rijksuniversiteit
groningen

Erasmus Mundus European Master in
Language & Communication Technologies (LCT)
Master of Science in Computer Science
Free University of Bozen-Bolzano
Research Master in Linguistics
University of Groningen

Topic Adaptation for Lecture Translation through Bilingual Latent Semantic Models

Thesis Submission for a
Master of Science in
Computer Science

Nicholas Ruiz
Defense on
23^{rd}, March, 2012

Supervisor: Marcello Federico, FBK-irst
Co-Supervisor: Giancarlo Succi, Free University of Bozen-Bolzano
Co-Supervisor: Gertjan van Noord, University of Groningen

Acknowledgments

I would first like to express my gratitude to Marcello Federico for his support, encouragement, and advice during the completion of the thesis. Marcello's expertise and excitement for statistical machine translation helped me find my research interests. I greatly appreciate the opportunity to work alongside him and other colleagues at FBK-irst and look forward to future research collaboration. I would also like to thank Nicola Bertoldi, Mauro Cettolo, and Arianna Bisazza for always being available to answer my questions and for providing recommendations to enhance my research experiments.

I would also like to thank my university supervisors, Gertjan van Noord and Giancarlo Succi for their evaluation and feedback on the progress of my thesis.

I would also like to thank Andreas Eisele, who accepted my interest in machine translation with enthusiasm and introduced me to the machine translation research community through a summer internship with DFKI in Saarbrücken.

I would also like to thank the past and present local LCT coordinators at the University of Groningen and the Free University of Bozen-Bolzano: Raffaella Bernardi, Gosse Bouma, Valeria Fionda, and Gisela Redeker. In particular, I would like to thank Gisela Redeker for helping me to get acclimated to my first year of study in Groningen and for many fruitful meetings to plan my schedule and discuss research interests. I would also like to thank Valia Kordoni for her coordination of the entire LCT consortium.

I would additionally like to thank Bobbye Pernice for helping me overcome a multitude of administrative hurdles in my journeys from Groningen to Saarbrücken to Bolzano. In spite of assisting dozens of students each year, she constantly focused on my needs as if I was the only student to help.

I additionally would like to thank my family for their constant support. I would like to thank my wife Jennifer for her unceasing love and encouragement and for sharing this journey with me. And finally, I would like to thank God for opening doors that in normal circumstances could never have been opened and for the motivation to persevere in everything I do.

Abstract

Language models (LMs) are used in Statistical Machine Translation (SMT) to improve the fluency of translation output by assigning high probabilities to sequences of words observed in training data. However, SMT systems are trained with large amounts of data that may differ in style and genre from a text to be translated. Language models can be adapted through various techniques, including topic modeling approaches, which describe documents as a mixture of topics.

Several bilingual topic modeling approaches have been recently constructed to adapt language models to reward translated word sequences that include words that better fit the topics that represent the translation text. Most topic modeling approaches use Latent Dirichlet Allocation, which makes a prior assumption about the distribution of topics within a document.

This work presents a simplified approach to bilingual topic modeling for language model adaptation by combining text in the source and target language into very short documents and performing Probabilistic Latent Semantic Analysis (PLSA) during model training. During inference, documents containing only the source language can be used to infer a full topic-word distribution on all words in the target language's vocabulary, from which we perform Minimum Discrimination Information (MDI) adaptation on a background language model (LM). We apply our approach on the English-French IWSLT 2010 TED Talk exercise, and report a 15% reduction in perplexity and relative BLEU and NIST improvements of 3% and 2.4%, respectively over a baseline only using a 5-gram background LM over the entire translation task. Our topic modeling approach is simpler to construct than its counterparts.

Keywords:
Machine Translation, Language Modeling, Topic Adaptation, Topic Modeling

Contents

Chapter 1

Introduction

> If as one people speaking the same language they have begun to do this, then nothing they plan to do will be impossible for them.
>
> Genesis 11:6 (NIV)
> *The Bible*

According to the story of the Tower of Babel, people once spoke a common language. Since those times in the past, there are thousands of languages being spoken all over the world, causing great difficulties in multicultural communication. Machine translation is an important breakthrough that has greatly improved the quality of multilingual interaction in society. Like many natural language processing tasks, machine translation involves the use of computers to automatically translate from one language to another. After over 50 years of research, beginning in the World War II era, machine translation is still an open problem. While research areas like speech recognition have a clear objective to transcribe sound into the actual words perceived by a human being, translation itself has many complications, due to the fact that even language experts do not necessarily agree on the translation for a given sentence or utterance.

Machine translation began with rule-based systems, such as Systran[1] in the late 1960s, which rely on numerous linguistic rules and bilingual dictionaries for each language pair. A source text is parsed and converted into an interlingual representation and the output text in the target language is generated based on lexicons with syntactic, semantic, and morphological information and a cascading set of rules.

Today, statistical machine translation is the state of the art approach. Large collections of parallel corpora to learn the likelihood of phrases being translated from source to target language and large monolingual corpora are used to ensure the fluency of the target output. The goal is to generate a translation that has the highest likelihood of observation, given a model constructed by the training data.

[1] http://www.systran.co.uk/

1.1 Scope of the Thesis

In the task of lecture translation, lectures can be categorized as formal or informal. Formal lectures consist of the use of formal English constructions; the style of the lecture is highly structured and the speaker will be directly to the point. In informal lectures, however, an author may use colloquial language and expressions that are not often used in formal contexts. Additionally, lectures can cover a wide variety of topics. On the TED (Technology, Entertainment, Design) website[2], users have tagged online lectures with approximately 300 topics which may not be mutually exclusive. Such a diversity in speaker style and topic selection undermine the robustness of statistical machine translation systems trained with data from specific domains and literary styles.

In this thesis, we focus on the problem of language model adaptation in the domain of lecture translation. By analyzing the text that is to be translated, we can learn interesting statistics on a speaker's word selection and style that can be used to adapt the probability of word sequences in a background language model in a manner that improves the fluency and adequacy of word selection during the translation phase.

Our goal is to construct a simplified language modeling framework based on existing topic modeling approaches that use the notion of topics to assign probabilities to sequences of words. We evaluate the utility of this framework in the context of lecture translation in the task of translating TED lectures from English to French.

1.2 Structure of the Thesis

The remainder of the thesis is structured as follows:

Chapter 2 provides an overview of statistical machine translation. The problem of statistical machine translation is decomposed into models related to translation lookup tables, alignment models, and language models. The word alignment problem and its utility in aligning phrases to be translated is discussed, in addition to the training, tuning, and evaluation of phrase-based machine translation systems.

Chapter 3 provides an overview on language models, which model the fluency of translation output. The Markov assumption in language modeling is discussed, particularly regarding how n-gram language models are constructed. The problem of data sparsity is addressed by identifying smoothing techniques to improve the quality of language models. Evaluation metrics are presented to judge the quality of language models and finally techniques are outlined to adapt language models to texts that differ in domain and genre.

Chapter 4 discusses the concept topic modeling, in which a collection of texts can be described by a subset of latent topics. Two particularly popular topic modeling techniques are introduced: Probabilistic Latent Semantic Analysis and Latent Dirichlet Allocation; a comparison of the two techniques is provided. Subsequently,

[2] http://www.ted.com

an outline of how topic modeling can be used in conjunction with language model adaptation is presented.

Chapter 5 presents an extension of topic modeling to bilingual scenarios, such as machine translation. Several methods that researchers have used to adapt language models within the discipline in statistical machine translation are introduced – several of which use topic modeling approaches. We then discuss our simplified bilingual topic modeling approach that combines the theory from the previous chapters. We evaluate our approach in the context of lecture translation under the IWSLT 2010 TED talk translation task.

Chapter 6 summarizes the topics covered in this thesis. Future research topics are mentioned, which include techniques for multilingual topic modeling and ideas regarding how to simplify the task of language model adaptation.

Chapter 2

Statistical Machine Translation

2.1 The Noisy Channel Model

In 1947 and later elaborated in 1949, Warren Weaver, the vice president of the Rockefeller Foundation, defined a generative story for the translation of Russian to English, as follows:

> When I look at an article in Russian, I say: 'This is really written in English, but it has been coded in some strange symbols. I will now proceed to decode' (Weaver, 1949/1955).

While translation is not simply an exercise in decoding, Weaver's generative story serves as the foundation of statistical machine translation by suggesting that translation can be modeled by using Shannon (1948)'s *noisy-channel model*. Figure 2.1 provides a graphical representation of Weaver's conception of statistical machine translation.

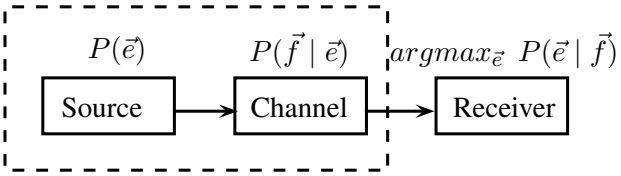


Figure 2.1: Noisy-channel model for translation. An English message is passed through a noisy channel, which causes a "corruption" of the message to a foreign language. The original English message is reconstructed via a source model $P(\vec{e})$ and a channel model $P(\vec{f} \mid \vec{e})$.

The goal in statistical machine translation is to reconstruct the "original" English message that is passed through a noisy channel. This is done by a generative model in which we maximize the posterior probability of an English message $\vec{e} = (e_1, e_2, ..., e_{l_{\vec{e}}})$ (described as a vector of words) given that we have observed the foreign message $\vec{f} = (f_1, f_2, ..., f_{l_{\vec{f}}})$. Using Bayes' rule, we can express this probability as:

$$P(\vec{e} \mid \vec{f}) = \frac{P(\vec{f} \mid \vec{e})P(\vec{e})}{P(\vec{f})}. \tag{2.1}$$

To find the most likely English message, we find the $\vec{e}$ with the maximum probability. Since we are only concerned with determining the most likely $\vec{e}$, $P(\vec{f})$ becomes a constant term that can be discarded.

$$\arg\max_{\vec{e}} P(\vec{e} \mid \vec{f}) = \arg\max_{\vec{e}} \frac{P(\vec{f} \mid \vec{e})P(\vec{e})}{P(\vec{f})}$$
$$= \arg\max_{\vec{e}} P(\vec{f} \mid \vec{e})P(\vec{e}). \tag{2.2}$$

From the noisy-channel model, the objective function in (2.2) is understood as a composition of a source model and a channel model. In SMT, the source model is referred to as a *language model* (LM), which models fluent English output. The channel model is referred to as the *translation model* (TM), which models the conditional probability of English words (or phrases) given foreign words. The language model is estimated using monolingual text, while the translation model is constructed using parallel texts (or *bitexts*) – passages of text that are typically aligned at the sentence or clause level. Several schemes have been attempted to construct translation models, based on lexical alignments and phrase-based alignments.

While the noisy-channel description above refers to the translation of foreign words to English, the same construction exists without loss of generality for translation tasks of any source language to any target language.

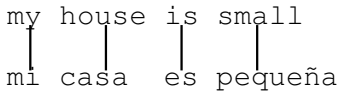
2.2 Lexical Translation Models

We begin our discussion of translation modeling with constructions that translate words in isolation. These approaches are also known as *lexical translation* models. These models originate from early work on statistical machine translation by the IBM Candide project in the late 1980s and early 1990s (Brown et al., 1993). In order to learn translations of individual words, it is first necessary to construct a bilingual dictionary (or translation table) by deriving a probability distribution of words from the source language being aligned to words in the target language. Assuming that the alignments between words in our parallel text are known, we can simply use *maximum likelihood estimation* to calculate the probability distribution of the data; however, in normal translation scenarios, we only know the alignment of sentences and must learn the word alignments.

We can define an alignment function a as a mapping between positions of each target output word at position i to a source input word at position j:

$$a : j \rightarrow i \tag{2.3}$$

If a sentence pair has a direct alignment, the source and target sentences have the same number of words and the words will be aligned in the exact same order, such as the following English to Spanish translation example:

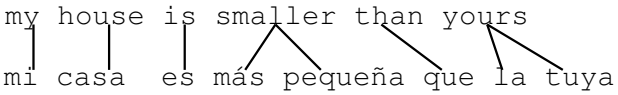


This alignment is represented by the mapping:

$$a : \{1 \to 1, 2 \to 2, 3 \to 3, 4 \to 4\} \tag{2.4}$$

In other cases, languages may differ in word order:

In Spanish, the comparative uses more words than its English counterpart, thus two words are required to capture the meaning of "smaller":



which yields the mapping:

$$a : \{1 \to 1, 2 \to 2, 3 \to 3, 4 \to 4, 4 \to 5, 5 \to 6, 6 \to 7, 6 \to 8\} \tag{2.5}$$

Other alignment scenarios include source words that should be dropped during translation and words in the target language that do not have an equivalent source word, under which we introduce a special NULL token. While unknown at the time of translation, the alignment of words is an important consideration in defining the best translation; thus we can express the translation model as:

$$\arg\max_{\vec{e}} P(\vec{e} \mid \vec{f}) = \arg\max_{\vec{e}} \sum_{a} P(\vec{e}, a \mid \vec{f}), \tag{2.6}$$

which marginalizes over all possible word alignments in the sentence. Such a model favors translations that better follow the alignment rules of the translation pair. All the following derivations are due to Brown et al. (1993). We follow here the exposition given in Koehn (2010).

2.2.1 IBM Model 1

Each IBM model describes a generative story for how the joint conditional probabilities $P(\vec{e}, a \mid \vec{f})$ are computed. IBM Model 1 is a generative model that only uses the lexical translation probability distributions, defined as $t(e_j \mid f_{a(j)})$. Model 1 assumes that each word in the sentence is independently translated, thus ignoring context. As such, the translation probability for a foreign sentence $\vec{f}$ of length l_f to an English sentence $\vec{e}$ of length l_e is defined as:

$$P(\vec{e}, a \mid \vec{f}) = \frac{\varepsilon}{(l_f + 1)^{l_e}} \prod_{j=1}^{l_e} t(e_j \mid f_{a(j)}), \tag{2.7}$$

where ε is a normalization factor.

Training. Since we are only given a sentence-aligned corpus, we do not know the word alignments, or the translation probabilities of words in the corpus. Thus, we need to use *Expectation Maximization* (EM) (Dempster et al., 1977). In the E-step, we compute $P(a \mid \vec{e}, \vec{f})$, the probabilities of different alignments given a sentence pair. Using the chain rule and Bayes' formula, we compute (2.8):

$$P(a \mid \vec{e}, \vec{f}) = \frac{P(\vec{e}, a \mid \vec{f})}{P(\vec{e} \mid \vec{f})}. \tag{2.8}$$

Koehn (2010) derives $P(\vec{e} \mid \vec{f})$ into the tractable solution:

$$P(\vec{e} \mid \vec{f}) = \frac{\varepsilon}{(l_f + 1)^{l_e}} \prod_{j=1}^{l_e} \sum_{i=0}^{l_f} t(e_j \mid f_i). \tag{2.9}$$

Thus, after simplification, (2.8) becomes:

$$P(a \mid \vec{e}, \vec{f}) = \prod_{j=1}^{l_e} \frac{t(e_j \mid f_{a(j)})}{\sum_{i=0}^{l_f} t(e_j \mid f_i)}, \tag{2.10}$$

which simply describes the E-step as computing the factored probabilities of each word translation and normalizing it over each possible translation pair.

