Context-aware Language Modeling for Conversational Speech Translation

Avneesh Saluja, Ian Lane, Ying Zhang
Carnegie Mellon University
Moffett Field, CA
{avneesh.saluja,ian.lane,joy.zhang}@sv.cmu.edu

Abstract

Context plays a critical role in the understanding of language, especially conversational speech. However, few approaches exist to utilize the external contextual knowledge which is readily available to practical speech translation systems deployed in the field. In this work, we propose a novel framework to integrate context in the language models used for conversational speech translation. The proposed approach takes into account the contextual distance between a test utterance and the training corpus, a measure obtained from the external context in which the utterances were spoken. Language model probabilities are adjusted through a sentence-level weighting scheme based on this context-distance measure. When incorporated into our English-Iraqi Arabic speech-to-speech translation system, the proposed approach obtains improvements in both speech recognition accuracy and translation quality compared to the baseline system.

1 Introduction

Conversational speech is full of ambiguities, yet humans can easily overcome this under-specification by leveraging contextual knowledge that is not present in the surface form of an utterance. A common conversational speech scenario is task-orientated dialog where lexical and grammatical features differ significantly based on situation and participants, for example ordering at a restaurant or asking for directions on the street. The sentence "please do not make it spicy" has a higher likelihood in one context, namely the restaurant, than the other. In many languages, context information such as the gender of the speakers in a conversation can alter the sentence structure and word choice. These examples are natural indicators of the role that context can play in speech translation.

Conversational speech translation is particularly applicable in mobile settings, where portable handheld devices can be used to support cross-lingual communication while in the field. Recent developments in mobile speech-to-speech translation (Zhang and Vogel, 2007; Tan et al., 2008; Prasad et al., 2007; Bach et al., 2007; Eck et al., 2010) have worked to overcome challenges encountered in fielded systems, making this use case more common and effective.

However, modeling conversational speech in dynamic, mobile settings presents additional challenges to standard speech translation, such as constantly varying settings and high-noise environments. Yet, these systems are unique in that context information can be obtained from sources other than the surface-level of utterances relatively easily. Examples include implicit context such as GPS location, semi-implicit context such as gender from the user profile or topic from a calendar appointment, or explicit context defining the setting or task the user is attempting to perform. Contextual knowledge can play an important role in constraining the acoustic, translation, alignment, and language models of a speech translation system (Figure 1). Current speech translation systems however, are unable to effectively utilize such information due to the lack of a consistent framework for context adaptation.

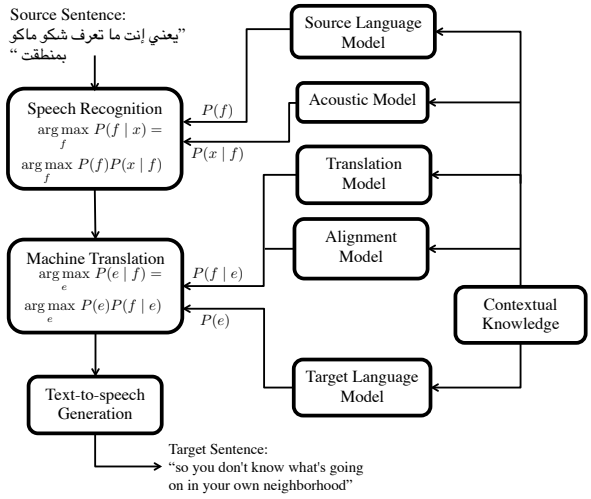

Figure 1: Overview of the main components in a speech-to-speech translation system and the internal models that can be affected by contextual knowledge

Extending our early work of developing context-aware translation systems in the virtual world (Zhang, 2009), we present a novel framework for incorporating contextual knowledge into statistical language models, namely Distance-Measure Tuning (Section 2). When applied to conversational speech translation, our proposed approach improves both the speech recognition accuracy and the machine translation quality (Sections 3 and 4) compared to a context-independent system.

2 Context and Language Models

Statistical language modeling assigns a probability to a sequence of words and is commonly used in natural language processing to model the properties of language and predict the next word in a sequence given a known linguistic history. In speech translation, the language model is an integral component across several different stages (Figure 1). First, during speech recognition, the language model (LM) guides the decoder, inferring the most likely sequence of words in the source language given an observed speech signal. Similarly, in machine translation, the LM helps eliminate unlikely translations and re-orderings by evaluating likelihood of word sequences in the target language.

