A Lightweight on-the-fly Capitalization System for Automatic Speech Recognition

Fernando Batista[1,2], Nuno Mamede[1,3], Diamantino Caseiro[1,3], Isabel Trancoso[1,3]
[1] L^2F – Spoken Language Systems Laboratory - INESC ID Lisboa
R. Alves Redol, 9, 1000-029 Lisboa, Portugal
[2] ISCTE – Instituto de Ciências do Trabalho e da Empresa, Portugal
[3] IST – Instituto Superior Técnico - Technical University of Lisbon, Portugal
{fmmb, njm, dcaseiro, imt}@l2f.inesc-id.pt

Abstract

This paper describes a lightweight method for capitalizing speech transcriptions. Several resources were used, including a lexicon, newspaper written corpora and speech transcriptions. Different approaches were tested both generative and discriminative: finite state transducers, automatically built from Language Models; and maximum entropy models. Evaluation results are presented both for written newspaper corpora and speech transcriptions of broadcast news corpora.

Keywords

Rich transcription, capitalization, truecasing, maximum entropy, language models, weighted finite state transducers

1 Introduction

Enormous quantities of digital and video data are daily produced by media organizations, such as radio and TV stations. Automatic speech recognition systems can now be applied to such sources of information in order to enrich it with alternate information for applications, such as: indexing, cataloging, subtitling, translation and multimedia content production. The Automatic Speech Recognition (ASR) output consists of raw text, often in lower-case format. Even if useful for many applications, such as indexing and cataloging, the ASR output benefits from capitalization information for other tasks, such as subtitling and multimedia content production. In general, enriching the speech output aims to improve legibility, enhancing information for future human and machine processing. Besides capitalization, enriching speech recognition covers other activities, such as insertion of punctuation marks and detection and filtering of disfluencies, not addressed in this paper.

This paper describes a method for capitalization of automatic speech recognition transcriptions, using a reduced set of data, which can be integrated, for example, on an on-the-fly system for subtitling. The different data sources used for our experiments are described in section 2. Section 3 defines the performance measures used for evaluation. Section 4 describes the different methodologies employed. The results achieved for capitalization are presented in section 5. The paper ends with some final comments and ideas for future work.

Corpus	Duration	Tokens	
train	61h	467k	81%
development	8h	64k	11%
test	6h	46k	8%

Table 1: *Different parts of the SR corpus*

2 Data sources

The ultimate goal of this work is to perform automatic capitalization on the output of an ASR system. We will start by using written newspaper corpora for training and testing a set of methods and finally we will apply these methods on speech transcriptions. By doing so, we expect to analyze the performance degradation when moving from written corpora to speech transcriptions, and combine the available data sources in order to provide richer training sets, thus enhancing final results. Some small lexicons are also experimented in order to overcome the problem of using small data sets for training. The following subsections provide details about each one of the used data sources.

2.1 Speech Recognition Corpus

The Speech Recognition (SR) is an European Portuguese broadcast news corpus, collected in the scope of the ALERT international project[1]. The training data of the SR corpus was recorded during October and November 2000, the development data was recorded during December, and the evaluation data was recorded during January 2001[2]. Table 1 shows details about the corpus data sets.

The manual orthographic transcription of this corpus includes information such as punctuation marks, capital letters and special marks for proper nouns, acronyms and abbreviations. Each file in the corpus is divided into segments with information about the

[1] https://www.l2f.inesc-id.pt/wiki/index.php/ALERT

[2] https://www.l2f.inesc-id.pt/wiki/index.php/ALERT_Corpus

Corpus	Period	Words	
train	1995 to 2000	97.9 M	76%
development	1st sem. 2001	15.7 M	12%
test	2nd sem. 2001	15.1 M	12%

Table 2: *Different parts of the RecPUB corpus*

List	Words
Acronyms and Abbreviations	72
Proper nouns	466
Names of countries and cities	357
Nouns and abbreviations (POS selection)	652
Acronyms (POS selection)	14

Table 3: *The different information sources used for building LEX*

start and end locations in the signal file, speaker id, speaker gender and focus conditions.

