Indicatements that character language models learn English morpho-syntactic units and regularities

Yova Kementchedjhieva
University of Copenhagen
`yova@di.ku.dk`

Adam Lopez
University of Edinburgh
`alopez@inf.ed.ac.uk`

Abstract

Character language models have access to surface morphological patterns, but it is not clear whether or *how* they learn abstract morphological regularities. We instrument a character language model with several probes, finding that it can develop a specific unit to identify word boundaries and, by extension, morpheme boundaries, which allows it to capture linguistic properties and regularities of these units. Our language model proves surprisingly good at identifying the selectional restrictions of English derivational morphemes, a task that requires both morphological and syntactic awareness. Thus we conclude that, when morphemes overlap extensively with the words of a language, a character language model can perform morphological abstraction.

1 Introduction

Character-level language models (Sutskever et al., 2011) are appealing because they enable open-vocabulary generation of language, and *conditional* character language models have now been convincingly used in speech recognition (Chan et al., 2016) and machine translation (Weiss et al., 2017; Lee et al., 2016; Chung et al., 2016). They succeed due to parameter-sharing between frequent, rare, and even unobserved training words, prompting claims that they learn morphosyntactic properties of words. For example, Chung et al. (2016) claim that character language models yield "better modelling [of] rare morphological variants" while Kim et al. (2016) claim that "Character-level models obviate the need for morphological tagging or manual feature engineering." But these claims of morphological awareness are backed more by intuition than direct empirical evidence. What do these models really learn about morphology? And, to the extent that they learn about morphology, *how* do they learn it?

Our goal is to shed light on these questions, and to that end, we study the behavior of a character-level language model (hereafter LM) applied to English. We observe that, when generating text, the LM applies certain morphological processes of English productively, i.e. in novel contexts (§3). This rather surprising finding suggests that the model can identify the morphemes relevant to these processes. An analysis of the LM's hidden units presents a possible explanation: there appears to be one particular unit that fires at morpheme and word boundaries (§4). Further experiments reveal that the LM learns morpheme boundaries through extrapolation from word boundaries (§5). In addition to morphology, the LM appears to encode syntactic information about words, i.e. their part of speech (§6). With access to both morphology and syntax, the model should also be able to learn linguistic phenomena at the intersection of the two domains, which we indeed find to be the case: the LM captures the (syntactic) selectional restrictions of English derivational morphemes, albeit with some incorrect generalizations (§7). The conclusions of this work can thus be summarized in two main points—a character-level language model can:

1. learn to identify linguistic units of higher order, such as morphemes and words.

2. learn some underlying linguistic properties and regularities of said units.

2 Language Modeling

The LM explored in this work is a 'wordless' character RNN with LSTM units (Karpathy, 2015).[1]

---

[1] Karpathy (2015) is a blog post that discusses exactly this model. We are unaware of scholarly publications that use this model in isolation, though it is used in several conditional models (Chan et al., 2016; Weiss et al., 2017; Lee et al., 2016; Chung et al., 2016), and it is similar to the character RNN

It is 'wordless' in the sense that input is not segmented into words, and spaces are treated just like any other character. This architecture allows for experiments on a subword, i.e. morphological level: we can feed a partial word and ask the model to complete it or record the probability the model assigns to an ending of our choice.

2.1 Formulation

At each timestep t, character c_t is projected into a high-dimensional space by a character embedding matrix $\mathbf{E} \in \mathbb{R}^{|V| \times d}$: $\mathbf{x}_{c_t} = \mathbf{E}^T \mathbf{v}_{c_t}$, where $|V|$ is the vocabulary of characters encountered in the training data, d is the dimension of the character embeddings and $\mathbf{v}_{c_t} \in \mathbb{R}^{|V|}$ is a one-hot vector with c_tth element set to 1 and all other elements set to zero.

The hidden state of the neural network is obtained as: $\mathbf{h}_t = \text{LSTM}(\mathbf{x}_{c_t}; \mathbf{h}_{t-1})$. This hidden state is followed by a linear transformation and a softmax function over all elements of V, which results in a probability distribution.

$$p(c_{t+1} = c \mid \mathbf{h}_t) = \text{softmax}(\mathbf{W}_o \mathbf{h}_t + \mathbf{b}_o)_i \quad \forall c \in V$$

where i is the index of c in V.