In this work, we focus on utterances with explicitly marked context for adaptation. We look at the question of incorporating context at a broader level, potentially including context represented in any form, such as text or real numbers. For mobile speech-to-speech translation, we can augment spoken utterances with explicit context through a variety of means, either via sensors built into the phone (e.g., location of the user defined by GPS), user profile information, or within the recognition process (e.g., gender detection or topic classification). The framework we propose can handle any type of contextual knowledge once it has been extracted.

To define the context associated with a given utterance, we introduce the concept of a *context vector*, which indicates the external context in which the sentence was spoken. The context vector provides an intuitive way to attach detailed context information to a particular utterance or dialog, and only requires the definition of context *categories*. Context categories, such as "location" or "task", are easier to pre-designate compared to specific contexts like "restaurant" or "ordering food". For example, the context vector for the sentence "so you don't know what's going on in your own neighborhood", taken directly from the English-Iraqi corpus we used for evaluation is:

$$\langle \text{topic}_1\text{: policing}, \text{topic}_2\text{: intelligence}, \text{tone: cooperative} \rangle$$

Henceforth, we refer to this particular context vector as the *example context vector*. In this case, the system has 3 context categories: two topic contexts, and one tone.

2.1 Context-dependent LMs through Linear Interpolation

One common approach for context adaptation is to train individual models on corpora segmented by context, and linearly interpolate these models (Iyer and Ostendorf, 1996; Bulyko et al., 2007; Sanchis-Trilles and Cettolo, 2010), obtaining the interpolation weights by minimizing perplexity over a held-out tuning set. However, there are several limitations to this approach. First, the number of unique context vectors grows polynomially in the number of contexts, assuming a fixed number of context categories, resulting in fewer sentences assigned to each context vector. Using the *example context vector* as our example with 3 context categories (two topics and one tone), if we initially had five topic values (e.g., "checkpoint", "intelligence", "policing", "small talk", and "medical") and two tones ("cooperative" and "adversarial"), then we would have a possible set of $5 \text{ topics} \times 5 \text{ topics} \times 2 \text{ tones} = 50$ unique context vectors. If we then double the number of topics, the number of unique context vectors increases four-fold. Thus, adding contexts to the system will lead to data sparseness issues which in turn will potentially lead to problems during parameter estimation (Section 3.3).

Second, the approach assumes the impact of multiple contexts on the language model probability (or score) can be expressed as a linear combination of language model scores provided by models that are trained on individual contexts. In reality, the relationship is more complex and would be better served by a more granular level of LM adjustment, for example at the sentence level.

2.2 Distance-Measure Tuning (DMT)

To overcome the limitations inherent in linear interpolation, we propose a new approach in which we generate an LM for each dialog scenario based entirely on its context vector. For every input sentence, each sentence in the training corpus is assigned a weight, expressed as the function of a distance between the context vectors of the input and training utterances. We introduce a distance metric that weighs each context category individually and estimate sentence weights to minimize the perplexity over a tuning set.

For a given input utterance $s = (w_1, \ldots, w_l)$ where l is the sentence length (word count), with associated context vector $\vec{c_s}$, we evaluate its likelihood in a modified manner as:

$$\begin{aligned} \log P(s) &= \sum_{i=1}^{l} \log P(w_i \mid h) \qquad (1) \\ &= \sum_{i=1}^{l} \log \left(\sum_{j=1}^{N} \alpha_{js} \frac{C_j(w_i, h)}{C_j(h)} \right) \end{aligned}$$

where N is the number of sentences in the training corpus, h is the history of word w_i, $C_j(\cdot)$ is the count of the word sequence in training corpus sentence j, and α_{js} is the weight assigned to sentence j and is a function of the context-distance measure between training sentence j and test sentence s. The relationship is defined as:

$$\alpha_{js} = \frac{1 - D(\vec{c_j}, \vec{c_s}) + b}{Z} \quad (2)$$

$D : S^{|S|} \times S^{|S|} \mapsto \mathbb{R}$ is the distance measure between the input and training sentence context vectors, where S is the set of all possible context values. b is a parameter that ensures we assign a strictly positive[1] weight to each training sentence, and Z is a normalization factor to ensure that $\sum_{j=1}^{N} \alpha_{js} = N$, which guarantees we do not distort the word count of the corpus.

