trans Lectures

3.1.1: Progress Report on Massive Adaptation

RWTH, UPVLC, XEROX, EML

Distribution: Public

trans Lectures
Transcription and Translation of Video Lectures

ICT Project 287755 Deliverable 3.1.1

October 31, 2012

SEVENTH FRAMEWORK PROGRAMME

Project funded by the European Community under the Seventh Framework Programme for Research and Technological Development.

Information Society Technologies

Project ref no.	ICT-287755
Project acronym	*trans* **Lectures**
Project full title	Transcription and Translation of Video Lectures
Instrument	STREP
Thematic Priority	ICT-2011.4.2 Language Technologies
Start date / duration	01 November 2011 / 36 Months

Distribution	Public
Contractual date of delivery	October 31, 2012
Actual date of delivery	November 18, 2012
Date of last update	October 31, 2012
Deliverable number	3.1.1
Deliverable title	Progress Report on Massive Adaptation
Type	Report
Status & version	Draft
Number of pages	48
Contributing WP(s)	WP3
WP / Task responsible	RWTH (WP) / EML (3.1) RWTH (3.2) XRCE (3.3)
Other contributors	UPVLC
Internal reviewers	J. Andrés-Ferrer, M. Mast, M. Mylonakis and M. Sundermeyer
Author(s)	RWTH, UPVLC, XEROX, EML
EC project officer	Susan Fraser

The partners in *trans* **Lectures** are:

Universitat Politècnica de València (UPVLC)
XEROX Research Center Europe (XRCE)
Josef Stefan Institute (JSI)
Knowledge for All Foundation (K4A)
RWTH Aachen University (RWTH)
European Media Laboratory GmbH (EML)
Deluxe Digital Studios Limited (DDS)

For copies of reports, updates on project activities and other *trans* **Lectures** related information, contact:

The *trans* **Lectures** Project Co-ordinator
Alfons Juan, Universitat Politècnica de València
Camí de Vera s/n, 46018 València, Spain
ajuan@dsic.upv.es
Phone +34 699-307-095 - Fax +34 963-877-359

Copies of reports and other material can also be accessed via the project's homepage: http://www.translectures.eu

Executive Summary

This deliverable reports results for work package 3 on massive adaptation for the two lecture repositories VideoLectures.net and poliMedia. The research covered by this work package is divided into three tasks, according to the core components of automatic speech recognition and statistical machine translation systems, i. e., the acoustic model (task 3.1), the language model (task 3.2), and the translation model (task 3.3).

For each language or language pair, baseline systems are developed to define initial starting conditions for adaptation. Then, for each task, existing state-of-the-art adaptation methods are investigated, and new adaptation methods are developed. Finally, the approaches are evaluated empirically on the development and testing data of the project.

Contents

1 Introduction

Today, large amounts of audio and video data covering a broad range of university lectures are available to students and interested individuals. The goal of the transLectures project is to obtain high-quality automatic transcriptions and translations for two specific collections of lecture videos, namely the videoLectures.net and the poliMedia archive.

In the past, research mainly has focused on transcribing and translating broadcast news and broadcast conversational data. In such a setting, for example the following conditions can usually be expected: (1) The audio data stem from radio and TV stations with well-defined, clean audio conditions. (2) The speakers are natives. (3) The domain of the speech data is rather general, and no special vocabulary is required. (4) Large amounts of in-domain training data are available.

When dealing with recorded lectures, often none of these conditions are met. It is thus necessary to apply adaptation methods to overcome the difficulties imposed by the lecture domain. In work package 3, we address these problems in three different tasks.

In task 3.1, we investigate the adaptation of acoustic models. Several different kinds of adaptation are considered, namely at the speaker level (CMLLR), at the lecture level (MAP adaptation), and at the lecture domain level (training with in-domain data).

Task 3.2 addresses the adaptation of language models. An efficient adaptation method is to interpolate different language models that were trained on data originating from different sources. In this connection it is important that lectures most of the time are accompanied by a set of slides, roughly summarizing a speaker's presentation by key words. In this task, we explore the benefits of including these additional data for adaptation. Finally, dynamic interpolation methods for adaptation are analyzed in detail.

Task 3.3 deals with the adaptation of translation models. Massively adaptating translation models in order to target a particular domain, such as the scientific video lecture transcriptions of *trans* **Lectures** , naturally involves the task of maximising the performance potential of large, diverse arrays of training data collections. In this report, we present two methods that explore this: lexical coverage features and multi-domain language model arrays. We use these in our translation systems to learn how to combine together the different translation options that can be extracted from 6 different training collections, in order to produce translations that better match the lecture transcription genre and the related style of language use. Moreover, for this task we also consider how we can better address the *trans* **Lectures** domain by using specially selected in-domain data, by taking advantage of the content of the slides that were used during video lecture presentations, as well as by employing data selection methods.

Sections 2 to 4 cover our work for each task in detail, section 5 concludes our proposed approaches for adaptation.

2 Massive adaptation of acoustic models

Today's state-of-the-art speech recognition sytems rely on probabilistic models to find the most likely word sequence. In mathematical notation, this means that such a system tries to maximize the quantity

$$p(w_1^N|x_1^T),$$

where w_1^N denotes a sequence of words $w_1, \ldots, w_N$, and $x_1, \ldots, x_T$ are the acoustic observations. This decision rule can be factorized and simplified using Bayes' Theorem:

$$\arg\max_{w_1^N} p(w_1^N|x_1^T) = \arg\max_{w_1^N} p(w_1^N) \cdot p(x_1^T|w_1^N).$$

The first factor $p(w_1^N)$ in the above equation is commonly referred to as the *language model*, whereas the second is denoted as the *acoustic model*.

In principle, the probability of the acoustic observations depends on the full word sequence. As this is too difficult for an efficient modelling, a simplified approach based on Hidden Markov Models (HMMs) is used. For each word from the lexicon of the recognizer, a finite state machine similar to the one depicted in Fig. 1 can be built. Each acoustic observation x_t for $t = 1, \ldots, T$

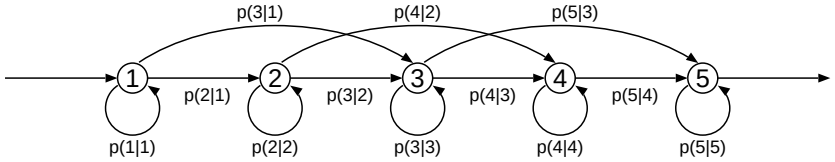

Figure 1: Example Hidden Markov Model

can then be associated with one of the states of the corresponding word automaton. The alignment of acoustic features to HMM states defines a path through the automaton, and for this path an acoustic probability can be computed in the following way:

$$p(x_1^T, s_1^T|w) = \prod_{t=1}^{T} \underbrace{p(s_t|s_{t-1}, w)}_{\text{transition probability}} \cdot \underbrace{p(x_t|s_t, w)}_{\text{emission probability}},$$

where the first factor is the *transition probability*, and the second is referred to as the *emission probability*.

In Fig. 1, the transition probabilities are depicted as arrows. Commonly, only transitions spanning zero, one, or two states are allowed. For practical applications, the transition probability is modelled by a small table of constant values that have to be tuned manually. The emission probabilities form the crucial part of the acoustic model, they can be modelled by Gaussian mixtures.

From a theoretic point of view, to obtain a valid acoustic model $p(x_1^T|w)$, the term $p(x_1^T, s_1^T|w)$ needs to be summed over all possible paths s_1^T in the HMM automaton. Instead, the Viterbi approximation is used in practice, and only the most likely path is retained by the search algorithm.

2.1 Adaptation with CMLLR

2.1.1 English lectures

Table 1 shows word error rate (WER) results for the RWTH baseline system that was used to produce initial results on the development and test data of the transLectures English task. This

system had been optimized for the recognition of podcast and broadcast conversational data, which puts emphasis on spontaneous speech and vivid discussions, sometimes with background noise.

For the first pass recognition, the acoustic features were warped using Vocal Tract Length Normalization (VTLN) as a simple form of speaker adaptation ([60]).

The basic observation underlying this normalization technique is that different speakers usually have vocal tracts of different lengths. These different lengths result in linear shifts of the formant frequencies (i. e., of those frequency regions where the acoustic energy of the voice is high). To accomodate this effect, all frequencies from the acoustic data are linearly transformed. The parameter of this transformation can be found by trying different values (grid search) and choosing the one that maximizes the log-likelihood.

For the second recognition pass, a Constrained Maximum Likelihood Linear Regression (CMLLR) adaptation approach was adopted ([13]). For the general case of a Maximum Likelihood Linear Regression (MLLR), this means that the mean μ_s and variance Σ_s of a Gaussian mixture density for an HMM state s are transformed to $\hat{\mu}_s$ and $\hat{\Sigma}_s$ such that

$$\hat{\mu}_s = G\mu_s + b \qquad \text{and} \qquad H\Sigma_s H^T$$

for adaptation matrices G, H and an adaptation vector b. In case of CMLLR, we have $G = H$. This constrained form of adaptation has several advantages: On the one hand, closed-form solutions exist to compute b and $G = H$. On the other hand, while the transform originally affects the acoustic modelling, it can be shown that it is equivalent to a transformation of the input features (feature space CMLLR) which is more convenient for implementation.

In the third recognition pass, the full language model was used to rescore word lattices created during the second recognition pass, where only a pruned language model had been used.

Table 1: WER results for the baseline English RWTH ASR system on the development and test sets.

	Dev.	Test
Baseline + VTLN	59.5	55.5
+ CMLLR	44.8	36.1
+ LM rescoring	43.0	35.9

For the transLectures project, RWTH trained an adapted English speech recognition system to improve over the baseline results. This system relied on Mel Feature Cepstral Coefficient (MFCC) and Voicedness features that were extracted every 10 milliseconds. Nine of these consecutive feature vectors were concatenated, resulting in a new feature vector of dimension 153. Using a Linear Discriminative Analysis (LDA), this dimension was reduced to 45. These are considered as short-term features as they only cover a comparatively short time interval of the acoustic signal.

In addition, RWTH extracted long-term features and concatenated them with the LDA-reduced MFCCs (TANDEM approach). For these long-term features, RWTH trained a Multilayer Perceptron (MLP). As input data for the MLP the so-called Multiresolution RASTA (MRASTA) features were used. The output of the MLP was trained in such a way that for each time frame of the acoustic input features, the phoneme state that was spoken at this time frame was provided as the desired output. As training criterion the cross-entropy error was used.

The mapping from a given time frame to the spoken phoneme state can be obtained by computing a forced alignment on the acoustic data, incorporating the manually transcribed word sequence. (For this, an initial English speech recognition system had to be used.)

RWTH used 268 hours of broadcast conversational data for training the MLP. A phoneme accuracy of 57.9 % on cross-validation data was obtained. Instead of directly using the phoneme posterior probabilities (which corresponds to a vector of dimension 133 as there exist 133 phoneme states in the training lexicon), RWTH extracted the corresponding activations of a *bottleneck* layer directly preceding the output layer. This layer only has size 75. Therefore this procedure can be interpreted as non-linear dimensionality reduction. More details on the training of MLP features can be found in [39].

In previous experiments, RWTH found that best results can be obtained when combining this non-linear dimensionality reduction with an additional linear dimensionality reduction, thereby reducing the dimension of the MLP features even further. Applying a Principal Components Analysis (PCA), the final dimension was set to 100.

For the acoustic model, 763 hours of manually transcribed speech data were used, including 102 hours from the European Parliament Plenary Sessions (EPPS), the aforementioned Broadcast Conversational (BC) data as well as 393 hours from the publically available HUB4 and TDT4 corpora (cf. Table 2).

Table 2: Acoustic training data for the RWTH English ASR system

Data source	Duration	Type
Quaero	268 h	Broadcast Conversational data, podcasts
EPPS	102 h	European Parliament Plenary Session transcriptions
Hub4 + TDT4	393 h	Broadcast News data

Due to the fact that large amounts of training data available for the English language, at least for acoustic training including the 20 hours of transcribed transLectures acoustic data did not seem promising to RWTH. Instead, RWTH made use of the transLectures transcriptions for language model training (see below). In addition, RWTH decided to include the EPPS data for training because they stem from a strongly related domain, and much more data are available from this source.

A Gaussian Mixture Model (GMM) was trained on these data with a globally pooled diagonal covariance matrix, resulting in a total of 1.2 million densities. The number of possible triphone states was reduced to 4501 states by clustering using a Classification and Regression Tree (CART), see[64].

For the recognition lexicon, the phoneme set was based on the BEEP dictionary (which is available from Cambridge University). The vocabulary included the most frequent 150,000 words from the language model training data. As the language model data consist of a multitude of corpora having different sizes, the word frequencies were normalized over the available corpora.

Table 3: WER results for the improved English RWTH ASR system.

	Dev.	Test
Baseline + VTLN	44.5	34.0
+ CMLLR	39.4	28.4
+ LM rescoring	38.2	27.9

Table 3 states the current results of RWTH's adapted English ASR system. Huge improvements can be observed which are mainly caused by three different factors: Inclusion of state-of-the-art acoustic MLP features, adaptation to the lecture domain by including EPPS and BN data, and the incorporation of in-domain training data into the language model.

2.1.2 Slovenian lectures

RWTH's phoneme set for the recognition of Slovene lectures is based on the one used in the LC-STAR project. In addition to 39 phonemes it contains one phoneme for silence, one for hesitation and one "garbage" phoneme for all kinds of noise. A grapheme-to-phoneme (G2P) model was trained on the LC-STAR lexicon (~100k words) and used to create the training lexicon.

The transcription of the training data was preprocessed as follows:

1. convert to lower case
2. collapse all types of hesitation tags to one
3. remove all other tags and punctuation marks
4. remove segments marked with `ignore_time_segment_in_scoring`
5. map incomplete words to `[unknown]`
6. use original spelling of foreign words

The acoustic training was performed on 34 lectures with a total duration of approx. 27 hours of speech downsampled to 16 kHz. The initial Gaussian mixture model (GMM) with a globally pooled diagonal covariance matrix was trained on 33-dimensional features (16 MFCC plus first order derivative and first component of the second order derivative). The underlying Hidden Markov model (HMM) has a three-state left-to-right topology and allows for loops and skips. Also across-word transitions are taken into account. The monophone system was trained with respect to the maximum likelihood (ML) criterion by means of the Expectation Maximization (EM) algorithm. The number of Gaussians was increased in every iteration by splitting. The best alignment was then used for training of a triphone system. The number of triphone states was reduced to 4500 generalized states by tying with a Classification And Regression Tree (CART). The CART labels were used as classes to estimate an LDA transform that maps nine consecutive MFCC feature vectors (four frames left and right context) to 45 dimensions. The ML training of the triphone system was performed on the LDA-transformed features, resulting in approx. 440k Gaussians.

