Using A Probabilistic Syllable Model to Improve Scene Text Recognition

Jacqueline L. Feild
University of
Massachusetts Amherst
jfeild@cs.umass.edu

Erik G. Learned-Miller
University of
Massachusetts Amherst
elm@cs.umass.edu

David A. Smith
College of Computer
and Information Science
Northeastern University
dasmith@ccs.neu.edu

***Abstract*—This paper presents a new language model for text recognition in natural images. Many existing techniques incorporate n-gram information as an additional source of information. One problem is that some n-grams are very uncommon, but will still appear in a word across a syllable boundary. These words are given a low probability under an n-gram model. To overcome this problem, we introduce a probabilistic syllable model that uses a probabilistic context-free grammar to generate recognized word labels that are consistent with syllables. In other words, labels generated by this model are pronounceable. This is important for scene text recognition where text often includes proper nouns and standard dictionary information cannot be a useful resource. We show that this language model leads to increased recognition accuracy over a bigram model and discuss the benefits over a dictionary model.**

I. Introduction

The problem of recognizing text in natural images has gained a lot of popularity in recent years. The rise of smartphone users and the potential applications of translating text in the environment for travelers and aiding in navigation for people with low vision make this a particularly interesting and useful area of research. New approaches are needed, since text in the environment can exhibit characteristics that are very different from text in documents. Traditional optical character recognition solutions do not need to take artistic fonts or widely varying backgrounds into account and do not need to consider the unconstrained nature in which images may be captured.

While many appearance models have been shown to perform very well for recognizing individual characters [1], [2], [3], language information is also important for improving recognition results. Many existing techniques incorporate n-gram information into their models, which describes how likely groups of characters are to occur next to each other [2], [4], [5], [6]. This information is very informative, but it is a highly local source of information so it can lead to word labeling errors. For example, bigram models allow a word to have a high probability as long as neighboring character labels have a high probability of occurring together. This means that a word may have a sequence of three unlikely consonants, but the probability will be high as long as each pair is likely to occur next to each other. Additionally, pairs of neighboring characters that occur across a syllable boundary may have a very low probability of occurring together, giving the entire word a low probability. As an example, consider the word 'Amherst'. The combination of 'm' followed by 'h' is very rare in English, and as a result the word has a low probability under a bigram model.

Fig. 1: Sample scene text images with fonts that are difficult to recognize. Performance can be improved by combining appearance information with language information.

In this paper we introduce a new probabilistic syllable language model that overcomes this problem by incorporating additional information about syllables into the model. We demonstrate the use of a probabilistic context-free grammar (PCFG), which encapsulates information about syllables, consonant groups and vowel groups in English and forces word labels to be consistent with a grammar. When humans encounter a new word, we often parse the word into syllables first and then look at the vowel and consonant sequences. This model produces word labels that can be parsed in the same way, because each will be made up of syllables. As a result, each recognized word generated under this language model is pronounceable. This type of syllable-based language model is particularly useful for the domain of scene text recognition where many of the words are proper nouns. These words are not likely to be in a standard dictionary, but we can take advantage of the fact that they should all be pronounceable.

The remainder of the paper is organized as follows. In section two we describe related work. Then in section three we describe our new language model in greater detail. Section four describes our data set, experimental setup and results and in section five we discuss related future work. We conclude in section six.

II. Related Work

There is a large amount of existing work on many different subproblems of scene text recognition. In this section, we will discuss those that are most closely related to our work.

There are several existing methods that incorporate language information from a small lexicon. Wang et al. first introduced this subproblem, called word spotting, where recognized words must be drawn from a given lexicon of 50 to 1000 words [7], [8]. Mishra et al. [9] and Wang et al. [10] also present competitive systems that provide word spotting solutions. Similarly, Novikova et al. present a system that uses a larger lexicon of 90,000 words [11]. These techniques all provide solutions for closed-vocabulary text recognition, because recognized words are restricted to those in the lexicon. This can be a problem for recognizing text in the environment, because a lot of text is from street signs or business signs and is not likely to appear in a standard lexicon and would have to be added manually. The technique we present in this paper allows recognized words that are not found in a lexicon.

