Mechanical Turkish

Veltzer Doron

Department of Linguistics

Tel Aviv University

Ramat Aviv, Tel Aviv, Israel

November 4, 2016

Abstract

This paper proposes Recursive Neural Networks (RNNs) as phonological models. In order to demonstrate their effectiveness I revisit Becker (2009) (and Becker et al. (2011)) summarize its OT account of Turkish's stem final voicing alternations and criticize it on the grounds of implausible learnability, I then show how RNN structure based models would handle the same phenomenon in a simpler and more learnable manner ending in displaying results of an RNN topology used to model the phenomenon tracing its development motivation to such innate facts as the temporal nature of speech and the articulator.

Contents

1 'Surfeit of the stimulus' revisited **3**

1.1 The phonetic phenomenon at hand 3

1.2 Analysis within OT 6

1.3 Critique 7

2 Phonological neural network (NN) modelling **9**

2.1 Some technical details 10

2.2 Weakening NNs' powers of generalization 12

2.2.1 Experiment 1: deep NN modelling 12

2.2.2 Experiment 2: Simple RNN modelling 13

2.2.3 Experiment 3: Final RNN modelling 14

2.2.4 Experiment 3: Results 16

3 Conclusion **19**

A Appendix A - On the history of Machine Learning (ML) **20**

A.1 Deep networks as automatic learners of constraints 21

A.2 RNNs as a model for finding structure in time 22

A.3 On the psychology of the phonologist & economics 23

B Appendix B - Previous uses of NNs to model phonology **24**

1 'Surfeit of the stimulus' revisited

Turkish is an SOV agglutinative Altaic language that has come a long way from the Mongolian steppes, in it's passage it has come into extensive language contact with dominant representatives of two other language families, namely Persian (IE) and Arabic (Semitic).

The agglutinative nature of Turkish means that suffixes are omnipresent and morpho-phonological phenomena centers around the points of agglutination, here I survey one of the simplest morphemes, the accusative form, a high vowel (marked /I/) suffix under-specified for back and round features. The flagship of Turkish phonology Vowel Harmony (VH) then causes the suffix to harmonize its missing features. we disregard VH here and focus instead on the consonants p,t,t͡ʃ,k at the endings of stems.

1.1 The phonetic phenomenon at hand

The accusative form frequently causes voicing alternation, this has traditionally been analyzed as final devoicing in the stem's final consonant rather than voicing in the accusative form (in which case the UR is assumed to be voiced), since we examine the task of deducing the accusative form given the stem, we follow Becker et al. (2011) in disregarding the traditional analysis and assume the final consonant's unvoiced PR to be the UR input for the accusative form.

The alternation is lexically specified and largely unpredictable environment wise (unless that environment encompasses the entire stem). Observe the extreme case of examples [amat͡ʃ] 'purpose' [amat͡ʃɯ] (+acc) and [anat͡ʃ] 'mother hen' [anad͡ʒɯ] (+acc). That said, the alternation does however correlate strongly to certain features of the alternating consonant and its nearby

environment.

In Becker (2009), results by Moreton (2008) are reasserted in Turkish. While inspecting the Turkish lexicon as gleaned from Berkeley's TELL database (Inkelas et al. (2000)) correlations between accusative voicing alternation of final consonants and various environment aspects are found statistically significant and thus predictive of the phenomena[1]:

- The stem length greatly increases alternations, three C final stem forms were analyzed, CVC, CVCC & CVCVC, (longer than that the alternation seems to tend to 100% in an oscillating manner).
- The Place Of Articulation (POA) of the alternating consonant affects alternations forming a U shape as a function of backwardness.
- High vowels preceding alternating consonants increase alternations.
- Back vowels preceding alternating consonants increase alternations.
- Voice for preceding consonants do not have any significant effect.

In order to examine these in terms of speaker productivity Istanbul Turkish speakers were asked to apply the accusative form to nonce words, the results showed that speakers had productive knowledge as to the mutual correlations of length and POA effects (figure 1), but little to no (to opposite) knowledge with respect to qualities of the preceding vowel (figures 2 & 3).

This result is in line with cross linguistic universal language tendencies. Length and POA effects are prevalent across many languages whereas vowel quality effects are extremely rare. In those rare cases where languages exhibit

[1] All of the following also have non trivial mutual correlations (i.e. in order to predict the probability of alternation in the lexicon it is beneficial to model all correlating parameters together rather than one at a time)

