EXPLORING THE USE OF ACOUSTIC EMBEDDINGS IN NEURAL MACHINE TRANSLATION

Salil Deena[1]*, Raymond W. M. Ng*[1]*, Pranava Madhyastha*[2]*, Lucia Specia*[2] *and Thomas Hain*[1]

[1]Speech and Hearing Research Group, The University of Sheffield, UK
[2]Natural Language Processing Research Group, The University of Sheffield, UK

ABSTRACT

Neural Machine Translation (NMT) has recently demonstrated improved performance over statistical machine translation and relies on an encoder-decoder framework for translating text from source to target. The structure of NMT makes it amenable to add auxiliary features, which can provide complementary information to that present in the source text. In this paper, auxiliary features derived from accompanying audio, are investigated for NMT and are compared and combined with text-derived features. These acoustic embeddings can help resolve ambiguity in the translation, thus improving the output. The following features are experimented with: Latent Dirichlet Allocation (LDA) topic vectors and GMM subspace i-vectors derived from audio. These are contrasted against: skip-gram/Word2Vec features and LDA features derived from text. The results are encouraging and show that acoustic information does help with NMT, leading to an overall 3.3% relative improvement in BLEU scores.

Index Terms**—** Neural Machine Translation, LDA topics, Acoustic Embeddings

1. INTRODUCTION

In Neural Machine Translation (NMT) [1], text from a source language is first encoded using a recurrent neural network (RNN), resulting in compressed context vector, which is then passed to the decoder, also a RNN, and takes the encoded context vector and the previously translated word as input and produces the target translated word at the current time step. The compressed context vector is derived by applying an attention mechanism [1], which is a measure of alignment between the source and target text, to the RNN hidden state vectors of the encoder up to the current time-step.

Auxiliary features can be integrated at the encoder by concatenating the word vectors with features [2, 3, 4]. Linguistic input features such as lemmas were found to improve NMT results when they are appended to the word vector at the encoder [2] or even when added as an extra output at the decoder [3]. In [4], latent Dirichlet allocation (LDA) [5] topic vectors were appended to the hidden state vector for each word and subsequently used to obtain a topic-informed encoder context vector, which is then passed to the decoder. In [6] domain information was incorporated by concatenating a 3-letter word representing the domain to the source text.

Auxiliary information in the form of multi-modal streams such as images have also been integrated in NMT [7, 8, 9, 10]. In most cases, the visual features are extracted from a convolutional neural network (CNN) and can be appended to the head or tail of the original text sequence in the encoder [7], added to the word embeddings after a linear projection to match the word feature dimensionality [8] or even encoded separately with a separate attention mechanism as in [9, 10].

This work focuses on the integration of auxiliary features extracted from audio accompanying the text. Whilst features extracted from text and images have been explored, the use of audio information for NMT remains an open question. In this work, audio features in the form of show-level i-vectors [11] and Latent Dirichlet Allocation (LDA) topic vectors extracted from audio (acoustic LDA) [12] are explored for machine translation (MT) of source text. These auxiliary features are compared and combined with show-level LDA topic vectors derived from text [5] as well as word embeddings that preserve distance of similar words in vector space [13]. The combination of features at different levels of granularity (show-level and word-level) is also investigated.

The aims of this work are thus three-fold. First, the use of audio as an extra stream of information is investigated within the framework of NMT. Second, we aim to investigate whether ways of structuring this diversity through topic modelling on both the text and acoustic data can help improve translation results. Third, we look at whether semantically motivated text embeddings can help with NMT when used as an auxiliary feature on top of the default word representation and in combination with other embeddings. Evaluation is carried out on a English to French MT task on public TED lecture data based on the IWSLT 2015 evaluation [14], which consists of TED talks/shows with accompanying audio. A key characteristic of the TED data is that it has diversity in the variety of topics that are spoken and multimedia information in the form of text, audio and video are available, even though visual information is not considered within the scope of this work.

