INTERSPEECH 2019
September 15–19, 2019, Graz, Austria

Automatic Compression of Subtitles with Neural Networks and its Effect on User Experience

Katrin Angerbauer[1,2], *Heike Adel*[1,3], *Ngoc Thang Vu*[1]

[1]Institute for Natural Language Processing (IMS), University of Stuttgart, Germany
[2]Visualisation Research Centre (VISUS), University of Stuttgart, Germany
[3]Bosch Center for Artificial Intelligence (BCAI), Renningen, Germany
katrin.angerbauer@visus.uni-stuttgart.de, thang.vu@ims.uni-stuttgart.de

Abstract

Understanding spoken language can be impeded through factors like noisy environments, hearing impairments or lack of proficiency. Subtitles can help in those cases. However, for fast speech or limited screen size, it might be advantageous to compress the subtitles to their most relevant content. Therefore, we address automatic sentence compression in this paper. We propose a neural network model based on an encoder-decoder approach with the possibility of integrating the desired compression ratio. Using this model, we conduct a user study to investigate the effects of compressed subtitles on user experience. Our results show that compressed subtitles can suffice for comprehension but may pose additional cognitive load.
Index Terms: automatic sentence compression, subtitles, recurrent neural networks, user study

1. Introduction

Spoken language is ubiquitous in our daily lives. However, numerous factors, such as background noise, lack of proficiency in a particular language, or difficulties with hearing, can impede the understanding of it [1, 2]. In all those settings, subtitles can improve the accessibility of spoken content by adding a visual channel to the oral information [3]. According to Zanón [4] subtitles are a *"dynamic and rich source of communicative language use"*. Unfortunately, reading subtitles can be cumbersome, as we generally speak faster than we read [5]. Therefore, subtitles are often edited to enable a more comfortable reading. Manual editing, however, is expensive since it is time-consuming and requires specific training. Thus, automatic systems for compressing the spoken content are desired. Such systems could be used in various places, such as foreign language courses [6, 7], lectures, talks or meetings.

Previous studies on simplifying subtitles replace words with synonyms that are more frequent or increase the cohesion between words [8]. Others reduce subtitles to keyword captions [6, 9]. In contrast, we approach the simplification of subtitles as a sentence compression problem [10] and learn automatic models to only keep the most relevant parts of the subtitles. The goal of sentence compression is the creation of a grammatically correct extractive summary of the most relevant information [11, 12]. Several methods have been proposed for automatic sentence compression. Examples are tree-based methods [11, 13], optimization approaches [14] and neural methods [15, 16]. Filippova et al. [15] treat sentence compression as a sequence labelling task, where each word in the sentence is labeled as KEEP (1) or DROP (0), and use stacked long short-term memory networks (LSTMs) for that task. Similar to other studies [16, 17, 18, 19], we use a baseline that builds upon their model. Other work on sentence compression investigates the benefits of multi-task learning [16], bi-directional LSTMs [17], attention [17] and encoder-decoder architectures [18, 19]. We also propose a model with an encoder-decoder structure but in difference to [18, 19] who use two encoders, we use only one encoder and enhance it with linguistic features, namely part-of-speech tags. Moreover, our model allows to specify a desired compression ratio to control how many words of the original sentence should be kept after compression. This is similar to the idea of Kikuchi et al. [20]. However, we propose a different approach and, to the best of our knowledge, we are the first to investigate the effects of controlling the output length in the context of sentence compression.

Because automatic compression of spoken language is relevant for daily life applications, it is important to evaluate not only the performance on datasets of limited size but also the applicability to real-life settings. Thus, additional to an evaluation on a benchmark dataset for comparison to state of the art, we assess the effect of compressed subtitles on comprehension and cognitive load in a user study. Related user studies regarding partial and keyword captions provide controversial results: Guillory [21], Rooney [22] and Mirzaei et al. [7] find positive effects of partial captions and no significant differences in comprehension compared to full captions. They argue based on the dual-coding theory that as they lower the input on the visual channel, the cognitive load has to be smaller. Others, however, report worse results with keyword captions compared to full captions or no captions due to confusion and distraction [23, 24, 25]. These results stress the need of additional user studies, as mere technical evaluations cannot provide enough insights on how the chosen technological methods affect the user. This is why we also enrich the evaluation of our paper with a user study on how manual and automatic subtitles are perceived.