2.2 Training

The model was trained on a continuous stream of the first 7M character tokens from the English Wikipedia corpus. Data was not lowercased and it was randomly split into training (90%) and development (10%). Following a grid search over hyperparameters on a subset of the training data, we chose to use one layer with a hidden unit size of 256, a learning rate of 0.003, minibatch size of 50, and dropout rate of 0.2, applied to the input of the hidden layer.

3 The English dialect of a character LM

In an initial analysis of the learned model we studied text generated with the LM and found that it closely resembled English on the word level and, to some degree, on the level of syntax.

3.1 Words

When sampled, the LM generates real English words most of the time, and only about 1 in every 20 tokens is a nonce word.[2] Regular morpho-

---

of Sutskever et al. (2011), which uses a multiplicative RNN rather than an LSTM unit.

[2] As measured by checking whether the word appeared in the training data or in the *pyenchant* UK or US English dictionaries.

| sinding, fatities, complessed |
| breaked, indicatement |
| applie, therapie |
| knwotator, mindt, ouctromor |

Table 1: Nonce words generated with the LM through sampling.

| The novel regarded the modern Laboratory has a weaken-little director and many of them in 2012 to defeat in 1973 - or eviven of Artitagements. |

Table 2: A sentence generated with the LM through sampling.

logical patterns can be observed within some of these nonce words (Table 1). The words *sinding, fatities* and *complessed* all seem like well-formed inflected variants of English-looking words. The forms *breaked* and *indicatement* show productive morphological patterns of inflection and derivation being applied to bases of the correct syntactic class, namely verbs. It happens that *break* is an irregular verb and *indicate* forms a noun with suffix *-ion* rather than *-ment*, but these are lexical rules that block the more regular inflectional and derivational rules the LM has applied. In addition to composing morphologically complex words, the LM also attempts to decompose them, as can be seen with the forms *therapie* and *applie*. Here the inflectional suffix *-s* has been dropped, but the orthographic change associated with it has not been successfully reversed. Not all nonce words generated by the LM can be explained in terms of morphological productivity, however: *knwotator, mindt*, and *ouctromor* don't resemble any real morphemes and don't follow English phonotactics. These forms may be highly improbable accidents of the sampling process.

3.2 Sentences

Consider the sentence in Table 2 generated with the LM through sampling. The sentence could not be considered fluent or grammatical: there are no clear dependencies between verbs, subjects, and objects; it contains the nonce word *eviven* and the novel and unlikely compound *weaken-little*. Yet, some short-distance syntactic regularities can be observed. Articles precede adjectives and nouns but not verbs, prepositions precede nouns, and particle *to* precedes a verb. On an even larger scale, the clause *the novel regarded the modern Labora-*

Unit	t_{-13} ... t
Punctuation	`r ( 1936-1939)`
	`chool in 1921.`
	`il 13 , 1813 .`
	`ified in 1901.`
	`_( 1993-1998 )`
Word	`'s predictions`
	`ral relativism`
	`_contributions`
	`_were contract`
	`at connections`
Latinate suffix	`ered in inform`
	`_the concentra`
	`ultural recrea`
	`_was accommoda`
	`_Reyes introdu`

Table 3: Top 5 Contexts for Three Units in the Network of the LM. The last character in each string marks the peak in activation.

tory has a weaken-little director is grammatical in terms of the order between parts of speech[3]. The sentence appears unnatural due to its odd semantics, but consider the following alternative choice of words for the same syntactic structure: *the man thought the modern laboratory has a weaken-little director*. This sentence sounds only marginally anomalous.

The predominantly well-formed output of the LM suggests that it is appropriate to further study the linguistic regularities learned by it.

4 Meaningful hidden units in the LM

The hidden units of the LM were analyzed by feeding the training data back into the system and tracking unit activations on each timestep, i.e. after every character. For each unit, the five inputs which triggered highest activation (highest absolute value) were recorded (Kadar et al., 2016). About 40 units exhibited patterns of activation that could be identified as meaningful with the human eye. We selected three of the more interesting units to briefly discuss here. Table 3 shows a list of the top five triggers for each of these units, together with up to 13 characters that preceded them, to put them in context. One unit, which we'll dub the *punctuation* unit, seems to respond to closing punctuation marks. Another, dubbed the *Latinate suffix* unit, appears to recognize contexts that are likely to precede suffix *-ion* and its variants, *-ation*, *-ction* and *-tion*.