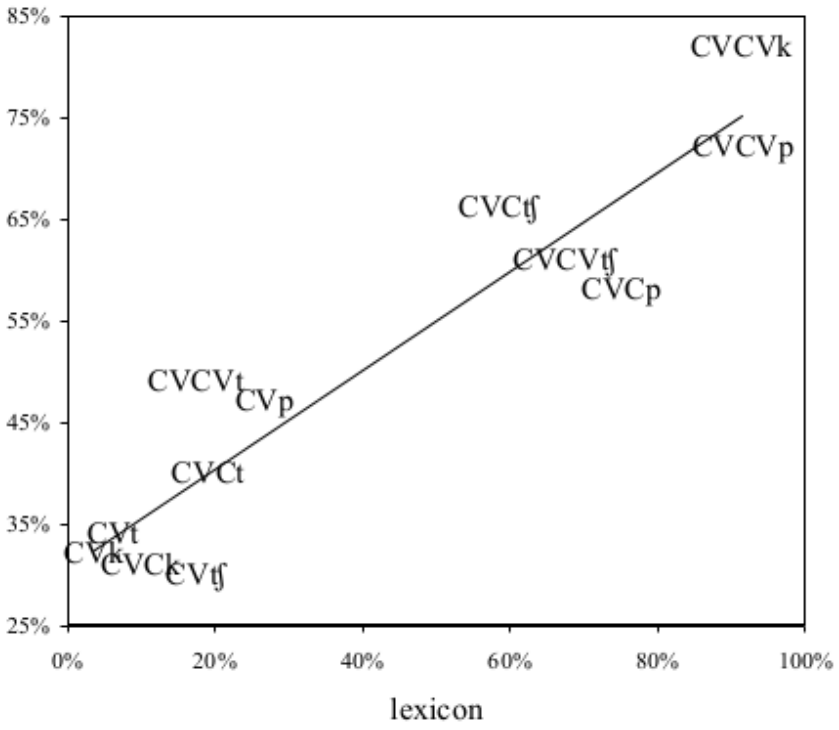

Figure 1: Speaker production rates plotted and regressed against tendencies in the lexicon. Data points relate to stem length and alternating consonants POA.

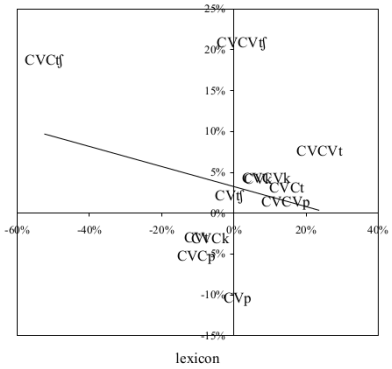

Figure 2: Speaker alternation differences between high preceding vowels plotted and regressed against tendencies in the lexicon. Different data points relate to stem length and alternating consonant's POA.

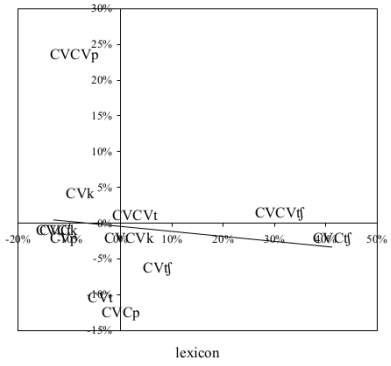

Figure 3: Speaker alternation differences between back preceding vowels plotted and regressed against tendencies in the lexicon. Different data points relate to stem length and alternating consonant's POA.

effects of preceding vowel quality it is usually attributed to perception and has a reversed correlation trend to the one present in the Turkish lexicon.

Existence of the trend itself is attributed to Arabic loan words as Becker (2009) explains: "In Turkish borrowings of words with voiced stops in the source language, final devoicing in the bare stem but not in the forms with vowel-initial suffixes causes nouns to become alternating (e.g. Arabic burdʒ 'sign' ¿ Turkish burtʃ burdʒu), whereas source words that end in a voiceless stop don't alternate across the paradigm. Arabic lacks the consonants [p] and [t͡ʃ] and has many nouns that end in [b] and [d͡ʒ], and as consequence the lexicons overall alternation rates are boosted for those places of articulation. On the other hand, the existence of many Arabic nouns with feminine suffix -at/-et boosted the number of non-alternating, non-high vowel, coronal-final nouns. Ultimately, however, the historical explanation for these lexical trends is obviously completely inaccessible to speakers who are not experts in historical linguistics."

1.2 Analysis within OT

Becker (2009) then offers an OT model for this seemingly selective statistical learning phenomenon by making use of the mechanisms of Constraint Cloning (CC) and Recursive Constraint Demotion (RCD). Briefly, CC operates so that whenever learners detect a grammar conflict between a pair of surface forms they clone the constraint that caused the conflict (choosing which constraint to clone is somewhat more complicated than might appear at first glance, for details see Pater (2007)). After cloning, the two constraints subdivide the lexicon between them with indexed UR relating the stem to it's delegated constraints, this is repeated until the behavior of speakers is modeled. I assume the reader to be acquainted with RCD (for details see Tesar (1997)).

The relevant OT constraints cloned in this analysis are IDENT(voice) and

MAX producing 2 clones for each combination of the examined stem length and POA, i.e. 2*3*4 constraints that partition the lexicon.

The unlearned relatedness to preceding vowel quality is explained away on the other hand by simply claiming no markedness constraints exist to correlate preceding vowel quality and voicing ($*V_{[+HIGH]}C_{[+VOICE]}$ or $*V_{[+BACK]}C_{[+VOICE]}$), not allowing learners to learn such correlations.