2. BACKGROUND

2.1. Neural Machine Translation

Neural Machine Translation allows the decoder to predict a word sequence $\mathbf{y}_1, \ldots, \mathbf{y}_T$ in the target language from the previous word, the RNN hidden state $\mathbf{s}_i$ and the compressed context vector from the encoder $\mathbf{c}$:

$$p(\mathbf{y}_i|\mathbf{y}_1, \ldots, \mathbf{y}_{i-1}, \mathbf{c}) = g(\mathbf{y}_{i-1}, \mathbf{s}_i, \mathbf{c}), \tag{1}$$

The context vector is computed by fusing the past RNN hidden states, $\{\mathbf{h}_j\}_{j=1}^{T}$ using an attention mechanism.

$$\mathbf{c} = \sum_{j=1}^{T} \alpha_{ij} \mathbf{h}_j \tag{2}$$

Where α_{ij} is called the attention weights and is computed as follows:

$$\alpha_{ij} = \frac{\exp{(e_{ij})}}{\sum_{k=1}^{T} \exp{(e_{ik})}}, \tag{3}$$

Where $e_{ij} = a(\mathbf{s}_{i-1}, \mathbf{h}_j)$ is a measure of the alignment between the RNN hidden state at position $i-1$ and the decoder state at position j.

The encoder is a standard RNN that predicts the current word in the source language, $\mathbf{x}_i$ from the previous word $\mathbf{x}_{i-1}$ and the RNN hidden state vector at the current time, $\mathbf{h}_i$.

$$p(\mathbf{x}_i|\mathbf{x}_1, \ldots, \mathbf{x}_{i-1}, \mathbf{h}_i) = f(\mathbf{x}_{i-1}, \mathbf{h}_i) \tag{4}$$

2.2. NMT Augmented with Auxiliary Features

Auxiliary features can be integrated into Recurrent Network Networks (RNN) in various ways. They can be added either at the input layer, hidden layer or output layer through concatenation, addition or composition. A detailed review of feature integration into Recurrent Neural Network Language Models (RNNLM) is presented in [15].

In NMT, auxiliary features can be integrated at the encoder through a simple weighted concatenation in the way proposed by Sennrich *et al.* [2] as follows:

$$\mathbf{h}_i = \sigma(\mathbf{W}\mathbf{E}\mathbf{x}_i + \mathbf{U}\mathbf{h}_{i-1}) \tag{5}$$

Where $\mathbf{W}$ is a word embedding matrix, $\mathbf{W}$ and $\mathbf{U}$ are weight matrices and σ is a non-linear activation function such as sigmoid or tanh.

This assumes that the features are aligned for each word/token, thus requiring a one-to-one mapping between the features and words. In our case, we are interested in the integration of both word-level and sentence-level features. As a result of this, a robust way of concatenating different kinds of information extracted on both word and the sentence levels, through asynchronous fusion, has been proposed in [16].

Assume k_t are word-level and k_s are sentence-level features and taking $\phi(w_i)$ to be the NMT word embedding for word w_i. The concatenation method produces a new word embedding $\phi(\bar{w}_i)$ according to the following:

$$\phi(\bar{w}_i) = \sigma(\phi(w_i) + \mathbf{W}_{k_t}\phi_{k_t}(w_i) + \mathbf{W}_{k_s}\phi_{k_s}(w_i)) \tag{6}$$

Where $\phi(\bar{w}_i)$ is the resultant word information that is composed of $\phi(w_i)$ – the word embedding vector; $\phi_{k_t}(w_i)$ – the token-level external information; and $\phi_{k_s}(w_i)$ – the sentence-level external information. $\mathbf{W}_{k_t}$ and $\mathbf{W}_{k_s}$ are affine transforms that help both to account for the difference in dimensionality between the word embedding vector and the features and they are learnt during the NMT training process, where the aim is to weight and balance the contributions of different features in order to give an optimal translation result.

Document-level features can also be integrated by replicating them at the sentence-level, because in NMT, each sentence is independent and there is no propagation of states across sentences.

This approach has the advantage that features can be composed at multiple levels, which can be useful for disambiguating translation as it is known that the result NMT can be improved by attending to both local and global context [17].

In the next sections, the text and acoustic auxiliary features used in this work are described.

3. TEXT FEATURES FOR NMT

3.1. Word2Vec

Word2Vec [13] is a distributed representation of words in a vector space and allows semantically similar words to be mapped close in the vector space. Word2Vec is a class of neural networks that can produce a vector for each word in the corpus given an unlabelled training corpus, where the word vectors encode semantic information. There are two main Word2Vec models, the skip-gram model and the continuous bag-of-words model. In this work, the skip-gram model is used and this is now further explained.

