Multimodel Exploration of Neural Machine Translation: English to French

Course Project for Machine Learning Deep Learning (ELEC825)

April 2025

Jiayi Zhu
20463909

Juanchao Pan
20494892

Maahum Khan
20232476

Tomi Kofo-Alada
20152073

Xingwei Gu
20452395

Abstract

This project explores the comparative performance of various deep learning models for English-to-French translation. We implemented and evaluated four architectures: a decoder-only GPT-2 transformer (trained from scratch and fine-tuned), an RNN-based sequence-to-sequence (Seq2Seq) model with attention, a Transformer encoder-decoder model, and a multilingual pretrained mBART model. Each model was trained on a subset of a parallel English-to-French Wikimedia corpus and evaluated using the WMT14 test set. Our findings demonstrate that while simpler architectures like the RNN-Seq2Seq model offer intuitive implementation and decent performance with limited data, pretrained transformer models, particularly mBART, consistently outperform others in fluency and grammar due to their large-scale multilingual training. We analyze the strengths and limitations of each approach and provide insight into the impact of architecture, training strategy, and data efficiency on translation quality. Our results highlight the importance of pretrained multilingual knowledge and transformer-based attention mechanisms in achieving high-quality neural machine translation under resource-constrained settings.

1. Introduction

Despite major advancements in machine translation, achieving fluent and accurate translation between English and French remains a non-trivial task, especially when computational resources and dataset sizes are limited. Many state-of-the-art models are trained on vast corpora using extensive compute power, making them difficult to replicate in academic or resource-constrained environments. Moreover, not all translation models generalize well when shifted to new domains or fine-tuned on smaller, task-specific datasets. The objective of this project is to evaluate and compare the performance of four different deep learning architectures-GPT-2 (from scratch and fine-tuned), RNN-based Seq2Seq, Transformer, and mBART-for English-to-French translation. This comparison aims to provide insight into the trade-offs between training from scratch versus fine-tuning and between pretrained versus non-pretrained models under realistic computational constraints.

2. Methodology

To investigate the effectiveness of different translation strategies, we implemented four deep learning models for English-to-French translation. These include: (1) a GPT-2-style transformer trained from scratch and a version fine-tuned from a pretrained checkpoint, (2) a traditional RNN-based sequence-to-sequence model with attention, (3) a transformer-based encoder-decoder model, and (4) a fine-tuned version of the multilingual pretrained mBART model.

2.1. GPT-2 (From Scratch and Fine-Tuned) Models

This project first explored English-to-French translation using a decoder-only transformer architecture modelled after GPT. Two training strategies were explored: training from scratch and fine-tuning an existing pretrained model. Training from scratch involved building the model architecture manually, including embedding layers, positional encodings, normalization, and a stack of decoder blocks with causal attention masking. In this configuration, the model relied entirely on the parallel corpus to learn linguistic patterns, lexical correspondences, and syntactic transformations. However, due to data and resource constraints, this approach struggled with generalization and ultimately failed to produce meaningful output.

As an alternative, the project pivoted to a fine-tuning approach, using the publicly available GPT-2 model from Hugging Face's `openai-community/gpt2` repository as a starting point [4]. This approach leveraged the pretrained model's internal language representations and adapted them to the translation task by injecting translation-specific tokens and prompting patterns. This format for all training data in both methods was consistent, using the string pattern:

```
Translate English to French:
```

```
{src_text} [SEP] {tgt_text}
```

to clearly separate source and target languages.

The model trained from scratch was configured with 12 decoder layers, a hidden size of 768, 12 attention heads, a feedforward size of 3072, and a maximum sequence length of 512 tokens. A dropout rate of 0.2 was used throughout the decoder layers, with GeLU (Gaussian Error Linear Units) activation functions. The tokenizer was adapted from EleutherAI's GPT-Neo-125M, with the addition of two special tokens: `[SEP]` for separating input and output, and `[PAD]` for padding to fixed sequence lengths.

For fine-tuning, GPT-2 was used without modification to the architecture, though `[SEP]` was added to the tokenizer and the end-of-sequence token was re-used as a padding token.

2.2. RNN-Seq2Seq Model

The second model implemented for the translation task was an RNN-based sequence-to-sequence (Seq2Seq) architecture with attention, inspired by the work of Bahdanau et al[1]. Unlike pretrained transformer models, this architecture was implemented entirely from scratch. It consists of three main components: a bidirectional GRU encoder, a GRU decoder, and a Bahdanau attention mechanism.

