Interspeech 2025
17-21 August 2025, Rotterdam, The Netherlands

WHISTRESS: Enriching Transcriptions with Sentence Stress Detection

*Iddo Yosha, *Dorin Shteyman, Yossi Adi

The School of Computer Science and Engineering
The Hebrew University of Jerusalem, Israel
{iddo.yosha, dorin.shteyman}@mail.huji.ac.il

Abstract

Spoken language conveys meaning not only through words but also through intonation, emotion, and emphasis. Sentence stress, the emphasis placed on specific words within a sentence, is crucial for conveying speaker intent and has been extensively studied in linguistics. In this work, we introduce WHISTRESS, an alignment-free approach for enhancing transcription systems with sentence stress detection. We propose TINYSTRESS-15K, a scalable, synthetic training data for the task of sentence stress detection which resulted from a fully automated dataset creation process. We train WHISTRESS on TINYSTRESS-15K and evaluate it against several competitive baselines. Our results show that WHISTRESS outperforms existing methods while requiring no additional input priors during training or inference. Notably, despite being trained on synthetic data, WHISTRESS demonstrates strong zero-shot generalization across diverse benchmarks. Project page: https://pages.cs.huji.ac.il/adiyoss-lab/whistress.

Index Terms: Sentence stress prediction, computational paralinguistics, automatic speech recognition

1. Introduction

Theoretical work on sentence stress can be divided into linguistic and acoustic research. The linguistic formulation of stress falls into two perspectives, as described in [1]. The first, defines normal stress as a default pattern, independent of meaning, that follows phonological constraints [2]. The second perspective views sentence stress as a semantic tool, where stress may be placed on any word to highlight its semantic importance [3]. Furthermore, acoustic research has established that sentence stress is manifested in the speech signal mainly through variations in duration, amplitude and pitch [4].

Despite these linguistic insights, some computational models for sentence stress detection have relied on acoustic features, with limited integration of linguistic information while using both classical machine learning methods [5, 6, 7] and deep learning methods [8, 9]. Other works show that leveraging language priors, such as phrase and word boundaries, intonation units, and syntactic features can improve sentence stress prediction [10, 11, 12]. Although demonstrating impressive sentence stress prediction performance, the above-mentioned models require during inference one or more of the following: (i) transcriptions of the spoken utterances, (ii) word boundaries, via forced alignment or manual annotation. This reliance introduces a critical constraint, as model performance is influenced by the accuracy of the forced aligner, transcription quality and data scalability. As an exception to these, the authors of [13], suggested fine-tuning Whisper [14] by adding tokens in the text for stress labeling. However, their method do not take into account possible performance degradation caused by altering transcriptions and the fine-tuning process [15]. Another notable work, attempted to use BERT [16] to predict stress at the token level directly from text for controlled prominence in Text-to-Speech (TTS) [17].

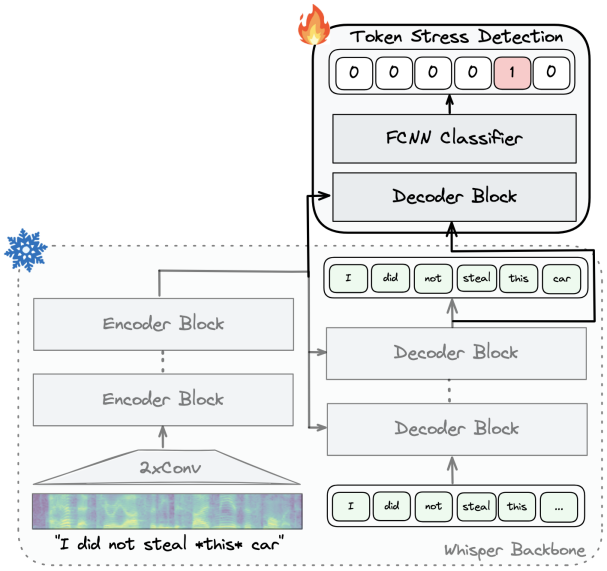

Figure 1: WHISTRESS *Architecture. The Whisper model is kept frozen during training. The extension is a transformer decoder block with cross-attention for the audio embeddings, followed by an FCNN classifier that outputs the stress score per token.*

In this work, we introduce WHISTRESS, a novel, alignment-free approach for sentence stress detection. The proposed approach leverages a Language Model (LM) conditioned on acoustic signals to improve sentence stress detection. We focus on English speech, and equip the Whisper model [14] with an additional stress detection head, that predicts stress targets for each token, thus allowing the model to generate more informative transcriptions without affecting the original model's performance. See Figure 1 for a visual description.