Skip-gram is based on a neural network model that is trained by feeding it word pairs found in training documents. The network is then going to learn the statistics from the number of times each pairing shows up. The words are represented as a 1-of-K encoding (1-hot vector) when used as input or output to the neural network. The input of the skip-gram neural network model is a single word W_I from a given sentence and the output are the words in W_I's context, $\{W_{O,1}, \ldots, W_{O,c}\}$ and defined by the word window size c.

More formally, given a sequence of words $\{w_t\}_{t=1}^{T}$, the objective of the Skip-gram model is to maximise the average log probability:

$$L = \frac{1}{T} \sum_{t=1}^{T} \sum_{-c \leq j \leq c, j \neq 0} \log p(w_{t+j}|w_t) \tag{7}$$

Where where c is the size of the training context and the Skip-gram formulation defines $p(w_{t+j}|w_t)$ according to:

$$p(w_j|W_I) = \frac{\exp\left({v'_{w_j}}^T v_{w_I}\right)}{\sum_{j'=1}^{N} \exp\left({v'_{w'_j}}^T v_{w_I}\right)} \tag{8}$$

Where v_w and v'_w are the input and output vector representations of w_j, and N is the number of words in the vocabulary. The model is trained using back-propagation and the final value of w_j is taken as the Word2Vec vector of the word.

Both monolingual and bilingual skip-grams were found to lead to small but non-significant improvements in BLEU score for English to Spanish statistical machine translation on the News Commentary corpus [18]. In this work, we aim to investigate the same on English to French translation on TED Talks using the NMT framework. Moreover, the composition of Word2Vec (token-level) features and show-level features derived from audio and text, are investigated.

3.2. Show-based Text LDA

Text-based LDA [5], referred to in this paper at text LDA (tLDA), is an unsupervised probabilistic generative model that allows text data to be represented by a set of unobserved topics. It aims to describe how every item within a collection is generated, assuming that there are a set of latent variables and that each item is modelled as a finite mixture over those latent variables. It can be used to extract show-level topic information, which can help disambiguate the context of translation. LDA features can be obtained by first extracting term frequency-inverse document frequency (TF-IDF) vectors that are computed for each document of the training text data, which are then used to train LDA models. LDA features are then obtained by computing Dirichlet posteriors over the topics for each document, where a document corresponds to a specific show in the case of the TED data.

A dataset is defined as a collection of documents where each document is in turn a collection of discrete symbols (in case of topic modelling of text documents, a document is equivalent to a set and words inside a document are equivalent to the discrete symbols). Each document is represented by a V-dimensional vector based on the histogram of the symbols' table which has size of V. It is assumed that the documents were generated by the following generative process:

1. For each document $d_m, m \in \{1...M\}$, choose a K–dimensional latent variable weight vector θ_m from the Dirichlet distribution with scaling parameter α: $p(\theta_m|\alpha) = Dir(\alpha)$
2. For each discrete item $w_n, n \in \{1...N\}$ in document d_m
   (a) Draw a latent variable $z_n \in \{1...K\}$ from the multinomial distribution $p(z_n = k|\theta_m)$

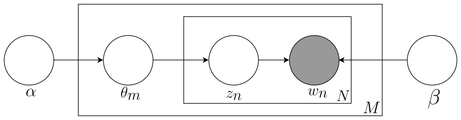

Fig. 1. Graphical model representation of LDA

   (b) Given the latent variable, draw a symbol from $p(w_n|z_n, \beta)$, where β is a $V \times K$ matrix and $\beta_{ij} = p(w_n = i|z_n = j, \beta)$

It is assumed that each document can be represented as a bag–of–symbols - i.e. by first–order statistics, which means any symbol sequence relationship is disregarded. Since speech and text are highly ordered processes this can be an issue. Another assumption is that the dimensionality of the Dirichlet distribution K is fixed and known (and thus the dimensionality of the latent variable z).

