Title Generation For Multilingual Session-Based Recommendation Systems

Minh Duc Nguyen
Department of Computer Science Engineering
UNIST
ducnm@unist.ac.kr

Thu Phuong Nguyen
Department of Computer Science Engineering
UNIST
phuongnt@unist.ac.kr

Abstract

E-commerce is becoming more popular thanks to its convenience and its advanced recommendation systems. Despite that, there are few studies on session-based recommendation systems in the multilingual setting. In particular, engaging product title generation can help companies to personalize recommendations and advertisements. In this work, we introduce an overview of the problem and propose a title-generation framework utilizing Natural Language Processing techniques together with benchmarking our model against multiple approaches.

1 Introduction

In recent years, e-commerce has gradually expanded its popularity and slowly taken over traditional shopping as the preferred method of buying goods because of its inherent convenience in this digital era. However, its success is not just due to its convenience, with more and more e-commerce platforms entering the market, they are constantly looking for ways to enhance users' shopping experiences. One of the most vital components when it comes to improving users' experience is the recommendation system as it is the fundamental functionality that makes such platforms convenient. Although there are numerous studies on state-of-the-art recommendation systems, few studies have investigated session-based recommendations, that is, given user previous interaction, recommend the next engaging product, in practical settings with imbalanced and multilingual data scenarios. In particular, predicting or generating product titles is extremely helpful when it comes to personalized recommendation systems or personalized advertisements, which will benefit e-commerce businesses by improving customer experience and increasing sales.

In this paper, we propose a title-generation framework using natural processing techniques for better session-based recommendation systems. The system will receive users' previous interactions in a session and produce the title for the next engaging product. To successfully achieve the goal, we will utilize Amazon's "Multilingual Shopping Session Dataset" [1] consisting of millions of user sessions from six different locales: English, German, Japanese, French, Italian, and Spanish. With the use of natural language processing techniques and this comprehensive dataset, we aim to develop a highly accurate and efficient model.

The contributions of our work will be as follows:

- An exploratory data analysis on the real world "Multilingual Shopping Session Dataset".
- Deep learning frameworks using Large Language Models (LLM) to generate engaging product titles.
- A comprehensive empirical study on the efficiency of different LLMs.

Preprint. Under review.

2 Related Works

2.1 Literature Survey

The early appearances of deep learning in the research of recommendation systems suggest the use of Multi-Layer Perceptron (MLP) and Recurrent Neural Networks (RNN). He et al. [12] proposed Neural Collaborative Filtering to generate embeddings for each user and item. Hidasi et al. [13] proposed the use of Gated Recurrent Units [7] to capture the sequential properties of the session. Recently, most deep-learning approaches are heavily influenced by Transformer-based architecture [34] thanks to its self-attention innovation. Kang and McAuley [16] stacked this powerful architecture block to auto-regressively predict the next engaged item given N-length sequence.

These techniques could be helpful for title generation since we need information from previously interacted products. However, none of them could handle textual inputs, therefore, it would be hard to process multilingual information from the dataset if we want to directly utilize those. To address the multilingual problem, Long Short Term Memory was introduced [14]. Artetxe and Schwenk [2] proposed LASER, a toolkit that can generate cross-lingual sentence embeddings for over 90 languages by using a Bi-directional LSTM encoder and a traditional LSTM decoder. Mathur et al. [22] suggest using sequence-to-sequence models like Bi-directional RNN for title generation with transfer learning to address the data imbalance problem.

In recent approaches related to title generation for e-commerce platforms, Large Language Models (LLMs) are frequently used because of their state-of-the-art performance on various benchmarks in Natural Language Processing (NLP). Liu et al. [19] utilized GRU to produce embeddings and keywords generated from Transformer to provide recommendations given previous product titles. The main architecture of their keyword generation model is OpenNMT [17], a machine translation model capable of handling multiple languages. The model was trained on one day's worth of data from a real-world e-commerce platform with a similar number of entries to our data. Their proposed solution achieved state-of-the-art accuracy on top-MRR and top-Recall metrics compared to other baseline models. Bai et al. [4] was the first to use a generative approach for identifying product attributes. They concatenate the product title, the Google Product Taxonomy category, and the ground truth attributes into two sentences with special tokens to separate the different parts, then a base BERT [9] model was fine-tuned with a sequence-to-sequence objective to generate product names from the textual content of a product. Their model achieved the highest accuracy among the baseline models.

Although the above previous work has achieved great results, no discovery is found when it comes to efficiently generating engaging product titles, and there are only a few studies on how modern LLMs can contribute to this problem.

