# CULG: Commercial Universal Language Generation

**Haonan Li**[1*] **Yameng Huang**[2*] **Yeyun Gong**[3] **Jian Jiao**[2]
**Ruofei Zhang**[2] **Timothy Baldwin**[1,4] **Nan Duan**[3]
[1]School of Computing and Information Systems, The University of Melbourne
[2]Microsoft [3]Microsoft Research Asia [4]MBZUAI
haonanl5@student.unimelb.edu.au, tb@ldwin.net
{yamhuang,yegong,Jian.Jiao,bzhang,nanduan}@microsoft.com

## Abstract

Pre-trained language models (PLMs) have dramatically improved performance for many natural language processing (NLP) tasks in domains such as finance and healthcare. However, the application of PLMs in the domain of commerce, especially marketing and advertising, remains less studied. In this work, we adapt pre-training methods to the domain of commerce, by proposing **CULG**, a large-scale commercial universal language generation model which is pre-trained on a corpus drawn from 10 markets across 7 languages. We propose 4 commercial generation tasks and a two-stage training strategy for pre-training, and demonstrate that the proposed strategy yields performance improvements on three generation tasks as compared to single-stage pre-training. Extensive experiments show that our model outperforms other models by a large margin on commercial generation tasks.

## 1 Introduction

Pre-trained language models (PLMs) have achieved impressive success in many NLP tasks across natural language understanding (NLU) and natural language generation (NLG) (Devlin et al., 2019; Yang et al., 2019; Liu et al., 2019; Lewis et al., 2020; Brown et al., 2020; Raffel et al., 2020; He et al., 2020). These successes are usually achieved by pre-training models on large corpora in a task-independent way, and fine-tuning them on a specific downstream task. Researchers have also developed PLMs for specific domains or multiple languages by conducting either pre-training from scratch (Huang et al., 2019; Liu et al., 2020; Xue et al., 2021) or a second phase of pre-training on the basis of existing checkpoints (Howard and Ruder, 2018; Lee et al., 2020; Gururangan et al., 2020). However, PLMs in the domain of commerce, especially for marketing and advertising, remain less studied. On the one hand, adapting PLMs to the advertising domain is challenging because existing pre-training methods usually use open-domain corpora containing largely well-structured text such as books (Zhu et al., 2015), news (Liu et al., 2019), stories (Trinh and Le, 2018), or web text (Radford et al., 2019a) to learn text representations. However, the input text for selecting advertisements is primarily web search queries, which are usually not complete, grammatical sentences. On the other hand, there is no publicly-available PLM in the commercial domain.

This paper introduces **C**ommercial **U**niversal **L**anguage **G**eneration model (CULG), which supports multi-lingual, multi-market, and multi-task ad generation. CULG adopts a transformer-based (Vaswani et al., 2017) encoder–decoder generative framework similar to ProphetNet (Qi et al., 2020), which uses an $n$-stream self-attention mechanism and supports future $n$-gram prediction. To adapt to diverse markets, we use the multi-lingual version of ProphetNet — ProphetNet-X (Qi et al., 2021) as our foundation model, and conduct a second phase of pre-training using a self-constructed large-scale commercial corpus.

CULG is trained auto-regressively on four sequence-to-sequence (seq2seq) generation tasks, including: (1) **G**enerate **K**eywords with the **S**ame intent as the query (GKS); (2) **G**enerate **K**eywords that are **R**elevant to a query (GKR); (3) **G**enerate an **A**d **T**itle based on a query (GAT); and (4) **G**enerate an **A**d **D**escription based on a query (GAD). The motivation of these tasks is to infer the user's intention based on the query as well as perform product matching and recommendation. All queries used in this research are real-life search queries that have been submitted to the Bing[1] search engine, and the ground truth targets are created according to either the records of user's click behaviour or labels from hired human annotators. We collected more

[*] Equal contribution

[1] https://www.bing.com

*Proceedings of NAACL-HLT 2022: Industry Track Papers*, pages 112 - 120
July 10-15, 2022 ©2022 Association for Computational Linguistics

than ten million queries from 10 markets in 7 languages, and split them into three classes according to data quality. Given the user query, the gold class is ads that were deemed as relevant to the query by human judges, the silver class is made up of ads clicked on by users, and the bronze class is all ads that been selected by search engine to show to users. Splitting the data into different markets, tasks, and quality classes provides us with flexibility to compare the model's performance under different training setups.