This framework offers a flexible definition of the distance measure. Here, we use a linear function $D(\vec{c_j}, \vec{c_s}) = \vec{\lambda}^T \cdot f(\vec{c_j}, \vec{c_s})$, where $\vec{\lambda}$ is a parameter vector equivalent in length to the context vector, and $f(\vec{c_j}, \vec{c_s})$ is the modified Hamming distance function that measures the similarity between context vectors by comparing each entry in the vectors and outputs the corresponding binary vector:

$$f_i(\vec{c_j}, \vec{c_s}) = \begin{cases} 0 & \text{if } c_j(i) = c_s(i) \\ 1 & \text{otherwise} \end{cases} \quad (3)$$

where i refers to the i^{th} entry in the context vector. As an example, if we use our *example context vector* as a test context vector and the context vector

$$\langle \text{topic}_1\text{: policing}, \text{topic}_2\text{: checkpoint}, \text{tone: adversarial} \rangle$$

then the modified Hamming distance function returns a vector $\langle 0, 1, 1 \rangle$.

In order to find the optimal parameters for $\vec{\lambda}$, we use the Nelder-Mead method (Nelder and Mead, 1965) over the tuning set to minimize total perplexity.

3 Experimental Evaluation

Evaluation was performed using phase one of the English-Iraqi TransTAC corpus, which consists of spoken language transcriptions of dialogs between US personnel and Iraqis in the field. The corpus covers tasks including: vehicle checkpoints, training of soldiers and medical assistance. There are a total of 19 context values in the corpus. We chose two context categories to form the context vector of length 3 (two topics and one tone), reducing the set of possible unique context vectors to 220. The tone context category contains two values, either "cooperative" (84.5%) or "adversarial" (15.5%). The topic contexts distribution for the first topic (topic_1) is presented in Figure 2. Table 1 presents the top ten context vectors in our corpus, making up 62.7% of the corpus in total. Our training, tuning, and test sets consisted of roughly 38,000, 1000, and 1000 utterances respectively (Table 2). Dialogs were randomly sampled to create the tuning and test sets.

Unique Context Vector	**Number of Utterances**	**% of Total Utterances**
Other (no marker)	4349	11.3%
Intelligence None Cooperative	2957	7.7%
Checkpoint None Cooperative	2897	7.5%
Swet None Cooperative	2773	7.2%
Medical None Cooperative	2234	5.8%
Community Issues None Cooperative	2185	5.7%
Raidsearch Intell Cooperative	2103	5.5%
Joint Ops None Cooperative	1610	4.2%
Checkpoint Intell Cooperative	1504	3.9%
Checkpoint None Adversarial	1453	3.8%
Remaining Unique Contexts	14,319	37.3%
Total	**38,384**	**100%**

Table 1: Top 10 unique context vectors in corpus

Corpus	**Sentences**	**Words**
Training	38,384	441,094
Tuning	948	8657
Test	990	9642

Table 2: Number of words and sentences for the training, tuning, and test corpora that we used

3.1 Experimental Setup

To generate the sentence-level weighted language models, we evaluated the contextual difference between the training corpus and test sentences through a perl script, and incorporated the weights when training the models through built-in features in the SRI-LM (Stolcke, 2002) toolkit. We used order 3 n-grams, and performed Witten-Bell smoothing due to limited support of fractional counts for other smoothing techniques in the SRI-LM toolkit. SRI-LM was also used for perplexity measurements, evaluated on the English LMs.

When evaluating the effectiveness of our approach on speech recognition accuracy and machine translation quality, we used the CMU English-Iraqi Arabic speech translation system. For speech recognition, we evaluated English Word Error Rate (WER), keeping the entire system setup constant apart from the language models. Our English ASR system consisted of a sub-phonetically tied, semi-continuous, HMM acoustic model which was composed of 7000 context dependent senones and up to 64

[1] $\alpha_{js} = 0$ should be avoided as that essentially discards the training sentence and unnecessarily shrinks the training corpus

