Impact of Deep Learning on Speech Synthesis with Embedded or Mobile Devices

Literature Review

Hannes Bohnengel
Technical University of Munich
hannes.bohnengel@tum.de

ABSTRACT

Multifrequency media access control has been well understood in general wireless ad hoc networks, while in wireless sensor networks, researchers still focus on single frequency solutions. In wireless sensor networks, each device is typically equipped with a single radio transceiver and applications adopt much smaller packet sizes compared to those in general wireless ad hoc networks. Hence, the multifrequency MAC protocols proposed for general wireless ad hoc networks are not suitable for wireless sensor network applications, which we further demonstrate through our simulation experiments. In this article, we propose MMSN, which takes advantage of multifrequency availability while, at the same time, takes into consideration the restrictions of wireless sensor networks. Through extensive experiments, MMSN exhibits the prominent ability to utilize parallel transmissions among neighboring nodes.

KEYWORDS

Deep Learning, Deep Neural Network, Embedded System, Mobile Device, Speech Synthesis, Text-to-Speech

1 INTRODUCTION

The *proceedings* are the records of a conference.[1] ACM seeks to give these conference by-products a uniform, high-quality appearance. To do this, ACM has some rigid requirements for the format of the proceedings documents: there is a specified format (balanced double columns), a specified set of fonts (Arial or Helvetica and Times Roman) in certain specified sizes, a specified live area, centered on the page, specified size of margins, specified column width and gutter size.

These are the core papers:

- Robust Deep-learning Models for Text-to-speech Synthesis Support on Embedded Devices [2]
- Statistical parametric speech synthesis using deep neural networks [18]
- Deep neural networks employing Multi-Task Learning and stacked bottleneck features for speech synthesis [16]
- Efficient deep neural networks for speech synthesis using bottleneck features [10]
- On the training aspects of Deep Neural Network (DNN) for parametric TTS synthesis [14]
- TTS synthesis with bidirectional LSTM based recurrent neural networks [6]
- The effect of neural networks in statistical parametric speech synthesis [8]

- Efficient memory compression in deep neural networks using coarse-grain sparsification for speech applications [11]
- Speeding up deep neural networks for speech recognition on ARM Cortex-A series processors [17]

These are interesting references:

- Deep Learning for Acoustic Modeling in Parametric Speech Generation: A systematic review of existing techniques and future trends [13]

2 THE BODY OF THE PAPER

Typically, the body of a paper is organized into a hierarchical structure, with numbered or unnumbered headings for sections, subsections, sub-subsections, and even smaller sections. The command \section that precedes this paragraph is part of such a hierarchy.[2] LATEX handles the numbering and placement of these headings for you, when you use the appropriate heading commands around the titles of the headings. If you want a sub-subsection or smaller part to be unnumbered in your output, simply append an asterisk to the command name. Examples of both numbered and unnumbered headings will appear throughout the balance of this sample document.

Because the entire article is contained in the **document** environment, you can indicate the start of a new paragraph with a blank line in your input file; that is why this sentence forms a separate paragraph.

2.1 Type Changes and *Special* Characters

We have already seen several typeface changes in this sample. You can indicate italicized words or phrases in your text with the command \textit; emboldening with the command \textbf and typewriter-style (for instance, for computer code) with \texttt. But remember, you do not have to indicate typestyle changes when such changes are part of the *structural* elements of your article; for instance, the heading of this subsection will be in a sans serif[3] typeface, but that is handled by the document class file. Take care with the use of[4] the curly braces in typeface changes; they mark the beginning and end of the text that is to be in the different typeface.

You can use whatever symbols, accented characters, or non-English characters you need anywhere in your document; you can find a complete list of what is available in the *LATEX User's Guide* [12].

---

[1]This is a footnote

[2]This is a footnote.
[3]Another footnote here. Let's make this a rather long one to see how it looks.
[4]Another footnote.

2.2 Math Equations

You may want to display math equations in three distinct styles: inline, numbered or non-numbered display. Each of the three are discussed in the next sections.

2.2.1 Inline (In-text) Equations.

A formula that appears in the running text is called an inline or in-text formula. It is produced by the **math** environment, which can be invoked with the usual \begin . . . \end construction or with the short form $ . . . $. You can use any of the symbols and structures, from α to ω, available in LaTeX [12]; this section will simply show a few examples of in-text equations in context. Notice how this equation: $\lim_{n\to\infty} x = 0$, set here in in-line math style, looks slightly different when set in display style. (See next section).

2.2.2 Display Equations.

A numbered display equation—one set off by vertical space from the text and centered horizontally—is produced by the **equation** environment. An unnumbered display equation is produced by the **displaymath** environment.