The encoder processes the input sentence using a bidirectional GRU, allowing it to capture both past and future context in the source language. Each token is first embedded into a dense vector of dimension 256. The GRU hidden size is set to 512, and the outputs from both directions of the encoder are concatenated and passed to the attention mechanism. The decoder uses a unidirectional GRU that generates output tokens one step at a time, attending to the encoder's hidden states via the Bahdanau-style addictive attention. This mechanism dynamically computes a context vector for each output token, allowing the decoder to focus on relevant parts of the input at each step.

The attention mechanism, decoder, and encoder were combined into a full Seq2Seq model without relying on external libraries for architecture. Tokenization was handled using a shared BPE tokenizer trained on both English and French text, with a combined vocabulary size of 30,000. This shared tokenizer reduced parameter count and encouraged cross-lingual consistency. The final model had approximately 60 million parameters.

2.3. Transformer Model

The third model implemented was the Transformer, an influential architecture introduced by Vaswani et al. that relies on attention mechanisms to significantly improve the ability to understand long-range context in sequential data [5].

To prepare the data, the SentencePiece tokenizer was used for both English and French inputs [2]. SentencePiece is a subword tokenizer that addresses the open-vocabulary problem by breaking down words into subword units, making the model more robust to rare or unseen words. The vocabulary size was set to 50,000, and a special `<pad>` token was included to support dynamic padding during batch processing.

Since the dataset contained sentences of varying lengths—some exceeding 1,000 tokens—a dynamic batching approach was implemented. A custom collate function ensured that each batch was only padded to the maximum sequence length within that batch, which improved memory efficiency and training speed. Each input sequence was first converted into embeddings, which were combined with sinusoidal positional encodings [5].

Positional encoding is a key component for learning positional information in Transformer-based models. It generates positional features using a set of preset sine and cosine functions, where every two adjacent dimensions (e.g., the $2i$-th and $(2i+1)$-th dimensions) share the same frequency. Notably, the frequency decreases exponentially with increasing dimension, enabling the model to capture both fine-grained short-range dependencies and broad long-range relationships. In our Transformer model, positional encoding was implemented as a standalone module and registered as a buffer to ensure it would not be updated during training.

The model followed a standard setup: 8 layers in both the encoder and decoder, with multi-head attention, layer normalization, and feedforward layers. A Pre-Norm configuration was used, applying layer normalization before each sub-layer to improve training stability in deeper networks. The decoder included masked self-attention to prevent it from accessing future tokens during decoding. Two types of masking were applied: one for padding (to ignore `<pad>` tokens) and one for causality (to enforce autoregressive decoding).

2.4. mBART Model

The final model explored in this project was mBART, a multilingual encoder-decoder transformer architecture designed for sequence-to-sequence generation tasks such as translation[3]. Initially, BERT was considered for this task, but its encoder-only structure and inability to generate output sequences made it unsuitable. In contrast, mBART includes both an encoder and a decoder, enabling it to generate fluent translations in an autoregressive manner while conditioning on the source language.

This project used the pre-trained `facebook/mbart-large-50-many-to-many-mmt` model from Hugging Face Transformers, which was originally trained using a denoising objective on large-scale multilingual corpora spanning 50 languages. mBART allows for explicit specification of source and target languages via language codes (e.g., en_XX, fr_XX), enabling controlled multilingual translation.

Tokenization was handled using mBART's SentencePiece-based tokenizer, which segments text into subword units and appends the appropriate language tokens to the input and output sequences. The model was fine-tuned on parallel English–French sentence pairs using teacher forcing, where the decoder was trained to predict the next token given the ground truth context. The core architecture remained unchanged during fine-tuning, retaining mBART's default decoder attention masking and positional encoding mechanisms.

3. EXPERIMENT SETUP

Each model was trained and evaluated under different computational constraints and environments, using variations in data size, batch size, learning rates, and optimization strategies tailored to their architecture. To ensure fair comparison, all models were trained on subsets of the same parallel English-French corpus derived from Wikimedia and evaluated on the WMT14 English–French test set. Particular attention was given to resource efficiency and model stability, with experiments adjusted according to available hardware, ranging from MacBooks to Colab GPUs.