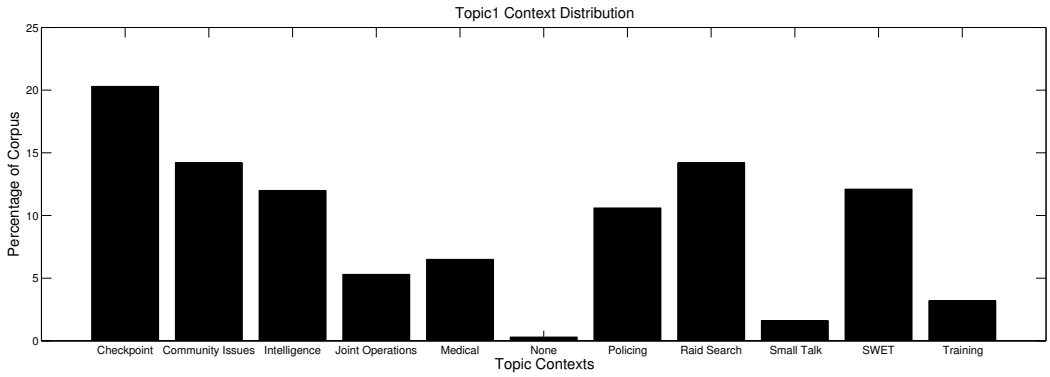

Figure 2: Distribution of the context category topic_1 by its context values in our corpus. For topic_2, roughly 50% of the context values were "none", with the remaining topics distributed in a similar manner to topic_1. Only a few sentences in the corpus did not have *at least one* topic

Gaussians per state. ASR decoding was performed using the Ibis decoder (Soltau et al., 2001), which was developed as part of our Janus Recognition Toolkit (JRTk) (Finke et al., 1997).

We used Moses (Koehn et al., 2007) for statistical machine translation (SMT) decoding and training all of the SMT models except the language models. MERT tuning was achieved by the associated `mert-moses.pl` script (Och, 2003). The translation hypotheses were evaluated with BLEU (Papineni et al., 2002), and we evaluated the Iraqi Arabic to English source-to-target direction, keeping the entire translation setup static except for the language model component.

3.2 Baseline Systems

First, two context-independent LMs were trained. In the first baseline ("baseline"), all sentences were given equal weights during LM training, namely weight 1. For the linear interpolation case ("baseline intLM"), we first created multiple sub-corpora, wherein each corpus contained training sentences from only one context value. Note that these sub-corpora overlapped as training sentences were often marked with multiple context values and categories. For example, a sentence with the *example context vector* as its context vector existed in the "policing", "intelligence", and "cooperative" sub-corpora. We then trained an LM on each sub-corpus, and estimated a set of mixture weights for each LM by minimizing the perplexity over the entire unsegmented tuning set, resulting in a single context-weighted LM.

3.3 Context and Language Model Perplexity

The "baseline intLM" outperformed "baseline" by 7.6% on the unseen test set in terms of perplexity (Table 3, columns 2 and 3), indicating the usefulness of interpolation in reducing perplexity.

Next, we evaluated the effectiveness of the context-dependent linear interpolation (Section 2.1) and DMT (Section 2.2) methods. With the context-dependent linearly interpolated LMs, we maintained different sets of mixture weights for each context vector in the tuning set. We split the tuning set into 13 subsets: 12 corresponded to the 12 unique context vectors that constituted 80% of the tuning set. The context vectors of the remaining 20% of the tuning set had few training examples per context vector, and were grouped together to ensure a model ("other") of sufficient size.

We tuned for an optimal set of interpolation weights for each context vector-specific subset of utterances in the tuning set. Through this tuning process, we obtained a context vector-dependent set of interpolation weights, resulting in a context-dependent LM. During evaluation, when a context vector that we had tuned for was encountered, the corresponding mixture weights vector was applied, otherwise we used the "other" mixture weights. Since the context vectors in the tuning and test sets were not identical, unique contexts that existed in the test but not in the tuning set fall back to the "other" case, resulting in suboptimal performance[2].

When evaluating the DMT methods, we used the same 13-way split of the tuning set and varied the way in which we selected or computed $\vec{\lambda}$, the parameter set used within the context distance measure (Equation 2):

1. DMT Uniform: uniform parameters $\vec{\lambda} = \langle \frac{1}{k}, \ldots, \frac{1}{k} \rangle$. In our case, $k = 3$ (Table 3, column 5) .

2. DMT Tuned-ppl: Nelder-Mead (multidimensional downhill simplex) tuned parameters. The optimization goal is to minimize perplexity on the tuning set (Table 3, column 6).