---

[3]That is, assuming that *novel* is a noun in this context and *weaken-little* is an adjective, by analogy with its second base.
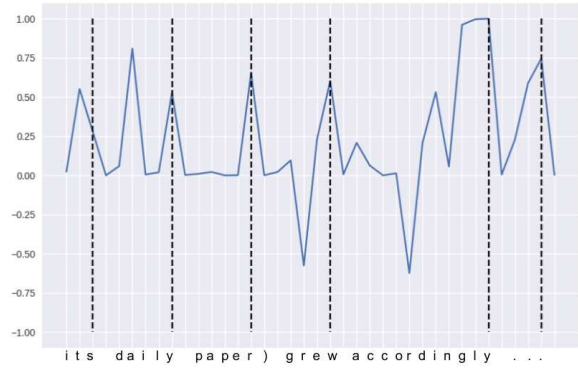

Figure 1: Activation of the *word* unit. Query: *its daily paper) grew accordingly*

4.1 The *word* unit

The most interesting unit, the *word* unit, appears to recognize complete words and sub-word units within them. Figure 1 shows the activation pattern of the *word* unit over a partial sentence from the training data. The dotted lines, which mark the end of tokens, often coincide with the peaks in activation. The unit also recognizes the base *it* within *its* and *according* within *accordingly*. The behavior of the unit could be explained either as a signal for the end of a familiar, repeated pattern, or as a predictor of a following space. The fact that we don't see a peak in activation at the right bracket symbol (which should be a cue for a following space) suggests that the former explanation is more plausible. Support for this idea comes from the low correlation coefficient between the activation of the unit and the probability the model assigns to a following space: only 0.08 across the entire training set. It appears that the LM knows that linguistic units need not occur on their own, i.e. that in a lot of cases a suffix is very likely to follow.

5 Morphemes encoded by the LM

Morphological segmentation aims to identify the boundaries in morphologically complex words, i.e. words consisting of multiple morphemes. A morpheme boundary could separate a base from an inflectional morpheme, e.g. *like+s* and *carrie+s*, a base from a derivational morpheme, e.g. *consider+able*, or two bases, e.g. *air+plane*. One approach to morphological segmentation is to see it as a sequence-labeling task, where words are processed one character at a time and every between-character position is considered a potential boundary. RNNs are particularly suitable for sequence

labeling and recent work on supervised morphological segmentation with RNNs shows promising results (Wang et al., 2016).

In this experiment we probe the LM using a model for morphological segmentation to test the extent to which the LM captures morphological regularities.

5.1 Formulation

At each timestep t, character c_t is projected into high-dimensional space:

$$\mathbf{x}_{c_t} = \mathbf{E}^T \mathbf{v}_i \qquad \mathbf{E} \in \mathbb{R}^{|V_{char}| \times d_{char}}$$

The hidden state of the encoder is obtained as before: $\mathbf{h}_t^{enc} = LSTM^{enc}(\mathbf{x}_{c_t}; \mathbf{h}_{t-1}^{enc})$. The hidden state of the decoder is then obtained as: $\mathbf{h}_t^{dec} = LSTM^{dec}(\mathbf{h}_t^{enc}; \mathbf{h}_{t-1}^{dec})$ and followed by a linear transformation and a softmax function over all elements in V_{lab}, which results in a probability distribution over labels.

$$p(l_t = l \mid \mathbf{c}, \mathbf{l}_{word}) = \text{softmax}(\mathbf{W}_o^{dec}\mathbf{h}_t^{dec} + \mathbf{b}_o^{dec})_i$$
$$\forall \quad l \in V_{lab}$$

where $\mathbf{l}_{word}$ refers to all previous labels for the current word and i refers to the index of l in V_{lab}.

The embedding matrix, $\mathbf{E}$ and $LSTM^{enc}$ weights are taken from the LM. Decoder weights and bias terms are learned during training.

5.2 Data

The model (hereafter referred to as C2M, character-to-morpheme) was trained on a combined set of gold standard (GS) segmentations from MorphoChallenge 2010 and Hutmegs 1.0 (free data). The data consisted of 2275 word forms; 90% were used for training and 10% for testing. For the purposes of meaningful LM embeddings, which are highly contextual, a past context is necessary. Since GS segmentations are available for words in isolation only, we extract contexts for every word from the Wiki data, taking the 15 word tokens that preceded the word on up to 15 of its appearances in the dump. The occurrence of a word in each of its contexts was then treated as a separate training instance.