2.2 Pre-trained Large Language Models

In the past few years, the boom of deep learning in Natural Language Processing and the birth of the Transformers [34] architecture dictates the trend in how language models are designed. Bigger and deeper models are preferred since they are capable of producing human-like text and achieving state-of-the-art performance on most of the benchmarks. In 2018, Google and OpenAI research groups released the first generation of Large Language Models, namely, BERT [9] and GPT [26]. Following their work, LLMs are improved with deeper architectures and bigger datasets that contain multiple languages. Shortly after BERT's release, the same group introduced its multilingual variation. Brown et al. [6] proposed GPT-3, the most capable LLM at the time, however, it was closed-source and people have to pay for API access. Shliazhko et al. [30] reproduced the GPT-3 architecture using GPT-2 sources and introduced mGPT with multilingual capability. BigScience Workshop [5] introduced BLOOM, an open-source LLM with similar sizes and performance to GPT-3. Touvron et al. [33] released a new LLM namely LLaMA with similar performance to GPT-3 but ten times smaller and faster. However, all of them are only encoder or decoder-based architectures, which might fail to deliver good results on problems that require sequence-to-sequence objectives. To address this issue, Raffel et al. [27] introduced their brand new large-scale encoder-decoder model, T5, which was pre-trained on a massive data set consisting of multiple domain knowledge. Shortly after the release of T5, Xue et al. [35] proposed its multilingual variant, mT5, which was pre-trained on a new Common Crawl-based dataset covering 101 languages.

We will conduct experiments with some of those LLMs to investigate their performance on this particular problem.

3 Approaches

In this section, we will formalize the problem description in terms of an NLP problem, and discuss various approaches that we experimented with to solve the problem.

3.1 Problem Formulation

Definition 1. Let $S = \{p_1, p_2, \ldots, p_n\}$, where p_i is the i^{th} interacted product, be the session information of n products in a chronological order.

Definition 2. Let $X = \{x_1, x_2, \ldots, x_n\}$ be an ordered set of titles in a session of n products. Note that the i^{th} sequence of words, $x_i = \left(w_1^i, w_2^i, \ldots, w_{l_i}^i\right)$ is the p_i's title, $|x_i| = l_i$, and $w_j^i \in \mathbb{V}$ (The set of vocabulary).

Definition 3. Given the title information of a session, i.e., $X = \{x_1, x_2, \ldots, x_n\}$ and the maximum length of the title L, define a function $\mathcal{F} : \mathbb{V}^{n \times L} \rightarrow \mathbb{V}^{1 \times L}$ that predicts the next engaging product title, i.e., $\mathcal{F}(X) = x_{n+1}$ should be the product title at the time $n + 1$ that the user interacts with the system. Our objective is to learn the function $\mathcal{F}$.

Definition 4 (Input title concatenation). In all of our approaches, before X was fed into the model, we had concatenated the input title sequence as follows

$$\text{[CLS]}\ x_1\ \text{[SEP]}\ x_2\ \text{[SEP]}\ \ldots\ \text{[SEP]}\ x_n\ \text{[EOS]},$$

and the ground truth will become

$$\text{[CLS]}\ x_{n+1}\text{[EOS]},$$

where [CLS], [SEP], and [EOS] denote the start of the sequence, the separation between input titles, and the end of the sequence respectively.

3.2 The baseline BERT encoder + Transformer decoder

Mane et al. [21] proposed a simple but powerful architecture that utilizes the basic pre-trained BERT to encode the titles and a vanilla Transformer decoder trained from scratch to generate the title for conversational systems. However, it cannot deal with multilingual data since they used the `bert-base-uncased` version of BERT, which was mainly trained on English texts. In our case, we can replace the BERT version with its multilingual cousin, `bert-base-multilingual-cased` [9] to capture a more useful representation of the titles.

As for the decoder, we decided to follow Mane et al. [21]'s architecture, i.e., the vanilla Transformer decoder [34], and train it from scratch. We pick this as the baseline since it is the smallest and easiest to train.

3.3 Fine tuning mT5 model with sequence-to-sequence objective

Our problem formulation resembles closely to the text-to-text paradigm in NLP, hence, it would be beneficial to adopt a large seq2seq model to our problem domain. Raffel et al. [27] provide a powerful encoder-decoder model that is capable of adapting to many downstream tasks, however, it still lacks the ability to understand multiple languages. Xue et al. [35] solved the issue by introducing `T5`'s multilingual cousin, namely, `mT5`. Unfortunately, `mT5` was only pre-trained in an unsupervised manner on the mC4 data set, and it requires to be fine-tuned before applying to any other downstream task [15]. Fortunately, Hasan et al. [11] released the `mT5-multilingual-XLSum`, which was fine-tuned on a massive `XLSUM` data set with 45 different languages specialized for text summarization. We decided to adopt this `mT5` variant, model our problem as a special "summarization" problem, and fine-tune their pre-trained `mT5` on our preprocessed data.