Table 4: Corpus statistics

Corpus	Number of lectures	Duration [h]
train	34	27.0
dev	4	3.1
test	4	3.3

A second system was trained in the same fashion as the baseline system on the MFCC features warped for Vocal tract length normalization (VTLN). The evaluation was performed on a development and a test set. The corpus statistics are given in Table 4. The results for the first two systems are given in Table 5.

During the recognition a 4-gram language model (LM) was used in all systems. See Section 3.2 for details on language modeling. The recognition lexicon has been compiled from 400k words that occur in all available text data most freqently. This way the ratio of out of vocabulary (OOV) words on the development set is approximately 2.4%.

Table 5: Recognition results (word error rates in %)

System	Pass	dev	test
Baseline MFCC	1	38.8	59.2
(a) MFCC + VTLN	1	38.6	57.6
(b) CMLLR in training	1	37.8	57.2
(c) CMLLR in training+recognition	2	35.3	47.3

The transcription obtained from the first pass recognition was used for performing an unsupervised adaptation with CMLLR, aligning the features to the recognition output. The results of the second pass recognition performed on the CMLLR adapted features are given in Table 5.

While on the development data the relative improvement by CMLLR is in the usual range of about 10 %, the gain is much larger on the test data. This indicates a mismatch in the acoustic conditions in the training and test data which is normalized by the CMLLR transform.

Such a mismatch also exists on the level of the language model, partly because of the wide spectrum of topics covered in the lectures.

Moreover, the lecturers speak freely ("spontaneous speech" as opposed to "prepared" or "scripted"), while the language model is trained on written texts like books and articles. Lastly, Slovene is characterized by a large number of dialects, which makes the automatic transcription of lectures with many different speakers a difficult task.

2.1.3 Spanish lectures

The various sources of the audio training data are summarized in Table 6

Table 6: Audio data for transLectures Spanish ASR system

Data source	No. of hours	Type
transLectures	100	spontaneous, clean
QUAERO	170	News and podcasts, background noise/music
EPPS	90	Parliament speeches, clean
Hub4	30	American Spanish

Some pre-processing steps are applied to all transcriptions; the numbers are expanded to their word form, and for alternate terms (i.e. of the form /phrase 1/phrase 2//) the first term is retained.

The acoustic model training for the transLectures Spanish ASR system was done with the RWTH ASR software. The basic phoneme set consists of 31 language phonemes, plus one silence and four non-language sound phonemes (representing noise, hesitation, articulation and breath). The acoustic input features are three-layer Multilayer-Perceptron (MLP), which in turn have been trained on a concatenation of MFCC, PLP and Gammatone features. The 9 output frames of the MLP are then concatenated together and then transformed by a linear discriminant analysis (LDA) transformation to 45 dimensions, and similarly transformed 45 dimensional MFCC features are concatenated to it. The resulting 90 dimensional input feature vectors (with a resolution of 10 milliseconds per vector) are used as input features.

A previously trained QUAERO Spanish ASR system is used to generate an initial phoneme-to-feature-vector alignment for the audio data. The triphones are clustered into 4500 classification and regression tree (CART) states. From this alignment, Gaussian mixture means

are estimated for each of these CART states (up to 256 Gaussian densities per state), and a pooled diagonal covariance matrix. Using this newly calculated acoustic model, the phoneme-to-feature-vector alignments are refined. Speaker dependent constrained maximum likelihood linear regression (CMLLR) matrices are calculated for each speaker according to the speaker labels as provided by the transcriptions. After adding these CMLLR matrices to the feature extraction pipeline, new Gaussian mixtures are estimated to the same resolution as before. The Gaussian mixtures are then converted to their log-linear parameter form λ_s and α_s for an HMM state s, so that the state posterior becomes

$$p_\Lambda(s_t|x_t) = \frac{\exp\left(\lambda_{s_t}^\top x_t + \hat{\alpha}_{s_t}\right)}{\sum_{s'} \exp\left(\lambda_{s'}^\top x_t + \hat{\alpha}_{s'}\right)} \tag{1}$$

for a fixed alignment s_1^T and acoustic observations x_1^T. These new log-linear parameters are then discriminatively trained according to the minimum phone error (MPE) objective function. Details of the Gaussian to log-linear conversion and the MPE optimization procedure can be found in [56]. The discriminative training resulted in about 1% absolute WER improvement on the development and test data.

Table 7: WER (%) for transLectures Spanish

transLectures Corpus	dev-2012	eval-2012
1st-pass	20.3	21.4
with CMLLR	16.7	18.2
with CMLLR (log-linear)	15.7	17.1

In addition to RWTH, UPVLC also investigated the adaptation of acoustic models for the Spanish poliMedia repository.

For this task, UPVLC focused on the development of a Spanish large-vocabulary ASR system for the poliMedia task. Specifically, UPVLC focused here on the development and adaptation of acoustic models, which have been modeled using Hidden Markov Models (HMMs). HMMs have been built using the AK toolkit [1]. This toolkit is released under a free software license and provides similar features to other HMM toolkits as HTK [63] or RASR [42]. However, in this task UPVLC extended the AK features to support several widespread ASR techniques as triphone HMMs or cross-word recognition, as well as some adaptation techniques.

UPVLC has focused on transcriptions of the Spanish lectures from poliMedia. At the beginning of the project, 704 manually transcribed and segmented poliMedia lectures were provided for the development of Spanish ASR systems. From these, 49 lectures where extracted to define development and test sets, leaving the rest of lectures for training purposes. Table 8 summarizes the main statistics of the poliMedia speech database.

In order to develop Spanish acoustic models, training speech segments were first transformed into 16KHz, and then parameterized into sequences of 39-dimensional real feature vectors. In particular, every 10 milliseconds 12 Mel Frequency Cepstral Coefficients (MFCC), the log-energy, and their first and second order time derivatives were calculated. Regarding the transcriptions, 23 basic phonemes were considered plus a silence model. Words were transformed into phonemes according to the Spanish pronunciations rules. The resulting phoneme transcriptions also include speech disfluencies as hesitations, incorrect pronunciations, etc.

The previously extracted features were used to train a tied triphone HMM with Gaussian mixture models for emission probabilities. The training scheme is similar to the one described in [63]. Firstly, each Spanish phoneme is modeled as a three state HMM with a conventional

Table 8: poliMedia Speech Corpus

Set	Lectures	Time (h)	Phrases	Running Words	Vocabulary size
Train	655	96	41.5k	96.8k	28k
Development	26	3.5	1.4k	34k	4.5k
Test	23	3	1.1k	28.7k	4k

Table 9: PPLs, OOVs (%) and WER (%) using several language models for the baseline Spanish ASR system.

Language Model	Development			Test		
	PPL	OOV	WER	PPL	OOV	WER
3-gram	298.0	4.5	36.4	317.6	5.2	36.2
4-gram	293.2	4.5	36.3	313.0	5.2	36.0
3-gram (External)	186.1	2.2	28.7	244.1	3.0	30.5
4-gram (External)	179.7	2.2	28.1	240.2	3.0	30.3

left to right topology. Secondly, these HMMs models are used to initialize triphones which are extracted by modeling each phoneme with its context. Thirdly, an automatic tree-based clustering based on manually crafted rules is used to tie the original states. As a product of the previous process, a tied triphone HMM with only one Gaussian component per state is obtained similarly to [64]. Finally, mixture components in each state are repeatedly split using an iterative training process. During all training steps the Baum-Welch algorithm was used to estimate the model parameters that maximizes the log-likelihood for the input data [40, 63]. In summary, UPVLC obtained 1745 HMM triphones with 3 342 tied-states having up to 64 Gaussians per state. The training configuration was tuned on preliminary experiments using the development set.

In order to evaluate the trained acoustic models, UPVLC built n-gram language models using the SRILM toolkit [52]. Two kinds of language models have been trained. The first approach is a simple n-gram model trained only with the utterance transcriptions of the poliMedia speech database. The second approach is fully described in Section 3.2.2 as the baseline. In summary, this approach computes a linear mixture of language models trained with the poliMedia transcriptions along with other external resources described in Table 23, see Section 3.2.2 for further information. For each case, UPVLC compared results for both 3-grams and 4-grams. In the case in which only poliMedia was used, the size of the vocabulary is about 28K, whereas for the external data, the vocabulary was restricted to the 60K most frequent words. Regarding to the poliMedia transcriptions, UPVLC used what the speaker said instead of the corrected transcription. Hesitations were considered a special word for the language model. Finally, the well-known Viterbi algorithm was used to automatically recognize the videos. Recognition parameters were tuned on development as for instance the grammar scale factor which was set to 11, or the word insertion penalty, which was set to 0.

Table 9 reports WER results on development and test, for all the proposed language models. It is observed that when external text corpora is used a big improvement of about 6 points of WER is obtained. The order of the language models slightly improves the WER when it is increased from 3 to 4. The best result, 30.3%, is obtained with a 4-gram model.

Two speaker adaptation techniques were implemented in order to improve our best baseline model (4-gram language model with external data): fast Vocal Tract Length Normalization [60], and constrained MLLR (CMLLR) features [50, 16].

Table 10: WER (%) using several adaptation techniques on baseline Spanish ASR system.

	Development	Test
Baseline	28.1	30.3
VTN	26.2	28.8
CMLLR (GMM-Diag.)	27.8	30.1
CMLLR (HMM-Diag.)	27.3	29.7
CMLLR (HMM-Full)	22.3	24.8
CMLLR (HMM-Full 2nd Pass)	22.2	24.6

In order to apply the correct transformation, it is necessary to determine a suitable warping factor for each speech segment in training and recognition. In contrast to other methods, in fast VTLN the selection of the warping factor during recognition is carried out without the need of a preliminary transcription. Specifically, the selection is carried out using a Gaussian mixture model which is trained using the normalized training data [60]. In the recognition process, UPVLC used a Gaussian mixture model of 128 components to select the optimal warping factor, whereas, in the training process, our baseline HMM triphone model with only one Gaussian per state was used. In both training and recognition stages, the warping factor was optimized at speech segment level. Results are shown in Table 10. As expected, the use of fast VTLN outperforms our baseline model. In particular, UPVLC obtained a WER of 28.8%, 1.5 points better than the baseline.

CMLLR is a technique for finding a linear transformation of the parameters of either a Gaussian mixture model, or the Gaussian mixtures of a conventional HMM. This transformation is aimed at obtaining a data representation independent of the speaker.

In a first experiment, UPVLC compared two different target models: a mixture Gaussian of 512 mixture components (GMM), and our baseline triphone HMM with only one Gaussian per state. It is worth noting that in order to use a HMM as the target model for CMLLR a preliminary transcription is required. In our case, UPVLC used the transcriptions of the baseline system. This first CMLLR experiment was carried out using diagonal transformation matrices. UPVLC clustered each video in training and tested as a single CMLLR cluster with its own transformation. This decision was taken in light of poliMedia characteristics, since in most of the poliMedia lectures there is only one speaker per video.

Results are shown in Table 10. In the GMM approach a small improvement of only 0.2 points (30.1%) over the test baseline was obtained. Similarly, for the HMM case UPVLC achieved an improvement of 0.6 points (29.7%). In both cases the improvement obtained was small, however the HMM target model had a better performance.

A second experiment was carried out using full CMLLR transformation matrices instead of diagonal ones. Full CMLLR transformation matrices are more powerful than diagonal matrices since they can model covariances. The result, using the HMM target model, is also shown in Table 10. It is observed that for the case of full matrices, a much larger improvement is obtained than that of the diagonal case.

Finally, given the significant improvement obtained with full CMLLR (6 WER points), UPVLC performed a second pass with the transcriptions obtained by the previous CMLLR model. This second pass was performed with the expectation that improving the quality of the transcriptions used to compute the transformation may improve the transformation itself. After this new transformation is obtained, the same system is used to transcribe the videos again. As it is observed in Table 10 the second pass obtains only a slight improvement yielding the best result of 24.6% WER points.

2.2 MAP adaptation of English lectures

2.2.1 Introduction

Acoustic adaptation has been investigated in two different scenarios, namely the supervised adaptation of a general purpose acoustic model towards the lectures domain, and the unsupervised adaptation of a lectures specific acoustic model towards a particular lecture.

The first scenario is considered relevant for the rapid prototyping of initial models for the lectures domain which can help in further data collection and transcription efforts needed for the development of improved models. The second scenario aims at the creation of high quality manuscripts for individual lectures. The investigations carried out in this direction focus on the amount of data that is needed for adaptation and on a processing setup that foresees the decoding of thousands of hours of lecture data.

The proper choice of an adaptation technique depends on the amount of adaptation data that is available in a particular scenario. Maximum a posteriori (MAP) adaptation [15] is one of the most efficient techniques if a large amount of data, say more than 10 – 15 minutes, is available.

The underlying principle of MAP adaptation is to consider the *weighted* log-likelihood function F_{MAP}

$$F_{\mathrm{MAP}}(\lambda) = \sum_{t=1}^{T} \log\big(p_{\mathrm{AM}}(x_t|\lambda)p(\lambda)\big),$$

where λ denotes the parameters of the acoustic model p_{AM} (i. e., mean vectors and covariance matrices in case of a Gaussian mixture model). Here, x_1^T denotes acoustic adaptation data to which an existing acoustic model shall be fitted. Under certain assumptions, closed-form solutions exist to compute the adapted mean vectors and covariance matrices.

Originally developed for speaker adaptation, it has been successfully used for the adaptation of acoustic models towards particular dialects, languages, and/or acoustic conditions in the past. MAP is the method of choice for both cases described above, because in the transLectures scenario one can safely assume the availability of a large adaptation corpus — a single lecture is usually much longer than the required 10 – 15 minutes. Moreover, its conceptual simplicity allows the rapid development of adaptation tools for the various acoustic models needed in today's multi-pass speech recognizers.