5.3 Performance

The system achieved a rather low F1 score: 53.3. Compared to Ruokolainen et al. (2013), who obtain F1 score 86.5 with a bidirectional CRF model, C2M is clearly inferior. This is not particularly

Model	Precision	Recall	F1
C2M - WE	76.6	62.6	68.9
C2M - ¬WE	23.1	34.2	27.6
C2M - EOW	98.5	84.4	90.90
C2M - ¬PREF	53.6	59.2	56.3

Table 4: C2M Performance. WE stand for word edge, EOW for end of word, and PREF for prefix.

surprising given that the CRF makes predictions conditioned on past and future context, while C2M only has access to the past context. Recall also that the encoder of C2M shares the weights of the LM and is not fine-tuned for morphological segmentation. But taken as a probe of the LM's encoding, the F1 score of 53.3 suggests that this encoding still contains some information about morpheme boundaries. A breakdown of morphemic boundaries by type provides insights into the source of performance and limitations of C2M.

Potential word endings as cues for morpheme boundaries The results labeled *C2M - WE* and *C2M - ¬WE* in Table 4 refer to two types of morpheme boundaries: boundaries that could also be a word ending (WE), e.g. *drink+ing, agree+ment*, and boundaries that could not be a word ending (¬ WE), e.g. *dis+like, intens+ify*. It becomes apparent that a large portion of the correct segmentations produced by C2M can be attributed to an ability to recognize word endings. Earlier findings relating to the *word* unit of the LM (section 4.1) align with this line of argument: the unit indeed detects words, and those morphemes that resemble words. The sample segmentations in A and C of Table 5 can be straightforwardly explained in terms of transfer knowledge on word endings: *act, action* and *ant* are all words the LM has encountered during training. Notice that the morpheme *ant* has not been observed by C2M, i.e. it is not in the training data, but its status as a word is encoded by the LM.

Actual word endings An interesting result emerges when C2M's performance is tested on word-final characters, which by default should all be labeled as a morpheme boundary (*C2M - EOW* in Table 4). Recall that the rest of the results exclude these predictions, since morphological segmentation concerns word-internal morpheme boundaries. C2M performs extremely well at identifying actual word endings. The margin between *C2M - WE* results and *C2M - EOW* results is

	Input	True Segmentation	Predicted Segmentation	Correct
A.	actions	act+ion+s	act+ion+s	✓
B.	acquisition	acquisit+ion	acquisit+ion	✓
C.	antenna	antenna	ant+enna	
D.	included	in+clud+ed	in+clude+d	✓
E.	intensely	in+tense+ly	intense+ly	
F.	misunderstanding	mis+under+stand+ing	misunder+stand+ing	
G.	woodwork	wood+work	wood+work	✓

Table 5: Sample predictions of morphological segmentations.

substantial, even though both look at units of the same type, namely words. The higher accuracy in the EOW setting shows that the LM prefers to ends word where they actually end, rather than at earlier points that would have also allowed it. The LM thus appears to take into consideration context and what words would syntactically fit in it. Consider example E in Table 5. This instance of the word *in+tense+ly* occurred in the context of *the Himalayan regions of*. In this context C2M fails to predict a morpheme boundary, even though *in* is a very frequent word on its own. The LM may be aware that preposition *in* would not fit syntactically in the context, i.e. that the sequence *regions of in* is ungrammatical. It thus waits to see a longer sequence that would better fit the context, such as *intense* or *intensely*. Example D shows that C2M is indeed capable of segmenting prefix *in* in other contexts. The word *included* is preceded by the context *the English term propaganda*. This context allows a following preposition: *the English term propaganda in*, so the LM predicted that the word may end after just *in*, which allowed C2M to correctly predict a boundary after the prefix.

6 Parts of speech encoded by the LM

Part-of-speech (POS) tagging is also seen as a sequence labeling task because words can take on different parts of speech in different contexts. Access to the subword level can be highly beneficial to POS tagging, since the shape of words often reveals their syntax: words ending in *-ed*, for example, are much more often verbs or adjectives than nouns. Recent studies in the area of POS tagging demonstrate that processing input on a subword-level indeed boosts the performance of such systems (dos Santos and Zadrozny, 2014). Here we probe the LM with a POS-tagging model, hereafter C2T (character-to-tag). Its formulation is identical to that of C2M.

