http://dx.doi.org/10.17959/sppm.2016.22.3.533

Capturing variation and gradience in identity avoidance: A case of machine learning

Young-ran An
(KC University)

An, Young-ran. 2016. Capturing variation and gradience in identity avoidance: A case of machine learning. *Studies in Phonetics, Phonology and Morphology* 22.3. 533-557. This paper makes the point that a grammar appears to be a sum of tendencies, rather than an aggregate of all-or-none instances. That is, the grammar is not formed by an across-the-board law, but teems with variation and gradience. For a case in point, this paper presents the phenomenon of consonant insertion in Korean total reduplication. To see whether this kind of grammar with variation and gradience can be possibly, and eventually humanly, learned, it is simulated using a model of grammar learning. The instantiation of machine learning in this paper shows that a grammar with variation and gradience can indeed be learned. **(KC University)**

Keywords: variation, gradience, identity avoidance, machine learning

1. Introduction

Identity avoidance has been attested in the data of Korean total reduplication with consonants inserted (CIs, in boldface in the examples; cf. Jun and Lee 2006), both in the corpus and an experiment.

(1)	a.	alok-**t**alok	‘dappled’
	b.	oson-**t**oson	‘harmoniously’
	c.	ulak-**p**ulak	‘roughly’
	d.	umul-**tʃ**’umul	‘hesitantly’
	e.	aki-**tʃ**aki	‘charming’

In determining the base between V-initial portion and C-initial portion in the reduplicative forms in (1), I adopted the argument presented in Jun and Lee (2006). According to Jun and Lee, whether the first or second portion can be used independently is a decisive criterion. That is, the portion that can be used alone is regarded as a base. As for the other instances of neither of the portions being used independently, and either of the portions being able to stand alone, I included these cases in my data of consonant insertion, based on the universal principle of phonology: a syllable onset is required in such an unmarked form as a reduplicant.

As was found out, CIs in the reduplicants were inclined to be dissimilar from the existing Cs in the bases. This phenomenon of identity avoidance shows tendency, which is not categorical, but gradient. Furthermore, it was also found in a word-creation experiment that different speakers appear to have different Cs to insert as CIs (An 2012, 2013).

Table 1. Frequency of CIs in a word creation experiment (Tokens = 1646)

CI	*t*	*ʧ*	*k*	*p*	*n*	*s*	*m*	*l*
Tokens	514	497	252	148	101	90	41	3
(%)	31.25	30.19	15.31	8.99	6.14	5.47	2.49	0.18

Among these possible CIs, the participating speakers (N=15) particularly tended to put in *t*, *ʧ* when they were asked to make a reduplicated word with a given base, which is a nonce base of a form, VCVC-_____. I call them *t*-dominant and *ʧ*-dominant group, respectively. The stimuli in this experiment did not have *ʧ* in the context, which made it possible to focus on contextual effects only in the *t*-dominant group. Among the 15 participants, there were 4 participants (P5, P13, P14, P15) identified as belonging to the *t*-dominant group and 5 participants (P2, P4, P7, P11, P12) that could be classified into the *ʧ*-dominant group. I could see if the speakers who prefer *t* in general would be more likely to insert other Cs when the context contains *t*, in order to go by the identity avoidance. Meanwhile, I did not go into details about the *ʧ*-dominant group since there is no contextual effect testable under the circumstances where there is no *ʧ* in the contexts.

Table 2. *t* choice in the *t* context (36 words) and in the no-*t* context (75 words) in the case of *t*-dominant group

Participants	Observed	Expected	O/E	Observed	Expected	O/E
P5	8	13.3	0.6	33	27.7	1.19
P13	12	21.1	0.57	53	43.9	1.21
P14	28	27.2	1.03	56	56.8	0.99
P15	15	19.5	0.77	45	40.5	1.11

The O/E ratio was calculated with the Observed and Expected values: if the O/E is greater than 1, it means that the number of *t* occurrences in the given context is overrepresented than expected; and if the O/E is less than 1, it means that the number of *t* occurrences is underrepresented than expected. The O/E ratios in Table 2 are for the *t* choice in the *t* context (the O/E ratio to the left) and for the *t* choice in the no-*t* context (the O/E ratio to the right). These two kinds of O/E ratios were simultaneously presented for comparison's sake. Table 2 shows that in general the O/E ratio for the *t* choice in the no-*t* context indicates overrepresentation; that is, the participants showed a tendency to insert *t* more often than expected in the no-*t* context, whereas the *t* choice is underrepresented in the *t* context for some participants, observing the principle of identity avoidance.

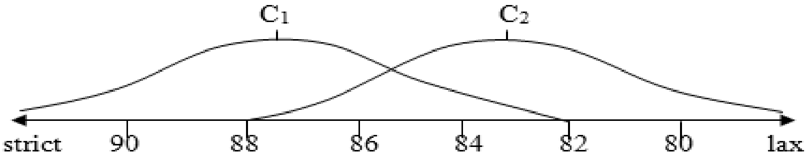
In this paper, I make an attempt to present the grammar of the Korean CI-reduplication, incorporating what has been found, i.e., identity avoidance and speaker preference, based on the classical Optimality Theory (OT) and its modified version, Gradual Learning Algorithm (GLA), which I am adopting here. First, I introduce the GLA in section 2. For the data, I analyze them due to the classical OT and the GLA in section 3, which helps to compare the two versions. I finish up the paper with some remarks on variation and gradience in identity avoidance, which should be considered in the grammar, in section 4.

