Evaluating a Phonotactic Learner for MITSL_2^2 Languages

Jacob K. Johnson
University of Utah
jacob.k.johnson@utah.edu

Aniello De Santo
University of Utah
aniello.desanto@utah.edu

1 Introduction

In the computational study of natural language phonotactics, work in formal language theory has highlighted subclasses of regular languages (*subregular* classes) as providing insights into the fundamental properties underlying a variety of typologically attested patterns (McNaughton and Papert, 1971; Chandlee, 2017). Of particular note are Tier-based Strictly Local (TSL) languages, which capture long-distance phenomena through a formalization of the notion of phonological tier (Heinz et al., 2011). Beyond typological coverage, attention has been paid recently to how observations about the subregularity of phonotactics can be tied to learnability.

In this sense, grammatical inference algorithms have shown how TSL languages can be learned efficiently from positive data only (Jardine and Heinz, 2016a; Jardine and McMullin, 2017). Additionally, super-classes of TSL have been defined to extend its typological coverage, while retaining (and improving on) its desirable (subregular) properties (Graf and Mayer, 2018, a.o.).

Here, we provide an implementation of De Santo and Aksënova (2021)'s grammatical inference learning algorithm for one such extension — *Multiple Input-sensitive Tier-based Strictly Local* languages (MITSL; De Santo and Graf, 2019) — following the standard of *SigmaPie* (Aksënova, 2020)[1], and evaluate it on an array of patterns with varying degrees of (subregular) complexity.[2]

As we illustrate below, MISTL languages are able to capture the interaction of local and non-local constraints, and while also handling multiple dependencies simultaneously. Their practical learnability thus has strong implications for the viability of grammatical inference/subregular approaches to phonotactic learning broadly. Additionally, the transparency and provable correctness of the learning algorithms developed for such formal classes can be of help in probing properties of phonotactic corpora more generally.

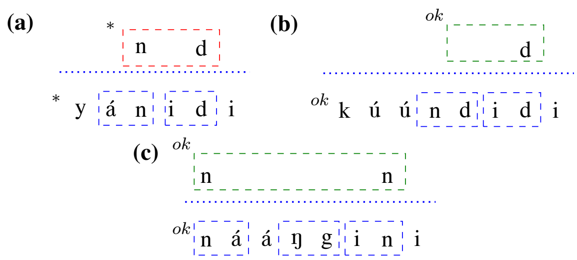

Figure 1: ITSL_2^2 analysis of Yaka nasal harmony from (De Santo and Aksënova, 2021), illustrating a 2-local projection and 2-local tier constraints. (a) is ill-formed because of adjacent *[nd], but [n,d,g,ŋ] are projected on the tier only when not in a nasal-stop cluster in the input (cf. (b), (c)).

2 MITSL

TSL grammars encode long-distance dependencies by enforcing local constraints over a subset of segments in the input alphabet (the tier) identified via a projection function. While TSL covers many unbounded phonotactic patterns, past work has highlighted two limits: 1) since TSL's tier projection is only sensitive to properties of individual segments, it cannot handle cases where local and non-local requirements interact; 2) TSL's reliance on a single tier (i.e., lack of intersection closure) makes it unable to handle multiple non-interacting dependencies. MITSL languages (De Santo and Graf, 2019) address these issues thanks to:

1. an *input-sensitive* (ITSL) projection, sensitive not only to a segment by itself but also to its *context* in the input string (Fig. 1);
2. closure under intersection (MTSL): *multiple projection functions* (tiers) are available, each with its dedicated local constraints.

[1] https://pypi.org/project/SigmaPie/
[2] Our codebase is available here.

Essentially, an ITSL projection is an input strictly local function (Chandlee and Heinz, 2018), such that projection is decided by whether a segment belong to the tier alphabet, and by its immediately adjacent segment. The MTSL component makes it so each string is evaluated on any number of distinct tier, and wellformedness is guaranteed if and only if the string is wellformed on each individual tier simultaneously.