[2]Roughly 50% of the test utterances were evaluated with the "other" parameters

Set	Baseline	Baseline int. LM	Cont.-Dep. int. LM	DMT Uniform	DMT Tuned-ppl	DMT Tuned-aug
Tune	48.3	43.8	41.2 (6.0%)	47.3 (2.0%)	41.4 (14.2%)	46.7 (3.2%)
Test	63.5	58.7	58.4 (0.4%)	63.5 (0.0%)	59.6 (6.1%)	60.1 (5.4%)

Table 3: Summary of Results: Perplexity (% relative improvement). The results show marginal improvement over the baseline for context-dependent interpolated LMs, and noticeable improvement for DMT

3. DMT Tuned-aug: in addition to tuning the parameters in 2, we augmented our parameter set to also include parameters corresponding to the most frequently occurring topic pairs and tuned for these augmented parameters as well (Table 3, column 7). In particular, we modified Equation 2 slightly by introducing hierarchy in the topics. We picked the n most frequent topics in the corpus and introduced $\frac{n(n-1)}{2}$ additional parameters corresponding to the pairs amongst these n topics. These parameters were used whenever any of the n most frequent topics occurred as a pair when evaluating the contextual distance between two context vectors, otherwise we backed off to the more general parameters (as tuned in 2).

We also tuned for b, the bias introduced in Equation 2, and found that a value of 0.25 performed consistently well across the experiments.

Context-dependent interpolated LMs reduced perplexity (Table 3, column 4) by 6.0% (tuning) and 0.4% (test). Perplexity improvements for each context vector ranged from 1.5% to 20.7% on the tuning set, and -8.2% to 17.8% on the test set. DMT decreased perplexity on the tuning set by 14.2%, and while its performance on the test set in terms of absolute perplexity levels was not better than the linearly interpolated models, relative to its baseline ("baseline") its improvement was higher than the linearly interpolated model compared to its baseline ("baseline intLM"). Context vector-wise, DMT improvements ranged from -10.8% (which occurred during "DMT Uniform" evaluation) to 10.4% ("DMT Tuned-ppl") with no apparent link between the size of the context vector-based subset and the amount of improvement on tuning or test sets.

3.4 Context-dependent Language Modeling for Speech Recognition

Next, we evaluated the performance of context-dependent language models for ASR. Between the two baselines, the WER did not change (Table 4, columns 2 and 3). The context-dependent interpolated LMs method achieved a WER of 17.1% on the test set, an absolute improvement of 0.4% compared to "baseline intLM" and "baseline" (Table 4, column 4), with context vector-specific improvements ranging from -8.4% to 18.4%. Our approach, DMT, further reduced WER compared to the interpolated case through an absolute additional improvement of 0.2%, with an absolute improvement over "baseline" of 0.6%. Improvements by context varied from -30.0% to 33.3% with an overall relative reduction of 3.1% over "baseline".

In addition to the results presented in Table 4, we found that if DMT is used in all cases (as opposed to backing-off to the interpolated LMs when we encountered a pre-tuned context vector in the test set), a WER of 17.0% is obtained, a 3.0% relative reduction compared to "baseline". In addition, we note that on the test set, if an efficient pre-selection mechanism is implemented such that one can choose between the interpolated LMs and DMT approach for context vectors encountered in the tuning set, we achieve a WER of 16.9%, a 3.9% improvement over "baseline".

3.5 Context-dependent Language Modeling for Machine Translation

Finally, we evaluated the effectiveness of context-aware language models on machine translation quality. The BLEU evaluation was preceded by a tuning step, where we used the same held-out tuning set to obtain the machine translation weights, namely the translation, distortion, language model and word penalty weights used in the computation of the translation hypothesis during decoding. The "baseline intLM" obtained a small increase in BLEU (0.29 BLEU points) compared to the "baseline" (Table 5, columns 2 and 3) method where interpolation was not performed.

The DMT BLEU results exhibited a similar trend to their corresponding WER results. We see that LMs optimized for minimizing perplexity (Table 5, column 6) make marginal improvements over the baseline when evaluated for BLEU (0.13 point increase), and that augmenting these parameters (column 8) does little to help the results. In fact, a uniform λ approach (column 5), where each $\lambda = \frac{1}{3}$, outperforms both tuned techniques. In light of these results we decided to tune our λ parameters in DMT with the aim of maximizing BLEU on the tuning set (column 7), by performing a simple grid search with a step size of $\Delta\lambda = 0.05$. Through this method, we found that the parameter settings that maximized tuning set BLEU improved unseen test set BLEU by 0.6 BLEU points compared to "baseline".

