lenguas en el perú: hacia un estado de la cuestión. *Lexis*, 43(2):271–337.

A Languages

Asháninka (cni) is an Arawak language variety that takes part in the so-called Asháninka-Ashéninka dialect complex, spoken by more than 77,000 people in Central and Eastern Peru and in the state of Acre in Eastern Brazil. Ashaninka, which belongs to the Nihagantsi subgroup of the Arawak family, is spoken along the Tambo, Ene, Apurímac, Urubamba and Bajo Perené rivers in Central Peruvian Amazon. Asháninka has 16 consonants (including a nonspecified nasal consonant) and four vowels. Ashaninka has an official alphabet recognised by the Ministry of Education of Peru since 2015. Previous to that, the Summer Institute of Linguistics published some materials in the language using an early orthographic proposal. Both traditions are only slightly different. Asháninka is an agglutinating, polysynthetic and verb-initial language. It is also strongly head-marking and thus the verbal word is often highly morphologically complex, with several positional slots and a large inventory of aspectual and modal categories. Grammatical relations (subject and object) are indexed as affixes on the verb itself.

Yanesha' (ame) is an Peruvian Arawak language that belongs to the Pre-Andine branch. It is spoken in the Amazonian highlands of Central Peruvian by approximately 5,000 people. Yanesha' exhibits a saliently large phonological inventory with 12 vowels (including long, aspirated and glottalised segments) and 23 consonants, some of which is typologically unusual. Yanesha' exhibits two currently competing orthographic traditions, one early proposed by the Summer Institute of Linguistics and a full revision of it conducted in 2011 and recognized as the official alphabet of the language. Yanesha' is an agglutinating, polysynthetic language with a VSO constituent order. Yanesha' is strongly head-marking and therefore the verbal word is highly morphologically complex.

Yine (pib) is a Peruvian language of the Arawak family spoken by approximately 3,000 people along the the Ucayali and Madre de Dios rivers. Yine has five vowels and 16 consonants. There are two currently competing orthographic traditions for Yine, one proposed by the Summer Institute of Linguistics in 1965 and an official alphabet recognized by the Ministry of Education of Peru since 2015. Yine is highly polysynthetic and agglutinating. Since it is a predominantly head marking langauge, most of the morphological complexity of the language is related to verbs.

Shipibo-Konibo (shp) is a Pano language spoken by approximately 35,000 native speakers in central Peruvian Amazon. Shipibo-Konibo exhibits 15 consonants and four vowels. As is the case with other Peruvian Amazonian languages, the language exhibits two competing orthographic traditions, one early proposed by the Summer Institute of Linguistics and another official one, promoted by the Ministry of Education of Peru. These orthographies are sometimes randomly used by the speakers, creating salient amount of cross-speaker variation. Shipibo-Konibo is mainly agglutinating, synthetic and almost exclusively suffixing (with only a closed set of prefixes related to body-part concepts) Word order is pragmatically oriented, but there is some tendency towards SOV constructions. Verbs lack subject and object crossreference, but exhibit a large set of TAME markers.

B Hyperparameters

- Architecture: Bi-directional LSTM
- Hidden layer: 50
- Vocabulary size: 5,000 for Shipibo-Konibo, Asháninka y Yanesha; and 3,000 for Yine
- Epochs: 100
- Batch size: 32
- Optimiser: Adam
- Learning rate: 0.001
- Loss function: categorial cross-entropy