To sum up, our contributions are: (i) We propose an adapted recurrent encoder-decoder model for sentence compression which takes the desired compression ratio into account. (ii) We perform a user study on the effectiveness of subtitles which shows the potential of compressed subtitles. Our results can be used as a starting point for further investigations of sentence compression for spoken language and contribute to future research to make spoken language more accessible.

2. Neural networks for sentence compression

Recent approaches for sentence compression treat the task as a sequence labeling problem [26]. For each word of a sentence, the model decides whether it should be kept or dropped in the

Copyright © 2019 ISCA

http://dx.doi.org/10.21437/Interspeech.2019-1750

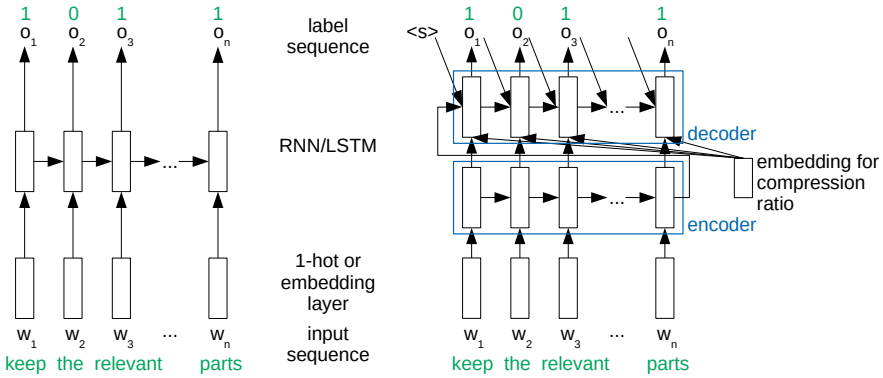

Figure 1: *Left: traditional recurrent model for sentence compression; right: our model.*

compression. Thus, the model performs a binary classification task for each word, as shown for an example phrase in green in Figure 1. Popular models for this task are recurrent neural networks (RNNs) which predict one output label for each input token (left part of Figure 1). However, these models classify each token individually without knowing how long the remaining part of the sentence is and how the final compression would look like. Therefore, it is hard to integrate sentence-level information, such as the desired compression ratio, into the model. However, we argue that especially in the context of real-world applications it is important to be able to control the compression ratio: For example, if only small screens are available for showing the subtitles, it might be advantageous to increase the compression ratio. Thus, in order to model the sentence as a whole and control the compression ratio, we build upon the idea of an encoder-decoder model [27], similar to Filippova et al. [15].

2.1. Our model

Our model (right part of Figure 1) first reads the whole sentence and builds a representation for it (encoder). A traditional encoder-decoder model [27, 28] would then generate another sequence with the decoder which can be of different length than the original sentence. Our setup is different due to the particularities of sentence compression: Building upon the representation for the whole sentence, we aim for a decision for each token of the input sentence whether to keep it or to drop it in the compression. Therefore, we adapt the decoder and feed the hidden states of the encoder into it again, so that we can make a keep and drop decision for each individual token. This is similar to the approach taken by Filippova et al. [15]. In contrast to their work, however, we extend the input for the decoder with an encoding of the desired compression ratio which can be used to produce more keep or more drop labels, respectively.

Encoder. We use a bi-LSTM encoder. Its inputs are word embeddings which are initialized with pre-trained word2vec [29] embeddings of 256 dimensions and randomly initialized embeddings of 10 dimensions for the part-of-speech (POS) tags of the words. All embeddings are fine-tuned during training. The bi-LSTM layer of the encoder transforms the input sequence into a sequence of hidden states which then forms the input to the decoder.