In an effort to keep the system static to isolate the effects of the language model on translation quality, we

Set	Baseline	Baseline int. LM	Cont.-Dep. int. LM	DMT Uniform	DMT Tuned-ppl	DMT Tuned-aug
Test	17.5%	17.5%	17.1% (1.9%)	16.9% (3.1%)	16.9% (3.1%)	16.9% (3.1%)

Table 4: Summary of Results: WER (% relative improvement). The results show consistent improvement when incorporating context

Set	Baseline	Baseline int. LM	Cont.-Dep. int. LM	DMT Uniform	DMT Tuned-ppl	DMT Tuned-bleu	DMT Tuned-aug
Test	18.24	18.53	18.21 (-1.7%)	18.45 (1.2%)	18.37 (0.7%)	18.84 (3.3%)	18.37 (0.7%)

Table 5: Summary of Results: BLEU (% relative improvement). The results show significant outperformance versus interpolated LMs, and mild improvement over the baseline for perplexity-optimized LMs. Parameters tuned to optimize BLEU outperform other experimental setups

used the same set of MERT weights obtained by the "baseline" setup for the DMT experiments. However for the context-dependent interpolated LMs we found that the MERT weights obtained from "baseline intLM" setup were inappropriate for the context-dependent models and severely penalized hypothesis sentence length, and thus underwent the MERT step again for each model. Despite this retuning, the context-dependent interpolated LMs approach obtained a lower BLEU score (Table 5, Column 4) than its corresponding baseline, "baseline intLM".

4 Discussion

Overall, one sees a) that context plays an important role in statistical language modeling since all context-dependent evaluations outperform the baseline and b) relative to their respective baselines, DMT outperforms the LM interpolation approach, as described in Section 2.1, on the unseen test set.

For perplexity evaluations, we combined the interpolated LM approach with DMT and used the mixture models setup if a pre-tuned context vector was encountered in the test corpus, as the interpolated LMs approach had equivalent or slightly better perplexity performance on pre-tuned context vectors. Otherwise, we generated a context vector-specific LM on the fly, as generalizing to unseen contexts is a main advantage of DMT. While computational costs may inhibit the construction of an LM online, we note that techniques to estimate LM probabilities on the fly (Zhang and Vogel, 2006) can help to alleviate such costs. In addition, we found that "DMT Tuned-aug", with its topic pair-specific parameters, generally was subpar compared to the smaller parameter set, most likely due to overfitting during the tuning process.

In these experiments we observed that perplexity does not correspond with WER, in a consistent manner, i.e. techniques that perform well on an absolute level of perplexity do not necessarily translate into better recognition accuracy. While two models may have similar perplexities, this fact does not imply that the probability distributions are similar; rather, certain distributions may outperform others during speech recognition decoding since the acoustic confusability and word selection now plays a role. Thus, from a recognition standpoint it would be more suitable to tune LMs with the goal of minimizing WER directly, and not perplexity. The same phenomenon can be seen when we optimize our LMs for BLEU in the machine translation experiments.

To see how context-aware language models work at the sentence level, let us revisit the example sentence from Section 2 and compare the hypothesis from "baseline" and BLEU-optimized DMT. Table 6 presents two hypotheses and the reference sentence (from the source sentence in Figure 1) and one can see that by incorporating context, a more appropriate (i.e. higher n-gram similarity with the reference) hypothesis is generated by the decoder.

Baseline	so you didn't do you know what there is no in your neighborhood
DMT	so you don't know what's going on in your neighborhood
Reference	so you don't know what 's going on in your own neighborhood

Table 6: Comparison of hypotheses generated by "baseline" and BLEU-optimized DMT language models, along with the reference sentence.

Lastly, we present in Table 7 the λ parameter values tuned to minimize perplexity and maximize BLEU. DMT allows us to interpret the λ parameter values as an indicator of the relative importance of a particular context category versus other categories. In this case, we see that optimizing for BLEU has tuned the second topic weight to be insignificant, and the first topic is weighted roughly twice as much as the tone. Optimizing for perplexity gives some weight to the second, less dominant topic, so we can infer that the second topic marker is important in reducing the confusability of the language model, but plays no role in terms of translation quality. The parameter values are thus useful in providing intuition on the role of context markers for a given corpus.