As described in Koehn (2010), the M-step consists of collecting counts for the word translations over all possible alignments, weighted by their probability. The counts are computed as follows:

$$c(e \mid f; \vec{e}, \vec{f}) = \sum_{a} P(a \mid \vec{e}, \vec{f}) \sum_{j=1}^{l_e} \delta(e, e_j)\delta(f, f_{a(j)}), \tag{2.11}$$

where the Kronecker delta function $\delta(x, y)$ is 1 if $x = y$ and 0 otherwise. Using maximum likelihood estimation, we estimate the new translation probability distribution by:

$$t(e \mid f; \vec{e}, \vec{f}) = \frac{\sum_{(\vec{e},\vec{f})} c(e \mid f; \vec{e}, \vec{f})}{\sum_{e} \sum_{(\vec{e},\vec{f})} c(e \mid f; \vec{e}, \vec{f})}. \tag{2.12}$$

Model 1 fails to handle the alignment scenarios of adding and dropping words, as described above. Additionally, it does not handle reordering well.

2.2.2 IBM Model 2

IBM Model 2 accounts for Model 1's faults by incorporating local alignment through the modeling of the probability distribution: $a(i \mid j, l_e, l_f)$. This distribution models the likelihood that an arbitrary foreign sentence of length l_f aligns position j with position i in any English translation of length l_e, without accounting for the actual words at these positions. Used in conjunction with Model 1, Model 2 is defined as:

$$P(\vec{e}, a \mid \vec{f}) = \varepsilon \prod_{j=1}^{l_e} t(e_j \mid f_{a_j}) a(a(j) \mid j, l_e, l_f). \tag{2.13}$$

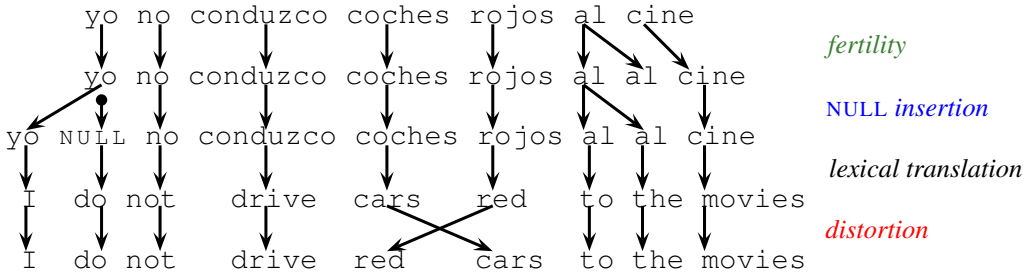
Similar to Model 1, Koehn (2010) shows that the conditional probability of English sentences given foreign sentences calculated in the E-step of training can be simplified to a problem with polynomial complexity, with the result of:

$$P(\vec{e} \mid \vec{f}) = \varepsilon \prod_{j=1}^{l_e} \sum_{i=0}^{l_f} t(e_j \mid f_{a_j}) a(a(j) \mid j, l_e, l_f). \tag{2.14}$$

The count functions for the lexical translation and alignment probability distributions used in the M-step follow similar maximum likelihood estimate computations to Model 1. Essentially, Model 1 is a special case of Model 2, in which the alignment probability distribution is fixed to a uniform distribution, with respect to the number of foreign words in the sentence.

2.2.3 IBM Model 3

IBM Model 3 adds the additional concept of *fertility*, which models the probability distribution of foreign word f generating $\phi = 0, 1, 2, ...$ output words, written as $n(\phi \mid f)$. Additionally, NULL tokens ϕ_0 can be inserted for each generated word, under a binomial distribution with insertion probability p_1. Model 3 consists of four steps, outlined below with an English to Spanish lexical translation example.



Fertility is modeled in the first step, where the word *al* is duplicated under the probability $n(2 \mid \text{al})$. NULL insertion is modeled by the insertion of the NULL token after *yo*. The lexical translation and distortion steps closely follow IBM Models 1 and 2, respectively. However, the distortion probability distribution $d(j \mid i, l_e, l_f)$ predicts output word positions based on the input word positions – the opposite direction of the alignment probability distribution in Model 2. Like Model 2, the distortion model does not depend on word identities. Due to fertility, there can be multiple ways that the foreign words can produce the same output; thus, we must sum the probabilities of each possible construction (i.e. over all possible alignments of each word) via a *tableau*.

Combining the four steps, the conditional probability of English words given

foreign words is now:

$$\begin{aligned} P(\vec{e} \mid \vec{f}) &= \sum_{a} P(\vec{e}, a \mid \vec{f}) \\ &= \sum_{a(1)=0}^{l_f} \cdots \sum_{a(l_e)=0}^{l_f} \prod_{j=1}^{l_e} \binom{l_e - \phi_0}{\phi_0} p_1^{\phi_0} p_0^{l_e - 2\phi_0} \prod_{i=1}^{l_f} \phi_i! n(\phi_i \mid e_i) \\ &\quad \times \prod_{j=1}^{l_e} t(e_j \mid f_{a(j)}) d(j \mid a(j), l_e, l_f). \end{aligned} \tag{2.15}$$

Unfortunately, there is no tractable computation for (2.15) due to an exponential number of possible alignments. As a workaround, we employ *hill climbing*, in which alignment comprising most of the probability mass are sampled by exploring neighboring alignments that differ by a *move* (i.e. a difference in alignment of only one word), or a *swap* (i.e. all words except for two have identical alignments; those two differing alignments have exchanged their alignment points). Sadly, hill climbing can result in local maxima, since the maximized likelihood function is not convex; however, by initializing our best alignment with the results of IBM Model 2, we obtain a reliable starting point for the search. Koehn (2010) provides a detailed explanation of the implementation of Model 3.

2.2.4 IBM Model 4

Unfortunately, Model 3 does not have sufficient statistics to model the distortion for long sentences. IBM Model 4 further improves on the distortion step from Model 3 by introducing a *relative distortion* model. Model 4 makes a more stringent assumption, based on the notion that in most cases, reordering occurs locally and is context-dependent – in particular, it is dependent on the preceding input word.

As described in Koehn (2010), each foreign word f_i that is aligned to at least one output word forms a *cept* π_i, which defines a span of output positions that are filled by the alignment of f_i in a particular tableau. The relative distortion of each output word is determined by three cases:

1. Words generated from the NULL have a uniform distortion distribution.
2. The first word in a cept is distorted with a probability distribution defined by its positional distance from the center of the preceding cept.
3. Subsequent words in the cept are distorted with a probability distribution relative to their positional distance from the placement of the previous word in the cept.

Model 4 also expresses alignments in terms of parameters that depend on the classes of words that lie at the aligned positions (Berger et al., 1994). Since individual words will not yield sufficient statistics to properly model distortion distributions, Model 4 assumes that words are clustered into classes. Thus, we can define

the distortion distributions as follows:

$$\begin{aligned} \text{for an initial word in a cept:} \quad & d_1(j - \odot_{i-1} \mid A(f_{i-1}), B(e_j)) \\ \text{for additional words:} \quad & d_{>1}(j - \pi_{i,k-1} \mid B(e_j)), \end{aligned} \tag{2.16}$$

where $\odot_{i-1}$ is the center of the previous cept, $\pi_{i,k-1}$ is the position of the $k-1$th word in the cept, and A and B map foreign and English words to their classes, respectively. Training of Model 4 requires similar hill climbing techniques to Model 3.

There is additionally a Model 5, which models *deficiency*, in which multiple output words could be placed in the same position in Models 1-4. Deficiency is resolved by keeping track of the number of vacant positions in the output sentence and enforcing that remaining words fill only these positions.

2.3 Phrase-Based Models

Phrase-based SMT models are currently the best performing systems. Unlike word-based models, phrase-based models are capable of translating chunks of words at a time. The notion of a "phrase" should not be confused with a linguistic understanding of phrases; chunks may be smaller or larger than a phrasal constituent and could potentially span across multiple constituents. As a result, the translation models are simpler, since the concepts of fertility, NULL word insertion, and deletion are no longer necessary. Instead, we assume that foreign and English sentences are decomposed into exactly I phrases $\bar{f}_i$ and $\bar{e}_i$. Under the phrase paradigm, the translation model becomes:

$$P(\vec{f} \mid \vec{e}) = \prod_{i=1}^{I} \phi(\bar{f}_i \mid \bar{e}_i) d(\text{start}_i - \text{end}_{i-1} - 1), \tag{2.17}$$

where $\phi(\bar{f}_i \mid \bar{e}_i)$ models the phrase translation probability and $d(\cdot)$ is a *distance-based reordering model*, relative to the position of the last word in the previous phrase. $d(\cdot)$ assigns an exponentially decaying cost function for the number of words skipped in either direction from the position of the previous phrase (Och and Ney, 2004).

2.3.1 Building the Translation Table

Using source-to-target alignments generated from the IBM Models and alignments in the reverse direction, we can extract phrase pairs by finding word alignments $(\bar{f}, \bar{e})$ that "match up consistently" with an alignment A (Koehn, 2010). Under the definition of *consistency*, all words $e_i \in \bar{e}$ must be aligned with $f_j \in \bar{f}$ and vice versa; additionally, all such alignments must be defined in A, ignoring unaligned and NULL aligned words in the phrase pair. Under the phrase extraction algorithm, consistent phrase pairs are extracted from the word-aligned sentence pair. After completion, any unaligned foreign words are merged into neighboring foreign phrases and added as additional translations of the corresponding English phrases.

Translation table probabilities are estimated via maximum likelihood estimation, given the counts $n(\cdot)$ of the extracted phrase pairs:

$$\phi(\bar{f} \mid \bar{e}) = \frac{n(\bar{e}, \bar{f})}{\sum_{\bar{f}_i} n(\bar{e}, \bar{f}_i)}. \tag{2.18}$$

2.3.2 Reordering

Instead of the simplified reordering model described in (2.17), we can construct a lexicalized reordering model, in which reordering is dependent on the actual phrase pair. To overcome sparsity among phrase pairs, three reordering *orientations* are defined:

- *monotone*: the phrase incurs no reordering.
- *swap*: the phrase swaps positions with the previous phrase.
- *discontinuous*: the phrase does not swap with an adjacent phrase and is not monotonic (allows for long-distance reordering).

The probability distribution of each orientation is estimated using maximum likelihood and can be smoothed to account for sparse phrase pairs as follows:

$$p_o(\text{orientation} \mid \bar{f}, \bar{e}) = \frac{\sigma p_o(\text{orientation}) + n(\text{orientation}, \bar{e}, \bar{f})}{\sigma + \sum_o n(o, \bar{e}, \bar{f})}, \tag{2.19}$$

where σ is a smoothing factor and $p_o(\text{orientation})$ is the observed likelihood of a given orientation over all alignment pairs.

2.4 Log-Linear Models

Given that our translation model $P(\vec{f} \mid \vec{e})$ consists of a phrase translation table $\phi(\bar{f} \mid \bar{e})$ and a reordering model $d(\cdot)$, the phrase-based generative model is factorized as:

$$\vec{e}_{\text{best}} = \arg\max_{\vec{e}} \prod_{i=1}^{I} \phi(\bar{f}_i \mid \bar{e}_i) d(\cdot) \prod_{i=1}^{|\vec{e}|} p_{\text{LM}}(e_i \mid e_1 ... e_{i-1}) \tag{2.20}$$

when combining the translation model with the language model. (2.20) assumes that each component has equal weight; however, this is empirically not the case. We assign specific weights through the construction of a log-linear model, in which each component is exponentially scaled by its corresponding weight λ_ϕ, λ_d, and λ_{LM}. Thus, our phrase-based model becomes:

$$P(\vec{e}, a \mid \vec{f}) = \exp\left[\lambda_\phi \sum_{i=1}^{I} \log \phi(\bar{f}_i \mid \bar{e}_i) + \lambda_d \sum_{i=1}^{I} \log d(\cdot) + \lambda_{\text{LM}} \sum_{i=1}^{|\vec{e}|} \log p_{\text{LM}}(e_i \mid e_1 ... e_{i-1}) \right] \tag{2.21}$$

In the case of the reordering model described in Section 2.3.2, the $d(\cdot)$ log-linear feature function is factorized into three features with individual weights, corresponding to each orientation type. Likewise, the translation table and language model features can also be factorized into multiple log-linear feature functions with distinct weights. Techniques to optimize the feature weights are described in Section 2.7.

2.5 Decoding

Given a trained phrase-based model, our goal is to translate unobserved sentences. Again, assuming the foreign language to English scenario, our generative model as described by Warren Weaver is to decode foreign sentences into English sentences using the most likely translations:

$$\vec{e}^{*} = \arg\max_{\vec{e}} P(\vec{e}) \sum_{a} P(\vec{f}, a \mid \vec{e}). \tag{2.22}$$

We generally use the *Viterbi approximation*, which states that a single alignment sticks out with the highest probability. Thus, we approximate (2.22) as:

$$\vec{e}^{*} \approx \arg\max_{\vec{e}} P(\vec{e}) \max_{a} P(\vec{f}, a \mid \vec{e}), \tag{2.23}$$

which allows the use of a *beam-search* dynamic programming algorithm to compute the best translation. Tillmann and Ney (2003) describe a specific algorithm, called *DP beam-search*, which is similar to that used in the Moses toolkit. Since there are still exponentially many possible translation options listed in the phrase translation table for the phrase chunks in the input sentence, several search heuristics are necessary to reduce the computational complexity of the decoding phase.

The beam-search algorithm incrementally constructs hypothesis that consist of partial translations of the input sentence. The first step of the decoding process begins with the empty hypothesis. The hypothesis is expanded by selecting each translation option that generates the initial phrase in the English sentence. Expanded hypotheses are placed in a stack that corresponds to the number of English words covered by the hypothesis (i.e. if hypothesis h contains i translated words, it is placed in the ith stack). Each hypothesis has an associated cost, determined by its current cost and its future cost. The current cost for a set of partially translated phrases is determined from the probability of the phrases already in the hypothesis, which, in the case of a log-linear model, follows (2.21). A low probability corresponds with a high cost. The future cost is the expected cost of translating the rest of the sentence. The future cost of the remaining span is total cost of each contiguous untranslated span, which is estimated as the minimum of the cost of the entire span, or a decomposition of the span into two smaller units.

Pruning techniques are used to limit the number of hypotheses per stack. In histogram pruning, a maximum number of n hypotheses with the lowest cost are preserved in each stack. In threshold pruning, hypotheses with scores that are worse than the best hypothesis in its corresponding stack by a specific threshold α are pruned.

2.6 Evaluation

Several evaluation techniques exist to assess the quality of machine translation output. The most widely used metric is *BLEU*, the "BiLingual Evaluation Understudy" (Papineni et al., 2001), which was developed by IBM. Another similar measure is the *NIST* score (Doddington, 2002).