There are other existing techniques that incorporate n-gram language information, such as bigram or trigram probabilities to provide solutions for open-vocabulary text recognition. Smith et al. [4] incorporate bigram probabilities into their model. Weinman et al. [5] and Neumann et al. [2] combine information from bigram probabilities and a lexicon. Mishra et al [6] incorporate higher order n-grams into their model to improve text recognition performance. Unfortunately, n-gram probabilities penalize rare letter combinations that can often occur across syllable boundaries in words. In this work, we introduce a language model that overcomes this limitation by modeling syllables directly.

This work is also related to literature on probabilistic context-free grammars (PCFG). In this work, we use a PCFG as a language model for a text recognition task. This is done previously for mathematical equation recognition [12]. In addition, probabilistic context-free grammars have been used as language models for speech recognition tasks [13], [14]. They have also been used for syllabification tasks [15], [16].

III. Probabilistic Syllable Model

We model syllables in words with a probabilistic context-free grammar (PCFG). A context-free grammar G is formally defined as a four tuple $G =< V, \Sigma, R, S >$, where V is a set of non-terminal characters, Σ is a set of terminal characters, R is a set of production rules and S is the start symbol. A probabilistic context-free grammar associates a probability with each production rule. The probability of a particular parse under a grammar can be found by multiplying the probabilities of each rule in the parse.

Using a PCFG for our language model will incorporate a broader range of information. Instead of producing results which are consistent at the level of pairs of characters, results under this model will be consistent at the syllable level. This syllable model will also alleviate the problem of penalties on neighboring labels that cross a syllable boundary. Consider the example of the word 'Amherst' which was mentioned previously. A syllable model can produce the syllables 'am' and 'herst', which are both likely under a standard English syllable model, giving 'Amherst' a high probability. Next we define the probabilistic context-free grammar and explain our training method.

A. Probabilistic Context-Free Grammar Definition

Here we define a PCFG G that models syllables. G has the following set of terminal characters,

Σ = { A,B,C,D,E,F,G,H,I,J,K,L,M,N,O,P,Q,R,S,T,U,V,W,X,Y,Z, a,b,c,d,e,f,g,h,i,j,k,l,m,n,o,p,q,r,s,t,u,v,w,x,y,z }.

The set of non-terminals is,

V = {W,S,S_1,S_2,S_3,S_4,S_5,S_6,S_7,S_8,V,C,V_1,..,V_3, C_1,..,C_5}

with the start symbol of W.

The start symbol W represents a word. The non-terminal S represents a syllable and S_1-S_8 represent the eight types of syllables in this grammar. Each syllable type is made up of some combination of vowel and consonant sequences, represented by the non-terminals V and C. Vowel sequences can be one to three vowels long and consonant sequences can be one to five consonants long. Within each sequence, this grammar models the character at each position explicitly from training data, represented by the non-terminals $V_1 - V_3$ and $C_1 - C5$.

The rules R are the following set:

W	$\rightarrow$	S
W	$\rightarrow$	SW
S	$\rightarrow$	$S_1 \vert S_2 \vert S_3 \vert S_4 \vert S_5 \vert S_6 S_7 \vert S_8$
S_1	$\rightarrow$	V
S_2	$\rightarrow$	CV
S_3	$\rightarrow$	VC
S_4	$\rightarrow$	CVC
S_5	$\rightarrow$	VCe
S_6	$\rightarrow$	CVCe
S_7	$\rightarrow$	CVCeC
S_8	$\rightarrow$	VCeC
V	$\rightarrow$	$V_1 \vert V_2 \vert V_3$
C	$\rightarrow$	$C_1 \vert C_2 \vert C_3 \vert C_4 \vert C_5$
V_1	$\rightarrow$	$a \vert e \vert i \vert o \vert u \vert y$
V_2	$\rightarrow$	$aa \vert ae \vert ai \vert ao \vert au \vert ay \vert ... \vert yy$
V_3	$\rightarrow$	$aaa \vert aae \vert aai \vert aao \vert aau \vert aay \vert ... \vert yyy$
C_1	$\rightarrow$	$b \vert ... \vert z$
C_2	$\rightarrow$	$bb \vert ... \vert zz$
C_3	$\rightarrow$	$bbb \vert ... \vert zzz$
C_4	$\rightarrow$	$bbbb \vert ... \vert zzzz$
C_5	$\rightarrow$	$bbbbb \vert ... \vert zzzzz$

The following table contains examples of words of varying lengths that are randomly generated from this grammar:

Length	2	3	4	6	8
Words	co	nag	tear	tanluw	ancenner
	el	sel	pene	enples	opintest
	ta	bal	whin	esshep	ritfurci
	ni	ner	bini	tyfmyc	itentlec
	am	dow	thaw	enodan	iinefoth

These examples show that this grammar generates words that are pronounceable. Note that they are not necessarily words in English, since this grammar is only a basic approximation of English grammatical rules. The case of each character is not taken into account by this grammar, so we converted these examples to lowercase for readability, since letters can swap between uppercase and lowercase within a word.

B. Model Training

We estimated the probabilities for this context free grammar on a combination of two types of documents. First, we used a syllabified dictionary to count and normalize the information needed. Since a dictionary does not contain a proportional amount of syllables (i.e. there are many words in a dictionary that start with zy, but these do not occur nearly as often in real documents), we augmented this training data with the same information from the top ten books from Project Gutenburg. We tested the three methods of just dictionary information, just book information and both types of information together and found that all three performed similarly. For the experiments in this paper we use the combination method.

IV. Experiments

In this section we compare the performance of a probabilistic syllable model to three different models for text recognition. These include an appearance model, a model that combines appearance and bigram information, and a model that combines appearance and dictionary information.

A. Data Sets

We use two publicly available data sets in our experiments. The first is the VIDI data set provided by Weinman et al. [5]. It contains 215 cropped word images taken from signs in a city and truth labels. The data set contains character bounding boxes, which we use in our experiments. The second data set was created for the ICDAR 2011 Robust Reading competition [17]. It contains 1189 cropped word images and truth labels. These images contain text in the environment, but not necessarily from a sign. This data set does not include character bounding boxes, so we use an existing text segmentation method to identify character locations [18]. Since the probabilistic syllable model produces labels from the 52 character classes A...Za...z, we use subsets of both of these data sets created by removing words that include punctuation and numbers. The ICDAR 2011 subset includes 1008 words and the VIDI subset includes 209 words.

B. Appearance Model

Since the focus of this paper is on demonstrating the benefit of using a probabilistic syllable model, we use a very simple appearance model in our experiments. We choose to use a logistic regression classifier because it is easy to train and produces a conditional probability for each character class, given an input feature descriptor. Note that this will not produce state-of-the-art character recognition results, but is sufficient for showing the benefits of using our new language model over a bigram model and a dictionary model.

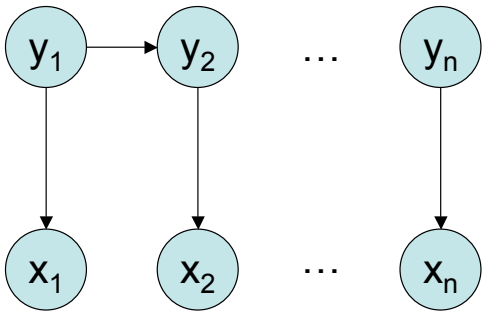

Fig. 2: Hidden Markov model used to combine appearance information with bigram probabilities.