The EML transcription engine implements a two-pass recognition strategy that can also be found in several (open source) toolkits; see, for example, [43, 47]. After model based speech/silence segmentation, speaker independent, gender (or vocal tract length) normalized acoustic models are used for the creation of a preliminary transcript in a first recognition pass. This transcript is used for the online computation of a speaker- specific affine feature transformation that is required by the speaker adaptive acoustic models employed in a second recognition pass. The second pass recognition results can be further improved, e.g. by a consensus network algorithm [28], or by word lattice rescoring, if processing time requirements permit.

Tools have been developed in the reporting period for MAP adaptation of the various models needed in the recognition process sketched above:

- the (coarse, monophone) speaker independent acoustic models used for speech/silence segmentation,
- the Bayesian classifier for text independent determination of a warping factor for vocal tract length normalization [61],
- the vocal tract length normalized acoustic models used in the first recognition pass,

Table 11: Properties of EML's acoustic models used for massive adaptation.

EML English AMs	lectures AM	general purpose AM
training data (16 kHz)	100 hours	900 hours
HMMs (triphone context)	4000	4000
1st pass densities	680k	600k
2nd pass densities	550k	430k

- the (coarse) acoustic model that serves as a target model in speaker adaptive training and recognition [51], and
- the speaker adaptive acoustic models employed in the second recognition pass.

During development, special focus has been given to the integration of the adaptation tools into a (semi-)automatic workflow that allows the direct use of lecture transcripts produced by the EML transcription platform for unsupervised adaptation, or the seamless use of manually corrected transcripts for (lightly) supervised adaptation. Another important aspect has been the integration of the developed tools into the acoustic model training environment used by EML's language partners and customers for the development of acoustic models for several languages and/or target domains. Here, massive parallel processing is a key feature for the rapid creation or adaptation of acoustic models from real application data.

2.2.2 Experimental Setup

Table 11 gives some details about the acoustic models for single pass and two pass recognition employed in our study. The lecture specific acoustic models are created from approximately 100 hours of audio that were agreed upon for training in the transLectures consortium. In contrast, the general purpose acoustic models used in the supervised adaptation scenario are created from an EML in-house English speech data base available at EML. It consists of roughly 900 hours of speech from many sources, but does not include any of the first models' training data that is of restricted (research only) use. Both acoustic models make use of the same 45-dimensional feature vector which results from the application of a linear discriminant analysis (LDA) to a sliding window of 9 consecutive vectors consisting of 16 MFCCs and a voicing feature. The MFCCs are computed in a vocal tract length normalized feature space that uses 13 warping factors in a range from 0.88 to 1.12. CMLLR is applied for the training of acoustic models for the second decoding pass.

Both the lectures and the general purpose acoustic models use a single state HMM inventory that consists of 4000 context dependent triphone HMMs. In both cases the HMMs are shared between the acoustic models for the first and second decoding pass in order to provide a reasonable small memory footprint. Whereas model training originally aimed at the same number of densities in both cases, it turned out that the general purpose acoustic model performed better with fewer densities in both the first and second pass model.

The transLectures development set for the English language consists of audio from 4 lectures from different speakers and serves as adaptation data for the supervised adaptation of the general purpose acoustic models. Both baseline and adapted models are tested against the transLectures test set, which also consists of audio from 4 lectures. In case of the unsupervised adaptation towards a particular lecture, different portions of the test set were used for the offline adaptation of the baseline models, see below. Finally, a "general English", word based 4-gram language model with a total of approximately 15 million n-grams was used in all experiments. No efforts have been made to adapt the language model towards the domains covered in the lectures data used here.

Table 12: Supervised adaptation of a general purpose acoustic model; single pass recognition word error rates for different adaptation weights.

General AM	1st-base	1st-5	1st-50	1st-100	1st-500
Lecture 1	63.3	62.8	61.9	62.0	63.3
Lecture 2	35.8	35.6	35.5	35.4	36.8
Lecture 3	50.3	50.2	49.9	49.4	52.6
Lecture 4	48.3	48.2	48.0	48.3	49.0
Total	48.7	48.4	48.2	48.1	49.7

Table 13: Supervised adaptation of a general purpose acoustic model; word error rates for single and two pass recognition.

General AM	1st-base	2nd-base	1st-100	2nd-100
Lecture 1	63.3	60.1	62.0	57.9
Lecture 2	35.8	35.8	35.4	36.0
Lecture 3	50.3	48.2	49.4	47.0
Lecture 4	48.3	44.2	48.3	44.6
Total	48.7	46.1	48.1	45.6

2.2.3 Supervised Adaptation

Table 12 shows word error rates (different lectures from the transLectures English test set. The overall baseline WER is 48.7 percent; it reduces to 48.1 percent for the best adaptation weight (100) and increases to 49.7 percent, if too much weight is given to the adaptation data.

The best adaptation weight has also been used for the adaptation of the speaker adaptive models employed in the second recognition pass; results are given in Table 13.

It is worth noting that for the two lectures with lowest WER (Lecture 2 and Lecture 4) there is a small degradation when using the MAP adapted models for the second recognition pass. However, a closer inspection shows that this degradation is due to an increased WER for some background speakers, whereas results for the main speaker (lecturer) remain at least as good as for the baseline models.

2.2.4 Unsupervised Adaptation

The unsupervised adaptation scenario aims at the rapid creation of the best possible transcript for a given lecture. The idea pursued in the experiments described here is to replace the standard two pass recognition strategy by a setup that collects some amount of adaptation data in an unsupervised manner, adapts the CMLLR models used in the second recognition pass, computes the usual feature space transformation, and finally runs recognition with the adapted acoustic CMLLR models. The interesting quantity in this setup is the amount of adaptation data: on the one hand it should be as much as possible in order to achieve the best word error rate for each show, on the other hand it should be as little as possible in order to guarantee a high throughput if hundreds of lectures have to be transcribed.

Table 14 gives the baseline results for the transLectures test set and the lecture specific acoustic models, showing a baseline WER of 54.2 percent for single pass recognition and a WER of 47.9 percent for the two pass recognition setup. Note the degradation compared to the general acoustic model above, which incorporates much more acoustic data.

Table 14: Baseline results (WER in percent) for the lecture specific acoustic models.

Lecture AM	1st-base	2nd-base
Lecture 1	63.4	53.9
Lecture 2	37.8	36.2
Lecture 3	50.2	44.8
Lecture 4	62.5	54.2
Total	54.2	47.9

Table 15: Word error rates for adapted second pass (CMLLR) acoustic models (unsupervised adaptation with different portions of the same lecture).

Lecture AM	2nd-base	2nd-10	2nd-25	2nd-50
Lecture 1	53.9	52.5	51.7	51.6
Lecture 2	36.2	36.1	35.3	35.3
Lecture 3	44.8	43.9	36.0	35.8
Lecture 4	54.2	52.7	42.2	52.5
Total	47.9	47.3	40.6	45.3

Preliminary transcripts of different length have been created by decoding 10, 25, and 50 percent of each lecture, corresponding to roughly 5, 12.5, and 25 minutes of audio. Segment confidence scores computed by the recognizer were then used for the selection of data for unsupervised adaptation. Following a rule of thumb, thresholds were set in a fashion that discarded the 50 percent worst scoring segments of each (partial) lecture, resulting in a net amount of roughly 2.5 – 12 minutes of audio data per lecture for adaptation. Finally, this data was used for both MAP adaptation of the CMLLR models and the computation of a single speaker transform for each lecture. Table 15 compares the word error rates for the standard two pass recognition with baseline acoustic models and single pass recognition with CMLLR-adapted models (using different amounts of adaptation data). While the degradation for Lecture 4 (2nd-50) deserves a closer examination, the sketched approach seems feasible. However, further implementation work (dealing with the integration into the EML transcription platform) is needed in order to judge the approach in terms of system throughput.

3 Massive adaptation of language models

3.1 Adaptation with in-domain data

During the recognition of the Slovenian lectures, RWTH used a 4-gram language model (LM) in all systems. The transcriptions of the acoustic training data have been interpolated with some freely accessible texts collected on the web. The interpolation weights have been optimized on the development set. The resulting perplexities on the development set are given in Table 16.

Table 16: Slovenian language model perplexities (lexicon size: 400k)

Text source, content	Sentences	Running words	Perplexity (dev)
AM training data (lectures)	6.2k	136k	978
Web data (books, articles)	1.5M	74.6M	518
Interpolated LM	1.6M	74.7M	468

For the Spanish lectures, various sources of language model text are summarized in Table 17

Table 17: LM data for transLectures Spanish ASR system

Data source	Running words
Translectures transcriptions	1M
Other transcriptions	8M
QUAERO	600M
Gigaword corpus	1000M

The language model is 5-gram with Kneser-Ney discounting to account for unknown n-grams. There are 320K words in the lexicon, with pronunciation variants for many frequently used words. The initial pronunciation model is based on a 60K-word lexicon for the TC-STAR project. A grapheme-to-phoneme (G2P) model trained on this initial lexicon is used to create the pronunciations for those words which are present in the LM text but not in the initial lexicon. This works quite well in Spanish language, because the orthography to pronunciation rules are more consistent as compared to many other languages.

We use the measure of perplexity for adapting the language model. It is defined as the inverse probability indicating between how many different words the system chooses at every time frame. The lower the perplexity the better. The formal definition is:

$$\mathrm{PPL} = p(w_1^N)^{-\frac{1}{N}} = \left[\prod_{n=1}^{N} p(w_n|h_n)\right]^{-\frac{1}{N}} \tag{2}$$

for a text w_1^N of length N.

Table 18: Perplexity of development corpus

Data source	ppl (dev-12)
Gigaword + QUAERO + old-transcriptions	274
Gigaword + QUAERO + old-transcriptions + transLectures-transcription	195

As can be seen in Table 18, the perplexity of development corpus for the initial language model (i.e. the one containing all the sources other than transLectures transcriptions) is significantly reduced by adding the transLectures transcriptions. This is inspite of the fact that the size of transLectures transcriptions data is tiny as compared to the other sources. Therefore we can assume that adding more and more in-domain LM data will make the perplexity (and hence the WER) better. The OOV rate is 1.6%.

3.2 Adaptation based on lecture slides

3.2.1 English lectures

For the initial automatic transcription of the English video lectures a base language model was used that is trained on transcribed lectures, subtitles, parliament speeches and general purpose text.

The video lecture repositories contain not only the recorded lecture but also corresponding slides. In Task 3.2 EML uses this additional information to adapt the language model. The accompanying slides for the lectures were extracted and preprocessed in Task 2.1. Furthermore relevant material was downloaded from the web and preprocessed.

The EML English base language model was created from the text material as agreed on in the Scientific Evaluation Proposal and additional EML in-house data resulting in a corpus of about 350 million running words. Details on the composition of the corpus are shown in Table 19.

Table 19: English language model corpora

Corpus	Sentences	Running words
EPPS	52.477	780.616
EuroParl corpus[1]	2.358.325	57.809.769
JRC-Acquis corpus	3.839.309	60.985.421
OPUS-OpenSubs	2.692.137	17.288.604
TED	11.463	91.013
VL lectures manually transcribed	13.641	143.044
VL lectures with subtitles	364.597	2.069.683
EML General Purpose Text	9.532.057	185.350.475
EML Technical Texts and Lectures	1.509.503	21.434.875
Total	20.373.509	345.953.500

As baseline an EML-created general purpose vocabulary of 185.000 words was used. This vocabulary was enriched with the most frequent words with varying threshold from the above mentioned text material resulting in a total vocabulary of 235.000 words. The thresholds reflect the relevance and reliability of the corpus.

One source of data for adaptation are the slides that accompany the lectures. We decided to only use the material from ppt-slides, because the quality of text extracted from pdf- and jpg-slides is rather poor. Only in case there are no ppt-slides available for a specific lecture EML used the pdf-slides. Furthermore material was downloaded from the web by search requests about the author, category and keywords of the lectures (described in Deliverable 2.1). Table 20 shows the amount of data used for the adaptation experiments.

The base model is a 4-gram language model trained using modified Kneser-Ney smoothing. This model was adapted with the different corpora from Table 20 resulting in 10 adapted

Table 20: Adaptation Corpora

Material	Sentences	Running words
All material downloaded	1.307.616	22.454.212
All available ppt-slides	427.557	3.554.763
All ppt-slides per category		
Material	Sentences	Running words
Mathematics	8.794	48.783
Computers	14.666	103.700
Business	33.180	212.738
Society	29.184	157.125
ppt/pdf-slides per lecture		
Material	Sentences	Running words
webstart08_leitersdorf08_esvc (ppt)	234	1.122
wict07_divjak_deome (ppt)	563	3.979
acs07_moulines_mcm (pdf)	652	8.719
apr08_culver_aws (ppt)	201	962

Table 21: Results for the test and development set for adaptation with all material

Development set			
Model	Perpl	OOV (%)	WER (%)
Baseline	632	0.65	60.6
Adapt. with all ppt-slides	454	0.35	55.9
Adapt. with all slides and downloads	421	0.21	55.1
Test set			
Model	Perpl	OOV (%)	WER (%)
Baseline	395	0.45	46.0
Adapt. with all slides	317	0.18	42.1
Adapt. with all slides and downloads	308	0.15	42.1

language models. For each corpus the vocabulary was extracted and all unknown words occuring more than once were added to the vocabulary. Subsequently a 4-gram language model was trained from each corpus and linearly interpolated with the base model weighting the base model with 0.75 and the new model with 0.25.

Table 21 shows the baseline results and the results for the adaptation with all ppt-slides and finally with all ppt-slides and the downloaded material.

Most of the benefit for WER, perplexity and OOV can be attributed to LM adaptation with data extracted from the lecture accompanying slides, while the incorporation of more data downloaded from the web has little or no effect.

Table 22 shows the baseline results, the results for the adaptation with all slides and downloads (a), the results for the adaptation with the ppt-slides from all lectures with the same category (b) and with the ppt-slides from the specific lecture only (c).

For all but one lecture the adapation with all slides and downloaded material (a) gives slightly better results than the adaptation with the slides for the specific category (b).