Decoder. The first hidden state of the decoder is initialized with the last hidden state of the encoder, i.e., with a representation for the whole sentence. As motivated before, for producing the i-th output label, the decoder gets the i-th hidden state of the encoder as input. Additional inputs are an encoding of the compression ratio and – as for standard decoders – the previously predicted label (or a special start-of-sentence token for the first output). To encode the compression ratio, we first define classes $[c, c+0.1), 0 \leq c \leq 0.9$ and map each compression ratio to their respective class. For example, a compression ratio of 0.45 is mapped to the class `[0.4;0.5)`. Then, we represent each class with a one-hot vector (1-of-N encoding) of 10 dimensions. The output layer of the decoder is a standard softmax layer.

3. Experiments for sentence compression

3.1. Data and evaluation measures

We use the Google benchmark dataset for sentence compression [26][1]. It consists of news texts and compressions automatically derived based on the headlines and the parse trees of the sentences. For more details, see Filippova and Altun [26]. The dataset consists of 200,000 training and 10,000 test instances. We split the training set into a core-training (180,000 instances) and development set (20,000 instances). Additionally, the dataset also provides the results of tokenizers, lemmatizers, POS taggers, named entitiy recognizers and syntactic dependency parsers for the sentences as well as the compression ratio for each sentence. For our model, we use only the tokenized sentences, the POS tags of the words and the compression ratios as input.

Since the real-life application scenario of our model is the compression of spoken language, we cannot assume to have clean punctuation marks as in news article texts. Therefore, we evaluate our models both with (`punct`[2]) and without (`no_punct`) punctuation marks.

We report the sentence-level accuracy A_s, token-level accuracy A_t, the F_1 score for the keep label as well as the compression ratio accuracy A_c.

3.2. Training and hyperparameters

We implement our models in PyTorch[3]. For pretraining the word embeddings, we use the word2vec [29] implementation provided by gensim [30] with a minimum word occurrence count of 5. We set the dimensionality of the word embeddings to 256 and train the embeddings on the training part of

[1] https://github.com/google-research-datasets/sentence-compression

[2] We removed the sentence-ending punctuation marks from the test instances since the training instances do not provide them either. Alternatively, we could have added a hand-crafted rule for how to compress a sentence-ending punctuation mark.

[3] https://pytorch.org, version 0.40

Table 1: *Experimental results (in %) on compression benchmark dataset with (punct) and without (no_punct) punctuation marks.*

	Model	no_punct				punct			
		A_s	A_t	A_c	F_1	A_s	A_t	A_c	F_1
(i)	Baseline (LSTM)	19.7	84.5	30.5	79.6	20.0	85.4	32.0	79.7
(ii)	(i) + POS tags	21.2	85.3	30.7	80.6	22.1	85.9	33.0	80.4
(ii)	(ii) + compression ratio	26.0	87.5	41.9	84.0	26.3	87.6	38.3	84.0
(iv)	Our proposed method	**32.7**	**88.8**	**41.0**	**85.9**	31.6	**89.2**	**41.7**	**85.9**
	Filippova et al. [15] base	-	-	-	-	30.0	-	-	80.0
	Filippova et al. [15] + dep	-	-	-	-	**34.0**	-	-	82.0

the dataset. Out-of-vocabulary (OOV) words are mapped to a special OOV embedding which is initialized with the average of all other word vectors. The parameters of the other layers of our model are initialized with uniform Glorot [31]. For training, we use mini-batch gradient descent with a batch size of 100 and Adam [32] as the optimizer. We train the model for ten epochs in total and shuffle the training data after each epoch. After each epoch, we evaluate the performance of our model on the development set. To avoid overfitting, we select the model with the best performance on the development set after training. In addition, we employ weight decay with a parameter of $1e^{-5}$. For the LSTM, 256 hidden units per direction led to the best results on the development set.

3.3. Experiments and results

Baseline and feature exploration. Our baseline model is a bi-directional LSTM which predicts one label for each input token. Building upon the baseline, we explore different additional input features, namely POS tags and an encoding for the target compression ratio. The results show that these features increase the performance of the baseline.

Our model. In comparison to the baseline model, our proposed encoder-decoder approach performs better w.r.t. all evaluation measures.