6.1 Data

We used the English UD corpus (with UD POS tags) with an 80-10-10 train-dev-test split. Training data for character-level prediction was created by pairing each character of a word with the POS tag of that word, e.g. 'like$_{VERB}$' was labeled as as $\langle VERB\ VERB\ VERB\ VERB \rangle$. UD doesn't specify a POS tag for the space character, so we used the generic X tag for it. Similarly to C2M, encodings for C2T were obtained for words in context.

6.2 Performance

C2T obtained an accuracy score of 78.85% on the character level and 87.06% on the word level, where word-level accuracy was measured by comparing the tag predicted for the last character of a word to the gold standard tag. The per-character score is naturally lower by a large margin, as predictions early on in the word are based on very little information about the identity of the word. Notice that the per-word score for C2T falls short of the state-of-the-art in POS tagging due a structural limitation: the tagger assigns tags based on just past and present information. The high accuracy of C2T in spite of this limitation suggests that the majority of the information concerning the POS tag of a word is contained within that word and its past context, and that the LM is particularly good at encoding this information.

Evolution of Tag Predictions over Time Figure 6.2 illustrates the evolution of POS tag predictions over the string *and I have already overheard youngsters* (extract from the UD data) as processed by C2T. Early into the word *already*, for example, C2T identifies the word, recognizes it as an adverb and maintains this hypothesis throughout. With respect to the morphologically complex word *youngsters* we see C2T making reasonable predictions, predicting PRON for *you-*, ADJ for the next two characters and NOUN for *youngster-* and *youngsters*.
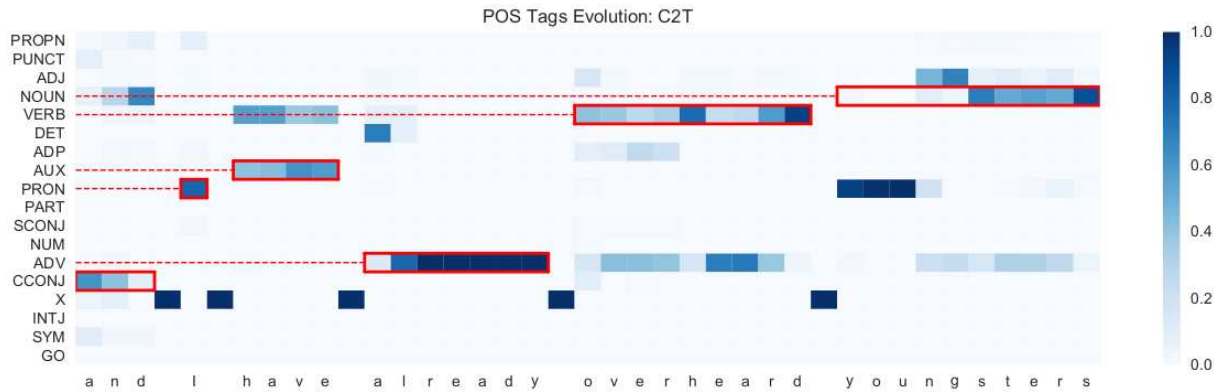
Figure 2: C2T Evolution of POS Tag Predictions. Red rectangles point to the correct tags of words.

7 Selectional restrictions in the LM

English derivational suffixes have **selectional restrictions** with respect to the syntactic category of the base they would attach to. Suffixes *-al, -ment* and *-ance*, for example, only attach to verbs, e.g. *betrayal, annoyance, containment*, while *-hood, -ous*, and *-ic* only attach to nouns, as in *nationhood, spacious and metallic*.[4] The former are thus known as **deverbal**, and the latter as **denominal**. Certain suffixes are members of more than one class, e.g. *-ful* attaches to both verbs and nouns, as in *forgetful* and *peaceful*, respectively. Since our LM appears to encode information about (some) morphological units and part of speech, it is natural to wonder whether it also encodes information about selectional restrictions of derivational suffixes. If it does, then the probability of a deverbal suffix should be significantly higher after a verbal base than after other bases, likewise with denominal suffixes and nominal bases. Our next experiment tests whether this is so.

7.1 Method

Our experiment measures and compares the probability of suffixes with different selectional restrictions across subsets of nominal, verbal, and adjectival bases, as processed by the LM. We use carefully chosen nonce words as bases in order to abstract away from previously seen base-suffix combinations, which the model may have simply memorized.