Besides this manual orthographic transcription, other transcriptions are available: the one automatically produced by the Audio Preprocessor module (APP) and the one automatically produced by the ASR module. Nevertheless, for the results presented in this paper only manual transcriptions are used.

2.2 Corpus "Recolha do Público"

Most of the experiments here described use a limited vocabulary of 57k words, extracted from the lexicon of our ASR module. The BN speech transcription information is without doubt insufficient to provide enough training material for all words in our vocabulary. "Recolha do Público" corpus (RecPUB) is a written newspaper corpus of about 130 million words that can be used to provide the remaining information. Table 2 provides details on each part of the corpus.

The properties of a written newspaper corpus are quite different from what can be found in speech transcriptions. For example, a speech transcription may be produced from spontaneous or planned speech and may contain phenomena, such as filled pauses and disfluencies. However, the co-occurrence of words found in written corpora may be a valuable resource for the capitalization task, which can also be applied to speech transcriptions.

2.3 Lexicons

A lexicon (LEX) built from several lists of words was also used in order to overcome the small size of the training data. Apart from existent lists of acronyms, proper nouns, names of countries and capitals, a POS-tagger was also used for identifying unambiguous Nouns and Abbreviations in the vocabulary. Table 3 shows the different lists that compose our lexicon. After joining all the separate components, a lexicon of about 1500 unique words is achieved.

An additional lexicon (PubLEX) was also built, writing each word of the vocabulary with the most common graphical form, as appearing in the RecPUB corpus training data. The lexicon size is 57k.

3 Performance measures

The following performance measures are used: Precision, Recall, F-measure, and Slot Error Rate (SER) [4], defined in equations (1) to (4). For the capitalization task here performed, a slot corresponds to the occurrence of a word containing capital letters.

$$Precision = \frac{C}{hyp} = \frac{C}{C+S+I} \tag{1}$$

$$Recall = \frac{C}{ref} = \frac{C}{C+S+D} \tag{2}$$

$$Fmeasure = \frac{2 * Precision * Recall}{Precision + Recall} \tag{3}$$

$$SER = \frac{total\ slot\ errors}{ref} = \frac{I+D+S}{C+D+S} \tag{4}$$

For the equations: C is the number of correct slots; I is number of insertions (spurious slots / false acceptances); D is number of deletions (missing slots / false rejections); S is number of substitutions (incorrect slots); ref is number of slots in reference; and hyp is number of slots in hypothesis.

```
Reference:  here is an Example of a big SER
Hypothesis: here Is an example of a big SER
                 ins   del              cor
```

Figure 1: *Example of slot occurences*

Applying the performance measures to the example of figure 1, a 50% Precision, Recall and F-Measure is achieved, but the SER is still 100%, which may be a more meaningful measure, once the number of slot mistakes is greater than the number of correct ones.

4 Methodologies

Different methodologies are exploited in order to recover capitalization information: (1) using the SRILM toolkit [6]; (2) using a transducer, built from a previously created Language Model (LM); and (3) using maximum entropy models. The first two methodologies are generative (joint) modeling approaches, while the last one is discriminative (conditional). The following subsections provide details on each of the methodologies.

4.1 SRILM toolkit and transducers

For our generative modeling approach, the initial step consists of creating an N-gram language model from the corpus. This step is performed using the SRILM toolkit. For trigram language models, we use Chen and Goodman's modified Kneser-Ney discounting, with backoff or with interpolation, as implemented by the `ngram-count` tool.