We choose to use a histogram of oriented gradients (HOG) descriptor to model the appearance of characters. This descriptor has been shown to work well for scene text images [7], [8], [9], [6]. We resize each character to 60 by 60 pixels, and we extract one HOG descriptor, centered over the image.

We use these descriptors to train a 52 class (A-Za-z) logistic regression classifier. We use an implementation by Mark Schmidt [19]. This classifier is trained with synthetic font images provided by Weinman et al. [5]. These are binary character images for each character class in 1866 different fonts. We used 1866 positive example images and 200 negative example images for each class.

Once trained, this classifier takes a feature descriptor from an image and produces the conditional probability of each character class. To compute a word label for a new word image using only appearance information, we extract a HOG descriptor from each character image and find the maximum probability label for each using the classifier.

C. Bigram Language Model

We also show the results of using appearance information with bigram language information. These two sources of information can be combined using a standard hidden Markov model (HMM). This is represented by the graphical model in Figure 2. Each output label y_i takes into account the appearance of that character x_i and the previous label y_{i-1} Given this model, we know that,

$$p(\mathbf{x}, \mathbf{y}) = p(y_1) * \prod_{i=1}^{N-1} p(y_{i+1}|y_i) * \prod_{i=1}^{N} p(x_i|y_i)$$

Our goal is to find the word labels $\mathbf{y}$ that maximize that probability. We do this using the Viterbi algorithm, which uses dynamic programming to efficiently compute the most probable character labels, given appearance and bigram probabilities [20].

To compute a word label for a new word image we extract appearance information using the process described in the previous section and estimate bigram probabilities from a collection of books from Project Gutenburg.[1] We then use the

[1] http://www.gutenberg.org/

	VIDI	ICDAR11
Appearance	29.19	14.09
Appearance + Bigrams	31.10	15.38
Appearance + PSM	33.49	16.37

Fig. 3: Word accuracy results comparing a probabilistic syllable model to a bigram model on the VIDI and ICDAR11 data sets.

	VIDI	ICDAR11
Appearance + PSM_{case}	59.33	27.38
Appearance + Dictionary_{case}	57.42	30.46

Fig. 4: Word accuracy results comparing a probabilistic syllable model to a dictionary model on the VIDI and ICDAR11 data sets.

Viterbi algorithm to compute the most probable word label given the appearance and bigram information.

D. Probabilistic Syllable Language Model

In comparison, we show the result of using appearance information with our probabilistic syllable model (PSM). One of the benefits of using a probabilistic context-free grammar is that a dynamic programming algorithm exists to efficiently search for the most probable parse of a sequence of characters under a grammar. This algorithm is called CYK [21]. So for a new word image, we extract HOG descriptors for each character, and calculate the conditional probability for each class using the logistic regression classifier described above. We alter CYK slightly to include these appearance probabilities. So for each character, we give CYK a different distribution over the terminal characters, based on the appearance model probabilities for that character. Then, we run the standard CYK algorithm to find the most probable output labels using our probabilistic syllable language model.

E. Dictionary Language Model

We also compare the performance of a probabilistic syllable model to the performance of a dictionary model. To label a new word image using a dictionary, we evaluate the probability of each word in the dictionary by multiplying the appearance probabilities of each character in the word. Then, we choose the dictionary word with the highest probability as the label. Since a dictionary does not include case information, we evaluate three versions of each dictionary word, one in all uppercase letters, one in all lowercase letters and one in title case with the first letter in uppercase and the rest in lowercase. In order to make a fair comparison to the probabilistic syllable model, we modify the labeling process to include case as well. We generate label versions using CYK, restricted to choose only uppercase letters, only lowercase letters, or an uppercase letter followed by all lowercase letters.