3.4 Fine tuning mGPT model as a casual language model

Similar to the previous approach, we fine-tuned an open-source multilingual pre-trained LLM, namely, `mGPT` [30]. Since `mGPT` is a language model that was not trained for a sequence-to-sequence objective, we decided to fine-tune the model as a casual language model with the following modified input title concatenation format

$$\text{[CLS] } x_1 \text{ [SEP] } x_2 \text{ [SEP] } \ldots \text{ [SEP] } x_n \text{ [EOS] } x_{n+1}\text{[EOS].}$$

During inference, the model would receive "[CLS] x_1 [SEP] x_2 [SEP] $\ldots$ [SEP] x_n [EOS] " as the input (prompt), and it is expected to produce "x_{n+1}[EOS]" as the predicted title.

In the following approaches, we will propose the use of bigger LLMs with billions of parameters that are beyond our computing power to train. Instead, we will leverage the instruction following variants of those big LLMs with prompt engineering.

3.5 Prompt engineering Alpaca and BLOOMZ model

Sanh et al. [28] have developed multitask fine-tuning as a method for enhancing big language model zero-shot task generalization, which may be suitable to solve our problem. In this approach, we will introduce the use of LLMs via zero-shot learning specified for our multilingual titles-to-title objective.

To ensure that the prompts used for Alpaca [31] and BLOOMZ [23] strike the right balance between simplicity and complexity, we have carefully designed our basic prompt. Our approach draws inspiration from established practices in sequence-to-sequence generation work of Bach et al. [3], Scao and Rush [29], as well as widely adopted experiences of Yong and Nikoulina [36] with building a prompt template for low resource languages. We proposed an adaptive prompt generator that generates a prompt suitable for users' language preferences. The generator consists of three main components: (i) Language detector, (ii) natural language task description, and (iii) input sequence (`INP_SEQ`). Each part serves a specific purpose, and detailed explanations of both components are as follows:

- Language detector: To detect the appropriate language for the prompt, we leverage the locale ID of the current session to choose the task description in the corresponding language.
- Natural language task description: It provides an explanation of the task to the model, which has 3 requirements: (a) generating the next engaging product title based on the previous titles to inspire LLMs models' capability of sequence generation, which is a common prompt optimization strategy, (b) generating in the same language as the input, and (c) the length of the output should be roughly the same as the length of the one in the input sequence.
- `INP_SEQ`: This is the titles extracted from all the `prev_items` in the current session in the form of the aforementioned input title concatenation definition.

Then we get a corresponding language template and combine these three elements to build the basic prompt for the large language models, as shown in Figure 1.

4 Experiments

4.1 Data set

We will conduct an experiment on Amazon's "Multilingual Shopping Session Dataset" [1], which contains anonymized customer sessions and product attributes from six different locales: English, German, Japanese, French, Italian, and Spanish. The dataset statistics, including the total number of sessions and the number of products for each locale.

There are about 1.55 million products in the dataset collected in multilingual languages with information about their locale, id, title, price, brand, color, size, model, material, author, and description, as can be seen in Figure 2. Most of the components in product attributes are unstructured, and even the same product ID, which is a unique Amazon Standard Identification Number (ASIN), could have different prices by locale. In addition, all the texture data has a varied range of options to represent

Figure 1: Basic prompt for an input sequence with its respective language.

	id	locale	title	price	brand	color	size	model	material	author	desc
0	B005ZSSN10	DE	RED DRAGON Amberjack 3 - Steel Tip 22 Gramm Wo...	30.95	RED DRAGON	NaN	NaN	RDD0089	NaN	NaN	Amberjacks Steel Dartpfeile sind verfügbar in ...
1	B08PRYN6LD	DE	Simply Keto Lower Carb* Schokodrops ohne Zucke...	17.90	Simply Keto	NaN	750 g (1er Pack)	NaN	NaN	NaN	NATÜRLICHE SÜSSE DURCH ERYTHRIT - Wir stelle...
2	B09MBZJ48V	DE	Sennheiser 508377 PC 5.2 Chat, Stilvolles Mult...	68.89	Sennheiser	Multi-Colour	One size	508377	Kunstleder	NaN	3.5 MM BUCHSE - Kann problemlos an Geräte mit ...
3	B08ZN6F26S	DE	AmyBenton Auto ab 1 2 3 ahre - Baby Aufziehbar...	18.99	Amy & Benton	Animal Car	NaN	2008B	aufziehauto 1 jahr	NaN	【Auto aufziehbar】: Drücken Sie einfach leicht ...
4	B094DGRV7D	DE	PLAYMOBIL - 70522 - Cavaliere mit grauem Pony	7.17	PLAYMOBIL	Nicht Zutreffend.	OneSize	70522	Polypropylen	NaN	Inhalt: 1 Stück

