UNIVERSITAT POLITÈCNICA DE VALÈNCIA

DEPARTAMENT DE SISTEMES INFORMÀTICS I COMPUTACIÓ

Language Model Adaptation for Video Lecture Transcription

Master Thesis - MIARFID

Adrià A. Martínez Villaronga

Directors:
Dr. Alfons Juan Ciscar
Dr. Jesús Andrés Ferrer

September 16, 2013

Abstract

In this work we propose a method to adapt language models to specific lectures in the context of automatic transcription of video lectures. We explore different variations of the adaptation technique obtaining significant WER reductions for the Spanish repository poli[Media] .

Contents

Chapter 1

Introduction

1.1 Motivation

In the last decade several online video repositories have arisen, rapidly becoming fundamental knowledge assets, being particularly important in the area of education. Many educational institutions as well as other organizations have created their own repositories for online education [1, 2, 3, 4].

These repositories are making the education accessible to a wide community of potential students. As with many other repositories, most lectures are not transcribed because of the lack of efficient solutions to obtain them at a reasonable level of accuracy. However, transcription of video lectures is clearly necessary to make them more accessible. Also, they would facilitate lecture searchability and analysis, such as classification, summarisation, or plagiarism detection. In addition, communities of people with hearing disabilities would be able to follow the lectures just by reading the transcriptions.

Manual transcription of these repositories is excessively expensive and time-consuming and current state-of-the-art automatic speech recognition (ASR) has not yet demonstrated its potential to provide acceptable transcriptions on large-scale collections of audiovisual objects. However, it has such potential by simply exploiting the rich knowledge we have at hand. More precisely, in this kind of videos the speaker is accompanied by some kind of background slides during its presentation. In these cases, a strong correlation can be observed between slides and speech. Consequently, this slides provide an interesting opportunity to adapt general-purpose ASR models by massive adaptation from lecture-specific knowledge.

The proposed scenario is considered by some projects which aim at providing full set of transcriptions for online lecture repositories. Our work is framed in the European trans**Lectures** project [5, 6], which is explained in Section 1.2 and whose objective is to develop innovative and cost-effective solutions to produce accurate transcriptions and translations in VideoLectures.NET [3] and poli[Media] [1] through the free and open-source platform Matterhorn [7].

Within the framework of the transLectures project, our intention is to improve the video lecture transcription of the poli[Media] database by adapting language models using the content of the slides as well as other resources obtained from the web.

1.2 transLectures and poli[Media]

The aim of the transLectures project is to develop innovative, cost-effective solutions to produce accurate transcriptions and translations in VideoLectures.net and poli[Media] through a free and open-source platform called Matterhorn [7]. Matterhorn is a platform designed to support the creation and management of educational audio and video content.

The poli[Media] database was created for production and distribution of multimedia educational content at the Universitat Politècnica de València. Lecturers are able to record lectures under controlled conditions which are distributed along with time-aligned slides. For the time being, the poli[Media] catalogue includes almost 10000 Spanish videos accounting for more than 1500 hours of lectures of which only about 2000 videos can be accessed freely.

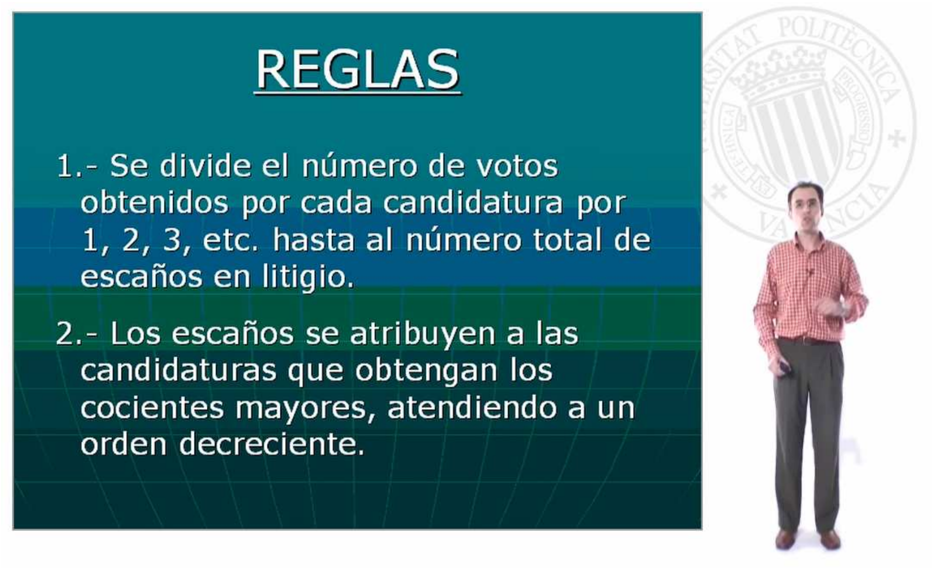
In the frame of transLectures project a poli[Media] corpus have been created for both acoustic and language modelling, using more than 100 hours of poli[Media] open access videos so that the corpus will be accessible by the research community beyond the scope of the transLectures project. The details about this corpus will be introduced later in Chapter 4.

A typical video capture from a poli[Media] video lecture is depicted in figure 1.1. The lecturer is localised at the right side of the screenshot, while the slides are shown at the left side of the video. The fact that poli[Media] lectures are recorded under studio conditions make this study case specially appropriate for ASR tasks.

1.3 Structure of the document

In Chapter 2 Automatic Speech Recognition and Language Modelling are introduced. We will present the basis of Statistical ASR and we will explain the key role that the language model plays in this task. Different techniques to train the language model will be explained and we will introduce the metrics that will be used to evaluate the models. We will briefly introduce the language modelling tools used in this work.

The adaptation techniques proposed in this work are presented in Chapter 3. We will explain the fundamentals of the adaptation and how to train the different language models that have a role in it.



REGLAS

1.- Se divide el número de votos obtenidos por cada candidatura por 1, 2, 3, etc. hasta al número total de escaños en litigio.

2.- Los escaños se atribuyen a las candidaturas que obtengan los cocientes mayores, atendiendo a un orden decreciente.

Figure 1.1: A poli[Media] video capture

The details of all corpora used in this work can be found in Chapter 4, including the techniques used to retrieve the text in the slides from the video, as well as the method used to obtain related documents from the web.