2. Gradual Learning Algorithm

A grammar that captures the findings of variation and gradience in speakers' choice of CIs in the word creation experiments must produce variable outputs. The Gradual Learning Algorithm, GLA, has been claimed to be an appropriate algorithm for learning grammars, particularly based on Optimality Theory, OT (Boersma 1997, Boersma and Hayes 2001). It is argued to be better at dealing with free variation, noisy learning data, and gradient well-formedness. The GLA is characterized by its assumption that the constraints are continuous, not discrete, and the grammar is stochastic. Therefore, the GLA allows the grammar to produce variable outputs, which may account for different outputs by different speakers of a language. In this section I will briefly describe how the GLA works and how it can capture variation in grammar.

The conceptual bases for the GLA are a "continuous ranking scale" and "stochastic candidate evaluation." With the continuous scale, the GLA can handle both categorical and gradient rankings. The position of a constraint, at an evaluation time, is perturbed by random noise, before it is finally selected. If the ranges of selection points for constraints do not overlap, then the ranking scale ends up with the traditional categorical ranking. However, if some constraints turn out to have overlapping ranges, then the ranking scale will show free ranking.

(2) Overlapping ranking distributions (Boersma and Hayes 2001: 49)



According to Boersma and Hayes (2001), the hypothetical ranking values for the constraints, C_1 and C_2 are 87.7 and 83.1, respectively. Thus we may see a ranking of C_2 >> C_1 at some occasions (5.2%), although C_1 >> C_2 should hold most of the time (94.8%).

The GLA locates an appropriate ranking value for a constraint, in the process from the initial state up to the final state. Every constraint starts at the same value. A learning datum consisting of adult surface forms is presented to the algorithm. Then for each constraint a selection point is picked according to the constraint's current ranking value. This generation process follows the standard mechanisms of OT. If a generated form is identical to the learning datum, no further actions will take place. However, if there happens to be a mismatch between the generated form and the learning datum, then the algorithm will take measures in order for the grammar to generate the learning datum. It will basically change the ranking values of the constraints, in which violations matching in the two rival candidates will be cancelled out, as in the following:

(3) Mark cancellation (Boersma and Hayes 2001: 52)

/underlying form/	C_1	C_2	C_3	C_4	C_5	C_6	C_7	C_8
✓ Candidate 1 (learning datum)	*!	*~~*~~	~~*~~		*			~~*~~
☞ Candidate 2 (learner's output)		~~*~~	~~*~~	*		*		~~*~~

Then the adjustment of the ranking values is made repeatedly with further exposure to learning data through the cycle of presentation of learning datum – generation – comparison – adjustment, until the learner's output matches the learning datum.

(4) Adjusting the ranking values (Boersma and Hayes 2001: 53)

/underlying form/	C_1	C_2	C_3	C_4	C_5	C_6	C_7	C_8
✓ Candidate 1 (learning datum)	*→	*→			*→			
☞ Candidate 2 (learner's output)				←*		←*		

The GLA has been relatively successful in dealing with example data with free variation: for instance, the resulting grammar via machine ranking for the variation data, e.g., in Ilokano could generate the predicted variations by running the input underlying forms with the output probabilities. Not only could the grammar generate all and only the correct forms, but it also could produce the matching frequencies in the learning data. That is, when the language has a single output form 100% of the time, the machine grammar also generated only that output, e.g., /paʔlak/ → [pa.lak]. When the language has variation between two forms, 50% of the time each, the machine grammar also generated the two forms with closely matching percent of the time for each form, e.g., /taʔo-en/ → [taw.ʔen] ~ [taʔ.wen]. When the language has variation among three forms, the machine grammar successfully predicted that alternation with about a third of the frequency for each form, e.g., /bwaja/ → [bu:.bwa.ja] ~ [bwaj.bwa.ja] ~ [bub.wa.ja].

3. The grammar

In this section I apply the GLA, specifically OTSoft (Hayes 2010), to the reduplication data with a CI, to see how learning of the CI choice takes place, which includes variation. I pick some representative examples to illustrate the learning, among the dictionary (*Essence Korean Dictionary* 2006) data and the data from the distinct groups in an experiment. For example, I will show learning of a single, general grammar for the dictionary data, which are categorical and the data from the word creation experiment, which displays variation in the choice of CIs (e.g., *t* and *tʃ*).

In addition, I will show how an individual speaker's grammar can be learned: an individual grammar which prefers a specific C (*t* or *tʃ*) with no consideration of context; and an individual grammar which prefers a specific C (*t* or *tʃ*) that takes context into account.

First, I provide description of the data and explanation of the relevant constraints on the basis of OT, which in turn are to be used in an input file for running the GLA. Second, I run the input file in the GLA, so that we can see how well the model could learn a grammar for the reduplication case. Then the resulting grammar is assessed in comparison to the targeted grammar.

3.1 The learning data

With respect to the dictionary data, I use the reduplicative forms with VCVC-bases that carry *t*, *p*, *tʃ* as CIs, (5) = (1).

(5) a.	alok-**t**alok	'dappled'
b.	oson-**t**oson	'harmoniously'
c.	ulak-**p**ulak	'roughly'
d.	umul-**tʃ'**umul	'hesitantly'
e.	aki-**tʃ**aki	'charming'

The forms with VCVC-bases and CI = {*t*, *p*, *tʃ*} amounted to 51. The surface forms in the dictionary do not have variants, and they will produce a categorical grammar with only a single optimal output for an underlying form.

