**A Chinese** **Couplet Completer with Controlled Text Generation**

**Shiqi Xiao**

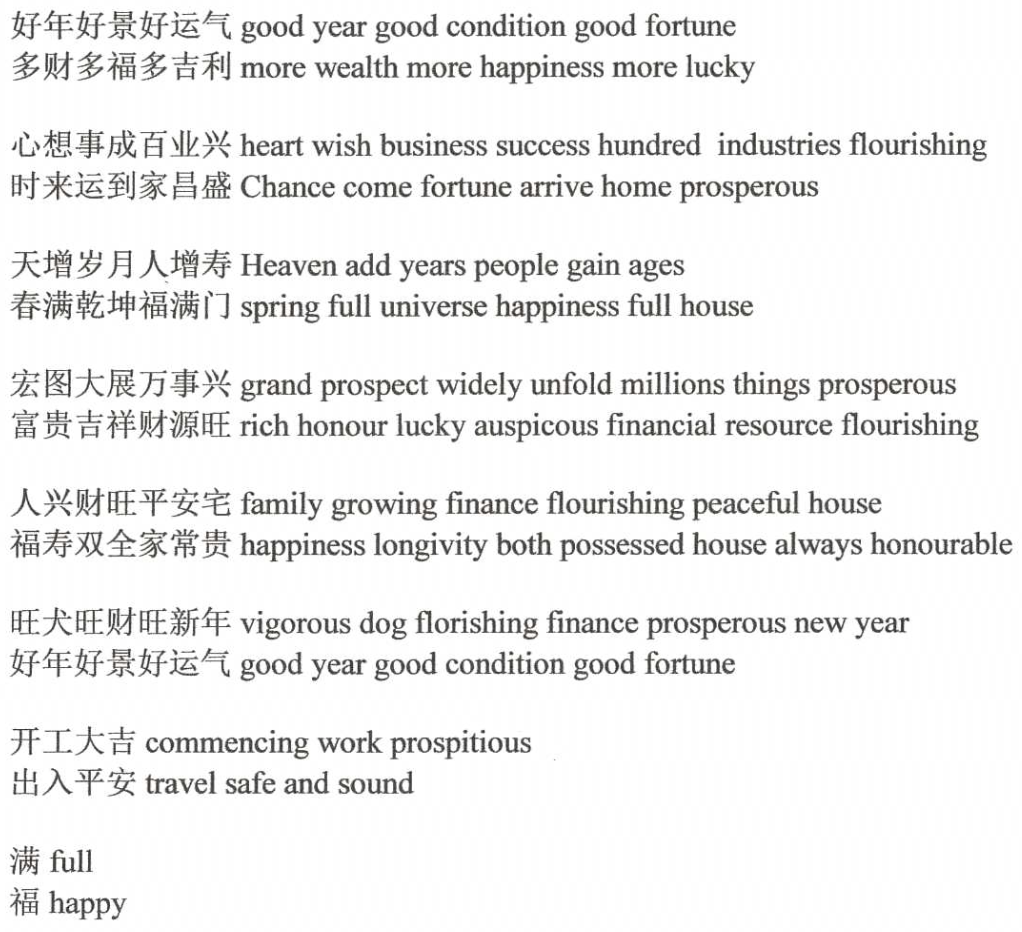
[**s43xiao@uwaterloo.ca**](mailto:s43xiao@uwaterloo.ca)

**Abstract**

This application project proposed a transformer-based generation model that is focusing on completing Chinese couplets. This model has integrated controlled text generation with plug and play language model by Uber’s research team.

**Introduction**

Chinese couplet, also referenced as the antithetical couplet, is a cultural form of expression passed in Chinese society as well as other east and southeast Asian countries, including Vietnam, Korea and Luchu. Chinese couplets can be used on various occasions such as new year celebrations, wedding ceremonies and funerals.



Some examples of Chinese couplets

A proper couplet takes the form of a pair of poetic lines that comply with specific rules. In this project, we only focus on the literal information of couplets.