Chapter 5 focuses on the different experiments carried out to test the performance of the different models developed during the research.

Finally, Chapter 6 summarises the achieved results and presents the conclusions and the scientific contributions as well as the further work that can be derived from this research.

Chapter 2

Overview of Automatic Speech Recognition and Language Modelling

2.1 Automatic Speech Recognition

Automatic Speech Recognition (ASR) is a field in Computer Sciences which aims at developing computer systems that are able to automatically generate transcriptions for a given acoustic signal. Current techniques for ASR are based on statistical pattern recognition, consisting in, given the acoustic signal find the transcription that maximises the likelihood, i.e. let x be an acoustic signal and considering w a sequence of words:

$$\hat{w} = \arg\max_{w} p(w|x) \tag{2.1}$$

where $\hat{w}$ is the most likely word sequence for the input signal x. The probability distribution $p(w|x)$ is not usually easy to model so Bayes rule is applied to obtain an equivalent expression:

$$\hat{w} = \arg\max_{w} p(w|x) = \arg\max_{w} \frac{p(x|w)\,p(w)}{p(x)} = \arg\max_{w} p(x|w)\,p(w) \tag{2.2}$$

This decomposition allowed us to separate the original probability in two terms, which are easier to estimate: $p(x|w)$, the acoustic model; and $p(w)$, the language model. Figure 2.1 displays the basic architecture of such system.

The acoustic model estimates the probability of, given a word sequence w, the feature vector x is observed, i.e., $p(x|w)$. To calculate this model Hidden Markov models are often used, which will be trained using Baum-Welch algorithm.

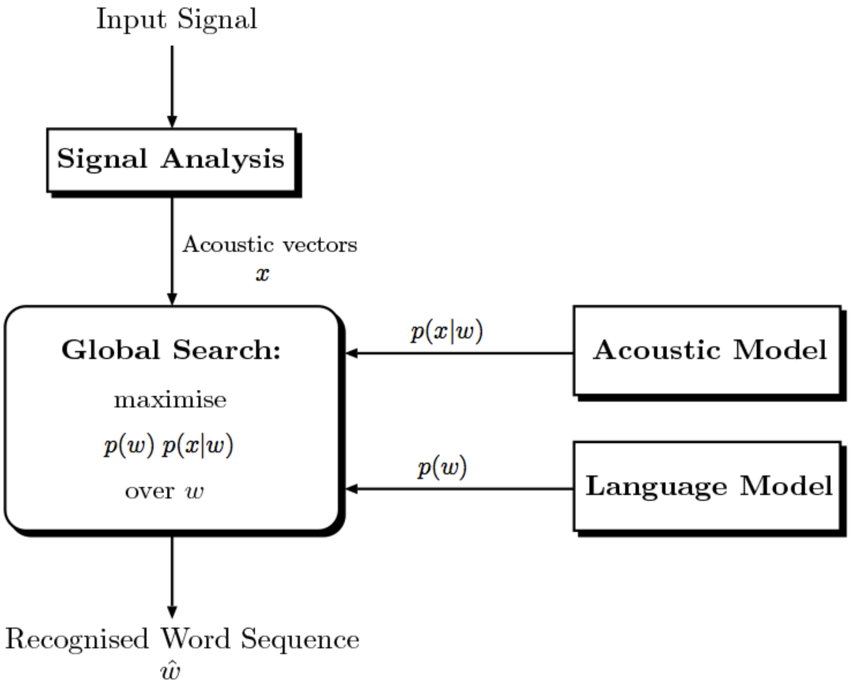


Figure 2.1: Basic architecture of an ASR system

The language model provide an estimation of the probability the sequence w appears, i.e., $p(w)$. There is no need of labelled data to train language models, just a set of sentences in the desired language. There are several ways to train language models, some of which are detailed below in Section 2.2.

2.2 Language Modelling

Language models play an important role in many applications of language technology such as Machine Translation, Information Retrieval or Speech Recognition, where the transcription $\hat{w}$ for a given acoustic signal is calculated

$$\hat{w} = \arg\max_{w} p(w|x) = \arg\max_{w} p(x|w)\, p(w)$$

where $p(w)$ is the language model. This model should assign non-zero probabilities to every word sequence $w = w_1, w_2, \ldots, w_k = w_1^k$ possible in the target language:

$$p(w_1^k) = p(w_1) \prod_{i=2}^{k} p(w_i|w_1^{i-1}) \tag{2.3}$$

where w_1^{i-1} is the history

Due to the spareness of the data, it is almost impossible to consider the full word history w_1^{i-1} and equivalence classes are used. Different approaches have been developed over the years, changing the way they build the equivalence classes [8, 9]. The n-gram approach, which consists in considering not the whole sequence w_1^i, but only n elements (n-gram), i.e., w_{i-n+1}^i, using only the $n-1$ previous words of the history (Equation 2.4), is the most used nowadays.

$$p(w_1^k) = p(w_1)\prod_{i=2}^{k} p(w_i|w_1^{i-1}) \approx p(w_1)\prod_{i=2}^{k} p(w_i|w_{i-n+1}^{i-1}) \tag{2.4}$$

2.2.1 N-gram Language Models

As explained before, an n-gram language model estimates the probability $p(w_i|w_1^{i-1})$ by limiting the length of the history to only $n-1$ elements, i.e., $p(w_i|w_1^{i-1}) \approx p(w_i|w_{i-n+1}^{i-1})$. This probability is estimated by maximum likelihood obtaining the expression in Equation 2.5:

$$p(w_i|w_{i-n+1}^{i-1}) = \frac{c(w_{i-n+1}^{i})}{c(w_{i-n+1}^{i-1})} \tag{2.5}$$

where $c(s)$ represents the number of occurrences of the sequence s in the training corpus. This n-gram approach has reduced the spareness of the data, however in a speech recognition task it is still usual to find words or n-grams that have not been seen in the training, obtaining zero probabilities for sentences that, even being uncommon, are still correct. Smoothing techniques address this issue by assigning small (but non-zero) probabilities to unseen events and obtaining smoother distributions.