A graphical representation of the LDA model is shown at Figure 1 as a three–level hierarchical Bayesian model. In this model, the only observed variable is w and the rest are all latent. α and β are dataset level parameters, θ_m is a document level variable and z_n, w_n are symbol level variables. The generative process is described formally as:

$$p(\theta, \mathbf{z}, \mathbf{w}|\alpha, \beta) = p(\theta|\alpha) \prod_{n=1}^{N} p(z_n|\theta) p(w_n|z_n, \beta) \tag{9}$$

The posterior distribution of the latent variables given the symbols and α and β parameters is:

$$p(\theta, \mathbf{z}|\mathbf{w}, \alpha, \beta) = \frac{p(\theta, \mathbf{z}, \mathbf{w}|\alpha, \beta)}{p(\mathbf{w}|\alpha, \beta)} \tag{10}$$

Computing $p(\mathbf{w}|\alpha, \beta)$ requires some intractable integrals. A reasonable approximate can be acquired using variational approximation, which is shown to work reasonably well in various applications [5]. The approximated posterior distribution is:

$$q(\theta, \mathbf{z}|\gamma, \phi) = q(\theta|\gamma) \prod_{n=1}^{N} q(z_n|\phi_n) \tag{11}$$

where γ is the Dirichlet parameter that determines θ and ϕ is the parameter for the multinomial that generates the latent variables.

Training tries to minimise the Kullback–Leiber Divergence (KLD) between the real and the approximated joint probabilities (equations 10 and 11) [5]:

$$\underset{\gamma,\phi}{argmin}\ KLD\big(q(\theta, \mathbf{z}|\gamma, \phi) \ || \ p(\theta, \mathbf{z}|\mathbf{w}, \alpha, \beta)\big) \tag{12}$$

Compared to the Word2Vec representation which is at word level, the LDA features are extracted at show level and thus provides a complementary source of information, where global context is taken into account. Text LDA features have been shown to be useful both for SMT [19] and NMT [4], where LDA topic vectors were included both at the encoder and decoder. Whilst both [19, 4] investigated Chinese to English translation, this work investigates LDA for English to French translation, on TED Talks, which are diverse in topics spoken. LDA can help to structure this diversity and thus provide global context to help disambiguate translation.

4. ACOUSTIC FEATURES FOR NMT

4.1. Show-based Acoustic LDA

Acoustic LDA (aLDA) is a specific form of acoustic embedding and represents an acoustic signal as a distribution of latent topics, which can embody information such as speaking style, genre, as well as linguistic information and can thus help disambiguate machine translation. Typically speech is represented using continuous features such as Mel frequency cepstral coefficients (MFCCs), and has variable length. In order to extract acoustic LDA, vector quantisation needs to be performed to represent the speech signal as a sequence of discrete symbols. This is done using the same method described in [20], where a GMM model with V components is trained using all of the training data. The model is then used to get the posterior probabilities of the Gaussian components to represent each frame with index of the Gaussian component with the highest posterior probability. Frames of every speech segment of length T, $\mathbf{X} = \{\mathbf{x}_t\}_{t=1}^{T}$ are represented as:

$$\tilde{x}_t = \underset{i}{argmax}\ P(G_i|\mathbf{x}_t) \tag{13}$$

where G_i $(1 \leq i \leq V)$ is the i^{th} Gaussian component. After applying this process to each utterance, each speech segment is represented as $\{\tilde{x}_t\}_{t=1}^{T}$ where $\tilde{x}_t$ is index of the Gaussian component and thus a natural number $(1 \leq \tilde{x}_t \leq V)$. Here we refer to each speech utterance as an acoustic document.

With this information, a fixed length vector $\hat{a} = \{\hat{x}_t\}_{t=1}^{T}$ of size V is constructed to represent the count of every Gaussian component in an acoustic document.

This leads to a type of bag-of-sounds representation. The sounds would normally be expected to relate to phones, however given the acoustic diversity of background conditions many other factors may play a role. Once these bag-of-sounds representations of acoustic documents are derived, LDA models can be trained. After training the LDA acoustic model, a similar procedure is followed to extract acoustic LDA features from test data.

Acoustic LDA has been found to be useful for unsupervised latent domain discovery in automatic speech recognition [12], where the discovered domains where then used for maximum-a-posteriori (MAP) domain adaptation. The aim in this work is to investigate whether acoustic LDA extracted at the show level, can have similar value in helping disambiguate machine translation.

4.2. Show-based i-Vectors

I-vectors are motivated by Joint Factor Analysis [21], and were originally proposed in the context of speaker recognition [11]. An i-vector represents the specific characteristics of the audio as a point in total variability space.