Given that the collected data varies in quality, we split both the pre-training and fine-tuning into two stages, using low-quality data in the first stage and high-quality in the second stage. To demonstrate the effectiveness of this approach, we compare it with alternative combinations of pre-training and fine-tuning. We evaluate CULG on three commercial generation tasks. Experimental results show that splitting pre-training and fine-tuning into two stages not only outperforms the widely-used single-stage pre-train and fine-tune schema, but is also better than other combinations of pre-training and fine-tuning. We further compare CULG with existing pre-trained multi-lingual models (Liu et al., 2020; Qi et al., 2021) and show that it surpasses other models on commercial generation tasks. Finally, we conduct transfer learning experiments on different markets, languages, and tasks by fine-tuning CULG on a market, language, and task that has not been seen during pre-training. The results demonstrate that CLUG also generalizes well to unseen markets, languages, and tasks.

| Code | Language | Code | Country |
|---|---|---|---|
| De | German | Au | Australia |
| En | English | Ca | Canada |
| Es | Spanish | Ch | Switzerland |
| Fr | French | De | Germany |
| It | Italian | Es | Spain |
| Nl | Dutch | Fr | France |
| Sv | Swedish | Gb | United Kingdom |
|  |  | It | Italy |
|  |  | Nl | Netherlands |
|  |  | Se | Sweden |

Table 1: Languages and countries contained in our corpus. Throughout this paper, we refer to languages and country names with their ISO codes.

## 2 Approach

### 2.1 Model Architecture

CULG adopts the architecture of ProphetNet, an encoder–decoder language generation model with $n$-stream self-attention mechanism and future $n$-gram prediction. Instead of optimizing one-step-ahead prediction as with most sequence-to-sequence models, future $n$-gram prediction aims to prevent overfitting on strong local correlations by simultaneously predicting the next $n$ tokens.

The ProphetNet encoder uses stacked transformer layers with multi-head self-attention, and the decoder uses stacked multi-head multi-stream self-attention layers to enable $n$-gram prediction. Given the input sequence $x = (x_1, x_2, ..., x_L)$ and output sequence $y = (y_1, y_2, ..., y_M)$, ProphetNet implements future $n$-gram prediction by replacing the auto-regressive predicting dependency relationship $p(y_t|y_{<t}, x)$ with $p(y_{t:t+n-1}|y_{<t}, x)$. In detail, it first obtains the encoded sequence representation $H_{enc}$ from stacked encoder layers, where $H_{enc} = \textbf{Encoder}(x_1, x_2, ..., x_L)$. Then the decoder predicts $n$ future tokens simultaneously as $p(y_t|y_{<t}, x), ..., p(y_{t+n-1|y_{<t},x}) = \textbf{Decoder}(y_{<t}, H_{enc})$, where $n$ probabilities are generated at each time step and the probability $p(y_{t+i}|y_{<t}, x)$ is generated by the $i$-th predicting stream. The future $n$-gram prediction objective can be formalized as:

$$
\begin{aligned}
\mathcal{L} = &- \sum_{j=0}^{n-1} \alpha_j \cdot \left( \sum_{t=1}^{M-j} \log\, p_\theta(y_{t+j}|y_{<t}, x) \right) \\
= &- \underbrace{\alpha_0 \cdot \left( \sum_{t=1}^{M} \log\, p_\theta(y_t|y_{<t}, x) \right)}_{\text{language modeling loss}} \\
&- \underbrace{\sum_{j=1}^{n-1} \alpha_j \cdot \left( \sum_{t=1}^{M-j} \log\, p_\theta(y_{t+j}|y_{<t}, x) \right)}_{\text{future } n\text{-gram loss}} \quad (1)
\end{aligned}
$$

The details of ProphetNet can be found in Qi et al. (2020).

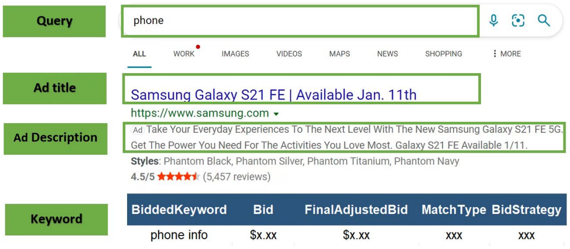
### 2.2 Data Collection

The corpus was collected from 10 markets across 7 languages (Table 1), where a "market" refers to queries issued from a country in a specific language (and is represented as Language–Country in the remainder of the paper), and the corresponding ads and product information. For each market, three types of data were collected:

**Impressed** Given a user query, a collection of ads is chosen from the full ads corpus by the Bing

| Market | GKS | | | GKR | | | GAT/GAD | | | Total |
|---|---|---|---|---|---|---|---|---|---|---|
| | Bronze | Silver | Gold | Bronze | Silver | Gold | Bronze | Silver | Gold | |
| De–Ch | 1,129K | 140K | 2K | 15,288K | 812K | 91K | 3,033K | 332K | 67K | 20,898K |
| De–De | 8,847K | 2,096K | 97K | 135,835K | 14,000K | 413K | 18,625K | 4,122K | 1,711K | 185,751K |
| En–Au | 1,992K | 383K | 75K | 25,768K | 2,078K | 356K | 2,820K | 580K | 1,437K | 35,494K |
| En–Ca | 3,412K | 586K | 58K | 24,324K | 2,117K | 410K | 3,081K | 640K | 619K | 35,251K |
| En–Gb | 8,803K | 1,741K | 137K | 89,385K | 7,819K | 480K | 12,416K | 2,520K | 2,084K | 125,389K |
| Es–Es | 1,387K | 255K | 15K | 73,747K | 3,792K | 103K | 11,858K | 1,084K | 71K | 92,317K |
| Fr–Fr | 5,114K | 1,259K | 105K | 102,538K | 11,000K | 392K | 13,239K | 2,891K | 1,493K | 138,035K |
| It–It | 831K | 148K | 2K | 49,352K | 2,596K | 72K | 8,664K | 879K | 51K | 62,600K |
| Nl–Nl | 1,389K | 301K | 2K | 55,619K | 3,704K | 93K | 9,268K | 1,177K | 77K | 71,633K |
| Sv–Se | 409K | 88K | 2K | 11,732K | 982K | 81K | 2,888K | 431K | 88K | 16,703K |
| Total | 33,318K | 7,002K | 498K | 583,593K | 48,414K | 2,496K | 85,897K | 14,661K | 7,702K | 783,585K |

Table 2: Statistics of source–target pairs in the CULG corpus partitioned by task, quality, and market.



phone

Samsung Galaxy S21 FE | Available Jan. 11th

https://www.samsung.com

Ad Take Your Everyday Experiences To The Next Level With The New Samsung Galaxy S21 FE 5G. Get The Power You Need For The Activities You Love Most. Galaxy S21 FE Available 1/11.

**Styles**: Phantom Black, Phantom Silver, Phantom Titanium, Phantom Navy

4.5/5 ★★★★★ (5,457 reviews)

| BiddedKeyword | Bid | FinalAdjustedBid | MatchType | BidStrategy |
|---|---|---|---|---|
| phone info | $x.xx | $x.xx | xxx | xxx |

Figure 1: An illustration of a user query, ad title, ad description, and bidded keyword.

search engine and shown to the user. This decision process is aimed at maximizing the combined utility of users, advertisers, and publishers by taking the query–ad relevance, bidding, and marketplace policy into account. We collect the pairs of impressed ads and user queries in 2020 based on the system log, and treat them as **bronze** quality data. Figure 1 provides an example user query, ad title, ad description, and bidded keyword.

**Clicked** Among those ads impressed to users, some attract the attention of users and are clicked on for more details. We collect all these clicked ads from the impressed set, and treat them as **silver** quality data.

**Labeled** We developed detailed guidelines to measure the relevance between queries and keywords, queries and ads (including the ad title, ad description, and displayed URL). We hired and trained a team of judges to measure the quality of keywords and ads, sampling data from the "impressed" data above based on our annotation budget. Those instances that are labeled as "Good" are treated as **gold** quality data.

Table 2 presents the statistics of the CULG corpus. From the data quality perspective, we can see the bronze impressed data is much larger than the silver clicked data, which is in turn larger than the gold labeled data for each market and task. From the perspective of different tasks, the task of GKR contains more data than GKS and GAT/GAD (see below for task details).

### 2.3 Tasks

We propose four generation tasks for CULG pretraining. Detailed task descriptions are given below, and examples are provided in Table 3.

**Query to keywords with exactly the same intent (GKS)**: Given a user query, generate a list of keywords that have exactly the same intent as the source query. Such a situation usually occurs when advertisers have a clear targeted audience, judging from the search queries.

**Query to keywords that are relevant (GKR)**: Given a user query, generate a list of keywords that is relevant to the query but don't necessarily have exactly the same intent. This happens when advertisers want to reach to a broader slice of users that may be interested in their product.

**Query to ad title (GAT)**: Given a user query, generate an ad title that is relevant to the query. For many electronic business platforms, there are lots of products without ready-made ad titles and descriptions. This task tends to automatically generate titles that attract users.