Again, in either environment, you can use any of the symbols and structures available in LaTeX; this section will just give a couple of examples of display equations in context. First, consider the equation, shown as an inline equation above:

$$\lim_{n\to\infty} x = 0 \tag{1}$$

Notice how it is formatted somewhat differently in the **displaymath** environment. Now, we'll enter an unnumbered equation:

$$\sum_{i=0}^{\infty} x + 1$$

and follow it with another numbered equation:

$$\sum_{i=0}^{\infty} x_i = \int_0^{\pi+2} f \tag{2}$$

just to demonstrate LaTeX's able handling of numbering.

2.3 Citations

Citations to articles [3–5, 9], conference proceedings [5] or maybe books [12, 15] listed in the Bibliography section of your article will occur throughout the text of your article. You should use BibTeX to automatically produce this bibliography; you simply need to insert one of several citation commands with a key of the item cited in the proper location in the .tex file [12]. The key is a short reference you invent to uniquely identify each work; in this sample document, the key is the first author's surname and a word from the title. This identifying key is included with each item in the .bib file for your article.

The details of the construction of the .bib file are beyond the scope of this sample document, but more information can be found in the *Author's Guide*, and exhaustive details in the *LaTeX User's Guide* by Lamport [12].

This article shows only the plainest form of the citation command, using \cite.

2.4 Tables

Because tables cannot be split across pages, the best placement for them is typically the top of the page nearest their initial cite. To ensure this proper "floating" placement of tables, use the environment **table** to enclose the table's contents and the table caption. The contents of the table itself must go in the **tabular** environment, to be aligned properly in rows and columns, with the desired horizontal and vertical rules. Again, detailed instructions on **tabular** material are found in the *LaTeX User's Guide*.

To set a wider table, which takes up the whole width of the page's live area, use the environment **table*** to enclose the table's contents and the table caption. As with a single-column table, this wide table will "float" to a location deemed more desirable. Immediately following this sentence is the point at which Table 1 is included in the input file; again, it is instructive to compare the placement of the table here with the table in the printed output of this document.

Table 1: Some Typical Commands

Command	A Number	Comments
\author	100	Author
\table	300	For tables
\table*	400	For wider tables

It is strongly recommended to use the package booktabs [7] and follow its main principles of typography with respect to tables:

(1) Never, ever use vertical rules.
(2) Never use double rules.

It is also a good idea not to overuse horizontal rules.

2.5 Figures

Like tables, figures cannot be split across pages; the best placement for them is typically the top or the bottom of the page nearest their initial cite. To ensure this proper "floating" placement of figures, use the environment **figure** to enclose the figure and its caption.

This sample document contains examples of .eps files to be displayable with LaTeX. If you work with pdfLaTeX, use files in the .pdf format. Note that most modern TeX systems will convert .eps to .pdf for you on the fly. More details on each of these are found in the *Author's Guide*. This is a reference to a figure (see Figure 1).
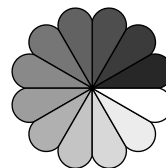

Figure 1: A sample black and white graphic that has been resized with the `includegraphics` command.

As was the case with tables, you may want a figure that spans two columns. To do this, and still to ensure proper "floating" placement of tables, use the environment **figure*** to enclose the figure and its

caption. And don't forget to end the environment with **figure***, not **figure**!

2.6 Theorem-like Constructs

Other common constructs that may occur in your article are the forms for logical constructs like theorems, axioms, corollaries and proofs. ACM uses two types of these constructs: theorem-like and definition-like.

Here is a theorem:

THEOREM 2.1. *Let f be continuous on $[a,b]$. If G is an antiderivative for f on $[a,b]$, then*

$$\int_a^b f(t)\,dt = G(b) - G(a).$$

Here is a definition:

Definition 2.2. If z is irrational, then by e^z we mean the unique number that has logarithm z:

$$\log e^z = z.$$

The pre-defined theorem-like constructs are **theorem**, **conjecture**, **proposition**, **lemma** and **corollary**. The pre-defined definition-like constructs are **example** and **definition**. You can add your own constructs using the *amsthm* interface [1]. The styles used in the \theoremstyle command are **acmplain** and **acmdefinition**.

Another construct is **proof**, for example,

PROOF. Suppose on the contrary there exists a real number L such that

$$\lim_{x \to \infty} \frac{f(x)}{g(x)} = L.$$

Then

$$l = \lim_{x \to c} f(x) = \lim_{x \to c}\left[gx \cdot \frac{f(x)}{g(x)}\right] = \lim_{x \to c} g(x) \cdot \lim_{x \to c} \frac{f(x)}{g(x)} = 0 \cdot L = 0,$$

which contradicts our assumption that $l \neq 0$. □

3 CONCLUSIONS

This paragraph will end the body of this sample document. Remember that you might still have Acknowledgments or Appendices; brief samples of these follow. There is still the Bibliography to deal with; and we will make a disclaimer about that here: with the exception of the reference to the LaTeX book, the citations in this paper are to articles which have nothing to do with the present subject and are used as examples only.