Smoothing Techniques

Additive Smoothing

One of the simplest smoothing techniques is additive smoothing which consists in adding a factor δ to the count of all words and n-grams, typically $0 < \delta \leq 1$. Thus, we set

$$p_{\text{add}}(w_i|w_{i-n+1}^{i-1}) = \frac{\delta + c(w_{i-n+1}^{i})}{\delta|V| + \sum_{w_i} c(w_{i-n+1}^{i})} \tag{2.6}$$

where V is the vocabulary. This method, although simple and easy to implement usually performs poorly.

Absolute discounting

When there is not enough data to properly estimate the probability of a given n-gram, the probability of the corresponding $(n-1)$-gram could be used to approximate it, combining both probabilities, $p(w_i|w_{i-n+1}^{i-1})$ and $p(w_i|w_{i-n+2}^{i-1})$. The way to do so is using a linear interpolation, as described in [10]

$$p_{\text{interp}}(w_i|w_{i-n+1}^{i-1}) = \lambda p_{\text{ML}}(w_i|w_{i-n+1}^{i-1}) + (1-\lambda)p_{\text{interp}}(w_i|w_{i-n+2}^{i-1}) \tag{2.7}$$

Using this idea, absolute discount smoothing [11, 12], instead of multiplying higher order distribution by a factor λ, subtracts a constant discount D from each non-zero count, obtaining

$$p_{\text{abs}}(w_i|w_{i-n+1}^{i-1}) = \frac{\max\{c(w_{i-n+1}^{i}) - D, 0\}}{\sum_{w_i} c(w_{i-n+1}^{i})} + (1-\lambda) p_{\text{abs}}(w_i|w_{i-n+2}^{i-1}) \tag{2.8}$$

Subtracting D from each non-zero count we gain some probability mass that should be distributed uniformly among all the n-grams, and to make the distribution sum to 1, we take

$$1 - \lambda = \frac{D}{\sum_{w_i} c(w_{i-n+1}^{i})} N_{1+}(w_{i-n+1}^{i-1}\ \cdot) \tag{2.9}$$

Function $N_{1+}(w_{i-n+1}^{i-1}\ \cdot)$ in equation 2.9 indicates the number of different words that can appear following w_{i-n+1}^{i-1} and it is formally defined

$$N_{1+}(w_{i-n+1}^{i-1}\ \cdot) = |\{w_i : c(w_{i-n+1}^{i-1} w_i) > 0\}| \tag{2.10}$$

Kneser-Ney

Kneser and Ney [13] introduced an extension of absolute discounting where the lower order distribution, which is used when the higher order one is zero or near-zero, is built in a novel manner, specially optimised for these situations. The intuitive idea behind Kneser-Ney is that the probability of an n-gram should not be proportional to the number of occurrences of the n-gram, but to the number of different contexts that precede the n-gram. A good example that illustrates this idea can be found in [14].

Kneser-Ney probability is defined as follows

$$p_{\text{KN}}(w_i|w_{i-n+1}^{i-1}) = \frac{\max\{c(w_{i-n+1}^{i}) - D, 0\}}{\sum_{w_i} c(w_{i-n+1}^{i})} + \frac{D}{\sum_{w_i} c(w_{i-n+1}^{i})} N_{1+}(w_{i-n+1}^{i-1}\cdot) p_{\text{KN}}(w_i|w_{i-n+2}^{i-1}) \tag{2.11}$$

Where the recursion of $p_{\text{KN}}(w_i|w_{i-n+2}^{i-1})$ is calculated as follows:

$$p_{\text{KN}}(w_i|w_{i-n+2}^{i-1}) = \frac{N_{1+}(\cdot\ w_{i-n+2}^{i})}{N_{1+}(\cdot\ w_{i-n+2}^{i-1}\ \cdot)} \tag{2.12}$$

And the base case:

$$p_{\text{KN}}(w_i) = \frac{N_{1+}(\cdot\ w_i)}{N_{1+}(\cdot\ \cdot)} \tag{2.13}$$

Where $N_{1+}(\cdot\ w_{i-n+2}^{i})$, $N_{1+}(\cdot\ w_{i-n+2}^{i-1}\ \cdot)$, $N_{1+}(\cdot\ w_i)$ and $N_{1+}(\cdot\ \cdot)$ are defined as follows:

$$\begin{aligned} N_{1+}(\cdot\, w_{i-n+2}^{i}) &= |\{w_{i-n+1} : c(w_{i-n+1}^{i}) > 0\}| && (2.14)\\ N_{1+}(\cdot\, w_{i-n+2}^{i-1}\, \cdot) &= |\{(w_{i-n+1}, w_i) : c(w_{i-n+1}^{i}) > 0\}| = \sum_{w_i} N_{1+}(\cdot\, w_{i-n+2}^{i}) && (2.15)\\ N_{1+}(\cdot\, w_i) &= |\{w_{i-1} : c(w_{i-1}^{i}) > 0\}| && (2.16)\\ N_{1+}(\cdot\,\cdot) &= \sum_{w_i} N_{1+}(\cdot\, w_i) && (2.17) \end{aligned}$$

Modified Kneser-Ney

Chen and Goodman [14] proposed a modified version of Kneser-Ney smoothing that reported very good results. Their purpose consists in using a variable discount value, rather than a constant value. This variable value is given by the equation

$$D(c) = \begin{cases} 0 & \text{if } c = 0 \\ D_1 & \text{if } c = 1 \\ D_2 & \text{if } c = 2 \\ D_{3+} & \text{if } c \geq 3 \end{cases} \tag{2.18}$$

Instead of using Equation 2.11, we take

$$p_{\text{KN}}(w_i|w_{i-n+1}^{i-1}) = \frac{c(w_{i-n+1}^{i}) - D(c(w_{i-n+1}^{i}))}{\sum_{w_i} c(w_{i-n+1}^{i})} + \gamma(w_{i-n+1}^{i-1}) p_{\text{KN}}(w_i|w_{i-n+2}^{i-1}) \tag{2.19}$$

To make the distribution sum to 1, we take

$$\gamma(w_{i-n+1}^{i-1}) = \frac{D_1 N_1(w_{i-n+1}^{i-1}\,\cdot) + D_2 N_2(w_{i-n+1}^{i-1}\,\cdot) + D_{3+} N_{3+}(w_{i-n+1}^{i-1}\,\cdot)}{\sum_{w_i} c(w_{i-n+1}^{i})} \tag{2.20}$$