**Query to ad description (GAD)**: Similar to GAT, generate an ad description that is relevant to a given query. This task helps sellers reduce their copywriting workload. However, as the real product parameters are neither collected nor embedded in the model, we do not evaluate CULG on this task.

| Task | Source | Target |
|---|---|---|
| GKS | sandstone | sandstones |
| | kempton park races | kempton park racing |
| | debenhams ladies clothing | debenhams ladies fashions |
| GKR | print out boarding pass | boarding pass holder |
| | perth australia city transport | visiting perth australia |
| | wood effect gas fire | gas fire repairer prices |
| GAT | expedia uk | Up to 80% off uk hotels - lowest hotel prices guaranteed |
| | liverpool | liverpool flights - fly to liverpool |
| | just eat | official site - just eat |
| GAD | expedia uk | compare prices on 1000+ sites. the best way to save on your uk hotel! |
| | liverpool | compare prices on liverpool flights with the edreams official website |
| | just eat | even more of your favourite restaurants are now on just eat, and just a tap away |

Table 3: Examples of the four CULG tasks from the En-Gb market.

## 2.4 Two-stage Pre-training and Fine-tuning

The model parameters of CULG are initialized from ProphetNet-X, which is pre-trained on the 100Gb wiki-100 corpus and 500Gb of Common-Crawl[2] data. As a state-of-the-art pre-trained NLG model, its NLU and NLG capabilities (including open-domain multi-lingual generation) are roughly comparable to other encoder–decoder models such as BART (Lewis et al., 2020), GPT-3 (Brown et al., 2020), and T5 (Raffel et al., 2020).

To adapt it to the domain of commerce, we conduct a second phase of pre-training on our commercial corpus. Given that data varies in terms of quality and is large in size, we propose splitting the pre-training into two stages and training on data of increasing quality. The same strategy is applied to model fine-tuning. In detail, the proposed stages are as follows:

**Pre-train stage I** All data including bronze, silver, and gold data from all tasks are used to train the model. As most of the data ($> 90\%$) used in this stage is unlabeled, this stage of training can be considered as unsupervised (in terms of data labeling).

**Pre-train stage II** The gold data from all tasks is used to train the model. This can be considered to be supervised training, given that all of the gold data has been hand-labeled.

**Fine-tune stage I** The generative model is fine-tuned on task-specific bronze, silver, and gold data from multiple markets. This stage helps the model to capture the general features of different languages and markets.

**Fine-tune stage II** The model is fine-tuned on task- and market-specific labeled data to generate high-quality representations, and capture high-level language and market features.

| Method | Pre-train | | Fine-tune | |
|---|---|---|---|---|
| | Stage I | Stage II | Stage I | Stage II |
| 1 | | | | ✓ |
| 2 | | | ✓ | ✓ |
| 3 | ✓ | ✓ | | ✓ |
| 4 | ✓ | ✓ | ✓ | ✓ |

Table 4: Illustration of settings of different methods.

For pre-training, we argue that the unsupervised stage helps the model to learn general text representations, while the supervised stage improves the quality of the learned latent representations using a small amount of high-quality data. For fine-tuning, general-purpose features can be learned from multi-market and -lingual data during stage I, and specific features can be learned during stage II.

## 2.5 Training Methods

To validate the effectiveness of the proposed pre-training and fine-tuning strategies, we create four methods using different combinations of the proposed stages in our experiments (Table 4). **Method-1** involves stage II fine-tuning only without CULG pre-training, which means only a small amount of market-specific labeled data is used to fine-tune the model. This is the most commom mode of fine-tuning after pre-training on publicly available checkpoints. **Method-2** adds stage I fine-tuning before method-1, so that multi-lingual and multi-market data is used to force the model to learn general information across markets first. This is the best that can be achieved on publicly available checkpoints. Note that both method-1 and method-2 use task-specific data. **Method-3** and **method-4** add pre-training stages before method-1 and method-2, respectively.