Punctuation marks. Further, we test the effect of punctuation marks in the input data by comparing models trained on data with punctuation marks (`punct`) to models trained on data without punctuation marks (`no-punct`). The results show that especially our encoder-decoder model is able to perform well without punctuation marks, an observation which is important when applying it to spoken language transcriptions.

Comparison to state of the art. We compare our results to Filippova et al. [15], the closest state-of-the-art approach to our architecture (and to the best of our knowledge the best performing model on this dataset so far). In Table 1, we show the results of their base model and the results of their best model which also uses different features from the dependency parse trees of the sentences. Our results are comparable to theirs even though we do not use dependency parse features: Our F_1 score is better while our sentence-level accuracy score is slightly worse.

Compression ratio. To further investigate the effect of the target compression ratio on the output, we compress each sentence of the test set with each of our possible target compression ratio classes. Then, we evaluate how close the actual compression ratio in the output is to the desired one. Figure 2 shows the results. Especially for medium compression ratios, the model is able to control the length of the output of the sentence accordingly. For very small compression ratios, many sentences might not have a reasonable compression. This might be a reason why the accuracy drops considerably for those ratios. Thus, the output is influenced by both the input sentence and the target compression ratio.

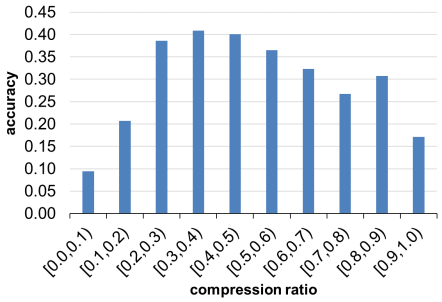

Figure 2: *Compression ratio accuracy when assigning one specific compression ratio class for all sentences of the test data.*

4. User study

To evaluate the applicability of compressed subtitles in reality, we conduct a user study. In particular, we assess three conditions: human compressed subtitles (*compressed_h*), system compressed subtitles (*compressed_h*) and original subtitles without compression (*full*).

We target the following research questions with the study:

RQ1: Compared to standard subtitles, what are the effects of compressed subtitles on cognitive load and comprehension?

RQ2: What is the perceived usefulness of the three different subtitle conditions?

Moreover, the design of our conditions will allow us to assess potential differences between manually and automatically compressed subtitles regarding cognitive load, subjective feedback and comprehension.

4.1. Data and apparatus

To make the setting of the user study similar to possible applications in real life (as described in Section 1), we choose TED talks[4] as our dataset. We choose nine videos[5] of 3–4.5min length and download them together with their subtitles from Amara[6]. The subtitles have 15–27 sentences (456–717 words) per video and 204 sentences in total. For the *full* condition, we directly use them without further processing. For the *compressed_h* condition (compressed by human), we ask a person to mark the important parts of each sentence, retaining $\sim 50\%$ of the words. For the *compressed_s* condition (compressed by system), we run our model (without POS tags) from Section 2 on the subtitles with a target compression ratio class of `[0.5;0.6)`.

[4]https://www.ted.com/, available under Creative Commons License.

[5]https://www.youtube.com/watch?v=ID with ID $\in$ {qpfq3xCdAu4, TRQdHrGuVgI, uv5-hIif7BQ, YX_OxBfsvbk, IBf9pXOmpFw, EaY_6muHSSI, xqzLm0Xua8g, 3Va3oY8pfSI, NAYkF04IZHI}

[6]https://amara.org/en/teams/ted/videos/

4.2. Study design and participants

Each participant experiences every of the three conditions (repeated measures design, counterbalanced via Latin Latin Square [33]) with three videos per condition to minimize the confounding effect of the speakers. We measure subjective cognitive load and usefulness as Likert-style ratings, comprehension scores and free-form comments.

We recruit 30 university students as participants. Their age ranges between 20 and 32 ($\mu = 25.06$, $\sigma = 3.46$). Three of the participants are (near) native English speakers, 24 of them consider themselves fluent and three report to have a good knowledge of English. Their mother tongues vary among German (14 participants), English (2) and others (14). The majority of them watches or listens to English content *"often"* (14 participants) to *"always"* (12). However, most of them use subtitles *"sometimes"* (10 participants) to *"rarely"* (11).