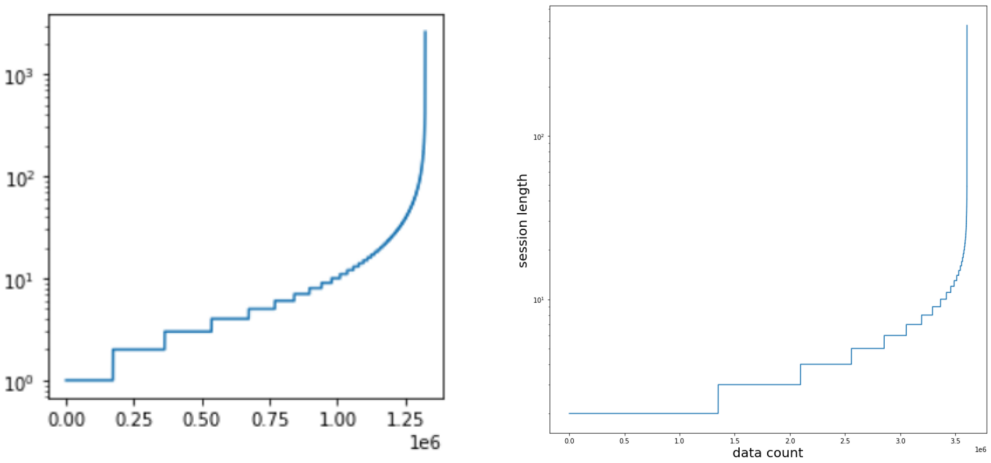
Figure 2: Product attributes in the dataset.

because of the usage of space, bracket, and capitalized words making it hard to tokenize and model texture information, especially for Japanese product titles.

	prev_items	next_item	locale
0	[B09W9FND7K, B09JSPLN1M]	B09M7GY217	DE
1	[B076THCGSG, B007MO8IME, B08MF65MLV, B001B4TKA0]	B001B4THSA	DE
2	[B0B1LGXWDS, B00AZYORS2, B0B1LGXWDS, B00AZYORS...	B0767DTG2Q	DE
3	[B09XMTWDVT, B0B4MZZ8MB, B0B7HZ2GWX, B09XMTWDV...	B0B4R9NN4B	DE
4	[B09Y5CSL3T, B09Y5DPTXN, B09FKD61R8]	B0BGVBKWGZ	DE

Figure 3: User sessions in the dataset.

All user sessions are stored in a list of products a user has engaged with in chronological order, as shown in Figure 3. Each data point contains information such as the IDs of previously selected items by users, the ID of the next item, and the user's location. In this labeled session dataset, the IDs of the next_item as well as their respective locations enable us to search for the names of the next_item in the product dataset, which can then be utilized as gold standards to calculate the loss function. We split the dataset into three parts: a training set, a validation set, and a testing set. The first 76% of the dataset is chosen for training our model, the next 19% of the sample are validation set for turning weight parameters and the final 5% is for testing the performance of those models that we mentioned above.



(a) Frequency of IDs in user sessions data.

(b) User sessions in the data set

Figure 4: Skew data

The fact that the majority of item IDs in the collection have been accessed fewer than 10 times is a challenge (Figure 4a). This suggests that just a small number of items account for a considerable proportion of user interactions. The reason for this is that the search algorithm of the systems tends to show items that are currently trendy or new on the market, resulting in fewer searches for older items. As a result, the training dataset is quite unbalanced, with a long tail distribution in which a few popular goods dominate while many others have minimal representation. This imbalance makes it difficult to train the model properly. Besides, the range of products chosen in each session varies between 2 and 474, which can be seen in Figure 4b. Due to the token limitations of the models (e.g., BERT with 512 tokens, mT5 with 512 tokens, Alpaca with 4096 tokens, and 2048 tokens for BLOOMZ), it is not feasible to include all the product names in the model. To ensure that the prompt does not get cut off halfway, only the most recent N items (in this experiment, we set $N = 5$) are considered for analysis. However, this approach has a drawback of information loss. By focusing exclusively on the latest items and disregarding earlier interactions, valuable insights and patterns from the beginning of the user session are neglected. This loss of information can restrict the model's understanding of user preferences and behavior throughout the entire session, potentially leading to inaccuracies in predictions and recommendations. Nonetheless, this approach helps manage computational complexity and resource requirements while the model also may capture more detailed patterns and generalize better to the specific latest items by utilizing a short of item visits. This method can assist to minimize the long tail distribution's influence and enhance the model's performance.

