

Supplementary material: Dissecting Temporal Understanding in Text-to-Audio Retrieval

In this supplementary material, we provide additional information about AudioCaps [1] validation data and text-to-text contrastive loss experiments. Then, we describe in more details why we have generated a more uniform version of AudioCaps. Lastly, we provide an approach to more confidently evaluate the ‘correctness’ of AudioCaps descriptions using LLMs.

1 AUDIOCAPS VALIDATION DATA

We consider the validation set of AudioCaps and plot the distribution of sentences containing temporal cues in Fig. 1. We notice that differently from the train and the test set, there are considerably more sentences containing the temporal cues ‘As’ and ‘While’. It is important for the validation set to be representative of the training and test sets and at the same time to contain examples of different temporal cues. This is to select the best model that has the potential of performing well on the test set and at the same time, to understand temporal constraints.

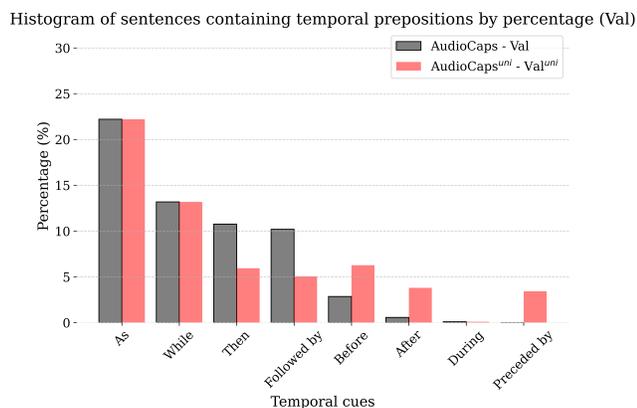


Figure 1: Distribution of temporal conjunctions and prepositions in the AudioCaps validation dataset.

2 ADDITIONAL EXPERIMENTS ON AUDIOCAPS

The performance on *TempTest*, and on the modified *TempTest^{rev}* and *TempTest^{rep}* sets obtained from AudioCaps is almost the same in Tab. 4. This means that the model [2] does not differentiate much between the right order of the text sound events or them being reversed. Therefore, we investigate if adding our text-text contrastive loss \mathcal{L}_{tt} encourages the model to pay more attention to temporal ordering on AudioCaps, as we found this to be the case on our synthetic SynCaps dataset.

When using our additional text-to-text loss for finetuning the model, it is more sensitive to the ‘reversed’ and ‘replaced’ subsets (Tab. 1). However, the overall results are similar to when the loss is not used, warranting further research into improving the training objective.

Table 1: Text-to-audio retrieval and audio-to-text retrieval results on AudioCaps for the model fine-tuned on AudioCaps^{uni} (Train^{uni}) using our text-text contrastive loss \mathcal{L}_{tt} .

Eval Dataset	Subset	Loss	T→A A→T	
			R@1	R@1
AudioCaps ^{uni}	Test	$\mathcal{L}_{ta} + \lambda \mathcal{L}_{tt}$	42.48	53.43
	TempTest	$\mathcal{L}_{ta} + \lambda \mathcal{L}_{tt}$	49.33	61.86
	TempTest ^{rev}	$\mathcal{L}_{ta} + \lambda \mathcal{L}_{tt}$	39.06	50.85
	TempTest ^{rep}	$\mathcal{L}_{ta} + \lambda \mathcal{L}_{tt}$	38.78	51.40

3 WHY AUDIOCAPS UNIFORM DATA?

We propose a more uniform version of the AudioCaps dataset that is easy to use in experiments, as it only changes the original text descriptions to have the same meaning but use a more varied set of temporal cues. If models reach a point where they truly understand temporal ordering, we should expect that, the performance on the *Test* and *Test^{uni}* sets will be very similar. The same should apply to *TempTest* and *TempTest^{uni}*.

[3] perform one experiment where they take 88 text-audio examples containing ‘before’ and 88 text-audio examples containing ‘after’. Then, ‘before’ and ‘after’ are swapped and vice versa. This experiment is used to evaluate if the model understands the temporal ordering implied by these prepositions. The reason such a small test set is used for this experiment is that there are not enough example sentences containing ‘before’ and ‘after’. We believe that this constraints the statistical significance of the experiments in [3]. However, more sentences containing these prepositions can easily be obtained by re-writing what we already have. This allows us to run a more reliable experiment regarding the effect of replacing ‘before’ and ‘after’. It also allows for a more comprehensive experiment where we not only replace ‘before’ with ‘after’, but generate other negative ‘swaps’, as done in our *rep* experiment.

4 VERIFY CORRECTNESS OF LLM EVALUATIONS OF AUDIOCAPS DESCRIPTIONS

In Sec. 3.1., we proposed an empirical evaluation of the correctness and completeness of the AudioCaps descriptions. We have done so by starting from the assumption that LLMs act as oracles, providing reliable evaluations. To further verify the correctness of the LLM evaluations, one approach is to ask the LLM to not only classify the descriptions into ‘correct’, ‘incomplete’ and ‘incorrect’, but also ask for the correct description given the inputs. Then, for the cases where the original AudioCaps descriptions were considered ‘incomplete’ or ‘wrong’, we can repeat the process, but this time provide the LLM-generated ‘corrected’ descriptions and the sound timestamps as inputs. If the LLM considers the generated description as

117 ‘correct’, we can assume that the original evaluation of the original
118 AudioCaps description of ‘incomplete’ or ‘wrong’ was reliable. If
119 not, we can repeat the process from the beginning. We repeat this
120 until the two LLM steps agree for all examples.

121 REFERENCES

122 [1] Chris Dongjoo Kim, Byeongchang Kim, Hyunmin Lee, and Gunhee Kim. 2019.
123 AudioCaps: Generating captions for audios in the wild. In *Proc. NACCL*.
124

[2] Xinhao Mei, Chutong Meng, Haohe Liu, Qiuqiang Kong, Tom Ko, Chengqi Zhao,
175 Mark D Plumbley, Yuexian Zou, and Wenwu Wang. 2023. WavCaps: A ChatGPT-
176 Assisted Weakly-Labelled Audio Captioning Dataset for Audio-Language Multi-
177 modal Research. *arXiv:2303.17395* (2023).
[3] Ho-Hsiang Wu, Oriol Nieto, Juan Pablo Bello, and Justin Salamon. 2023. Audio-
178 Text Models Do Not Yet Leverage Natural Language. *IEEE international conference*
179 *on acoustics, speech and signal processing (ICASSP)* (2023), 1–5.

125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174

180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232