Subword Spotting for Use in a Computer Assisted Transcription System

Brian Davis, Robert Clawson and William Barrett
Brigham Young University, Provo, Utah
Email: briandavis@byu.net

***Abstract*—Recently, computer assisted transcription (CAT) systems for handwritten documents have been proposed that use word spotting to speed up a human transcriber's work. They are, however, dependent on frequent repetition of words in documents to be effective. We propose that character n-grams could be used in place of words to construct a CAT system as n-grams occur much more frequently. We demonstrate some preliminary results in spotting subword character bigrams and trigrams that are adequate for a feasible CAT system.**

I. Introduction

With current technology, a fully automated, unconstrained handwriting recognition system on historical documents is not possible for the required level of accuracy for many applications. During a recent competition for handwriting recognition on historical documents, the top method had a word error rate above 25% [1]. However, there are still ways to leverage recognition techniques to speed up the transcription process, while allowing human oversight and guidance to maintain accuracy. These are computer assisted transcription (CAT) methods, in which the computer and human user's efforts are coupled.

Robert Clawson designed Intelligent Indexing [2][1], a CAT system for tabular documents. Intelligent Indexing relies on finding matching word images in a document column and assigning them the same user-specified label, as seen in Fig. 1. This provides an accurate CAT system where the user oversees all transcription. The user oversight of matches is accomplished by showing a list of matches to the user (with an adjustable threshold for sensitivity) from which the user removes the false-positive matches. This leverages the human user's natural ability to discriminate. Zagoris et al. [3][2] also proposed a CAT system using word spotting. Rather than focusing on having a user remove bad spotting results, as the user confirms spotted word images are correct, a relevance feedback loop helps create a more refined query for that word. However, both of these approaches are limited as they require frequent word repetition to be effective.

II. Using Character N-Grams

While many words do not repeat with a high frequency in most documents, character n-grams do. We refer to a character n-gram as groups of n letters within a word. In particular, we focus on bigrams and trigrams. Let us examine

[1]See http://tiny.cc/intelind for a short demo.
[2]See http://vc.ee.duth.gr/ws/ for an interactive demo.

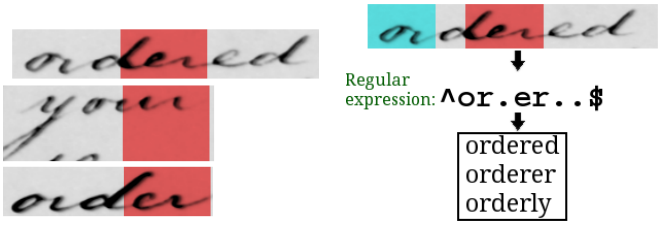
(a) A small window shows matching words from the column. The user can get rid of bad matches (e.g. "Wife") by clicking on them.

(b) The matched words are given the same label, indicated by the highlighting. and the red box is advanced to the next word to be transcribed.

Fig. 1: Clawson's CAT system, Intelligent Indexing.

the George Washington (GW) dataset [4] as an example. The most common bigram in the English language ("th") occurs 622 times, and the 50th most common bigram ("ur") occurs 131 times in this dataset. This is compared to the most common word in the English language ("the") occurring 82 times, and the 50th most common word ("us") occurring 9 times. Additionally, there are many words (such as "river") that occur only once in the dataset, but are composed of common character n-grams.

The system we propose will begin by spotting a particular n-gram (the bigram "er" in Fig. 2a). User oversight of less confident spottings will remove most of the false-positives. The n-grams spotted in a particular word are compiled into a regular expression that is used to query the lexicon, as seen in Fig. 2b. If the returned list of possible trascriptions is short enough, a user can then select the correct transcription for the word with very little effort. We will now walk you through a simulation of the system we have done. If the 100 most frequent bigrams in the English language are spotted in the GW dataset with 50% recall, 54% of the words in the corpus can be narrowed down to 10 or fewer possible transcriptions, assuming (a) that we can correctly guess the correct number of characters we haven't spotted in the words, (b) that user oversight removing false-positives gives us 100% precision, and (c) that we are using a 115,000 word lexicon. Additionally,



(a) (b)

Fig. 2: (a) shows results of spotting the bigram "er" with our adaption of [5]. The false positive ("ur" in "your") could be removed with user oversight. (b) shows that from the bigrams "or" and "er," a regular expression can be constructed that returns only three possible transcriptions from a 115,000 word lexicon.

the system will be iterative; it can learn from correct spottings and transcriptions, creating new spotting queries. Character n-grams that were missed initially will then be spotted with subsequent queries. If we simulate these subsequent queries achieving 50% recall on the remaining bigrams, then by spotting the 100 bigrams again, 74% of the corpus will be narrowed down to 10 or fewer possible transcriptions. Some of our assumptions in our simulation are optimistic. In practice, an uncertain number of characters not spotted would result in longer lists of possible transcriptions, and some false-positives would have to be detected at the users' word-selection step, slowing down the system. We are are confident, however, that it still would be effective.