The `disambig` tool, an HMM-based tagger that uses an hidden-event N-gram LM [7], is also part of

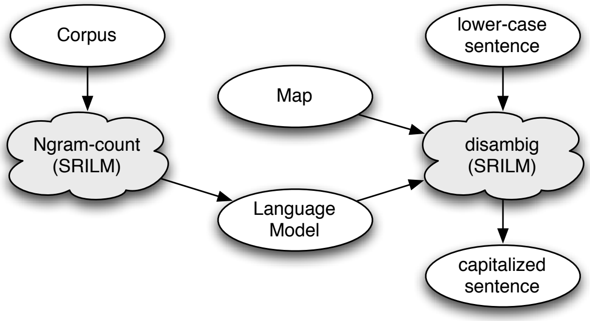

Figure 2: *Using only the SRILM toolkit*

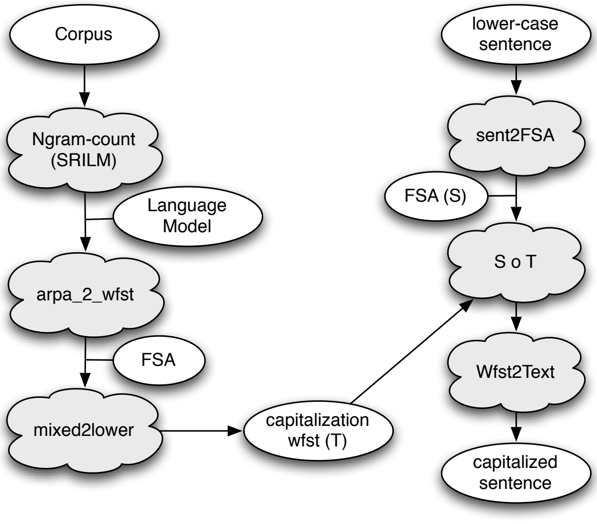

Figure 3: *Using a WFST to perform capitalization*

the SRILM toolkit, and can be used to perform capitalization directly from the language model. Figure 2 illustrates the process, where each cloud represents a process and a ellipse represents data. The *Map* corresponds to a file with all alternate forms of writing each word in the vocabulary. This is the most straightforward method, producing fast results, often used by the scientific community for this kind of task. It was part of the baseline suggested in the IWSLT2006 workshop competition[3].

Another method, based on Weighted Finite State Transducers (WFST), is illustrated in figure 3. The SRILM toolkit is firstly used produce an LM from the corpus and then the LM is converted into a finite state automaton (FSA), which can be viewed as a WFST having the input equal to the output. The transducer T, used for performing capitalization, results from the previous transducer where each input word was converted to its lower-case representation. The input of the resultant transducer can be represented by a lower-case vocabulary, while the output contains all graphical forms. The right side of figure 3 shows the process of capitalizing a sentence. The input sentence is firstly converted into an FSA (S) and then the operation $bestpath(S \, o \, T)$ produces the desired result, in the form of another automaton.

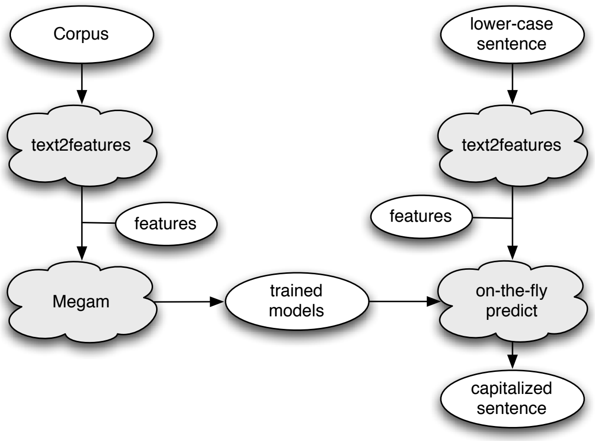

Figure 4: *The maximum entropy approach*

Both methods use the `ngram-count` tool for creating the LM from the training data. As a consequence of that, experiments performed in the same conditions by the two methods share the same language model.

4.2 Maximum entropy

The discriminative modeling approach here applied is based on Maximum Entropy (ME) models. The MegaM tool - Maximum Entropy Model Optimization Package [2] is used for training, and the `on-the-fly` predicting tool uses previously trained models for performing the capitalization task. Figure 4 illustrates the overall process. The first step consists of training the models using a set of predefined features and the next step consists of using that information in order to predict the results. The MegaM tool includes an option for predicting results from previously trained models, but unfortunately it was not prepared to deal with a stream of data and produces results only after completely reading the input. The `on-the-fly` predicting tool overcomes this problem while using previously trained models in the original format.