Another significant factor in determining the performance, generalization, and robustness of any proposed model is the quality of training data. Most of the current models that detect sentence stress were trained on either: (i) closed source data [9]; (ii) datasets with non-standard emphasis annotations [10, 11]; or (iii) crowdsourced relying on human annotator judgment to annotate stressed words [8]. Consequently, the quality

*Equal contribution.

10.21437/Interspeech.2025-1475

and consistency of the data can potentially lead to suboptimal performance of any model trained for sentence stress detection.

This motivated the development of TINYSTRESS-15K, a scalable, synthetically generated dataset, designed for sentence stress detection amounting to ∼15 hours of speech. TINYSTRESS-15K was derived from text data, with sentence-stress annotations carrying semantic and rhetorical meaning, generated by a large language model. The corresponding speech was then synthesized using a text-to-speech service, with altered prosodic features to replicate natural vocal stress. This automated approach enabled the generation of a large, diverse dataset, tailored for training stress detection models.

Our contributions: (i) We present WHISTRESS, an alignment-free approach for token-level stress classification in Automatic Speech Recognition (ASR), which can be extended to other paralinguistic and language tasks that predict targets at the word level in future research; (ii) We introduce a method for generating synthetic, stress-annotated speech data to advance future research; (iii) We provide valuable analysis of the Whisper backbone layers to highlight a potential prosodic-linguistic trade-off for sentence stress detection in WHISTRESS; and (iv) We publicly release our code, WHISTRESS model weights and TINYSTRESS-15K dataset.

2. Synthetic Data

We develop an automated pipeline for synthetic dataset generation, consisting of three key steps: (i) transcription selection, (ii) stressed words labeling, and (iii) speech synthesis.

Transcription Selection. To get transcriptions of coherent, diverse, everyday language, we use the TinyStories dataset, a common choice for small language models training [18]. Each sample in the tinyStories dataset is a paragraph of a short story. We generate each sample as a sentence extracted from each paragraph using the NLTK library [19]. Sentences with fewer than three words are filtered out.

Stressed Words Labeling. We chose which words to emphasize by instructing GPT-4o-mini [20] to provide two different options for stressed words in each sentence, such that selected words reflect natural sentence stress, meaning they influence the sentence's interpretation in a semantically significant way. In the training set, we used both stress patterns, while in the test set we randomly chose one stress pattern of the two.

Speech Synthesis. We use the Google Text-to-Speech API to synthesize the stressed speech based on the prior steps, by using SSML syntax that enables editing prosodic features on the word or constituent level.[1] The following steps were taken: (i) *Prosodic features adjustment:* A core step in the data creation procedure. Every emphasized word in the transcription has its amplitude, duration and pitch adjusted. Initially, the adjustments were based on empirical results from [4]. However, since the TTS service generated overly robotic speech, we manually refine them using the following heuristics to achieve a more natural sound: The volume and duration are in proportion to the length of the word, with shorter words having a higher volume and longer duration than longer words. Specifically, duration changed by reducing the speaking rate by 30% to 85% relative to the original rate, and the volume was increased by 3 to 6db. The pitch of emphasized words was adjusted to be higher by 1.5 semitones. To introduce variability, all prosodic features were augmented by adding random standard Gaussian noise.; (ii) *Speaker voice selection:* To ensure diversity and fairness in the dataset, each sample was synthesized by a randomly chosen female or male speaker from a pool of 10 possible voices; and (iii) *Timestamps annotations:* We generate start timestamps for each word in the transcription using the TTS service, to get a word level time alignment.

Overall, we result in a dataset of 1k test samples and 15k training samples which we denote as TINYSTRESS-15K. The total time of synthesized speech in the training set is ∼15 hours.