III. Character N-Gram Spotting Experiment

Spotting subword character n-grams is a largely unexplored area. We have run some initial tests using some word spotting techniques in the application of subword spotting to demonstrate that this is feasible. We evaluated two spotting methods: a part-structured inkball method by Howe [6] and an attribute embedding method presented by Almazán et al. [5]. Because these are whole word spotting methods, they are not suited to perform subword spotting without some modification. We modified the code provided by Almazán[3] and Howe[4].

We used two datasets to evaluate the subword spotting performance of these methods: the George Washington dataset [4] and the IAM off-line dataset [7]. The testing queries consisted of 10 images of each of the 20 most frequent bigrams in the English language and 5 images of each of the 10 most frequent trigrams. Query images were manually cropped from word images of the datasets. Both methods were tested using the same first fold of the testing partitions used by as Almazán et al. (available with the provided code). The GW testing set contains 1215 word images, and the IAM testing set contains 13752 word images. Stop words were not ignored in any way. The attribute embedding method was not trained on character n-gram images but on word images.

[3]http://github.com/almazan/watts

[4]http://cs.smith.edu/~nhowe/research/code/index.html#psm

TABLE I: Mean-average-precision on 20 most frequent bigrams and 10 most frequent trigrams

	Bigrams		Trigrams	
Method	GW	IAM	GW	IAM
Howe (query by image)	22.11%	9.36%	28.13%	6.25%
Almazán et al. query by image	40.67%	23.10%	69.79%	42.91%
Almazán et al. query by string	64.20%	32.99%	65.05%	48.87%
Almazán et al. hybrid query	64.32%	33.04%	72.38%	49.81%

IV. Results and Conclusion

Table I shows the results of the experiment. The method by Almazán et al. achieved a mean-average-precision that is acceptable for a CAT system as a user can discard incorrect spotting results. The 50% recall rate used in the previous section was drawn from these results; it will roughly correspond to 66% precision. This is an acceptable level as users will be discarding less than half of the bigram spotting results, however, it would be desirable to relieve the user burden more by having a more accurate spotting method. We are confident better subword spotting accuracy will be able to be achieved with further work given we have used only a naïve adaptation of whole-word spotting approaches.

Clawson and Zagoris et al. show that word spotting can create effective CAT systems that are reliant high word repetition. We argue, and our preliminary results indicate, that character n-gram spotting can be used to construct a CAT system able to perform well independent of high word repetition. A user's oversight will allow precise n-gram spotting results. As more n-grams are spotted in a particular word, its list of possible transcriptions is constrained so that a user can easily select the correct one. As spotting results are harvested to create more queries, more n-grams will be spotted, and more of the document will be transcribed.

References

[1] J. A. Sánchez, A. H. Toselli, V. Romero, and E. Vidal, "ICDAR 2015 competition HTRtS: Handwritten text recognition on the tranScriptorium dataset," in *Proc. ICDAR*, 2015. [Online]. Available: icdar.org/proceedings

[2] R. Clawson, "Intelligent indexing: A semi-automated, trainable system for field labeling," Master's thesis, Brigham Young University, 2014. [Online]. Available: scholarsarchive.byu.edu/etd/5307/

[3] K. Zagoris, I. Pratikakis, and B. Gatos, "A framework for efficient transcription of historical documents using keyword spotting," in *Proc. HIP*. ACM, 2015.

[4] V. Lavrenko, T. Rath, and R. Manmatha, "Holistic word recognition for handwritten historical documents," in *Proc. DIAL*. IEEE, 2004, pp. 278–287.

[5] J. Almazán, A. Gordo, A. Fornés, and E. Valveny, "Word spotting and recognition with embedded attributes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 12, pp. 2552–2566, 2014.

[6] N. Howe, "Part-structured inkball models for one-shot handwritten word spotting," in *Proc. ICDAR*. IEEE, 2013, pp. 582–586.

[7] U. Marti and H. Bunke, "The IAM-database: An English sentence database for off-line handwriting recognition," *Int. Journal on Document Analysis and Recognition*, vol. 5, pp. 39–46, 2002.