探索合成器增強雙向編碼器於大規模科學語言編輯評測
Synthesizer-Enhanced Bidirectional Encoders for Large-scale Scientific Language Editing Evaluation

許哲瑜 **Zhe-Yu Xu**[1] 褚柏均 **Po-Chun Chu**[1] 王昱翔 **Yuh-Shyang Wang**[2]
王界人 **Chieh-Jen Wang**[2] 涂家章 **Jia-Jang Tu**[2] 李龍豪 **Lung-Hao Lee**[1,*]

[1] 國立陽明交通大學智能系統研究所
Institute of Artificial Intelligence Innovation,
National Yang Ming Chiao Tung University, Hsinchu, Taiwan
[2] 財團法人工業技術研究院
Industrial Technology Research Institute
[*] lhlee@nycu.edu.tw

摘要

科學英文寫作除須確保語法正確外，亦須符合學術寫作慣例。對非英語母語研究者而言，語言編輯往往耗費大量時間與人力。本研究提出 SynBERT，一種以 ELECTRA 爲基礎的科學英文語言編輯偵測模型，用以判斷句子是否需要進一步編輯。模型利用 S2ORC 科學論文語料進行領域特化預訓練，並導入結合自注意力、區間動態卷積與隨機合成注意力的混合注意力機制，以建模全域語意、局部上下文及全域結構資訊。AESW 2016 資料集實驗結果顯示，SynBERT 的 F1 分數優於共享任務系統及多種預訓練語言模型基線，且領域特化預訓練與適當的混合注意力配置皆能有效提升模型效能。

Abstract

Scientific English writing requires not only grammatical correctness but also adherence to established academic writing conventions. For non-native English-speaking researchers, language editing often requires substantial time and effort. This study proposes SynBERT, an ELECTRA-based model for scientific language editing detection that determines whether a scientific English sentence requires further editing. SynBERT is domain-adaptively pretrained on the S2ORC scientific corpus and incorporates a hybrid synthesizer attention mechanism that integrates self-attention, span-based dynamic convolution, and random synthesized attention to capture global semantics, local contextual dependencies, and global structural patterns, respectively. Experiments on the AESW 2016 scientific writing dataset demonstrate that SynBERT achieves a higher F1 score than shared-task systems and several pretrained language model baselines. Further analysis shows that both domain-adaptive pretraining and an appropriate configuration of hybrid attention contribute to improved performance.

關鍵字：科學英文、合成器、混合注意力、預訓練語言模型

Keywords: Scientific English, synthesizers, hybrid attention, pretrained language models

1 緒論

英文撰寫的科學論文是跨國學術交流的重要媒介，然而，許多研究者並非以英語爲母語，因此在投稿前往往需要進行額外校訂，以修正語法、標點、名詞單複數及冠詞使用上的錯誤，以及不符合學術寫作慣例的表達 (Lee et al., 2021b)。科學語言編輯偵測的目標，是判斷科學英文句子是否需要進一步編輯，進而支援後續的自動校訂、人工審稿與寫作輔助流程。AESW 2016 共享任務（Shared Task）即以科學論文句子爲對象，要求系統預測句子是否需要語言編輯 (Daudaravicius et al., 2016)。

傳統的語法錯誤偵測方法多仰賴專家制定的規則或人工設計的語言特徵 (Lee et al., 2013; Flickinger et al., 2016; Remse et al., 2016)，這類方法雖具有一定的可解釋性，但其規則覆蓋範圍與跨領域泛化能力仍然有限。神經網路序列標記及預訓練語言模型則能從語料中自動學習上下文表示 (Rei and Yannakoudakis, 2016; Devlin et al., 2019)，然而，以通用語料訓練的模型未必能充分掌握科學論文中的專業術語、句法結構與寫作慣例。SciBERT (Beltagy et al., 2019) 研究亦顯示科學領域語料可形成有別於通用文本的語言表示。此外，

Proceedings of the 38th Conference on Computational Linguistics and Speech Processing (ROCLING 2026), pages 188–197
November 6-7, 2026 ©2026 Association for Computational Linguistics

Transformer 編碼器的標準自注意力雖擅長建模全域依賴關係 (Vaswani et al., 2017)，仍可藉由動態卷積補強局部依賴，或以合成注意力引入不同的結構偏好 (Jiang et al., 2020; Tay et al., 2021)。

針對上述問題，本文提出 SynBERT 語言編輯偵測模型，以 ELECTRA 的生成器—判別器預訓練框架爲基礎 (Clark et al., 2020)，使用 S2ORC 科學論文語料進行領域特化預訓練 (Lo et al., 2020)，並將 Transformer 編碼器中的多頭自注意力（multi-head self-attention）擴充爲合成器混合注意力。此機制結合自注意力、區間動態卷積與隨機合成注意力，使模型能同時建模輸入相關的全域語意、局部上下文依賴，以及輸入無關的全域結構偏好。我們於 AESW 2016 科學英文寫作評估資料集驗證所提方法的效能，並透過錯誤分析探討模型的表現與限制。