The ME modeling approach allows easy combination of several sources of information, such as word information and POS tagging information. Nevertheless, the experiments here described only use features capturing word information, sometimes combined as bigrams and trigrams. The delay between the input and the output constitutes a problem for a module required to work on an on-the-fly system. Besides the computational time delay, an important aspect to be taken into consideration is the number of words on the right of the current word required to make a decision. For the results presented here, the feature set was chosen in order to avoid a right context greater than one.

5 Results

We assume that the first word of each sentence will always be capitalized in other processing step, for example along with the punctuation, since its correct graphical form mostly depends on its position in the

[3] http://www.slt.atr.jp/IWSLT2006/downloads/case+punc_tool_using_SRILM.instructions.txt

LM options	LM size
unigrams	7.3Mb
bigrams	27Mb
trigrams	78Mb

Table 4: *Different LM sizes*

LM options	Prec	Recall	F	SER
unigrams	91%	74%	82%	0.333
bigrams	**94%**	**84%**	**89%**	**0.212**
3-gram	93%	79%	85%	0.271
3-gram, interpol.	93%	80%	86%	0.266

Table 5: *SRILM Toolkit results over RecPUB corpus*

sentence. These words are excluded from training and evaluation, seeing that evaluation results may be influenced when taking such words into account [3].

The next subsections will show results achieved with both the generative and discriminative approaches: We will start by presenting some results obtained with the SRILM toolkit and the WFST, applied to both written newspaper corpora and speech transcriptions. Then some experiments, using maximum entropy with a limited quantity of data, will be described. Results achieved using only the most common graphical form are included in all experiments, which is a popular baseline for similar work [1, 3].

5.1 The generative approach

The first set of experiments were performed on written newspaper corpora, using RecPUB both for training and testing. As we use a vocabulary, all words outside vocabulary were marked "unknown" and punctuation marks were also removed from the corpus. The content of the corpus became closer to a speech transcription, but without recognition errors or disfluencies. A large size written corpora often contains a number of orthographic errors and less common words which, used in bigrams and trigrams, originates large quantities of ineffective data. Because of that, bigrams and trigrams occurring less than 5 times were not considered for LM training. Table 4 shows the size of each LM depending on the building options: unigrams, bigrams, and trigrams.

The first capitalization results for written newspaper corpus are presented in table 5. Both training and evaluation were performed with the RecPUB corpus, using the SRILM toolkit. Results achieved by unigrams show that, using only the current word, an SER of 33% can be achieved. The use of bigrams conducts to the best result, increasing both precision and recall, and showing that word co-occurrence is an important aspect to be taken into consideration for a capitalization task. The disambig tool has produced poor results for trigrams, which can be related to an increase of the search space when moving to a trigram language model. These results provide a baseline for the following experiments.

LM options	Prec	Recall	F	SER
unigrams	91%	77%	83%	0.307
bigrams	94%	88%	91%	0.176
3-gram	95%	89%	92%	**0.155**
3-gram, interpol.	95%	89%	92%	**0.154**

Table 6: *WFST results over RecPUB corpus*

LM options	Prec	Recall	F	SER
unigrams	81%	76%	78%	0.418
bigrams	78%	85%	81%	**0.388**
3-gram	79%	81%	80%	0.409
3-gram, interpol.	80%	81%	81%	0.390

Table 7: *Results of SRILM method on the SR corpus*

LM options	Prec	Recall	F	SER
unigrams	81%	77%	79%	0.422
bigrams	79%	86%	82%	**0.368**
3-gram	78%	87%	82%	0.380
3-gram, interpol.	78%	86%	82%	0.382

Table 8: *Results of WFST method on the SR corpus*

The second experiment was performed using WFSTs on the same corpus. Moreover, the capitalization transducers were produced from the same LM used in the previous experiment. Results from this experiment are shown on table 6. This method produces better results independently of the option for building the LM. The increase in the precision and recall values is correlated with the usage of higher order ngrams, and trigram models achieves the best results. The biggest difference, in terms of SER, occurs when moving from unigrams to bigrams, given that trigram models only add about 1% to precision and recall values.

