UNIVERSAL SPEECH TRANSLATOR

[1*]*Sai Prakash R,* [2]*Ayesha ML,* [3] *Srusti,* [4] *Divya MO*

Kristu Jayanti College (Autonomous), Bengaluru, Karnataka, India.

Abstract

In summary, Meta explains that this all-in-one translator is "pushed via way of means of the purpose of breaking down language boundaries on an international scale" the use of system gaining knowledge of technology. Since a mechanical perception, "This is a provisional computational prototype based on the Thinly Gated Mixture of Experts, qualified on data gained using novel and effective data mining methods personalized to low-resource languages. It is a model that forms the basis of universal translation. In the system, BLEU (short for Bilingual Evaluation Understudy).

1. Introduction

"People have always desired the superpower of being able to speak with anyone in any language, and AI will grant us that power within our lifetimes". said Zuckerberg. At the occasion, Zuckerberg disclosed that Meta intends to build the idea in two stages. The "No Language Left Behind" programme is the first step in developing AI models that can learn to translate languages with fewer training samples and inputs.

We're developing a single model that one can translate hundreds of languages with cutting-edge outcomes and the majority of language pairs, from Austrian to Ugandan to Urdu.

A local approach only looks at a subset of source words at a time, whereas a global approach only looks at all source words[1].

The goal of the project, as suggested by its name, is to close the gap between the options available for less commonly used native languages and the major languages like English, Mandarin, and Spanish in terms of access to cutting-edge translation technologies. In this final category, Meta still wants to include the billions of humans who live on the planet.

The network learns to translate speech into target speech in another language, corresponding to the translated content, end-to-end. It is used to demonstrate the ability to translate speech using the voice of the source speaker[4].

In recent years, these networks have become the most sophisticated models for various machine learning problems. This has sparked renewed interest in understanding the roles and usefulness of different computational components in typical LSTM variants[5].

We are investigating various structural and optimization improvements to the LAS model that significantly improve performance. Structurally, we show that word piece models can be used instead of graphemes. It also introduces a multi-headed attention architecture that improves on the commonly used single-headed attention [8].

The language model used is a bigram or a common m-gram model. The translation model is decomposed into a lexical model and an alignment model. We describe two different approaches to statistical translation and present experimental results[11].

The concept of word-by-word alignment is defined. Each model assigns a probability to each of the possible word-by-word alignments [2].

1.1 Meta AI

The study of meta-learning is one of the machine learning research fields that is expanding the quickest. In machine-learning, meta-learning refers to the use of machine-learning algorithms to support the training and improvement of other machine-learning models.

Understanding what meta-learning is and being aware of the numerous applications it may be put to use are both advantageous as meta-learning grows in popularity and the number of meta-learning approaches increases. Let's look at the concepts behind meta-learning, different forms of meta-learning, and possible applications for meta-learning. / Donald Maudsley first used the word "meta-learning" to refer to the process by which people start to influence what they learn, taking more and more "control of habits of observation, inquiry, learning, and growth that they have internalized."

Figure 1. Devices supporting meta learning

Later, psychologists and cognitive scientists would refer to meta-learning as "learning how to learn." The concept of "learning how to learn" is applied to AI systems in the machine learning version of meta-learning. The ability of an artificially intelligent computer to learn how to execute a variety of complex tasks by applying the same concepts it used to master one task to other tasks is known as meta-learning in the context of AI. In order to complete a task, AI systems often need to be trained to master a number of minor subtasks. It can take a while AI agent find it difficult to apply learned skills to new situations.

A Mel-spectrogram-tuned WaveNet vocoder is constructed to reconstruct the waveform from the output of the SCENT model. It is noteworthy that the proposed method can realize appropriate duration conversion, which was difficult with conventional methods[6].