[2] https://commoncrawl.org/

| Task | Market | Method 1 | | | Method 2 | | | Method 3 | | | Method 4 | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | BLEU-3, 4, AVG | | | BLEU-3, 4, AVG | | | BLEU-3, 4, AVG | | | BLEU-3, 4, AVG | | |
| **GKS** | De–Ch | 6.18 | 0.00 | 12.15 | 21.91 | 9.32 | 32.74 | 21.28 | 9.10 | 31.41 | 24.40 | 10.98 | **34.53** |
| | De–De | 27.38 | 22.22 | 35.58 | 33.73 | 28.98 | 40.98 | 32.12 | 27.90 | 39.59 | 34.94 | 30.21 | **42.08** |
| | En–Au | 34.26 | 25.83 | 42.94 | 40.01 | 32.19 | 47.81 | 38.97 | 30.89 | 46.81 | 41.27 | 33.62 | **48.92** |
| | En–Gb | 32.28 | 24.17 | 40.83 | 37.83 | 30.46 | 45.82 | 36.28 | 28.48 | 44.36 | 38.67 | 31.03 | **46.55** |
| | Es–Es | 31.88 | 24.53 | 39.78 | 50.65 | 45.60 | 55.28 | 46.99 | 43.12 | 51.90 | 52.36 | 47.20 | **56.70** |
| | Fr–Fr | 32.26 | 25.07 | 40.69 | 44.85 | 38.30 | 51.73 | 42.12 | 35.63 | 49.13 | 45.76 | 39.61 | **52.56** |
| | It–It | 13.17 | 7.79 | 19.72 | 34.80 | 19.28 | **43.04** | 31.85 | 20.11 | 40.23 | 34.08 | 18.98 | 41.92 |
| | Nl–Nl | 6.55 | 0.00 | 12.42 | 23.66 | 12.84 | 33.93 | 24.15 | 14.22 | **34.83** | 24.97 | 15.02 | 34.74 |
| | Sv–Se | 6.17 | 0.00 | 11.44 | 22.25 | 11.55 | 32.23 | 21.98 | 10.39 | 32.33 | 22.94 | 12.15 | **32.87** |
| **GKR** | De–Ch | 25.18 | 18.56 | 32.20 | 29.17 | 24.66 | 36.78 | 28.98 | 25.58 | 37.02 | 29.43 | 25.19 | **37.23** |
| | De–De | 20.90 | 16.05 | 27.53 | 25.07 | 20.11 | 32.05 | 23.46 | 18.00 | 30.43 | 25.02 | 20.08 | **32.09** |
| | En–Au | 21.32 | 15.13 | 28.74 | 24.24 | 17.85 | 31.87 | 24.08 | 17.21 | 31.58 | 24.96 | 18.44 | **32.56** |
| | En–Gb | 16.99 | 12.45 | 23.95 | 20.38 | 15.98 | 27.55 | 19.51 | 14.39 | 26.66 | 20.84 | 16.09 | **27.97** |
| | Es–Es | 23.17 | 19.28 | 28.89 | 27.02 | 22.11 | 33.45 | 26.07 | 21.18 | 32.42 | 27.51 | 22.83 | **33.87** |
| | Fr–Fr | 20.20 | 14.19 | 26.90 | 23.41 | 17.00 | 30.40 | 22.85 | 16.11 | 29.92 | 24.08 | 17.53 | **31.13** |
| | It–It | 26.38 | 23.82 | 31.15 | 31.36 | 29.00 | 37.04 | 30.39 | 29.19 | 36.14 | 31.84 | 29.54 | **37.62** |
| | Nl–Nl | 9.13 | 2.43 | 20.85 | 12.36 | 4.30 | 24.66 | 12.21 | 4.33 | 24.37 | 13.23 | 4.88 | **25.54** |
| | Sv–Se | 20.59 | 17.34 | 28.85 | 25.14 | 20.99 | 33.48 | 26.34 | 21.96 | 34.00 | 25.76 | 19.53 | **33.55** |
| **GAT** | De–Ch | 6.20 | 4.02 | 9.18 | 8.05 | 5.86 | 11.04 | 7.30 | 5.10 | 10.30 | 8.34 | 6.14 | **11.31** |
| | De–De | 9.05 | 6.50 | 12.02 | 11.92 | 9.48 | 15.06 | 10.92 | 8.41 | 14.10 | 12.62 | 10.16 | **15.75** |
| | En–Au | 6.50 | 4.11 | 10.03 | 9.80 | 7.22 | 13.35 | 8.78 | 6.20 | 12.35 | 10.06 | 7.50 | **13.62** |
| | En–Gb | 5.06 | 3.06 | 8.13 | 7.73 | 5.86 | 10.58 | 6.14 | 4.27 | 9.02 | 8.46 | 6.51 | **11.39** |
| | Es–Es | 9.69 | 6.84 | 13.64 | 13.12 | 10.24 | 16.95 | 11.95 | 8.99 | 15.91 | 13.85 | 10.95 | **17.67** |
| | Fr–Fr | 2.96 | 1.30 | 5.62 | 3.45 | 1.63 | 6.41 | 3.31 | 1.50 | 6.23 | 3.62 | 1.76 | **6.58** |
| | It–It | 24.90 | 21.24 | 28.12 | 26.70 | 23.03 | 30.05 | 25.89 | 22.09 | 29.37 | 26.91 | 23.24 | **30.25** |
| | NL–NL | 5.18 | 3.29 | 8.29 | 8.66 | 6.60 | 11.84 | 7.28 | 5.15 | 10.58 | 9.07 | 6.94 | **12.24** |
| | Sv–Se | 4.28 | 2.40 | 7.64 | 7.27 | 5.36 | 10.47 | 6.39 | 4.48 | 9.62 | 7.76 | 5.72 | **10.98** |

Table 5: Main results on GKS,GKR and GAT tasks. BLEU-3, BLEU-4, and BLEU-AVG are reported where "BLEU-AVG" means the average score of BLEU-1, 2, 3 and 4.