[1] SSML (Speech Synthesis Markup Language) is a standardized markup language used to control various aspects of speech synthesis for TTS systems.

3. Method

WHISTRESS enhances transcription models with additional sentence stress detection objective. Specifically, we propose generating two outputs for each input speech signal: the transcription and an emphasis score for each token in the transcription. To achieve this, we modify both the architecture of the Whisper model and its training procedure.

3.1. Architecture

We propose a modification to the Whisper model [14] by incorporating a stress detection head that learns to identify sentence stress in the speech signal for each transcribed token. As shown in Figure 1, WHISTRESS model consists of two components:

Whisper Model. The backbone of the WHISTRESS model. It is used to process raw audio into hidden representations that encode phonetic, linguistic, and prosodic features. These representations serve a dual purpose: (i) input for the stress detection head; and (ii) producing Whisper's speech transcription.

Stress Detection Head. A learnable component extending the Whisper model, consisting of: (i) Whisper Decoder Block; and (ii) Fully Connected Neural Network (FCNN) Classifier. The additional Whisper decoder block applies cross-attention between decoder and encoder hidden states of the backbone Whisper model. This component learns the acoustic and linguistic features that contribute to the stress detection. According to our findings in Section 4, the inputs to the additional decoder head are the encoder and decoder embeddings of the 9th layer of the backbone Whisper model. The FCNN classifier is two-layer fully connected neural network that acts as a binary classifier, processing the output of the additional decoder block to assign a stress label to each token (1 if stressed, 0 otherwise).

3.2. Training

Label Alignment. As a first step, the ground-truth word-level stress labels are converted to token-level labels, aligned with error-free transcription tokens. However, Whisper-generated hidden states, which may encode transcription errors, serve as input to the stress detection head. Therefore, any Whisper transcription error can propagate and misalign stress labels by shifting decoder input tokens. Meanwhile, samples with minor transcription errors at word-level remain acoustically informative. To retain such samples, we filter out training examples where the Whisper-generated transcription word length differs from the ground truth word length, allowing for word-level transcription errors. The described length filtering approach mitigates mislabeling stressed words thus ensures reliable supervision.

Unlike previous methods, our label-alignment procedure relies solely on a relaxed word-to-word matching between ground truth (i.e., stress labels) and generated transcriptions as

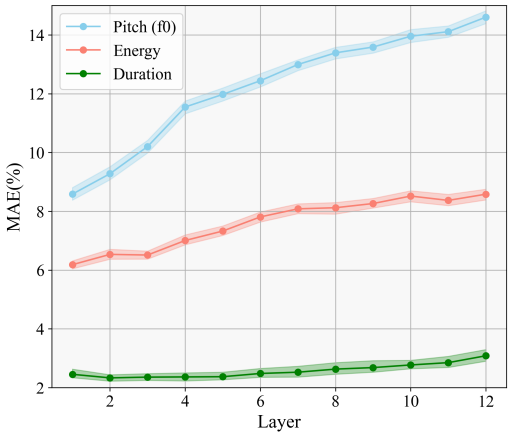

Figure 2: *Prosodic features prediction by Mean Absolute Error (MAE) percentage of Whisper layer embeddings. A lower MAE percentage indicates better prediction. Each curve shows confidence intervals.*

a pre-processing step before training, and is alignment-free during inference. Notably, at any stage, it does not require word-level timestamps (i.e., no time alignment is needed), as Whisper inherently aligns generated tokens with audio features.

Training Procedure. During training, the backbone Whisper model remains frozen, and only the stress detection head (Section 3.1) is trained using cross-entropy loss. The number of trainable parameters in the additional head is $\sim$ 7 million. Additionally, non-word tokens, such as punctuation marks, are included to help WHISTRESS identify them as unstressed.

4. Model Analysis

We analyze Whisper's internal representations to determine which layers capture prosodic features, specifically pitch, energy, and duration. Furthermore, we investigate how WHISTRESS identifies sentence stress by evaluating different layers as input for the stress detection head.

4.1. Prosodic information analysis

To understand where prosodic information is stored in WHISTRESS, we analyze the embeddings of the Whisper backbone model. We use a subset of CREMA-D dataset [21] that contains speech samples with varying emotional content, making prosodic features more prominent in the signal.