The following experiments use the previous LM models, built for written newspaper data, in order to capitalize broadcast news speech transcriptions. Tables 7 and 8 shows the results of these experiments, using both the SRILM toolkit and WFST methods, over the SR corpus evaluation data. Results show the expected decrease of performance when going from written newspaper corpora to speech transcriptions. Notice however that the training was performed in the written newspaper corpora, which do not share the same properties as the speech transcription. The best results were achieved using bigrams for both methods, revealing a significant difference between written corpora and speech transcriptions.

Other experiments on capitalization were also performed for BN speech transcriptions, using only the SR data for training. The best result in terms of SER was 0.434, corresponding to a precision of 82% and recall of 72%. This result is no better than the worse result achieved using the written newspaper corpora for training, even so this was an expected result given the small training data size.

The WFST method consistently produces better results than using the disambig tool. Nevertheless, the current implementation of the WFST method implies loading, composing and searching a large nondeterministic transducer, thus being the most computationally expensive method proposed.

Exp	Corpora features	Lexicons	Prec	Rec	F	SER
1	w_i		85%	65%		0.466
2	w_i (w_{i-1}, w_i) (w_i, w_{i+1})		84%	67%		0.455
3	w_i (w_{i-1}, w_i) (w_i, w_{i+1}) $(w_{i-2}, w_{i-1}, w_i)(w_{i-1}, w_i, w_{i+1})$		84%	67%		0.458
4		PubLEX	80%	73%	76%	0.453
5	w_i (w_{i-1}, w_i) (w_i, w_{i+1})	LEX	84%	68%	75%	0.446
6	w_i (w_{i-1}, w_i) (w_i, w_{i+1})	PubLEX	85%	73%	79%	**0.391**
7	w_i (w_{i-1}, w_i) (w_i, w_{i+1})	LEX, PubLEX	85%	73%	79%	**0.391**

Table 9: *Results of maxent over the BN speech transcriptions (SR corpus)*

5.2 The discriminative approach

The Maximum Entropy approach requires that all information be expressed in terms of features, according to a previously defined feature set. The resultant data size may be several times the original one, making it difficult to use large corpora, such as the RecPUB corpus, for training purposes. The SR corpus training material (467k words) is clearly insufficient for covering the 57k vocabulary. In order to mitigate this problem we also used the two lexicons, previously described in section 2. By using this approach we expect to achieve gains while introducing small data resources.

Table 9 shows results for the most relevant experiments, combining different feature sets and information sources. For each one of the experiments, the table describes all the features used for capturing knowledge from SR corpus, where: w_i is the word at position i of the corpus, (w_i, w_j) is the bigram containing words w_i and w_j and (w_i, w_j, w_k) is the trigram containing words w_i, w_j and w_k .

The first 3 experiments were conducted using only the speech transcription data for training, without any additional resource. Experiment 1 establishes a baseline for what can be achieved using only the most common way of writing a given word, taking the SR corpus training data as reference. For this experiment, if no training data was available for a given word, it was kept lower-case. Experiments 2 and 3 show that adding bigrams and trigrams do not produce large changes, even so, bigram models is a good compromise between size and performance. These three experiments show that the SR corpus is far from sufficient for training.

Experiment 4 shows that by using only the most common way of writing a word, taking RecPUB data as reference, produces better results than using SR corpus alone. This experiment also shows that the ME approach produces lower results than previous generative approaches. The first line of each one of the tables 7 and 8 corresponds to the same task performed either with SRILM toolkit or the WFST, and the SER is about 3.3% better than current results. This is due to the representation of the information used in both approaches: the generative approaches considers the two words from bigram (w_i, w_j) independently, while the ME approach consider the bigram as a whole.