## 3 Experiments and results

**Experimental setup** For each market dataset, we split it into training, validation, and test set in proportions 80%:10%:10%. The training set is used for CULG pre-training and task-specific fine-tuning.

For pre-training, we fetch the pretrained ProphetNet-X as the basis of CULG, which contains 12 layers in the encoder and decoder respectively, with 1024d hidden size and 4096d feed forward size. The future token prediction length is set to 2, and the max sequence length of the input and output is set to 512. We train the model on all data (stage I) for 1 epoch, and on labeled data only (stage II) for 5 epochs. For training, we use the Adam optimizer (Kingma and Ba, 2015) with a learning rate of $10^{-6}$ and $10^{-5}$ and batch size of 1024. We use the sentencepiece tokenizer with the XLM-R (Conneau et al., 2020) 250k vocabulary, which support 100 languages.

CULG is pre-trained on 8×32Gb NVIDIA Tesla V100 GPUs, at a total cost of around 1500 GPU hours.

For fine-tuning, we use a constant learning rate of $10^{-5}$ and dropout rate of 0.1 for all tasks. We save checkpoints every 10000 steps, and choose the checkpoint with the best performance on the validation set.

### 3.1 Main results

Table 5 presents the main results on GKS, GKR, and GAT. Several observations can be made. First, method-2 consistently outperforms method-1, and method-4 consistently outperforms method-3. We suggest there are two reasons for this: (a) multilingual and multi-market data helps the model to learn general task features; and (b) during fine-tuning, method-2 and method-4 use $>$ 20 times the amount of data of method-1 and method-3 respectively, for most markets and tasks. Second, method-3 beats method-1 for all tasks and markets, while method-4 beats method-2 for most tasks and markets (with the exception of the GKS task in market It–It). This demonstrates the effectiveness of the pre-training. Third, method-1 and method-3 can be treated as few-shot setups, as the amount of labeled data is much less than the unlabeled data. We find

| Task | M-1 | M-2 | M-3 | M-4 | mBART |
|---|---|---|---|---|---|
| GKS | 35.58 | 40.98 | 39.59 | **42.08** | 33.97 |
| GKR | 27.53 | 32.05 | 30.43 | **32.09** | 24.29 |
| GAT | 12.02 | 15.06 | 14.10 | **15.75** | 13.00 |

Table 6: Performance comparison between CULG and mBART on the De–De market, based on BLEU-AVG. 'M-$i$" means method-$i$.

that method-3 outperforms method-1 by a large margin, demonstrating that our pre-trained model can greatly boost the performance in few-shot settings. Finally, the overall performance on GAT is worse than on GKS and GKR, which appears to be because ad titles usually contain advertiser-specific information, which is difficult to infer from a user query.

## 3.2 Comparison to mBART

To compare CULG with models that have different architectures and pre-training data, we choose mBART (Liu et al., 2020), a state-of-the-art multilingual encoder–decoder model. mBART is pre-trained on a large-scale monolingual corpus containing many languages, with a denosing objective function. We download checkpoint *mbart.cc25* and fine-tune it on labeled task-specific data.

We compare CULG with mBART on the De–De market (Table 6). We find that even method-1 achieves better results than mBART on GKS and GKR, and comparable results on GAT, which demonstrates the superiority of our model versus mBART. In addition, with ads data pre-training or multi-lingual fine-tuning, each of method-2, method-3 and method-4 exceed mBART by a large margin, verifying the effectiveness of the pre-training and fine-tuning strategies for commercial tasks. For all tasks, method-4 achieves the best performance.

## 3.3 Transferability

Next, we evaluate the transferability of CULG. Specifically, we use data for a new market, new language, and new task to fine-tune a CULG checkpoint (method-3). For comparison, we choose the publicly available ProphetNet-X checkpoint and fine-tune it using the same data (method-1).

**Market Transferability** To test the transferability of CULG model over markets, we exclude the data from En–Ca during pre-training and use it for fine-tuning. Table 7 shows the results on the three different tasks. We observe a consistent and substantial improvement by CULG (method-3) versus method-1, which suggests that our model performs well over new markets (in a language that is covered in CULG pre-training).

| Task | M | B-1 | B-2 | B-3 | B-4 | B-AVG |
|---|---|---|---|---|---|---|
| GKS | M-1 | 57.59 | 39.13 | 28.04 | 21.60 | 36.59 |
|  | M-3 | 60.76 | 43.94 | 33.20 | 26.67 | 41.14 |
| GKR | M-1 | 45.45 | 31.39 | 21.20 | 15.25 | 28.33 |
|  | M-3 | 47.73 | 34.17 | 24.20 | 18.55 | 31.16 |
| GAT | M-1 | 11.14 | 6.61 | 4.81 | 3.84 | 6.60 |
|  | M-3 | 15.74 | 10.12 | 7.75 | 6.43 | 10.01 |