Developing meta-learning models and approaches can generalize the learning process and accelerate AI's ability to learn new skills. Different kinds of Meta-Learning, Optimizers Meta-Learning is rarely used to improve the performance of existing neural networks. As a result, the target network bequeaths be better

able to perform the task it is being trained on. Applying a network to improve the results of gradient descent is an example of a meta-learning optimizer.

meta-learning in to complete this training, and AI agents find it difficult to apply the skills they have acquired to new situations' meta-learning in to complete this training, and AI agents find it difficult to apply the skills they have acquired to new situations. It is possible to accelerate AI's ability to generalize learning processes and learn new skills by developing meta-learning models and approaches.

1.2 Kinds of Meta-Learning:

I. Few-Shots Meta-Learning
II. Metric Meta-Learning
III. Recurrent Model Meta-Learning

2. How Does Meta-Learning Work?

The working of meta-learning based on the type of task been assigned to the model. But in common, the meta-learning involved copying the model parameters of the initial network to the 2nd network. In meta-learning he has two learning processes. A meta-learning model naturally trained subsequently after having enough practice/experience on base model. Later the forward, backward, and optimization stages of exercise the base model, a forward exercise clearance of the optimization model is accomplished. For example, after 3 or 4 exercise steps on the base model, meta-loss is figured.

A meta optimizer is one more meta He could be a learner, but at some point, he should use a separate optimizer like ADAM or SGD.

Numerous deep learning prototypes can have hundreds of thousands, even lots, of constraints. Generating a meta-learner with a completely new set of parameters is computationally expensive, so a tactic known as coordinate sharing is commonly used. Coordinate sharing means that from the base model he learns one parameter and builds a meta-his learner/optimizer that replicates only that parameter and not all other parameters. As a result, the parameters that the optimizer has do not depend on the parameters of the model.

Since the factors in the output set are assumed to be fixed, there is no decoding step that creates output factors from other output factors. However, Moses can have ambiguous inputs in the form of confusion networks[12].

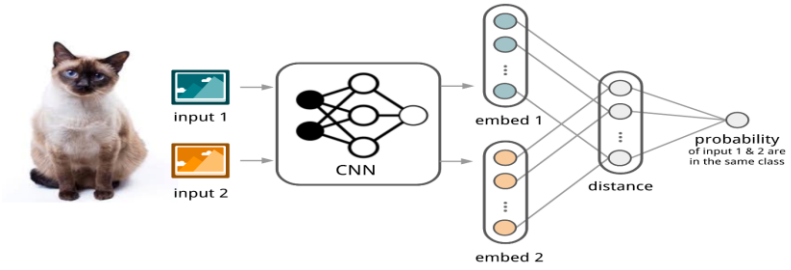

Figure 2. Meta learning process flow

2.1 Meta Intends to Create AI-powered 'Universal Speech Translator'

Figure 3. Universal speech translation enabling better communication

Meta announced on February 23rd that as part of its efforts to build the Metaverse, it plans to build an artificial intelligence (AI)-based translation system that works for anyone in the world, including an ambitious universal language translator.

Meta co-founder Mark Zuckerberg predicts that the Metaverse, an inherently more immersive version of the internet as we know it today, will be the successor to the mobile internet.

On February 23, Meta announced plans to build an artificial intelligence (AI)-based translation system that works for anyone in the world, including an aspiring universal language translator, as part of its efforts to build the Metaverse.

Meta co-founder Mark Zuckerberg predicts that the Metaverse, an inherently more immersive version of the Internet as we know it today, will be the successor to the mobile Internet.

2.2 New digital assistants for the metaverse

Creating a meta-learner with a completely new set of parameters is computationally expensive, so a tactic known as coordinate sharing is commonly used. "These worlds are dynamic and ever-changing, so AI must be able to understand context and learn like humans do.

Meta also announced a new initiative called Project CAIRaoke. This is an end-to-end neural model for building on-device assistants that enable people to have more natural and contextual conversations with voice assistants.

The advantages and drawbacks of statistical parametric synthesis are highlighted and we see where we expect key developments to appear in the immediate future[3].