As expected the lecture specific adaptation gives for 3 lectures the best results.

Table 22: Results for the single lectures of the test set for adaptation with all material, with all ppt-slides and with the ppt-slides for the specific lecture

acs07_moulines_mcm (Mathematics)	Perplexity	OOV (%)	WER (%)
Baseline	549	0.3	59.0
(a) Adapt with slides and downloads	347	0.07	53.3
(b) Adapt with slides from category	365	0.21	54.0
(c) Adapt with slides from lecture	341	0.2	52.4
apr08_culver_aws (Computers)	Perplexity	OOV (%)	WER (%)
Baseline	319	0.77	41.5
(a) Adapt with slides and downloads	260	0.27	38.3
(b) Adapt with slides from category	286	0.47	39.0
(c) Adapt with slides from lecture	284	0.47	38.7
webstart08_leitersdorf_esvc (Business)	Perplexity	OOV (%)	WER (%)
Baseline	371	0.45	34.7
(a) Adapt with slides and downloads	316	0.15	31.5
(b) Adapt with slides from category	331	0.23	31.7
(c) Adapt with slides from lecture	311	0.24	31.0
wict07_divjak_deome (Society)	Perplexity	OOV (%)	WER (%)
Baseline	456	0.07	54.1
(a) Adapt with slides and downloads	336	0.03	49.5
(b) Adapt with slides from category	320	0.04	47.0
(c) Adapt with slides from lecture	294	0.05	45.8

3.2.2 Spanish lectures

The transLectures project offers several adaptation oportunities for the language models. For instance, the raw text of the slides, when available, can be used for adapting the language models. Unfortunately, many times slide texts have not been acquired, and, then, they have to be extracted from images in which the text information is not available.

During this period, the UPVLC has focused on adapting the language models in two scenarios. On the one hand, a very optimistic scenario is considered in which a perfect transcription of the slide texts is available together with the synchronisation between the video and the text. This is an unrealistic case for some repositories, in which the system does not have the original slide documents as it is the case for most of the poliMedia videos. It may also be the case that the original document is available with the correct text, but no synchronisation between this text and the video is available. On the other hand, UPVLC considers a more adverse situation in which the slide information is automatically extracted from the videos. This introduces recognition errors in the slide text as well as in the synchronisation, that might ruin the potential improvement that slides may introduce.

The UPVLC language model baseline is an n-gram language model composed by several n-gram models which were trained in different copora. 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:

$$p(w|h) = \sum_{i=1}^{I} \lambda_i p_i(w|h) \tag{3}$$

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 [19].

Table 23: Basic statistics of corpora used to generate the LM

Corpus	Sentences	Running words	Vocabulary size
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 24: Basic statistics of poliMedia corpus.

Set	Sentences	Running words	Vocabulary size
Train	41.5K	96.8K	28K
Development	1.4K	34K	4.5K
Test	1.1K	28.7K	4K

Each individual language model was trained using the SRILM [52] toolkit in a different corpus. Table 23 summarises the main statistics of the used corpora. In addition, a language model was trained from the Google counts corpus [31]. Finally, the poliMedia corpus was also used as the in-domain corpus, see details in Table 24. All n-gram models were smoothed with modified Kneser-Ney absolute interpolation method [20]. The models used order 4, i.e. 4-grams, which were also pruned such that the perplexity increased by less than 2^{-10}. Perplexities obtained for each of these individual models are reported in Table 25.

The baseline, as well as all further experiments have been conducted using the acoustic model with CMLLR feature normalisation (full transformation matrices), which reports one of the best UPVLC results (see Table 10). More details on this acoustic model are given in the previous Section 2.1.3.

Several experiments were carried out in order to assess the improvements that can be obtained with the text of slides in the poliMedia corpus (see Table 24). In all experiments, an additional language model computed from the text of the slides was introduced in the linear mixture described in Equation (3). It is worth noting that this slide-dependent language model varies from one video to another. However, we optimised the linear interpolation weights for all the n-gram models including the varying slide-dependent language model so that the perplexity is minimised on the poliMedia dev set.

As discussed above we considered two scenarios which were implemented by two sets of slides transcriptions:

- human transcribed slides
- automatic transcription obtained with an OCR

In the first case, a human annotator transcribed the text of development and test slides so that they were perfectly synchronized as well as error free.

For the second case, automatic transcription were extracted from the videos in two steps. Firstly, each video was segmented into its corresponding slides by a very simple criterion. We

Table 25: Perplexities on the development and test sets computed by the language models trained in each corpus.

	Perplexity	
Corpus	dev	test
EPPS	543.7	710.8
news-commentary	636	747.7
TED	615.6	521.2
United Nations	754	802.9
Europarl-v7	460.6	605.7
El Periódico	450.2	545.9
news (07-11)	358.9	747.6
UnDoc	544.9	802.8
Google	1370.3	1954.8
poliMedia train	317.9	332.5

counted the number of changing pixels from one frame to another and if the current frame goes beyond a given threshold, the next frame is considered to be a new slide. Once the slides have been detected, OCR was perfomed using the free OCR software Tesseract [45].

When Tesseract receives an image from the slide, it is binarized by adaptive thresholding, afterwards character outlines are estimated, then text lines detected and finally, the characters in the image are recognized. However, Tesseract is aimed at recognising text documents with the same font and regular structure. Since our goal is to transcribe slides, this produces several problems such as poor binarization. In order to amend this problem, the output is improved by a very simple spell-checker. All in all, the WER of this automatically obtained slides is 64.48%. Although this WER is high, we will see in the experiments that the ASR performance is significantly increased by using this automatic slides.

Results in terms of PPL and WER are summarised in table 26. The first row depicts the results obtained with the baseline language model without any information from the slides. The second row, adds a 3-gram language model trained with the human annotated slides for each video. It is observed that error free slide text heavily improves performance. Similarly to the second row, the third row adds a slide-dependent 3-gram to the linear interpolation but using the slide transcriptions obtained automatically instead of the error-free transcriptions. When comparing automatic versus error-free slides transcriptions, it is observed that roughly 50% of the improvement is lost. However, even a slide-dependent language model obtained from noisy text outperforms the baseline.

The last step in the proposed adapatation model employs not only a slide-dependent model for the whole video, but also a specific language model for the current slide. In the last row in Table 26, we added to the +slide-dependent language model, a language model that depends on the previous, the current and the following slide. Due to the small amount of text in each slide, there is not enough data to compute modified Kneser-Ney discounts for +most models, and instead a constant 0.8 discount was used to smooth these models. This synchronised language model, denoted by SYNC, obtains similar performance to that of the slide-dependent language model for the whole video. Therefore, it does not pays off to use a synchronised language model for each slide.

In conclusion, the text of the slides has been proved to be a valuable resource to improve the system performance. It has also been observed, that even when the text extracted from slides is noisy containing a large amount of mistakes, it is still profitable using it for adapting the language models. As future work, we will analyse the repercussion of using a more sophisticated OCR. We will also investigate another ways of interpolating language models. Finally, we will analyse other ways to use the text extracted from the slides such as corpora selection.

Table 26: WER (%) and PPLs on the poliMedia corpus for several adapted language models

	Development		Test	
	PPL	WER	PPL	WER
Baseline	179.4	22.3	238.6	24.8
(a) Human Slides	108.4	21	125.9	21.4
(b) OCR Slides	128.7	22.4	143.5	23.8
(c) Human Slides+SYNC	95	21.1	113.1	21.9

3.3 Adaptation by dynamic interpolation for English lectures

RWTH started massive adaptation of language models for the English lectures with 4-gram language models (LM) estimated with the unmodified Kneser-Ney method [21]. 16 different sources from Hub4, TDT, Gigaword, Quaero, TED and Translectures were used to train an individual LM each. Table 27 gives an overview of the corpora. Hub4 and TDT contain data from broadcasting news (BN) with up to 75 million words, while the Gigaword corpus consists of data from six different newspapers with up to 1.3 billion words. The Quaero datasets mostly contain data collected from blogs with up to 700 million words. The two training corpora from TED and Translectures have only 2 million and 130 thousand words, respectively. Finally, data from the Translecures-Slides were also used for an additional corpus, only containing data of slides belonging to the development and testing data set with about 30 thousand words.

For evaluation, four different sessions from Translectures were chosen for a development (Dev) and a testing (Test) corpus each. Theses eight recordings show good audio quality, cover different topics with different speakers and have minimal overlap with training data. The number of words vary for each session from 2,500 to 13,000 words with 80 to 600 sentences. The exact numbers for the Dev and Test set can be found in Table 28 and 29.

The used vocabulary is based on the Quaero and Gigaword data, containing the most frequent words of each corpus weighted by the size of the corresponding corpus. The total number of words in the vocabulary is 150,035.

Table 27: English lectures: Training Data

Model	Corpus	Running words	Sentences	Type
1	Hub4	1,839,801	136,051	BN
2	TDT2	22,904,366	1,115,587	BN
3	TDT3	75,031,714	17,266,607	BN
4	TDT4	10,552,411	2,483,927	BN
5	Gigaword-AFP	552,882,685	20,509,548	newspaper
6	Gigaword-APW	973,188,204	46,157,888	newspaper
7	Gigaword-CNA	26,985,917	912,878	newspaper
8	Gigaword-LTW	245,483,068	12,042,621	newspaper
9	Gigaword-NYT	1,256,893,260	62,821,025	newspaper
10	Gigaword-XIN	267,990,718	10,351,283	newspaper
11	Quaero Blog-News 2010	520,296,305	31,176,335	blog-news
12	Quaero 2011	2,962,862	125,245	news
13	Quaero Blog-News 2011	698,199,462	33,835,464	blog-news
14	TED	2,518,335	1,112	talks
15	Translectures-TL	131,366	5,335	talks
16	Translectures-Slides (Dev+Test)	32,434	10,813	talks

As basic approach all 16 training corpora were used to train a single LM using the SRILM

Table 28: English lectures: Dev Data

Recording	Material	Running words	Sentences
1	etvc08_barbaresco_aoigt	5982	258
2	mbc07_abdallah_idt	8972	293
3	sep09_jablonka_eifd	9542	364
4	slonano07_makovec_spm	2442	83

Table 29: English lectures: Test Data

Recording	Material	Running words	Sentences
1	acs07_moulines_mcm	5886	169
2	apr08_culver_aws	12784	610
3	webstart08_leitersdorf_esvc	7555	281
4	wict07_divjak_deome	7330	262

toolkit [54]. The models were then tested on the Dev and Test data. For this purpose all four texts of each set were concatenated to one text and then evaluated for each LM.

The results are shown in Table 30. Mainly, the PPL ranges from 300 to 900 for the Dev set and from 200 to 700 for the Test set. Only the perplexity for the Translectures-Slides is very high with over 1.000, which is not surprising, since it contains only few data.

Table 30: English lectures: Language model perplexities for single models

Model	Dev	Test
1	485.4	345.5
2	468.8	336.1
3	494.4	353.5
4	521.3	374.2
5	666.1	626.7
6	626.7	437.3
7	908.1	739.0
8	413.0	302.9
9	426.0	301.3
10	686.6	544.1
11	327.1	218.1
12	330.7	263.3
13	353.5	264.4
14	358.1	295.1
15	303.5	300.3
16	1140.1	1852.0

Next, the 16 language models were interpolated for the two concatenated evaluation corpora, which is named *fixed* interpolation. The interpolation weights were optimized on the Dev corpus and then also applied to the Test corpus with a resulting PPL of 138.8 and 132.2, respectively. Obviously, weighting down the worse language models leads to far better results than the best single model. Additionally, the combination of multiple models offers better decisions for all n-grams, also lowering the perplexity.

For better adaptation interpolation was also carried out separately for each recording of the

Dev and Test set. The resulting perplexities were then combined to a single PPL, using the number of words per recordings as weighting factor.

This approach is named here as *adapted* interpolation and leads to small improvements of the PPL on both, Dev and Test set, down to 127.3 and 125.7, respectively. For better comparison all interpolation results are shown in Table 31.

In later experiments the impact of the interpolation and adaptation shall also be measured in terms of recognition error rates. For this purpose the language models have to be combined to a single LM in a single file. The problem is that the data of all models are too much for such an approach, since the calculation exceeds any memory capacity. Hence, RWTH chose to use just the important LMs, meaning those with an interpolation weight above one percent. These are here the models 8, 9, 11, 12, 13, 14, 15 and 16, being two of the Gigaword corpora and all Quaero, the TED and the Translectures corpora. As shown in Table 31, a new interpolation with these eight models leads to nearly the same results as an interpolation with all available language models.

As further modification, RWTH chose to use a background model to enhance PPL. The idea of a background model is to concatenate all training corpora to a new additional corpora, which is used to train a new language model. This model is then also included in the interpolation process for further improvement of the overall model. Again, due to memory limits it was not possible to use all data for such a model. Since the calculation of a background model needs even more memory, RWTH was forced to further reduce the number of involved models. Hence, only the best models with an interpolation weight above ten percent were chosen, not considering the model based on Translectures-Slides, since it holds to few data. This choice resulted in model 11, 13, 14 and 15, the two Quaero Blog-News and again the TED and Translectures corpora without slides. For completeness the interpolation results for those four models are also shown in Table 31.

After concatenating the data and calculating the background model it turned out that this model alone comes to a perplexity of 312.3 for the Dev set and 219.1 for the Test set, also depicted in Table 32. This is not surprising since the model contains all data without any weighting and worse corpora with more text just outbalance better but smaller ones like the Translectures corpora in this case. But combining the background model with the already chosen eight LMs slightly improves the overall perplexity nearly fully negating the effect of the missing eight language models.

Detailed results for this last combination are shown in Table 33. It should be noted that for the second and third recording of the Dev and the Test set very low perplexities from 116 to 127 can be achieved.

As further research, RWTH plans the evaluation of cache language models as improvement of the shown adaptation approach. Furthermore, the generated language models will also be tested in a speech recognition framework to verify the optimization of the perplexity in terms of word error rates.