4.2 Evaluation method

The data set also contains ground truth labels that do not appear in the training set for each entry in the testing data. Thus, we will proceed to use the standard evaluation metrics to evaluate the quality of the title generation, namely, Bilingual Evaluation Understudy (BLEU) [25]. Formally, BLEU could be mathematically represented as

$$\text{BLEU} = \text{BP} \times \exp\left(\sum_{n=1}^{N} w_n \log p_n\right), \tag{1}$$

where BP, N, w_n, and p_n is the brevity penalty, maximum n-gram length, weight assigned to each n-gram, and p_n is the precision score of each n-gram respectively. In particular, we will use $N = 4$, namely, BLEU-4, and a uniform distribution of weights $w_n = \frac{1}{N}$ for evaluation.

4.3 Experimental details

4.3.1 Encoder-Decoder models

In the first approach, we implemented the encoder by importing `bert-base-multilingual-cased` from the huggingface model hub, and the decoder was implemented by `TransformerDecoder` module in PyTorch. We conducted a grid search and obtain the following hyperparameter: `learning_rate`= 10^{-3}, `batch_size`= 32, the decoder hyperparameters were kept the same as the big Transformer in the original paper [34], and the encoder's parameters were frozen. The model was trained for 10 epochs with a 512 token context window, taking 3 hours to finish.

In the second approach, we import the `csebuetnlp/mT5_multilingual_XLSum` checkpoint from huggingface and fine-tuned it for 6 epochs with `learning_rate`= 10^{-4}, `batch_size`= 4, `max_length`= 512 tokens, and `gradient_accumulation_steps`= 16, taking 18 hours to finish.

Both two models were optimized by the AdamW optimizer [20].

4.3.2 Fine tuning a Large Language Model

In the third approach, we obtained the `mGPT` checkpoint from Shliazhko et al. [30]'s official huggingface repository. Following the official notebook, we froze almost all layers but the first and the last ones. The model was trained for 5 epochs with the following set of hyperparameters after a grid search: 1024 token context window, `learning_rate`= 10^{-5}, `batch_size`= 2, and `gradient_accumulation_steps`= 16, taking one day to terminate.

4.3.3 Inference with Large Language Models

In our last approach, we attempted to download the two LLMs on their respective huggingface repository and tried to test them with the proposed solution, but they were too big to fit into our computation units. Fortunately, thanks to the recent advances in quantization techniques ([24], [8], and [10]), Large Language Models can now be compressed and require much less memory and computation. Inference can even be conducted on a single CPU with comparable performance to those on GPU. We obtained the optimized and quantized implementations of the Alpaca [18] and Bloomz [32] model in C++. Experiments were conducted on those variants of the two LLMs, and each took one and a half hours to finish the inference.

4.4 Results

The comprehensive experimental results can be found in Table 1.

Model	BLEU score
Baseline	4.01
Fine-tuned `mT5_multilingual_XLSum`	**10.17**
Fine-tuned `mGPT`	8.21
Prompted `Alpaca-7b`	5.63
Prompted `BloomZ-7b1`	7.91

Table 1: Experimental results on different approaches

As expected, the baseline's performance is the worst among all of the proposed methods. Surprisingly, the fine-tuned `mT5_multilingual_XLSum` is better than the fine-tuned `mGPT` despite having fewer parameters. This behavior might be due to the nature of the problem that it is more suitable for encoder-decoder architectures rather than a casual language model. Or it could be that `mT5` was trained with more data points from more languages (101 compared to 61), which could favor cross-lingual reasoning ability. Unfortunately, our prompt generator cannot beat the fine-tuned model since this well-defined problem is where fine-tuning thrives. However, prompting still has the advantage of speed, efficiency, and interoperability compared to the fine-tuned model. Moreover, `Alpaca-7b` performs more poorly than `BloomZ-7b1` as `Alpaca` is based on `LLaMA`, which is mainly trained on English and acrylic alphabets.

5 Analysis

To start with, we analyze the response behavior of the fine-tuned mT5. Next, we discuss the influence of different prompting combinations with the prompt generator to address the sensitivity of LLMs towards the prompt.

5.1 Response behavior of the fine-tuned mT5

In this section, we discuss the mT5's behavior with challenging input. Firstly, let's consider short inputs with only two items coming from the same locale. In this case, mT5 can catch the context and produce coherent titles. Secondly, let's consider challenging inputs that have multiple locales within the same session. This time, mT5 struggles to even find the correct language for the output, and mainly produce the language appearing the most in that session, which is not always the case. Please refer to Tables 3 and 5 for concrete examples.

