Phrase break prediction for long-form reading TTS: exploiting text structure information

Viacheslav Klimkov, Adam Nadolski, Alexis Moinet, Bartosz Putrycz, Roberto Barra-Chicote, Thomas Merritt, Thomas Drugman

Amazon.com

vklimkov,anadolsk,amoinet,bartosz,rchicote,thommer,drugman@amazon.com

Abstract

Phrasing structure is one of the most important factors in increasing the naturalness of text-to-speech (TTS) systems, in particular for long-form reading. Most existing TTS systems are optimized for isolated short sentences, and completely discard the larger context or structure of the text.

This paper presents how we have built phrasing models based on data extracted from audiobooks. We investigate how various types of textual features can improve phrase break prediction: part-of-speech (POS), guess POS (GPOS), dependency tree features and word embeddings. These features are fed into a bidirectional LSTM or a CART baseline. The resulting systems are compared using both objective and subjective evaluations. Using BiLSTM and word embeddings proves to be beneficial.

Index Terms: speech synthesis, TTS, BiLSTM, long form reading, phrasing, respiratory pauses, audiobooks

1. Introduction

The term *phrasing* is used to describe the phenomenon of grouping words into phrases and separating those phrases with a pause and/or intonational reset. The latter is usually associated with the fundamental frequency contour and segmental duration changes. In a text, the phrasing structure is usually marked up with punctuation. When the TTS system observes a comma, it is generally accepted that a phrase boundary should be placed, as the author intended that words before and after the punctuation should be separated. This approach performs reasonably well: about 54% of breaks are set up correctly, while only 0.05% are inserted erroneously [1]. This under-prediction comes from the fact that human speakers usually insert phrase breaks on word transitions without any punctuation either for the sake of expressivity, better comprehension or simply to take a breath. We refer to these as *respiratory breaks* and their reliable prediction with statistical models is the focus of this paper.

Respiratory break prediction is especially important in the domain of long-form reading. We use the term 'long-form reading' (LFR) to describe passages of text (such as newspapers, books or plays) that consist of multiple sentences. Most existing TTS systems are optimized for isolated short sentences (we refer to this as 'short-form reading' – SFR), and completely discard the larger context or structure of the long-form text. In the LFR domain the text provides the wider semantic context. Learning how to exploit LFR data could lead to improvements in the customer experience with various Amazon applications such as 'news flash briefings' read by Alexa or reading books.

Phrasing has been the focus of various studies. Part-of-speech (POS) tags are conventional features for this task [2–5]. Some approaches have, however, focused on removing the reliance upon POS taggers. In [6], unsupervised POS induction with a grammar-based phrasing model are used for low-resourced languages. In [7], Watts et al. use a shared projection neural network which does not need conventional knowledge-based resources. In addition to POS tags, other types of features derived from the text have been investigated. These include dependency tree relations [8] and word embeddings [7, 9].

Following the selection of the features to use, various machine learning methods have been proposed for the phrasing modelling. Classification And Regression Tree (CART) and binary tree models are the most popular and have been used across multiple studies [3, 5, 10, 11]. Alternative techniques have followed the advances in machine learning: HMMs [2] and neural networks [9, 12, 13]. Most recent findings show that bidirectional Long Short Term Memory (BiLSTM) models give superior performance when the amount of training data is large enough [9].

The goal of this paper is to improve the prediction of phrase breaks for LFR TTS. Using a large amount of audiobook data, we explore various types of input features: POS tags, guess POS, dependency parsing and word embeddings. The modelling techniques used range from conventional CART to more advanced BiLSTM. These approaches are compared both in an objective test based on F-scores, but also in an extensive subjective evaluation involving linguists, professional data-analysts and mechanical Turk crowdsourcing. Compared to the literature, the main contributions of this paper are: *i)* we are not limited by the amount of data; *ii)* we explore in a systematic fashion what are the best feature representations for phrase break prediction in the context of LFR TTS; *iii)* we investigate whether advanced feature representations and statistical modelling techniques bring an improvement in perceptual tests. Information about the reproducibility of our research flow using openly-available data and tools is provided throughout the paper.