MFCC vectors are extracted from audio files and show-dependent Gaussian Mixture Models (GMM) are trained on the audio features, which make up a Universal Background Model (UBM). The mean vectors of all Gaussian Mixture Models (GMMs) in this UBM are concatenated into a super-vector $\boldsymbol{\mu}_0$. Correspondingly, a set of show-dependent GMMs is derived for each show, and its mean vectors are concatenated into a show dependent super-vector, i.e. $\boldsymbol{\mu}^s$ for show s. The total variability matrix $\mathbf{M}$ spans the bases with highest variability in the mean super-vector space according to the following.

$$\boldsymbol{\mu}^s = \boldsymbol{\mu}_0 + \mathbf{M}\boldsymbol{\lambda}^s. \tag{14}$$

where $\boldsymbol{\lambda}^s$ represents the i-vector for show s.

Show-based i-vectors are also an unsupervised audio embedding just like acoustic LDA. However, the key difference is that acoustic LDA is based on a topic model built on a vector-quantisation of the audio data, whilst i-vectors are a subspace representation of the audio at a show level. These two representations embed different types of information from the audio at a show level with acoustic LDA providing a characterisation of genre and speaking-style, whilst the i-vector would capture salient features for each show, including the speaker characteristics, accents, etc.

5. EXPERIMENTS AND RESULTS

5.1. Data

The data used in this work is the IWSLT 2015 TED Talks [14]. Training data conforms to the IWSLT 2015 evaluation criteria for both the ASR and MT task, with 1711 talks consisting of parallel English and French talks. As acoustic data is used to extract auxiliary features in this work, the training data was filtered to retain only 1622 talks where the corresponding multimedia clips can be crawled from `TED.com`. This data set is referred to **TEDtrain**. **TEDdev** was extracted from IWSLT 2010 (dev+test) data and was used to provide stopping criterion in NMT training. **TEDeval** was from IWSLT 2012 (test) data. The statistics of the TED data are given in Table 1.

5.2. Experimental Setup

We use the standard LSTM-based bidirectional encoder-decoder architecture with global attention [17]. All our NMT models have the following architecture: the input and output vocabulary are limited to words that appear at least three times in the training data and the remaining words are replaced by

	Snt	Types	Tokens	Avg. Length
(**TEDtrain**)				
English	201,719	58.0k	3.512M	17.4
French		74.6k	3.680M	18.2
(**TEDdev**)				
English	2,551	5.5k	44.2k	17.4
French		6.7k	44.8k	17.6
(**TEDeval**)				
English	1,124	2.9k	18.5k	16.5
French		3.5k	20.0k	17.8

Table 1. Statistics of TED data

the $< UNK >$ token. The hidden layer dimensionality is set to 256 and the word dimensionality is set to 128, for both the encoder and decoder, as this configuration was found to lead to faster training times without sacrificing translation performance. At decoding time, the topmost probable word at each time step, is computed.

Concerning the auxiliary features, both 50 and 100 dimensional vectors were extracted for i-vectors, acoustic and text LDA. 300-dimensional Word2Vec embeddings were extracted after training on the Google news corpus[1].

5.3. Results

Table 2 show the results obtained using each auxiliary feature on the dev and test sets in terms of BLEU [22] and METEOR [23] scores. "Baseline" corresponds to a standard NMT model trained without any additional features.

Model	**TEDdev**		**TEDeval**	
	BLEU	METEOR	BLEU	METEOR
Baseline	30.38	**0.6158**	36.02	0.6485
Word2Vec (300d)	**30.44**	0.6116	35.89	0.6424
i-vector (50d)	29.97	0.6118	35.87	0.6455
i-vector (100d)	29.77	0.6065	36.14	0.6428
tLDA (50d)	30.12	0.6092	36.09	0.6432
tLDA (100d)	30.12	0.6126	36.14	0.6449
aLDA (50d)	30.32	0.6118	36.11	**0.6506**
aLDA (100d)	29.93	0.6125	**36.51**	0.6474

Table 2. BLEU and METEOR scores on TED data in NMT setting

These results shows that when used independently, the Word2Vec features give the best BLEU score on the dev set whilst the 100-dimensional acoustic LDA gives the best result on the eval set. The results of text and acoustic LDA vary across the dev and test sets with both the 50-dim and the 100-dim acoustic LDA slightly outperforming the baseline in terms of BLEU score on the eval set but not on the dev set. Both the 50-dim and 100-dim text LDA slightly outperforms the baseline on the dev set but not on the eval set. The Word2Vec feature gives the best result in terms of BLEU score on the TEDDev data but not on TEDEval.