Backoff and interpolated models

Equation 2.7 presented a basic model where higher-order and lower-order probabilities were combined by means of a linear interpolation, but it is possible to combine both probabilities in a different way. Interpolation, which is a weighted sum of higher-order and lower-order probabilities, can be expressed as follows:

$$p(w_i|w_{i-n+1}^{i-1}) = \tau(w_i|w_{i-n+1}^{i-1}) + \gamma(w_{i-n+1}^{i-1}) p(w_i|w_{i-n+2}^{i-1}) \tag{2.21}$$

The interpolated model uses lower-order probabilities even if the higher order probability is not zero. In contrast to these models, backoff models only use lower-order probabilities for those n-grams whose higher-order probabilities is zero.

$$p(w_i|w_{i-n+1}^{i-1}) = \begin{cases} \tau(w_i|w_{i-n+1}^{i-1}) & \text{if } c(w_{i-n+1}^{i}) > 0 \\ \gamma(w_{i-n+1}^{i-1}) p(w_i|w_{i-n+2}^{i-1}) & \text{if } c(w_{i-n+1}^{i}) = 0. \end{cases} \tag{2.22}$$

Interpolated model is proven [14] to perform better so this will be the model used for this work.

2.3 Evaluation

In this work different language models for ASR will be presented. In order to evaluate the performance of these models two different measures will be used:

- **Perplexity**: Perplexity measures the average amount of different elements (words) that can follow a given prefix (history) regarding the language model. Given a test set $T = \{t^{(1)}, t^{(2)}, \ldots, t^{(M)}\}$ with M sentences and N words, perplexity is defined as follows:

  $$PPL(T) = 2^{-\frac{1}{N}\sum_{m=1}^{M} \log_2 p(t^{(m)})} \tag{2.23}$$

  A lower perplexity value will indicate that our model fits better to the task. Perplexity may be a good measure for tasks with common vocabularies, but if the vocabulary is not the same perplexities are not comparable. As an example, let's take a model with no vocabulary but the unknown word. The perplexity for this model will be 1, but this model will not be a good model for ASR.

- **WER**: The Word Error Rate will measure the quality of the transcriptions rather than the models themselves. This technique consists of calculating the minimum number of edition operations (substitutions, insertions and deletions) necessaries to obtain the reference sequence given the system output divided by the number of words in the reference. The lower this rate, the better the quality of the system.

  $$WER = \frac{S + I + D}{N} \tag{2.24}$$

2.4 Language Modelling and Automatic Speech Recognition Tools

SRILM

This work has relayed on SRILM toolkit [15] to work with language models. SRILM is a toolkit in development since 1995 at the SRI Speech and Technology research Laboratory. It includes several programs and script to work with language models, and other structures that use LM probabilities, like lattices and n-best lists.

This toolkit allows training models using different types of smoothing techniques, calculate perplexities on a given text, estimate optimal weights on language model interpolation and interpolate several models given the weight of each one.

TLK

As for the speech recognition task we used The **transLectures** -UPV toolkit (TLK) [16], an open source set of tools for Automatic Speech Recognition developed at the Universitat Politècnica de València by the **transLectures** -UPV Team.

Chapter 3
Language Model Adaptation

Our interest is to elucidate whether the slide information provides useful information with respect to a competitive LM baseline. If we compare the improvements obtained by adding slide information to a simple in-domain language model, we would obviously observe an astonishing improvement. For this reason we built a initial competitive baseline.

In order to build this competitive baseline, several n-gram models trained from different out-of-domain corpora, which will be described in Chapter 4, were linearly mixed together with the in-domain model as follows. Let w be the current word within a sentence, and let h be the $n-1$ previous words, then the mixture is made by linear interpolation as follows:

$$p(w|h) = \sum_i \lambda_i p_i(w|h) \tag{3.1}$$

where λ_i is the weight of the linear interpolation corresponding to the i-th n-gram model $p_i(w|h)$. The weights $\{\lambda_1^I\}$ must add up to 1 so that the mixture is a probability. Finally, these weights are used to adapt the model by optimising them with the EM algorithm to maximise the log-likelihood or equivalently to minimise the perplexity of a given development set [17].

The adaptation technique proposed consists in adding one ore more language models to the linear interpolation discussed above. We consider different ways to train these models that can be combined:

(a) Using all the text in the slides of a given video, train one language model for the video using this text extracted from the slides.

(b) Considering time-aligned slides, where for each of the slides there is available the start time and the end time, we can train one model for each one of the slides, using only the text in that slide, and use this model to recognise only the corresponding segment of the video.

(c) Train a model for each video using related documents downloaded from the web.

In this work we will consider adaptations with the following combinations of models:

- (a) Equation 3.2.
- (c) Equation 3.3.
- (a)+(c) Equation 3.4.

In previous works [18] we also considered the combination (a)+(b), but it did not lead to significant improvements in the recognition despite the cost (both temporal and spatial) of adapting and recognising a video with this model was about 10 times slower, so this combination will not be further considered.

$$p(w|h,V) = \sum_i \lambda_i p_i(w|h) + \lambda_V p_V(w|h) \tag{3.2}$$

$$p(w|h,V) = \sum_i \lambda_i p_i(w|h) + \lambda_D p_D(w|h) \tag{3.3}$$

$$p(w|h,V) = \sum_i \lambda_i p_i(w|h) + \lambda_V p_V(w|h) + \lambda_D p_D(w|h) \tag{3.4}$$

where V stands for the current video, $p_V(w|h)$ for the language model trained on the video slides and $p_D(w|h)$ for the language model trained on the documents retrieved for V.

In order to optimise parameters λ_V and λ_D , we had to extend the optimisation proposed in [17] to allow a changing language model, since the models $p_V(w|h)$ and $p_D(w|h)$ vary from one video to another, and the development set is supposed to be made up of several videos. In this way, we obtain a general parameter for all the slides. However, there are videos for which the slides do not contain text or do not make use of slides at all, as well as videos for which no documents are available. Considering this videos as normal videos will cause a distortion in the calculation of the interpolation weights, specially for this dynamic video-dependent n-gram model. Therefore, these special videos should be considered apart when optimising weights and we add a constraint to the optimisation process such that if the slide does not contain text, then the λ_V is forced to be 0. In the same way, if there are no documents λ_D must be also 0.