2.6.1 BLEU

BLEU is a numeric metric based on the similarity between texts. In the context of translation, it uses n-gram based matching between MT outputs against reference translations. The geometric average of modified n-gram precisions p_n are computed, using n-grams up to length N (typically 4) and positive weights w_n attributed to each n-gram level, which sum to one. For typical evaluations, uniform weights are assumed. A *brevity penalty* is introduced to ensure that exceedingly short translations are not favored over longer translations. The brevity penalty is defined as:

$$\mathrm{BP} = \begin{cases} 1 & \text{if } L_{\text{sys}} > \bar{L}_{\text{ref}} \\ e^{(1-\bar{L}_{\text{ref}})/L_{\text{sys}}} & \text{if } L_{\text{sys}} \leq \bar{L}_{\text{ref}} \end{cases}, \tag{2.24}$$

where L_{sys} is the candidate translation length and $\bar{L}_{\text{ref}}$ is the average reference translation length. Thus,

$$\mathrm{BLEU}_N = \mathrm{BP} \cdot \exp\left(\sum_{n=1}^{N} w_n \log p_n\right). \tag{2.25}$$

BLEU scores range between the interval [0,1], based on the n-gram similarity between the candidate and reference. It should be noted that BLEU scores are relative to the translation task and thus cannot be compared universally. BLEU uses a geometric mean of co-occurrences.

2.6.2 NIST

NIST, whose name comes from the US National Institute of Standards and Technology, is a modification of BLEU that calculates the informativeness of particular n-grams, awarding rarer n-gram matches with higher weights. Information weights are computed using n-gram counts over the set of reference translations in a manner that favors rarer n-grams:

$$Info(w_1 ... w_n) = \log \frac{n(w_1 ... w_{n-1})}{n(w_1 ... w_n)}, \tag{2.26}$$

where $n(\cdot)$ is a function that determines the number of occurrences of an n-gram in the reference translations. NIST uses an arithmetic mean of n-gram counts, rather than BLEU's geometric mean. The full equation is:

$$NIST_N = \sum_{n=1}^{N} \{ \frac{\sum_{w_1 ... w_n \text{that co-occur}} Info(w_1 ... w_n)}{\sum_{w_1 ... w_n \text{ in system output}} (1)} \cdot \exp\left(\beta \left[\min\left(L_{\text{sys}}/\bar{L}_{\text{ref}}, 1\right)\right)\right] \}, \tag{2.27}$$

where β is a weight chosen to enforce a brevity penalty factor of 0.5 when the average number of words in the system output is two-thirds that of the reference translations. NIST is usually calculated with $N = 5$ n-grams.

2.7 Tuning

In Section 2.4, we constructed a log-linear phrase-based translation model in (2.21). One way to learn the optimal weights λ_i for each feature function $h_1, ..., h_m$ is to employ *minimum error rate training* (MERT) (Och, 2003). In MERT, the objective is to find the optimal $\hat{\lambda}_1, ..., \hat{\lambda}_m$ weights that minimize the error between a set of K candidate translations for each foreign sentence in a tuning set – parallel texts that were not used to train the components of the phrase-based translation model. The objective function can be a metric such as BLEU (n-gram based matching against reference translations) (Papineni et al., 2001), or another evaluation metric used for testing the output from the decoder, such as NIST (Doddington, 2002) or ranking error.

MERT iterates over the training process by translating the tuning set and generating n-best lists; after which the optimal parameters are calculated. MERT iterates until the optimal parameters $\hat{\lambda}_i$ converge, or a fixed number of iterations is exceeded. One search strategy is Powell's method (Och, 2003). Powell's method involves iteratively finding the single weight update that yields the highest improvement on the error score and makes the change. Och (2003) shows that Powell's method explores threshold points at which a change in candidate λ_c causes changes in the highest ranked sentences in an n-best list. Threshold points are collected for all sentences in the tuning set and the λ_c value that yields the best overall error score becomes the new parameter value for the iteration.

2.8 Chapter Summary

In this chapter we provided an overview of statistical machine translation. We defined a generative model for SMT based on the noisy-channel model. We outlined the original IBM models for word alignment, which assume that words are individually translated in a sentence. We then extended the discussion to phrase-based models, which leverage word alignments from the IBM models to construct a phrase translation table and a richer reordering model. We then factorized the phrase-based translation model into a log-linear model composed of a phrase translation table feature, a reordering feature, and a language modeling feature that can be assigned different weights.

We additionally summarized the decoding process in which an input sentence is translated into an output sentence and outlined techniques to reduce the search space into a tractable problem via a beam-search. We next discussed evaluation techniques for SMT, such as BLEU and NIST, which use n-gram based matching

techniques between candidate translations and reference translations to assess the adequacy and fluency of translation output. Finally, we discussed the tuning phase, in which the feature weights from the phrase-based log-linear translation model are optimized, given a held-out tuning set of parallel sentences.

Chapter 3

Language Modeling

The *language model* (LM) is an important component of a statistical machine translation system that measures the fluency of translated text. It does so by defining a probability distribution that describes the likelihood of a sequence of words being uttered or written by a native speaker. As part of the log-linear model described in Section 2.4, it also guides the decoding process by suggesting word ordering and lexical choices in translation.

3.1 Markov Assumption

Given a sequence of words $W = w_1, w_2, ..., w_n$, the language model computes the joint probability of every word w_i in the sequence. By using the chain rule, we can factorize the probability of the sequence as:

$$P(w_1, w_2, ..., w_n) = P(w_1)P(w_2 \mid w_1)...P(w_n \mid w_1, w_2, ...w_{n-1}). \tag{3.1}$$

Each component in (3.1) is understood as the conditional probability of the current word, given a history of preceding words in the sequence. In order to accurately model these probabilities, a simplifying assumption (i.e. a *Markov assumption*) is made, under which we assume that the history is limited to a window of m preceding words. Thus, we assume that:

$$P(w_n \mid w_1, w_2, ..., w_{n-1}) \approx P(w_n \mid w_{n-m}, ..., w_{n-2}, w_{n-1}). \tag{3.2}$$

Such models that follow the Markov assumption are called n-gram language models. The smaller history window allows language models to adequately model unobserved texts and to model fluent sequences of phrases in a text.

3.2 Building n-gram Language Models

In their simplest form, n-gram language models can be computed according to maximum likelihood estimation. Given a large monolingual corpus, we can compute the probability of the next word in a sequence as the relative frequency:

$$P(w_n \mid w_{n-m}, ..., w_{n-2}, w_{n-1}) = \frac{\text{count}(w_{n-m}, ..., w_{n-1}, w_n)}{\sum_w \text{count}(w_{n-m}, ..., w_{n-1}, w)}, \tag{3.3}$$

In order to model the words that occur at the start and end of sentences, we frame each sentence with $n-1$ special start and end symbols <s> and </s>, respectively. This also ensures that the n-gram probabilities sum to unity.

3.2.1 Sparsity

The trade-off in deciding an suitable size for n is as follows: larger ns better model the original training data and yield more coherent sentences and thus improve fluency. A disadvantage is that larger corpora are necessary to calculate sufficient statistics on higher order n-grams. One of the properties of language models is that the vocabulary size V is known in advance.

Under the *closed vocabulary* assumption, a sentence outside of our training corpus only contains words that are in V. If a test sentence has an *out of vocabulary* (OOV) word, the language model does not know how to assign probability to the word and thus assigns zero probability of the word being observed. Based on the Markov assumption in (3.2), the entire sentence is subsequently assigned zero probability.

3.3 Smoothing

In addition to the sparsity problems mentioned in the previous section, n-gram probabilities in the LM are highly dependent on their relative frequencies in the training corpus and may not accurately represent real-world probabilities – particularly n-grams consisting of vocabulary words with low observed counts. Jurafsky and Martin (2008) outline several smoothing techniques exist to alleviate these problems, including Laplace smoothing and Good-Turing smoothing. Laplace smoothing simply adds one count to each word and includes an OOV word in the vocabulary. However, this kind of count-based smoothing does not preserve the original probability distribution of n-grams well. Instead, absolute discounting methods can be used to redistribute probability mass based on a combination of n-gram frequency discounting and back-off to lower order models.

3.3.1 Good-Turing Smoothing

Good-Turing smoothing (Good, 1953) models the expected future counts of n-grams, given the observations in the training data. Assuming that all occurrences of an n-grams are independent, Good-Turing posits that expected counts for n-grams that occur r times can be computed by a ratio of n-grams that are observed to occur $r+1$ times, versus those that are observed to occur only r times:

$$r^* = (r+1)\frac{N_{r+1}}{N_r}. \tag{3.4}$$

3.3.2 Interpolation

In order to address the sparsity of higher order n-gram language models, we can design language models that combine higher and lower n-gram counts. This essentially means that we choose to rely on lower order language models when we are not confident of the probability mass assigned to a particular n-gram – either because the observed counts are low, or the n-gram in question was not observed in training. One such method is *interpolation* (Jelinek and Mercer, 1980), in which we express an interpolated language model as a linear combination of n-gram language models of varying size. We can also define a recursive interpolation as:

$$P_n^I(w_i \mid h_{i,n}) = \lambda_{h_{i,n}} P_n^I(w_i \mid h_{i,n}) + (1 - \lambda_{h_{i,n}}) P_{n-1}^I(w_i \mid h_{i,n-1}), \tag{3.5}$$

where $h_{i,n} = w_{i-n+1}, ..., w_{i-1}$ is the n-gram history of word w_i. $\lambda_{h_{i,n}}$ sets how confident we are that we trust the n-gram language model, verses backing off to a lower order model. The λ values can be optimized with Expectation Maximization, with unique values for histories with different relative frequency counts (Jelinek and Mercer, 1980).

3.3.3 Back-off Models

Back-off modeling was introduced by Katz (1987) as an alternative to interpolation that relies on lower n-gram models only if the particular n-gram's history is not observed in training. Otherwise, a discounted n-gram probability $P^*(w_i \mid h_{i,n})$ is used, such as Good-Turing smoothing. The back-off model is defined as a system of equations:

$$P_n^{\text{BO}}(w_i \mid h_{i,n}) = \begin{cases} P^*(w_i \mid h_{i,n}) & \text{if } \text{count}_n(h_{i,n}) > 0 \\ z(h_{i,n})^{-1} P_{n-1}^{\text{BO}}(w_i \mid h_{i,n-1}) & \text{otherwise} \end{cases} \tag{3.6}$$

where $z(h_{i,n})$ normalizes the back-off probability.

3.3.4 Kneser-Ney Smoothing

Kneser-Ney smoothing was designed by Kneser and Ney (1995) to modify the role of lower order n-gram models under the observation that lower order models are only used in back-off models when the higher order model has little to no observed counts. Thus, lower order models should be adjusted to better serve a back-off role. Koehn (2010) illustrates this by describing the English word *york*. In the majority of cases, the bigram preceding word is *new*: referring to the city. While *york* appears many times in the training corpus, it is only observed with a small number of unique preceding history (referred to as the *diversity of histories*). Thus, the unigram *york* should be discounted to better reflect a bigram conditional probability. Formally, we can define the number of diverse histories for n-grams of arbitrary length as:

$$N_{1+}(\bullet, h_{n-1}, w) = \left|\{w_i : \text{count}(w_i, h_{n-1}, w) > 0\}\right|, \tag{3.7}$$

and the normalization factor as:

$$N_{1+}(\bullet, h_{n-1}, \bullet) = \sum_{w} N_{1+}(\bullet, h_{n-1}, w). \tag{3.8}$$

In layman's terms, (3.7) defines the number of distinct n-grams that can be generated by changing the first word in the sequence. The term in (3.8) normalizes (3.7) in the same way that probabilities of n-grams are calculated in (3.3) through maximum likelihood estimation. Thus, the discounted back-off probabilities are calculated as:

$$P^{\text{KN}}(w \mid h_{n-1}) = \frac{N_{1+}(\bullet, h_{n-1}, w)}{N_{1+}(\bullet, h_{n-1}, \bullet)}. \tag{3.9}$$

The back-off model for Kneser-Ney smoothing is now:

$$P^{\text{KN}}(w_i \mid h_{i,n}) = \begin{cases} \frac{\text{count}(h_{i,n}, w_i) - D}{\text{count}(h_{i,n})} & \text{if count}(h_{i,n}, w_i) > 0 \\ \frac{N_{1+}(\bullet, h_{i,n-1}, w)}{N_{1+}(\bullet, h_{i,n-1}, \bullet)} \beta_{h_{i,n}} & \text{otherwise} \end{cases} \tag{3.10}$$

where $\beta_{h_{i,n}}$ is the weight to normalize the back-off probability.

3.3.5 Modified Kneser-Ney Smoothing

Other smoothing techniques exist for language models constructed via back-off and interpolation. One smoothing technique that uses a combination of interpolation and back-off is *Modified Kneser-Ney smoothing* (Chen and Goodman, 1998). Modified Kneser-Ney smoothing uses absolute discounting to reduce the probability mass for observed word sequences. A fixed value D in the interval $[0, 1]$ is subtracted from the higher and lower n-gram models, modeled by Good-Turing estimates:

$$D(\text{count}(\cdot)) = \begin{cases} 1 - 2Y\frac{N_2}{N_1} & \text{count}(\cdot) = 1 \\ 2 - 3Y\frac{N_3}{N_2} & \text{count}(\cdot) = 2 \\ 3 - 4Y\frac{N_4}{N_3} & \text{count}(\cdot) \geq 3 \end{cases} \tag{3.11}$$

where N_c are the counts of n-grams with a relative frequency of c, and Y is defined as:

$$Y = \frac{N_1}{N_1 + 2N_2}.$$

The back-off function now becomes:

$$P^I(w_i \mid h_{i,n}) = \begin{cases} \alpha^I(w_i \mid h_{i,n}) & \text{if count}(h_{i,n}, w_i) > 0 \\ \gamma(h_{i,n}) P^I(w_i \mid h_{i,n-1}) & \text{otherwise} \end{cases} \tag{3.12}$$

and $\alpha^I(\cdot)$ is the interpolated function:

$$\alpha^I(w_i \mid h_{i,n}) = \alpha(w_i \mid h_{i,n}) + \gamma(h_{i,n}) P^I(w_i \mid h_{i,n-1}). \tag{3.13}$$

For higher order n-gram models, $\alpha(\cdot)$ is the count-based n-gram probability used in (3.10), while for lower order models, it is the discounted back-off probabilities defined in (3.9). $\gamma(\cdot)$ is the back-off probability weight.

Chen and Goodman (1998) showed in an extensive study that modified Kneser-Ney smoothing outperformed the other smoothing techniques listed above.

3.4 Evaluation

The most common metric for evaluating the quality of language models is *perplexity*, which measures how well a language model predicts a sequence of words in a test set. Perplexity refers to the average number of equally probable words the language model must choose from when predicting the next word in a sequence. Thus, a lower perplexity implies that the language model assigns higher probabilities to the test set. The perplexity measure is based on the principle of cross-entropy, which is defined as:

$$H(P_{\mathrm{LM}}) = -\frac{1}{n}\sum_{i=1}^{n} \log P_{\mathrm{LM}}(w_i \mid h_{i,n}) \tag{3.14}$$

for a sequence of length n. The perplexity is simply the exponential of the cross-entropy:

$$PP = 2^{H(P_{\mathrm{LM}})}. \tag{3.15}$$

3.5 Language Model Adaptation

As mentioned earlier, the purpose of language modeling is to reward fluent outputs in our translation task by weighting good phrase translations selected by the translation table with a high probability. It additionally confirms reordering choices suggested by the reordering model. In order to best model the fluency of our translation task, we would like the language model to reward fluent translation output based on the domain or genre of our translation task. Different styles of writing greatly affect the n-gram statistics of a language model. For example, news texts from the Wall Street Journal will select more formal word constructions than blogs. Email texts will contain more second person pronouns.