4.3. Procedure

The study was conducted in a computer lab at the university in a supervised environment using an online survey.[7] First, the participants are briefed about the study purpose and sign a form of consent. Second, they answer demographic questions (occupation, age, gender, mother tongue, English proficiency) and state their daily-life exposure to English videos and their subtitle usage behaviour with possible answers ranging from *"1 - never"* to *"5 - always"* (5-point Likert scale) [34]. In the main part, they are shown 3 videos per condition in random order and without knowing the condition. After each video, they are asked to answer questions regarding the cognitive load (mental demand, temporal demand, effort and frustration), based on NASA TLX [35], subjective feedback questions with 5-point Likert-scale answer options ranging from *"1-disagree"* to *"5-agree"* (see Table 2) as well as comprehension questions on the video content. Finally, we ask them for concluding feedback.

Table 2: *Subjective Feedback Questions.*

The subtitles were easy to read.	**(s1)**
The subtitles helped me to understand the content.	**(s2)**
The subtitles were confusing.	**(s3)**
The subtitles were too short.	**(s4)**
The subtitles were too long.	**(s5)**
The subtitles contained all important information.	**(s6)**

4.4. Results

We assess the results of the different conditions for statistical significance with $\alpha = 0.05$. For ordinal data, we employ repeated-measures Friedman tests followed by Dunn's multiple comparisons test with p-value correction for multiple comparisons. For interval data, we use a one-way repeated-measures ANOVA with a Geisser-Greenhouse correction.

RQ1: effects of compressed subtitles. We use the answers for the comprehension questions and the cognitive load questions to investigate the effects of compressed subtitles on the users. Our results show slightly higher comprehension scores for the *full* condition than for the other conditions but no significant difference ($F(2, 29) = 0.601$, $p = 0.5506$) between the conditions. Thus, we cautiously conclude that 50% of the subtitle content might be enough for comprehension. This confirms the findings of Rooney [22] and Guillory [21].

For the cognitive load scores[8], however, the Friedman test finds a significant difference between the subtitle conditions in the categories mental ($\chi^2(2) = 7.719$, $p = 0.0211 < \alpha$), effort ($\chi^2(2) = 12.87$, $p = 0.0016 < \alpha$), and frustration ($\chi^2(2) = 21.82$, $p < 0.0001$). Only, for the temporal demand, we do not see a significant difference ($\chi^2(2) = 0.3158$, $p = 0.8539$). The participants seem to be more irritated by compressed subtitles. However, there is no significant difference between human compressed and system compressed subtitles ($p_{corrected} > 0.99$). These higher cognitive load scores for partial subtitles are in line with the findings of Montero Perez et al. [23], Behroozizad and Majidi [24] and Bensalem [25]. This could be explained by a lack of *"belongingness"* (c.f. Grimes [36]) of the two stimuli: Audio and subtitles cannot be processed together and instead compete for attention, resulting in higher frustration and effort [36].

RQ2: usefulness. We use the answers to the subjective feedback questions (see Table 2) to determine the usefulness of (compressed) subtitles. We find significant differences between full and compressed subtitles for s1 ($\chi^2(2) = 16.32$, $p < \alpha$), s2 ($\chi^2(2) = 21.68$, $p < \alpha$), s3 ($\chi^2(2) = 38.21$, $p < \alpha$), s4 ($\chi^2(2) = 38.21$, $p < \alpha$) and s6 ($\chi^2(2) = 51.86$, $p < \alpha$) but not for s5 ($\chi^2(2) = 5.450$, $p = 0.655$). The Dunn's post-hoc test shows no significant difference between human- and system-compressed subtitles for neither of the questions.

To sum up, the perceived usefulness of the compressed subtitles is mixed with a tendency towards full subtitles because the participants are used to it (as stated in their final feedback). This is in line with the findings of Berke et al. [37], who also observe that people tend to reject unknown subtitle designs.