5.2 The sensitivity of LLMs towards the prompt

In this section, we discuss the LLMs' behavior when the prompt changes. As we all know, Large Language Models are really sensitive to prompt change and also suffer from challenging inputs. For example, in the case of BLOOMZ, denoting where the end of the input is very important as the model might just auto-complete the INP_SEQ and not provide the desired title. Please refer to Table 4 for a concrete example, where the prompt does not have any notation of the end of input other than the mask [EOS], while adding a simple dot to physically signify the end of the input helps the model producing a more coherent title.

6 Conclusion

In this paper, we introduced insights into a massive real-world multilingual online shopping data set. We proposed various Deep Learning frameworks to solve the session-based multilingual title generation such as the fine-tuned mT5 model, and the prompt generator algorithm. We also provided a comprehensive performance comparison among the introduced methods and came to the conclusion that mT5_multilingual_XLSum outperforms all other approaches. Despite good achievements, our work still has some limitations. Firstly, the experiment lacks comparison to other LLMs. Secondly, different sizes of the models should also be experimented with and compared to the existing solutions. Lastly, comparisons between different prompting techniques should also be discussed and compared. In the future, we hope to address those issues along with adding a few-shot approach to the LLM with better computational units. During the project, we learned a lot of skills related to the field of NLP such as research, fine-tuning techniques for different kinds of models, using LLMs to solve a specific downstream task via prompt engineering, and quantization techniques in optimizing LLMs' performance.

7 Contribution of each team member

Minh Duc Nguyen	**Thu Phuong Nguyen**
Literature survey on the topic and Large Language Models	Exploratory Data Analysis, Preprocessing the data, and visualization
Implementation of the fine-tuning codes	Implementation of the baseline model
Implementation of the prompt generator	Propose the prompt generator mechanism
Evaluation of all proposed methods	Writing introduction, prompt generator approach, and the data sections
Writing the remaining part of the paper	Making the slides

Table 2: Individual contribution of each member

References

[1] Amazon. Multilingual Shopping Session Dataset. https://www.aicrowd.com/challenges/amazon-kdd-cup-23-multilingual-recommendation-challenge, 2023.

[2] Mikel Artetxe and Holger Schwenk. Massively multilingual sentence embeddings for zero-shot cross-lingual transfer and beyond. *Transactions of the Association for Computational Linguistics*, 7:597–610, 2018.

[3] Stephen H. Bach, Victor Sanh, Zheng Xin Yong, Albert Webson, Colin Raffel, Nihal V. Nayak, Abheesht Sharma, Taewoon Kim, M Saiful Bari, Thibault Févry, Zaid Alyafeai, Manan Dey, Andrea Santilli, Zhiqing Sun, Srulik Ben-David, Canwen Xu, Gunjan Chhablani, Han Wang, Jason Alan Fries, Maged S. Al-shaibani, Shanya Sharma, Urmish Thakker, Khalid Almubarak, Xiangru Tang, Mike Tian-Jian Jiang, and Alexander M. Rush. Promptsource: An integrated development environment and repository for natural language prompts. *ArXiv*, abs/2202.01279, 2022.

[4] Xiao Bai, Lei Duan, Richard Wing Cheong Tang, Gaurav Batra, and Ritesh Agrawal. Improving text-based similar product recommendation for dynamic product advertising at yahoo. *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, 2022.

[5] BigScience Workshop. BLOOM. https://huggingface.co/bigscience/bloom, 2022.

[6] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, T. J. Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeff Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. *ArXiv*, abs/2005.14165, 2020.

[7] Kyunghyun Cho, Bart van Merrienboer, Çaglar Gülçehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder–decoder for statistical machine translation. In *Conference on Empirical Methods in Natural Language Processing*, 2014.

[8] Tim Dettmers, Mike Lewis, Younes Belkada, and Luke Zettlemoyer. Llm.int8(): 8-bit matrix multiplication for transformers at scale. *ArXiv*, abs/2208.07339, 2022.

[9] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *ArXiv*, abs/1810.04805, 2019.

[10] Gerganov. Tensor library for machine learning. https://github.com/ggerganov/ggml, last accessed: June, 2023.

[11] Tahmid Hasan, Abhik Bhattacharjee, Md. Saiful Islam, Kazi Mubasshir, Yuan-Fang Li, Yong-Bin Kang, M. Sohel Rahman, and Rifat Shahriyar. XL-sum: Large-scale multilingual abstractive summarization for 44 languages. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 4693–4703, Online, August 2021. Association for Computational Linguistics. URL https://aclanthology.org/2021.findings-acl.413.

[12] Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. Neural collaborative filtering. *Proceedings of the 26th International Conference on World Wide Web*, 2017.