To analyze the energy and pitch targets, we utilize the Whisper encoder embeddings, which capture pure acoustic features without conditioning on transcription. We compute fundamental frequency (F0) and root mean square (RMS) energy in 75 ms windows with a 20 ms stride to align with the frame rate of the audio embeddings. To construct targets, we apply max pooling to F0 and mean pooling to RMS energy over 300 ms windows. For each window, at every layer, we pool the mean encoder embeddings, forming corresponding embeddings and targets for each encoder layer.

For the duration, we analyze the Whisper decoder embeddings, assuming that duration correlates with text-speech alignment learned through cross-attention. To generate target durations, we force-align the speech signal using WhisperX [22] and extract the duration of each word in the transcription. We then compute the mean decoder embeddings corresponding to each word, forming embeddings and targets for every decoder layer.

Finally, to capture non-linear dependencies, we train a random forest regression model for each encoder and decoder layer to predict prosodic targets from embeddings [23]. To estimate the 95% confidence intervals of the predictions, we use 100 bootstrap samples [24]. Figure 2 summarizes model performance using Mean Average Error percentage (MAE%) as the evaluation metric, where the MAE% is a max-min normalization of each prosodic feature target. We find that deeper layers in both the encoder and decoder capture less prosodic information in their embeddings.

Table 1: *Word-level sentence stress detection of* WHISTRESS *on* TINYSTRESS-15K *by input layer embeddings.*

Model	Layer	Prec	Rec	F1
WHISTRESS	3	0.781	0.624	0.693
	6	0.83	0.86	0.845
	9	**0.912**	**0.906**	**0.909**
	12	0.902	0.868	0.884

4.2. Layer selection

To further explore the relationship between prosodic information and stress detection, we investigate whether layers that capture prosody more effectively contribute to better stress detection. As shown in Table 1, using embeddings from earlier layers did not produce results comparable to those obtained from deeper encoder and decoder layers.

However, we observe that the 9th layer outperforms the final 12th layer. This finding suggests a potential trade-off between prosodic information in embeddings and the encoding of linguistic knowledge, where intermediate to final layers may offer the optimal balance for stress detection. This aligns with previous research [25] on the wav2vec 2.0 speech representation transformer model [26], which observed opposing trends between acoustic and semantic correlations across its layers.

5. Results

We evaluate our model by comparing it to previously proposed approaches for sentence stress detection and established benchmarks. Our experiments across different datasets validate our model's ability to identify stressed words accurately while also demonstrating impressive generalization capabilities. We report performance using standard classification metrics: precision, recall and F1 score. In our settings, a word is considered stressed by the WHISTRESS model if at least one of its tokens is marked as stressed. We use english-only model of Whisper small. All models were trained for 4 epochs on each dataset, except for the model trained on TINYSTRESS-15K for zero-shot evaluation, which was trained for two epochs.

5.1. Datasets

We consider TINYSTRESS-15K and Aix-MARSEC [27] for validation. In addition, we use Expresso [28] and EmphAssess [29] benchmarks for evaluation, described below.

Aix-MARSEC is a speech corpus containing over 5 hours of 1980s BBC radio recordings, featuring 53 speakers across different speech styles [27]. In accordance with previous work [11], we define a word as stressed for the Aix-MARSEC ground-truth sentence-stress labels if it contains the first syl-

lable in Jassem's narrow rhythm unit (NRU) notation [30]. We split the recordings and their corresponding transcriptions into sentences and obtain 2400 samples. We use 70% of the samples for training and 30% for testing, similar to previous work [10].

Expresso. An expressive speech dataset [28] comprising 47 hours of recordings from four speakers (2 female, 2 male) across a diverse range of speaking styles, including both improvised and read speech. Following the filtering criteria in [29], we include only samples that contain at least one emphasized word. For fair comparison, we use the same test set configuration as [29], selecting the speakers with IDs `ex01` and `ex02`.

EmphAssess. A synthetically generated dataset [29] of speech sentences, each containing at least one stressed word. EmphAssess amounts to 3652 samples, such that 913 unique transcripts are rendered using 4 distinct voices with varying stressed words. The voices used to synthesize all the transcripts, are the 4 distinct Expresso voices, namely `ex01`, `ex02`, `ex03` and `ex04`.