Table 3 shows the results of composing word-level features (Word2Vec) with show-level features (text&acoustic LDA, i-vectors) according to Eqn. 6.

Model	**TEDdev**		**TEDeval**	
	BLEU	METEOR	BLEU	METEOR
Baseline	30.38	**0.6158**	36.02	0.6485
Word2Vec (300d)	30.44	0.6116	35.89	0.6424
Word2Vec+i-vector (50d)	30.09	0.6146	36.15	0.6499
Word2Vec+i-vector (100d)	30.20	0.6105	36.73	0.6524
Word2Vec+tLDA (50d)	30.38	0.6128	36.23	0.6482
Word2Vec+tLDA (100d)	**30.57**	0.6123	36.27	0.6479
Word2Vec+aLDA (50d)	30.50	0.6087	36.04	0.6463
Word2Vec+aLDA (100d)	30.16	0.6140	**37.21**	**0.6525**

Table 3. BLEU and METEOR scores on TED data in compositional NMT setting

A different pattern is observed when composing the Word2Vec token-level embeddings with the show-level features, with 100-dimensional acoustic LDA giving the best results on the eval set with a BLEU score of 37.21, representing a relative improvement of 3.3% over the baseline result of 36.02. However, the 100-dim text LDA gives better results than other embeddings on the dev set but only narrowly outperforming the 50-dimensional acoustic LDA. The i-vectors have a different behaviour in the compositional setting with both the 50-dim and the 100-dim i-vectors leading to improvements in terms of BLEU scores, over both the baseline and the Word2Vec-only result, on both the TEDDev and the TEDEval data.

The results from the i-vector experiments suggest that some features can be complementary with used in the compositional setting. Whilst the i-vectors do not seem useful on their own, they lead to gains when used in composition with Word2Vec. The results also indicate that both text-based and acoustic-based topic information from LDA help to disambiguate translation and lead to improved results and so do word embeddings that preserve distance in vector space. In order to better understand these results, some further analysis has been carried out on the outputs of the translation so as to better understand under which conditions the features help.

5.4. Further Analysis of Results

In this section, we aim to take a closer look at particular TED Talks shows where each of the features perform best, based on per-sentence METEOR scores. Table 4 illustrates the shows in the TEDEval data.

It can be seen in table 3, that the METEOR scores are most highly correlated to the BLEU scores for the TEDEval data. As a result of this, we compute per-show METEOR scores for TEDEval for the compositional case and the results are given in Table 5.

[1] `https://code.google.com/archive/p/Word2Vec/`

TED Show	Title	Keywords
1	Jack Choi: On the virtual dissection table	education, health care, interface design, medical research, technology
2	Frank Warren: Half a million secrets	arts, creativity, design, memory, storytelling
3	Lucy McRae: How can technology transform the human body?	architecture, design, technology
4	Drew Curtis: How I beat a patent troll	business, entrepreneur, law
5	Frans de Waal: Moral behavior in animals	engineering, animals, community, morality, science
6	Tal Golesworthy: How I repaired my own heart	engineering, health, innovation, medicine, science, technology
7	Sherry Turkle: Connected, but alone?	communication, community, culture, technology
8	Atul Gawande: How do we heal medicine?	health care, medicine
9	Laura Carstensen: Older people are happier	aging, culture, science
10	Michael Norton: How to buy happiness	business, community, money, philanthropy, psychology, shopping
11	Christina Warinner: Tracking ancient diseases using ... plaque	evolution, medicine, paleontology, science