The work ends by reasserting that when exposed to the lexicon a general statistical learner, such as the Minimal Generalization Learner (MGL) whose results are presented, will learn all available correlations including the surfeit and in doing so will greedily overshoot the mark and "outperform" native speakers[2].

1.3 Critique

First for a trite. In the vowel quality height production test (figure 2) it is claimed that the [tʃ] and [p] consonants cannot be considered outliers since they include too many of the alternations, that however does not deny the possibility that their behaviors are the results of a different strata at work [3]. As stated, both these consonants usually originate stem finally from Arabic load words, additional evidence for the operation of a second stratum in Arabic loan words can be found in the fact that Arabic loan words often do not comply with VH and are easily identifiable by speakers[4].

[2]This in turn is used to suggest that speakers are not full statistical learners but rather learn correlations when those are in line with UG

[3]The outlier comment was made in the first place since with these points removed a near perfect 1-1 regression line would be made quite apparently

[4]It remains to be lamented that this phase of the experiment not carried out using child speakers (as were all other following parts of the experiment) since it would have ascertained if less knowledge of word origin produces more production errors.

This critique doesn't hold for the backwardness test (figure 3) since that test clearly demonstrates no speaker reproduction ability. For the sake of simplicity, from now on, I follow Becker in claiming that no speaker reproduction of preceding vowel quality effects exist.

The second point to be raised is a much more problematic one, namely, that of learnability. How can one entertain the learning process involved in a constraint cloning process? How can the speaker hold several simultaneous constraint EVAL derivations when the production mechanism itself is singular? Such a double learning mechanism would require at least some form of neural structure cloning which is extremely unlikely, How can the speaker then compare the two derivations side by side and detect the correct cause of the conflict? (remembering that this is not at all a trivial process). This follows explanations of variation (such as Coetzee (2008)) in positing the ability of an online learning process to keep two derivations side by side, but whereas those presuppose an inner homunculus, CC exaggerates in presupposing a linguistically inclined inner homunculus complete with pen and paper.

One might claim that this is an OT problem at the edge of phonology and one possible modelling of it at that but clashes between statistics and UG such as these are omnipresent and the given explanation is in fact inevitable. As a side note, I'll mention having tried to reformulate this within OT in a more plausible learnable manner and failed. Furthermore, up to some fundamental shift in OT I believe an explanation of this sort being inevitable could be formally proven.

One might also claim the model does not pertain to the actual learning or productive process, this is a claim much harder to rebuke but other problems exist. Out of the overall errors and variations produced by the two speakers

in the 6000 relevant stems in the TELL database 95% relate to our phenomena (the other mostly relate to the Arabic originating and much harder to transcribe double consonants) suggesting extreme learning difficulties even for native speakers in the lexicon proper, these same speakers exhibit a 94% agreement on alternations in lexical terms. CC would have an extremely hard time explaining such errors and productive variations in the lexicon.

In nonce words speaker agreement reaches for obvious reasons a much lower average of 58% (Pearson r correlation between two speakers' outputs is about 0.2 with a p value of around 0.3). CCs apply to nonce words by competing as to which is applied and hence operate with the frequency of their respective percent of the lexicon. Why then would such a mechanism prove confusing for speakers to use?

Last but not least is the fact that all speakers' nonce productions are lower than the matching percentage existent in the lexicon and while even child speakers exhibit the correct statistical patterns these patterns are attenuated so that children produce much less alternations overall than both adults and lexicon statistics.

All of this begs for an altogether different explanation.

2 Phonological neural network (NN) modelling

This paper on the one hand is quite unorthodox in trying to bridge the gap between Computational Linguistics and Natural Language Processing (NLP) but does not on the other hand have the scope needed in order to go in depth into the topic of Machine Learning (ML) NNs and more specifically deep learning and Recursive NNs (RNNs). Despite many of these models being

initially proposed for and inspired by phonology they have seen little to no use throughout the years in phonological modelling and are instead kept busy producing state of the art results cross the board in NLP solving problems higher up in the language tree, from morphology to sentiment analysis.

For a quick review of the history of machine learning, see appendix a. For a referential look at whatever work has been previously done in modelling phonology with NNs see appendix b. I kept the discourse in these appendices mostly referential and hopefully in par with the understanding levels of a budding NLP scientist and/or an interested phonological minded computational linguist and will henceforth limit my NLP outbursts in the work itself to topics directly related to and facilitating modelling of the phonological Turkish phenomenon.

2.1 Some technical details

Perusing the TELL lexicon I first qualitatively reaffirmed all the observations regarding the correlations. As for speaker productivity, I have not the means to reproduce the nonce word tests conducted on native Turkish speakers, so, taking these results as given fact all my tests involved teaching the lexicon to a learning neural net model and examining its effectiveness both in quickly memorizing the forms recorded in the TELL corpus and in reproducing the nonce accusative form statistics of Turkish speakers, i.e. reproducing correlations of stem length and POA while ignoring correlations of preceding Vowel quality.