The responses in the experiment consist of nonce reduplicative forms with VCVC-bases, which were open to any choice of CI (= **C** in (6)), some of whose examples are:

(6) a. ikip-**C**ikip
b. isim-**C**isim
c. unup-**C**unup
d. ukuŋ-**C**ukuŋ
e. atan-**C**atan
f. apam-**C**apam

I will consider the forms with CI = {*t*, *p*, *tʃ*}, among others, for the sake of comparison with the grammar of the dictionary data. Regarding the response data from the experiment, I will obtain three kinds of grammar with the GLA: a single general grammar for all cases with CI = {*t*, *p*, *tʃ*}; an individual grammar for a speaker who prefers *t* without consideration of *t* in the context; and an individual grammar for a speaker who prefers *t* with consideration of *t* in the context.

3.2 The classical OT-based analysis

I have identified two major factors which appear to affect the choice of CIs: speakers' preference and consideration of context. Concerning the speakers' preference, I found two distinct groups: *t*-dominant group and *tʃ*-dominant group. There is no specific constraint that can impose the tendency of preference; therefore, I will simply use a segmental markedness constraint to show preference for a certain segment.

With regard to the concept of constraints, the classical OT assumes a universal set of ranked and violable output constraints (Prince and Smolensky 1993); however, there is an alternative approach which argues that constraints are learned from language-specific data on the basis of Universal Grammar that consists of a feature set and a constraint format (Hayes and Wilson 2008). I do not necessarily adopt one approach over the other in my analysis; however, I am aware of these different views, and I rather sympathize with Yip's (1995) remark that the strongest claims of universality for constraints may not be too agreeable and reasonable particularly when it comes to the issue of how to handle language-specific and morpheme-specific constraints in OT.

As regards a constraint for the contextual factor (avoiding inserting the same C as a C in the context, in particular), I use a family of *REPEAT constraints, *à la* Yip (1995), since there was avoidance of repeating the same segment *t* when *t* already exists in context. This type of constraint has a long tradition in phonology, which has been called the Obligatory Contour Principle, OCP (Goldsmith 1976, Leben 1973, McCarthy 1979, 1981, 1986, Steriade 1982, Clements and Keyser 1983, Yip 1988, 1995, 1998, among many others). Yip utilized the constraint, *REPEAT in a morphological sense, particularly when discussing reduplication data, and I also use the same constraint in considering the sensitivity to context in this section, i.e., avoiding the same C as a context C for a CI. I use sensitivity to context interchangeably with identity avoidance.

(7) Constraints

C: The reduplicant must begin with a consonant C.
(e.g., Constraint, t means that *t* is inserted in the beginning of the reduplicant.)

*REPEAT(segment): Output must not contain identical segments.
(e.g., *REPEAT(t) requires that *t* cannot occur repeatedly.)

ONSET: Syllables must have onsets.

DEP-BR: Every element of the reduplicant has a correspondent in the base. ("No epenthesis")

MAX-BR: Every segment of the base has a correspondent in the reduplicant. ("No deletion")

Place-Markedness Hierarchy:
*PL/LAB, *PL/DORS >> *PL/COR
(Alderete et al. 1999, Prince and Smolensky 1993)

*ONSETV: The leftmost onset segment in a syllable does not have the specified sonority level.
(This constraint family assumes a hierarchy, e.g., *SONV >> *OBSV, which prefers an obstruent onset to a sonorant onset: Lombardi 2002, Smith 2003)

I did not present all constraints that may be at work for this grammar; rather, I provided all and only the constraints that are apparently more relevant for the time being in accounting for the choice of CIs. For example, I have not listed some constraints like Syllable Contact Law, SYLLCON ("Rising sonority across a syllable boundary is not allowed"), which seems to play a role in this grammar but not to be as critical as the other constraints.

Constraint, C is given along the lines of Yip's constraints for a specific consonant; for example, she used "p" to stand for "the Intensive prefix should end in [p]" for the Turkish reduplication case with a prefix. I use three segmental constraints, t, p, ʧ, to require that the "reduplicant in the CI-reduplication must start with *t* (or *p* or *ʧ*)." *REPEAT(segment) could have been presented in other term such as *REPEAT(feature) so that it can show what important role features (e.g., place of articulation and manner of articulation), rather than segments, are playing in avoiding repetition; however, I chose *REPEAT(segment) for the current data since I have not had discussion on avoidance of identical features in this section. Besides, a grammar may become more powerful in its explanatory capacity if we elaborate on the constraint, e.g., *REPEAT(t): we could make it into two separate constraints, *REPEAT(t=C_1) which militates against repetition of a segment that is identical to the first consonant in the base and *REPEAT(t=C_2) which militates against repetition of a segment that is identical to the second consonant in the base. Indeed this idea is sensible enough considering relevant data in the languages that distinguish the influential status of consonants in different positions, but I do not adopt this idea in here yet since I can do without it for the current grammar.

Constraints, ONSET, DEP-BR, and MAX-BR are presented in their canonical sense. As for the Place-Markedness Hierarchy, I will use *LAB, *DOR, *COR, instead of *PL/LAB, *PL/DORS, *PL/COR, respectively, in tableaux henceforth, for convenience's sake (Lombardi 2002). With regard to the *ONSET/X constraints, I use several constraints like the following[1]:

(8) Some sonority cline constraints (Lombardi 2002: 240)
a. *FRICV (prohibits a fricative onset)
*STOPV (prohibits a stop onset)
Universal ranking *FRICV >> *STOPV
b. *SONV (prohibits a sonorant onset)
*OBSV (prohibits an obstruent onset)
Universal ranking *SONV >> *OBSV

I will use *FRICV, *STOPV, *NASV, which bans the occurrence of a nasal in onset, among other sonorants: these constraints can be seen in the example tableaux in this section. In principle, I can utilize constraints like *GLIDEV (prohibits a glide onset) and *LIQUIDV (prohibits a liquid onset). I also employed *AFFRICV (prohibits an affricate onset), the sonority value of which is not completely clear and indicated with (>>) due to the unclear domination relationship with *FRICV and with *STOPV; I used this constraint to demonstrate that there must be some difference between stop *t* and affricate *ʧ* although they are both highly frequent as a CI in the data.