Probability We compute the probability of a suffix given a base as the joint probability of its characters. For example, the probability of suffix

---

-ion attaching to base *edit* is:

$$p(ion \mid edit) = p(i \mid edit) \times p(o \mid editi) \times p(n \mid editio)$$

Since the LM is a wordless language model, $p(\cdot|base)$ is approximated from $p(\cdot|\mathbf{c})$ where $\mathbf{c}$ is the entire past. The probability of a suffix in the context of a particular syntactic category was computed as the average probability over all bases belonging to that category.

Nonce Bases Nonce bases were obtained by sampling complete words from the LM—that is, sequences delimited by a space or punctuation on both sides. We discarded all words that appeared in an English dictionary, and imposed several restrictions on the remaining candidates: their probability had to be at most one standard deviation below the mean for real words (to ensure they weren't highly unlikely accidents of the sampling procedure), and the probability of a following space character had to be at most one standard deviation below the mean for real words (to avoid prematurely finished words, such as *measu* and *experimen*). In addition, nonce words had to be composed entirely of lowercase characters and couldn't end in a suffix (as certain suffixes included in the experiment only attach to base stems). The candidates that met these conditions were labeled for POS using C2T. The final nonce bases used were the ones whose POS tag confidence was at most one standard deviation below the mean confidence with which tags of real words were assigned. Some examples of nonce words from the final selection are shown in Table 6. An embedding was recorded for every nonce word that met these conditions by taking the hid-

Noun	crystale, algorithm, cosmony, landlough
Verb	underspire, restruct, actrace
Adjective	nucleent, transplet, orthouble

Table 6: Sample Nonce Bases

Noun	-ous, -an, -ic, -ate, -ary, -hood, -less, -ish
Verb	-ance, -ment, -ant, -ory, -ive, -ion, -able, -ably
Adjective	-ness, -ity, -en

Table 7: Syntactically unambiguous derivational suffixes

den state of the language model at the end of the word in context.

Suffixes The suffixes included in this experiment (listed in Table 7) were taken from Fabb (1988), one of the most extensive studies of the selectional restrictions of English derivational suffixes. Fabb discussed 43 suffixes, many of which attach to a base of two out of three available syntactic categories, e.g. *-ize* attaches to both nouns, as in *symbolize*, and adjectives, as in *specialize*. The analysis of such syntactically ambiguous suffixes is complex since the frequency with which they attach to each base type should be taken into consideration, but such statistics are not readily available and require morphological parsing. For the purposes of the present study ambiguous suffixes were thus excluded and only the remaining nineteen suffixes were used.

7.2 Results

Figure 3 shows the results from the experiment. Eleven out of nineteen suffixes exhibit the expected behavior: suffixes *-ment, -ive, -able, -ably, -an, -ic, -ary, -hood, -less, ness* and *-ity* are more probable in the context of their corresponding syntactic base than in other contexts. Suffix *-ment*, for instance, is more than twice as probable in the context of a verbal base than in the context of a nominal or an adjectival base. The fact that almost 70% of suffixes 'select' their correct bases, points to a linguistic awareness within the LM with respect to the selectional restrictions of suffixes.

Despite the overall success of the LM in this respect, some suffixes show a definitive preference for the wrong base. A further analysis of some of these cases shows that they don't necessarily counter the evidence for syntactic awareness within the LM.
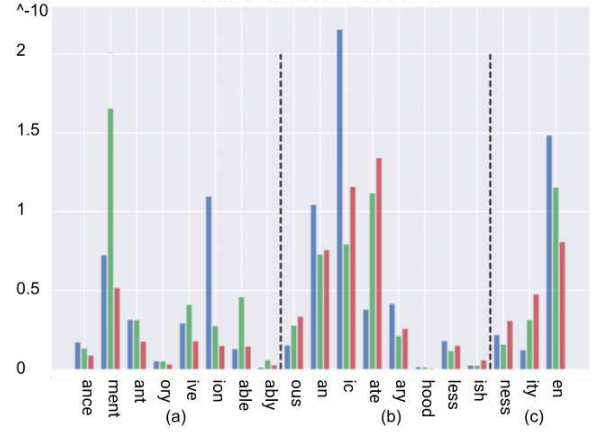

Figure 3: Suffix Probability Following Nominal (blue), Verbal (green) and Adjectival (red) Bases. Suffixes are grouped according to their selectional restrictions: (a) deverbal, (b) denominal and (c) deadjectival. Eleven out of nineteen suffixes obtained highest probability following the syntactic category that matched their selectional restrictions.