5.2. Baselines

We implement a simple BLSTM model with two layers, each containing 64 hidden units, followed by a linear layer with two outputs for the stress detection task [31]. Inspired by [10], for each word in the transcription, we extract duration, mean energy and max pitch as audio features, in the corresponding speech segment. We train the baseline model using ground truth timestamps on each dataset. To further compare the baseline model with WHISTRESS, which is alignment-free (i.e., it does not require timestamps for training or inference), we also evaluate the baseline models on a forced-aligned test set generated using the Montreal Forced Aligner (MFA) [32]. These models are referred to as *Baseline (+GT alignment)* and *Baseline (+MFA)*, respectively. We also compare WHISTRESS against EmphaClass [29] and a Conditional Random Fields (CRF) model as in [11], and hierarchical BLSTM network as in [10].

5.3. Results

Validation on TINYSTRESS-15K. We compare the performance of WHISTRESS against the baseline outlined in Section 5.2, showing that WHISTRESS achieves better results across all evaluation metrics. Furthermore, we show that in the common scenario where word-level timestamps are unavailable, force alignment may negatively impact the performance of models relying on their availability. Results are summarized in Table 2.

Validation on Aix-MARSEC. As summarized in Table 2, WHISTRESS achieves the best scores over all evaluation metrics compared to the BLSTM hierarchical network in [10] and the CRF in [11]. Additionally, we conduct analysis over the entire Aix-MARSEC dataset revealing that $\sim$ 20K of its $\sim$ 50K words are non-emphasized. However, the set of non-emphasized words contains only $\sim$ 100 unique words that frequently recur and are consistently labeled as non-emphasized. Therefore, emphasis detection over Aix-MARSEC dataset using NRU tagging is relatively straightforward. Consequently, it presents a scenario where words meaning plays a significant role in sentence-stress classification. This supports our hypothesis in Section 4 on the importance of language understanding captured by Whisper's decoder layers.

Evaluation on Expresso and EmphAssess. For zero-shot evaluation, we use a WHISTRESS variant trained solely on TINYSTRESS-15K. Surprisingly, despite the distribution shift, WHISTRESS achieves impressive zero-shot performance on both benchmarks, surpassing EmphaClass on Expresso. This while our method uses general-purpose synthetic data and EmphaClass was trained on a closed-source dataset with a distribution closely matching Expresso, inherently aligning it better with the evaluation data.

Table 2: *Word-level sentence-stress detection validation on* TINYSTRESS-15K *(denoted as TS-15K) and Aix-MARSEC datasets; evaluation on Expresso and EmphAssess. For Expresso and EmphAssess, we compare two variants of* WHISTRESS*: (i) 0-shot and evaluation over all four speakers, and (ii) Training on speakers ex03, ex04, and evaluation over speakers ex01, ex02 (marked with *). Baseline variants are denoted "MFA" and "GT", as explained in section 5.2.*

Evaluated Dataset	Model	Prec	Rec	F1
TS-15K	Baseline (+GT alignment)	0.862	0.853	0.858
	Baseline (+MFA)	0.776	0.859	0.815
	WHISTRESS	**0.912**	**0.906**	**0.909**
Aix-MARSEC	CRF (N=5) [11]	0.85	0.89	0.87
	BLSTM [10]	0.88	0.92	0.90
	Baseline (+GT alignment)	0.905	0.953	0.928
	Baseline (+MFA)	0.904	0.958	0.930
	WHISTRESS	**0.953**	**0.968**	**0.961**
Expresso	EmphaClass	0.569	0.769	0.654
	Baseline (+MFA) [0-shot]	0.404	0.599	0.482
	WHISTRESS [0-shot]	**0.573**	**0.863**	**0.689**
EmphAssess	EmphaClass	0.938	0.94	0.938
	Baseline (+MFA) [0-shot]	0.566	0.609	0.587
	WHISTRESS	**0.945***	0.942*	**0.943***
	WHISTRESS [0-shot]	0.672	**0.98**	0.797

To assess WHISTRESS in a similar setting on a task-oriented dataset, we train it on speaker IDs `ex03`, `ex04` of EmphAssess, which amount to $\sim 1,500$ samples. Despite the limited data, WHISTRESS surpasses EmphaClass when evaluated on the unseen speakers `ex01`, `ex02`. For Expresso, surpassing zero-shot performance requires significantly more samples and diversity of human-recorded speech with sentence stress annotations, which currently remain unavailable. Hence, we do not report results for WHISTRESS using in-domain training data.