3.1. GPT-2 (From Scratch and Fine-Tuned) Setup

For the GPT-2 model (both the one built from scratch and the fine-tuned GPT-2 model), the training corpus used was a subset of the Wikimedia-20230407 English-French dataset, containing approximately 1.4 million aligned sentence pairs. Given time and hardware constraints (only being able to train on an Apple Silicon M3 Max Macbook Pro which would take over 15 days to train from scratch once on the entire dataset), only one-fifth of the dataset was used for both training from scratch and fine-tuning. For evaluation, the WMT14 benchmark dataset was used via Hugging Face's datasets library, and SacreBLEU was employed for consistent metric evaluation. Training for the model built from scratch used a batch size of 16, a peak learning rate of 4e-4 following a one-cycle schedule, and three epochs of training. Mixed precision (bfloat16) and a dynamic learning rate scheduler were employed to improve efficiency and mitigate memory constraints. For the fine-tuned GPT-2 model, the training configuration remained similar: three epochs, batch size of 16, learning rate of 5e-5, and 1000 warm-up steps. Mixed precision and dynamic learning rate scheduling were again applied.

3.2. RNN-Seq2Seq Setup

The RNN-Seq2Seq-Attention model was trained on Google Colab with an L4 GPU. The training dataset consisted of the English-French Wikipedia corpus, with the first 80% used for training and the remaining 20% for validation. The WMT14 test set was used for the final evaluation. Key hyperparameters included a 356-dimensional embedding size, 512 hidden units in both encoder and decoder, and a batch size of 64. The model was trained using the Adam optimizer and CrossEntropyLoss, with a dropout rate of 0.3. During training, teacher forcing was used. To mitigate overfitting, a staged training strategy was used: 100% of the data was used in epoch 1, followed by 60%, 40%, and 20% in subsequent epochs. This reduced total training time while maintaining model performance. Additional experiments were conducted to address a common error where the model repeated the token "the" excessively. These included modifying the loss function, introducing coverage-based attention, and penalizing tokens during decoding.

3.3. Transformer Setup

The Transformer model was trained from scratch using a parallel English–French corpus obtained from Wikimedia, while the WMT14 validation set served to tune hyperparameters and evaluate generalization. We trained the model on an NVIDIA RTX 4080 with 12GB of VRAM. Our main hyperparameters included a hidden size of 512, feedforward size of 2048, batch size of 32, sequence length of 256, and learning rate of 1e-4. We originally planned to train for 30 epochs but stopped at epoch 3, once validation loss started to rise while training loss continued to drop-classic overfitting.

3.4. mBART Setup

The mBART model was fine-tuned using a subset of 20,000 English–French sentence pairs from the Wikimedia corpus, selected for its manageable size under limited computational resources. All training was conducted on Google Colab Pro using a T4 GPU with 16 GB of VRAM. The model was trained for 3 epochs using a batch size of 2 with gradient accumulation to simulate a larger batch size, a learning rate of 3e-5, and the AdamW optimizer. Mixed precision (fp16) training was enabled to reduce memory usage and accelerate computation. For evaluation, a 100-sample and a 1000-sample subset from the WMT14 English–French test set were used, and translation quality was assessed using corpus-level BLEU scores calculated via SacreBLEU.

4. Result and Analysis

In this section, we present a comparative analysis of the translation performance across all four models. We evaluate both quantitative metrics, such as BLEU scores, and qualitative aspects of the generated translations. Additionally, we explore training dynamics, such as convergence behavior and generalization, and highlight model-specific behaviors, including repetition, token overgeneration, and syntactic fluency.

4.1. GPT-2 (From Scratch and Fine-Tuned)

The training-from-scratch experiment was ultimately unsuccessful. Despite the model's capacity and structurally correct implementation, its outputs were degenerate and semantically void—it frequently repeated the same word in French (such as *"de"* or *"à"*) dozens of times in a row, regardless of the English input. This behavior is consistent with both overfitting on limited data and a failure to learn effective token-level dependencies in a low-data, high-parameter regime. The model may also have struggled with the length of its input sequences relative to the amount of meaningful supervision it received, particularly given that the task required it to generate target-side content without prior pretraining.

In contrast, the fine-tuned GPT-2 model was far more stable and coherent. After three epochs of training, it reached a BLEU score of 11.95 on the WMT14 validation set—a modest but meaningful result, especially considering that the model was only exposed to a fraction of the available data. Its translations were generally grammatical and showed awareness of basic word order and vocabulary, though semantic accuracy was often shaky.