Chapter 4
Corpora

4.1 Out-of-domain Corpora

Several out-of-domain corpora are used to train language models. All these corpora have been preprocessed according to the needs of an ASR task, removing punctuation marks, converting all the text to lower case and transcribing numbers into letters. Table 4.1 summarises the basic the main statistics of these corpora once the preprocess is applied.

Corpus	# sentences	# words	Vocabulary
EPPS	132K	0.9M	27K
news-commentary	183K	4.6M	174K
TED	316K	2.3M	133K
UnitedNations	448K	10.8M	234K
Europarl-v7	2 123K	54.9M	439K
El Periódico	2 695K	45.4M	916K
news (07-11)	8 627K	217.2M	2 852K
UnDoc	9 968K	318.0M	1 854K

Table 4.1: Basic statistics of the out-of-domain corpora used to generate the LM

4.1.1 Google n-gram

Google Ngram corpus is a corpus [19] provided by Google which contains counts for n-grams ($1 \leq n \leq 5$) from all books digitised by Google up to the moment of the launch, including books from 1538 to 2008. It is offered in several languages, including Spanish. Table 4.2 shows more details about Google Ngram corpus for Spanish.

# unigrams	# pages	# books	Vocabulary
45 360M	128M	521K	292K

Table 4.2: Google Ngram corpus details

4.2 The poli[Media] corpus

The poli[Media] corpus was created by manually transcribing 704 video lectures in Spanish, corresponding more than 100 hours so as to provide in-domain data sets for training, adaptation and internal evaluation in the **transLectures** project. The corpus contains transcriptions of the lectures, as well as transcriptions of the slides for development and test sets. Slides contain time marks, so it is possible to know in each instant of the video, which slide is being displayed. Tables 4.3 and 4.4 contain statistics about this corpus.

	Videos	Time (hours)	# sentences	# words	Vocabulary
train	655	96	41.5K	96.8K	28K
dev	26	3.5	1.4K	34K	4.5K
test	23	3	1.1K	28.7K	4K

Table 4.3: poli[Media] corpus details.

	# videos	# slides	# sentences	# words	Vocabulary
dev	26	107	1865	16.2K	3.5K
test	23	363	1796	14.5K	2.9K

Table 4.4: poli[Media] slides details.

4.2.1 Slides Text Retrieval

In many online repositories the electronic format of the slides is typically not available together with the video. For instance, in the poli[Media] case uploading the slides with the video is an optional step that is many times disregarded. Consequently, there are two types of videos: those with the slides attached, and those without slides. For the former, slides text extraction only depends on tools such as *pdf2text*. Conversely, for the latter, slides must be automatically extracted from each video lecture. This automatic process is divided into 2 steps: first the slide is detected, and then a OCR tool, such as *Tesseract*, is used to extract the text from the detected slide.

Regarding the slide detection technique a very naive yet effective technique is proposed. Specifically, we count the changing pixels from frame to frame, and determine that a change in the slide has been performed if the number of changing pixels exceeds a specified threshold. Each time a new slide is detected, the corresponding frame is captured and passed to the Optical Character Recognition tool.

OCR has become an important and widely used technology for document annotation. However, when dealing with complex images the results turn out to be not as good [20] where an appropriate image preprocessing, text-line detection and text post-processing steps are fundamental. We used *Tesseract*[21] for optical character recognition (OCR).

Two different OCR approaches have been applied using Tesseract. Firstly, we carried out a slide recognition process where each slide was recognised according to different Tesseract parameter configuration in order to improve the transcriptions results. After the recognition, the output was filtered by some simple word generation rules.

Unfortunately, the previous approach provided poor performance due to irregular slide structure such as images, charts, tables and a wide variability in background and font colors, obtaining an OCR WER of 70% . Consequently, for the second approach, we developed a preprocessing module which applies various filters such as despeckling, enhancing or pixel negation and obtains different versions of the same slide by applying several thresholds for binarisation. Each preprocessed slide version is processed by *Tesseract* and post-processed combining all the outputs. This process dramatically improved the accuracy of the obtained text, decreasing the OCR error down to 40%. These are the slides that will be used for the experiments.

Details of these automatically transcribed slides are presented in Table 4.5. We cab observe that the vocabulary and the number of words is slightly larger than in correct slides (Table 4.4).

	# videos	# slides	# words	Vocabulary
dev	26	107	17.4K	3.9K
test	23	363	16.4K	3.1K

Table 4.5: poli[Media] OCR slides details.

4.2.2 External documents

To improve adaptation, external documents have been obtained from the web, performing a search in Google using the title of the video as query. Language detection is applied to be sure that only documents in Spanish are downloaded and a maximum of 5 documents will be retrieved for each video. The documents are downloaded in pdf format and they are automatically converted to text. The preprocess applied to them is the same that we applied for the other corpora.

- **Exact search** Only documents that match the exact title of the video are downloaded. Due to the specific nature of the search it is impossible to find 5 documents for some videos, and there are a few searches that provide no results.
- **Extended search** In this method an exact search is performed first, but if less than 5 documents are found for a given video a non-exact search is preformed (i.e. documents that contain some of the words, but not necessarily all of them, and they do not need to appear together) downloading the remaining videos from these results.

	# words	Vocabulary
Exact Search	333K	67K
Extended Search	410K	93K

Table 4.6: Downloaded documents data

Chapter 5
Experiments

The proposed techniques for language model adaptation are measured in terms of both perplexity and WER obtained with state of the art ASR system [16]. The acoustic model has been trained using the poli[Media] corpus, employing triphonemes inferred using the conventional CART with almost 3900 leaves. Each triphoneme was trained for up to 128 mixture components per Gaussian, 4 iterations per mixture and 3 states per phoneme with the typical left-to-right topology without skips. Additionally, speaker adaptation was performed applying CMLLR feature normalisation (full transformation matrices). The results obtained with this model were competitive in the last trans**Lectures** evaluation [22].

As for the language model, we computed the baseline model as discussed in Chapter 3 by interpolating several individual language models trained in several corpora (Chapter 4). For each out-of-domain corpora, including Google Ngram, we trained a 4-gram language model with SRILM [15] toolkit . The individual 4-gram models were smoothed with modified Kneser-Ney absolute interpolation method [13]. Finally, the training set of poli[Media] was also used as the in-domain corpus. Perplexities obtained for each of this individual models are reported in Table 5.1. As for the vocabulary, we used the top 50K most frequent words over all the corpora plus the in-domain vocabulary.