6. Conclusion

In this work, we introduced WHISTRESS, an extension to the Whisper model to enrich its transcriptions by marking sentence stress while preserving its core functionality. We also presented a fully automated pipeline for synthetically generating training data specific for sentence stress detection. Equipped with this automatic pipeline, we created TINYSTRESS-15K, a synthetic dataset of $\sim$15 hours. When trained on TINYSTRESS-15K, WHISTRESS demonstrated strong performance and impressive zero-shot generalization across benchmarks, outperforming baselines that utilized training data with additional information or relying on closed-source, domain-targeted datasets.

Analysis revealed that while deeper Whisper layers encode less prosodic information, they are more effective for stress detection, with intermediate layers striking the best balance. By eliminating the need for forced alignment or manual human annotations, WHISTRESS provides a cleaner, more accessible and alignment-free approach to integrating sentence stress detection into ASR systems. We hope our work will encourage further advancements in speech related tasks by enriching speech signal interpretation with semantically significant features.

Acknowledgments. This research work was supported by ISF grant 2049/22.

7. References

[1] D. R. Ladd, *Intonational Phonology*, 2nd ed., ser. Cambridge Studies in Linguistics. Cambridge University Press, 2008.

[2] N. Chomsky and M. Halle, "The sound pattern of english," 1968. [Online]. Available: https://api.semanticscholar.org/CorpusID:60457972

[3] D. Bolinger, "Accent is predictable (if you're a mind-reader)," *Language*, vol. 48, no. 3, pp. 633–644, 1972. [Online]. Available: http://www.jstor.org/stable/412039

[4] V. Van Heuven, *Acoustic Correlates and Perceptual Cues of Word and Sentence Stress: Theories, Methods and Data*, 12 2018, pp. 15–59.

[5] R. Silipo and S. Greenberg, "Prosodic stress revisited: Reassessing the role of fundamental frequency," 01 2000.

[6] S. Kakouros and O. Räsänen, "3pro – an unsupervised method for the automatic detection of sentence prominence in speech," *Speech Communication*, vol. 82, p. 67–84, 09 2016.

[7] T. Mishra, V. R. Sridhar, and A. Conkie, "Word prominence detection using robust yet simple prosodic features," in *Interspeech 2012*, 2012, pp. 1864–1867.

[8] M. Morrison, P. Pawar, N. Pruyne, J. Cole, and B. Pardo, "Crowdsourced and automatic speech prominence estimation," 2023. [Online]. Available: https://arxiv.org/abs/2310.08464

[9] M. de Seyssel, A. D'Avirro, A. Williams, and E. Dupoux, "Emphassess : a prosodic benchmark on assessing emphasis transfer in speech-to-speech models," 2024. [Online]. Available: https://arxiv.org/abs/2312.14069

[10] B. Lin, L. Wang, X. Feng, and J. Zhang, "Joint detection of sentence stress and phrase boundary for prosody," in *Interspeech*, 2020. [Online]. Available: https://api.semanticscholar.org/CorpusID:226201825

[11] G. Lee, H.-Y. Lee, J. Song, B. Kim, S. Kang, J. Lee, and H. Hwang, "Automatic sentence stress feedback for non-native english learners," *Computer Speech & Language*, vol. 41, 06 2016.

[12] A. Suni, D. Aalto, and M. Vainio, "Hierarchical representation of prosody for statistical speech synthesis," *arXiv preprint arXiv:1510.01949*, 2015.

[13] T. Biron, M. Barboy, E. Ben-Artzy, A. Golubchik, Y. Marmor, S. Szekely, Y. Winter, and D. Harel, "Non-verbal information in spontaneous speech-towards a new framework of analysis," *arXiv preprint arXiv:2403.03522*, 2024.