One example of the fine-tuned model's output:

> **Source:** A black box in your car?
> **Reference:** Une boîte noire dans votre voiture ?
> **Predicted:** Un box blanc dans sa voiture?

Clearly, the model seems to have a general understanding of what is being said, but may vary from the reference or ground truth output in its wording or synonyms, and may sometimes forget to translate an English word.

Another example of the fine-tuned model's output:

> **Source:** Wearing a wingsuit, he flew past over the famous Monserrate Sanctuary at 160km/h. The sanctuary is located at an altitude of over 3000 meters and numerous spectators had gathered there to watch his exploit.
> **Reference:** Equipé d'un wingsuit (une combinaison munie d'ailes), il est passé à 160 km/h au-dessus du célèbre sanctuaire Monserrate, situé à plus de 3 000 mètres d'altitude, où de nombreux badauds s'étaient rassemblés pour observer son exploit.
> **Predicted:** Wearing un avion de wingsuit, il avait près d'échapper sur la Santé Monserrate à 160km/h. La Santé Monserrate est située à une altitude de plus de 3000 mètres et de nombreux joueurs ont été près d'échapper sa exploit.

As demonstrated in the example, more errors emerged with longer or more nuanced inputs. At times, the model missed idiomatic phrasing or varied in syntax. However, it generally handled simple subject-verb-object structures well.

4.2. RNN-Seq2Seq Results

The model exhibited clear initial learning in epoch 1, as both training and validation loss dropped significantly (see Figure 1). Subsequent epochs used decreasing portions of the training set, which helped prevent overfitting but also limited further performance gains. The BLEU score on the WMT14 test improved from 4.73 after the first epoch to 5.61 in epoch 3, as summarized in Table 1, before slightly declining to 5.53 in epoch 4. While the gains were modest, they demonstrated the model's capacity to generalize better with some refinement.

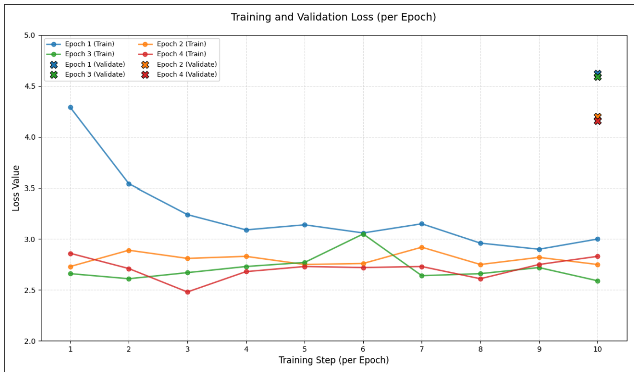

Figure 1. Training and validation loss curves across epochs.

Across all epochs, outputs exhibited a persistent issue: excessive repetition of the word "the". Early predictions included long sequences of repeated tokens, resulting in low BLEU scores and poor semantic alignment with the reference. This issue was analyzed using attention heatmaps, which showed weak or diffuse attention across source tokens during decoding (see Figure 2).

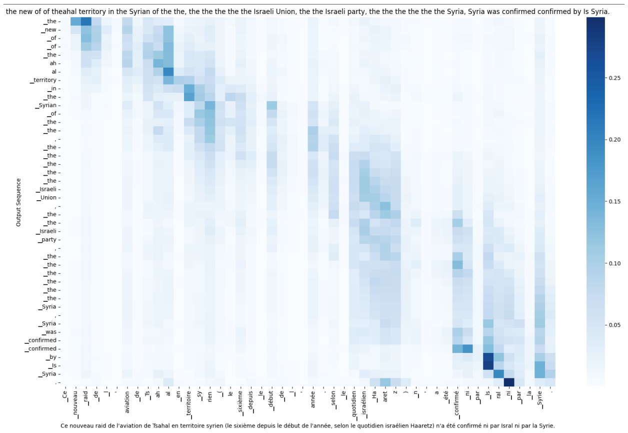

Figure 2. Attention heatmap prior to coverage-based attention modification.

To mitigate repetition, several strategies were explored. First, adding a decoding-time penalty for repeated occurrences of the word the" led to more varied outputs and increased the BLEU score from 4.73 to 4.94. Next, modifying the loss function to penalize incorrect generation of the" during training reduced the BLEU score to 4.28, suggesting diminished learning stability. Finally, adjusting the scale of attention weights—either doubling or halving them—produced more diverse or degenerate outputs, but failed to improve translation quality.