2. Phrasing corpus from audiobooks

Typically the corpora used to build TTS systems is designed using SFR text. The subsequent recordings are too strict and contain too little respiratory breaks to train reliable models for LFR purpose. In this paper we assume that there are some latent rules for respiratory pause insertion that are shared between speakers. In order to obtain a large corpus with phrase breaks annotated, we applied the approach described in [6] to audiobooks and corresponding ebooks. We selected a set of approximately 500 non-fiction books, from 9 voice talents (around 55 books per speaker). They were segmented into paragraphs and forced alignment was performed using speaker-dependent acoustic models. These acoustic models were trained on a subset of the data. Paragraphs were heavily pruned using an approach similar to [14], keeping only sentences for which the audio fully corresponds to the transcription. We also dropped all of the paragraphs containing reported speech and dialogs. This

was done to fix the domain of the data used, because according to our informal observations, reported speech and dialogs have a specific phrasal structure to enhance comprehension and character diarization. As future work, we plan to develop a reliable dialog/reported speech detector and train a separate phrasing model for these. After pruning, we split paragraphs into sentences and selected only sentences with at least 1 respiratory break. This resulted in 663k utterances. To obtain similar amounts of data, one could use the Librispeech dataset [14]. Making use of multiple sentences for phrase break prediction to avoid semantic context constraining is left as future work.

According to prosody labelling standards [15], there are up to seven levels to describe the strength of breaks. As this paper is focused on respiratory phrase breaks, we can simplify the break index tier to respiratory phrase break with pause (*4*), intonation reset without pause (*3*) and no break (*0*, *1*, *2*). We tried to apply AuToBI [16] with open-source models for automatic annotation, but a manual check revealed an extremely low reliability of intonation resets (break index *3*). Nonetheless, a high correlation between actual phrase breaks with pauses and aligned between-word silences was noticed (break index *4*). To enhance the annotation of intonation resets (break index *3*), custom models for AuToBI should be built. This would require costly manual annotation of a few thousand utterances by experts. We decided to postpone this activity, and build models for phrase breaks with a pause. Two factors contributed to this decision: 1) we wanted to confirm the reliability and applicability of the data obtained from audiobooks; 2) state-of-art TTS systems (parametric or guided unit-selection) have a prosody modelling component which implicitly learns intonation resets in certain contexts. We leave automatic annotation of intonation resets without pauses, to obtain explicit control over resets insertion, as future work.

Having silences aligned between words, we mark word transitions with silence $>$100ms as phrase breaks. For word transitions with punctuation, silences $>$30ms indicate a break. The resulting corpus consists of 17M word transitions including 1M respiratory breaks. 10% of utterances were excluded from training and split into two folds of equal size: development and test. The former was held out for selecting the best training epoch of the DNN models and defining the phrasing probability threshold (see Section 5). The latter was used to report objective performance.

3. Input features

Phrasing prediction from text has so far mostly been based on POS tags of various complexities, word punctuation and distances from a given word to: the neighbouring breaks; the start of the sentence; the end of the sentence [2–5]. There were attempts to introduce syntactic knowledge by providing dependency relations between the words [8]. Recent advances in representing words in a continuous semantic space (word embeddings) were also investigated for phrasing [7, 9].

The features used as input to the models presented in Section sec:modelling are described below. Contrary to the previous work, in this paper we investigate all of these features on a single corpus in matching conditions.

- **Word punctuation:** Since punctuation mark-up is intended to create phrasal structure, this input feature was used throughout all experiments. Even though we were focused on respiratory pauses, we assume that they should be greatly influenced by surrounding breaks on punctuation. A 9-dimensional one-hot vector is used for these features (8 dimensions relate to different types of punctuation marks and 1 dimension represents no punctuation mark).
- **Distance:** CART models do not take context into account. Additional distance features are therefore needed. The number of syllables from the current word to the previous and next punctuation mark was used. This results in 2 additional features.
- **GPOS:** Guessed POS as in [17]. Tags are derived from trivial look up table for function words. All others are defined as content words. We also include POS derived from the lexicon. The total number of dimensions for these features is 39.
- **POS:** Penn POS tags [18] extracted with our internal NLP toolkit. As an open-source alternative, we would suggest Stanford CoreNLP [19]. The total number of dimensions for these features is 266. These features have such high dimension because the NLP framework used in this investigation expands tokens such as "wasn't" into "was not", and assigns a POS tag to each word separately. Since all other features were at token level, we had to introduce a collapsed token-level POS, such as "VBD+RB".
- **Dependency tree features (Dep):** Features derived from the dependency grammar for a given word, extracted with our internal NLP toolkit. Stanford CoreNLP is again a good alternative. The features we used are as follows: the identity of the parent of the current word in the dependency tree and constituency tree, the relative position of the word in the sentence, the relative position of a word to its parent in the dependency tree and the depth of the constituency tree for a given word. The total dimension of this feature set is 67. The feature set derived from the dependency tree could be greatly expanded, for example to include information about children and the distance in the tree between neighbouring words. However this is left as future work.
- **W2V:** Word embeddings (a.k.a. word2vec) are a representation of words in a continuous space [20]. These representations are learned from huge amounts of unannotated text. In our experiments we used word embedding vectors pre-trained on ebooks and wikipedia articles. It was decided to train word embeddings specifically for this task instead of using the freely-available word2vec models (which are trained on news domain text), in order to overcome a possible domain mismatch and to apply the same text normalization. Alternatively, one can use the pre-trained models shared in [21]. The vectors used in this investigation have a dimensionality of 300.