The most crucial issue in learning is balancing memorization (over-fitting) and generalizing. NNs are deterministic machines, interestingly enough in ML as opposed to in linguistics variation is rarely as major a concern as accuracy, noise is thus not typically used to model variation but rather to

force the NN to generalize. The noise levels are set high enough to make it impossible for the NN to model the noise existing in the data itself (called over-fitting the error signal). Such noisy models with increased generalization cause quite a counter intuitive speedup in learning where the size of the learned corpus is rendered quite irrelevant, this is a well known fact in NLP and was also noticed and mentioned by Boersma and Pater (2008).

The nonce words for speaker productivity tests were carefully constructed by Turkish phonologists to "sound" Turkish, generating a large enough list proved problematic. As a result I had to generate enough "speakers" whose outputs I could average. Since NNs are deterministic machines for each utilized network topology I computed 100 random weight settings with different random seeds and the output of these I averaged to compute production statistics.

Since the TELL corpus contains output variation and disagreement between its two speakers (with agreement measured at about 94%) I provided the networks with a gold standard goal of around 96% accuracy for lexicon learning and removed all variation in the output, leaving a sensible margin so as not to force too exact a memorization of the data and instead solicit generalizing as much as possible.

For the networks' input, after culling the overall 6000 possible stems for duplicate forms and variation I removed all but the CVC, CVCC and CVCVC prosodic structure stems studied in the article, the final input thus consisted of a collection of 854 stem forms. Once these were set I used what is called a one hot representation, where each input phoneme is given as a long vector (with one dimension per language phoneme) with its proprietary dimension coordinate set to 1. The phonemes then enter an embedding layer which translates them to certain representations found beneficial by the network's

learning process for modelling the input to output mapping.

The output had two distinct setups described further on.

All models were built trained and tested using Chollet (2015) (under Theano Development Team (2016)). [5]

2.2 Weakening NNs' powers of generalization

NNs are as a whole generalizing machines and given sufficient neurons and structural connectivity will approximate any mapping. In ML, weakening a model's powers of generalization is, as far as I know, never an issue.

2.2.1 Experiment 1: deep NN modelling

Output of the following two experiments is a single bit denoting the alternation of the input stem.

First I let an NN with enough neurons, arranged in multiple layers, learn the alternation, the middle layers of such NNs are called hidden and the NNs themselves are dubbed deep NNs. These deep networks' learning process allow each of their neurons to deduce their local constraints in order to solve the global mapping problem induced by the given input output pairs.

The goal in this experiment was to show that such an NN would learn the lexicon and reproduce all existent statistical correlations but in fact for reasons explained further on, the NN never generalized well enough and instead simply memorized the lexicon, so that when it achieved the required 96%

[5]Not having the breadth necessary to fully disclose all NN modelling and results I instead verbally detail all results that promote the goal of relating the development process that went into engineering the topology of the network and will present only the results for the final structure, however, all csv data used for input & output as well as the full training and testing scripts and all results can be found at https://github.com/veltzerdoron/Mekanichal-Turkish

accuracy it processed nonce words in patterns that bared little resemblance to those produced by speakers.

2.2.2 Experiment 2: Simple RNN modelling

Next I shifted the network's topology into the time dimension by using an RNN structure, RNNs differ from NN models in that each layer projects not only to the next layer but also to itself in the next time phase (here, the next phoneme's processing step). This sort of network topology should echo in the mind of a phonologist structures and notions of auto-segmental phonology.

The serial time shift dramatically changed the probability distribution of the output and took care of two crucial points in the modelling:

- The strong signal in the output showing the importance of length for prediction meant that the RNN would model it automatically.
- By questioning the model only as to its last output, the network will put a strong attention emphasis[6] on the features of the last input, i.e. the alternating consonant.

Results showed that RNNs greatly facilitated learning the alternation, despite reducing the number of neurons in the network to a mere 80 they learned the alternations about 20 times faster than NNs with the same comparative degree of required lexicon accuracy. A strong indication of generalizing taking place. It took 5 hours to train the required 100 "speaker" models.

[6] A related problem that might arise from this emphasis is the possibility of it expanding to include the immediate environment of the last input, specifically the quality of the preceding vowel

Examining output statistics on lexicon stems, despite their being 96% accurate, the RNNs seemed to have a significantly harder time in correlating preceding vowels to alternation than learning the mutual POA/length correlations. When producing alternation outputs for nonce word this effect seemed to amplify giving first hints that the RNN was indeed a promising model.

Two other, perhaps related, interesting facts about the nonce word results are the fact that the entire POA length pattern was attenuated in a manner that placed the RNN model somewhere between adult and child speakers (and away from lexicon statistics) and the fact that the alternation of CVC stems was greatly underestimated.