Table 7: Evaluation of market transferability on the En–Ca market. "M" and "B" represent method and BLEU, respectively.

| Method | B-1 | B-2 | B-3 | B-4 | B-AVG |
|---|---|---|---|---|---|
| Method-1 | 14.37 | 8.06 | 4.80 | 2.99 | 7.56 |
| Method-3 | 20.52 | 12.17 | 7.98 | 5.54 | 11.55 |

Table 8: Evaluation of language transferability on the GAT task for the DA–DK market. "B" represents BLEU.

| Method | B-1 | B-2 | B-3 | B-4 | B-AVG |
|---|---|---|---|---|---|
| Method-1 | 47.70 | 42.99 | 31.46 | 11.50 | 33.41 |
| Method-3 | 50.49 | 45.17 | 33.58 | 13.24 | 35.62 |

Table 9: Evaluation of task transferability on the GBK task for the De–De market. "B" represents BLEU.

**Language Transferability** Data in the En–Ca market is potentially similar to that in En–Us, En–Au, and En–Uk market because of sharing the same language (and having many cultural similarities). It is natural to ask whether our model can also be applied to markets with a language that is unseen in pre-training.

In this experiment, we use data from the Da–Dk (Denmark) market to evaluate language transferability. Note that no Danish data is used during CULG pre-training. At the time of writing this paper, we did not have market data for GKS and GKR, so we will focus exclusively on GAT in this experiment. From the results in Table 8, we see that CULG performs much better than ProphetNet-X, suggesting that our model generalizes to new languages that were not included in pre-training.

**Task Transferability** The generation model can potentially be applied to many scenarios and downstream tasks. We propose four different tasks for

CULG training but wider demand might be required as products evolve. To test whether CULG can be generalized to a task it has not been trained on, we propose another task, which is to **G**enerate the **B**idding **K**eywords (GBK) for an advertiser automatically given the ad description. Experimental results (Table 9) show that method-3 leads to solid improvements on this task vs. method-1, even though this task is not included in pre-training. This demonstrates that CULG is able to leverage information from other tasks for a new task, suggesting greater scope for its applicability.

## 4 Related Work

**Pre-training for Text Generation** Pre-training has been widely used in NLP tasks to learn language representations (Devlin et al., 2019; Liu et al., 2019; Lan et al., 2020; Clark et al., 2020; Yang et al., 2019; Radford et al., 2019b). GPT (Radford et al., 2018) takes plain text as pre-training data to predict the next token in a left-to-right fashion. It performs well on story generation and creative writing. BART (Lewis et al., 2020) uses an encoder–decoder structure to regenerate the original text from a corrupted input using an arbitrary noising function. The denoising training strategy and encoder–decoder structure lead to impressive results on generation tasks. MASS (Song et al., 2019) pre-trains a seq2seq model by masking continuous spans and learn to recover them. T5 (Raffel et al., 2020) investigates different pre-training objectives and model architectures, and pre-trains on a large-scale corpus containing 750Gb of text data. ProphetNet (Qi et al., 2020) introduces a novel self-supervised objective named future $n$-gram prediction, that explicitly encourages the model to plan for future tokens and prevent overfitting on strong local correlations. In this paper, we use the model structure of ProphetNet, and the same $n$-gram objective function.

**Multi-lingual Model in NLP** Building multi-lingual models is becoming more common across NLP tasks. Support for multi-lingual text is either implemented by aligning multi-lingual word embeddings in a universal space (Chen and Cardie, 2018; Lample et al., 2018) or by learning cross-lingual models using a different corpus to exploit shared representations across languages. Models such as mBERT (and), mBART (Liu et al., 2020), XLM-R (Conneau et al., 2020), mT5 (Xue et al., 2021), and ProphetNet-X (Qi et al., 2021) are multi-lingual variants of BERT, BART, RoBERTa, T5, and ProphetNet, respectively.

**Domain Adaptive Pre-training** In this paper, we adapt the pre-trained ProphetNet-X to a commercial domain by continuing to pre-train. Similar work has been done by researchers in other domains. BioBERT (Lee et al., 2020) is obtained by performing additional BERT pre-training on a biomedical corpora, leading to improvements on a variety of biomedical text mining tasks. Alsentzer et al. (2019) continues pre-training BioBERT on clinical data, and achieves performance gains on three clinical NLP tasks. ULMFit (Howard and Ruder, 2018) introduced task-specific fine-tuning, with the core idea being to continue pre-training language models on task/domain specific data. Chakrabarty et al. (2019) used the approach of ULMFit and continued training it on a Reddit corpus, achieving state-of-the-art performance on four claim detection datasets in doing so. Most recently, Gururangan et al. (2020) continued training RoBERTa across 4 domains and 8 tasks, and showed that both domain adaptive pre-training and task adaptive pre-training lead to performance gains.