For example, our trained language model may assign high probability to the word sequence "kick the bucket", a slang phrase for death. However, if our translation task is about sports, we prefer to assign a higher probability to the sequence "kick the ball". The estimated probabilities in our language model may not adequately match our intended translation. Additionally, language is constantly changing. We intend to construct language models that will continue to be robust over time.

Language model adaptation seeks to adjust the n-gram probability distributions given a sample of adaptation text. In the following sections, we discuss various adaptation techniques.

3.5.1 Domain Adaptation vs. Topic Adaptation

Before discussing several adaptation techniques, we should disambiguate several adaptation tasks that vary on the nature of the adaptation text. When adaptation data represents the translation task domain one generally refers to *domain adaptation*, while when they just represent the content of the single document to be translated one typically refers to *topic adaptation*. Domain adaptation is useful when

we are confident that our translation task adheres to a specific genre. For example, if our translation task involves translating speeches from the European Parliament, we are confident that each text will have a similar structure and will cover topics related to governmental issues. If our task is to translate videos online, such as TED talks[1] or YouTube[2], we scale a language model's probability distribution based on each video clip.

3.5.2 MDI Adaptation

An n-gram language model approximates the probability of a sequence of words in a text $W_1^T = w_1, ..., w_T$ drawn from a vocabulary V by the Markov assumption described in (3.2):

$$P(W_1^T) = \prod_{i=1}^{T} P(w_i|h_i),$$

where $h_i = w_{i-n+1}, ..., w_{i-1}$ is the history of $n-1$ words preceding w_i. Given a training corpus B, we can compute the probability of a n-gram from a smoothed model via interpolation as:

$$P_B(w|h) = f_B^*(w|h) + \lambda_B(h) P_B(w|h'), \tag{3.16}$$

where $f_B^*(w|h)$ is the discounted frequency of sequence hw, h' is the lower order history, where $|h| - 1 = |h'|$, and $\lambda_B(h)$ is the zero-frequency probability of h, defined as:

$$\lambda_B(h) = 1.0 - \sum_{w \in V} f_B^*(w|h).$$

Federico (1999) has shown that MDI Adaptation is useful to adapt a background language model with a small adaptation text sample A, by assuming to have only sufficient statistics on unigrams. Thus, we can reliably estimate $\hat{P}_A(w)$ constraints on the marginal distribution of an adapted language model $P_A(h, w)$ which minimizes the Kullback-Leibler distance from B, i.e.:

$$P_A(\cdot) = \arg\min_{Q(\cdot)} \sum_{hw \in V^n} Q(h, w) \log \frac{Q(h, w)}{P_B(h, w)}. \tag{3.17}$$

The joint distribution in (3.17) can be computed using Generalized Iterative Scaling (Darroch and Ratcliff, 1972). Under the unigram constraints, the GIS algorithm reduces to the closed form:

$$P_A(h, w) = P_B(h, w)\alpha(w), \tag{3.18}$$

where

$$\alpha(w) = \frac{\hat{P}_A(w)}{P_B(w)}. \tag{3.19}$$

[1] http://www.ted.com/talks
[2] http://www.youtube.com

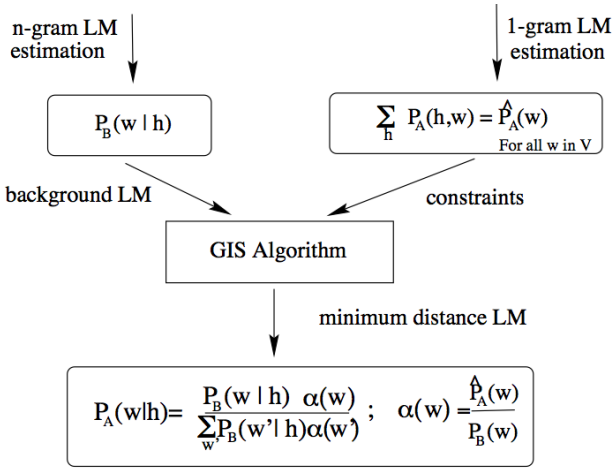


Figure 3.1: A graphical representation of MDI adaptation. A background language model is adapted from unigram statistics on an adaptation text, using Generative Iterative Scaling. The result is an adapted language model with a minimal distance from the background language model Federico (1999)

.

In order to estimate the conditional distribution of the adapted LM, we rewrite (3.18) and simplify the equation to:

$$P_A(w|h) = \frac{P_B(w|h)\alpha(w)}{\sum_{\hat{w}\in V} P_B(\hat{w}|h)\alpha(\hat{w})}. \tag{3.20}$$

Figure 3.1 provides a graphical representation of MDI adaptation.

The adaptation model can be improved by smoothing the scaling factor in (3.19) by an exponential term γ (Kneser et al., 1997):

$$\alpha(w) = \left(\frac{\hat{P}_A(w)}{P_B(w)}\right)^{\gamma}, \tag{3.21}$$

where $0 < \gamma \leq 1$. Empirically, γ values less than one decrease the effect of the adaptation ratio to reduce the bias.

As outlined in Federico (2002), the adapted language model can also be written in an interpolation form:

$$f_A^*(w|h) = \frac{f_B^*(w|h)\alpha(w)}{z(h)}, \tag{3.22}$$

$$\lambda_A(h) = \frac{\lambda_B(h)z(h')}{z(h)}, \tag{3.23}$$

$$z(h) = \left(\sum_{w:N_B(h,w)>0} f_B^*(w|h)\alpha(w)\right) + \lambda_B(h)z(h'), \tag{3.24}$$

which permits to efficiently compute the normalization term for high order n-grams recursively and by just summing over observed n-grams. The recursion ends with the following initial values for the empty history ε:

$$z(\varepsilon) = \sum_{w} P_B(w)\alpha(w), \tag{3.25}$$

$$P_A(w|\varepsilon) = P_B(w)\alpha(w)z(\varepsilon)^{-1}. \tag{3.26}$$

3.5.3 Other approaches

DeMori and Federico (1999) highlight several alternatives to MDI estimation for language model adaptation. We summarize these alternatives below.

MAP estimation Maximum a posteriori (MAP) estimation assumes that $P(w \mid h)$ belongs to a parametric family $P(w;\vec{\theta})$, whose likelihood function is the multinomial distribution $\vec{\theta}$. The a posteriori distribution combines the prior assumption of a multinomial distribution with empirical evidence provided by a text sample S. The objective is find the $\vec{\theta}$ that maximizes the posterior probability:

$$\vec{\theta}^{\text{MAP}} = \arg\max_{\vec{\theta}} P(S \mid \vec{\theta})P(\vec{\theta}). \tag{3.27}$$

This can be simplified via maximum likelihood estimation as recalculating that posterior probability distribution by considering the frequencies of the original training data with the adaptation sample S':

$$P_{MAP}(w \mid h) = \frac{n(hw) + n'(hw)}{n(h) + n'(h)}, \tag{3.28}$$

where $n(\cdot)$ and $n'(\cdot)$ are relative frequency estimates for S and S', respectively.

Linear interpolation Using the same terminology as Section 3.3.2, the recursive interpolation function follows the form of (3.5):

$$P(w \mid h) = \lambda(h)P_{MAP}(w \mid h) + (1 - \lambda(h))P(w \mid h'), \tag{3.29}$$

where $P(w \mid h)$ is defined as the piecewise function that incorporates MAP estimation:

$$P_{MAP}(w \mid h) = \begin{cases} \frac{n(hw)+n'(hw)}{n(h)+n'(h)} & \text{if } n(h) + n'(h) > 0 \\ 0 & \text{otherwise} \end{cases} \tag{3.30}$$

According to DeMori and Federico (1999), in order to reduce the number of parameters $\lambda(h)$, histories h are grouped into buckets $[h]$ based on n-gram frequency counts:

$$[h] = \begin{cases} 0 & \text{if } n(h) < k_1 \\ n(h) & \text{if } k_1 \leq n(h) \leq k_2 \\ k_2 + ord(h) & \text{if } k_2 \leq n(h) \end{cases} \tag{3.31}$$

where $ord(h)$ is a lexical ordering function and k_1 and k_2 are threshold parameters. DeMori and Federico (1999) state that thresholds are set to cluster based on low, intermediate, and high frequencies, respectively.

A *mixture model* can also be used for adaptation by estimating the interpolations weights separately via expectation maximization on the adaptation sample S' to the following model:

$$P(w \mid h) = \lambda_1([h])f(w \mid h) + \lambda_2([h])f'(w \mid h) + \lambda_3([h])P(w \mid h'). \tag{3.32}$$

Back-off Masataki et al. (1997) demonstrate that a MAP estimate can be integrated into a back-off language model in a similar way as interpolation; the *fill-up* model, for example, backs-off based on relative counts, according to the formula:

$$P_{BO}(w \mid h) = \begin{cases} f^*(w \mid h) & \text{if } n(h,w) > 0 \\ \alpha(h)f'^*(w \mid h) & \text{if } n(h,w) = 0 \text{ and } n(h) > 0 \text{ and } n'(h,w) > 0 \\ \beta(h)P_{BO}(w \mid h') & \text{otherwise} \end{cases}, \tag{3.33}$$

where $f^*(\cdot)$ is the discounting frequency, and $\alpha(h)$ and $\beta(h)$ are normalization constants.

Maximum Entropy Maximum entropy adaptation seeks to estimate $\vec{\theta}$ such that the entropy $H(\vec{\theta})$ is maximized.

$$H(\vec{\theta}) = -\sum_{w \in V} \vec{\theta}_w \log \vec{\theta}_w, \tag{3.34}$$

subject to constraints

$$\sum_{w \in V} \alpha_i(w)\theta_w = p_i(i = 1, ..., d). \tag{3.35}$$

Constraints are introduced for each n-grams observed on the training data. After training, adaptation can simply be performed on the adaptation sample by assigning values to each of the features.

3.6 Chapter Summary

In this chapter, we provided an introduction to n-gram based language modeling, which relies on the Markov assumption. We discussed various smoothing techniques to address the issue of sparsity in language models, including techniques to utilize lower order n-gram statistics via back-off and interpolation. We discussed Good-Turing discounting as a building block for widely utilized smoothing technique, such as Kneser-Ney smoothing and its modification that incorporates both back-off and interpolation. We additionally discussed perplexity as an evaluation metric.

In the second part of the chapter, we discussed language model adaptation, which improves the robustness of a language model with respect to a adaptation text. We

disambiguated domain adaptation, which focuses on adapting and entire translation task from topic adaptation, which adapts each individual document under the assumption that the topic or genre of the translation task varies by document. We finally discussed several popular adaptation techniques that involve interpolation, maximum a posteriori (MAP) estimation, and minimum discrimination information (MDI) estimation. We will continue to use MDI adaptation in our experimental framework.

Chapter 4

Topic Adaptation

In Chapter 3 we discussed language modeling and topic adaptation. We now discuss one method of generating statistics on an adaptation text through *topic modeling*.

4.1 Topic Modeling

In topic modeling, we seek to provide a compact representation of documents in our collection. Topic models allow a compact representation of documents as a set of features that correspond to topics. Each document can be represented as a mixture of topics, if we assume that words within each document are *exchangeable*. By exchangeable, we refer to the "bag-of-words" assumption that the position of each word in the document is irrelevant. Under this assumption, we can model documents by the extent to which each topic describes the document.

4.2 Latent Semantic Analysis

The original idea of LSA is to map documents to a *latent semantic space*, which reduces the dimensionality by means of singular value decomposition (Deerwester et al., 1990). A word-document matrix A is decomposed by the formula:

$$A = U\Sigma V^t, \tag{4.1}$$

where U and V are orthogonal matrices with unit-length columns and Σ is a diagonal matrix containing the singular values of A. LSA approximates Σ by casting all but the largest k singular values in Σ to zero.

Deerwester et al. (1990) empirically showed that the co-occurrence of terms within text documents was sufficient to recover the latent topic structure in an unsupervised manner.

4.3 Probabilistic Latent Semantic Analysis

Probabilistic Latent Semantic Analysis (PLSA) is an extension of LSA constructed by Hofmann (1999). Similar to singular value decomposition, we assume that each document can be "described" as a mixture of latent class variables. In the context of

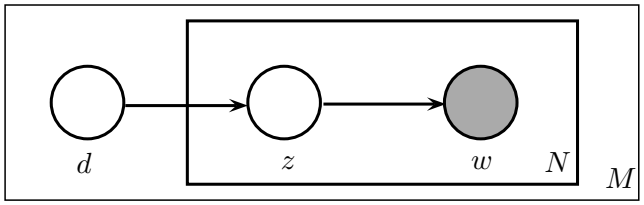


Figure 4.1: A graphical model representation of PLSA.

topic modeling, the latent class variables $z \in Z = \{z_1, ..., z_k\}$ correspond to topics, from which we can derive probabilistic distributions of the words in a document.

Figure 4.1 provides a graphical model representation of PLSA. PLSA assumes that a both collection of documents and the words in each document are known, but the topics that generate each word are unknown. PLSA defines a generative story as follows: A document $d \in D = \{d_1, ..., d_M\}$ is selected from a collection of M documents, with a probability $P(d)$. PLSA assumes that each word w in d is generated from some latent topic z. Assuming that the length of the document is known a priori, we pick a topic z for each expected word with probability $P(z \mid d)$. Finally, each word w is generated from our vocabulary V with probability $P(w \mid z)$.

Since the assignments between topics and words cannot be observed directly, we marginalize over all possible topic assignments for each word to assess the total probability of a word in a document. Thus, using Bayes' formula, the conditional probability of each word in a document is:

$$P(w \mid d) = \sum_{z} P(w \mid z)P(z \mid d). \tag{4.2}$$

In training the model, the goal is to learn $P(z \mid d)$ and $P(w|z)$ by maximizing the log-likelihood function:

$$L(W, D) = \sum_{d \in D} \sum_{w \in W} n(w, d) \log P(w \mid d), \tag{4.3}$$

where $n(w, d)$ is the term frequency of w in d.

Using the Expectation Maximization algorithm, we estimate the parameters $P(z|d)$ and $P(w|z)$ via an iterative process that alternates two steps: (i) an expectation step (E) in which posterior probabilities are computed for each latent topic z; and (ii) a maximization (M) step, in which the parameters are updated for the posterior probabilities computed in the previous E-step. These steps can be consolidated into two simple parameter updates:

$$P(w \mid z) \propto \sum_{d \in D} n(w, d)P'(w \mid z)P'(z \mid d) \tag{4.4}$$

$$P(z \mid d) \propto \sum_{d \in D, w \in V} n(w, d)P'(w \mid z)P'(z \mid d) \tag{4.5}$$

Iterating the E- and M-steps will lead to a convergence that approximates the maximum likelihood equation in (4.3). Details of how to efficiently implement the re-estimation formulas can be found in Federico (2002).