[14] A. Radford, J. W. Kim, T. Xu, G. Brockman, C. McLeavey, and I. Sutskever, "Robust speech recognition via large-scale weak supervision," 2022. [Online]. Available: https://arxiv.org/abs/2212.04356

[15] Z. Mai, A. Chowdhury, P. Zhang, C.-H. Tu, H.-Y. Chen, V. Pahuja, T. Berger-Wolf, S. Gao, C. Stewart, Y. Su, and W.-L. Chao, "Fine-tuning is fine, if calibrated," 2024. [Online]. Available: https://arxiv.org/abs/2409.16223

[16] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," 2019. [Online]. Available: https://arxiv.org/abs/1810.04805

[17] B. Stephenson, L. Besacier, L. Girin, and T. Hueber, "Bert, can he predict contrastive focus? predicting and controlling prominence in neural tts using a language model," 2022. [Online]. Available: https://arxiv.org/abs/2207.01718

[18] R. Eldan and Y. Li, "Tinystories: How small can language models be and still speak coherent english?" 2023. [Online]. Available: https://arxiv.org/abs/2305.07759

[19] S. Bird, E. Klein, and E. Loper, *Natural Language Processing with Python*, 1st ed. O'Reilly Media, Inc., 2009.

[20] A. Hurst, A. Lerer, A. P. Goucher, A. Perelman, A. Ramesh, A. Clark, A. Ostrow, A. Welihinda, A. Hayes, A. Radford *et al.*, "Gpt-4o system card," *arXiv preprint arXiv:2410.21276*, 2024.

[21] H. Cao, D. G. Cooper, M. K. Keutmann, R. C. Gur, A. Nenkova, and R. Verma, "Crema-d: Crowd-sourced emotional multimodal actors dataset," *IEEE Transactions on Affective Computing*, vol. 5, no. 4, pp. 377–390, 2014.

[22] M. Bain, J. Huh, T. Han, and A. Zisserman, "Whisperx: Time-accurate speech transcription of long-form audio," 2023. [Online]. Available: https://arxiv.org/abs/2303.00747

[23] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.

[24] B. Efron and R. Tibshirani, "An introduction to the bootstrap," 1995. [Online]. Available: https://api.semanticscholar.org/CorpusID:123398593

[25] A. Pasad, J.-C. Chou, and K. Livescu, "Layer-wise analysis of a self-supervised speech representation model," in *2021 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*. IEEE, 2021, pp. 914–921.

[26] A. Baevski, Y. Zhou, A. Mohamed, and M. Auli, "wav2vec 2.0: A framework for self-supervised learning of speech representations," *Advances in neural information processing systems*, vol. 33, pp. 12 449–12 460, 2020.

[27] C. Auran, C. Bouzon, and D. J. Hirst, "The aix-marsec project: an evolutive database of spoken british english," *Speech Prosody 2004*, 2004. [Online]. Available: https://api.semanticscholar.org/CorpusID:204957167

[28] T. A. Nguyen, W.-N. Hsu, A. d'Avirro, B. Shi, I. Gat, M. Fazel-Zarani, T. Remez, J. Copet, G. Synnaeve, M. Hassid *et al.*, "Expresso: A benchmark and analysis of discrete expressive speech resynthesis," *arXiv preprint arXiv:2308.05725*, 2023.

[29] M. de Seyssel, A. D'Avirro, A. Williams, and E. Dupoux, "Emphassess: a prosodic benchmark on assessing emphasis transfer in speech-to-speech models," *arXiv preprint arXiv:2312.14069*, 2023.

[30] W. Jassem, "Stress in modern english," *Bulletin de la Société Linguistique Polonaise*, vol. 11, pp. 23–49, 1952.

[31] Z. Huang, W. Xu, and K. Yu, "Bidirectional lstm-crf models for sequence tagging," 2015. [Online]. Available: https://arxiv.org/abs/1508.01991

[32] M. McAuliffe, M. Socolof, S. Mihuc, M. Wagner, and M. Sonderegger, "Montreal Forced Aligner: Trainable Text-Speech Alignment Using Kaldi," in *Proc. Interspeech 2017*, 2017, pp. 498–502.