3 Learning MITSL$^2_2$ Patterns

De Santo and Aksënova (2021) propose a learning algorithm for MITSL$^2_2$ languages, where the tier projection and the tier constraints are bounded to bigrams (MITSL2IA), extending McMullin et al. (2019)'s algorithm for MTSL$^2_2$ languages.

Consistently with (McMullin et al., 2019), MITSL2IA builds on the intuition that if a bigram $\rho_1\rho_2$ is banned on some tier, then it will never appear in string-adjacent contexts. For each $\rho_1\rho_2$ absent from the training data, the goal is therefore to determine which segments can be safely removed from the associated tier. To do so, the algorithm incorporates the notion of a 2-path (Jardine and Heinz, 2016b). Intuitively, a 2-path can be thought of as a precedence relation $(\rho_1 \ldots \rho_2)$ accompanied by the set X of symbols that intervene between ρ_1 and ρ_2. Formally, each 2-path is therefore a 3-tuple of the form $\langle\rho_1, X, \rho_2\rangle$.

In short, by examining the set of 2-paths present in the training data, we can determine which segments are freely distributed with respect to a bigram $\rho_1\rho_2$ that is known to be banned on some tier. By associating each potential bigram constraint to a specific tier, MITSL2IA is thus able to handle multiple tier projections. In addition, De Santo and Aksënova (2021) handle the extended input contexts by generalizing the definition of tier symbols from segments to bigrams. Each tier-bigram $\rho_1\rho_2$ is thus a 4-factor for MITSL2IA, such that the algorithm can evaluate a target element σ and its left or right local context (see De Santo and Aksënova, 2021, for technical details).

Consistently with other work of this type, MITSL2IA is *guaranteed* to learn target grammars efficiently (polynomial in time and data) in the limit, *if* the input sample is *characteristic* — it contains all the information necessary to distinguish the specific target pattern(s) (De la Higuera, 2010). This property of learning algorithms in this tradition makes them not only valuable from

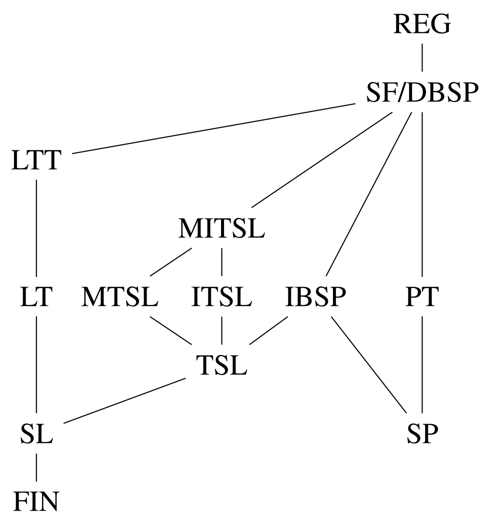

Figure 2: Subsumption of subregular classes, with the TSL extensions as of (De Santo and Graf, 2019).

a theoretical perspective, but it also allows us to use them for a deeper exploration of how target phenomena of interest are represented in data sets of different kind. In this work, we start pursuing questions in this direction, with a preliminary investigation of the practical efficacy of De Santo and Aksënova (2021)'s learner.

4 Evaluating MITSL2IA

We implemented MITSL2IA in Python 3 following requirements of the *SigmaPie* toolkit, and evaluated it over patterns resembling natural language phenomena belonging to different subregular classes, according to the pipeline argued for by Aksënova (2020).

The learner had as input datasets which were either the artificial outputs of harmonic string generators incorporating a target grammar, or natural language word-lists with simplified alphabets created by masking "irrelevant" symbols (for details see Aksënova, 2020). Each artificial dataset contained 1000 randomly sampled strings, and up to $130K$ words for the simplified natural language corpora.

MITSL2IA's output grammars were then used as input to string generators. We evaluated the *consistency* of the learned grammar with respect to the target grammar, by computing the number of strings in the newly generated sample that were well-formed according to the target generalization (% of new strings accepted by the original grammar).