4. Modelling

Decision trees have been widely used to predict the phrasing structure of text [3, 5, 10, 11]. HMMs [2] and neural networks [12] have been investigated as possible alternative solutions. The growth of deep learning brought modelling techniques that do not require explicit context injection, tracking distance to previous and next break or weighting output with n-grams. DNNs [13] and LSTMs [9] have naturally been tried and were shown to improve phrasing prediction accuracy.

In our experiments, we use a decision tree as a baseline system, and focus on the development of a BiLSTM as the most

Table 1: *Objective performance of phrasing models*

System	Input features	Model	True positives	False negatives	False positives	$precision$	$recall$	$F_{0.25}$	$threshold$, T
1	GPOS + Distance	CART	10099	41240	3443	0.746	0.197	0.641	0.25
2	GPOS	BiLSTM	10553	40786	2351	0.818	0.206	0.696	0.355
3	POS	BiLSTM	10139	41200	2237	0.819	0.197	0.691	0.355
4	Dep	BiLSTM	10283	41056	2145	0.827	0.2	0.699	0.365
5	w2v	BiLSTM	12745	38594	2426	0.84	0.248	0.737	0.285
6	GPOS + w2v	BiLSTM	13969	37370	3080	0.819	0.272	0.733	0.34
7	GPOS + Dep + w2v	BiLSTM	12793	38546	2378	0.843	0.239	**0.74**	0.28

promising approach. To embed context into the decision tree, we had to explicitly splice the input features. In addition to the features for the current word, the 3 previous and 3 following words in the sentence were included to generate the probability of pause insertion after a given word. The CART implementation in [22] was used in the experiments. Running several trainings, we selected the best depth of the tree based on the development set, which turned out to be a maximum leaf node occupancy of 200 examples.

Our internal toolkit for deep learning was used to train the BiLSTM. Any open-source framework, such as [23] & [24], could be used for reproducibility though. The LSTM components described in [25] were used in our experiments. The model architecture used in this investigation consists of 2 bidirectional layers, each contains 32 units with a distinct gating function. Each unit contains 64 memory cells. The output dimension of each layer is transformed to 128. After each layer, a splicing "window" component is used. It stacks 7 frames together (the current frame, 3 before and 3 after) and feeds them to the next layer. The last layer is a softmax which estimates the posterior probabilities of no break and of a respiratory break with a pause. Wittingly, large dimensions of hidden layers were used to be able to learn from both high-dimensional word embeddings and small one-hot vectors (e.g. for limited POS tags). Large batch size (6000) and small learning rate (1e-5) ensured gradual slow convergence on our relatively large dataset. After 20 training epochs, we start using an exponential learning rate scheduler of ratio 0.97. During training, the cross-entropy error was minimized using stochastic gradient descent. The training was stopped when there were no more improvements over the past 10 epochs on the development set.

5. Objective evaluations

Models with different input feature sets were trained. Results are reported on the test set in terms of precision and recall for word transitions without punctuation.

It is widely known that it is preferable to skip pauses rather than inserting them in inappropriate places. We therefore aim for models shifted towards precision. This can be described with an unbalanced F-score:

$$F_\beta = (1+\beta^2)\frac{precision \cdot recall}{(\beta^2 \cdot precision) + recall} \quad (1)$$

A value of $\beta = 0.25$ was empirically determined for this investigation. For each model, we have tuned on the development set the threshold T applied to the pause insertion posteriors in order to produce the best possible F-score.