We thus find that it is better to optimize metrics, such

Optimization Criterion	λ_1	λ_2	λ_3
Min Perplexity	0.62	0.16	0.22
Max BLEU	0.65	0.00	0.35

Table 7: Comparison of parameter values for the two optimization criteria. The values show that the second topic is relevant when minimizing perplexity, but does not contribute to improved translation quality

as recognition accuracy or translation quality, that reflect the end-to-end performance of a system rather than an intermediate metric, such as perplexity.

5 Related Work

Several works have addressed the related question of extracting and incorporating meta-data to enhance models, mainly with the aim of minimizing discrepancies between test and training corpora. The idea has been applied to a variety of natural language processing applications, from dialog-act tagging (Sridhar et al., 2009) to disfluency detection (Liu et al., 2003). Model adaptation has also been studied in Statistical Machine Translation, for example Hildebrand et al. (2005) use Information Retrieval to adapt the translation model for an SMT system, and in Zhao et al. (2004) the same technique is applied to the language model. Our framework has the same goal of minimizing test and training data differences, but we aim to minimize these differences by using the context of the test and training sentences, encapsulated in context vectors.

Several frameworks have looked to incorporate topic dependencies in the language model. Iyer and Ostendorf (1996) build a sentence-level scheme, and Florian and Yarowsky (1999) incorporate topic information when backing off to lower n-grams. The methods proposed in these works rely on mixture models, and form the basis for the interpolated LMs approach presented in Section 2.1. Bulyko et al. (2007) discuss several language model adaptation methods applied to machine translation of Arabic broadcast speech, focusing primarily on mixture-based models as well, and Sanchis-Trilles and Cettolo (2010) look at dynamically adapting the mixture weights using an EM-based procedure. Incorporating implicit topics in the form of LDA-based approaches (Hsu and Glass, 2006; Tam and Schultz, 2005) have also been popular.

A similar method to optimize parameter estimation based on sentence-level weights was used by Matsoukas et al. (2009), applied in that case to the translation model. The authors optimize Translation Error Rate in an end-to-end machine translation framework and use the BFGS method to optimize; we hope to extend DMT in the future in a similar manner for the translation and alignment models in speech-to-speech translation.

Additionally, the maximum entropy framework (Rosenfeld, 1996) can also incorporate contextual features, but is not flexible enough to incorporate multiple definitions of the distance metric as we discuss in Section 2.2.

6 Conclusion & Future Work

In this work, we propose a novel framework, Distance-Measure Tuning, to incorporate contextual information at the sentence-level by calculating the distance between two context vectors (the input utterance and the current training corpus utterance in question). This in turn is used to generate a context vector-specific language model for evaluation. While we emphasize that a variety of distance measures can be used, we conducted our experiments with a linear distance measure and compared this approach with an interpolated LM-based approach. DMT was much better at generalizing to unseen context vectors than interpolated LMs, and this aspect resulted in better absolute performance of DMT versus interpolated LMs in terms of both WER and BLEU, and better relative performance (percentage improvement over baseline) in perplexity.

In the near future, various definitions of the distance metric and alternative optimization techniques will also be investigated, as well as additional smoothing techniques.

7 Acknowledgements

This work is supported by DARPA Transformative App program under the contract D11PC20022.

References

Nguyen Bach, Matthias Eck, Paisarn Charoenpornsawat, Thilo Köhler, S. Stüker, T.L. Nguyen, Roger Hsiao, Alex Waibel, Stephan Vogel, Tanja Schultz, and Others. 2007. The CMU TransTac 2007 eyes-free and hands-free two-way speech-to-speech translation system. In *Proc. of the International Workshop on Spoken Language Translation*. Citeseer.

Ivan Bulyko, Spyros Matsoukas, Richard Schwartz, Long Nguyen, and John Makhoul. 2007. Language Model Adaptation in Machine Translation from Speech. In *ICASSP 2007*, pages 117–120, Honolulu, Hawaii.

M. Eck, I. Lane, Y. Zhang, and A. Waibel. 2010. Jibbigo: Speech-to-speech translation on mobile devices. In *Spoken Language Technology Workshop (SLT), 2010 IEEE*, pages 165–166.

Michael Finke, Petra Geutner, Hermann Hild, Thomas Kemp, Klaus Ries, and Martin Westphal. 1997. The karlsruhe-verbmobil speech recognition engine.