[1] With regard to the Place-Markedness Hierarchy, *REPEAT(segment), and the sonority cline constraints, I limit the domain of evaluation to the reduplicants, in (10) and (11). The Place-Markedness constraints, in particular, are evaluated against the properties of CIs only, for the sake of clarity and simplicity of comparison among the candidates.

I employ the above constraints on a necessity basis; i.e., not all of them are shown in tableaux when they are undominated or too low in the ranking hierarchy. For instance, Max-BR is important but I do not include it in tableaux since it is assumed to be respected by all output candidates I am putting forward. Dep-BR is also part of the grammar, but it is always critically violated by winning candidates, because epenthesis of a consonant must take place to obtain an optimal reduplicative form in our data.

The set of constraints and a grammar may not be perfect as they are presented in this section; my goal of proposing constraints in this section is not to furnish a full-fledged grammar at this point, but rather to see if the current data can be accounted for more or less with a grammar.

(9) Constraints suggested with a partial hierarchy
Max-BR, Onset >>
*Dor, *Lab >> *Cor
*NasV >> *FricV >> *AffricV (>>) *StopV
t, ʧ, p
*Repeat(p), *Repeat(ʧ), *Repeat(t)
>> Dep-BR

The domination relationship among some of the constraints is not crystal clear, and we can see some examples based on these constraints and their rankings as follows: I instantiate an example from the dictionary data, and a nonword example from the word creation experiment.

(10) [oson-**t**oson] 'harmoniously' (=1b, 5b)

/oson-Red/	On-set	*Lab	*Cor	*Affric V	*Stop V	t	ʧ	p	*Repeat (p)	*Repeat (ʧ)	*Repeat (t)
o.son.-o.son	**!										
☞ o.son.-**t**o.son	*		*		*		*	*			
o.son.-**p**o.son	*	*!			*	*	*				
☹ o.son.-**ʧ**o.son	*		*	*		*		*			

The actual winner in the tableau (10) should be oson-**t**oson, as it is the real output in lexicon; however, there is one more output that is predicted according to the given grammar, i.e., oson-**ʧ**oson. This other candidate can be also an optimal output based on the suggested constraints and the ranking hierarchy. This is problematic since the proposed grammar cannot produce a single optimal output; on the other hand, this suggests that there could be more than one output generated by a grammar. It further implies that the strict domination relationship among constraints may not be able to account for every datum, which may need a stochastic ranking of constraints. It is possible that some constraints like *t* and *ʧ* above are overlapping in their ranges, by which *t* can be chosen as an optimal winner sometimes and *ʧ* can be chosen as an optimal winner some other times.

(11) [atan-**C**atan] (=6e, nonce word)

/atan- Red/	On- set	*Lab	*Cor	*Affric V	*Stop V	t	ʧ	p	*Repeat (p)	*Repeat (ʧ)	*Repeat (t)
a.tan.- a.tan	**!										
a.tan.- **t**a.tan	*		*		*		*	*			*!
a.tan.- **p**a.tan	*	*!			*	*	*				
☞ a.tan.- **ʧ**a.tan	*		*	*		*		*			

The tableau in (11) shows that *t* in the context does not welcome another *t* to be inserted by virtue of *Repeat(t), and therefore, an underlying form that contains *t* ends up surfacing with the reduplicant containing another CI, *ʧ* in the example above. This will work out very well if all speakers are sensitive to context all the time. If every speaker detests repetition of a consonant, they would avoid using the same consonant as one of the consonants in the base when they inserted a consonant in the reduplicant. However, things are not that straightforward, and back in Table 2 we saw that the speakers tend to have their own preferred segments in consonant insertion, sensitive or insensitive to context (Wedel 1999, Yu 1999, An 2012). If a speaker, who normally prefers *t* in consonant insertion, is not sensitive to context and does not pay attention to what there is already in context, then s/he would epenthesize *t* in spite of existing *t* in context. In this case, we can make use of a higher-ranked constraint like REPEAT(t), which is from a family of REPEAT constraints, to counteract *REPEAT(t) and stick to *t* regardless of context. In this regard, we cannot be satisfied with a wholesale grammar; rather, we need to consider individual speakers' preferences.

Consequently, what we need is a stochastic grammar that can capture individual preferences, as well. In the following section, I provide the grammars that I acquired as a result of applying the learning data (from the dictionary and from the word creation experiment) to the GLA, an OT-based stochastic model.

3.3 The GLA-based analysis

In this section I will see how well the GLA can perform in generating the data based on its stochastic learning algorithm. First, I made an input file which contains multiple tableaux with underlying forms, output candidates and their frequencies in number, relevant constraints, and constraint violations marked. I used the learning data that are from the dictionary data and from the experiment which contains at least two distinct groups of speakers, *t*-dominant group and *ʧ*-dominant group. A sample of partial input files for learning data is provided in Appendix A.