The performance of the various systems is reported in Table 1. Firstly, we can see that the BiLSTM (*system 2*) clearly outperforms the CART baseline (*system 1*). Secondly, using more complex Penn POS tags (*system 3*) or dependency tree features (*system 4*) did not show any gain over GPOS features (*system 2*). Finally, an appreciable improvement was reached using word embeddings (*system 5*). Combining word embeddings with other linguistic features (*systems 6 & 7*), however, did not provide further improvements.

6. Subjective evaluations

During the subjective evaluation, we tried to quantitatively answer the following questions:

1. How much can be gained by introducing a phrasing module?
2. How much can be gained by using advanced statistical modelling techniques?
3. How much can be gained by using advanced feature sets?

These questions are investigated in the subsequent subsections. For each of these investigations, three methods of subjective evaluation were conducted: 1) blind listening test with naïve listeners from Amazon Mechanical Turk; 2) Listening test with 10 professional data-analysts internal to Amazon; 3) Reading test with a linguist. For each of these three methods of testing, only native speakers of English were used. Such extensive evaluation was driven by the intuition that inserting phrase breaks in different contexts and conditions have diverse weight in perception, i.e., not all errors are equally bad. This is not taken into account in objective scores.

Participants of the three different methods of subjective evaluation were all instructed to answer **"which phrasing structure sounds better?"**. Testing was performed on sentences outside of the audiobook corpus described in Section 2 to confirm the wide applicability of our models. A small subset of CELI-Text English 2013 (monolingual text corpus of modern English language) was used in evaluations. We randomly selected 1000 sentences from the non-fiction domain. Only sentences for which the systems selected different pause insertion points were used for testing. For the listening tests, text with breaks inserted by various phrasing models, was synthesized using a hybrid TTS system (statistical parametric models are used to guide unit selection). In the MTurk listening tests, each test-case received 10 responses. In the internal listening tests, each test-case received approximately 5 responses.

In the reading tests, the respiratory breaks were marked up and the linguist was asked to define which sentence has better structure (forced choice between the two systems). Only 200 of the sentences for which the systems selected different pause insertion points were assessed for each test.

Additionally they were asked to flag up where the phrase breaks inserted are absolutely inappropriate, i.e. they change the meaning of the message or the pause would never be observed in human speech. An example is shown in Figure 1.

Augustine was the prior of a monastery in Rome <break/>when Pope Gregory ***<break/>*** *the Great chose him in five hundred ninety five to lead a mission...*

Figure 1: *Example of an inappropriate break (in bold font).*

Results for this metric across the tests described in sections 6.1, 6.2 and 6.3 are reported in Table 2.

Table 2: *Percentage of inappropriate breaks reported by the linguist in each of the reading tests.*

Section 6.1	System 1	5.96%
Section 6.2	System 1	14.8%
	System 2	0.72%
Section 6.3	System 2	4.16%
	System 7	4.16%

6.1. Basic phrasing

In order to answer question 1 a deterministic approach (i.e., only inserting pauses at punctuation marks – *system DA*) was compared to a CART phrasing model with GPOS input features (*system 1*). The test was run on the 421 sentences where the phrase break predictions differed between the systems. Results are shown in Figure 2.

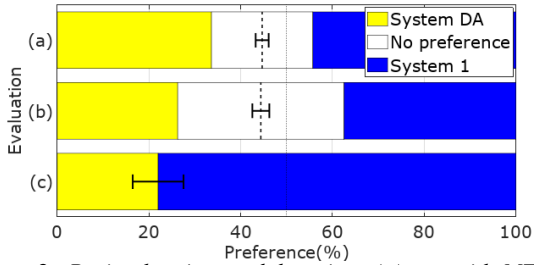

Figure 2: *Basic phrasing module gains. (a) test with MTurk, (b) test with internal listeners, (c) reading test with linguists. Whiskers show confidence interval at 95%.*

The MTurk test and the internal test both show a statistically significant (with *p-value<0.01%*) preference for synthesis with the phrasing module (*system 1*). These findings are supported by the reading test. The reported number of "inappropriate breaks" are shown in Table 2.

6.2. Advanced modelling

In order to answer question 2, the CART (*system 1*) and BiLSTM (*system 2*) phrasing models were compared. GPOS tags were used as input features for both systems. The test was run on the 353 sentences where the phrase break predictions differed between the systems. Results are shown in Figure 3.