My hypothesis relating these two facts is that since the lexicon alternation representation is built bottom up in treelike fashion, the short stems are first learned. Their alternation being as scant as it is (11%) allows a learner to entertain a null hypothesis, simply asserting alternations never occur, has an accuracy of 89%, with learning and exposure to longer and longer stems (almost always alternating) this null hypothesis slowly erodes. Null hypotheses are a common issue in ML usually mitigated by ways such as balancing the examples shown to the network in terms of the output. I did not test this hypothesis.

2.2.3 Experiment 3: Final RNN modelling

There are generally two ways in which a network's generalizing power can be weakened, one way is to overload its task with other simultaneous tasks, the other is structural modification (for instance the example we've already seen of using an RNN). The final model combines these two notions.

A network's topology might be altered more in the direction of the desired

output if for instance we would envision a system in which consonant features are processed in a totally separate path from vowel features such as height and backward positions no interaction could possible ensue [7] Although this might work well enough for the problem dealt with in the paper it is highly improbable as a solution that could later be extended to other problems.

However, the proposed thought experiment inspires a structural change motivated by auto-segmental phonology as well. Features tend to influence similar features in adjacent phonemes and/or other features in the same phoneme but to a much lesser extent other features in adjacent phonemes.

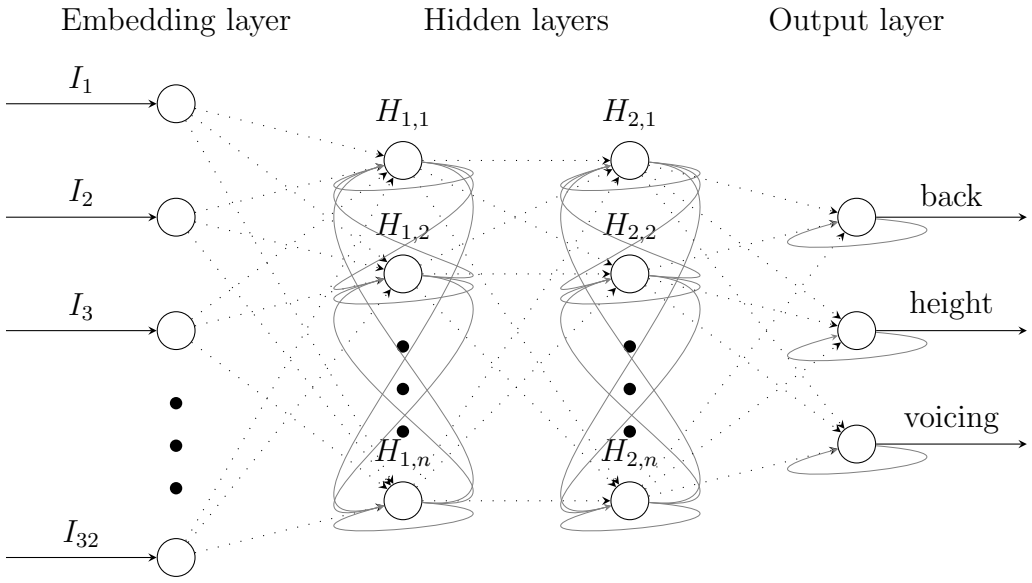

Figure 4: Diagram of the final network topology used. Shown is the input embedding layer with its 32 inputs (1 per Turkish phoneme) and the output layer's 3 neurons with their self recurrent connectivity and output features (back, height and vocality). Other layers are fully recurrent (all recursive connections drawn in gray)

In the proposed model, features start out mixed up in standard phonemic one hot input fashion and separate while processing takes place till they reach the end of processing and the last layer's 3 neurons, one for each of

[7]This extreme solution would require that the stems be memorized solely in terms reminiscent of a root based language

the output features (back, height & voicing). The output layer's recursive connections allow each of its neurons to take into account their own output in the previous time-step and/or the new phonemic input's representation as it is constrained by neurons in previous layers but not (and this is crucial) the previous time-step output of the other two feature neurons[8]. This causes the correlating of different features across time-steps to become more difficult.

If the outputs of the three neurons were to be seen as representing command lines going to the articulator organs, this recurrent severing could also be motivated by the fact that the tongue articulator must have a different command line than that of the vocal folds[9]

The output of this model was thus changed to include all three output features of the alternating consonant. For specific feature encodings, results by Stachowski (2015) were used. Overloading the task of alternation (or in this case output vocality) prediction with that of decoding and transmitting as is the back and height features of the input stem's last consonant, despite being as simple as it is, helps focus height and back features onto their own lines of control and dissuades the network to a large extent from letting them influence vocality.