Our couplet completer takes an incomplete couplet as the input sequence. Then it outputs the sequence representing the complete couplet. An incomplete couplet could mean:

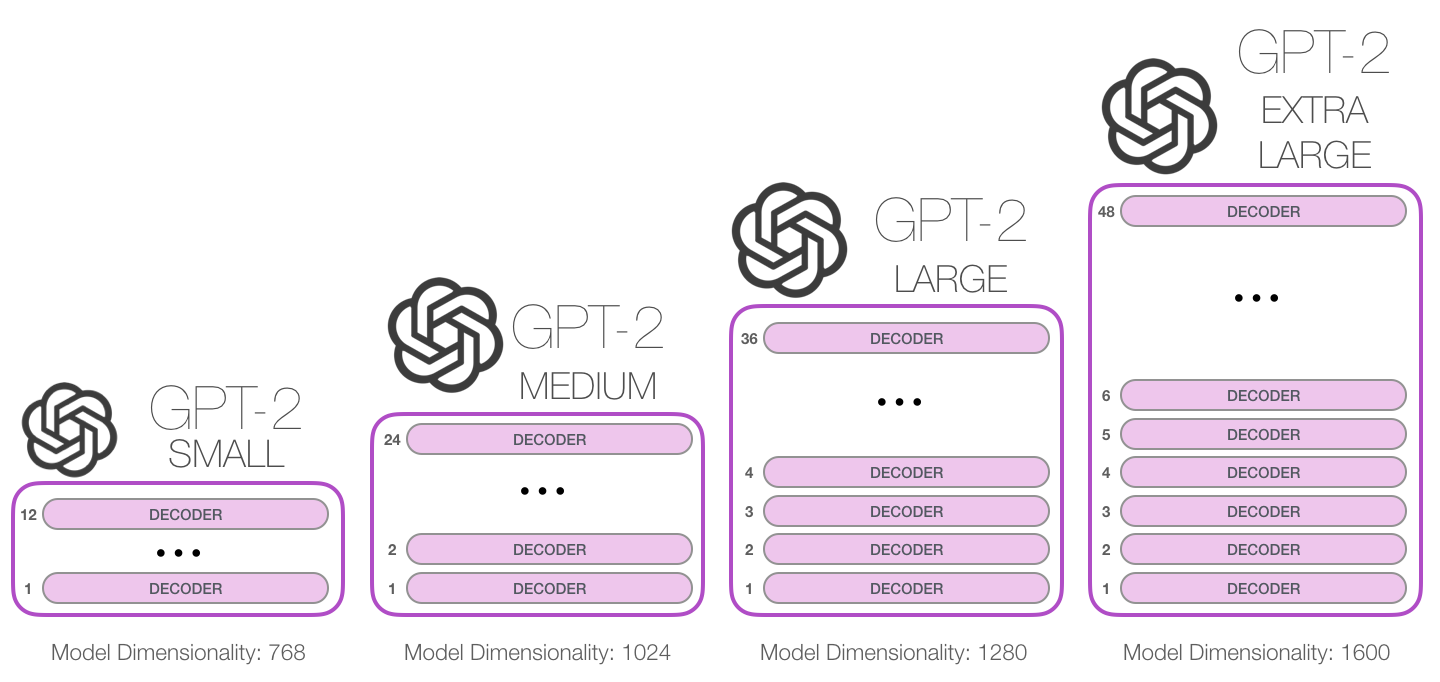
* an incomplete first line
* a completed first line with no second line
* a completed first line with an incomplete second line

If the first line is incomplete, the model will generate a couplet of variable length. Otherwise, the second line will have the same length as the first line.

The generated line can be commanded to talk about a particular topic or have a specified attribute by using plug and play language model method.