The final modification involved incorporating a coverage-based vector into the attention mechanism, inspired by coverage-aware attention models. This change tracked which source tokens had already been attended to, helping the decoder avoid overusing the same regions of the input. After one epoch of retraining with coverage attention, the model produced more diverse outputs and achieved a BLEU score of 5.00. However, despite improved alignment in attention heatmaps, repetition issues were not entirely eliminated.

A final evaluation using a new input from WMT14 revealed persistent fluency issues and token repetition, although attention alignment had improved. The model's output BLEU score on this example was 4.17. These findings highlight the difficulty of training Seq2Seq models from scratch without pretrained embeddings or extensive data. Nonetheless, the experiments provided valuable insights into decoder behavior, attention dynamics, and training strategy design. For additional details on the Seq2Seq model, including training procedure, visualizations, and further analysis, refer to the extended report available in the model's GitHub repository.

Table 1. Predicted Translation Quality Across Epochs

Epoch	Predicted Translation (Excerpt)	BLEU Score
1	's new of of theahalah territory...	4.73
2	the new of the theahal territory...	5.08
3	new new air of Tsahal territory...	5.61
4	's new regime of the Syrian government...	5.53

4.3. Transformer Results

Even though the Transformer was only trained for a few epochs, it learned quickly and showed strong performance early on. The training loss curve is shown below in Figure 3.

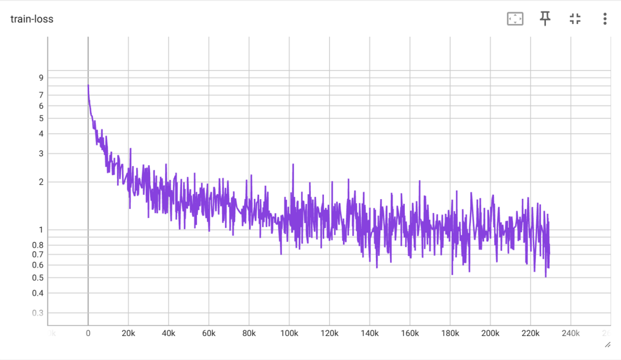

Figure 3. Training loss of the Transformer model

The validation loss, shown in Figure 4, decreased during the first couple of epochs but began increasing in the third, which prompted an early stop to training. This pattern is typical when training a high-capacity model on a relatively small dataset—the model memorizes faster than it generalizes. Nevertheless, the Transformer's translation quality was significantly better than the RNN baseline. Its attention-based design enabled it to capture long-distance dependencies and subtle contextual patterns that RNNs tend to miss. The translations were smoother, more fluent, and generally more faithful to the source meaning.

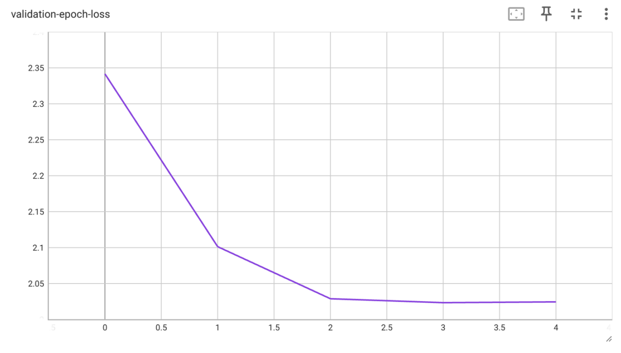

Figure 4. Validation loss of the Transformer model

An example of the model's output is provided below:

Source: The detractors are accusing the government of trying to save money by not allowing injured soldiers—who do not meet the army's rule of "universality of service", which requires that personnel be able to carry out a series of varying tasks—to reach the ten-year period of admissibility required for retirement benefits.

Reference: Les détracteurs accusent le gouvernement de vouloir économiser de l'argent en ne permettant pas aux militaires blessés — qui ne répondent pas à la règle de “l’universalité du service” de l’armée exigeant que le personnel soit en mesure d’effectuer une série de tâches variées — d’atteindre la période d’admissibilité de 10 ans requise pour les prestations de retraite.