5.1 Preliminary Experiments

Before the adaptation experiments can be performed, we need to carry out some preliminary experiments. The models generated are really big (about 22 GB) and it leads to slow recognitions, high memory demands and high disk space usages. A way to reduce the size of the models is pruning them. The pruning method applied in this work consists in prune n-gram ($n \geq 2$) probabilities if their removal causes training set perplexity of the model to increase by less than threshold relative. These preliminary experiments will help us determine which threshold provides a better

	Perplexity		OOVs (%)	
Corpus	dev	test	dev	test
EPPS	543.7	710.8	8.21	12
news-commentary	636	747.7	6.73	9.4
TED	615.6	521.2	6.59	7.94
UnitedNations	754	802.9	7.77	10.94
Europarl-v7	460.6	605.7	5.75	8.59
El Periódico	450.2	545.9	5.95	8.61
news (07-11)	358.9	747.6	5.64	7.99
UnDoc	544.9	802.8	6.10	9.21
Google	1370.3	1954.8	4.71	6.95
poli[Media] train	317.9	332.5	4.61	5.23

Table 5.1: Perplexities and OOV words on the development and test sets for all corpora

compromise between space and results.

We will explore the performance of the models if pruning is applied to the individual models before the interpolation, or if it is applied to the resulting interpolated model. The threshold values that will be tested are $2e-8$, $2e-10$ and $2e-12$. We also will explore if a combined prune (after and before the interpolation) is interesting to apply. Table 5.2 shows the results of these experiments.

As explained before, the vocabulary will be conformed by the 50K most frequent words from the out-of-domain corpora plus the words in the in domain corpus, obtaining a final vocabulary of 58K words. This is our baseline vocabulary.

			Dev		Test	
		Model Size	PPL	WER	PPL	WER
No prunning		22 GB	137.98	21.72	169.83	24.25
Before	$2e-8$	2.2 GB	148.05	22.28	177.99	24.76
	$2e-10$	16 GB	138.40	21.80	170.04	24.29
	$2e-12$	22 GB	137.99	21.72	169.84	24.26
After	$2e-8$	163 MB	163.94	23.38	193.51	25.64
	$2e-10$	3 GB	140.84	21.95	172.08	24.25
	$2e-12$	14 GB	138.14	21.75	169.88	24.24
Bef. + Aft.	$2e-10$	3 GB	141.03	21.95	172.26	24.33

Table 5.2: Perplexity and WER for different pruning values

As shown in Table 5.2, a pruning threshold of $2e-10$ applied to the resulting interpolated model is the option with a better compromise between space and error. This result will be used as a baseline for our system.

5.2 Adaptation with slides

5.2.1 Adaptation with OCR Slides

The slides adaptation consists in training a language model for each video using the text in the slides extracted using the second OCR technique explained in chapter 4.2.1. This technique produced transcriptions for the slides with about 40% WER. Some of the words contain non-Spanish characters and it is safe to assume that these words are transcription errors and we decided to treat them as the unknown word. The final vocabulary, built by adding the words in the slides to the baseline vocabulary, contains 60K words.

It is very difficult to locate the limits of the sentences in OCR slides, since punctuation marks usually are not correctly detected, however the language model will consider by default the limits of the lines as the limits of the sentences. Some of these limits will be correctly considered, but not all of them, possibly adding noise to the model. The experiment carried out filtering the limits proves that, although some of the limits are incorrectly detected, the correctly detected ones are also important, resulting in no significant differences between both versions.

Finally we carried out an other experiment training one language model with the slides of all Development and Test videos, instead of using one different for each video.

	Dev		Test	
	PPL	WER	PPL	WER
OCR slides	110.90	21.25	131.84	22.05
No limits	110.31	21.29	132.18	22.06
Single model	126.08	21.06	152.82	22.12

Table 5.3: Adaptation with OCR slides

5.2.2 Adaptation with Manually Transcribed Slides

Improving the quality of the OCR slides can be important to improve the quality of the final ASR transcriptions and we wanted the impact of better slides. Using manually transcribed slides can provide a lower-bound of the error that we can achieve using this adaptation technique with slides. Results displayed in Table 5.4 show that there is still significant room for improvement.

Since including the slides also extends the vocabulary up to 59K words, we also computed the results obtained including this vocabulary in the baseline to check how much of the improvement was due to the vocabulary, and how much was due to the model. Results show that adding only the vocabulary is not useful and the result obtained with this model is not significantly different than the result that we obtained with the baseline model, therefore we can conclude that the improvement is not due to the new vocabulary, but the n-gram reestimation due to the new training data.

	Dev		Test	
	PPL	WER	PPL	WER
Only Vocab	150.8	21.95	195.7	24.25
Correct Slides	96.6	20.36	113.2	20.67

Table 5.4: Adaptation with Manually transcribed slides

5.3 Adaptation with Documents

These experiments have been carried out with the documents downloaded from the web as explained in Chapter 4. Up to this point, we added the full vocabulary of the adapted resources (slides), but for this experiment, the vocabulary of the documents is much bigger than it was in the case of the slides. Regarding this, we performed two version of each experiment: adding the full vocabulary of the documents to the baseline vocabulary and adding only a restricted vocabulary consisting in the words that appear more than 3 times in the full downloaded text. Table 5.5 shows the size of the final vocabulary for each experiment and the results in terms of perplexity and WER obtained in each case.

		Dev		Test	
	Vocabulary	PPL	WER	PPL	WER
Exact Search					
Full vocabulary	93K	149.20	21.72	198.79	22.38
Restricted vocabulary	64K	144.03	21.88	186.15	23.06
Extended Search					
Full vocabulary	116K	147.23	20.87	195.65	21.80
Restricted vocabulary	71K	140.10	21.19	183.20	22.53

Table 5.5: Adaptation with documents only

As observed, only the extended search with the full vocabulary is able to outperform the OCR slides, although it has twice the amount of words.

5.4 Combined Adaptation: Slides and Documents

In these experiments we combine the slides and the documents language models. Just as in the previous experiments, the vocabulary of the documents is very large and adding all these words to the baseline vocabulary results in really large vocabularies.