Table 4. TED Shows for TEDEval

TED Show	baseline	Word2Vec	Word2Vec+ i-vector(50)	Word2Vec+ +i-vector(100)	Word2Vec+ acoustic lda(50)	Word2Vec+ acoustic lda(100)	Word2Vec+ text lda(50)	Word2Vec+ +text lda(100)
1	0.6446	0.6372	0.6325	0.6432	0.6458	0.6479	0.6426	**0.6484**
2	0.6300	0.6286	**0.6416**	0.6396	0.6283	0.6371	0.6361	0.6296
3	0.6708	0.6657	0.6610	0.6674	0.6816	0.6692	**0.6828**	0.6527
4	0.6585	0.6315	0.6518	**0.6705**	0.6419	0.6529	0.6476	0.6582
5	0.6843	0.6566	0.6784	0.6715	0.6681	**0.6841**	0.6699	0.6715
6	0.6978	**0.6986**	0.6890	0.6926	0.6885	0.6976	0.6943	0.6867
7	0.6886	0.6820	0.6967	**0.7013**	0.6887	0.6767	0.6846	0.6729
8	0.7046	0.7078	**0.7119**	0.7178	0.7045	0.7056	0.7163	0.6941
9	0.6441	0.6803	0.6834	**0.6959**	0.6892	0.6884	0.6874	0.6822
10	0.6025	0.5928	**0.6186**	0.6002	0.6116	0.6077	0.5992	0.5937
11	0.6985	0.6954	0.7056	0.7042	0.6972	0.6997	0.6949	**0.7070**

Table 5. TED Talk Show-Specific METEOR scores for TEDEval

The results seem to indicate that different features lead to improvements on individual shows in a variable manner. Whilst the Word2Vec features alone do not seem to lead to an improvement for most shows, they lead to different behaviours when used with acoustic features with both the 50 and 100-dimensional i-vectors leading to improvements in BLEU score but only the 100-dimensional acoustic LDA giving consistently good results across all shows. However, this pattern is not observed for 50 and 100 dimensional text LDA with the 50-dimensional text LDA outperforming the 100-dimensional LDA features in most cases. One possible reason for this could be that an increase in number of text LDA topics could lead to higher sparsity, especially when the diversity of topics in the TED Talk is not very high. For example TED shows 2, 3, 4, 8 and 9 have fewer keywords according to Table 4 and thus more focussed in terms of topics. These shows also have a lower performance when using 100-dimensional compared to when using 50-dimensional text LDA features. In constrast, TED show 1 has the highest number of keywords and also gives the highest score with 100-dimensional text LDA. However, this generalisation does not apply for all shows and therefore, further investigation is needed.

Also, it is clear from Table 4 that different shows respond differently to acoustic embeddings. For example, some shows that give very good performance with i-vectors perform less well with acoustic LDA and vice-versa and the same is true for text-based features like Word2Vec and text LDA. This suggests that the different features are complementary and can lead to improvements if used in composition with each other.

6. CONCLUSIONS

This paper has investigated Neural Machine Translation augmented with auxiliary features, where the features are derived from accompanying audio and have been both composed and contrasted with text-based features. Both word-level and show-level embeddings have been explored. Acoustic embeddings like acoustic LDA show promise when used as a single auxiliary feature and so do semantically-motivated word embeddings. It was shown a composition of the acoustic features with word embeddings that preserve similarity in vector space, leads to further improvements of the results. Further analysis of the results also showed that different shows respond differently to text and acoustic features, thus highlighting their complementary nature.

In future work, we will further investigate the composition of features in different settings in order to better understand the type of complementary information they bring and how these can be leveraged effectively in NMT systems. Moreover, we will also investigate the use of different types of acoustic embeddings, such as those derived from siamese networks [24], that try to preserve distance of words both semantically and in acoustic space.

7. REFERENCES

[1] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio, "Neural machine translation by jointly learning to align and translate," in *ICLR'15: Proc. of the International Conference on Learning Representations*, 2015.

[2] Rico Sennrich and Barry Haddow, "Linguistic input features improve neural machine translation," in *WMT'16: Proceedings of the First Conference on Machine Translation*, 2016, pp. 83–91.

[3] Mercedes Garca Martnez, Loc Barrault, and Fethi Bougares, "Factored neural machine translation architectures," in *IWSTL'16: Proc. of International Workshop on Spoken Language Translation*, 2016.

[4] Jian Zhang, Liangyou Li, Andy Way, and Qun Liu, "Topic-informed neural machine translation," in *COLING'16: Proc. of the 26th International Conference on Computational Linguistics*, 2016, pp. 1807–1817.

[5] David M. Blei, Andrew Y. Ng, and Michael I. Jordan, "Latent Dirichlet allocation," *J. Mach. Learn. Res.*, vol. 3, pp. 993–1022, 2003.