Radu Florian and David Yarowsky. 1999. Dynamic nonlocal language modeling via hierarchical topic-based adaptation.

In *Proceedings of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics*, ACL '99, pages 167–174. Association for Computational Linguistics.

Almut Silja Hildebrand, Matthias Eck, Stephan Vogel, and Alex Waibel. 2005. Adaptation of the translation model for statistical machine translation based on information retrieval. In *Proceedings of the 10th EAMT conference "Practical applications of machine translation"*, pages 133–142, Budapest, May.

B.J.P. Hsu and James Glass. 2006. Style & topic language model adaptation using HMM-LDA. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, pages 373–381. Association for Computational Linguistics.

R. Iyer and M. Ostendorf. 1996. Modeling long range dependencies in languages: Topic mixtures vs. dynamic cache models. In *International Conference on Spoken Language Processing*, pages 236–239.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondřej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions*, ACL '07, pages 177–180, Stroudsburg, PA, USA. Association for Computational Linguistics.

Y. Liu, E. Shriberg, and A. Stolcke. 2003. Automatic disfluency identification in conversational speech using multiple knowledge sources. In *Proc. Eurospeech*, pages 957–960, Geneva, Switzerland, September. Association for Computational Linguistics.

Spyros Matsoukas, Antti-Veikko I Rosti, and Bing Zhang. 2009. Discriminative corpus weight estimation for machine translation. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing Volume 2 EMNLP 09*, page 708. Association for Computational Linguistics, August.

J. A. Nelder and R. Mead. 1965. A Simplex Method for Function Minimization. *The Computer Journal*, 7(4):308–313.

Franz Josef Och. 2003. Minimum error rate training in statistical machine translation. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics - Volume 1*, ACL '03, pages 160–167, Stroudsburg, PA, USA. Association for Computational Linguistics.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, pages 311–318, Stroudsburg, PA, USA. Association for Computational Linguistics.

R. Prasad, K. Krstovski, F. Choi, S. Saleem, P. Natarajan, M. Decerbo, and D. Stallard. 2007. Real-time speech-to-speech translation for pdas. In *Portable Information Devices, 2007. PORTABLE07. IEEE International Conference on*, pages 1 –5, May.

Ronald Rosenfeld. 1996. A maximum entropy approach to adaptive statistical language modeling. *Computer, Speech and Language*, 10:187–228.

German Sanchis-Trilles and Mauro Cettolo. 2010. Online Language Model adaptation via N-gram Mixtures for Statistical Machine Translation. In *European Association for Machine Translation*, May.

H. Soltau, F. Metze, F. Fugen, and A. Waibel. 2001. A onepass decoder based on polymorphic linguistic context assignment. In *Proc. ASRU*, pages 214–217.

V.K. Rangarajan Sridhar, S. Bangalore, and S. Narayanan. 2009. Combining lexical, syntactic and prosodic cues for improved dialog act tagging. *Computer Speech & Language*, 23(4):407–422, October.

Andreas Stolcke. 2002. SRILM An Extensible Language Modeling Toolkit. In *Proceedings of the International Conference in Spoken Language Processing*.

Y. C. Tam and T. Schultz. 2005. Language model adaptation using variational Bayes inference. In *Interspeech*.

Zheng-Hua Tan, Brge Lindberg, Yuqing Gao, Bowen Zhou, Weizhong Zhu, and Wei Zhang. 2008. Handheld speech to speech translation system. In *Automatic Speech Recognition on Mobile Devices and over Communication Networks*, Advances in Pattern Recognition, pages 327–346. Springer London.

Ying Zhang and Stephan Vogel. 2006. Suffix array and its applications in empirical natural language processing. Technical Report CMU-LTI-06-010, Language Technologies Institute, School of Computer Science, Carnegie Mellon University, Pittsburgh, PA, Dec.

Ying Zhang and Stephan Vogel. 2007. Pandora: A large-scale two-way statistical machine translation system for hand-held devices. In *Proceedings of MT Summit XI*, Copenhagen, Denmark, September.

Ying Zhang. 2009. Virtual babel: Towards context-aware machine translation in virtual worlds. In *Proceedings of MT Summit XII*, Ottawa, Ontario, Canada, August 26-30.

Bing Zhao, Matthias Eck, and Stephan Vogel. 2004. 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. COLING.