A document-topic distribution $\hat{P}(z \mid d')$ can be inferred on a new document d' by maximizing the following equation:

$$\hat{P}(z \mid d') = \arg\max_{P(z|d')} \sum_{w} n(w, d') \log \sum_{z} P(w \mid z) P(z \mid d'). \tag{4.6}$$

This can be maximized via Expectation Maximization on document d' by keeping fixed the word-topic distributions already estimated on the training data. Consequently, a word-document distribution can be inferred by applying the mixture model (4.2) (see Federico, 2002).

4.4 Latent Dirichlet Allocation

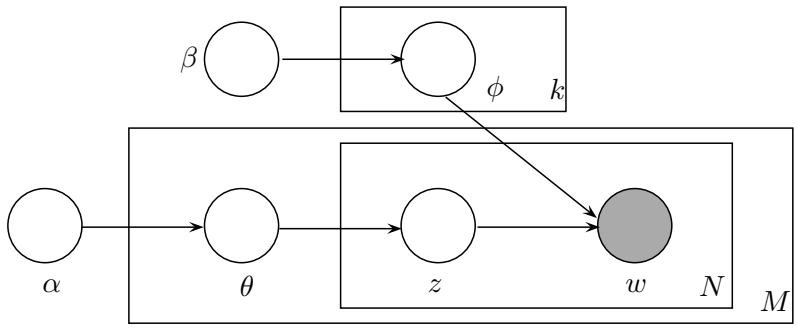


Figure 4.2: A graphical model representation of LDA.

Similar to PLSA, *Latent Dirichlet Allocation* (LDA) is a generative model constructed by Blei et al. (2003) which assumes that documents are represented as mixtures over latent topics; however, LDA makes a prior assumption that topics z_i are drawn from a multinomial distribution. The conjugate prior of a multinomial distribution $\vec{\theta}$ is the Dirichlet distribution, defined as:

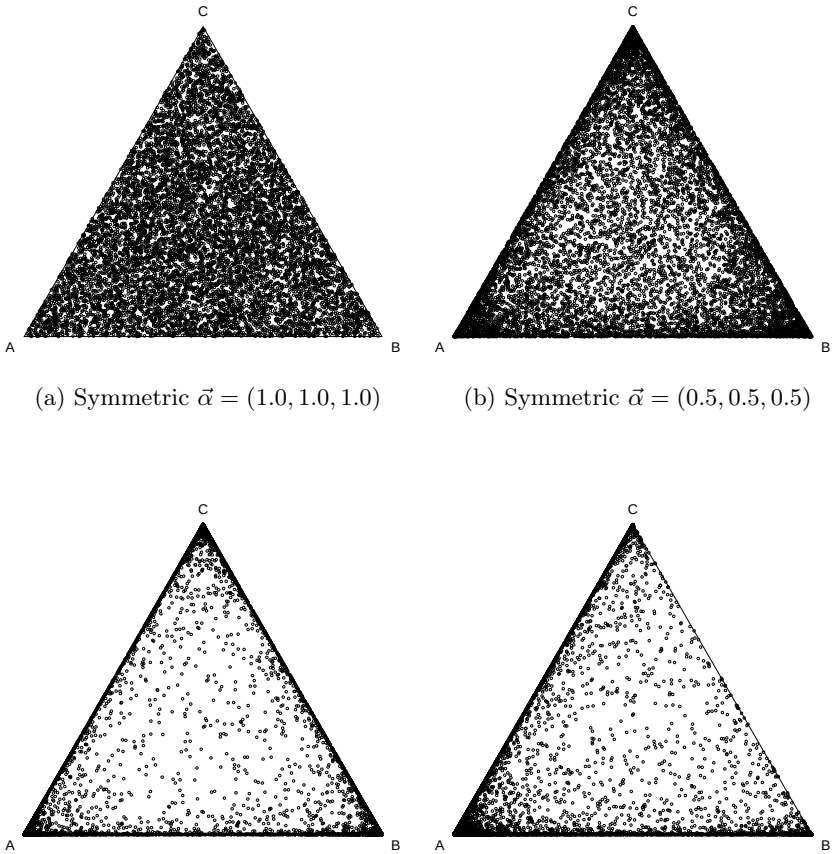
$$\mathrm{Dir}(\vec{\theta} \mid \vec{\alpha}) = \frac{\Gamma(\sum_{i=1}^{k} \alpha_i)}{\prod_{i=1}^{k} \Gamma(\alpha_i)} \prod_{i=1}^{k} \theta_i^{\alpha_i - 1}, \tag{4.7}$$

where $\Gamma(x) = (x-1)!$ and $\vec{\alpha}$ are the parameters of the model that introduce a priori knowledge about which topics are likely (one α for each latent topic).

Each document is usually generated from only a few topics; thus we need to define a Dirichlet distribution that favors posterior distributions which force the majority of the probability mass toward a small number of topics. Figure 4.3 shows examples of probability distributions drawn from four Dirichlet distributions, configured with different $\vec{\alpha}$ parameters. Individual α parameters within the interval $(0, 1)$ give the

desired effect. In the case of topic modeling, we can assume symmetric α parameters if there is no reason to believe that one latent topic is more likely than another.

Armed with the understanding of our Dirichlet prior, we can define the LDA's generative model shown in Figure 4.2 as follows: A document-topic distribution $\vec{\theta}$ is drawn from a Dirichlet prior with parameters $\vec{\alpha} = (\alpha_1, \alpha_2, ..., \alpha_k)$. For each word w_i in d, a latent topic z_i is sampled from Mult($\vec{\theta}$). Finally, word w_i is sampled from a multinomial based on the topic-word distribution $\vec{\phi}$, which is also drawn from a



(a) Symmetric $\vec{\alpha} = (1.0, 1.0, 1.0)$ (b) Symmetric $\vec{\alpha} = (0.5, 0.5, 0.5)$

(c) Symmetric $\vec{\alpha} = (0.1, 0.1, 0.1)$ (d) Asymmetric $\vec{\alpha} = (0.5, 0.1, 0.1)$

Figure 4.3: 10,000 draws from four Dirichlet distributions of three topics. The triangle is the 2-D simplex representing all possible probability distributions over the three topics: (a) shows a uniform prior, in which all probability distributions are equally likely for a symmetric $\alpha = 1$; (b) and (c) show increasing probability distributions favoring one or two topics as the symmetric α approaches zero; (d) shows that asymmetric α parameters favor topics with higher α values.

Dirichlet prior with a single symmetric parameter β, conditioned on the vocabulary size.

The inference problem requires computing the posterior distribution of the hidden variables, given a document:

$$P(\vec{z}, \vec{\theta}, \vec{\phi} \mid \vec{w}, \vec{\alpha}, \beta) = \frac{P(\vec{w}, \vec{z}, \vec{\theta}, \vec{\phi} \mid \vec{\alpha}, \beta)}{P(\vec{w} \mid \vec{\alpha}, \beta)}. \tag{4.8}$$

The model is intractable for exact inference; however, several approximation inference algorithms exist, including variational EM (Blei et al., 2003), Markov chain Monte Carlo methods, and maximum a posteriori (MAP) estimation.

Griffiths and Steyvers (2004) discuss a Markov chain Monte Carlo approximation for (4.8) using Gibbs Sampling. The Gibbs sampler constructs a Markov chain that uses a full conditional $P(z_i = j \mid \vec{z}_{-i}, \vec{w})$, which simulates $P(\vec{z} \mid \vec{w})$ by computing $\vec{\theta}$ and $\vec{\gamma}$ via maximum likelihood estimation, leaving out a single topic assignment for word w_i in document d. The full conditional is defined as follows:

$$P(z_i = j \mid \vec{z}_{-i}, \vec{w}) = \frac{n_{-i,j}^{(w_i)} + \beta}{\sum_{i'=1}^{V} \left(n_{-i',j}^{(w_i')} + \beta \right)} \cdot \frac{n_{-i,j}^{(d)} + \alpha_k}{\sum_{j'=1}^{k} \left(n_{-i,j'}^{(d)} + \alpha_k \right)}, \tag{4.9}$$

where $n_j^{(w_i)}$ is the number of times word w_i has been assigned to topic j, and $n_j^{(d)}$ is the number of times a word from the document d has been assigned to topic j. The $-i$ implies that the current assignment of z_i is excluded from the counts.

Individual topic assignments are left out in a round-robin fashion, for each word in each document and are resampled from (4.9). The sampling process is repeated until convergence and a minimum number of sampling iterations are carried out. To obtain the resulting model parameters, we can average over the full set of samples. The parameter estimates for any single sample are:

$$\hat{\phi}_j^{(w)} = \frac{n_j^{(w)} + \beta}{\sum_{w' \in V} \left(n_j^{(w')} + \beta \right)}, \tag{4.10}$$

$$\hat{\theta}_j^{(d)} = \frac{n_j^{(d)} + \alpha_j}{\sum_{j'=1}^{k} \left(n_{j'}^{(d)} + \alpha_j \right)}, \tag{4.11}$$

which can be understood as taking the expectation of the Dirichlet distribution.

Heinrich (2008) provides a detailed derivation of Gibbs Sampling for LDA.

4.5 PLSA vs. LDA

Blei et al. (2003) claim that the generative semantics of PLSA lead to problems in assigning probability to previously unobserved documents, since the generative model is based upon an indexing of documents observed in training. However,

Girolami and Kabán (2003) show that PLSA is equivalent to a MAP estimation of an LDA model with a uniform prior Dir(1) (see Figure 4.3a for an example of this distribution).

As mentioned earlier, PLSA assumes that each word of a training document is generated from a randomly chosen topic, which is drawn from a document-specific distribution over topics; thus, each document has one distribution over topics. Blei et al. (2003) contrast this generative approach with that of LDA, where each word from both observed and unseen documents is generated by drawing a topic from a distribution with a randomly chosen parameter. PLSA requires $kV + kM$ parameters; thus the model parameters grow linearly with the number of training documents, implying that PLSA is dependent on the training corpus to compute reliable statistics on the distribution of topics over documents and words over topics. LDA, on the other hand, only requires $k + kV$ parameters. By introducing a Dirichlet prior, LDA encodes a priori information about the distribution of topics within documents. Girolami and Kabán (2003) state that the Dirichlet prior avoids the problem of overfitting by adding prior pseudocounts to observed probabilities.

Unfortunately, a problem of assigning proper $\vec{\alpha}$ and β hyperparameters to the Dirichlet priors exists in LDA. How do we know if our Dirichlet prior accurately models the distribution of topics, since the topics themselves are latent variables? In Griffiths and Steyvers (2004)'s experiments, $\beta = 0.1$ and a symmetric $\alpha = 50/k$ is used to keep the sum of Dirichlet hyperparameters fixed for varying topic counts k; however, the hyperparameter configuration is dependent on the data experiment.

In evaluation experiments involving clustering Japanese news articles, a Japanese Q&A website, and Korean web articles, Masada et al. (2008) measured no improvement of LDA over PLSA as a dimensionality reduction method, noting that the standard deviation markers for PLSA intersect with those of LDA.

4.6 Topic Modeling via MDI Estimation

Given the theoretical motivations for language model adaptation and topic models, we can now perform topic adaptation on an adaptation text. A PLSA or LDA model can be trained on either the same texts from our background language model, or from another collection of texts. Then, our adaptation text is fed to the topic model and unigram probabilities are inferred. Given these unigram statistics, we can estimate an adapted language model by performing MDI estimation by computing smoothed frequencies and zero probability weights via (3.22) and (3.23).

4.7 Chapter Summary

In this chapter, we provided an overview on topic modeling. We first discussed latent semantic analysis as a means to reduce the dimensionality of a collection of documents from a bag of words to a compact representation using a mixture of latent

topics. We discussed probabilistic latent semantic analysis (PLSA), which provides a generative model that factorizes the probability of words given a document by assuming that each document has its own distribution of topics. Both training and inference on unseen documents can be performed using expectation maximization.

We then discussed Latent Dirichlet Allocation (LDA), another generative topic model that assumes that topics are drawn from a Multinomial distribution with a Dirichlet prior. We showed that the calculation of the posterior is intractable and discussed an approximation approach using Gibbs sampling. Other approximations include variational EM and maximum a posteriori estimation.

We then discussed the similarities and differences between LDA and PLSA and pointed out that PLSA can be seen as a special instance of LDA with a uniform Dirichlet prior that makes no prior assumptions regarding the distribution of topics and words.

Finally, we tied topic modeling with language model adaptation by briefly outlining how MDI adaptation can be used in conjunction with PLSA or LDA.

Chapter 5

Bilingual Topic Modeling

In Chapter 4 we discussed topic modeling and its application in language model adaptation. In this chapter, we extend the topic modeling framework to the bilingual case, which better fits the task of translating texts.

5.1 Related Work

Most works focus on monolingual language model adaptation in the context of automatic speech recognition. In this section, we discuss several approaches to language model adaptation that target the task of statistical machine translation.

5.1.1 Structured Query Models

Zhao et al. (2004) construct a baseline SMT system using a large background language model to generate N-best initial translations of an adaptation text. The initial translations are used as structured queries to retrieve similar sentences in a text collection. The *structured query model* is a collection of the subsets of translated n-grams for each source word, defined as follows:

$$Q_{st} = \{\vec{t}_{s_1}, \vec{t}_{s_2}, ..., \vec{t}s_l\}, \tag{5.1}$$

where s_i is the ith word in source text of length l: $s_1, s_2, ..., s_l$; $\vec{t}_{s_i}$ is a set of target n-grams for s_i:

$$\vec{t}_{s_i} = \{\{t_i, ...\}_1, \{t_i, t_{i+1}, ...\}_2, \{t_i, t_{i+1}, t_{i+2}, ...\}_3, ..., \{t_i, t_{i+1}, t_{i+n-1}\}_n\}. \tag{5.2}$$

Each n-gram collection in $\vec{t}_{s_i}$ is treated as bag-of-word features used to search for relevant texts in the text collection by incorporating four features that incorporate weighted and unweighted sums of word matches, as well as ordered and unordered word matches within N-word windows. The top M sentences retrieved from the query are combined with the background language model via linear interpolation to form the adapted language model.

5.1.2 Mixture model approaches

Researchers such as Foster and Kuhn (2007) and Koehn and Schroeder (2007) have investigated mixture model approaches to adaptation. Foster and Kuhn (2007) use

a mixture model approach that involves splitting a training corpus into different components, training separate models on each component, and applying mixture weights as a function of the distances of each component to the source text. Koehn and Schroeder (2007) learn mixture weights for language models trained with in-domain and out-of-domain data respectively by minimizing the perplexity of a tuning (development) set and interpolating the models. Although the application of mixture models yields significant results, the number of mixture weights to learn grows linearly with the number of independent language models applied.