[6] Catherine Kobus, Josep Maria Crego, and Jean Senellart, "Domain control for neural machine translation," *CoRR*, vol. abs/1612.06140, 2016.

[7] Po-Yao Huang, Frederick Liu, Sz-Rung Shiang, Jean Oh, and Chris Dyer, "Attention-based multimodal neural machine translation," in *WMT'16: Proceedings of the First Conference on Machine Translation*, 2016, pp. 639–645.

[8] Ozan Caglayan, Loïc Barrault, and Fethi Bougares, "Multimodal attention for neural machine translation," *CoRR*, vol. abs/1609.03976, 2016.

[9] Iacer Calixto, Qun Liu, and Nick Campbell, "Doubly-attentive decoder for multi-modal neural machine translation," *CoRR*, vol. abs/1702.01287, 2017.

[10] Iacer Calixto, Qun Liu, and Nick Campbell, "Incorporating global visual features into attention-based neural machine translation," *CoRR*, vol. abs/1701.06521, 2017.

[11] Najim Dehak, Patrick J Kenny, Réda Dehak, Pierre Dumouchel, and Pierre Ouellet, "Front-end factor analysis for speaker verification," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 4, pp. 788–798, 2011.

[12] Mortaza Doulaty, Oscar Saz, and Thomas Hain, "Unsupervised domain discovery using latent dirichlet allocation for acoustic modelling in speech recognition," in *Proc. of Interspeech*, Dresden, Germany, 2015.

[13] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean, "Efficient estimation of word representations in vector space," in *ICLR Workshop*, 2013.

[14] IWSLT 2015, "Web inventory of transcribed and translated talks," Jan 2015.

[15] Salil Deena, Raymond W. M. Ng, Pranava Madhyastha, Lucia Specia, and Thomas Hain, "Semi-supervised adaptation of RNNLMs by fine-tuning with domain-specific auxiliary features," in *INTERSPEECH'17: Proc. of ISCA Interspeech*, 2017.

[16] Pranava Swaroop Madhyastha, Salil Deena, Raymond W. M. Ng, and Lucia Specia, "Learning to compose features for neural machine translation," *IJCNLP'17 : International Joint Conference on Natural Language Processing*, vol. (Submitted), 2017.

[17] Thang Luong, Hieu Pham, and Christopher D. Manning, "Effective approaches to attention-based neural machine translation," in *EMNLP'15: Proc. of the 2015 Conference on Empirical Methods in Natural Language Processing*, 2015, pp. 1412–1421.

[18] Eva Martínez Garcia, Carles Creus, Cristina España-Bonet, and Lluís Màrquez, "Using word embeddings to enforce document-level lexical consistency in machine translation," *The Prague Bulletin of Mathematical Linguistics*, , no. 108, pp. 8596, 2017.

[19] Jinsong Su, Deyi Xiong, Yang Liu, Xianpei Han, Hongyu Lin, Junfeng Yao, and Min Zhang, "A context-aware topic model for statistical machine translation," in *Proceedings of the 53rd Annual Meeting of ACL and the 7th International Joint Conference on NLP of the Asian Federation of NLP*, 2015, pp. 229–238.

[20] Chongjia Ni, Cheung Chi Leung, Lei Wang, Nancy F. Chen, and Bin Ma, "Unsupervised data selection and word–morph mixed language model for tamil low-resource keyword search," in *Proc. of ICASSP*, 2015.

[21] Patrick Kenny, Gilles Boulianne, Pierre Ouellet, and Pierre Dumouchel, "Joint Factor Analysis Versus Eigenchannels in Speaker Recognition," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 4, pp. 1435–1447, 2007.

[22] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu, "BLEU: a method for automatic evaluation of machine translation," in *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, 2002, pp. 311–318.

[23] Satanjeev Banerjee and Alon Lavie, "METEOR: An automatic metric for MT evaluation with improved correlation with human judgments," in *Proceedings of the*

ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization. 2005, pp. 65–72, Association for Computational Linguistics.

[24] Herman Kamper, Weiran Wang, and Karen Livescu, “Deep convolutional acoustic word embeddings using word-pair side information,” in *ICASSP’16: Proc. of the IEEE International Conference on Acoustics, Speech and Signal Processing*, 2016, pp. 4950–4954.