**Architecture**

Our couplet completer is a GPT2 [1] pre-trained model with bag-of-words controlled generation (PPLM-BoW) feature. The GPT2 framework is extended from an open-source project called GPT2-Chinese [2]. GPT2 is a transformer-based autoregression language model. It is designed to pre-trained on unsupervised data hence precisely what this task need. This project has adopted some recommendations from an internet blog [6]. No source code is quoted from this blog since the author didn’t publish it.



GPT2 illustration [1]

A pre-trained demo is attached to the source code. The demo takes the shape of GPT2-small; it includes 12 layers of multi-head left-to-right decoders with a hidden size of 768. There is about a total of 92M parameters.

The PPLM method is integrated into the generation procedure, which will be mentioned in the following sections.

**Preprocess and Training**

The corpus we used to train the model is the unsupervised dataset of Chinese couplets organized by Bin Wang [4]. This dataset includes about 770 thousand couplets is the largest couplet dataset we can find online. The context length is set to 72 since all couplets in the dataset are no more than 70 characters long. The tokenization procedure is adjusted. It will put padding tokens at the end of each couplet for consistent sequence length. The step size of stride is set to be as long as the sequence length; therefore, two separate couplets will ensure not to be scoped into the same stride. In each training step, the model is trained to maximize the likelihood of each token conditioned on all previous tokens in the current stride.

A unique separator symbol is inserted after the first line of each couplet. Other unique tokens are concatenated at the beginning and end of each couplet as well. Note that the unique token at the end of each sequence is non-essential since our model is unidirectional. Those symbols can help us control some specific properties such as the beginning of the second line and termination of the generation process without causing the mismatch problem.

[MASK]爆竹声声辞旧岁|锣鼓阵阵迎新春[PAD]…[PAD][CLS]

Example of a couplet after preprocess

There are more than 80 thousand Chinese characters; many of them are uncommon to use. About 10 thousand Chinese characters and several punctuation marks are chosen to construct the vocabulary dictionary.

The samples are tokenized as a one-hot representation, the same as BERT. This model performed tokenization in character-level instead of word-level to accomplish the task because the strict segmental pattern that Chinese couplets required will be disorganized by the word segmenting.