Table 31: English lectures: Adaptation

Language Model	Perplexity			
	Dev		Test	
	Interpolation		Interpolation	
	Fixed	Adapted	Fixed	Adapted
16 Models (all)	138.8	132.2	127.2	125.7
8 Models (Interpolation Weights Above 1%)	138.7	133.1	127.3	126.4
4 Models (Models 11 13 14 15)	173.2	170.2	141.0	139.7
8 Models + Background Model	138.5	132.2	127.0	125.8

Table 32: English lectures: Background model

Language Model	PPL	
	Dev	Test
Background Model (Models 11 13 14 15)	312.3	219.1

4 Massive adaptation of translation models

4.1 Multi-domain translation model adaptation

4.1.1 Introduction

The performance of Statistical Machine Translation (SMT) models highly correlates with the domain congruence between the corpora that were used to train the model on the one hand, and the test data that we are interested to translate on the other. For example, systems that were trained on large collections of parliamentary proceeding translations such as the Europarl corpus [23], while achieving high BLEU scores on unseen test sentences from the same collection, typically underperform when translating text from a different domain, such as news stories or lecture transcriptions. However, while large in-domain bilingual training corpora are a dependable ingredient of strongly performing SMT systems, we can hardly expect to have access to such rare and expensive resources for every domain of interest.

A common solution to this problem explores the middle-ground between relying solely on large out-of-domain resources and only using small in-domain corpora. This involves training large models from arrays of in- and out-of-domain data, producing translation systems which combine the wide coverage of large out-of-domain corpora, with the information on domain-specific translation correspondences contained in in-domain data. Straightforward methods to bring in- and out-of-domain training corpora together in SMT systems mostly perform better than employing exclusively one or the other kind of data. Still, such methods also involve the danger of diluting the domain-specific translation correspondences contained in the in-domain corpus with irrelevant out-of-domain ones. Also, when bringing all training data together, we may end up with an incoherent translation model which while offering wide coverage, does not do particularly well on any kind of data.

Previous work addressed these issues by tracking from which subset of the training data each translation option (e.g. a phrase-pair) was extracted, and using this information to better target the translation of the in-domain data. [29] introduce sentence level features which register for each training sentence-pair the training corpus collection of origin and the language genre it belongs to. Using these features, they train a perceptron to compute a weight for each sentence-pair, which is used to down-weight the impact during training of translation examples

Table 33: English lectures: Adaptation detailed

Recording	Dev			Test		
		PPL Interpolation			PPL Interpolation	
	Words	Fixed	Adapted	Words	Fixed	Adapted
1	5982	121.1	108.5	5886	150.0	152.0
2	8972	122.6	121.7	12784	118.3	116.4
3	9542	132.9	126.9	7555	125.5	118.2
4	2442	354.6	339.8	7330	127.6	132.1
all	26938	138.5	132.2	33555	127.0	125.8

that are not helpful on the test-set domain. [10] use similar collection and genre features to distinguish between training sentence-pairs and compute separate lexical translation smoothing features [25] from the data falling under each collection and genre. Tuning on an in-domain development set allows to learn a preference for the lexical translation options found in the training examples which are similar in style and genre.

In this work, XEROX also focused on employing information on the origin of training points from particular subsets of a larger training set, which is sourced from a multitude of bilingual and monolingual data collections. We are particularly interested when such collections contain data of different linguistic styles or genres. In this multi-domain adaptation task, we aim to learn translation models which bring together the translation correspondences found in each training data subset, in such a way as to increase performance on the in-domain test data. The domain of interest for the *trans* **Lectures** project relates to transcriptions of video lectures from the scientific domain.

The first contribution of XEROX in this direction involves the introduction of additional translation model features in a log-linear model. For every translation hypothesis formulated during decoding, these track the number of source words that got translated using a translation option from each individual bilingual data collection of a large composite training corpus. This follows the basic intuition that information on the origin of each translation option is useful to distinguish those that are relevant to the domain of interest, which is also shared by work such as [29] and [10]. Contrary to previous work however, instead of indirectly employing this information to assign training data weights or using it to train separate lexical translation models, we opt for directly monitoring and tuning the coverage of test sentences on the word level by translation correspondences extracted from each training data collection.

The second contribution of XEROX is to complement a main Language Model (LM) trained on all available composite target language data, with an array of Language Models, each of which is only trained on target language data belonging to a particular training collection. The main LM assigns a score related to the overall conformity of target language output to the lexical and syntactic structure of the target language. In addition to this however, each additional domain-specific LM provides a measure of how close each candidate translation conforms to the characteristics of the data belonging to each constituent training data collection of the composite training corpus. By weighing these scores log-linearly and tuning their weights on an in-domain development corpus, we aim to bias translation output towards effectively combining the domain-specific characteristics of each subset of the training material to match the test data domain. In the empirical evidence we present below, we find that employing both approaches in tandem constitutes a robust framework to increase in-domain performance when training on diverse sets of bilingual corpora, including data from the *trans* **Lectures** domain.

4.1.2 Baseline

While our method is directly applicable to hierarchical [9] and syntactically-driven SMT approaches employing log-linear translation models, in the exposition and the experiments described below we will constrain ourselves to applying our approach on a Phrase-Based SMT (PBSMT) [25] baseline. In this section, we briefly discuss the main points of this family of SMT models.

A PBSMT model translates a source sentence by segmenting it into phrases and translating each phrase independently. To do so, it employs phrase-pairs extracted from word-aligned parallel corpora, which can be reordered during decoding. The conditional probability $p(\mathbf{e}|\mathbf{f})$ of each translation hypothesis $\mathbf{e}$ to be the correct translation of the source sentence $\mathbf{f}$ is assessed by a log-linear combination of a number of Φ weighted features ϕ. Their weights λ are tuned on a small development corpus. During decoding, the translation $\hat{\mathbf{e}}$ with the highest probability $p(\mathbf{e}|\mathbf{f})$ according to the model is output.

$$\hat{\mathbf{e}} = \arg\max_{\mathbf{e}} \; p(\mathbf{e}|\mathbf{f}) = \arg\max_{\mathbf{e}} \; \log p(\mathbf{e}|\mathbf{f}) \tag{4}$$

$$\log p(\mathbf{e}|\mathbf{f}) = \sum_{i=1}^{\Phi} \lambda_i \log \phi_i(\mathbf{e}, \mathbf{f}) \tag{5}$$

Typical features ϕ include conditional phrase and lexical translation probabilities, computed across both translation directions (source to target, target to source), features tracking phrase reordering operations, as well as word and phrase generation penalties. Finally, the Language Model score $\phi_{\mathrm{LM}}(\mathbf{e})$ assigned by an LM trained on target language monolingual data is also included as a crucial feature targeting the production of fluent target language output.

In the context of this part of the report, i.e. SMT systems trained on a composite corpus combining multiple collections of training data, we will consider as our baseline a PBSMT system trained on the concatenation of all available bilingual and monolingual data, without regard to which collection each training example originates from. In this way, we end up with a competitive baseline, which in typical applications combines the wide-coverage of larger out-of-domain corpora, with the domain-specific information contained in smaller corpora that have a smaller domain-divergence from the test data domain.

An often slightly more competitive baseline can be established by assigning a count higher than one to sentence-pairs from in-domain training data collections. This assumes, however, that the target domain is known before training data is word-aligned, and that in-domain training data is available. We make no such assumptions in this work, allowing our method to be applied to any test domain for which a small development corpus exists.

4.1.3 Lexical coverage features

The first extension of XEROX to the baseline involves the employment of additional lexical coverage features. These track during decoding how many source words get translated by using a phrase-pair extracted from each collection of bilingual training data belonging in our overall composite training corpus.

Let us assume that we are training our model using a training corpus $\mathbf{C}$ composed as a set of a number of D constituent collections of training data $\mathbf{C}_d$.

$$\mathbf{C} = \bigcup_{d=1}^{D} \mathbf{C}_d = \{\mathbf{C}_1, \mathbf{C}_2, \ldots, \mathbf{C}_D\} \tag{6}$$

During phrase-pair extraction, we label each extracted pair $\langle \tilde{e}, \tilde{f} \rangle$ of source phrase $\tilde{f}$ and target phrase $\tilde{e}$, with a bit-vector $\mathbf{b}^{\langle \tilde{e}, \tilde{f} \rangle}$ of length D. Each bit $b_d^{\langle \tilde{e}, \tilde{f} \rangle}$ is set to 1 if the related phrase-pair can be extracted from the training data collection $\mathbf{C}_d$, otherwise it is set to 0. Multiple bits can be set to 1 in a single $\mathbf{b}^{\langle \tilde{e}, \tilde{f} \rangle}$, as it is possible that a phrase-pair can be extracted from more than one training data collection $\mathbf{C}_d$.

We consider that when a phrase-pair $\langle \tilde{e}, \tilde{f} \rangle$ with a source phrase $\tilde{e}$ of length $\tilde{l}$ is used in a translation hypothesis, the training collection $\mathbf{C}_d$ that it was extracted from contributes $\tilde{l}$ words in the translation of source sentence $\mathbf{e}$. If $\langle \tilde{e}, \tilde{f} \rangle$ was extracted from a number $n > 1$ of collections $\mathbf{C}_d$, we consider that each such collection contributes $\tilde{l}/n$ words.

Accordingly, our new lexical coverage features ϕ_{LC}^d follow these assumptions to count how many source words were covered using phrase-pairs extracted from each $\mathbf{C}_d$. We add to the model of equation (5), D additional features, one for each training data collection. Their values are computed for a translation $\mathbf{f}$ of source sentence $\mathbf{e}$ constructed using phrase pairs $\langle \tilde{e}, \tilde{f} \rangle$ as follows.

$$\log \phi_{LC}^d(\langle \tilde{e}, \tilde{f} \rangle) = \frac{\tilde{l} * b_d^{\langle \tilde{e}, \tilde{f} \rangle}}{\sum_{i=1}^{D} b_i^{\langle \tilde{e}, \tilde{f} \rangle}} \tag{7}$$

$$\log \phi_{LC}^d(\mathbf{e}, \mathbf{f}) = \sum_{\langle \tilde{e}, \tilde{f} \rangle} \log \phi_{LC}^d(\langle \tilde{e}, \tilde{f} \rangle) \tag{8}$$

4.1.4 Domain-specific language model array

Typical PBSMT systems employ a single Language Modelling feature $\phi_{\text{LM}}(\mathbf{e})$, which allows the decoder to show a preference for output sentences which are fluent in the target language. This LM is commonly trained on all available target language resources, comprising of the target side of the bilingual corpora $\mathbf{C}_d$, together with any other available target monolingual resources.

However, when the monolingual training data are composed of a large array of individual collections, each falling under a different language domain, the LM score provided by such a feature can: (a) be dominated by the style and genre of the larger data collections that provide more training examples, or (b) assign scores preferring a 'lowest common denominator' kind of language use, promoting target output which favours general language use, but which crucially fails to match any style or genre in particular, including that of the test data domain.

In order to address these issues, XEROX trained individual Language Models from each target monolingual data collection, and we used these LMs side-by-side with a main LM trained on all the available target data. Namely, if our target language corpus $\mathbf{M}$ consists of M collections of target sentences $\mathbf{M}_m$, we train from each $\mathbf{M}_m$ a separate LM feature $\phi_{\text{LM}}^m(\mathbf{e})$. We then add these as additional Language Modelling features in the model of equation (5), not replacing but *in addition* to the main LM feature $\phi_{\text{LM}}(\mathbf{e})$.

For the total LM array $\langle \phi_{\text{LM}}, \phi_{\text{LM}}^1, \ldots, \phi_{\text{LM}}^M \rangle$, we tune the feature weights $\langle \lambda_{LM}, \lambda_{LM}^1, \ldots, \lambda_{LM}^M \rangle$ together with the rest of the feature weights on a development set. This allows to set a preference during decoding for the mixture of general language usage (as scored by $\phi_{\text{LM}}(\mathbf{e})$) and particular styles and genres (as scored by each $\phi_{\text{LM}}^m(\mathbf{e})$) that matches the test domain.

Table 34: Bilingual training data collections; size is measured in sentence-pairs

Corpus	Genre	Style	Size
Europarl	Parliament proceedings	Formal speeches	1M
JRC-Acquis	Legal	Formal text	1.2M
WIT-3	Science for laymen	Lectures	145K
WMT-Newstext	News	Articles	137K
COSMAT	Science for specialists	Abstracts & Articles	56K
trans **Lectures**	Science for specialists	Lectures	3.2K

Table 35: Single-domain and concatenated training data baselines, for the two test data domains

Test Domain	Europarl	JRC-Acq.	WIT-3	WMT-News	COSMAT	*trans* **Lectures**	Concat.
trans **Lectures**	23.93	18.14	28.36	22.27	19.08	24.04	**30.93**
WMT-News	24.47	20.50	22.38	24.05	15.23	11.02	**28.20**

4.1.5 Experiments

In order to empirically validate our method, we compiled a combined training corpus which consists of six distinct collections of bilingual training data between English and French, differing substantially in style and genre. Material includes content from the Europarl [23], JRC-Acquis [48], WIT-3 [7], 2011 WMT-Newstext [6], COSMAT [27] corpora, as well as the *trans* **Lectures** training corpus [58]. Table 34 lists the corpora and their key properties.

We experiment with two different domain adaptation scenarios, translating from English to French. Firstly, we explore translating transcriptions from the *trans* **Lectures** domain, where a very limited in-domain corpus is available (3.2K sentence-pairs), in addition to 1K/1.3K development and test data respectively. Secondly, we also opt for the task of translating newstext from the WMT-Newstext domain, where a sizeable in-domain corpus exists (137K sentence-pairs), alongside a 2K development set and a 3K test set.

For our experiments, we choose for the Moses implementation of a PBSMT system [17] and keep the default training and tuning parameters. We begin by building a series of single-domain baselines, each trained solely on each individual training data collection. Language models for each system are trained on the target side of each corpus and tuning of the feature weights is performed with the MERT algorithm [32]. We also train a further baseline for each experimental setting, using the concatenation of all bilingual and monolingual data collections to train the translation and language model respectively.

Table 35 lists the BLEU scores achieved from each such system for the two translation tasks. The importance of in-domain training data is highlighted by the strong performance of models trained on the in-domain corpora. Surprisingly, this applies even when training both translation and language model only on the tiny *trans* **Lectures** training corpus of 3.2K sentence-pairs. Still, larger corpora from more or less closely related domains (WIT-3 for the *trans* **Lectures** task, Europarl for the 'WMT-Newstext' task) are able to score even higher than the more limited in-domain data, while the rest of the scores correlate with domain distance and corpus size. Interestingly, the baseline training on the concatenation of all available resources offers the most competitive performance, significantly outperforming all other single-domain systems in both experimental settings.