	Aksënova (2020)				This Paper
	SP	SL	TSL	MTSL	MITSL
Word-final devoicing					
T	✗	✓	✓	✓	✓
A	68%	100%	100%	100%	100%
N_1	58%	100%	100%	100%	100%
Single vowel harmony without blocking					
T	✓	✗	✓	✓	✓
A	100%	83%	100%	100%	100%
N_2	100%	72%	100%	100%	100%
Single vowel harmony with blocking					
T	✗	✗	✓	✓	✓
A	84%	89%	100%	100%	99%
Several vowel harmonies without blocking					
T	✓	✗	✓	✓	✓
A	100%	69%	100%	100%	100%
Several vowel harmonies with blocking					
T	✗	✗	✓	✓	✓
A	76%	59%	100%	100%	99%
N_3	76%	70%	67%	95%	99%
Vowel harmony and consonant harmony without blocking					
T	✓	✗	✗	✓	✓
A	100%	64%	74%	100%	100%
Vowel harmony and consonant harmony with blocking					
T	✗	✗	✗	✓	✓
A	83%	64%	69%	100%	100%
Unbounded tone plateauing					
T	✓	✗	✗	✗	✓
A	100%	85%	90%		100%
Two locally-driven long-distance assimilations (ITSL restrictions)					
T	✗	✗	✗	✗	✓
A					100%

Table 1: (T)heoretical expectations and performance of 5 subregular learners on (A)rtificial and simplified (N)atural language input data-sets. MITSL corresponds to the learner evaluated in this paper. N_1: German; N_1: Finnish; N_1: Turkish.

Following (Aksënova, 2020), we included patterns from subclasses of MITSL which are also known to be attested in natural languages (cf. Fig. 2 and Tbl.1). An unbounded tone-plateauing pattern (Hyman and Katamba, 2010; Jardine, 2016) characterized as Strictly Piecewise (SP) in past literature also served as a simple ITSL pattern (De Santo and Graf, 2019), and we added an explicitly MITSL pattern not included in (Aksënova, 2020). The performance of our implementation on each dataset was compared to what reported by Aksënova (2020) for a battery of other subregular learners (Tbl. 1).

5 Results and Discussion

Recall that while MITSL2IA is guaranteed to learn a pattern when a sample is characteristic, we did not control for that requirement when generating the input datasets. Our results show that even with small, randomly generated datasets the learner performed well on all target patterns (Tbl. 1).

While a larger scale evaluation paradigm is an important future step, these results are encouraging in supporting the reliability of MITSL2IA in practical scenarios and highlight the importance of implementations to test grammatical inference algorithms beyond theoretical convergence.

Importantly, since subregular classes in the TSL "neighborhood" stand in a subsumption relation with respect to each other (Fig. 2), the learner is theoretically expected to be able to generalize correctly not only when trained on strictly MISTL patterns, but to each one of the simpler patterns as well. However, high-performance on subclasses of MITSL (SL, TSL, MTSL) was not trivially granted in practice. Since MITSL2IA needs evidence from all possible local contexts to recognize tier elements, it requires more evidence for simple patterns (e.g., whether to project a sibilant on a tier or not) than its less expressive counterparts. So one might expect the needed characteristic samples to be larger, when moving from simpler to more expressive learning algorithms. In fact, this effect is behind the slightly lower performance on the single harmony data, compared to the TSL and MTSL learners. However, this is where the transparent nature of the algorithm shines. By inspecting the grammar outputted by MITSL2IA, it was possible to infer that the initial sample for those patterns lacked evidence to discard one particular element from the harmony tier.

For example, the input data was insufficient for the learner to converge on the target grammar for the "single vowel harmony with blockers" and the "several vowel harmonies with blocking" cases. In both instances, this was because an element that should have been transparent to the restriction was not removed from the tier, because of the default assumption that all elements are on the tier restricting any unattested bigram (4-factor).