5.1.3 Hidden Markov Bilingual Topic AdMixture

Zhao and Xing (2008) propose a Hidden Markov Bilingual Topic AdMixture (HM-BiTAM) model, using the BiTAM formalism originally described in Zhao and Xing (2006). BiTAM is an alternative means for word alignment: a document pair $(\mathbf{F}, \mathbf{E})$ is treated as an admixture of topics. A sentence pair is assumed to be word-aligned based on a bilingual model that aligns words based on latent topical assignments. In BiTAM-1, topics are sampled at the sentence level and each topic z has a corresponding translation table that defines a translation lexicon:

$$B_{i,j,k} = P(f = f_j \mid e = e_i, z = k). \tag{5.3}$$

Using a construction similar to LDA, Zhao and Xing (2006) define a generative model as follows: a sentence number N is drawn from a Poisson distribution and a topic-weight vector θ_d from a Dirichlet distribution with a symmetric α parameter governing the distributions of the K topics. For each sentence pair $(\vec{f}_n, \vec{e}_n)$ in the dth document pair, a document length is sampled and a topic z_{dn} is sampled from Mult(θ_d). Each target word is sampled from a monolingual model $P(e_i)$ and the alignment links a_j are sampled from either a uniform model $P(a_j)$ or a Hidden Markov Model (HMM). Finally, each foreign word is sampled from the translation lexicon as $P(f_j \mid \vec{e}_n, a_j, z_n, \mathbf{B})$. The joint conditional posterior distribution after marginalizing over $\vec{\theta}$ and $\vec{z}$ can be written as:

$$p(\mathbf{F}, \mathbf{A} \mid \mathbf{E}, \alpha, \mathbf{B}) \quad = \int P(\vec{\theta} \mid \alpha) \left(\prod_{n=1}^{N} P(\vec{f}_n, \vec{a}_n \mid \vec{e}_n, B_{z_n}) \right) d\vec{\theta}. \tag{5.4}$$

BiTAM-1 is extended to a word-level admixture model (BiTAM-3) by sampling topic indicator $z_{n,j}$ for each word-pair (f_j, e_{a_j}) in the nth sentence-pair.

In the HM-BiTAM construction, foreign words are sampled via alignment links from a first-order Markov process and a topic specific translation lexicon. While HM-BiTAM has been used for bilingual topic extraction and topic-specific lexicon mapping in the context of SMT, Zhao and Xing (2008) note that HM-BiTAM can generate unigram language models for both the source and target language and thus can be used for language model adaptation through MDI in a similar manner as outlined in Federico (2002).

5.1.4 Bilingual LSA

Tam and Schultz (2007) construct a Latent Dirichlet-Tree Allocation (LDTA) model to relax the independence assumption between topics via a hierarchical structure; the original Dirichlet prior is replaced with a Dirichlet-Tree prior that models topic correlations in a tree structure. Each internal tree node represents a Dirichlet distribution over the branches of the node to its child nodes. Figure 5.1 illustrate the LDTA model with an example of a Dirichlet-Tree with a depth of two.

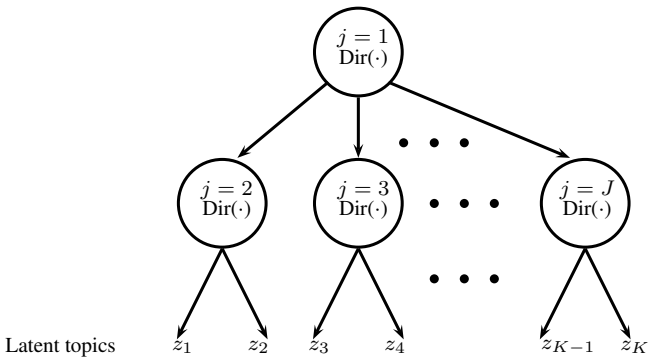


Figure 5.1: Dirichlet-tree prior of depth 2 (Tam and Schultz, 2007).

Given a set of tree nodes $1..J$, the LDTA model is generated as follows: a vector of node branch probabilities $\vec{b}_j$ is drawn from a Dirichlet distribution for each node j with prior α_{jc} parameters for each outgoing branch c. Topic proportions are generated from a multinomial distribution defined as the product of branch probabilities that lead to a leaf node; in other words, the probability mass for each topic k in $\vec{\theta}$ is defined as:

$$\theta_k = \prod_{jc} b_{jc}^{\delta_{jc}(k)}, \tag{5.5}$$

where $\delta_{jc}(k)$ indicates when the cth branch of node j leads to a leaf node of topic k in the tree. In a similar fashion to LDA, topics and words are respectively drawn from Mult($\vec{\theta}$) and Mult($\vec{\phi}$), where $\vec{\phi}$ is drawn from a Dirichlet prior with a symmetric β parameter for each word.

Using the LDTA formalism, Tam et al. (2007) propose a bilingual LSA approach to topic modeling, which consists of two hierarchical LDTA models constructed from parallel document corpora. A one-to-one correspondence between LDA models is enforced by learning the hyperparameters of the variational Dirichlet posteriors in one LDA model and bootstrapping the second model by fixing the resulting multinomial distribution $\vec{\theta}$ both for training and inference on target sentences. Figure 5.2 provides a pictorial representation of bilingual LSA. This technique is based on the assumption that the topic distributions of the source and target documents are identical. Tam et al. (2007) also use MDI adaptation from the unigram probabilities inferred by the target LSA model.

The bilingual LSA model is extended in Tam and Schultz (2009) by constructing

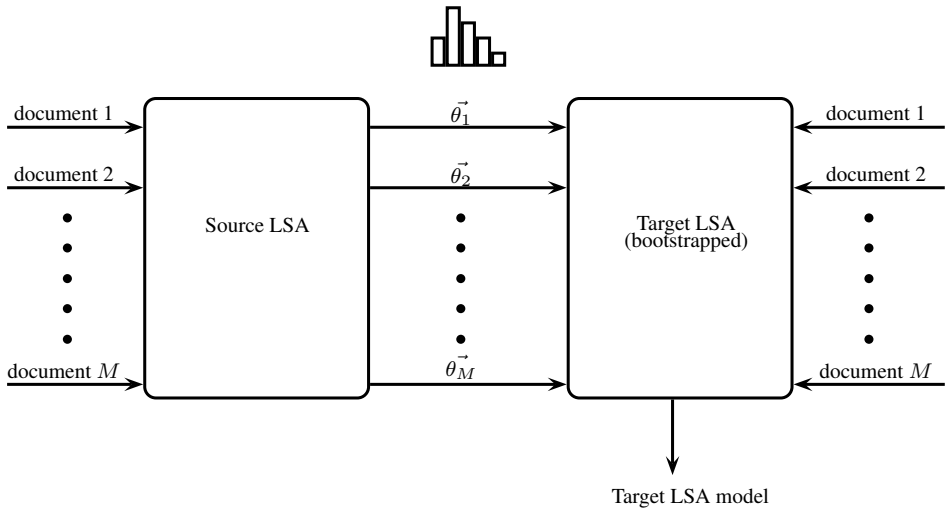


Figure 5.2: Bilingual LSA bootstrapping by sharing posterior topic distributions between parallel documents (Tam et al., 2007).

parallel document clusters formed by monolingual documents using M parallel seed documents as cluster centroids. Monolingual documents are represented by their $\vec{\theta}$ distribution, which is inferred from a monolingual LDTA model. Documents are assigned to the cluster whose centroid has the minimum distance, measured by a dot product between $\vec{\theta}$ vectors. Clusters are trained with a bilingual LSA model, in which a one-to-one topic correspondence between parallel clusters is assumed.

5.2 Our Approach

While the LDA-based bilingual topic modeling approaches mentioned in Section 5.1.4 are powerful, we believe that they introduce complexities that are unnecessary in the translation scenario. Tam et al. (2007) train two separate LDTA models and combines them by bootstrapping the second model from the topic-document posteriors per document form the first model, effectively inducing a one-to-one correspondence between topics in the source and target language. This approach, however, assumes that the topic model itself is incapable of inferring the one-to-one correspondence itself. Since the source and target documents are representing the same semantic concepts, it seems reasonable that a single topic model modeled on bilingual documents would have virtually the same topic distribution as its monolingual counterparts. Thus, we present a simplified topic modeling approach that inherently handles bilingual documents, as described in Ruiz and Federico (2011).

For the moment, we will assume that we are training a bilingual PLSA model; however, the same model can be constructed with the use of LDA-based topic models without loss of generality.

Similar to the treatment of documents in HM-BiTAM (Zhao and Xing, 2008),

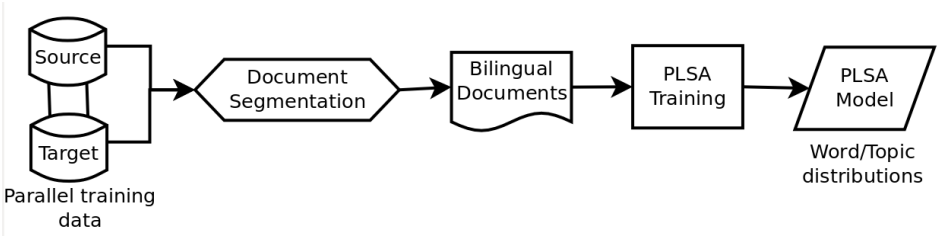


Figure 5.3: Training for a Bilingual PLSA model. Parallel texts are merged into bilingual documents with a joint vocabulary. The bilingual documents are trained via PLSA to infer word and topic distributions.

we combine parallel texts into a bilingual document-pair $(\mathbf{F}, \mathbf{E})$ containing n parallel sentence pairs (f_i, e_i) with $1 \leq i \leq n$, corresponding to the source and target languages, respectively. Based on the assumption that the topics in a parallel text share the same semantic meanings across languages, the topics are sampled from the same topic-document distribution. We make the additional assumption that stop-words and punctuation, although having high word frequencies in documents, will generally have a uniform topic distribution across documents; therefore, it is not necessary to remove them prior to model training, as they will not adversely affect the overall topic distribution in each document. In order to ensure the uniqueness between word tokens between vocabularies $V_{\mathbf{F}}$ and $V_{\mathbf{E}}$, we annotate all words within $\mathbf{F}$ with special characters. Our bilingual documents can now be considered as monolingual documents of a special language with vocabulary $V_{\mathbf{FE}}$, where

$$V_{\mathbf{FE}} = V_{\mathbf{F}} \cup V_{\mathbf{E}}. \tag{5.6}$$

Since we have essentially treated the bilingual topic modeling problem as a special case of monolingual topic modeling, we can perform PLSA training in the same manner as described in Section 4.3 in order to receive word-topic distributions $P(w|z), \forall w \in V_{\mathbf{FE}}$. Figure 5.3 provides a pictorial representation of bilingual PLSA-based training.

Inference is slightly different from the standard PLSA approach. Given an untranslated text $\hat{\mathbf{F}}$, our goal is to infer the word document distribution $P(w \mid d)$ of the translated text $\hat{\mathbf{E}}$; however, $\hat{\mathbf{E}}$ is unknown. If we again consider our bilingual topic modeling approach as a special case of a monolingual topic model, we can again construct a "bilingual" document by merging the known source and target texts. In our scenario, there is no target document; however, we can convert the source document into a special case of a monolingual document with vocabulary $V_{\mathbf{FE}}$ that doesn't contain any words in the foreign vocabulary. This is motivated by our definition in (5.6), since $V_{\mathbf{F}} \subset V_{\mathbf{FE}}$.

Thus, we split $\hat{\mathbf{F}}$ into a sequence of documents D. For each document $d_i \in D$, we infer a full word-document distribution by learning $\hat{\theta}$ via (4.6). Via (4.2), we can generate the full word-document distribution $P(w \mid d)$ for $w \in V_{\mathbf{FE}}$.

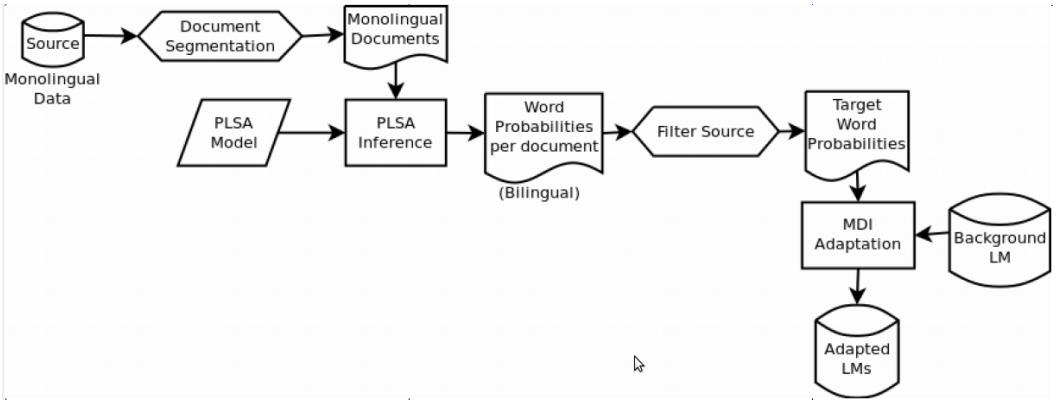


Figure 5.4: Inference and language model adaptation for a Bilingual PLSA model.

We then convert the word-document probabilities into pseudo-counts for a unigram language model via a scaling function:

$$n(w \mid d) = \frac{P(w \mid d)}{\max_{w'} P(w' \mid d)} \cdot \Delta, \tag{5.7}$$

where Δ is a scaling factor to raise the probability ratios for each word in the vocabulary above 1. Since our goal is to generate a unigram language model on the target language for adaptation (i.e. a unigram language model for words $w' \in V_{\mathbf{E}}$), we remove the source words generated in (5.7) prior to building the language model and rescale the probabilities to sum to unity.

From our newly generated unigram language model, we can now perform MDI adaptation on the background language model to yield an adapted language model for translating the source document used for the PLSA inference step. These steps are outlined graphically in Figure 5.4.

5.3 Experiments: IWSLT 2010

5.3.1 Evaluation Task

Our experiments were done using the TED Talks collection, used in the IWSLT 2010 evaluation task[1]. TED LLC is a nonprofit organization that regularly organizes two annual conferences in the US and the UK, where "prominent experts from different fields are invited to give short talks about topics relevant to global society" (Paul et al., 2010).

In IWSLT 2010, the challenge was to translate talks from the TED website[2] from English to French. The talks include a variety of topics, including photography and pyschology and thus do not adhere to a single genre. All talks were given in English

[1] http://iwslt2010.fbk.eu/
[2] http://www.ted.com/talks/

TALK	Data	Lang	Sentences	Avg. Length	Tokens	Types
train	(text)	E	86,225	9.8	842,145	31,429
	(text)	F	86,225	10.0	867,963	42,599
dev	(speech)	E	1,368	9.5	12,962	2,687
	(text)	F	1,368	9.3	12,712	3,246
test	(speech)	E	3,584	9.0	32,155	4,153
	(text)	F	3,584	9.2	33,010	5,571

Table 5.1: TED TALK English to French corpus from IWSLT 2010.

and were manually transcribed and translated into French. The TED training data consists of 329 parallel talk transcripts with approximately 86k sentences. The TED test data consists of transcriptions created via 1-best ASR outputs from the KIT Quaero Evaluation System with a language model updated with the TED training data. The TED talk data is segmented at the clause level, rather than at the level of sentences. An overview of the statistics of the TED talk data is listed in Table 5.1.