We then experiment with extending the concatenated training data baseline with our lexical coverage (`lexcov`) and domain-specific LM (`domain-lm`) features. The six lexical coverage fea-

Table 36: Adapted models trained on multi-domain data collections. The 'concatenated' training data baseline is compared to systems extending it with the additional lexical coverage (`lexcov`) and domain-specific LM (`domain-lm`) features. Scores with a (*) are significantly better than the best-scoring baseline at the 1% level. The best system(s) (if more than one, indistinguishable at the 1% significance level) are marked in bold.

Test Domain	baseline	lexcov	domain-lm	Tuning	BLEU
trans **Lectures**	■			MERT	30.93
	■			MIRA	31.52
	■	■		MIRA	**32.36***
	■		■	MIRA	**32.48***
	■	■	■	MIRA	**32.44***
WMT-Newstext	■			MERT	28.20
	■			MIRA	28.04
	■	■		MIRA	28.41*
	■		■	MIRA	28.82*
	■	■	■	MIRA	**29.01***

tures (one for each training data collection) always complement the existing features of the baseline (conditional phrase and lexical translation probabilities, phrase penalty) in a phrase-table which includes the same phrase-pairs as the baseline. Similarly, the array of six domain-specific 5-gram LMs, smoothed using modified Kneser-Ney [8], complement as additional features the LM trained on the combination of all available target language data.

Table 36 lists the performance of our adapted models in comparison to the baseline system, where the latter does not include the features we introduced in Sections 4.1.3 and 4.1.4. For each experimental setting, we test one system which only extends the baseline phrase-table with the `lexcov` features, one which only introduces the `domain-lm` features and a third system which combines both kinds of features. We note that, although the baseline is already competitive in relation to the single-domain systems, our systems which extend it with each of the two sets of contributed features further raise performance as measured by BLEU, with the improvement being statistically significant at the 1% level using the test of [22]. While both kinds of features contribute towards an in-domain performance improvement, combining them together manages to consistently deliver strong improvements across both test domains, with significantly higher BLEU than using each feature set on its own registered for the 'WMT-Newstext' task.

We must note that for our adapted systems, due to the large additional number of features they employ, we opt for the Moses implementation of the MIRA algorithm [11] instead of MERT to tune the log-linear model weights. In order to make sure that the performance improvement does not follow solely from using a different tuning algorithm, we also tune a baseline system using the MIRA algorithm. Such a baseline actually registers a drop in performance for the 'WMT-Newstext' task (28.04 BLEU) in comparison to using MERT for tuning. For the *trans* **Lectures** task, the BLEU score of the MIRA-tuned baseline increases to 31.52 BLEU, still significantly surpassed by our best performing system by close to 1 BLEU.

4.1.6 Related work

Domain adaptation work for SMT models can be grouped in two categories. On the one hand, work such as [57, 4] aim to raise translation performance on in-domain test data by generating additional synthetic training data, using for example bootstrapping. On the contrary, our approach falls into a second category of SMT adaptation methods, which focus on employing more efficiently the existing monolingual and bilingual training resources.

[29] exploit the information on the collections from which individual training examples originate, as well as indications of genre that is sometimes present in training set metadata, to compute a weight for each training sentence according to how helpful it is to translate data from the test domain. Our work is motivated by the same intuition, namely that we can better target the test domain by exploiting the information on the grouping of composite, inhomogeneous training data in more homogeneous data collections. However, previous methods try to tune the desired contribution of each training collection in formulating translations hypotheses for the test domain *indirectly*, through computing importance weights for each training point [29], or training separate lexical translation models [10]. In contrast, we pursue the same objective with the much more direct lexical coverage features, achieving substantial empirical improvements over similar baselines as those considered by previous authors with a simpler to implement method.

The concept of training separate language models from each training data collection instead of a single model trained on all available data and using them as separate features during decoding, is not new. Still, previous work such as [44] explores the log-linear combination of a very small number (usually 2) of domain-specific LMs. In contrast, in this part of the report we examine empirically the implications of employing large arrays of such domain-specific LMs in tandem with an LM trained on all available data, and evaluate how these LM arrays complement the simultaneous adaptation of the translation model.

Finally, recent work [12, 55] employs topic models to softly categorise composite training material in more homogeneous collections, instead of using the human classification found in corpora metadata. We consider as an interesting direction for future work, to evaluate the usefulness of such automatic clustering of training data as a replacement for the training collection membership information that we used in this work.

4.2 Adaptation using in-domain data

The Slovenian→English, English→Slovenian and English→German SMT systems are based on the open-source toolkit Jane [59, 62]. For Slovenian→English and English→Slovenian, RWTH uses the phrase-based decoder, while for English→German the hierarchical phrase-based decoder is utilized.

RWTH used GIZA++ [36] to word-align the bilingual data, from which the phrase table is extracted. The language models are 4-gram LMs trained with the SRILM toolkit [53]. We use the standard set of models with phrase translation probabilities and lexical smoothing in both directions, word and phrase penalty and an n-gram target language model. The phrase-based system also includes a distance-based reordering model and three binary count features. The binary count features can have the values 1 or 0 and they indicate, whether the phrase pair appeared in the training data at least a certain number of times. Here, the features are associated with the count values 1, 2 and 3. The log-linear parameter weights are optimized with MERT [35].

4.2.1 Slovenian↔English

RWTH used three different sources as training data, one large out-of-domain bilingual corpus and two small corpora containing translated video lectures. The baseline system is trained and optimized on the JRC-Acquis corpus [49]. This corpus is composed of law texts and therefore not related to the lecture domain. By adding training data collected from `videoLectures.net` within *trans* **Lectures** and TED talks available for the IWSLT 2012 shared evaluation task (`iwslt2012.org`), results are substantially improved. Their strong impact can be explained by their relevance to the domain. The Slovenian language model is trained on the bilingual data

Table 37: Corpus statistics: Slovenian↔English.

		Slovenian	English
JRC	Sentences	1.09M	
	Run. Words	24.7M	27.5M
	Vocabulary	199699	129335
JRC+TED+videoLectures	Sentences	1.11M	
	Run. Words	25.2M	28.0M
	Vocabulary	217731	136056
Dev	Sentences	715	
	Running Words	15K	19K
	Vocabulary	3428	2026
	OOVs (run. words)	1418	772
	OOVs (run. words)+TED,videoLectures	1024	602
Test	Sentences	1360	
	Running Words	27K	37K
	Vocabulary	4953	2832
	OOVs (run. words)	1562	891
	OOVs (run. words)+TED,videoLectures	730	473

Table 38: Language model perplexities: Slovenian↔English.

	Data	ppl
Slovenian	JRC	1397
	+ TED + videoLectures	825
English	JRC + Europarl + News Commentary	657
	+ TED + videoLectures bilingual	288
	+ TED + videoLectures monolingual	170

only. The English language model utilized additional sources, namely the Europarl and News-Commentary corpora provided for the WMT 2012 evaluation task (`http://www.statmt.org/wmt12`). For the Slovenian→English task, we finally added monolingual TED talk and textttvideoLectures.net data to train the language model, which again slightly improved the results. Bilingual data statistics are given in Table 37, language model perplexities in Table 38. The training data statistics for the final English language model can be found in Table 39. The results of the baseline translation systems are shown in Tables 40 and 41, and are measured in BLEU[%] and TER[%] on the development and test sets extracted from `videoLectures.net`.

Table 39: English language model training data, containing the JRC-Acquis, Europarl, News-Commentary, `videolectures.net` and TED corpora.

Sentences	3525055
Running Words	120389525
Vocabulary	427126

Table 40: Slovenian→English translation results.

System	Development		Test	
	BLEU	TER	BLEU	TER
JRC	12.5	67.0	9.1	73.1
+videoLectures & TED	20.5	61.8	15.0	67.8
+TED monolingual	21.6	59.5	15.7	65.2

Table 41: English→Slovenian translation results.

System	Development		Test	
	BLEU	TER	BLEU	TER
JRC	7.7	79.9	6.2	95.0
+ videoLectures & TED	13.4	68.5	12.0	77.4

4.2.2 English→German

For the English→German translation system, in addition to the 4,019 bilingual training sentences collected from `videoLectures.net`, we used the bilingual TED talk data, which added an additional 129 911 bilingual training sentences. The corpus statistics can be found in Table 43. The German language model is trained on the target side of this data and has a perplexity of 191. It is further improved by utilizing the slide content which is available for the lectures. This is described in the following Section.

Unfortunatelly, due to the tight schedule of the scientific evaluation procedure, both for English→Slovenian and English→German the systems used for evaluation are inferior to the ones described here. Therefore, we expect strong improvements in the next quality control.

4.3 Adaptation using slides

We focused on using the slide text as an additional source of information for massive adaptation.

4.3.1 Extracting the slide text

In addition to the transcribed and translated sentences, the slides shown during the lectures are also given. All slides are available in the jpeg format, and most of them also as pdf and/or powerpoint presentation. We extracted the textual information from the pdfs using the open source pdftotext tool. The powerpoint presentations could be processed as well by converting them into pdf files first. One lecture provided only the jpeg images of the slides. As the resolution of the jpegs was rather small, we manually transcribed these few slides instead of using an OCR software to ensure a good data quality.

We aligned the spoken sentences with the slides using the time stamps given in the xml data. Since the alignment information between the images and the extracted text was not provided, we relied on the assumption that each new shown slide was in fact the next slide given in the pdf. This procedure led to a parallel corpus consisting of three parts, the English sentence, its German translation and the text of the slide shown while the sentence was spoken.

4.3.2 Discriminative word lexica

We apply a discriminative word lexicon to incorporate the slide information into a statistical machine translation system. Discriminative word lexica (DWL) as proposed in [30] model the probability of the presence or absence of words in the target sentence based on the presence of words in the source sentence. In our scenario, we use the content of the slide as the source information and want to predict the presence or absence of words in the target sentence. As the spoken sentences are usually longer than the brief information presentented on the slides, regular translation models such as phrase models are not suitable, because many spoken words are not contained on the slides, and the words on the slides might be spread throughout the spoken text. The discriminative word lexicon on the other hand does not take into account the order in which the words appear and thus can be applied to this scenario. Instead of modeling phrases or sequences, the DWL only models the bag of words of the target sentence given the bag of words of the slide. Formally speaking, given the words on the slides, $g_1, g_2, ..., g_J = g_1^J$ and the target sentence $e_1, e_2, ..., e_I = e_1^I$, the discriminative word lexicon models the probability of the bag of words $\{e_1^I\}$ made up of words from the vocabulary V_e:

$$P(\{e_1^I\}|\{g_1^J\}) = \prod_{e \in e_1^I} P_+(e|\{g_1^J\}) \times \prod_{e \notin e_1^I} P_-(e|\{g_1^J\}) \tag{9}$$

Here, the two probabilities P_+ and P_- are modeled as log-linear models:

$$P_{+/-}(e|\{g_1^J\}) = \frac{exp(\sum_{g \in g_1^J} \lambda_{g,e}^{+/-})}{exp(\sum_{g \in g_1^J} \lambda_{g,e}^{+}) + exp(\sum_{g \in g_1^J} \lambda_{g,e}^{-})} \tag{10}$$

The DWL is trained on the slide information and the target sentences of the training data. In decoding, we combine the DWL model with other regular translation models in a log-linear framework and extend our decoder to use both the source sentence and the slide information as input information.

As the slides of the lectures contain a lot of mathematical symbols and formulas as well as numbers, we tested two scenarios, one using all possible words, and one using only words containing letters from the alphabet, thus excluding numbers and formulas. It turned out that the latter scenario led to a better translation performance. It seems that the key words on the slides triggered the translation of certain target phrases and formulations.

4.3.3 Adaptation using a recurrent neural network

As an alternative approach for utilizing the additional information given in the slides, RWTH trained recurrent neural network (RNN). The RNN structure is closely related with Multilayer Perceptrons (MLP) [41, 5], but the output of one or more hidden layers is reused as additional inputs for the network in the next time step. This structure allows the RNN to learn whole sequences without restricting itself to a fixed input window. The long short-term memory (LSTM) [18] is applied to counter the effect that long distance dependencies are hard to learn with gradient descent. This is often referred to as vanishing gradient problem [3].

The input layer of the recurrent neural represents the previous word in the target language, the aligned words in the source language, and all words that are shown on the current slide. The size of the input layer is therefore the input vocabulary + the output vocabulary + the slide vocabulary. In each step, all input nodes are inactive, except for the node representing the previous word, all aligned words and all words on the current slide. The RNN uses these inputs to predict the next word and calculate a score for the whole sentence using a softmax layer similar to the application of neural networks to language modeling [2]. This additional

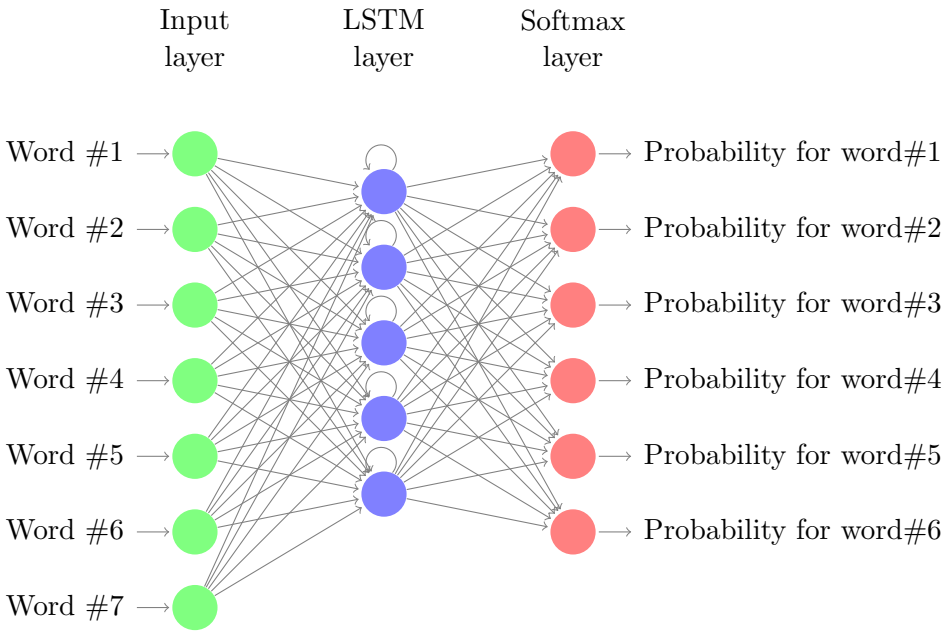

Figure 2: Simplified structure of the recurrent neural network

Table 42: English→German translation results. Improvements marked with ** are statistically significant with $p < 0.01$.