To further probe the relation between algorithm convergence and information gaps in the input sample, we thus defined an injection procedure on top of the learner itself. Specifically, we implemented an automatic processes to detect miscathegorized strings in the newly generated samples due an element β that should have been removed from the tier restricting the factor a as follows:

1. Find the set of all paths B which contain β and are attested to be interveners for a;

2. Let the set B' be formed by removing β from all paths in B;

3. Every path $b \in B' - A$ must be attested in the input while intervening a. In practice, for MITSL$^2_2$ grammars, these strings can be formed by overlapping m-factors in b, and, for $a = \alpha_1\alpha_2$, having the string start with α_1 and end with α_2. Importantly, because paths are sets, m-factors in b can be repeated.

Re-running the learner on the injected samples resulted in a 100% performance on all three of the 99% cases, suggesting that transparent learners of this kind could be used to inspect the quality of the data in natural language samples available to phonotactic learners more broadly. Additionally, this suggests ways of expanding the current batch learning approach to MISTL data to online algorithms.

6 Conclusion

This work adds to the theoretical contribution of (De Santo and Aksënova, 2021), by showing how an implementation of their algorithm is strikingly successful on a variety of phonotactic patterns. We also argue that the evaluation pipeline adopted here, following (Aksënova, 2020), is valuable for future empirical testing of subregular learning algorithms in the grammatical inference tradition.

At the same time, we suggest more broadly that implemented grammatical inference algorithms like MITSL2IA can be crucial in the future to more deeply probe how/whether sufficient information about target patterns appears in phonotactic corpora, contributing to the study of the relation between data and learning performance both in humans and machines.

References

Alëna Aksënova. 2020. *Tool-assisted induction of subregular languages and mappings*. Ph.D. thesis, State University of New York at Stony Brook.

Jane Chandlee. 2017. Computational locality in morphological maps. *Morphology*, 27(4):599–641.

Jane Chandlee and Jeffrey Heinz. 2018. Strict locality and phonological maps. *Linguistic Inquiry*, 49:23–60.

Colin De la Higuera. 2010. *Grammatical inference: learning automata and grammars*. Cambridge University Press.

Aniello De Santo and Alëna Aksënova. 2021. Learning interactions of local and non-local phonotactic constraints from positive input. *Proceedings of the Society for Computation in Linguistics*, 4(1):167–176.

Aniello De Santo and Thomas Graf. 2019. Structure sensitive tier projection: Applications and formal properties. In *International conference on formal grammar*, pages 35–50. Springer.

Thomas Graf and Connor Mayer. 2018. Sanskrit n-retroflexion is input-output tier-based strictly local. In *Proceedings of SIGMorPhon 2018*.

Jeffrey Heinz, Chetan Rawal, and Herbert G Tanner. 2011. Tier-based strictly local constraints for phonology. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human language technologies*, pages 58–64.

Larry M Hyman and Francis X Katamba. 2010. Tone, syntax, and prosodic domains in luganda. *ZAS Papers in Linguistics*, 53:69–98.

Adam Jardine. 2016. Computationally, tone is different. *Phonology*.

Adam Jardine and Jeffrey Heinz. 2016a. Learning tier-based strictly 2-local languages. *Transactions of the Association for Computational Linguistics*, 4:87–98.

Adam Jardine and Jeffrey Heinz. 2016b. Learning tier-based strictly 2-local languages. *Transactions of the ACL*, 4:87–98.

Adam Jardine and Kevin McMullin. 2017. Efficient learning of tier-based strictly k-local languages. In *Language and Automata Theory and Applications, 11th International Conference*, LNCS, pages 64–76. Springer.

Kevin McMullin, Alëna Aksënova, and Aniello De Santo. 2019. Learning phonotactic restrictions on multiple tiers. *Proccedings of SCiL 2019*, 2(1):377–378.

Robert McNaughton and Seymour Papert. 1971. *Counter-Free Automata*. MIT Press, Cambridge.