F. Results

We computed word labels for images in both data sets using appearance information, appearance and bigram language information, and appearance information combined with our probabilistic syllable model. The word accuracy results are shown in Figure 3. This experiment shows that on the VIDI data set, the word accuracy increased by around 2% when bigram language information is added and by another 2% when the probabilistic syllable model is used, compared to the bigram model. For the ICDAR 2011 data set, the word accuracy increased by 1% each time. This demonstrates the benefits of using a more sophisticated model, that can capture correct language information across syllable boundaries.

	Word	**HMM Output**	**PCFG Output**
1	AMHERST	LMBERst	AMHERst
2	PRODUCTS	pPOoUCTS	pRODUCTS
3	Essex	SssEx	EssEx
4	address	Rdiness	address
5	Attorney	Nttorney	Attorney
6	Oldenburg	Cldenburg	oldenburg

Fig. 5: Output of the HMM model vs. the PCFG model for sample scene text images.

Figure 5 shows the output of the HMM model and the PCFG model for some sample scene text images. Each of these examples shows the benefit of using a probabilistic context-free grammar as a language model instead of a bigram model. As mentioned previously, one of the downfalls of a bigram model is that it gives high probabilities to entire words as long as each pair of neighboring characters is likely to occur together. In the first example, 'lm' and 'mb' are common bigrams, but put together in a sequence they become highly unlikely. The PCFG constructs results by syllables instead, so the output in each example, even if it is incorrect, is pronounceable.

We also computed word labels for images in both data sets using appearance information and a dictionary, compared to appearance information and a probabilistic syllable model. The word accuracy results are shown in Figure 4. On the VIDI

data set, the syllable model performs better than the dictionary model with a word accuracy of 59.33% compared to 57.42% using the dictionary model. On the ICDAR 2011 data set, the dictionary model performs better with a word accuracy of 30.46 % compared to 27.38% using the probabilistic syllable model. The strength of the dictionary model is that it maps each word image to the best dictionary word. The downfall is that it cannot produce labels that do no occur in the dictionary. In contrast, the probabilistic syllable model labelled 16.67% of the ICDAR 2011 non-dictionary words correctly, and 33% of the VIDI non-dictionary words correctly. This makes the probabilistic syllable model a better choice for data sets that include a large fraction of non-dictionary words.

V. Discussion

This model suggests several directions for future work. The first is to explore changes to the grammar definition. In this paper, we defined a grammar that models each syllable as a sequence of consonant and vowel groups, and models the probabilities of each combination of consonants or vowels within those groups. This grammar does not use any information about how often syllable types occur next to each other, which can be a problem, for example, when words are generated with two vowel groups next to each other. We could alter the grammar to include information about what types of syllables occur near each other. As another extension, we could also learn how consonant and vowel groups relate to one another, i.e a particular vowel group follows a particular consonant group with high probability.

The experimental results in Figure 3 and Figure 4 also show the motivation for incorporating case information into the model. We see a large increase in accuracy on both data sets when case information is added to the probabilistic syllable model. Without this information, the case can swap between lowercase and uppercase in the middle of a word. One special case we discovered is when words have an uppercase letter in the middle of the word. This can occur in business names, i.e. PeoplesBank. The uppercase letter is likely to occur at the beginning of a syllable, so a syllable-based language model like this is a natural choice to handle this special case. In the future we will investigate the best way to design a grammar to allow this special case while incorporating additional information about case consistency.

VI. Conclusion

In this paper, we present a new language model for scene text recognition. It incorporates more sophisticated language information by modeling syllables with a probabilistic context-free grammar. This approach is a better model of language information across syllable boundaries, so words with unlikely bigrams that cross syllable boundaries are not penalized. In addition, words are made up of syllable components, so word labels produced are pronounceable. In our experiments, we show an increase in recognition performance when using this language model, compared to a bigram model and show the benefits of using it compared to a dictionary model.

Acknowledgments

We thank Jerod Weinman and Marwan Mattar for providing the VIDI data set. This material is based upon work supported by the National Science Foundation Graduate Research Fellowship under Grant No. S121000000211. It is also supported by NSF Grant IIS-0916555.