Full and restricted vocabulary versions explained before are used also in this section. We tested out the performance of the system with both correct and OCR slides.

5.4.1 Documents and OCR Slides

Table 5.6 presents the results of these experiments, as well as the size of the vocabularies used in each one.

		Dev		Test	
	Vocabulary	PPL	WER	PPL	WER
Exact Search					
Full vocabulary	94K	123.55	20.94	165.94	21.59
Restricted vocabulary	66K	120.54	21.04	158.95	21.78
Extended Search					
Full vocabulary	118K	121.84	20.45	159.95	21.33
Restricted vocabulary	73K	117.44	20.68	152.91	21.52

Table 5.6: Adaptation with OCR slides and documents

As expected, the combination of slides and documents significantly improves previous models, even keeping similar vocabulary sizes.

5.4.2 Documents and Correct Slides

For this experiment we used manually transcribed slides. As we explained before, this is not a usual scenario, but it will give us a lower bound of the best error we can obtain using documents and slides.

		Dev		Test	
	Vocabulary	PPL	WER	PPL	WER
Exact Search					
Full vocabulary	93K	108.51	20.15	144.66	20.38
Restricted vocabulary	65K	106.04	20.24	139.28	20.54
Extended Search					
Full vocabulary	117K	107.00	19.68	133.22	20.19
Restricted vocabulary	71K	103.64	19.90	138.59	20.33

Table 5.7: Adaptation with correct slides and documents

As shown in Table 5.7, adding documents to the adaptation with correct slides improves the error obtained, however this improvement is not as pronounced as it was in the case of OCR slides, but it is still a significant improvement.

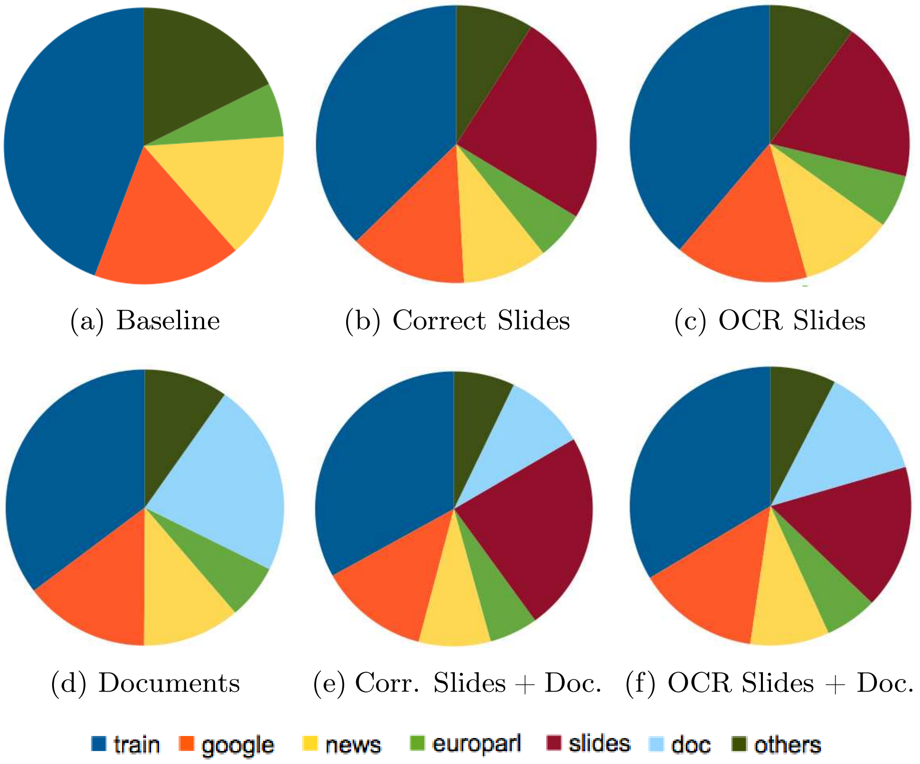
5.5 Summary of results

Table 5.8 pretends to be a summary of the most relevant results obtained in the development of this work. As we expected, the best results are obtained when we use manually transcribed slides and documents, improving the baseline in 4 absolute WER points. However adaptation with OCR and documents still achieves good results with an improvement of 2.8 absolute WER points over our baseline.

	Development		Test	
	PPL	WER	PPL	WER
Baseline	140.8	22.0	172.1	24.3
Correct Slides	96.6	20.4	113.2	20.7
OCR Slides	110.9	21.3	131.8	22.1
Documents (Extended Search, Restricted Vocab)	140.1	21.2	183.2	22.5
Correct Slides + Documents	103.6	19.9	138.6	20.3
OCR + Documents	117.4	20.7	152.9	21.5

Table 5.8: WER (%) and PPLs on the poli[Media] corpus for several adapted language models

In Figure 5.1 we can observe the weight given to each model in each one of these most relevant experiments. Some conclusions can be extracted from these data: the poli[Media] train model is the most important model i all the experiments. Slides also provide very valuable information (more than documents), and they are more important the better they are.



(a) Baseline

(b) Correct Slides

(c) OCR Slides

(d) Documents

(e) Corr. Slides + Doc.

(f) OCR Slides + Doc.

Figure 5.1: Weights of the different LM for the different experiments

Chapter 6

Conclusions and Further Work

6.1 Conclusions

Our intention developing this project was to improve the quality of automatic video lecture transcriptions, taking advantage of the fact that lecturers usually make use of slides to support their explanations.

In automatic speech recognition there two models that take part in the decoding process, the acoustic model and the language model. Our target was to create language models adapted to each video by combining different out-of-domain language model together with in-domain and specific language models trained with the text in the slides as well as with text extracted from external related documents. To measure the quality of these techniques we used the models in a real transcription task using the poli[Media] repository.

A simple yet effective method for adapting language models for video lectures using information from slides and/or documents is proposed.

Two different scenarios have been proposed: one in which the correct slides text was available and the other where only the slide image was available. Results have shown that the adaptation with correct slides and documents obtained an improvement of 4.1 absloute points in terms of WER. Surprisingly, OCR slides proved to be valuable even when they contain a large number of errors, reporting improvements up to 3 absolute WER points.