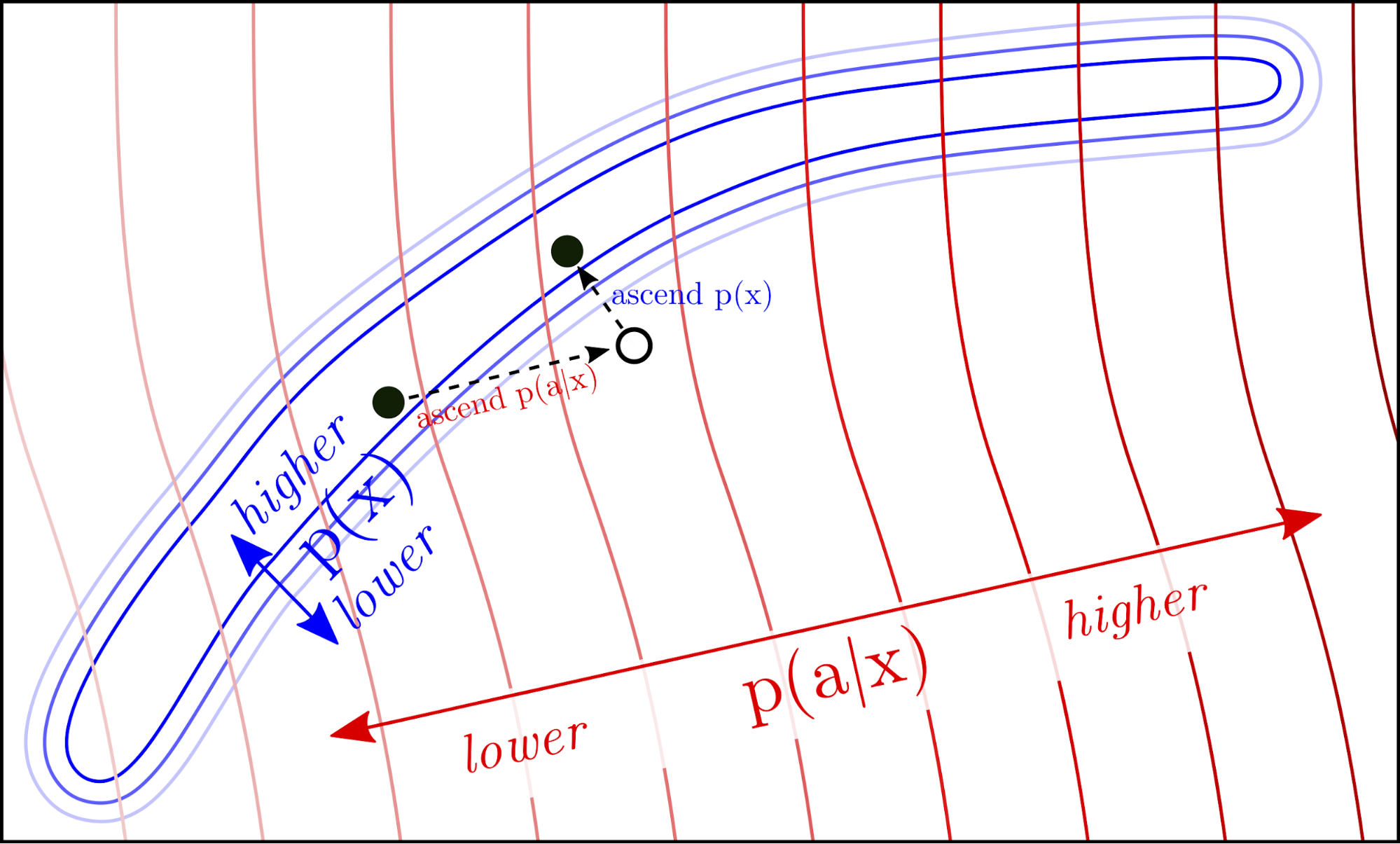
The demo uses a batch size of 16 and trained for approximately 1.2 million steps. The hyperparameters of training have consulted this paper [5].

**Controlled Generating**

The plug and play language model (PPLM) has been integrated into the ordinary autoregressive generation procedure, according to Uber’s paper [3] and its source code. The plug and play language model takes a pre-trained language model p(x) (in our case, it’s the GPT2 based couplet completer) and an attribute model p(a|x). The attribute model can help the language model emphasized the topic of the generated line or even force it to change to another topic.

In each generating step, the model first ran a forward pass on the language model and retrieved the unperturbed distribution. Then repeat the following procedure 3 to 10 times:

* The distribution vector is ascending according to the gradient of the attribute model, increasing the likelihood of the desired attribute.
* The distribution vector is ascending according to the gradient of the original language model, preserving the fluency of generated texts.



PPLM ascending steps illustration [3]

In our case, ascending along the gradient of the language model also preserved rhyming since we trained the language model with couplets.

The attribute model can be any model as long as it backpropagates the gradient. We choose the bag-of-words attribute model (PPLM-BOW), which is the simplest attribute model. A bag-of-words is a set of tokenizable words. In our case, it’s a set of Chinese characters. The likelihood of the model is the sum of the likelihoods of each character in the set.

Several bags of characters that are focusing on each specific topic are sort out for demonstrating. Each bag of words contains approximately a hundred characters. Those characters are chosen from the couplets and poems that strongly related to the topic.