References

[1] T. De Campos, B. Babu, and M. Varma, "Character recognition in natural images," in *International Conference on Computer Vision Theory and Applications*, 2009.

[2] L. Neumann and J. Matas, "A method for text localization and recognition in real-world images," in *Asian Conference on Computer Vision*, 2010.

[3] A. Coates, B. Carpenter, C. Case, S. Satheesh, B. Suresh, T. Wang, D. Wu, and A. Ng, "Text detection and character recognition in scene images with unsupervised feature learning," in *International Conference on Document Analysis and Recognition*, 2011.

[4] D. Smith, J. Feild, and E. Learned-Miller, "Enforcing similarity constraints with integer programming for better scene text recognition," in *CVPR*, 2011.

[5] J. Weinman, E. Learned-Miller, and A. Hanson, "Scene text recognition using similarity and a lexicon with sparse belief propagation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2009.

[6] A. Mishra, K. Alahari, and C. Jawahar, "Scene text recognition using higher order language priors," in *BMVC*, 2012.

[7] K. Wang and S. Belongie, "Word spotting in the wild," in *European Conference on Computer vision*, 2010.

[8] K. Wang, B. Babenko, and S. Belongie, "End-to-end scene text recognition," in *International Conference on Computer vision*, 2011.

[9] A. Mishra, K. Alahari, and C. Jawahar, "Top-down and bottom-up cues for scene text recognition," in *Computer Vision and Pattern Recognition*, 2012.

[10] T. Wang, D. Wu, A. Coates, and A. Ng, "End-to-end text recognition with convolutional neural networks," in *International Conference on Pattern Recognition*, 2012.

[11] T. Novikova, O. Barinova, P. Kohli, and V. Lempitsky, "Large-lexicon attribute-consistent text recognition in natural images," in *European Conference on Computer Vision*, 2012.

[12] E. Miller and P. Viola, "Ambiguity and constraint in mathematical expression recognition," in *Proceedings of the National Conference of Artificial Intelligence*, 1998, pp. 784–791.

[13] D. Jurafsky, C. Wooters, J. Segal, A. Stolcke, E. Fosler, G. Tajchman, and N. Morgan, "Using a stochastic context-free grammar as a language model for speech recognition," in *IEEE International Conference on Acoustics Speech and Signal Processing*, vol. 1, 1995, pp. 189–189.

[14] K. Lari and S. Young, "Applications of stochastic context-free grammars using the inside-outside algorithm," *Computer speech & language*, vol. 5, no. 3, pp. 237–257, 1991.

[15] K. Müller, "Probabilistic Context-Free Grammars for Syllabification and Grapheme-to-Phoneme Conversion," in *Proceedings of the Conference on Empirical Methods on Natural Language Processing*, 2001.

[16] K. Muller, "Probabilistic context-free grammars for phonology," in *Proceedings of the ACL Workshop on Morphological and Phonological Learning*. Association for Computational Linguistics, 2002.

[17] A. Shahab, F. Shafait, and A. Dengel, "Icdar 2011 robust reading competition challenge 2: Reading text in scene images," in *International Conference on Document Analysis and Recognition*, 2011.

[18] J. Feild and E. Learned-Miller, "Scene text recognition with bilateral regression." Department of Computer Science, University of Massachusetts Amherst, Tech. Rep. UM-CS-2012-021, 2012.

[19] M. Schmidt, "Matlab software," http://www.di.ens.fr/ mschmidt/Software/ index.html, 2013.

[20] A. Viterbi, "Error bounds for convolutional codes and an asymptotically optimum decoding algorithm," *IEEE Transactions on Information Theory*, 1967.

[21] D. H. Younger, "Recognition and parsing of context-free languages in time n^3," *Information and Control*, vol. 10, no. 2, pp. 189–208, 1967.