With regard to the data from the experiment, there were three input files altogether: the entire data of responses, the data from an individual speaker who preferred *t* irrespective of another *t* in context (context-insensitive speaker), and the data from an individual speaker who preferred *t* but avoided inserting *t* when *t* exists in context (context-sensitive speaker).

The learning datum from the dictionary consists of 51 reduplicative forms with VCVC-bases and CI = {*t*, *p*, *ʧ*}. There were 204 underlying/surface pairs presented with 4 output candidates per underlying form in the input file[2]. The learning datum from the experiment was also limited to the forms with VCVC-bases and CI = {*t*, *p*, *ʧ*}. The input file for this datum includes 444 underlying/surface pairs with 4 output candidates per underlying form (111 stimuli provided in the experiment). The learning datum for an individual speaker (P15) who is not sensitive to context in the experiment consisted of 444 underlying/surface pairs with 4 output candidates per underlying form: the four output candidates were generated based on the speaker's

[2] An output candidate with no epenthetic C, e.g., [unuk-unuk] (/unuk/), was presented for all learning data in the input files.

CI choice range, {*t*, *ʧ*, *n*}. The learning datum for an individual speaker (P13) who is sensitive to context in the experiment consisted of 666 underlying/surface pairs with 6 output candidates per underlying form: the six output candidates were generated based on the speaker's CI choice range, {*t*, *p*, *k*, *s*, *m*}.

I examine the machine rankings for the CI-reduplication data in this section, to see how well they could capture the nature of the data. The machine ranking of the constraints is basically to be acquired by entering the input file with the relevant information as was described in the beginning of this section, and getting the machine (OTSoft) to learn the data via many cycles of learning. I put 1,000,000 cycles of learning, which is said to be reliable enough to assume that learning happens. The ranking as a result of learning the dictionary data are as follows:

Table 3. Machine ranking for the dictionary data

Constraint	Ranking Value
ONSET	110.000
*REPEAT(p)	110.000
*REPEAT(ʧ)	110.000
*AFFRICV	96.061
*COR	95.070
*LAB	94.930
*STOPV	93.939
ʧ	93.939
t	90.991
p	-63,219.881
*REPEAT(t)	-108,314.944

The ranking values for all constraints are set to 100 in the initial state grammar and they keep being adjusted while leaning the data through a myriad of cycles. Note that the two *REPEAT constraints, *REPEAT(p) and *REPEAT(ʧ), are undominated, whereas another *REPEAT constraint, *REPEAT(t), has almost an infinitely low value in ranking (all three *REPEAT constraints shaded dark gray): first, it means that avoidance of identical consonant are really at work for the dictionary data; second, among the three consonants, *t*, *p*, and *ʧ*, with high frequencies, repetition of *p* or *ʧ* are very unlikely to be tolerated. However, repetition of *t* can be tolerated fairly well, which makes *t* common and abundant as a CI[3]. In the meantime, the two segmental constraints, ʧ and t (shaded light gray) are even more highly ranked than the constraint, p, which predicts that it is more likely to have similar frequencies for *ʧ* and *t* as a CI, whereas *p* will be much less popular. This implies that although the three consonants {*t*, *p*, *ʧ*} were among the most frequent Cs in the CI frequencies for the dictionary data, and it is eventually *t* and *ʧ*, but not *p* which will be preferred in the data of CI-reduplication. This can tell us why we ended up with {*t*, *ʧ*} that are the most frequently inserted as a CI in the experiment with speakers.

The matchup between input frequency and generated frequency showed that all input forms were generated with more or less similar frequencies of CIs, {*t*, *p*, *ʧ*}, albeit the input frequency given only to a single output form. In Table 4, the input frequency of 1 is given to the form, alok-**t**alok, since it is the only form found in the dictionary data. However, the learning ended up with distribution in the generated frequencies among different forms, alok-**t**alok, alok-**p**alok, and alok-**ʧ**alok.

Table 4. Matchup to input frequencies: e.g., [alok-talok] 'mottled' (=1a, 5a)

/alok/	Input Fr.	Gen. Fr.	Input #	Gen. #
alok-talok	1.000	0.324	198882	324266
alok-alok	0.000	0.000		
alok-palok	0.000	0.420		419747
alok-ʧalok	0.000	0.256		255987

The mismatch between the learning and generated datum, and particularly the generated frequencies divided up for the three candidate CIs hint at a possibility of variation: i.e., {*t*, *p*, *ʧ*} serve as variants.

The grammar obtained from running the data from the experiment produced the following ranking values, in general:

[3] According to the ranking hierarchy, although I did not look into the *ʧ*-dominant group for the context-sensitivity, it seems that *ʧ* is much more sensitive to context – in terms of avoidance of repetition, than *t* is.

Table 5. Machine ranking for the experimental data

Constraint	Ranking Value
Onset	108.000
*Repeat(ʧ)	100.000
*Lab	97.711
*AffricV	96.564
ʧ	95.436
*StopV	95.436
*Repeat(t)	95.342
*Cor	94.289
p	94.289
t	94.275
*Repeat(p)	93.298

The constraint, *Repeat(ʧ) was destined to be inactive in this grammar (given the value of 100, which is an initial default value for a constraint; shaded cell in dark gray) due to the fact that none of the forms in the experiment had *ʧ* in context. The similar ranking values among the constraints in the middle of the table above (shaded light gray) indicate that there are chances of variation among the three consonants, *t*, *p*, and *ʧ*.