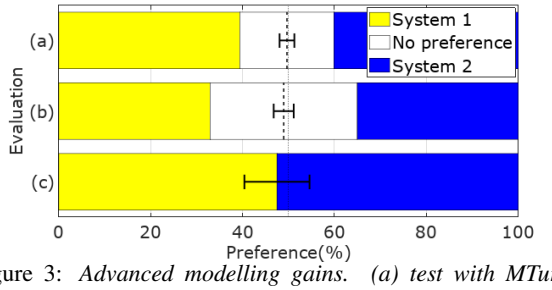

Figure 3: *Advanced modelling gains. (a) test with MTurk, (b) test with internal listeners, (c) reading test with linguists. Whiskers show confidence interval at 95%.*

Neither the MTurk nor internal tests show a statistically significant preference for either system (there may appear to be a slight tendency towards the BiLSTM, however further investigation is required to confirm this). Results from the linguist are in-line with those obtained from the listening tests. Even though gains are quite small in the overall picture, the BiLSTM make large contributions to improving the number of inappropriate breaks that it inserts (see Table 2). This is to be expected, since the CART model has much less contextual awareness than the DNN.

6.3. Advanced features

In order to answer question 3, BiLSTM phrasing models using different input features are compared (*system 2* against *system 7*). The test was run on the 323 sentences where the phrase break predictions differed between the systems. Results are shown in Figure 4. According to the objective evaluation, word embeddings is an important source of knowledge, which provides the model with a more accurate insight of the context.

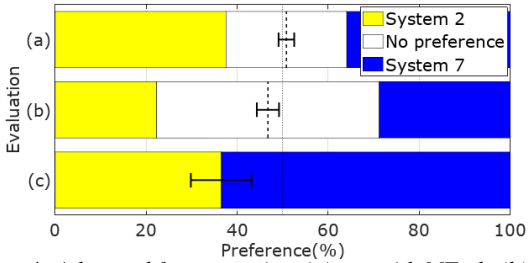

Figure 4: *Advanced features gains. (a) test with MTurk, (b) test with internal listeners, (c) reading test with linguists. Whiskers show confidence interval at 95%.*

Again the test with MTurk listeners did not produce statistically significant results. The internal test, however, showed statistically significant (with *p-value 0.5%*) preference of the model with advanced feature-set (*system 7*). The test with linguists showed that the advanced textual features doesn't provide additional gains in the number of "inappropriate breaks" (see Table 2), though the overall impression from phrasal structure seems to improve.

7. Conclusions

The phrasing models presented in this paper are a step towards achieving rich long-form reading prosody, based on audiobook data. Detailed instructions for building generic state-of-the-art phrasing models have been provided. Since punctuation is a reliable phrasal break indicator, objective and subjective performance is reported for respiratory pauses only. Incremental improvement of phrasal structure obtained by introducing BiLSTM for modelling and using word embeddings as input features is shown. The contextual awareness of BiLSTM greatly reduces the risk of introducing a break in a completely inappropriate location. More sophisticated textual features help to distinguish different word instances and their contexts more precisely. The unsupervised method of producing word embeddings is particularly attractive, compared to the large amount of manual annotation which are required for features such as POS.

Whilst we observe that there is not a strict binary preference towards adding phrase breaks on top of those derived from punctuation, these breaks make it easier to comprehend text in small word chunks. Therefore, we are interested in introducing a phrasing module into our text-to-speech system, in future we can then extend this to give the listener an opportunity to control the phrasing threshold. This can be also paired with a speaking rate setting.

In the future, we would like to invest more effort into automatic prosody annotation, which should greatly improve the quality of automatically obtained phrasing corpus. That would also give an opportunity to introduce more levels of phrase breaks and potentially improve the naturalness of prosody. Providing the model with more context is also among our future plans.

8. References

[1] P. Taylor, *Text-to-Speech Synthesis*. New York: Cambridge University Press, 2009.

[2] P. Taylor and A. W. Black, "Assigning phrase breaks from part-of-speech sequences," 1998.

[3] M. Ostendorf and N. Veilleux, "A hierarchical stochastic model for automatic prediction of prosodic boundary location," *Computational Linguistics*, vol. 20, no. 1, pp. 27–54, 1994.

[4] J. Adell, A. Bonafonte, and D. Escudero, "Filled pauses in speech synthesis: towards conversational speech," in *International Conference on Text, Speech and Dialogue*. Springer, 2007, pp. 358–365.