Experiment 5 shows the contribution of a small lexicon resource (LEX). The best result is achieved by combining the speech transcriptions from the SR corpus and the PubLEX lexicon, as shown in experiment 6. Experiment 7 also shows that LEX resource does not add much information when using PubLEX.

The SER achieved using bigrams with the maximum entropy is only 2% worse than best results achieved using a generative approach, however this method allows a much faster way of performing capitalization directly from an input stream, given that the correct graphical form of a given word is calculated by means of a weighted sum of values, given by the word's correspondent features.

6 Concluding remarks

This paper addresses the problem of producing the capitalization information for texts without that information, such as the output of an ASR system. Three different methods were described and results were presented both for manual transcriptions of speech and written newspaper corpora. One of the methods, described as lightweight, combines different data resources for training and uses a straightforward procedure for predicting results. The performance achieved using this method is almost as good as using our best approach, while using a smaller number of resources. It has been integrated on an on-the-fly subtitling module for broadcast news, deployed at the Portuguese national television broadcaster.

Results for recovering capitalization both from written unpunctuated newspaper corpora and from broadcast news transcription were presented. Concerning the written newspaper corpus, we conclude that bigram and trigram information significantly contributes to enhance results, despite that trigram information only contributes with about 1% to precision and recall values. The used BN speech transcription corpus is too small and does not cover much of the vocabulary. Results show that using trigrams do not significantly improve results achieved by bigram when dealing with speech transcriptions. Lexica contribute to enhance the results when dealing with small size training data.

7 Future work

For now only three ways of writing a word were explored: lower-case, all-upper, first-capitalized, not covering mixed-case words such as RTPi and SuSE. We expect to address these cases in a near future, perhaps using a small lexicon.

Experiments concerning speech transcriptions and achieved results were produced using a manual BN speech transcription. We plan to define a strategy for performing evaluation directly on automatic speech transcriptions, either performing a previous alignement with the manual transcriptions, or performing a human evaluation.

The problem of dealing with a dynamic vocabulary remains unaddressed in our experiments. Other features, such as word prefix and suffix, number of vowels and consonants shall also be explored. We also plan to introduce information coming from a part-of-speech tagger, in our ME models, already shown to improve results [5].

In the scope of the national TECNOVOZ[4] project, large amounts of broadcast news hand-annotated transcriptions, are now being daily produced. In the near future we plan to have much more training material, which will hopefully provide more accurate results.

8 Acknowledgments

This work was partially funded by the FCT projects LECTRA (POSC/PLP/58697/2004), NLE-GRID (POSC/PLP/60663/2004) and WFST (POSI/PLP/47175/2002). INESC-ID Lisboa had support from the POSI Program of the "Quadro Comunitário de Apoio III".

References

[1] C. Chelba and A. Acero. Adaptation of maximum entropy capitalizer: Little data can help a lot. *EMNLP '04*, 2004.

[2] H. Daumé III. Notes on CG and LM-BFGS optimization of logistic regression. Implementation available at http://hal3.name/megam/, August 2004.

[3] J.-H. Kim and P. C. Woodland. Automatic capitalisation generation for speech input. *Computer Speech & Language*, 18(1):67–90, 2004.

[4] J. Makhoul, F. Kubala, R. Schwartz, and R. Weischedel. Performance measures for information extraction. In *Proceedings of DARPA Broadcast News Workshop*, Herndon, VA, February 1999.

[5] A. Mikheev. Periods, capitalized words, etc. *Computational Linguistics*, 28(3):289–318, 2002.

[6] A. Stolcke. SRILM - An extensible language modeling toolkit. In *Proceedings of International Conference on Spoken Language Processing*, volume 2, pages 901–904, Denver, CO, 2002.

[7] A. Stolcke and E. Shriberg. Automatic linguistic segmentation of conversational speech. In *Proc. ICSLP '96*, volume 2, pages 1005–1008, Philadelphia, PA, 1996.

[4] http://www.tecnovoz.com.pt/