5.3.2 Experimental Settings

Baseline We construct a phrase-based machine translation system, built upon the Moses open-source SMT toolkit Koehn et al. (2007)[3]. The translation and lexicalized reordering models have been trained on the TED Talk parallel data from the training set. One 5-gram background language model is constructed from the French side of the TED training data (approximately 740k words after lowercasing and tokenization). The language model is smoothed with the modified Kneser-Ney technique (Chen and Gopalakrishnan, 1998) (see Section 3.3.5 for a detailed explanation) and computed with the IRSTLM toolkit (Federico et al., 2008)[4].

The weights of the SMT log-linear interpolation model were optimized via minimum error rate training (MERT) (Och, 2003) (see Section 2.7 for details), using 200 best translations at each tuning iteration.

Bilingual PLSA topic modeling Using the topic modeling technique outlined in Section 5.2 (note that we are translating from English to French), we construct bilingual documents by splitting the parallel TED training corpus into 41,847 documents of 5 lines each. While each individual TED lecture could be used as a document, our experimental goal is to simulate near-time translation of speeches; thus, we prefer to construct small documents to simulate topic modeling on a spo-

[3]http://www.statmt.org/moses/
[4]http://sourceforge.net/projects/irstlm/

robert lang is a pioneer of the newest kind of origami – using math and engineering principles to fold mind-blowingly intricate designs that are beautiful and , sometimes , very useful . my talk is " flapping birds and space telescopes . " and you would think that should have nothing to do with one another , but i hope by the end of these 18 minutes , you 'll see a little bit of a relation . robert lang est un pionnier des nouvelles techniques d' origami - basées sur des principes mathématiques et d' ingénierie permettant de créer des modèles complexes et époustouflants , qui sont beaux et parfois , très utiles . ma conférence s' intitule " oiseaux en papier et télescopes spatiaux " . et vous pensez probablement que les uns et les autres n' ont rien en commun , mais j' espère qu' à l' issue de ces 18 minutes , vous comprendrez ce qui les relie .

Figure 5.5: A sample (tokenized) bilingual document used for PLSA training.

ken language scenario in which the length of a talk is not known a priori. We annotate the English source text for removal after inference. Figure 5.5 contains a sample document constructed for PLSA training. (In fact, we distinguish English words from French words by attaching to the former a special suffix.) By using our in-house implementation, training of a PLSA model with 250 latent topics on the bilingual collection converged after 20 EM iterations.

Using our PLSA model, we run inference on each of the 476 test documents from the TED lectures, constructed by splitting the test set into 5-line documents. Since our goal is to translate and evaluate the test set, we construct monolingual (English) documents. Figure 5.6 provides an example of a document to be inferred. We collect the bilingual unigram pseudocounts after 10 iterations of inference and remove the English words. The TED lecture data is transcribed by clauses, rather than full sentences, so we do not add sentence splitting tags before training our unigram language models.

As a result of PLSA inference, the probabilities of target words increase with respect to the background language model. Table 5.2 demonstrates this phenomenon by outlining several of the top ranked words that have similar semantic meaning to non-stop words on the source side. In every case, the probability $P_A(w)$ increases fairly substantially with respect to the $P_B(w)$. As a result, we expect that the adapted language model will favor both fluent and semantically correct translations as the adaptation is suggesting better lexical choices of words.

MDI Adaptation We perform MDI adaptation with each of the unigram language models to update the background TED language model. We configure the adaptation rate parameter γ to 0.3, as recommended in Federico (2002). The base-

> we didn 't have money , so we had a cheap , little ad , but we wanted college students for a study of prison life . 75 people volunteered , took personality tests . we did interviews . picked two dozen : the most normal , the most healthy .

Figure 5.6: A sample English-only document (#230) used for PLSA inference. A full unigram word distribution will be inferred for both English and French.

line LM is replaced with each adapted LM, corresponding to the document to be translated. We then calculate the mean perplexity of the adapted LMs and the baseline, respectively.

Bilingual LDA topic modeling Our main goal is to simplify bilingual topic modeling as much as possible while yielding good language models and translation outputs; as a result, we compare the results of our bilingual PLSA adaptation experiment with a bilingual LDA experiment that uses the same document modeling and adaptation approaches mentioned above. We want to determine if assigning a Dirichlet prior over the topic and word distributions would make a large impact in the quality of the background language model and a subsequent improvement in translation, in comparison with a PLSA model that makes no prior assumptions about the distribution of topics. In our LDA experiment, we construct an LDA model, again with 250 latent topics, a symmetric α of 0.025, and a symmetric β of 0.01.

Using the GibbsLDA++ toolkit (Phan and Nguyen, 2007–2008), we perform LDA training via Gibbs Sampling (see Section 4.4 for details), using 100 sampling iterations. After running LDA inference on the test data, we remove the English words and perform MDI adaptation.

5.3.3 Evaluation Metrics

In our evaluation, we use three metrics. We use perplexity (see Section 3.4 for details) to assess the quality of the adapted language models. We extend the evaluation to assess the quality of machine translation outputs, via BLEU and NIST scores (see Sections 2.6.1 and 2.6.2).

5.3.4 Results

5.3.4.1 Bilingual PLSA Results

We perform MT experiments on the IWSLT 2010 evaluation set to compare the baseline and adapted LMs. Table 5.3 shows the evaluation results of the Bilingual PLSA model. We observe a 15.3% relative improvement in perplexity score over the baseline, indicating that the MDI-based topic adaptation is correctly using mixtures

of topics to adapt the background language model. As an example, Table 5.2 shows the probabilities of several words that capture the topics from document #230. The right-hand column shows the ratio of the adapted unigrams to the background unigrams. We see substantial improvement from each topic-related word in the list above. While some of these words will not appear in the translation, it is clearly visible that topics related to education and prison have emerged from the PLSA topic modeling based on the higher probability assignments.

While we have observed a significant improvement in the language model in terms of average perplexity, perplexity measures alone do not ensure a better translation. We must observe that the language model suggests good translations as well. Thus, let us now turn to the translation evaluation metrics. In the evaluation, we also notice a 0.85 improvement in BLEU (%), yielding a 3% improvement over the baseline. The same performance trend in NIST is observed with a 2.4% relative improvement compared to the unadapted baseline. Our PLSA and MDI-based adaptation method not only improves fluency but also improves adequacy: the topic-based adaptation approach is attempting to suggest more appropriate words based on increased unigram probabilities than that of the baseline LM. Without adapting the phrase translation table, we observe improvements in the translation.

While it is clear from the summary of the BLEU and NIST scores that the

Rank	Word	$P_A(w)$	$P_B(w)$	$P_A(w)/P_B(w)$
20	gens	8.41E-03	4.55E-05	184.84
22	vie	8.30E-03	1.09E-04	76.15
51	prix	2.59E-03	8.70E-05	29.77
80	école	1.70E-03	6.13E-05	27.73
83	argent	1.60E-03	3.96E-05	40.04
86	personnes	1.52E-03	2.75E-04	5.23
94	aide	1.27E-03	7.71E-05	16.47
98	étudiants	1.20E-03	7.12E-05	16.85
119	marché	9.22E-04	9.10E-05	10.13
133	étude	7.63E-04	4.55E-05	16.77
173	éducation	5.04E-04	2.97E-05	16.97
315	prison	2.65E-04	1.98E-05	13.38
323	université	2.60E-04	2.97E-05	8.75

Table 5.2: Sample unigram probabilities of the adaptation model for document #230, compared to the baseline unigram probabilities. The French words selected are semantically related to the English words in the adapted document. The PLSA adaptation infers higher unigram probabilities for words with latent topics related to the source document.

Bilingual PLSA model Table results in a performance increase, we would like to know whether the effects of our topic adaptation affect higher order n-grams. It is clear from Table 5.2 that unigram probabilities have improved. This improvement is also observed when looking at the NIST results in greater detail. Table 5.4 demonstrates a large improvement in unigram selection for the adapted TED model in terms of the individual contribution to the NIST score, with diminishing effects on larger n-grams. While we do see improvements in the second and third order n-grams, the majority of the overall improvements are on individual word selection.

Examples of improved fluency and adequacy are shown in Figure 5.7. Line 285 shows an example of a translation that doesn't provide much of an n-gram improvement, but demonstrates more fluent output, due to the deletion of the first comma and the movement of the second comma to the end of the clause. While "installation" remains an inadequate noun in this clause, the adapted model reorders the root words "rehab" and "installation" (in comparison with the baseline) and improves the

LM	Perplexity	BLEU (%)	NIST
Adapt TED	162.44	**28.49**	**6.5956**
Base TED	191.76	27.64	6.4405

Table 5.3: Perplexity, BLEU, and NIST scores for the baseline and adapted models. The perplexity scores are averaged across each document-specific LM adaptation.

(Line 285)

, j' ai eu la chance de travailler dans les *installations , rehab*

j' ai eu la chance de travailler dans les *rehab installation* ,

j' ai la chance de travailler dans un centre de désintoxication ,

(Line 597)

d' origine , les idées qui ont de la valeur –

d' avoir des idées originales qui ont de la valeur –

d' avoir des idées originales qui ont de la valeur –

(Line 752)

un nom qui appartient à *quelque* chose *d' autre* , le soleil .

un nom qui appartient à *autre* chose , le soleil .

le nom d' une *autre* chose , le soleil .

Figure 5.7: Three examples of improvement in MT results: the first sentence in each collection corresponds to the baseline, the second utilizes the adapted TED LMs, and the third is the reference translation.

NIST	1-gram	2-gram	3-gram
Adapt TED	4.8077	1.3925	0.3229
Base TED	4.6980	1.3527	0.3173
Difference	0.1097	0.0398	0.0056

Table 5.4: Individual unigram NIST scores for n-grams 1-3 of the baseline and adapted models. The improvement of the adapted model over the baseline is listed below.

grammaticality of the sentence; however, the number does not match between the determiner and the noun phrase. Line 597 demonstrates a perfect phrase translation with respect to the reference translation using semantic paraphrasing. The baseline phrase "d'origine" is transformed and attributed to the noun. Instead of translating "original" as a phrase for "home", the adapted model captures the original meaning of the word in the translation. Line 752 demonstrates an improvement in adequacy through the replacement of the word "quelque" with "autre." Additionally, extra words are removed.

These lexical changes result in the improvement in translation quality due to topic-based adaptation via PLSA.

5.3.4.2 Bilingual LDA Results

After repeating the same experiment with the LDA model described in Section 5.3.2, we observed a slightly lower performance in the Bilingual LDA approach, as opposed to its PLSA counterpart. However, these results do not appear to be statistically significant, which falls in line with Masada et al. (2008)'s findings. Table 5.5 shows the results of the evaluation task. The Bilingual LDA model demonstrates approximately a 13% improvement over the background TED language model in terms of perplexity. Additionally, we see a significant 0.67 improvement in BLEU score. These results are only slightly worse than the Bilingual PLSA counterpart.

In relation to the PLSA improvements shown in Figure 5.7, our LDA model yields identical translation outputs for the latter two sentences. The LDA model does not provide any changes to the first sentence in relation to the baseline translation output. Again, PLSA's modification did not yield an improvement in terms of BLEU; however, the output was slightly more fluent French.

One of the assumptions made in our bilingual topic modeling approach is that words in the aligned source and target languages will have similar (if not identical) topic distributions. Figure A.1 in the appendix shows the top 20 words for five latent topics, resulting from the Bilingual PLSA training in our experiment. Each list contains English and French words that are cognates of one another. We can additionally see what appear to be emerging topics from these top 20 word lists.

LM	Perplexity	BLEU (%)
PLSA Adapt	162.44	28.49
LDA Adapt	166.52	**28.31**
Base TED	191.76	27.64

Table 5.5: Perplexity, BLEU, and NIST scores for the baseline model and the PLSA and LDA models. The perplexity scores are averaged across each document-specific LM adaptation.

Topic heading suggestions have been listed in the figure. While these headings are speculative, they suggest a semantic grouping on the words in the bilingual texts based on their topics.

5.4 Chapter Summary

In this chapter, we extended topic modeling to the bilingual case. We discussed several language model adaptation approaches that target the statistical machine translation task. We discussed information retrieval-based approaches to language model adaptation, in which an adapted language model is constructed by searching for a subset of sentences in the training set that are most similar to the adaptation text. We briefly discussed mixture model techniques, in which language models are partitioned into domain-specific LMs and are recombined via interpolation. We additionally highlighted the HM-BiTAM approach which uses word alignment techniques using an admixture of topics to generate language models. We finally surveyed a Bilingual LSA approach that trains two Latent Dirichlet-tree models for the source and target languages, respectively, and enforces a one-to-one correspondence between posterior topic distributions.

We then discussed our bilingual topic modeling approach, which makes the naive assumption that since parallel texts refer to the same topics, parallel sentences will have virtually the same topic distributions and thus can be recombined into bilingual topics and treated in a monolingual fashion during PLSA or LDA training. During inference, monolingual texts are passed into the topic model, which yields word-document distributions for both the source and target language. After discarding the source distributions, we rescale the probabilities of target words into unigram language models for adaptation.

We evaluated our Bilingual PLSA and LDA models on the IWSLT 2010 TED lecture translation task and observed a significant improvement in terms of perplexity, BLEU, and NIST scores. We did not observe a significant difference between the use of PLSA or LDA in our model.

Chapter 6

Conclusion and Future Work

6.1 Summary

Language model adaptation remains a challenging problem for statistical machine translation. In this thesis, we provide an overview on several language model adaptation techniques that can be used in the context of machine translation.

We present a bilingual language model adaptation approach using Probabilistic Latent Semantic Analysis and MDI adaptation. Assuming a generative story that words in a document are generated from latent topics, we use PLSA and an adaptation text written in the source language to infer a unigram distribution over all target words in our language model's vocabulary. The unigram distribution is used to adapt our background language model via MDI adaptation and the adapted language model is subsequently used as a log-linear feature in our SMT system to translate the adaptation text. We test the use of this language model adaptation approach on the IWSLT 2010 evaluation task, which involves translating TED lectures from English to French.

In our experiment, we segment the parallel training data into "documents" consisting of 5 transcripts and train a PLSA model. During the decoding process, we segment the test data again into 5-line documents and construct an adapted language model for each document for the purposes of translation. We observe an average improvement in language model perplexity of 15% and a BLEU score improvement of 0.8 over the baseline. We observe translation improvements both in terms of fluency and adequacy. We performed the same experiment with a LDA model and observed an improvement of 13% and 0.67 in terms of average perplexity and BLEU, respectively.

6.2 Future Work

6.2.1 Multilingual topic-based language model adaptation

We have thus far presented a bilingual topic modeling approach that is applied to the task of lecture translation. Our model uses a simplified approach in which the bilingual topic modeling case is treated as a special case of a monolingual topic model in which the source and target languages are merged. Ideally, we may be able to generalize the bilingual assumption to the multilingual case, in which multiple languages are merged into a single topic model.