2.2.4 Experiment 3: Results

For the final model I increased the number of neurons in the network's hidden layers and modified the recursive connectivity of the last layer to comply with diagram 4). Once this was done teaching the network became surprisingly easier and faster than it was to train the RNN in experiment 2 despite it having less neurons/parameters and its task being a proper subset of this

[8]The question whether this loop represents neuronal looping connections in the actual brain or abstracts a longer loop involving actual audio is left unanswered

[9]Predicting a more probable interaction between tongue height and backwardness

syl & cons		RNN	Lexicon	Child	Adult
CVC	k	3.00%	4.84%	0.00%	3.00%
	tʃ	20.00%	24.24%	3.00%	28.00%
	t	0.00%	7.81%	7.00%	6.00%
	p	43.00%	37.50%	17.00%	34.00%
CVCC	k	26.00%	11.11%		
	tʃ	53.00%	75.61%		
	t	42.00%	27.87%		
	p	99.00%	85.71%		
CVCVC	k	100.00%	94.71%	60.00%	95.00%
	tʃ	84.00%	81.67%	40.00%	53.00%
	t	25.00%	31.01%	10.00%	31.00%
	p	99.00%	97.37%	20.00%	53.00%

Figure 5: results of our modified RNNs averaged and split into categories according to POA and length as they compare to both lexicon statistics and Turkish speaker productions [10]

network's task. Training the 100 required final "speaker" models took about 4 hours.

The first thing to note is that the RNN has greatly "outperformed" native speakers in producing the mutual pattern for length and POA. A second thing worth mentioning is that alternation are still slightly underestimated with overall alternations standing at 43% instead of the lexicon's 52%, reasons for this were hinted at in experiment 2.

Let's now take a look at alternation differences as a function of preceding vowel qualities. As illustrated in 6 and 7 we see that the tendency between

[10]speaker productions were missing for stems of CVCC structure in Becker (2009)

the lexicon and our models seems to be reversed (as is the case for native speakers). The numbers themselves, however, seem to me to be quite arbitrary and rather than say the model managed to mimic speaker behavior I will be more careful and say that once previous vowel quality was barred from affecting alternation it became a variable free to be moved around by the other requirements from the network, especially since the truly divergent values were achieved for the consonants with a smaller number of input output stems. Perhaps if one was to overload the network with a more tasking goal such as modelling the phonemes for the entire output word and with more input/output pairs a result such as true speaker behaviour could be hoped for.

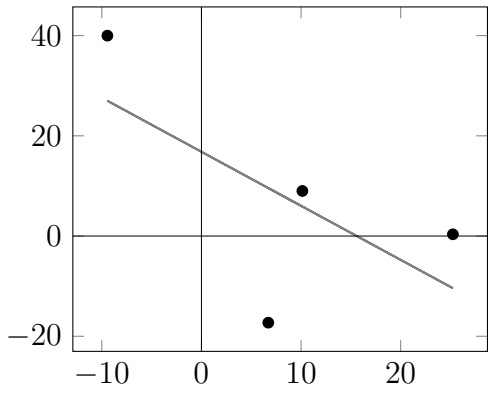

Figure 6: RNN alternation differences between high preceding vowels, plotted as a function of alternating consonant's POA and regressed against lexicon tendencies.

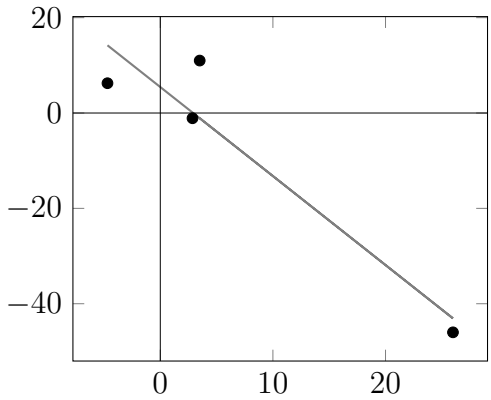

Figure 7: RNN alternation differences between back preceding vowels plotted and regressed against tendencies in the lexicon. Different points relate to alternating consonant's POA.

The given results do, however show that the final structure did achieve at least the goal of decoupling the influence of preceding vowel quality on alternation.

For completeness I also tried to teach the alternations to a network with the three neuron layer at its output layer without modifying its recursive connectivity, this structure created a bottleneck of three dimensional repre-

sentation and prevented learning altogether, indeed, it proved impossible to raise the network to the required accuracy no matter how the previous layers were boosted or how long training took place.

3 Conclusion

The paper has shown how a phenomena involving an amalgamation of statistical and universal behaviors can be effectively modeled step by step by using contemporary learning models such as RNNs. Starting with a model that performed general statistical learning and gradually working the universal limits into it by various methods. The actual engineering process involved many more experiments that proved to be dead ends and were thus not detailed here and yet the path finally followed had a compelling direct reasoning to it. Conveying the flavor of this sort of reasoning was a major goal in writing this work.

A Appendix A - On the history of Machine Learning (ML)

OT arose in the same years perceptron model (first proposed in Rosenblatt (1958)) based structures held sway in ML. Care, must be taken in order to avoid following the induced mixup in terms, the term constraints in OT parallels what are called features in the perceptron model (features perhaps being at the time taken up by phonological features). I took some care in using the terms constraints and features as they are used in phonology. Otherwise, from a strictly functional point of view, the similarity is uncanny.