Overall findings. Compressed subtitles seem to be sufficient for comprehension, but cause higher cognitive load. Note that this negative impact is in line with previous findings in [36]. Moreover in many real applications where information from acoustic signals is not accessible, the main objective of automatic compression approach is comprehension. Furthermore as mentioned before, the results of our study do not show significant differences between manually vs. automatically compressed subtitles. Thus, at first sight, automatic compression seems to be already applicable in real-life scenarios.

5. Conclusions

In this paper, we investigated the automatic compression of subtitles to aid spoken language understanding. We proposed an encoder-decoder architecture for sentence compression that allows to specify a target compression ratio for the output. Our model is state of the art on a popular benchmark dataset and is shown to be applicable to compress audio subtitles. Furthermore, we conducted a user study on the effects of compressed subtitles on user experience. The results show that automatically compressed subtitles are sufficient for comprehension and there are no significant differences between manual vs. automatic compressions. This indicates the usefulness of our automatic compression approach in real-life scenarios. However, more research needs to be done to verify this in the future.

6. References

[1] R. Vanderplank, "The value of teletext sub-titles in language learning," *ELT Journal*, vol. 42, no. 4, pp. 272–281, 1988.

[2] I. Krejtz, A. Szarkowska, and M. Łogińska, "Reading Function

[7] https://www.limesurvey.org/

[8] More information on the scores can be found in [35].

and Content Words in Subtitled Videos," *Journal of Deaf Studies and Deaf Education*, vol. 21, no. 2, pp. 222–232, 2016.

[3] D. Burnham, G. Leigh, W. Noble, C. Jones, M. Tyler, L. Grebennikov, and A. Varley, "Parameters in Television Captioning for Deaf and Hard-of-Hearing Adults: Effects of Caption Rate Versus Text Reduction on Comprehension," *Journal of Deaf Studies and Deaf Education*, vol. 13, no. 3, pp. 391–404, 2008.

[4] N. T. Zanón, "Using subtitles to enhance foreign language learning," *Porta Linguarum: revista internacional de didáctica de las lenguas extranjeras*, vol. 6, p. 4, 2006.

[5] H. Williams and D. Thorne, "The value of teletext subtitling as a medium for language learning," *System*, vol. 28, no. 2, pp. 217–228, 2000.

[6] V. Ferdiansyah and S. Nakagawa, "Effect of captioning lecture videos for learning in foreign language," Toyohashi University of Technology, Tech. Rep. 13, 2013.

[7] M. S. Mirzaei, K. Meshgi, Y. Akita, and T. Kawahara, "Partial and synchronized captioning: A new tool to assist learners in developing second language listening skill," *ReCALL - The Journal of the European Association for Computer Assisted Language Learning*, vol. 29, no. 2, pp. 178–199, 2017.

[8] S. Moran, "The effect of linguistic variation on subtitle reception," in *Eyetracking in Audiovisual Translation*, E. Perego, Ed. Aracne Editrice, 2012, pp. 183–222.

[9] J. C. Yang, C. L. Chang, Y. L. Lin, and M. J. A. Shih, "A study of the POS keyword caption effect on listening comprehension," in *ICCE*, 2010, pp. 708–712.

[10] J. Clarke and M. Lapata, "Models for Sentence Compression: A Comparison across Domains, Training Requirements and Evaluation Measures," in *ACL*, 2006, pp. 377–384.

[11] H. Jing and Hongyan, "Sentence reduction for automatic text summarization," in *ANLP*, 2000, pp. 310–315.

[12] T. Cohn and M. Lapata, "Sentence Compression Beyond Word Deletion," in *Coling*, 2008, pp. 137–144.

[13] ——, "Large Margin Synchronous Generation and its Application to Sentence Compression," in *EMNLP/CoNLL*, 2007, pp. 73–82.

[14] J. Clarke and M. Lapata, "Global Inference for Sentence Compression: An Integer Linear Programming Approach," *Journal of Artificial Intelligence Research*, vol. 31, pp. 399–429, 2008.

[15] K. Filippova, E. Alfonseca, C. A. Colmenares, L. Kaiser, and O. Vinyals, "Sentence Compression by Deletion with LSTMs," in *EMNLP*, 2015, pp. 360–368.