[13] Balázs Hidasi, Alexandros Karatzoglou, Linas Baltrunas, and Domonkos Tikk. Session-based recommendations with recurrent neural networks. *CoRR*, abs/1511.06939, 2015.

[14] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Computation*, 9: 1735–1780, 1997.

[15] Huggingface. mT5. https://huggingface.co/docs/transformers/model_doc/mt5, last accessed: June, 2023.

[16] Wang-Cheng Kang and Julian McAuley. Self-attentive sequential recommendation. *2018 IEEE International Conference on Data Mining (ICDM)*, pages 197–206, 2018.

[17] Guillaume Klein, François Hernandez, Vincent Nguyen, and Jean Senellart. The OpenNMT neural machine translation toolkit: 2020 edition. In *Proceedings of the 14th Conference of the Association for Machine Translation in the Americas (Volume 1: Research Track)*, pages 102–109, Virtual, October 2020. Association for Machine Translation in the Americas. URL https://aclanthology.org/2020.amta-research.9.

[18] Kevin Kwok. Alpaca.cpp. https://github.com/antimatter15/alpaca.cpp, last accessed: June, 2023.

[19] Yuanxing Liu, Zhaochun Ren, Weinan Zhang, Wanxiang Che, Ting Liu, and Dawei Yin. Keywords generation improves e-commerce session-based recommendation. *Proceedings of The Web Conference 2020*, 2020.

[20] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2017.

[21] Mansi Ranjit Mane, Shashank Kedia, Aditya Mantha, Stephen Guo, and Kannan Achan. Product title generation for conversational systems using bert. *ArXiv*, abs/2007.11768, 2020.

[22] Prashant Mathur, Nicola Ueffing, and Gregor Leusch. Multi-lingual neural title generation for e-commerce browse pages. *ArXiv*, abs/1804.01041, 2018.

[23] Niklas Muennighoff, Thomas Wang, Lintang Sutawika, Adam Roberts, Stella Biderman, Teven Le Scao, M Saiful Bari, Sheng Shen, Zheng-Xin Yong, Hailey Schoelkopf, et al. Crosslingual generalization through multitask finetuning. *arXiv preprint arXiv:2211.01786*, 2022.

[24] Markus Nagel, Marios Fournarakis, Rana Ali Amjad, Yelysei Bondarenko, Mart van Baalen, and Tijmen Blankevoort. A white paper on neural network quantization. *ArXiv*, abs/2106.08295, 2021.

[25] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Annual Meeting of the Association for Computational Linguistics*, 2002.

[26] Alec Radford and Karthik Narasimhan. Improving language understanding by generative pre-training. 2018.

[27] Colin Raffel, Noam M. Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *ArXiv*, abs/1910.10683, 2019.

[28] Victor Sanh, Albert Webson, Colin Raffel, Stephen H. Bach, Lintang Sutawika, Zaid Alyafeai, Antoine Chaffin, Arnaud Stiegler, Teven Le Scao, Arun Raja, Manan Dey, M Saiful Bari, Canwen Xu, Urmish Thakker, Shanya Sharma, Eliza Szczechla, Taewoon Kim, Gunjan Chhablani, Nihal V. Nayak, Debajyoti Datta, Jonathan Chang, Mike Tian-Jian Jiang, Han Wang, Matteo Manica, Sheng Shen, Zheng Xin Yong, Harshit Pandey, Rachel Bawden, Thomas Wang, Trishala Neeraj, Jos Rozen, Abheesht Sharma, Andrea Santilli, Thibault Févry, Jason Alan Fries, Ryan Teehan, Stella Rose Biderman, Leo Gao, Tali Bers, Thomas Wolf, and Alexander M. Rush. Multitask prompted training enables zero-shot task generalization. *ArXiv*, abs/2110.08207, 2021.

[29] Teven Le Scao and Alexander M. Rush. How many data points is a prompt worth? In *North American Chapter of the Association for Computational Linguistics*, 2021.

[30] Oleh Shliazhko, Alena Fenogenova, Maria Tikhonova, Vladislav Mikhailov, Anastasia Kozlova, and Tatiana Shavrina. mgpt: Few-shot learners go multilingual. *ArXiv*, abs/2204.07580, 2022.

[31] Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. Stanford alpaca: An instruction-following llama model. https://github.com/tatsu-lab/stanford_alpaca, 2023.

[32] Nouamane Tazi. bloomz.cpp. https://github.com/NouamaneTazi/bloomz.cpp, last accessed: June, 2023.

[33] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aur'elien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. Llama: Open and efficient foundation language models. *ArXiv*, abs/2302.13971, 2023.

[34] Ashish Vaswani, Noam M. Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *ArXiv*, abs/1706.03762, 2017.

[35] Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. mt5: A massively multilingual pre-trained text-to-text transformer. In *North American Chapter of the Association for Computational Linguistics*, 2020.