6.2 Scientific Contributions

This work derived in the following publications:

- Adrià Martínez-Villaronga, Miguel A. del Agua, Jesús Andrés-Ferrer and Alfons Juan. *Language Model Adaptation for Video Lectures Transcription.* In International Conference on Acoustics, Speech, and Signal Processing (ICASSP). Vancouver (Canada), May 2013 [18].
  - Type: Conference.
  - Ranking: Core B.
  - State: Published, pages 8450-8454.

6.3 Further Work

Adaptation techniques proposed have proven to be useful in language modelling for ASR, although regarding the obtained results we propose two major lines of research:

- Improving the quality of the OCR output
  Experiments with correct slides reported significant improvements (more than 1 absolute WER point) regarding the experiments with OCR slides. Although OCR slides will never be as good as ones transcribed manually, our current technique provides slides with around 40% of WER, which is still a large error. We think that developing a method to correct the OCR slides, based on spell checking algorithms, could drastically improve the OCR and therefore the quality of the ASR system.

- Different language modelling techniques
  All the models used in this work are n-gram language models. State-of-the-art ASR systems have been using these kind of models for almost two decades, but in the last years different approaches have been presented, beating n-gram models performance in speech recognition systems. Specially interesting are neural networks-based approaches, like the ones proposed in [23] or [24]. Neural network based models perform better than n-gram models at modelling dependencies between words as well as estimating probabilities for unseen sequences, and they work specially well for in-domain models. Some work in this field have been already performed, but we do not have results to report yet.

The good results obtained and these research lines opened, a lot of opportunities arise to keep working in an area of great utility and projection, with perspectives to provide significant improvements in video lectures transcription.

Bibliography

[1] "poliMedia: Videolectures from the Universitat Politecnica de Valencia," http://polimedia.upv.es/catalogo/.

[2] "coursera.org: Take the World's Best Courses, Online, For Free," http://www.coursera.org/.

[3] "Videolectures.NET: Exchange ideas and share knowledge," http://www.videolectures.net/.

[4] "SuperLectures: We take full care of your event video recordings.," http://www.superlectures.com.

[5] UPVLC, XEROX, JSI-K4A, RWTH, EML, and DDS, "Transcription and Translation of Video Lectures," in *Proc. of EAMT*, 2012.

[6] UPVLC, XEROX, JSI-K4A, RWTH, EML, and DDS, "trans**Lectures** ," https://translectures.eu/, 2012.

[7] Markus Ketterl, Olaf A. Schulte, and Adam Hochman, "Opencast matterhorn: A community-driven open source solution for creation, management and distribution of audio and video in academia," in *Proc. of the 11th IEEE International Symposium on Multimedia (ISM 2009)*, San Diego (USA), dec 2009, pp. 687–692.

[8] S. Katz, "Estimation of probabilities from sparse data for the language model component of a speech recognizer," *Acoustics, Speech and Signal Processing, IEEE Transactions on*, vol. 35, no. 3, pp. 400–401, 1987.

[9] Peter F. Brown, Peter V. deSouza, Robert L. Mercer, Vincent J. Della Pietra, and Jenifer C. Lai, "Class-based n-gram models of natural language," *Comput. Linguist.*, vol. 18, no. 4, pp. 467–479, Dec. 1992.

[10] P. F. Brown, S. A. Della Pietra, V. J. Della Pietra, J. C. Lai, and R. L. Mercer, "An estimate of an upper bound for the entropy of english," *Computational Linguistics*, vol. 18, pp. 31–40, 1992.

[11] H. Ney and U. Essen, "On smoothing techniques for bigram-based natural language modelling," *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, vol. 2, pp. 825–829, 1991.

[12] H. Ney, U. Essen, and R. Kneser, "On structuring probabillistic dependences in stochastic language modeling," *Computer Speech and Language*, vol. 3, pp. 1–38, 1994.

[13] R. Kneser and H. Ney, “Improved backing-off for m-gram language modelling,” *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal P rocessing*, vol. 1, pp. 181–184, 1995.

[14] Stanley F. Chen and Joshua Goodman, “An empirical study of smoothing techniques for language modeling,” *Computer Speech & Language*, vol. 13, no. 4, pp. 359–393, 1999.

[15] Andreas Stolcke, “SRILM – an extensible language modeling toolkit,” in *Proc. of ICSLP*, 2002.

[16] The transLectures-UPV Team. The transLectures-UPV toolkit (TLK) , “transLectures-UPV toolkit (TLK) for Automatic Speech Recognition,” `http://translectures.eu/tlk`.

[17] Frederick Jelinek and Robert L. Mercer, “Interpolated estimation of Markov source parameters from sparse data,” in *In Proceedings of the Workshop on Pattern Recognition in Practice*, Amsterdam, The Netherlands: North-Holland, May 1980, pp. 381–397.

[18] Adrià Martínez-Villaronga, Miguel A. del Agua, Jesús Andrés-Ferrer, and Alfons Juan, “Language model adaptation for video lectures transcription,” in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2013, pp. 8450–8454.

[19] Jean-Baptiste Michel, Yuan Kui Shen, Aviva Presser Aiden, Adrian Veres, Matthew K. Gray, The Google Books Team, Joseph P. Pickett, Dale Holberg, Dan Clancy, Peter Norvig, Jon Orwant, Steven Pinker, Martin A. Nowak, and Erez Lieberman Aiden, “Quantitative analysis of culture using millions of digitized books,” *Science*, 2010.

[20] Datong Chen, Jean-Marc Odobez, and Hervé Bourlard, “Text detection and recognition in images and video frames,” *Pattern Recognition*, vol. 37, no. 3, pp. 595 – 608, 2004.

[21] R. Smith, “An overview of the tesseract ocr engine,” in *Proceedings of the Ninth International Conference on Document Analysis and Recognition - Volume 02*, Washington, DC, USA, 2007, ICDAR ’07, pp. 629–633, IEEE Computer Society.

[22] UPVLC, XEROX, RWTH, and EML, “Progress report on masive adaptation,” Tech. Rep., October 2012.

[23] Holger Schwenk, “Continuous space language models,” *Comput. Speech Lang.*, vol. 21, no. 3, pp. 492–518, July 2007.

[24] Tomáš Mikolov, *Statistical Language Models Based on Neural Networks*, Ph.D. thesis, Brno University of Technology, 2012.

List of Figures

List of Tables