Table 6. Matchup to input frequencies: e.g., [amat-Camat]

/amat/	Input Fr.	Gen. Fr.	Input #	Gen. #
amat-tamat	0.667	0.365	69877	365455
amat-amat	0.000	0.000		
amat-ʧamat	0.333	0.474	34378	474438
amat-pamat	0.000	0.160		160107

An example in the learning datum shows that even though the input frequency did not indicate any occurrence of *p* as a CI, the generated frequency came to have some frequency for *p*. In addition, the input frequency for *t*-inserted form, twice as high as that for *ʧ*-inserted form, decreased about the half, resulting in similar frequencies between the *t*-inserted form and the *ʧ*-inserted form in the generated frequency. This is related to the frequency distribution of CIs in the experiment, in which *t* and *ʧ* were the most frequently inserted Cs, among others.

The following is the grammar learned by the model when the learning datum is from a speaker who generally inserts *t* with no consideration of context. Thus this speaker uses his or her preferred C in consonant insertion even if the context has the same C already.

Table 7. Machine ranking for the data by a speaker who is *not* sensitive to context (e.g., P15, whose CI = {*t*, *ʧ*, *n*})

Constraint	Ranking Value
Onset	110.000
*Lab	100.000
*Repeat(ʧ)	100.000
*AffricV	97.356
*NasV	96.881
*StopV	95.763
*Repeat(t)	95.540
t	94.237
ʧ	92.644
*Cor	90.000
p	90.000

The ranking values for the constraints, *Lab, *Repeat(ʧ), *Cor, p indicate that these constraints are inactive: *Lab and p are irrelevant since the speaker did not insert *p* at all in his or her outputs; *Repeat(ʧ) is irrelevant since *ʧ* was not given in context, at all; and *Cor is irrelevant since all consonants that this speaker epenthesized in his or her outputs were coronals, {*t*, *ʧ*, *n*}.

The values for *Repeat(t) and t are close to each other, which suggests that *t* will be inserted without much consideration of whether *t* exists in context. Another running of the data can possibly invert the ranking between them, and the generated frequencies generally showed a strong tendency to prefer *t*, regardless of context as in the following:

Table 8. Matchup to input frequencies: e.g., [asam-Casam]

/asam/	Input Fr.	Gen. Fr.	Input #	Gen. #
asam-nasam	1.000	0.229	90908	228921
asam-sasam	0.000	0.000		
asam-tasam	0.000	0.584		584473
asam-ʧasam	0.000	0.187		186606

Although the speaker actually chose *n* as a CI for the base form of /asam/, which is shown in the input frequency, the GLA recognized the speaker's general tendency to go for *t*, which was learned through the learning datum, and this tendency is shown in the generated frequency.

The grammar resulting from running the GLA for an individual speaker who is sensitive to context is quite different from that for the speaker who is not sensitive to context:

Table 9. Machine ranking for the data by a speaker who *is* sensitive to context (e.g., P13, whose CI = {*t*, *p*, *k*, *s*, *m*})

Constraint	Ranking Value
Onset	110.000
*Dor	99.969
*NasV	98.153
*Lab	97.138
*Repeat(t)	96.817
*FricV	96.781
*StopV	95.066
t	93.889
*Cor	92.893
p	91.014
ʧ	90.000
*Repeat(p)	-36,312.730

For this speaker, the ranking values between the two constraints, *Repeat(t) and t are much more apart than for the context-insensitive speaker:

Table 10. Machine rankings for a context-insensitive speaker vs. context-sensitive speaker

Context-insensitive speaker		Context-sensitive speaker	
Constraint	**Ranking Value**	**Constraint**	**Ranking Value**
ONSET	110.000	ONSET	110.000
*LAB	100.000	*DOR	99.969
*REPEAT(ʧ)	100.000	*NASV	98.153
*AFFRICV	97.356	*LAB	97.138
*NASV	96.881	*REPEAT(t)	96.817
*STOPV	95.763	*FRICV	96.781
*REPEAT(t)	95.540	*STOPV	95.066
t	94.237	t	93.889
ʧ	92.644	*COR	92.893
*COR	90.000	p	91.014
p	90.000	ʧ	90.000
		*REPEAT(p)	-36,312.730

The bigger difference in the ranking values between *Repeat(t) and t, for this context-sensitive speaker, suggests that their hierarchical relationship is robust enough not to be inverted in any trials of the grammar. Therefore, *t* is normally preferred as a CI for this speaker, but an existing *t* in context will prohibit the insertion of *t*.

Table 11. Matchup to input frequencies: e.g., [akan-Cakan]

/akan/	Input Fr.	Gen. Fr.	Input #	Gen. #
akan-takan	1.000	0.700	91229	699841
akan-akan	0.000	0.000		
akan-sakan	0.000	0.220		220145
akan-pakan	0.000	0.043		42520
akan-makan	0.000	0.034		34422
akan-kakan	0.000	0.003		3072

The generated frequency does not match the input frequency perfectly well; however, it can show a general tendency to prefer *t* as in Table 10. This tendency gets weakened when *t* exists in context as in the following table:

Table 12. Matchup to input frequencies: e.g., [itip-Citip]

/itip/	Input Fr.	Gen. Fr.	Input #	Gen. #
itip-sitip	1.000	0.405	89610	405086
itip-itip	0.000	0.000		
itip-titip	0.000	0.315		314977
itip-pitip	0.000	0.215		214548
itip-mitip	0.000	0.041		40883
itip-kitip	0.000	0.025		24506

The speaker chose *s* for this specific input, and it does not show his or her overall tendency for inserting Cs in the CI-reduplication. However, the machine learning algorithm could capture the general tendency via the leaning datum of this speaker's. That is, this speaker, who is sensitive to context, does not like repetition of consonants; thus s/he usually inserts *t* but opts for another C when s/he encounters another *t* in context (cf. generated frequencies in Table 11 vs. Table 12).