2.3 Real-Time Translator Apps

Figure 4. Importance of Real time translator apps

Top Real-Time translator apps

1. iTranslate app
2. iTranslate Voice 3
3. Easy language translator
4. Google translate

3. Machine learning libraries and platforms

Python

Python has rich libraries for computing complex databases.

Google Cloud and Colab

Google cloud and colab also similar like python only. With these we can straight away without any installation and storage allocation because in these all resources are configured we just need to call the service and start using.

It present a general end-to-end approach to sequence learning that makes minimal assumptions about sequence structure. Our method uses a multi-layer long short-term memory (LSTM) to map the input sequence to a fixed-dimensional vector, then another deep LSTM to decode the target sequence from the vector[9].

4. Results and Discussion

End-to-end language translation (ST) models have many potential advantages over automatic speech recognition (ASR) and machine text translation (MT) models, such as reduced inference latency and mismatch avoidance [7].

This communication technology can be used between computers and humans. This interaction takes place through interfaces and this area is called Human Computer Interaction (HCI). This paper provides an overview of the main definitions of Automatic Speech Recognition (ASR). This is an important area of artificial intelligence and should be considered in related research (language types, vocabulary size, etc.)[10].

Building a Speech Translator using Python

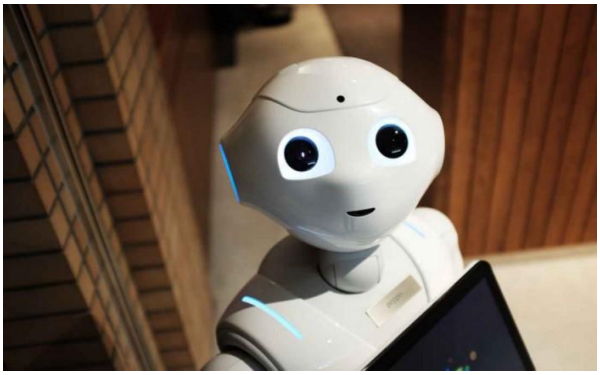

Figure 5. Results of Universal speech translator

First, we need to Import Required Libraries

Let's install the required modules that we will be using in this python program, Pyttsx3, Googletrans and SpeechRecognition modules. SpeechRecognition and Googletrans modules these are created by Google. All these modules are free to download and use. The pip code installs multiple modules:

```
pip install speechrecognition pyttsx3 googletrans
```

1.Speech Recognition

Define Recognition

```
recg = src.Recognizer()
```

Microphone Definition

Define microphone instance, select input device. Here might be numerous input devices connected to the computer system and let's choose which device we use. Below code recognize the connect input device.

```
print(src.Microphone.list_microphone_names())
```

The results show the list of connected devices. Microphone recognize code is follows:

```
microphone = src.Microphone(device_index=0)
```

Recognize Speech

we use *recognize_google* technique, it is created by Google.

```
microphone as source:

        recg.adjust_for_ambient_noise(source)
        audio_device = recg.listen(source)

results = recg.recognize_google(audio)
```

with following code you can check results.

```
print(results)
```

2. Building the Translator

Define Translator

You will have a lot of options to choose from. what language do you want choose from a bunch of different languages?

Now select the language:

```
translator = Translator()
key_lang = translator.translate(results, dest='french')
```

In the code below, we will convert the translated result into a text format, this will help our text to speech module to work properly. Because previous code, the result was stored as an object.

```
translated = str(key_lang.text)
print(translated)
```

3. Text to Speech Engine

Define Text to Speech Engine

```
engine_res = pyttsx3.init()
```

We've just defined the module as an engine. Now is the time to tell our program to speak the translated text, but before that, we have to define the language. Here is the code to see the list of languages and their IDs, which we will need when we are defining the translator. We recommend running this code in your terminal before going to the next step.

```
engine_res = pyttsx3.init()
voices = engine_res.getProperty('voices')
```

```
for voice in voices:
print("Voice:")
print(" - ID: %s" % voice.id)
print(" - Name: %s" % voice.name)
print(" - Languages: %s" % voice.languages)
print(" - Gender: %s" % voice.gender)
print(" - Age: %s" % voice.age)
```