System	Development		Test	
	BLEU	TER	BLEU	TER
Baseline	20.8	60.6	18.9	65.4
+ DWL on slides	21.0	60.1	19.3**	65.0**
+ DWL on alphabet	20.4	59.7	19.5**	64.0**
+ RNN on slides	20.7	60.4	19.0	65.2

score is calculated for the 1000 best translations. A new best sentence is then found using the new score in combination with the scores previously calculated during the regular translation process. We tested different network structures which were trained on the training data until no more improvement on the development data was observed. The RNN that performed best on the development data consists of one input layer, one hidden layer with 100 LSTM nodes, and one output layer.

4.3.4 Results

The results of the different adaptation techniques can be found in Table 42. While the DWL method using all words including numbers and mathematical symbols led to a slight improvement over the baseline, restricting the system to only regular words led to significant improvements of 0.6 BLEU and 1.5 TER. Both DWL results were significantly better than the baseline with $p < 0.01$ (marked with ** in the table). The RNN approach only led to slight improvements over the baseline. Probably the amount of training data was not sufficient enough to train such a model. During the next period, we want to explore both methods more deeply and also try other setups to adapt the translation models.

Table 43: Corpus Statistics English → German

		English	German
Train	Sentences	4019	
	Running Words	103306	96627
	Running Words without Punct. Marks	92709	83457
	Vocabulary	5930	9191
	Singletons	2491	4906
Train + TED	Sentences	133840	
	Running Words	2632805	2549528
	Running Words without Punct. Marks	2288376	2162976
	Vocabulary	45479	69522
	Singletons	18043	31877
Dev	Sentences	1013	
	Running Words	28885	26735
	Running Words without Punct. Marks	26151	23254
	Vocabulary	2701	3762
	OOVs (running words)	2210	2930
	+TED	711	1533
	OOVs (in voc.)	1027	1761
	+TED	350	886
Test	Sentences	1360	
	Running Words	36143	32015
	Running Words without Punct. Marks	32590	27649
	Vocabulary	2836	4140
	OOVs (running words)	2309	3231
	+TED	483	1483
	OOVs (in voc.)	985	1973
	+TED	242	937

4.4 Sentence selection

Among the *trans* Lectures tasks, obtaining automatic translations of educational videos is one of the most challenging tasks, since for many of the language pairs used in the project there are large amounts of external data available. For instance, there are several parallel corpora for Spanish and English such as the Europarl-v7. However, the domain of these external corpora significantly differs from that of the the educational lecture domain. The objective of this task is to use lecture-specific knowledge in order to adapt the translation models to the lecture domain.

Most recent literature defines the Statistical Machine Translation (SMT) problem [37, 33], as follows: given an input sentence f from a source language, the purpose of SMT is to find its translation $\hat{e}$ in a target language such that

$$\hat{e} = \arg\max_{e} \sum_{k=1}^{K} \lambda_k h_k(f, e) \tag{11}$$

where $h_k(f, e)$ denotes the so-called feature functions, and λ_k are their corresponding weights in the log-linear mixture. The feature functions model important factors for the translation of f into e, as for instance the target language model, the reordering models or several translation models.

Consequently, the problem of adapting an SMT model can be reformulated as adapting each of the individual features that compose the model. In this task, UPVLC has focused on the

adaptation of the language models, and the translation models in the context of SMT. In order to adapt the models to the domain of transcribed video lectures, the following techniques were applied:

1. A big language model was trained and adapted from several monolingual corpora similarly to Task 3.2 (see section 3.2.2).
2. A successful sentence selection technique was applied to obtain a representative corpus for the transcriptions to be translated.
3. The language model trained at step 1 is introduced into the model computed at step 2 as a feature function.

Similarly to Section 3.2.2, the big language models were obtained by linearly mixing several n-gram language models trained in different corpora. Let w be the current word within a transcription, and let h be the $n-1$ previous words, then the mixture is made by linear interpolating the language models as follows

$$p(w|h) = \sum_{i=1}^{I} \lambda_i p_i(w|h) \tag{12}$$

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 optimizing them with the EM algorithm to maximize the log-likelihood or equivalently to minimize the perplexity of a given development set [19].

As discussed above, there are many external bilingual corpora for training SMT systems. The main adversity when attempting to exploit such corpora is that there are considerable differences between their domains and the project domain. For instance, while in the *trans* Lectures case, the corpus to be translated are transcriptions of video lectures, one of the external corpora domain is parliamentary proceedings of the United Nations. Bilingual sentence selection techniques aim at selecting a subset of the available bilingual data with which to train a SMT system. This selection is performed to maximize performance, and then these techniques can play the role of domain adaptation techniques if used properly. In addition, as a side-effect, these techniques dramatically amend another problem related to huge amounts of external corpora, i.e. the high computational cost required to work with large amounts of data.

UPVLC has worked with a successful sentence selection technique called *infrequent n-gram recovery* [14]. This technique selects the minimal set of sentences that yield reliable estimations for all the n-grams that occur in the text to be translated. An n-gram is considered frequent, and consequently reliably estimated, if it occurs more than t times in the selected training data. In the case of adaptation, in which typically there is a small in-domain training corpus, the occurrences into the in-domain corpus are considered when deciding whether a n-gram is frequent or infrequent. For selecting, all the sentences in the external corpora are scored as follows

$$i(f) = \sum_{\mathbf{w} \in \mathcal{X}} \min(1, N(\mathbf{w})) \max(0, t - C(\mathbf{w})) \tag{13}$$

where $\mathcal{X}$ is the set of n-grams that appear in the sentences to be translated and $\mathbf{w}$ one of them; $C(\mathbf{w})$ the counts of $\mathbf{w}$ in the source language training set; and $N(\mathbf{w})$ the counts of $\mathbf{w}$ in the source sentence f to be scored. Note that the score $i(f)$ gives higher weight to the n-grams most infrequent n-grams in the training set. The scored sentences are then selected one by one updating the counts of the n-grams in the training set together with the scores of the external corpora sentences after each selection. The process is stopped whenever a maximum number of sentences is obtained, or no infrequent n-ngram is left in the training corpora [14].

In order to combine the large adapted language models from step 1 together with the model trained in a corpora selected with bilingual sentence selection techniques, a log-linear interpolation is performed [26]. This log-linear interpolation is implemented in many statistical machine translation toolkits as for example Moses [24].

Table 44: Basic statistics of the external corpora involved in the generation of the Spanish LM

Corpus	Number sentences	Number of words	Vocabulary size
EPPS	132K	0.9M	27K
News Commentary (Es)	183K	4.6M	174K
TED (Es)	316K	2.3M	133K
Europarl v7 (Es)	2 123K	54.9M	439K
El Periódico	2 695K	45.4M	916K
News (07-11) (Es)	8 627K	217.2M	2 852K
United Nations (Es)	9 493K	253M	1 643K
UnDoc (Es)	9 968K	318.0M	1 854K

Table 45: Basic statistics of the external corpora involved in the generation of the english LM

Corpus	Number of sentences	Number of words	Vocabulary size
TED (En)	142K	2.3M	100K
New Commentary (En)	208K	4.5M	150K
Europarl v7 (En)	2 218K	54.1M	326K
United Nations (En)	10 593K	286M	1 835K
10^9 (En)	19 842K	504.8M	5 834K
News (07-11) (En)	48 872K	986M	6 169K

4.4.1 Experiments

In this section, we describe the experiments carried out by the UPVLC in order to assess the performance of the proposed massive adaptation techniques. For reporting performance, we used BLEU [38] and TER [46]. It is worth highlighting that while BLEU is a precision metric (the higher, the better), TER is an error metric (the lower, the better). All the experiments use Moses [24] as the translation engine. The word alignments needed by Moses are computed using GIZA++ [34]. Finally, the n-gram language models were computed using the SRILM toolkit [52].

The experiments were carried out in two *trans* Lectures corpora: poliMedia from Spanish to English (Es→En) and VideoLectures.NET from English to Spanish (En→Es). The TED corpus, which is composed of educational talks, was used as additional in domain corpus. The main statistics of training, development and test sets of the in-domain corpora are shown in Table 46. As external out of domain corpora for the bilingual selection, we used Europarl and United Nations corpora. Table 47 summarizes the most important statistics of these corpora. It must be noted that, after selecting the sentence from the external corpora, the in-domain training corpora is added to them in order to create the final training set.

Regarding the large language models, since two directions were considered, we computed two external language models: one for Spanish, and another for English. In both cases and in order to compute the individual language models, we used several bilingual and monolingual external corpora in addition to Google counts [31] and the in-domain corpora. Prior to the computation of individual language models, all the corpora were cleaned in order to avoid having noise or very short sentences. Tables 44 and 45 summarize the most important statistics of the external corpora once they have been cleaned. Each individual language model was trained using the SRILM [52] toolkit in a different corpora. All n-gram models were smoothed with modified Kneser-Ney absolute interpolation method [20]. The models used order 4 and were interpolated to minimize the perplexity of the in-domain development set [19].

Table 46: Main figures of the in domain corpora.

Corpus	Sentences	Words		Vocabulary	
		en	es	en	es
TED	144K	2.6M	2.45M	46.4K	67.9K
VideoLectures.NET training	2142	58K	54.2K	3.74K	5.04K
VideoLectures.NET development	1013	28.6K	27.4K	6.23K	2.7K
VideoLectures.NET test	1360	36.1K	33.5K	6.23K	3.3K
poliMedia training	1529	40.2K	40.3K	3.86K	4.71K
poliMedia development	1401	38.7K	37.8K	3.66K	3.51K
poliMedia test	1139	32.1K	32.1K	3.26K	4.08K

Table 47: Main figures of the corpora from which the sentences are selected.

Corpus	Sentences	Words		Vocabulary	
		en	es	en	es
Europarl	1.73M	40.6M	42.3M	98.4K	149K
UN	11M	298M	340M	55.6K	57.1K

Table 48 summarizes the adaptation results obtained for the spanish to/from english translation. For computing the baseline, we used the in-domain training set and the Europarl corpus in order to train the translation and language models. The baseline language model is 5-gram model smoothed with modified Kneser-Ney absolute interpolation method [20]. For the baseline a result of 30.9 and 23.4 BLEU points were obtained for Es→En and En→Es, respectively. This figures were slightly improved by using the linearly interpolated language model instead of the baseline language model. A significant improvement is obtained by bilingual sentences selection techniques. This improvement is even larger with the linearly interpolated language model. However, the best results were obtained by a log-linear combination of both language models: the large linear mixture and the language model resulting from sentence selection. The final improvement after adaptation score a total of 33.5 and 26.0 BLEU points for Es→En and En→Es, respectively.

In summary, the three proposed techniques have successfully adapted the system. Actually, the best result was obtained by means of the combination of all three of them. In addition, as a byproduct of the bilingual sentence selection, the selected corpus was reduced to just 1.5% of the total available data, which significantly reduces the computation requirements for training the statistical translation system.

Table 48: Results in BLEU and TER in the test set of poliMedia and VideoLectures.NET

	VideoLectures.NET		poliMedia	
	BLEU	TER	BLEU	TER
Baseline	30.9	48.7	23.4	56.5
+linear interpolation	31.2	48.7	24.0	56.4
Selection	32.4	47.2	24.9	55.6
+linear interpolation	33.3	46.4	25.3	55.1
+log-linear interpolation	**33.5**	**46.1**	**26.0**	**54.6**

5 Conclusion

In this report, for each of the three languages and six language pairs, we documented the training of our baseline systems. These systems will form the basis for future adaptation experiments, and their continuous improvement will depict the progress of adaptation methods when evaluated on the test data as well as on the full videoLectures.net and poliMedia repositories.

Regarding acoustic adaption, we observed that large improvements in word error rate can be obtained when using CMLLR adaptation. For some of the languages, especially Slovene, we found that the improvements incurred by CMLLR are much larger than what can usually be expected from this technique. In addition, the number of different speakers in lecture recordings usually is very small. This may provide experimental evidence that not only the speaker variation is normalized by this transformation but also the acoustic conditions.

In a second set of experiments, we analyzed the effects of MAP adaptation. We distinguished a supervised versus an unsupervised setting. For the supervised case, a given acoustic model was adapted towards the lecture domain using all available acoustic in-domain data and its manual transcriptions. For the unsupervised case, the baseline acoustic model was adapted with varying amounts of audio data of a specific lecture, and only the recognized word sequence for this lecture was used. Considerable improvements were achieved with supervised MAP adaptation, even on top of a CMLLR adaptation, and even larger gains were obtained for unsupervised adaptation.

For language model adaptation, we investigated several variants of interpolation. It was found that the use of the text data extracted from the slides accompanying a specific lecture was of great benefit for the recognition quality. This was verified by all partners participating in task 3.2.

The slides were not available for each lecture in a format where the text could be directly extracted. As an alternative, by using OCR technology the text data were obtained from image files depicting the slides. Even though the error rates of the OCR system were comparatively high for this task, the language model adaptation based on the OCR texts still gave signification improvements, although these are still much lower than those for the correct slide texts.

For the adaptation of translation models, discriminative word lexica provided a significant improvement in translation quality. Recurrent neural networks so far only gave minor benefits, and we plan to extend our work along this direction in the future. Sentence selection helped to improve translation systems considerably. We observed the largest gains in BLEU when this technique was not only limited to the translation model but was extended to selecting training data for the language model, too.