Whether or not this resemblance is incidental or a result of the supportive theories arising in similar universities at the same time is besides the point, the farther OT went along in establishing itself as the go to phonological model the more similar it became to the perceptron model, this culminates in proofs for convergence of OT learners where an explicit reference to the proof being a reformulation of the perceptron convergence theorem is freely made by Boersma and Pater (2008).

Two points caused the downfall of perceptron based learning and the start of the last AI winter at the end of the last peak interest in Neural Networks (NNs) during the early 90s.

First, a simple proof made it abundantly clear that even basic functions such as an exclusive or (XOR) were unseparable and thus uncalculable by the model, such unseparable functions need to be encoded in the constraints themselves. This point, however, has no real relevance to OT since the term constraints itself implies that there is no case where a XOR function needs to be computed since violating two constraints always outweighs violating each of the constraints separately.

The second point, which is much more damning for OT, is that the resulting system is engineered rather then learned in the sense that constraints need to be precomputed and not derived by the ML system on its own. This engineered nature of the model has profound implications, it is not in itself a problem in case you assume innateness of the constraints in linguistics, or if you are trying to solve highly human engineered problems in Natural Language Processing (NLP) such as SPAM detection (a practical field where the perceptron model is still in play) but it is a problem if innateness is to be exchanged with universality or if an unsupervised ML solution is essential.

A.1 Deep networks as automatic learners of constraints

NNs can be seen as a natural solution when taking the perceptron model and applying it to a general problem of mapping inputs to outputs. An NN is composed of several connected neurons arranged in layers, each learning its weights when calculating its relative error in light of solving a general input to output mapping problem. Countless various algorithms exist and are applied for more effective learning with most deriving their basic notions from the Back Propagation (BP) algorithm developed in many stages (famously in Rumelhart et al. (1988)).

In cases where there are more than one layer of neurons (as in the case of the perceptron model) the networks are referred to as deep NNs and all layers not at the input or output layers are called hidden layers. The hidden layers are seen as searching for constraints that are worthwhile to compute locally in order to best construct the global mapping.

While writing this, deep NNs are probably the most actively studied topic in the world, it has been found to be highly effective for applications both in computer vision and in Natural Language Processing (NLP). Questions such

as how engineered perceptron constraints are embodied in deep networks when they approach the same problem, how one is to engineer a network's topology and how are the learned constraints mapped back onto the input level are on the agendas of both technological giants and academia luminaries.

A.2 RNNs as a model for finding structure in time

Recursive NNs (RNNs) are NNs that have backward projecting connections, i.e. networks where production involves connections from neurons further down the production line back to previous neurons or to themselves at a later time phase. Both testing and training such networks proceeds through a simulation of time phases and uses a process called Back Propagation Through Time (BPTT) reportedly developed independently three times (for instance by Mozer (1989)).

If a network is to predict the next phoneme in, say, language modelling, each neuron receives the last produced output in time and the set of features for the next phoneme and weighs both to produce the output for the next time sequence and so forth.

As early as they were introduced in Elman (1990) and Jordan and California Univ. (1986) RNNs seemed to be the correct way to represent the phonological serial to parallel to serial schisms the brain seems to be solving in its processing of time serialized phonetics. Due to technical computational and general loss of interest problems, this model was buried during the last AI winter and only resurfaced in full force when applied to economically motivated NLP problems higher up the language tree such as parsing, POS tagging, pragmatics and Sentiment Analysis (SA).

In all these fields more complex approaches such as bi directional LSTMs based on RNNs (but with added capability to hold memory for longer time

periods) currently produce state of the art results. Recently some attempts have been made to apply LSTMs to lower levels of the language tree such as for instance, morphology, where a bi-directional LSTM was taught to predict all morphological forms of ten languages). LSTM models, however, despite being much more effective than simple RNNs lack any remote biological precedent.

A.3 On the psychology of the phonologist & economics

Why then would such a model inspire OT as a model for phonology? Perhaps it was exactly its simplified and engineered nature which allowed it to become a useful tool in the hands of phonologists who were in turn surprised by such a simple first order approximation being more or less sufficient for explaining previously unexplained phenomena. For instance one clearly sees that the closer OT returns to its weighted ancestor the less clear its insightfulness and manipulability becomes.

A model's manipulability by phonologists and the understandability of its nature, useful as it may seem, cannot pose a criterion for validity. In the same way the economical applications of a system should not prove a significant dint in its academic usefulness.

Another explanation for the success of OT might lie in the fact that OT abides so well with the phonologist's intuitions themselves. In the same manner that the perceptron model is a perfect metaphor for the consciously mind when it contemplates possibile choice inputs from such unconscious parts of the mind as are the speech mechanisms, OT can provide a similar metaphor for the phonologist's conscious mind when it observes phonological phenomena and informs phonological action through the interaction between the conscious and subconscious parts of the brain by weighing constraints.