Define Speaker Language

Copy the ID of the language that you want to use, and let's paste it into our program. We are using setProperty method to define the speaker language.

```
fr_voice_id = "com.apple.speech.synthesis.voice.thomas"
engine_res.setProperty('voice', fr_voice_id)
```

Final Step:

```
engine_res.say(translated)
engine_res.runAndWait()
```

Conclusion

In this paper, we have considered on how to translate the speech into a different language using Python. On recording the speech, a speech translator will listen to it, understand what you are saying, and translate it into the language of your choice. You can modify your code when the translation process is finished so that it displays the translated text. This is a great project that you will enjoy working on and maybe even impress your friends and family. Imagine when you want to communicate with someone from a different country, you have to hire someone that can speak both languages and do all the translating. That was the situation in ancient times, but afterwards the progress of the internet and tools, communication has developed very easily. Using this method, we can build our own translator which can speaks many languages.

References

[1] T. Luong, H. Pham, and C. D. Manning, "Effective approaches to attention-based neural machine translation," in Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, (Lisbon, Portugal), pp. 1412–1421, Association for Computational Linguistics, Sept. 2015.

[2] P. Brown, S. Della Pietra, V. Pietra, and R. Mercer, "The mathematics of statistical machine translation: Parameter estimation," Computational Linguistics, vol. 19, pp. 263–311, 01 1993.

[3] A. W. Black, H. Zen, and K. Tokuda, "Statistical parametric speech synthesis," in 2007 IEEE International Conference on Acoustics, Speech and Signal Processing - ICASSP '07, vol. 4, pp. IV–1229–IV–1232, 2007.

[4] Y. Jia, R. J. Weiss, F. Biadsy, W. Macherey, M. Johnson, Z. Chen, and Y. Wu, "Direct speech-to-speech translation with a sequence-to-sequence model," CoRR, vol. abs/1904.06037, 2019.

[5] K. Greff, R. K. Srivastava, J. Koutník, B. R. Steunebrink, and J. Schmidhuber, "Lstm: A search space odyssey," IEEE Transactions on Neural Networks and Learning Systems, vol. 28, no. 10, pp. 2222–2232, 2017.

[6] J. Zhang, Z. Ling, L. Liu, Y. Jiang, and L. Dai, "Sequence-to-sequence acoustic modeling for voice conversion," CoRR, vol. abs/1810.06865, 2018.

[7] Y. Jia, M. Johnson, W. Macherey, R. J. Weiss, Y. Cao, C. Chiu, N. Ari, S. Laurenzo, and Y. Wu, "Leveraging weakly supervised data to improve end-to-end speech-to-text translation," CoRR, vol. abs/1811.02050, 2018.

[8] C. Chiu, T. N. Sainath, Y. Wu, R. Prabhavalkar, P. Nguyen, Z. Chen, A. Kannan, R. J. Weiss, K. Rao, K. Gonina, N. Jaitly, B. Li, J. Chorowski, and M. Bacchiani, "State-of-the-art speech recognition with sequence-to-sequence models," CoRR, vol. abs/1712.01769, 2017.

[9] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in Advances in Neural Information Processing Systems 27 (Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, eds.), pp. 3104–3112, Curran Associates, Inc., 2014.

[10] S. Benk, Y. Elmir, and A. Dennai, "A study on automatic speech recognition," vol. 10, pp. 77–85, 08 2019.

[11] F. J. Och, C. Tillmann, and H. Ney, "Improved alignment models for statistical machine translation," in 1999 Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora, 1999.

[12] P. Koehn, H. Hoang, A. Birch, C. Callison-Burch, M. Federico, N. Bertoldi, B. Cowan, W. Shen, C. Moran, R. Zens, C. Dyer, O. Bojar, A. Constantin, and E. Herbst, "Moses: Open source toolkit for statistical machine translation," in Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions, (Prague, Czech Republic), pp. 177–180, Association for Computational Linguistics, June 2007.