Predicted: Les détracteurs accusent le gouvernement d’essayer de gagner de l’argent en ne permettant pas aux soldats blessés — qui ne répondent pas à l’armée de la règle de “universalité de service”, qui exige que le personnel soit capable de effectuer une série de tâches variables — d’atteindre la période de dix ans d’admissibilité requise pour les prestations de retraite.

Clearly, the model is able to maintain structure and meaning even for longer, nuanced sentences. Given that this Transformer was trained from scratch, its performance was impressive. In practice, it handled common sentence structures and vocabulary well, though it occasionally struggled with idioms and less frequent expressions—likely due to the size and diversity of the training corpus. Overall, the model was efficient, interpretable, and surprisingly strong given the scope of the project.

4.4. mBART Results

Training loss decreased steadily over the course of fine-tuning, as shown in Figure 5. The curve shows a sharp initial drop, tapering off after approximately 5,000 steps, indicating stable convergence. The final training loss reached 0.2313 by epoch 3. Validation loss (Figure 6) remained relatively stable in the first two epochs (0.4266 and 0.4253), but rose to 0.4530 in the final epoch, suggesting mild overfitting. Despite this, BLEU scores improved consistently, from 44.62 in epoch 1 to 46.31 by epoch 3, indicating continued gains in translation quality.

Table 2. Training and Evaluation Metrics by Epoch

Epoch	Training Loss	Validation Loss	BLEU Score
1	0.4595	0.4266	44.62
2	0.3111	0.4253	46.06
3	0.2313	0.4530	46.31

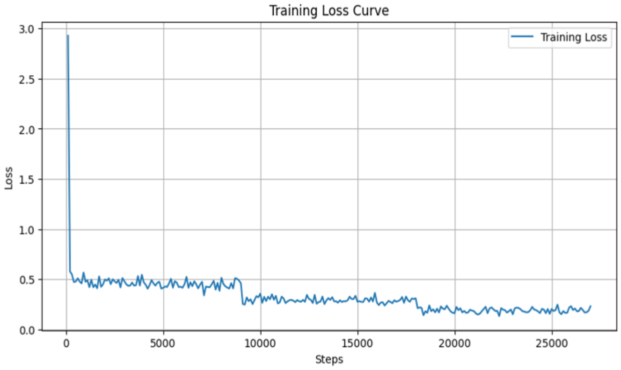

Figure 5. Training loss curve over steps

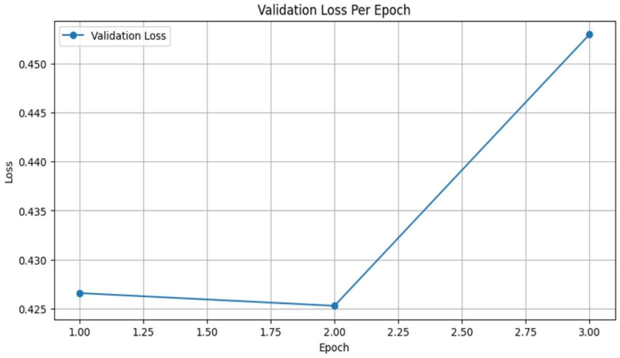

Figure 6. Validation loss per epoch

The model was then evaluated on subsets of the WMT14 English–French test set. On a 100-sample subset, it achieved a BLEU score of 35.11, and on 1,000 samples, a slightly lower score of 34.66. This drop suggests some sensitivity to broader linguistic variation, but the overall scores reflect strong generalization given the size and scope of fine-tuning. All outputs were generated using greedy decoding (`num_beams=1`) with a maximum length of 128 tokens. SacreBLEU was used for corpus-level BLEU scoring.

Sample Predictions from WMT14 Test Set

Example 1:

English: And, most recently, it cannot excuse the failure to design a simple website more than three years since the Affordable Care Act was signed into law.

Predicted French: Et, plus récemment, cela ne peut pas mentérer l’échec de la conception d’un simple site Web depuis plus de trois ans depuis la signature de la loi Affordable Care Act.

Reference: Et, plus récemment, elle ne peut pas excuser l’impossibilité de concevoir un simple site Internet pendant plus de trois années après que la loi sur l'Affordable Care Act a été ratifiée.

Example 2:

English: The bikes were jostling to get in front and take photos. They followed me all the way home, where I had to do some policing to stop some of them climbing up.