[36] Zheng Xin Yong and Vassilina Nikoulina. Adapting bigscience multilingual model to unseen languages. *ArXiv*, abs/2204.04873, 2022.

Apendix

Input	**Ground Truth** **Output from mT5**
[CLS] 150 Stück Natürliche Runde Holzperlen Set mit Box für DIY Schmuck Herstellung, 5 Größen (6 mm/ 8 mm/ 10 mm/ 12 mm/ 14 mm) [SEP] Vitamin D3 – Together Health – 1000iu Vitamin D3 – from Wild-Grown Lichen – Vegan Friendly – Made in The UK – 30 Vegecaps [SEP] Melitta 202034 Perfect Clean Milchsystem Reiniger | Entfernt einfach und gründlich Milchablagelagerungen | 250 ml [SEP] Kalender 2023 - Taschenkalender A6, 16 x 10 x 1,4 cm, Terminplaner für Zeitmanag auf Englischement, Shwarz [SEP] Bavarian Edge Messerschärfer – Messerschleifer für alle Messer inklusive Sägemesser – nie wieder stumpfe Messer mit dem Messerschärfer - Profi aus Wolframkarbid [EOS]	[CLS] MiniSun Large Modern White Cylinder Ceiling Pendant/Table Lamp Drum Light Shade [EOS] [CLS] Geldbörse mit Reißverschluss Gross Geldbörsen mit Kette und Lanyard,18 Kartenfächer,Geschenke für Damen(Schwarz) [EOS]

Table 3: Example of challenging input for mT5

Input	**Ground Truth** **Output with non-standard prompt** **Output with standard prompt**
[CLS] Medline Protection Plus Incontinence Disposable Bed Pads 60 x 60 cm Pack of 25 [SEP] Medline Protection Plus Incontinence Disposable Bed Pads 60 x 60 cm Pack of 25 [SEP] Intimates Luxury Thermal Crushed Velvet Navy/Dark Blue Ringtop Pair of Curtains (Plain, One Door Panel 46" Wide x 84" Drop) [SEP] Babycare TENS OBI TENS Plus with Booster Button for Extra Surge of Power Dual Channel Maternity TENS Machine for Pain Relief During Labour with Contraction Timer, White [SEP] Dockteck USB C Hub, 7 in 1 USB C Dock with RJ45 Ethernet, 4K 60Hz HDMI, 100W Power Delivery, 2 USB 3.0 Data Ports and MicroSD/SD, for MacBook Air, MacBook Pro, Surface Pro and More Type C Devices [EOS]	[CLS] Halloween Headbands, 4PCS Spider Headbands Hair Band Headdress Halloween Party Hair Accessories Masquerade Festive DecorGirls Women Halloween Party Masquerade Festive Decor Headdress, Black, One Size [EOS] [CLS] Best title suggestion: Best Title Suggestion for MedLine Protection + Incontinence Disposable Bed Pad [EOS] [CLS] M50 Mark II Camera with EF-M 18-55mm Lens (Black) $999.99 [EOS]

Table 4: Example of challenging input for BLOOMZ

Input	**Ground Truth** **Output from mT5**
[CLS] DWTECH 2022 Halloween Mask LED Halloween Costume LED Glow Scary Light Up Masks for Festival Party Carnival Costume Christmas Cosplay Glow in Dark (Red) [SEP] C5 A5 Royal Mail Large Letter PIP Boxes, 220x160x20mm Corrugated Cardboard Shipping Boxes, Brown Small Postal Packaging Mailing Boxes for Business, Posting, Storing or Gift (Pack of 10) [SEP] LOYAL 21G UK to US Plug Adaptor, Adapter UK to USA with 730 days warranty Type G UK to USA Plug Adapter, Plug Adapter UK to USA Travel Adapter [SEP] 50th Birthday Gifts for Women, Aged to Perfection - 14Oz Pink Ceramic Coffee Cup Printed with Gold, Happy Birthday 50th Mug Present for Her Born in 1972, Mum, Sister, Wife, Friend, Auntie, Gift Boxed [SEP] Upgrade Watch Charger 1.6 ft /0.5 m for iWatch Portable Wireless Charging Cable Compatible with Apple Watch Series SE/7/6/5/4/3/2/1 [EOS]	[CLS] 4 Set Lanyard with Card Holder, 4pc Lanyard and 4pc Waterproof Transparent Card Holder for Office, ID Card Keys Tickets and Exhibition (Black) [EOS] [CLS] Set M50 Mark II Digital Camera (Black) with 18-55mm Waterproof Lens, 4K UHD Video Camera for Film Photography [EOS]

Table 5: Normal input