Additionally, in our experiments, we assumed that one language in our topic model serves a distinct role as the "source" language, while the other serves as the "target" language. Under our bilingual topic model construction, this need not be the case. The PLSA and LDA models were trained on parallel data and inherently made no assumption regarding which language serves as the source language. The assumption is only made when applying the topic model to the task of machine translation. Thus, without loss of generality, our bilingual topic model could have also been used in a French to English translation experiment.

Likewise, assuming that our bilingual topic modeling approach generalizes to the multilingual case, we can construct a unified multilingual topic model that supports multiple translation directions. For example, if we were to include the Arabic language in our translation task, our topic model could support bidirectional translation between English and French, English and Arabic, and French and Arabic.

6.2.2 MDI adaptation alternatives

One future area of improvement involves exploring alternatives for MDI adaptation techniques. Recall from Section 3.5.2 that in MDI adaptation, a background language model is adapted by a ratio of the unigram probabilities of an adapted text and the unigram probabilities of the background text. In the computation of the adapted frequency counts (3.22) and back-off weights (3.23), a normalization factor $z(h)$ must be computed for MDI adaptation which consists of summing over every n-gram that appears in the background and adapted texts:

$$z(h) = \left(\sum_{w:N_B(h,w)>0} f_B^*(w|h)\alpha(w) \right) + \lambda_B(h) z(h').$$

While this normalization term is efficient to compute, it is expensive to compute for higher order n-gram models trained on corpora with large vocabularies. This becomes a computational bottleneck in the task of continuous speech translation, where the goal is to translate speeches in real-time.

Since the important part of MDI adaptation is the ratio of unigram probabilities between the adapted and background texts, we would ideally like to construct an adaptation model that uses unigram ratios without computing large normalization terms. One idea is to include the ratio of these unigram language models as additional features in our log-linear model and learning the weights of the corresponding parameter γ during minimum error rate training. Thus, if our feature is defined as:

$$g(w) = \left(\frac{P_A(w)}{P_B(w)} \right)^{\gamma}, \tag{6.1}$$

it can be encoded as a log-linear feature by simply taking the log:

$$\log g(w) = \gamma \log P_A(w) - \gamma \log P_B(w) \tag{6.2}$$

and adding it to our log-linear model.

6.2.3 Adapting translation tables via Bilingual PLSA

While our experiments only focused on language model adaptation, we are confident that our Bilingual PLSA (and LDA) approach can also be used to adapt the translation model. Tam et al. (2007) outline one translation table adaptation technique for LDTA which follows a similar technique to MDI adaptation. A similar approach could be adopted for our technique.

6.2.4 Advanced PLSA models

With the advent of LDA, additional research on PLSA has largely been ignored by the research community. While many advanced LDA models have been constructed, such as topic models that relax the independence assumption on latent topics (e.g. LDTA), PLSA more or less remains as its original construction by Hofmann (1999). Since it has been shown in our bilingual experiment and by that LDA models do not necessarily yield a better topic model than PLSA, PLSA and LDA should continue to be compared in more complex models.

Appendix A

Appendix

A.1 Bilingual LDA Evaluation

Topic 22 Time/Numbers?	Topic 30 Relationships?	Topic 39 Medical?	Topic 46 Buildings?	Topic 62 Masculine articles?
,	0	de	the	il
five	:	the	of	he
,	.	les	a	,
ans	,	des	de	,
de	:	of	and	and
cinq	,	la	,	et
10	love	0	l’	his
10	un	,	building	0
years	amour	patients	la	.
à	a	patients	une	son
15	friend	du	structure	a
0	le	and	,	him
15	ami	.	structure	lui
four	einstein	version	d’	sa
5	einstein	et	à	to
.	film	in	et	qu’
minutes	film	en	architecture	un
minutes	secret	version	architecture	then
quatre	!	process	un	à
ou	secret	protection	bâtiment	de

Figure A.1: Top 20 words in 5 topics from a LDA-trained model.

Bibliography

Adam L. Berger, Peter F. Brown, Stephen A. Della Pietra, Vincent J. Della Pietra, John R. Gillett, John D. Lafferty, Robert L. Mercer, Harry Printz, and Luboš Ureš. The Candide system for machine translation. In *Proceedings of the Workshop on Human Language Technology*, HLT '94, pages 157–162, Stroudsburg, PA, USA, 1994. Association for Computational Linguistics. ISBN 1-55860-357-3. URL http://dx.doi.org/10.3115/1075812.1075844. 12

David M. Blei, Andrew Ng, and Michael Jordan. Latent Dirichlet Allocation. *JMLR*, 3:993–1022, 2003. 31, 33, 34

Peter F. Brown, Stephen A. Della Pietra, Vincent J. Della Pietra, and Robert L. Mercer. The mathematics of statistical machine translation: Parameter estimation. *Computational Linguistics*, 19(2):263–312, 1993. URL http://aclweb.org/anthology-new/J/J93/J93-2003.pdf. 8, 9

Scott S. Chen and P. S. Gopalakrishnan. Speaker, Environment and Channel Change Detection and Clustering via the Bayesian Information Criterion. In *DARPA Broadcast News Transcription and Understanding Workshop*, Lansdowne, VA, 1998. 43

Stanley F. Chen and Joshua Goodman. An empirical study of smoothing techniques for language modeling. Technical Report TR-10-98, Harvard University, 1998. 22

J. N. Darroch and D. Ratcliff. Generalized Iterative Scaling for Log-Linear Models. *The Annals of Mathematical Statistics*, 43(5):1470–1480, 1972. 24

Scott Deerwester, Susan T. Dumais, George W. Furnas, Thomas K. Landauer, and Richard Harshman. Indexing by Latent Semantic Analysis. *Journal of the American Society for Information Science*, 41:391–407, 1990. 29

Renato DeMori and Marcello Federico. Language model adaptation. In K. Pointing, editor, *Computational Models of Speech Pattern Processing*, NATO ASI Series, pages 280–301. Addison Wesley, 1999. 26, 27

A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum-likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, B*, 39: 1–38, 1977. 10

George Doddington. Automatic evaluation of machine translation quality using n-gram co-occurrence statistics. In *Proceedings of the Second International Conference on Human Language Technology Research*, HLT '02, pages 138–145, San Francisco, CA, USA, 2002. Morgan Kaufmann Publishers Inc. URL http://portal.acm.org/citation.cfm?id=1289189.1289273. 16, 17

Marcello Federico. Efficient Language Model Adaptation through MDI Estimation. In *Proceedings of the 6th European Conference on Speech Communication and Technology*, volume 4, pages 1583–1586, Budapest, Hungary, 1999. 24, 25

Marcello Federico. Language Model Adaptation through Topic Decomposition and MDI Estimation. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, volume I, pages 703–706, Orlando, FL, 2002. 25, 31, 38, 44

Marcello Federico, Nicola Bertoldi, and Mauro Cettolo. IRSTLM: an Open Source Toolkit for Handling Large Scale Language Models. In *Proceedings of Interspeech*, pages 1618–1621, Melbourne, Australia, 2008. 43

George Foster and Roland Kuhn. Mixture-Model Adaptation for SMT. In *Proceedings of the Second Workshop on Statistical Machine Translation*, pages 128–135, Prague, Czech Republic, June 2007. Association for Computational Linguistics. URL http://www.aclweb.org/anthology/W/W07/W07-0217. 37

Mark Girolami and Ata Kabán. On an equivalence between PLSI and LDA. In *Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Informaion Retrieval*, SIGIR '03, pages 433–434, New York, NY, USA, 2003. ACM. ISBN 1-58113-646-3. URL http://doi.acm.org/10.1145/860435.860537. 34

I. J. Good. The Population Frequencies of Species and the Estimation of Population Parameters. *Biometrika*, 40:237–264, 1953. 20

Thomas L. Griffiths and Mark Steyvers. Finding scientific topics. *Proceedings of the National Academy of Sciences*, 101(Suppl. 1):5228–5235, April 2004. 33, 34

Gregor Heinrich. Parameter Estimation for Text Analysis. Technical report, University of Leipzig, 2008. URL http://www.arbylon.net/publications/text-est.pdf. 33

Thomas Hofmann. Probabilistic Latent Semantic Analysis. In *Proceedings of the 15th Conference on Uncertainty in AI*, pages 289–296, Stockholm, Sweden, 1999. 29, 53

Frederick Jelinek and Robert L. Mercer. Interpolated estimation of Markov source parameters from sparse data. In *Pattern Recognition in Practice*, pages 381–397, Amsterdam, Holland, 1980. 21

Daniel Jurafsky and James H. Martin. *Speech and Language Processing (2nd Edition) (Prentice Hall Series in Artificial Intelligence)*. Prentice Hall, 2 edition, 2008. ISBN 0131873210. 20

Slava M. Katz. Estimation of Probabilities from Sparse Data for the Language Model Component of a Speech Recognizer. *IEEE Trans. Acoust., Speech and Signal Proc.*, ASSP-35(3):400–401, 1987. 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, Detroit, MI, 1995. 21

Reinhard Kneser, Jochen Peters, and Dietrich Klakow. Language Model Adaptation Using Dynamic Marginals. In *Proceedings of the 5th European Conference on Speech Communication and Technology*, pages 1971–1974, Rhodes, Greece, 1997. 25

P. Koehn, H. Hoang, A. Birch, C. Callison-Burch, M. Federico, N. Bertoldi, B. Cowan, W. Shen, C. Moran, R. Zens, C. Dyer, O. Bojar, A. Constantin, and E. Herbst. Moses: Open Source Toolkit for Statistical Machine Translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions*, pages 177–180, Prague, Czech Republic, 2007. URL `http://aclweb.org/anthology-new/P/P07/P07-2045.pdf`. 43

Philipp Koehn. *Statistical Machine Translation*. Cambridge University Press, 2010. 9, 10, 11, 12, 13, 21

Philipp Koehn and Josh Schroeder. Experiments in Domain Adaptation for Statistical Machine Translation. In *Proceedings of the Second Workshop on Statistical Machine Translation*, pages 224–227, Prague, Czech Republic, June 2007. Association for Computational Linguistics. URL `http://www.aclweb.org/anthology/W/W07/W07-0233`. 37, 38

Tomonari Masada, Senya Kiyasu, and Sueharu Miyahara. Comparing LDA with pLSI as a dimensionality reduction method in document clustering. In *Proceedings of the 3rd International Conference on Large-scale Knowledge Resources: Construction and Application*, LKR'08, pages 13–26, Berlin, Heidelberg, 2008. Springer-Verlag. ISBN 3-540-78158-7, 978-3-540-78158-5. URL `http://portal.acm.org/citation.cfm?id=1787800.1787804`. 34, 48

Hirokazu Masataki, Yoshinory Sagisaka, Kazuya Hisaki, and Tatsuya Kawahara. Task adaptation using MAP estimation in n-gram language modelling. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, volume 1, pages 783–786, Munich, Germany, 1997. 27

Franz J. Och and Hermann Ney. The Alignment Template Approach to Statistical Machine Translation. *Computational Linguistics*, 30(4):417–450, 2004. 13

Franz Josef Och. Minimum Error Rate Training in Statistical Machine Translation. In Erhard Hinrichs and Dan Roth, editors, *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, pages 160–167, 2003. URL `http://www.aclweb.org/anthology/P03-1021.pdf`. 17, 43

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. BLEU: a Method for Automatic Evaluation of Machine Translation. Research Report RC22176, IBM Research Division, Thomas J. Watson Research Center, 2001. 16, 17

Michael Paul, Marcello Federico, and Sebastian Stücker. Overview of the IWSLT 2010 Evaluation Campaign. In Marcello Federico, Ian Lane, Michael Paul, and François Yvon, editors, *Proceedings of the seventh International Workshop on Spoken Language Translation (IWSLT)*, pages 3–27, 2010. 42

Xuan-Hieu Phan and Cam-Tu Nguyen. GibbsLDA++: A C/C++ Implementation of Latent Dirichlet Allocation (LDA) using Gibbs Sampling for Parameter Estimation and Inference. http://gibbslda.sourceforge.net/, 2007–2008. 45

Nick Ruiz and Marcello Federico. Topic Adaptation for Lecture Translation through Bilingual Latent Semantic Models. In *Proceedings of the Sixth Workshop on Statistical Machine Translation*, pages 294–302, Edinburgh, Scotland, July 2011. Association for Computational Linguistics. URL http://www.aclweb.org/anthology/W11-2133. 40

Claude E. Shannon. A mathematical theory of communication. *Bell System Technical Journal*, 27(3):379–423, 1948. 7

Yik-Cheung Tam and T. Schultz. Correlated Latent Semantic Model for Unsupervised LM Adaptation. In *Acoustics, Speech and Signal Processing, 2007. ICASSP 2007. IEEE International Conference on*, volume 4, pages IV–41–IV–44, 2007. doi: 10.1109/ICASSP.2007.367158. URL http://dx.doi.org/10.1109/ICASSP.2007.367158. 39

Yik-Cheung Tam and Tanja Schultz. Incorporating monolingual corpora into bilingual latent semantic analysis for crosslingual LM adaptation. In *Acoustics, Speech and Signal Processing, 2009. ICASSP 2009. IEEE International Conference on*, pages 4821 –4824, april 2009. doi: 10.1109/ICASSP.2009.4960710. 39

Yik-Cheung Tam, Ian Lane, and Tanja Schultz. Bilingual LSA-based adaptation for statistical machine translation. *Machine Translation*, 21:187–207, December 2007. ISSN 0922-6567. doi: 10.1007/s10590-008-9045-2. URL http://portal.acm.org/citation.cfm?id=1466799.1466803. 39, 40, 53

Christoph Tillmann and Hermann Ney. Word reordering and a dynamic programming beam search algorithm for statistical machine translation. *Computational Linguistics*, 29(1):97–133, 2003. URL http://aclweb.org/anthology-new/J/J03/J03-1005.pdf. 15

Warren Weaver. Translation. In William N. Locke and A. Donald Boothe, editors, *Machine Translation of Languages*, pages 15–23. MIT Press, Cambridge, MA, 1949/1955. Reprinted from a memorandum written by Weaver in 1949. 7

Bing Zhao and Eric P. Xing. Bitam: Bilingual topic admixture models for word alignment. In *Proceedings of the COLING/ACL 2006 Main Conference Poster Sessions*, pages 969–976, Sydney, Australia, July 2006. Association for Computational Linguistics. URL http://www.aclweb.org/anthology/P/P06/P06-2124. 38

Bing Zhao and Eric P. Xing. HM-BiTAM: Bilingual Topic Exploration, Word Alignment, and Translation. In J.C. Platt, D. Koller, Y. Singer, and S. Roweis, editors, *Advances in Neural Information Processing Systems 20*, pages 1689–1696. MIT Press, Cambridge, MA, 2008. 38, 40

Bing Zhao, Matthias Eck, and Stephan Vogel. Language Model Adaptation for Statistical Machine Translation via Structured Query Models. In *Proceedings of Coling 2004*, pages 411–417, Geneva, Switzerland, Aug 23–Aug 27 2004. COLING. 37