7.3 Suffix *-ion*

Deverbal suffix *-ion* should be most probable following verbs, but prefers nominal bases. Notice that *-ion* is a noun-forming suffix: $communicate_{VERB} \rightarrow communication_{NOUN}$, $regress_{VERB} \rightarrow regression_{NOUN}$. It appears that from the perspective of the LM, the syntactic category of such morphologically complex forms extends to their bases, e.g. the LM perceives the *populat* substring in *population* as nominal. This observation can be explained precisely with reference to the high frequency of suffix *-ion*: the suffix itself occurred in 18,945 words in the dataset, while the frequency of its various bases in isolation, e.g. of *populate, regress*, etc., was estimated to be only 9,755[5]. This shows that bases that can take suffix *-ion* were seen more often with it than without it. As a consequence, the LM is biased to expect a noun when seeing one of these character sequences and may thus perceive the base itself as a nominal one.

The tag prediction evolution over *population* and *renewable* in Figure 4, show that this is indeed the case, by comparing a base suffixed with *-ion* to a base suffixed with *-able* (whose selectional restrictions, we know, were learned correctly). For both words C2T starts off predicting NOUN. For

---

[5]To estimate this, we removed the suffix of each word and then searched for the remainder in isolation, followed by *e* and followed by *s/ es*—e.g. for *population* we counted the occurrences of *populat, populate, populates* and *populated*.
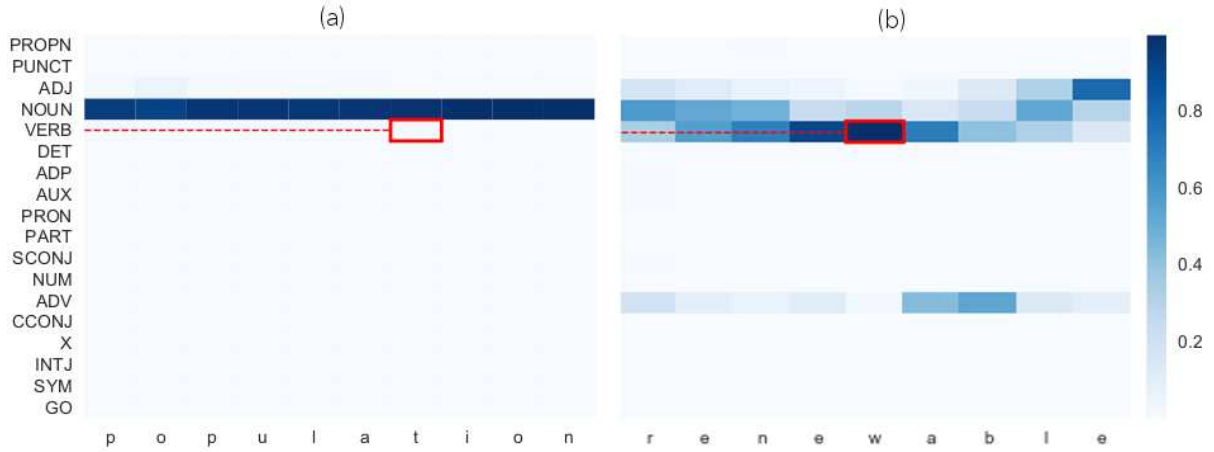
Figure 4: C2T Evolution: (a) *population* and (b) *renewable*. The red square points to the syntactic category of the base.

renew it switches to VERB, which is the correct tag for this base, and only upon seeing the suffix, progresses to the conclusive ADJ tag for *renewable*. For *population*, in contrast the prediction remains constant at NOUN, which is indeed the category of the word, as determined by the suffix.

8 Conclusion

This work presented an exploratory analysis of a 'wordless' character language model, aiming to identify the morpho-syntactic regularities captured by the model.

The first conclusion of this work is that morpheme boundaries are mainly learned by the LM through analogy with syntactic boundaries. Findings relating to the extremely frequent suffix *-ion* illustrate that the LM was able to learn to identify purely morphological boundaries through generalization. But a prerequisite for this generalization is that a morpho-syntactic boundary was also seen in the relevant position during training.