[16] S. Klerke, Y. Goldberg, and A. Søgaard, "Improving sentence compression by learning to predict gaze," in *NAACL*, 2016, pp. 1528–1533.

[17] N.-T. Tran, V.-T. Luong, N. L.-T. Nguyen, and M.-Q. Nghiem, "Effective attention-based neural architectures for sentence compression with bidirectional long short-term memory," in *SoICT*, 2016, pp. 123–130.

[18] Z. Lu, W. Liu, Y. Zhou, X. Hu, and B. Wang, "An Effective Approach of Sentence Compression Based on Re-read Mechanism and Bayesian Combination Model," in *Chinese National Conference on Social Media Processing*, 2017, pp. 129–140.

[19] D.-V. Lai, N. T. Son, and N. Le Minh, "Deletion-Based Sentence Compression Using Bi-enc-dec LSTM," in *PACLING*, 2017, pp. 249–260.

[20] Y. Kikuchi, G. Neubig, R. Sasano, H. Takamura, and M. Okumura, "Controlling output length in neural encoder-decoders," in *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*. Austin, Texas: Association for Computational Linguistics, Nov. 2016, pp. 1328–1338. [Online]. Available: https://www.aclweb.org/anthology/D16-1140

[21] H. G. Guillory, "The Effects of Keyword Captions to Authentic French Video on Learner Comprehension," *CALICO Journal*, vol. 15, no. 1-3, pp. 89–108, 1998.

[22] K. Rooney, "The Impact of Keyword Caption Ratio on Foreign Language Listening Comprehension," *International Journal of Computer-Assisted Language Learning and Teaching*, vol. 4, no. 2, pp. 11–28, 2014.

[23] M. Montero Perez, E. Peters, and P. Desmet, "Is less more? Effectiveness and perceived usefulness of keyword and full captioned video for L2 listening comprehension," *ReCALL - The Journal of the European Association for Computer Assisted Language Learning*, vol. 26, no. 1, pp. 21–43, 2014.

[24] S. Behroozizad and S. Majidi, "The Effect of Different Modes of English Captioning on EFL Learners' General Listening Comprehension: Full Text vs. Keyword Captions." *Advances in Language and Literary Studies*, vol. 6, no. 4, pp. 115–121, 2015.

[25] E. A. Bensalem, "The impact of keyword and full video captioning on listening comprehension," *Journal of Teaching English for Specific and Academic Purposes*, vol. 4, no. 3, pp. 453 – 463, 2016.

[26] K. Filippova and Y. Altun, "Overcoming the Lack of Parallel Data in Sentence Compression," in *EMNLP*, 2013, pp. 1481–1491.

[27] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in *NIPS*, 2014, pp. 3104–3112.

[28] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," in *ICLR*, 2015.

[29] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," 2013.

[30] R. Řehůřek and P. Sojka, "Software framework for topic modelling with large corpora," in *LREC Workshop New Challenges for NLP Frameworks*, 2010, pp. 45–50.

[31] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *AISTATS*, 2010, pp. 249–256.

[32] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," *ICLR*, 2014.

[33] A. P. Field and G. J. Hole, "Descriptive Statistics," in *How to Design and Report Experiments*. Sage Publications, 2003, pp. 109–140.

[34] W. M. Vagias, "Likert-type Scale Response Anchors. Clemson International Institute for Tourism," *& Research Development, Department of Parks, Recreation and Tourism Management, Clemson University*, 2006.

[35] S. G. Hart and L. E. Staveland, "Development of NASA-TLX (Task Load Index): Results of Empirical and Theoretical Research," *Advances in Psychology*, vol. 52, no. C, pp. 139–183, 1988.

[36] T. Grimes, "Mild AuditoryVisual Dissonance in Television News May Exceed Viewer Attentional Capacity," *Human Communication Research*, vol. 18, no. 2, pp. 268–298, 1991.

[37] L. Berke, C. Caulfield, and M. Huenerfauth, "Deaf and Hard-of-Hearing Perspectives on Imperfect Automatic Speech Recognition for Captioning One-on-One Meetings," in *ACM ASSETS*, 2017, pp. 155–164.