We also examined how we can increase in-domain translation performance when training on composite SMT training data. We proposed two adaptation methods, one targeting the translation model and one focusing on the language modeling component of phrase-based systems. We used information on the membership of training examples in training data collections to better combine together translation options extracted from each collection during decoding. We also examined the use of an array of domain-specific language models to learn a blend of the individual genre and style of each collection, which matches more closely the test data domain. Our empirical evaluation promotes the combination of the two methods as an effective and easy to implement method to raise in-domain performance when training on multi-domain corpora, and demonstrated that our methods deliver a significant performance improvement when translating scientific video lecture transcriptions for the *trans* **Lectures** project.

References

[1] AK toolkit. `http://aktoolkit.sourceforge.net/`.

[2] Y. Bengio, H. Schwenk, J.S. Senécal, F. Morin, and J.L. Gauvain. Neural probabilistic language models. *Innovations in Machine Learning*, pages 137–186, 2006.

[3] Y. Bengio, P. Simard, and P. Frasconi. Learning long-term dependencies with gradient descent is difficult. *IEEE transactions on neural networks / a publication of the IEEE Neural Networks Council*, 5(2):157–166, March 1994.

[4] Nicola Bertoldi and Marcello Federico. Domain Adaptation for Statistical Machine Translation with Monolingual Resources. In *Proceedings of the Fourth Workshop on Statistical Machine Translation*, pages 182–189, Athens, Greece, 2009. Association for Computational Linguistics.

[5] Christopher M. Bishop. *Neural Networks for Pattern Recognition*. Oxford University Press, USA, 1 edition, January 1996.

[6] Chris Callison-Burch, Philipp Koehn, Christof Monz, and Omar Zaidan. Findings of the 2011 Workshop on Statistical Machine Translation. In *Proceedings of the Sixth Workshop on Statistical Machine Translation*, pages 22–64, Edinburgh, Scotland, July 2011. Association for Computational Linguistics.

[7] Mauro Cettolo, Christian Girardi, and Marcello Federico. WIT^3: Web Inventory of Transcribed and Translated Talks. In *Proceedings of the 16th Conference of the European Association for Machine Translation (EAMT 2012)*, pages 261–268, Trento, Italy, May 2012.

[8] Stanley F. Chen and Joshua Goodman. An Empirical Study of Smoothing Techniques for Language Modeling. *Computer Speech & Language*, 13(4):359–393, 1999.

[9] David Chiang. A Hierarchical Phrase-Based Model for Statistical Machine Translation. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL' 05)*, pages 263–270, Ann Arbor, Michigan, June 2005.

[10] David Chiang, Steve DeNeefe, and Michael Pust. Two Easy Improvements to Lexical Weighting. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 455–460, Portland, Oregon, USA, June 2011. Association for Computational Linguistics.

[11] Koby Crammer, Ofer Dekel, Joseph Keshet, Shai Shalev-Shwartz, and Yoram Singer. Online Passive-Aggressive Algorithms. *Journal of Machine Learning Research*, 7:551–585, December 2006.

[12] Vladimir Eidelman, Jordan Boyd-Graber, and Philip Resnik. Topic Models for Dynamic Translation Model Adaptation. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 115–119, Jeju Island, Korea, July 2012. Association for Computational Linguistics.

[13] M.J.F. Gales. Maximum likelihood linear transformations for hmm-based speech recognition. *Computer Speech and Language*, 12:75–98, 1998.

[14] Guillem Gascó, Martha-Alicia Rocha, Germán Sanchis-Trilles, Jesús Andrés-Ferrer, and Francisco Casacuberta. Does more data always yield better translations? In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 152–161, Avignon, France, April 2012. Association for Computational Linguistics.

[15] J. Gauvain and C. Lee. Maximum a posteriori estimation of multivariate gaussian mixture observations of markov chains. *IEEE Trans. on Speech and Audio Processing*, 2(2):291–298, 1994.

[16] Diego Giuliani, Matteo Gerosa, and Fabio Brugnara. Speaker normalization through constrained mllr based transforms. In *INTERSPEECH*. ISCA, 2004.

[17] Hieu Hoang and Philipp Koehn. Design of the Moses Decoder for Statistical Machine Translation. In *ACL Workshop on Software Engineering, Testing, and Quality Assurance for NLP 2008*, 2008.

[18] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Comput.*, 9(8):1735–1780, November 1997.

[19] 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*, pages 381–397, Amsterdam, The Netherlands: North-Holland, May 1980.

[20] R Kneser and Hermann Ney. Improved backing-off for M-gram language modeling. In *Acoustics, Speech, and Signal Processing, 1995. ICASSP-95., 1995 International Conference on*, pages 181–184, 1995.

[21] Reinhard Kneser and Hermann Ney. Improved backing-off for M-gram language modeling. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 1, pages 181–184, May 1995.

[22] Philipp Koehn. Statistical Significance Tests for Machine Translation Evaluation . In Dekang Lin and Dekai Wu, editors, *Proceedings of EMNLP 2004*, pages 388–395, Barcelona, Spain, July 2004. Association for Computational Linguistics.

[23] Philipp Koehn. Europarl: A Parallel Corpus for Statistical Machine Translation. In *MT Summit 2005*, 2005.

[24] Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christie Moran, Richard Zens, Chris Dyer, Ontraj Bojar, Alexandra Constantin, and Evan Herbst. Moses: Open source toolkit for statistical machine translation. In *Proc. of ACL*, pages 177–180, 2007.

[25] Philipp Koehn, Franz J. Och, and Daniel Marcu. Statistical Phrase-Based Translation. In *Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, Edmonton, Canada, May 2003.

[26] Philipp Koehn and Josh Schroeder. Experiments in domain adaptation for statistical machine translation. In *Proceedings of the Second Workshop on Statistical Machine Translation*, StatMT '07, pages 224–227, Stroudsburg, PA, USA, 2007. Association for Computational Linguistics.

[27] Patrik Lambert, Jean Senellart, Laurent Romary, Holger Schwenk, Florian Zipser, Patrice Lopez, and Frédéric Blain. Collaborative Machine Translation Service for Scientific texts. In *Proceedings of the Demonstrations at the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 11–15, Avignon, France, April 2012. Association for Computational Linguistics.

[28] L. Mangu, E. Brill, and A. Stolcke. Finding consensus in speech recognition: word error minimization and other applications of confusion networks. *Computer, Speech & Language*, pages 373–400, 2000.

[29] Spyros Matsoukas, Antti-Veikko I. Rosti, and Bing Zhang. Discriminative Corpus Weight Estimation for Machine Translation. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 708–717, Singapore, August 2009. Association for Computational Linguistics.

[30] Arne Mauser, Saša Hasan, and Hermann Ney. Extending Statistical Machine Translation with Discriminative and Trigger-Based Lexicon Models. In *Conference on Empirical Methods in Natural Language Processing*, pages 210–218, Singapore, August 2009.

[31] 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.

[32] Franz J. Och. Minimum Error Rate Training in Statistical Machine Translation. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, pages 160–167, Sapporo, Japan, July 2003. Association for Computational Linguistics.

[33] Franz J. Och and Hermann Ney. Discriminative training and maximum entropy models for statistical machine translation. In *Proc. of ACL*, pages 295–302, 2002.

[34] Franz J. Och and Hermann Ney. A systematic comparison of various statistical alignment models. In *Computational Linguistics*, volume 29, pages 19–51, 2003.

[35] Franz Josef Och. Minimum Error Rate Training in Statistical Machine Translation. In *acl03*, pages 160–167, Sapporo, Japan, July 2003.

[36] Franz Josef Och and Hermann Ney. A Systematic Comparison of Various Statistical Alignment Models. *Computational Linguistics*, 29(1):19–51, March 2003.

[37] Kishore Papineni, Salim Roukos, and Todd Ward. Maximum likelihood and discriminative training of direct translation models. In *Proc. of ICASSP'98*, pages 189–192, 1998.

[38] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: A method for automatic evaluation of machine translation. In *Technical Report RC22176 (W0109-022)*, 2001.

[39] Christian Plahl, Ralf Schlüter, and Hermann Ney. Hierarchical bottle neck features for lvcsr. In *Interspeech*, pages 1197–1200, Makuhari, Japan, September 2010.

[40] Lawrence Rabiner and Biing-Hwang Juang. *Fundamentals of speech recognition*. Prentice-Hall, 1993.

[41] D. E. Rumelhart, G. E. Hinton, and R. J. Williams. Parallel distributed processing: explorations in the microstructure of cognition, vol. 1. chapter Learning internal representations by error propagation, pages 318–362. MIT Press, Cambridge, MA, USA, 1986.

[42] D. Rybach, C. Gollan, G. Heigold, B. Hoffmeister, J. Lööf, R. Schlüter, and H. Ney. The RWTH Aachen University open source speech recognition system. In *10th Annual Conference of the International Speech Communication Association*, pages 2111–2114, 2009.

[43] D. Rybach, C. Gollan, G. Heigold, B. Hoffmeister, J. Lööf, R. Schlüter, and H. Ney. The RWTH Aachen University Open Source Speech Recognition System. In *Proc. of Interspeech 2009*, pages 2111–2114, Brighton, UK, 2009.

[44] Holger Schwenk and Philipp Koehn. Large and Diverse Language Models for Statistical Machine Translation. In *Proceedings of the International Joint Conference on Natural Language Processing*, pages 661–666, Hyderabad, India, 2008.

[45] R. Smith. An overview of the tesseract ocr engine. In *Proceedings of the Ninth International Conference on Document Analysis and Recognition - Volume 02*, ICDAR '07, pages 629–633, Washington, DC, USA, 2007. IEEE Computer Society.

[46] Matthew Snover, Bonnie Dorr, Richard Schwartz, Linnea Micciulla, and John Makhoul. A study of translation edit rate with targeted human annotation. In *In Proceedings of Association for Machine Translation in the Americas*, pages 223–231, 2006.

[47] H. Soltau, G. Saon, and B. Kingsbury. The IBM Attila Speech Recognition Toolkit. In *IEEE Workshop on Spoken Language Technology*, pages 85–90, Berkeley, CA., 2010.

[48] Ralf Steinberger, Bruno Pouliquen, Anna Widiger, Camelia Ignat, Tomaz Erjavec, Dan Tufis, and Daniel Varga. The JRC-Acquis: A multilingual aligned parallel corpus with 20+ languages. In *Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC 2006)*, Genoa, Italy, May 2006.

[49] Ralf Steinberger, Bruno Pouliquen, Anna Widiger, Camelia Ignat, Tomaz Erjavec, Dan Tufis, and Daniel Varga. The JRC-Acquis: A multilingual aligned parallel corpus with 20+ languages. In *5th International Conference on Language Resources and Evaluation*, Genoa, Italy, May 2006.

[50] G. Stemmer, F. Brugnara, and D. Giuliani. Adaptive training using simple target models. In *Acoustics, Speech, and Signal Processing, 2005. Proceedings. (ICASSP '05). IEEE International Conference on*, volume 1, pages 997 – 1000, 18-23, 2005.

[51] G. Stemmer, F. Brugnara, and D. Giuliani. Adaptive training using simple target models. In *Proc. of the IEEE Int. Conf. on Acoustics, Speech, and Signal Processing*, pages 997 – 1000, Philadelphia, PA., 2005.

[52] Andreas Stolcke. SRILM – an extensible language modeling toolkit. In *Proc. of ICSLP*, 2002.

[53] Andreas Stolcke. SRILM – An Extensible Language Modeling Toolkit. In *Proc. of the Int. Conf. on Speech and Language Processing (ICSLP)*, volume 2, pages 901–904, Denver, CO, September 2002.

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

[55] Jinsong Su, Hua Wu, Haifeng Wang, Yidong Chen, Xiaodong Shi, Huailin Dong, and Qun Liu. Translation Model Adaptation for Statistical Machine Translation with Monolingual Topic Information. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 459–468, Jeju Island, Korea, July 2012. Association for Computational Linguistics.

[56] Muhammad Ali Tahir, Markus Nußbaum, Ralf Schlüter, and Hermann Ney. Simultaneous discriminative training and mixture splitting of hmms for speech recognition. In *Interspeech*, Portland, OR, USA, September 2012.

[57] Nicola Ueffing, Gholamreza Haffari, and Anoop Sarkar. Semi-supervised Model Adaptation for Statistical Machine Translation. *Machine Translation*, 21:77–94, 2007.

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

[59] David Vilar, Daniel Stein, Matthias Huck, and Hermann Ney. Jane: Open source hierarchical translation, extended with reordering and lexicon models. In *ACL 2010 Joint Fifth Workshop on Statistical Machine Translation and Metrics MATR*, pages 262–270, Uppsala, Sweden, July 2010.

[60] L. Welling, S. Kanthak, and H. Ney. Improved methods for vocal tract normalization. In *Acoustics, Speech, and Signal Processing, 1999. Proceedings., 1999 IEEE International Conference on*, volume 2, pages 761 –764 vol.2, mar 1999.

[61] L. Welling, S. Kanthak, and H. Ney. Improved Methods for Vocal Tract Normalization. In *Proc. of the IEEE Int. Conf. on Acoustics, Speech, and Signal Processing*, pages 761–764, Phoenix, AZ., 1999.

[62] Joern Wuebker, Hermann Ney, and Richard Zens. Fast and scalable decoding with language model look-ahead for phrase-based statistical machine translation. In *Annual Meeting of the Assoc. for Computational Linguistics*, pages 28–32, Jeju, Republic of Korea, July 2012.

[63] S. Young et al. *The HTK Book*. Cambridge University Engineering Department, 1995.

[64] S. J. Young, J. J. Odell, and P. C. Woodland. Tree-based state tying for high accuracy acoustic modelling. In *Proceedings of the workshop on Human Language Technology*, HLT '94, pages 307–312, Stroudsburg, PA, USA, 1994. Association for Computational Linguistics.

A Acronyms

UPVLC	Universitat Politècnica de València
XRCE	XEROX Research Center Europe
JSI	Josef Stefan Institute
K4A	Knowledge for All Foundation
RWTH	RWTH Aachen University
EML	European Media Laboratory GmbH
DDS	Deluxe Digital Studios Limited
ASR	Automatic Speech Recognition
HMM	Hidden Markov Model
HTR	Handwritten Text Recognition
MLE	Maximum Likelihood Estimation
MLLR	Maximum Likelihood Linear Regression
WER	Word Error Rate