Phonology is probably the most subconscious (and thus the most mathematical) of all fields of linguistic inquiry. It does not have the same subversive effects of conscious attention emphasis pragmatics or even grammar has, as a result it should have went the same way the visual cortex went by having its function be the first to be successfully modelled using NNs. The fact that this was not the case could probably be explained by the fact that tasks such as sentiment analysis were found much more lucrative.

NLP being largely text oriented and audio applications being so mathematical in nature little treatment of the intervening space which is Phonology has peaked the interest of technological companies outside of academia. TTS & STT applications on the other hand seem as a whole inclined sadly to ignore this immense field of knowledge. Thus phonology, despite being the original inspiration for the RNN model, has seen very few attempts throughout the years of RNN modelling.

B Appendix B - Previous uses of NNs to model phonology

The few attempts that have been made to inquire into phonology using NN models, some of the prominent of which made oddly enough (and luckily enough) in modelling Turkish phonology, are as a whole incredibly instructive. The reason for Turkish being so prominent in these attempts becomes clear when one considers that other attempts involve modelling such languages as Hungarian. Apparently there is something inherent in VH languages which yields to a computational analysis, moreover the entire Turkish vowel system (basically a 3 bit harmonizing system) is quite compelling as a foundation to lay bricks upon.

Rodd (1997), for instance, built a language model of Turkish words where an RNN was tasked with the language modelling task of predicting the next phoneme, his attempts involved creating a bottleneck of representation in the middle hidden layer, starting at a two neuron bottleneck and moving up to five neurons. his two neuron setup learned the most prominent alternating regularity of any language, i.e. the regular alternation of consonants and vowels. When the hidden layer reached four neurons the network showed that it could produced regularities that exhibited some powers of Vowel Harmony alternation prediction, above that size the networks' outputs and behaviors were rendered quite uninterpretable (as they eventually always tend to do).

Stachowski (2015) developed a specific way to encode Turkish phonemes for application in general NN representation and showed that it improved on previous encoding when used to model the Turkish language, I scavenged this paper for some valuable bits of knowledge I used in the 3rd experiment.

A more theoretical basic approach was taken by Boersma et al. (2013) in modelling phonetic category creation, Boersma used a Boltzmann machine based bi directional clamping neural structure to show how categories could emerge by teaching it to relate audio signals and phonemic categories.

References

Becker, M. (2009). Phonological trends in the lexicon: The role of constraints. *Open Access Dissertations*, page 3.

Becker, M., Ketrez, N., and Nevins, A. (2011). The surfeit of the stimulus: Analytic biases filter lexical statistics in Turkish laryngeal alternations. *Language*, 87(1):84–125.

Blutner, R. and others (2009). Neural Networks, Penalty Logic and Optimality Theory. *ZASPiL Nr. 51September 2009*, page 53.

Boersma, P., Benders, T., and Seinhorst, K. (2013). Neural network models for phonology and phonetics. *Manuscript in preparation*.

Boersma, P. and Pater, J. (2008). Convergence properties of a gradual learning algorithm for Harmonic Grammar. *Rutgers Optimality Archive*, 970.

Chollet, F. (2015). *Keras*. GitHub.

Coetzee, A. W. (2008). Phonological variation and lexical frequency.

Elman, J. L. (1990). Finding Structure in Time. *Cognitive Science*, 14(2):179–211.

Inkelas, S., Kntay, A., Orgun, O., and Sprouse, R. (2000). Turkish electronic living lexicon (L). *Turkic Languages*, 4:253–275.

Jordan, M. I. and California Univ., San Diego, L. J. I. f. C. S. (1986). *Serial Order [microform] : A Parallel Distributed Processing Approach / Michael I. Jordan*. Distributed by ERIC Clearinghouse [Washington, D.C.].

Moreton, E. (2008). Analytic bias and phonological typology. *Phonology*, 25(01):83–127.

Mozer, M. C. (1989). A focused back-propagation algorithm for temporal pattern recognition. *Complex systems*, 3(4):349–381.

Pater, J. (2007). The locus of exceptionality: Morpheme-specific phonology as constraint indexation. *University of Massachusetts Occasional Papers in Linguistics 32: Papers in Optimality Theory III, Leah Bateman, Michael OKeefe, Ehren Reilly, & Adam Werle, eds., BookSurge Publishing*.

Rodd, J. M. (1997). Recurrent Neural-Network Learning of Phonological Regularities in Turkish. In *CoNLL*, pages 97–106.

Rosenblatt, F. (1958). The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review*, 65(6):386.

Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1988). Learning representations by back-propagating errors. *Cognitive modeling*, 5(3):1.

Stachowski, K. (2015). A phonological encoding of Turkish for neural networks. *Studia Linguistica Universitatis Iagellonicae Cracoviensis*, 129(4):363–372.

Tesar, B. (1997). Multi-recursive constraint demotion. *ROA-197*.

Theano Development Team (2016). Theano: A Python framework for fast computation of mathematical expressions. *arXiv e-prints*, abs/1605.02688.