The second conclusion is that having recognized certain boundaries and by extension, the units that lie between them, the model could also learn the regularities that concern these units, e.g. the selectional restrictions of most derivational suffixes included in the study could be captured accurately.

9 Implications for future research

The above conclusions have strong implications with respect to the use of character-level LMs for languages other than English.

English is the perfect candidate for character-level language modeling, due to its fairly poor inflectional morphology. The nature of English is such that the boundary between a base and a suffix is often also a potential word boundary, which makes suffixes easily segmentable. This is not the case for many languages with richer and more complex morphology. Without access to the units of verbal morphology, it is less clear how the model would learn these types of regularities. This shortcoming should hold not just for the LM but for any character-level language model that processes input as a stream of characters without segmentation on the subword level.

This implication is in line with the results of Vania and Lopez (2017) showing that for many languages, language modeling accuracy improves when the model is provided with explicit morphological annotations during training, with English showing relatively small improvements. Our analysis might explain why this is so; we expect analyses of other languages to yield further insight.

Finally, we should point out that it may not be the case that a single, highly-specified *word* unit should exist in every character-level LM. Qian et al. (2016) find that different levels of linguistic knowledge are encoded with different model architectures, and Kádár et al. (2018) find that even a different initialization of an otherwise identical model can results in very different hierarchical processing of the input. We consider ourselves lucky for coming across this particular setup that produced a model with very interpretable behavior, but we also acknowledge the importance of evaluating the reliability of the *word* unit finding in future work.

Acknowledgements

References

William Chan, Navdeep Jaitly, Quoc Le, and Oriol Vinyals. 2016. Listen, attend and spell: A neural network for large vocabulary conversational speech recognition. In *Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on*, pages 4960–4964. IEEE.

Junyoung Chung, Kyunghyun Cho, and Yoshua Bengio. 2016. A character-level decoder without explicit segmentation for neural machine translation. *arXiv preprint arXiv:1603.06147*.

Nigel Fabb. 1988. English suffixation is constrained only by selectional restrictions. *Natural Language and Linguistic Theory*, 6(4):527–539.

Akos Kadar, Grzegorz Chrupala, and Afra Alishahi. 2016. Representation of linguistic form and function in recurrent neural networks.

Ákos Kádár, Marc-Alexandre Côté, Grzegorz Chrupała, and Afra Alishahi. 2018. Revisiting the hierarchical multiscale lstm. *arXiv preprint arXiv:1807.03595*.

Andrej Karpathy. 2015. The unreasonable effectiveness of recurrent neural networks. *Andrej Karpathy blog*.

Yoon Kim, Yacine Jernite, David Sontag, and Alexander M Rush. 2016. Character-aware neural language models. In *Proceedings of AAAI*, pages 2741–2749.

Jason Lee, Kyunghyun Cho, and Thomas Hofmann. 2016. Fully Character-Level Neural Machine Translation without Explicit Segmentation. *Acl-2016*, pages 1693–1703.

Peng Qian, Xipeng Qiu, and Xuanjing Huang. 2016. Analyzing linguistic knowledge in sequential model of sentence. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 826–835.

Teemu Ruokolainen, Oskar Kohonen, Sami Virpioja, and Mikko Kurimo. 2013. Supervised morphological segmentation in a low-resource learning setting using conditional random fields. In *CoNLL*, pages 29–37.

Cicero Nogueira dos Santos and Bianca Zadrozny. 2014. Learning Character-level Representations for Part-of-Speech Tagging. *Proceedings of the 31st International Conference on Machine Learning*, ICML-14(2011):1818–1826.

Ilya Sutskever, James Martens, and Geoffrey E Hinton. 2011. Generating text with recurrent neural networks. In *Proceedings of the 28th International Conference on Machine Learning (ICML)*, pages 1017–1024.

Clara Vania and Adam Lopez. 2017. From Characters to Words to in Between : Do We Capture Morphology ? *To appear in Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*.

Linlin Wang, Zhu Cao, Yu Xia, and Gerard de Melo. 2016. Morphological segmentation with window lstm neural networks. In *AAAI*, pages 2842–2848.

Ron J Weiss, Jan Chorowski, Navdeep Jaitly, Yonghui Wu, and Zhifeng Chen. 2017. Sequence-to-sequence models can directly translate foreign speech. *arXiv preprint arXiv:1703.08581*.