**Examples and Analysis**

Let us evaluate some examples. Giving the first line: 不见新年馨似酒. This line has mentioned the new year as well as alcohol. The pre-trained language model generated the following outputs with top-k sampling (each line is an independent output sequence, duplicate or proximate lines are screened out):

只缘今日醉如泥

又逢春雨贵如油

喜迎元旦客如流

每逢佳节倍思亲

难言往事乱如麻

又闻故友话如诗

但求每日暖如春

又观大地绿如茵

Even taking account of the spring as it’s directly correlated to the new year, we discover that generated lines are only partially related to the new year and the ambience of spring. Some lines are talking about drinking and partying referencing to the alcohols attribute in the given line. Plug and play language model can emphasize the new year characteristic by considering the characters in bow\_newyear.txt. It produces the following lines:

恭迎佳节喜添财

更添福气喜盈门

喜迎元旦乐如春

正临春日暖如家

欣迎元旦乐如春

又迎春节喜迎祥

又迎来岁福盈门

更添春色好迎新

The lines above are obviously more relevant to the desired topic than before. We can even force the second phrase to embrace completely unrelated topics such as heroic using bow\_heroic.txt. By adjusting the step-size, we can choose how much influence the plug and play language model gives to the unperturbed model.

正逢华诞颂忠魂

敢将热血铸成仁

难平往恨悔浮生

将生胆气壮中华

正逢盛世气冲霄

难寻青春悔还家

正当壮志壮国威

难将往日孝铭心

Although lines that depart from the original topics are generally not considered as good pairs, we can still see the above perturbed lines are completely bent to heroic speeches instead of the new year and alcohol attributes of the given line. These examples have demonstrated the capability of restriction by our controlled generation model.

The next examples demonstrate that the generated lines can be bent to opposite emotions. Giving the first line: 夜雨敲窗花落泪, we provide two sets of characters about wedding ceremonies and funerals to the plug and play language model separately. The model that considered bow\_romantic.txt generated the following lines:

秋霜满苑月牵情

春宵伴月意含情

春风入帐烛生辉

春宵一刻月牵情

长宵对月客生情

烛红牵线鸟投缘

晨箫伴枕梦含情

秋心对月夜牵情

On the other side, the model that considered bow\_mourn.txt generated the following sentimental lines:

清明逝水雾寒心

清灯慰寂泪凝心

清风叩案叶归根

长嗟残梦泪沾巾

清风拂柳燕啼悲

春心遗恨怨殇啼

寒风破户烛伤心

清心挽幛泪沾襟

We can see two rounds of outputs have completely diverse emotional attributes. This is because of the characters provided in the two sets are cohesively surround at each’s topic, and each topic has implied those emotional attributes.

**Problems and Further Developments**

Theoretically, we can consider multiple bags of characters in a single generation process, or simply increase the scale of each bag of characters. But in practice, the generated lines lose its readability when the plug and play language model has excessive considered characters. It is possible due to the bag-of-words attribute model gives the same weight regardless of every character in the bag. And the coherence of the considered characters decreases along with its quantity increases. The sequence length of common couplets is also not capable of holding too much information.

The completer could be tuned to generate the first line according to the second line of the couplet if we train the model with reversed sequences.

The attribute model p(a|x) could take many other forms. For example, we could train a simple discriminate model for topics using labelled corpus as our attribute model. The rare characters in some couplets may cause potential mismatch problems since the attribute models trained in this way are unlikely to learn that information.

Similarly, this Chinese couplet task may not be benefiting from an enormous amount of Chinese corpus for pre-training as most recent language models do. The corpus people created are growing exponentially over time, which means most corpus we can find is modernized. But the couplets are mostly constrained by rhymes and usually don’t obey day-to-day grammar.

**Reference**

1. GPT2. Radford et al., 2019 <https://openai.com/blog/better-language-models/>
2. GPT2-Chinese. Morizeyao (Zeyao Du). <https://github.com/Morizeyao/GPT2-Chinese>
3. Plug and Play Language Models: A Simple Approach to Controlled Text Generation. <https://arxiv.org/abs/1912.02164>
4. Couplet-dataset. wb14123 (Bin Wang). <https://github.com/wb14123/couplet-dataset>
5. Training Tips for the Transformer Model. Martin Popel and Ondřej Bojar. 2018. <https://arxiv.org/abs/1804.00247>
6. Pan Yang. 2020. <https://zhuanlan.zhihu.com/p/103722033>