## 5 Conclusion

In this paper, we propose CULG: a large-scale commercial universal language generation model which supports multi-lingual, multi-market, and multi-task ad generation. As part of this, we propose 4 ad generation tasks for CULG pre-training. We then propose a two-stage pre-training and fine-tuning strategy, and demonstrate the effectiveness of the proposed strategy through extensive experiments. We further compare CULG with other multi-lingual generation models, and show the superiority of CULG on commercial generation tasks. Finally, we demonstrate the transferability of CULG in three different settings.

## 6 Ethical Considerations

This work was conducted while the first author was an intern at Microsoft Research Asia. All data was sourced in strict adherence with the commercial terms of service of the Bing search engine, and no session history or personal data was used in this research.

## References

Emily Alsentzer, John Murphy, William Boag, Wei-Hung Weng, Di Jindi, Tristan Naumann, and Matthew McDermott. 2019. Publicly available clinical BERT embeddings. In *Proceedings of the 2nd Clinical Natural Language Processing Workshop*, pages 72–78.

Jacob Devlin and. Multilingual bert readme.

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, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey 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. 2020. Language models are few-shot learners. In *Annual Conference on Neural Information Processing Systems*.

Tuhin Chakrabarty, Christopher Hidey, and Kathy McKeown. 2019. IMHO fine-tuning improves claim detection. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 558–563.

Xilun Chen and Claire Cardie. 2018. Unsupervised multilingual word embeddings. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 261–270.

Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. ELECTRA: pre-training text encoders as discriminators rather than generators. In *8th International Conference on Learning Representations*.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4171–4186.

Suchin Gururangan, Ana Marasovic, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A. Smith. 2020. Don't stop pretraining: Adapt language models to domains and tasks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8342–8360.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2020. Deberta: Decoding-enhanced BERT with disentangled attention. *CoRR*, abs/2006.03654.

Jeremy Howard and Sebastian Ruder. 2018. Universal language model fine-tuning for text classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, pages 328–339.

Kexin Huang, Jaan Altosaar, and Rajesh Ranganath. 2019. ClinicalBERT: Modeling clinical notes and predicting hospital readmission. *CoRR*, abs/1904.05342.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations*.

Guillaume Lample, Alexis Conneau, Marc'Aurelio Ranzato, Ludovic Denoyer, and Hervé Jégou. 2018. Word translation without parallel data. In *6th International Conference on Learning Representations*.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2020. ALBERT: A lite BERT for self-supervised learning of language representations. In *8th International Conference on Learning Representations*.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2020. BioBERT: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880.

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. 2020. Multilingual denoising pre-training for neural machine translation. *Transactions of the Association for Computational Linguistics*, 8:726–742.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692.

Weizhen Qi, Yeyun Gong, Yu Yan, Can Xu, Bolun Yao, Bartuer Zhou, Biao Cheng, Daxin Jiang, Jiusheng Chen, Ruofei Zhang, Houqiang Li, and Nan Duan. 2021. ProphetNet-X: Large-scale pre-training models for English, Chinese, multi-lingual, dialog, and code generation. *CoRR*, abs/2104.08006.

Weizhen Qi, Yu Yan, Yeyun Gong, Dayiheng Liu, Nan Duan, Jiusheng Chen, Ruofei Zhang, and Ming Zhou. 2020. Prophetnet: Predicting future n-gram for sequence-to-sequence pre-training. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, pages 2401–2410.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding with unsupervised learning.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019a. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019b. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21:140:1–140:67.

Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. 2019. MASS: masked sequence to sequence pre-training for language generation. In *Proceedings of the 36th International Conference on Machine Learning, ICML*, volume 97, pages 5926–5936.

Trieu H. Trinh and Quoc V. Le. 2018. A simple method for commonsense reasoning. *CoRR*, abs/1806.02847.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 5998–6008.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2021. mT5: A massively multilingual pre-trained text-to-text transformer. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 483–498.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime G. Carbonell, Ruslan Salakhutdinov, and Quoc V. Le. 2019. XLNet: Generalized autoregressive pretraining for language understanding. In *Advances in Neural Information Processing Systems*, pages 5754–5764.

Yukun Zhu, Ryan Kiros, Richard S. Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. 2015. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *2015 IEEE International Conference on Computer Vision*, pages 19–27.