REFERENCES

[1] American Mathematical Society 2015. *Using the amsthm Package.* American Mathematical Society. http://www.ctan.org/pkg/amsthm.

[2] Tiberiu Boroş and Stefan Daniel Dumitrescu. 2015. Robust Deep-learning Models for Text-to-speech Synthesis Support on Embedded Devices. In *Proceedings of the 7th International Conference on Management of Computational and Collective intElligence in Digital EcoSystems (MEDES '15)*. ACM, New York, NY, USA, 98–102. https://doi.org/10.1145/2857218.2857234

[3] Mic Bowman, Saumya K. Debray, and Larry L. Peterson. 1993. Reasoning About Naming Systems. *ACM Trans. Program. Lang. Syst.* 15, 5 (November 1993), 795–825. https://doi.org/10.1145/161468.161471

[4] Johannes Braams. 1991. Babel, a Multilingual Style-Option System for Use with LaTeX's Standard Document Styles. *TUGboat* 12, 2 (June 1991), 291–301.

[5] Malcolm Clark. 1991. Post Congress Tristesse. In *TeX90 Conference Proceedings*. TeX Users Group, 84–89.

[6] Yuchen Fan, Yao Qian, Feng-Long Xie, and Frank K. Soong. 2014. TTS synthesis with bidirectional LSTM based recurrent neural networks. In *INTERSPEECH 2014, 15th Annual Conference of the International Speech Communication Association, Singapore, September 14-18, 2014.* 1964–1968. http://www.isca-speech.org/archive/interspeech_2014/i14_1964.html

[7] Simon Fear. 2005. *Publication quality tables in LaTeX.* http://www.ctan.org/pkg/booktabs.

[8] K. Hashimoto, K. Oura, Y. Nankaku, and K. Tokuda. 2015. The effect of neural networks in statistical parametric speech synthesis. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP).* 4455–4459. https://doi.org/10.1109/ICASSP.2015.7178813

[9] Maurice Herlihy. 1993. A Methodology for Implementing Highly Concurrent Data Objects. *ACM Trans. Program. Lang. Syst.* 15, 5 (November 1993), 745–770. https://doi.org/10.1145/161468.161469

[10] Y. S. Joo, W. S. Jun, and H. G. Kang. 2016. Efficient deep neural networks for speech synthesis using bottleneck features. In *2016 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA).* 1–4. https://doi.org/10.1109/APSIPA.2016.7820721

[11] D. Kadetotad, S. Arunachalam, C. Chakrabarti, and Jae sun Seo. 2016. Efficient memory compression in deep neural networks using coarse-grain sparsification for speech applications. In *2016 IEEE/ACM International Conference on Computer-Aided Design (ICCAD).* 1–8. https://doi.org/10.1145/2966986.2967028

[12] Leslie Lamport. 1986. *LaTeX User's Guide and Document Reference Manual.* Addison-Wesley Publishing Company, Reading, Massachusetts.

[13] Z. H. Ling, S. Y. Kang, H. Zen, A. Senior, M. Schuster, X. J. Qian, H. M. Meng, and L. Deng. 2015. Deep Learning for Acoustic Modeling in Parametric Speech Generation: A systematic review of existing techniques and future trends. *IEEE Signal Processing Magazine* 32, 3 (May 2015), 35–52. https://doi.org/10.1109/MSP.2014.2359987

[14] Y. Qian, Y. Fan, W. Hu, and F. K. Soong. 2014. On the training aspects of Deep Neural Network (DNN) for parametric TTS synthesis. In *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP).* 3829–3833. https://doi.org/10.1109/ICASSP.2014.6854318

[15] S.L. Salas and Einar Hille. 1978. *Calculus: One and Several Variable.* John Wiley and Sons, New York.

[16] Zhizheng Wu, Cassia Valentini-Botinhao, Oliver Watts, and Simon King. 2015. Deep neural networks employing Multi-Task Learning and stacked bottleneck features for speech synthesis. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP).* 4460–4464. https://doi.org/10.1109/ICASSP.2015.7178814

[17] A. Xing, X. Jin, T. Li, X. Wang, J. Pan, and Y. Yan. 2014. Speeding up deep neural networks for speech recognition on ARM Cortex-A series processors. In *2014 10th International Conference on Natural Computation (ICNC).* 123–127. https://doi.org/10.1109/ICNC.2014.6975821

[18] Heiga Zen, Andrew Senior, and Mike Schuster. 2013. Statistical parametric speech synthesis using deep neural networks. In *2013 IEEE International Conference on Acoustics, Speech and Signal Processing.* 7962–7966. https://doi.org/10.1109/ICASSP.2013.6639215