[5] J. Apel, F. Neubarth, H. Pirker, and H. Trost, "Have a break! modelling pauses in german speech," in *KONVENS*, 2004, pp. 5–12.

[6] A. Parlikar and A. W. Black, "Data-driven phrasing for speech synthesis in low-resource languages," in *Acoustics, Speech and Signal Processing (ICASSP), 2012 IEEE International Conference on*. IEEE, 2012, pp. 4013–4016.

[7] O. Watts, S. Gangireddy, J. Yamagishi, S. King, S. Renals, A. Stan, and M. Giurgiu, "Neural net word representations for phrase-break prediction without a part of speech tagger," in *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*. IEEE, 2014, pp. 2599–2603.

[8] T. T. Nguyen, G. Neubig, H. Shindo, S. Sakti, T. Toda, and S. Nakamura, "A latent variable model for joint pause prediction and dependency parsing," in *Sixteenth Annual Conference of the International Speech Communication Association*, 2015.

[9] A. Vadapalli and S. V. Gangashetty, "An investigation of recurrent neural network architectures using word embeddings for phrase break prediction," *Interspeech 2016*, pp. 2308–2312, 2016.

[10] M. Q. Wang and J. Hirschberg, "Automatic classification of intonational phrase boundaries," *Computer Speech & Language*, vol. 6, no. 2, pp. 175–196, 1992.

[11] A. Parlikar, "Style-specific phrasing in speech synthesis," Ph.D. dissertation, IIIT Hyderabad, India, 2013.

[12] M. Fishel and M. Mihkla, "Modelling the temporal structure of newsreaders' speech on neural networks for estonian text-to-speech synthesis," in *Proceedings of the 11th International Conference" Speech and Computer": SPECOM2006*, 2006, pp. 303–306.

[13] S. Pascual and A. Bonafonte, "Prosodic break prediction with rnns," in *Advances in Speech and Language Technologies for Iberian Languages: Third International Conference, IberSPEECH 2016, Lisbon, Portugal, November 23-25, 2016, Proceedings 3*. Springer, 2016, pp. 64–72.

[14] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: an asr corpus based on public domain audio books," in *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*. IEEE, 2015, pp. 5206–5210.

[15] K. E. Silverman, M. E. Beckman, J. F. Pitrelli, M. Ostendorf, C. W. Wightman, P. Price, J. B. Pierrehumbert, and J. Hirschberg, "Tobi: a standard for labeling english prosody." in *ICSLP*, vol. 2, 1992, pp. 867–870.

[16] A. Rosenberg, "Autobi-a tool for automatic tobi annotation." in *Interspeech*, 2010, pp. 146–149.

[17] A. Parlikar and A. W. Black, "A grammar based approach to style specific phrase prediction." in *Interspeech*, 2011, pp. 2149–2152.

[18] M. Marcus, B. Santorini, and M. Marcinkiewicz, "Building a large annotated corpus of English: the Penn Treebank," *Computational linguistics – Association for Computational Linguistics*, vol. 19, no. 2, pp. 313–330, 1993.

[19] C. D. Manning, M. Surdeanu, J. Bauer, J. R. Finkel, S. Bethard, and D. McClosky, "The stanford corenlp natural language processing toolkit." in *ACL (System Demonstrations)*, 2014, pp. 55–60.

[20] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *arXiv preprint arXiv:1301.3781*, 2013.

[21] (2013) word2vec pre-trained on news domain. [Online]. Available: https://code.google.com/archive/p/word2vec/

[22] P. Taylor, R. Caley, A. W. Black, and S. King, "Edinburgh speech tools library," *System Documentation Edition*, vol. 1, pp. 1994–1999, 1999.

[23] T. Chen, M. Li, Y. Li, M. Lin, N. Wang, M. Wang, T. Xiao, B. Xu, C. Zhang, and Z. Zhang, "Mxnet: A flexible and efficient machine learning library for heterogeneous distributed systems," *arXiv preprint arXiv:1512.01274*, 2015.

[24] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin *et al.*, "Tensorflow: Large-scale machine learning on heterogeneous distributed systems," *arXiv preprint arXiv:1603.04467*, 2016.

[25] H. Sak, A. W. Senior, and F. Beaufays, "Long short-term memory recurrent neural network architectures for large scale acoustic modeling." in *Interspeech*, 2014, pp. 338–342.