Predicted French: Les motos marchaient pour me faire des photos, et ils me suivaient tout le long de la route jusqu'à chez moi, où j'ai dû faire des polices pour empêcher certains d'entre eux d'escalader.

Reference: Les motards se tirent la bourre pour faire des photos et me suivent jusqu'à chez moi où je dois faire un peu la police pour empêcher certains de monter.

Overall, translations were fluent and semantically accurate. Errors typically involved lexical mismatches (e.g., *"motos"* vs. *"motards"*) or overly literal phrasing, but these did not significantly impact meaning. The model performed well across both formal and informal contexts and maintained grammatical structure throughout.

Example Predictions on Unseen Inputs

To further assess generalization, the model was also tested on a small set of unseen English sentences outside the benchmark dataset. Translations were grammatically correct and preserved the intent of each sentence, though some stylistic variety was limited.

Example 1:

English: The weather is nice today.

Predicted French: Le temps est beau aujourd'hui.

Example 2:

English: She studied computer science at university.

Predicted French: Elle a étudié l'informatique à l'université.

Example 3:

English: Artificial intelligence is changing the world.

Predicted French: L'intelligence artificielle est en train de changer le monde.

These results indicate that mBART, when fine-tuned under modest computational settings, can deliver competitive translation performance. Outputs were fluent and informative, and BLEU scores remained stable across multiple evaluation settings. While there is room for stylistic refinement, the model successfully generalized beyond its training distribution and exceeded expectations for a lightweight configuration.

5. Conclusion

This project provided a comparative exploration of neural architectures for English-to-French translation, under realistic compute and data limitations. Four models were investigated: a decoder-only GPT-2 (trained from scratch and fine-tuned), an RNN-based sequence-to-sequence model with attention, a Transformer encoder-decoder, and a pretrained multilingual mBART. Our findings show a clear trend: translation performance improves with the integration of pretrained knowledge and architectural capacity.

The GPT-2 model trained from scratch failed to learn meaningful token mappings and produced degenerate outputs, highlighting the challenges of training high-parameter models on limited data. In contrast, the fine-tuned GPT-2 showed grammatical fluency but suffered from semantic drift and partial translations. The RNN-Seq2Seq model offered better generalization, though it required attention refinements and heuristic interventions to address repetition and coverage issues. Despite modest BLEU scores, it offered interpretability and flexibility in experimentation.

The Transformer model, trained from scratch, demonstrated strong fluency and structural consistency after only a few epochs. It outperformed the RNN baseline and maintained robust performance even on long, complex sentences. However, the highest performance was achieved by the fine-tuned mBART model. Its pretrained multilingual representation and encoder-decoder structure enabled fluent, accurate, and grammatically strong outputs, even when fine-tuned on just 20,000 examples. It achieved BLEU scores over 46 on validation data and above 34 on WMT14 test sets, surpassing all other models.

Overall, this project underscores the importance of pretrained language knowledge, transformer-based attention, and architectural alignment with the task. While simpler models can provide useful baselines and insight, pretrained encoder-decoder architectures like mBART deliver superior translation performance under constrained settings.

Author Contributions and Code Links

We, the authors of this project, affirm that we have contributed equally and dedicated an equal amount of time to this project. All team members participated collaboratively in designing, implementing, and evaluating the models, as well as in writing and editing the report.

Table 3. Equal Contributions from Team Members

Name	Contribution
Jiayi Zhu	Equal contribution
Juanchao Pan	Equal contribution
Maahum Khan	Equal contribution
Tomi Kofo-Alada	Equal contribution
Xingwei Gu	Equal contribution

Code Repositories:

- **GPT-2 (from-scratch and fine-tuned):** https://github.com/billshooting/en-fr-translator/tree/main
- **Transformer:** https://github.com/juanchaopan/transformer
- **RNN Seq2Seq:** https://github.com/codyzhu29/seq2seq_translator
- **mBART:** https://github.com/TomiKofo-Alada/mBART-English-French-Translator

References

[1] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. In *International Conference on Learning Representations (ICLR)*, 2015. 2

[2] Taku Kudo and John Richardson. Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 66–71, 2018. 2

[3] Yinhan Liu and et al. Multilingual denoising pre-training for neural machine translation. In *Transactions of the Association for Computational Linguistics*, 2020. 2

[4] OpenAI Community. GPT2. https://huggingface.co/openai-community/gpt2, 2020. Accessed: 2024-12. 1

[5] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, 2017. 2