**Proposal –**

**Hierarchical-XLNet: Exploiting compositionality for permutation language modeling**

Xuan Chen

Student # 82286956

Tianyu Hua

Student #

# Introduction and Motivation

Because of bi-directional modelling capabilities, denoising autoencoding pre-training models (e.g. BERT) often give better results than autoregressive-based language models. However, BERT models also have drawbacks, such as ignoring the dependency between masked positions and the difference between pre-training and fine-tuning. Based on the paper "XLNet: Generalized Autoregressive Pretraining for Language Understanding", XLNet learns bidirectional text dependencies by maximising the expected likelihood of all possible permutations of the decomposition order, overcomes the shortcomings of BERT. In the process, we intend to introduce a hierarchical structure to the input sequence and train the model to predict the probability distribution over all possible permutations of the hierarchical structure, which we call “Hierarchical-XLNet (HXLNet)”. Incorporating a hierarchical structure into the current permutation language modeling approach can improve the model's ability to capture the structure of the input sequence and improve its performance on a range of NLP tasks. In this proposal, we will introduce the language model based on permutation, the research design of existing permutation language model, our expected contribution with HXLNet, description of the database we will use, and the evaluation metrics.

# Related Literature

Pre-training + fine-tuning NLP processing solutions are applied increasingly, where the pre-training phase can be divided into two types depending on the pre-training goal: autoregressive (AR) and autoencoding (AE) models. The AR uses an autoregressive model to estimate the probability distribution of a text corpus. This includes the conditional probability of a sequence of text from left to right and the conditional probability of a sequence from right to left. AR models encode text directionally, either left-to-right or right-to-left, which is not efficient to model deep bidirectional contexts, because language understanding tasks often require bidirectional contextual information.

In the paper "XLNet: Generalized Autoregressive Pretraining for Language Understanding," the authors introduce a novel pretraining method called permutation language modeling. Permutation language modeling is a type of autoregressive language modeling that overcomes the limitations of traditional autoregressive models, such as the left-to-right or right-to-left generation order.

BERT is a bidirectional transformer model that pretrains on masked language modeling and next sentence prediction tasks, while GPT-2 is a left-to-right autoregressive model that pretrains on language modeling tasks. However, both models suffer from the bias problem mentioned above. XLNet builds upon previous state-of-the-art language models such as BERT and GPT-2. To address this limitation, XLNet uses a permutation-based training approach that allows the model to capture dependencies between all positions in the input sequence without being biased towards a specific direction. In permutation language modeling, the goal is to predict the probability distribution over all possible permutations of the input sequence. Specifically, the model is trained to maximize the expected log-likelihood over all possible permutations of the input sequence. This allows the model to learn from all possible orders of the input sequence, which enables it to capture dependencies between all positions in the sequence.

The permutation-based objective is introduced based on the autoregressive language modeling objective, but instead of predicting the next word in a sequence based on the previous words, the model predicts the probability of a word at a specific position in the sequence given all other words in the sequence, regardless of their order.

# XLNet V.S. HXLNet

3.1 Contribution of XLNet

The XLNet proposed in this paper is a generic autoregressive method that takes full advantage of the benefits of AR and AE, meanwhile, avoiding their limitations:

1. Instead of using a fixed order of forward or reverse as input as in traditional AR models, XLNet maximizes expectation of all permutations from input sequence. Due to the use of sorted combinations, the context consists of tokens from the left and the right (Figure 1). In the expectation, each position learns to make use of contextual information from other positions, for example, capturing bidirectional contextual information.
2. As a generic AR language model, XLNet no longer uses data corruption (specific identifier numbers [MASK]). Therefore, there is no inconsistency in pre-training and fine-tuning as in BERT. Also, autoregression naturally uses the multiplicative law when decomposing the joint probability of predicted tokens, which eliminates the independence assumption in BERT.

In addition to proposing a new pre-training goal, XLNet also improves the design of the pre-training framework:

1. Inspired by AR language modelling, XLNet applies segment recurrence in Transformer-XL and the relative encoding scheme to pretraining, whose performance is proved to be particularly significant on long text sequences.
2. It is not realistic to use Transformer-XL framework alone in permutation-based language modelling, because the order is arbitrary after decomposition, and the target is unclear as well. As a result, Transformer(-XL) network is reparametrized to reduce the uncertainty.

Diagram

Description automatically generated

Figure 1. Permutation Language Modeling

3.1.1 Compare with other models

1. Comparison with BERT

BERT and XLNet both perform prediction partially, i.e., predicting only a partial subset of the sequence, which is also intended to reduce the difficulty of optimization. However, it is impossible to model among models because of the independence assumption.

Given an example, [New, York, is, a, city]. Assume that both BERT and XLNet choose 2 tokens [New, York] as predicting tokens, and maximize . And the decomposition sequence of sampling is [is, a, city, New, York], the targets of BERT and XLNet:

We notice that, contrary to BERT, XLNet is able to capture the dependency between (New, York). BERT does capture partial dependencies, like (New, city) and (York, city), but given a same target, XLNet learns more dependencies, which means it contains more training information.

2. Comparison with Language Modeling

A standard AR language model like GPT is only able to cover the dependency (x = York, u = {New}) but not (x = New, u = {York}) but XLNet is able to cover both dependencies in expectation over all factorization orders. Standard language models usually import representations of all tokens into softmax. Approaches like ELMo just connect forward and backward language models easily. Since it is only shallow feature stitching, it lacks deep interaction modelling in both directions.

3.2 Introduction to Hierarchical Language Modeling

In natural language, hierarchies refer to the organization of concepts or ideas into a structure of levels or tiers based on their relative importance or relationships. One common type of hierarchy is a tree structure, where a concept at the top serves as the root node and branches out into sub-concepts or sub-topics that are more specific or detailed. For example (Figure 2), in the graph “The song I came to sing remains unsung to this day. I have spent my days in stringing and in unstringing my instrument.”, we regard the root as:

The song I came to sing\n

remains unsung to this day.\n

I have spent my days in stringing\n

and in unstringing my instrument.\n

which then branches out into sub-nodes as two separate sentences.

Diagram

Description automatically generated

Figure 2. An example of hierarchies in natural language

# Expected Contribution

To improve the current permutation language modeling, we consider integrating hierarchical language modeling. We are going to introduce a hierarchical structure to the input sequence and train the model to predict the probability distribution over all possible permutations of the hierarchical structure. Hierarchical language modeling is a type of language modeling that incorporates a hierarchical structure into the input sequence. The idea behind hierarchical language modeling is to break down the input sequence into smaller, more manageable parts (Figure 3), which can be modeled independently and then combined to form a complete representation of the sequence.

Diagram

Description automatically generated

Figure 3. Hierarchical permutation language modeling

In this project, we would propose a permuting approach that respects the hierarchical structure of input language sequences. And also, we would demonstrate the superiority of our pretraining method compared with permutation language modeling in finetuning on downstream tasks.

# Database

We are going to use Wikipedia-en, is an English language version of Wikipedia, the world's largest online encyclopedia, with over 6 million articles and 3.7 billion words.

* Pretraining: HXLNet is pretrained on text sequences using masked language modeling, we input text with randomly masked tokens is fed into a Transformer encoder to predict the masked tokens.
* Finetuning: The pretrained HXLNet can be fine-tuned to downstream tasks, both discriminative and generative. During fine-tuning, additional layers can be added to the Transformer with randomized parameters: these parameters and those pretrained Transformer parameters will be updated to fit training data of downstream tasks.

# Evaluation Metrics