In all these grammars learned and generated by the GLA, through the learning data of the dictionary and the experiment (overall grammar, individual grammars (context-insensitive and context-sensitive)), an overall tendency of variation could be captured; however, the generated frequencies did not match the input frequencies to full extent. This may be partly because a complete set of constraints was not provided and/or partly because the number of times to go through underlying forms was not enough for the machine to come up with a perfect grammar[4]. It can be also due to some other factor innate to the algorithm itself, which is to be discussed in the next section.

[4] It is not likely that it is due to less than enough number of times to run underlying forms through the grammar: it was recommended to run through forms a million times, and I used ten million for the number. As for the other possible cause, incomplete set of constraints, it is plausible that there might be some other constraints involved.

4. Concluding remarks

We could see that the GLA, which I happened to adopt for my purpose of learning a grammar, can handle data which contain variation: as was seen in the preceding section, the GLA could generate differentiated frequencies for variants. That is, the GLA works well for those cases that have variation in the output for a single input.

We also saw that the GLA appears not to work well for the cases in which any given input is categorically mapped onto the same output. Hence we still came to have variable output forms based on generated frequencies, even for the dictionary data and the experimental results where an input came with an output with no variation. Many linguists are aware of this problem of "equating variation with gradience (Coetzee p.c.)". After all, these two concepts, variation and gradience, are not the same, and an OT-based stochastic model like the GLA cannot deal with gradient well-formedness in contexts without variation, although it is good at handling data which have variable outputs. There have been alternative approaches to better handling the data with gradient acceptability, e.g., Coetzee and Pater 2008 (based on weighted constraints of Harmonic Grammar), Hayes and Wilson 2008 (maximum entropy model with weighted constraints)[5] *inter alios*.

I do not attempt to find fault with the learning algorithm I utilized in this discussion; nor do I intend to repair the problem found with the algorithm. Rather, it suffices to realize that it was due to the nature of the algorithm *per se* that a correct grammar cannot be reached for categorical mappings between an input and an output. It also suffices to learn that gradient well-formedness, as well as variation, can be captured by a better developed learning model. Therefore, the data of CI-reduplication, laden with variation and gradience, can be learned; that is, the data can be handled by grammar.

[5] Refer to Jun (2015) for the learning simulation of Korean n-insertion via the maxent grammar.

Appendix A. Sample Input Files

The input files I used for the learning of the dictionary data and the experimental responses (for an overall grammar, for an individual grammar that is not sensitive to context, and for an individual grammar that is sensitive to context) are provided below: only some portion of each data file has been given in the interest of space.

Input file from the dictionary data:

			Onset	*Lab	*Cor	*AffricV	*StopV	t	ʧ	p	*Repeat(p)	*Repeat(ʧ)	*Repeat(t)
			Onset	*Lab	*Cor	*AffricV	*StopV	t	ʧ	p	*Repeat(p)	*Repeat(ʧ)	*Repeat(t)
ollok	ollok-ollok		2										
	ollok-**p**ollok	1	1	1			1	1	1				
	ollok-**t**ollok		1		1		1		1	1			
	ollok-**ʧ**ollok		1		1	1		1		1			
ulluk	ulluk-ulluk		2										
	ulluk-**p**ulluk	1	1	1			1	1	1				
	ulluk-**t**ulluk		1		1		1		1	1			
	ulluk-**ʧ**ulluk		1		1	1		1		1			
ulak	ulak-ulak		2										
	ulak-**p**ulak	1	1	1			1	1	1				
	ulak-**t**ulak		1		1		1		1	1			
	ulak-**ʧ**ulak		1		1	1		1		1			
alak	alak-alak		2										
	alak-**p**alak	1	1	1			1	1	1				
	alak-**t**alak		1		1		1		1	1			
	alak-**ʧ**alak		1		1	1		1		1			

Input file for the Experiment data:

			Onset	*Lab	*Cor	*AffricV	*StopV	t	ʧ	p	*Repeat(p)	*Repeat(ʧ)	*Repeat(t)
			Onset	*Lab	*Cor	*AffricV	*StopV	t	ʧ	p	*Repeat(p)	*Repeat(ʧ)	*Repeat(t)
akam	akam-akam		2										
	akam-takam	7	1		1		1		1	1			
	akam-ʧakam	3	1		1	1		1		1			
	akam-pakam	1	1	1			1	1	1				
akan	akan-akan		2										

	akan- takan	6	1		1		1		1	1			
	akan- ʧakan	5	1		1	1		1		1			
	akan- pakan	1	1	1			1	1	1				
akaŋ	akaŋ- akaŋ		2										
	akaŋ- takaŋ	3	1		1		1		1	1			
	akaŋ- ʧakaŋ	8	1		1	1		1		1			
	akaŋ- pakaŋ	1	1	1			1	1	1				
akap	akap- akap		2										
	akap- takap	7	1		1		1		1	1			
	akap- ʧakap	3	1		1	1		1		1			
	akap- pakap	1	1	1			1	1	1				

Input file based on the experiment responses:

			Onset	*Lab	*Cor	*AffricV	*StopV	t	ʧ	p	*Repeat(p)	*Repeat(ʧ)	*Repeat(t)
			Onset	*Lab	*Cor	*AffricV	*StopV	t	ʧ	p	*Repeat(p)	*Repeat(ʧ)	*Repeat(t)
akam	akam- akam		2										
	akam- takam	7	1		1		1		1	1			
	akam- ʧakam	3	1		1	1		1		1			
	akam- pakam	1	1	1			1	1	1				
akan	akan- akan		2										
	akan- takan	6	1		1		1		1	1			
	akan- ʧakan	5	1		1	1		1		1			
	akan- pakan	1	1	1			1	1	1				
akaŋ	akaŋ- akaŋ		2										
	akaŋ- takaŋ	3	1		1		1		1	1			
	akaŋ- ʧakaŋ	8	1		1	1		1		1			
	akaŋ- pakaŋ	1	1	1			1	1	1				
akap	akap- akap		2										
	akap- takap	7	1		1		1		1	1			
	akap- ʧakap	3	1		1	1		1		1			
	akap- pakap	1	1	1			1	1	1				

REFERENCES

ALDERETE, JOHN, JILL BECKMAN, LAURA BENUA, AMALIA GNANADESIKAN, JOHN MCCARTHY and SUZANNE URBANCZYK. 1999. Reduplication with fixed segmentism. *Linguistic Inquiry* 30.3, 327-364.

AN, YOUNG-RAN. 2012. Identity avoidance and speaker preference in Korean. *Studies in Phonetics, Phonology and Morphology* 18.3, 397-412. The Phonology-Morphology Circle of Korea.

_______________. 2013. Gradience in the Korean reduplication. *Korean Journal of Linguistics* 38.4, 921-943. The Linguistic Society of Korea.

BOERSMA, PAUL. 1997. How we can learn variation, optionality, and probability. Ms. University of Amsterdam.

BOERSMA, PAUL and BRUCE HAYES. 2001. Empirical tests of the Gradual Learning Algorithm. *Linguistic Inquiry* 32, 45-86.

CLEMENTS, GEORGE N. and SAMUEL J. KEYSER. 1983. *CV Phonology*. Cambridge, MA: MIT Press.

COETZEE, ANDRIES W. and JOE PATER. 2008. Weighted constraints and gradient restrictions on place co-occurrence in Muna and Arabic. *Natural Language and Linguistic Theory* 26, 289-337.

ESSENCE KOREAN DICTIONARY [EYSSEYNS KWUKE SACEN]. 2006. Phacwu, Korea: Mincwungselim Co.

GOLDSMITH, JOHN. 1976. *Autosegmental Phonology*. PhD Dissertation. MIT.

HAYES, BRUE. 2010. *OTSoft: Optimality Theory Software* (Version 2.3) [Computer program] http://www.linguistics.ucla.edu/people/hayes/ otsoft/.

HAYES, BRUCE and COLIN WILSON. 2008. A maximum entropy model of phonotactics and phonotactic learning. *Linguistic Inquiry* 39, 379-440.

JUN, JONGHO. 2015. Korean n-insertion: A mismatch between data and learning. *Phonology* 32, 417-458.

JUN, JUNGHO and HYEMIN LEE. 2006. Hankwue pwupwuncwungchepeyseui kapyencek cepsa (Variable affix position in Korean partial reduplication). *Studies in Phonetics, Phonology, and Morphology* 12.1, 149-159. The Phonology-Morphology Circle of Korea.

LEBEN, WILLIAM. 1973. *Suprasegmental Phonology*. PhD Dissertation. MIT.

LOMBARDI, LINDA. 2002. Coronal epenthesis and markedness. *Phonology* 19, 219-251.

MCCARTHY, JOHN J. 1979. *Formal Problems in Semitic Phonology and Morphology*. PhD Dissertation. MIT.

________________. 1981. A prosodic theory of non-concatenative morphology. *Linguistic Inquiry* 12, 373-418.

________________. 1986. OCP effects: Gemination and antigemination. *Linguistic Inquiry* 17, 207-263.

PRINCE, ALAN and PAUL SMOLENSKY. 1993. *Optimality Theory: Constraint Interaction in Generative Grammar*. MIT Press.

SMITH, JENNIFER L. 2003. Onset sonority constraints and subsyllabic structure. Revised version of paper presented at the 9th International Phonology Meeting. University of Vienna, 2002. Rutgers Optimality Archive #608.

STERIADE, DONCA. 1982. *Greek Prosodies and the Nature of Syllabification*. PhD Dissertation. MIT.

WEDEL, ANDREW. 1999. Turkish emphatic reduplication. Ms. Linguistics Research Center: Phonology at Santa Cruz.

YIP, MOIRA. 1988. The Obligatory Contour Principle and phonological rules: A loss of identity. *Linguistic Inquiry* 19, 65-100.

__________ . 1995. Repetition and its avoidance: The case of Javanese. In K. Suzuki and D. Elzinga (eds.). *Proceedings of South Western Optimality Theory Workshop 1995 Arizona Phonology Conference Vol 5: U. Of Arizona, Department of Linguistics Coyote Papers,* 238-262. Tucson AZ.

__________. 1998. Identity avoidance in phonology and morphology, In Steven G. Lapointe, Diane K. Brentari and Patrik M. Farrell (eds.). *Morphology and its Relation to Phonology and Syntax*, 216-246. Stanford: CSLI.

YU, ALAN. 1999. Dissimilation and allomorphy: The case of Turkish emphatic reduplication. Ms. University of California, Berkeley.

Young-ran An
Department of English
KC University
24gil 47 Kkachisan-ro, Gangseo-gu
Seoul 07661, Korea
e-mail: yyrran@kcu.ac.kr

received: July 30, 2016
revised: October 22, 2016
accepted: November 3, 2016