本文的主要研究貢獻可歸納爲以下兩點：

1. 提出合成器混合注意力，同時整合自注意力、區間動態卷積與隨機合成注意力，並分析不同注意力頭配置對模型表現的影響。

2. 使用 S2ORC 科學論文語料進行領域特化預訓練，並透過通用語料與科學語料的比較實驗，驗證領域語料對科學語言編輯偵測的效益。

2 相關研究

語法錯誤偵測（Grammatical Error Detection, GED）旨在辨識文本中的句子是否含有需要修改的文法錯誤。早期先依據常見學習者錯誤，手工建立含句法資訊的語言規則，以判斷中文句子是否含有語法錯誤 (Lee et al., 2013)。後續研究將語法規則與 n 元語法（n-gram）統計模型整合，用於語法錯誤偵測 (Lee et al., 2014)。亦有研究將問題建模爲序列標記，比較多種雙向長短期記憶網路（Bidirectional Long Short-Term Memory, BiLSTM）組合架構，同時利用詞彙與上下文資訊辨識學習者寫作中的錯誤 (Rei and Yannakoudakis, 2016)。進一步的研究則從表示學習改善模型效能，將特定錯誤與語法資訊納入詞嵌入，使其表示能反映詞彙在正確及錯誤語境中的差異 (Kaneko et al., 2017)。研究亦發現簡單的人工擾動與眞實學習者錯誤之間存在落差，因而產生較接近眞實分布的合成錯誤，以提升序列標記模型的偵測能力 (Kasewa et al., 2018)。這些工作顯示 GED 的效能不僅取決於分類器，也受到語言表示與錯誤資料品質影響。

後續研究以 TOCFL 學習者語料 (Lee et al., 2016a, 2018) 進行中文句子層級的語法錯誤偵測，提出 CNN-LSTM 模型 (Lee et al., 2017) 以及多通道 CNN-BiLSTM 版本 (Lee et al., 2020)，將 Word2Vec (Mikolov et al., 2013)、GloVe (Pennington et al., 2014) 與 ELMo (Peters et al., 2018) 等多來源詞表示分別輸入卷積神經網路（Convolutional Neural Networks, CNN）以擷取局部 n 元語法特徵，再由 BiLSTM 擷取長距離特徵。後續工作則採用 ELECTRA 架構 (Clark et al., 2020) 利用生成器與判別器的預訓練機制學習語言表示，以改善語法錯誤分類 (Lee et al., 2021a)。研究進一步建立多標籤錯誤基準，同一句子中的缺字、贅字、選詞與語序錯誤可同時標記，並比較十種代表性神經網路，結果顯示，多重錯誤共存時的偵測仍具挑戰 (Lin et al., 2022)。這些研究說明局部特徵、長距離依賴、預訓練表示及細粒度標籤均是 GED 的重要方向，研究方法呈現由人工規則、神經網路到多標籤分類的演進趨勢。

科學英文的語言編輯需求促成 AESW 2016 共享任務。該資料集取自經專業編輯處理的科學論文，提供 Boolean Decision 與 Probabilistic Estimation 兩種設定，前者判斷句子是否需要編輯，屬於二元分類任務，後者估計其需要編輯的機率 (Daudaravicius et al., 2016)，屬於回歸任務。AESW 的「需編輯」目標標籤除語法外，也可能涵蓋拼字、標點、用詞及科學文體適切性，因此它是以 GED 爲核心、但範圍較廣的科學語言編輯評估 (Daudaravicius et al., 2016; Lee et al., 2021b)。AESW 2016 最終共有六個團隊提交系統，方法涵蓋規則、人工特徵與統計分類。UW–Stanford 以精密語法（precision grammar）及人工制定的 mal-rules 辨識高頻語法錯誤 (Flickinger et al., 2016)。HITS 設計 82 項詞彙與語言特徵，再以邏輯迴歸估計句子需編輯的機率 (Remse et al., 2016)；ISWD 將剖析樹表示與樹核分類器結合，以捕捉句法結構品質 (Mamani Sanchez and Franco-Penya, 2016)；Knowlet 則整合外部工具的輸出，並以最大熵模型進行判斷 (Witte and Sateli, 2016)。NTNU–YZU 將句子表示爲預訓練詞嵌入序列，再以 CNN 擷取局部特徵並進行句子分類 (Lee et al., 2016b)。HU 結合字元層級與詞層級的序列到序列（sequence-to-sequence）修正模型，以及句子層級 CNN，最後以集成方式產生判斷 (Schmaltz et al., 2016)。整體而言，參賽結果顯示神經網路模型可減少對人工特徵的依賴，但不同方法分別偏重局部詞序、句法

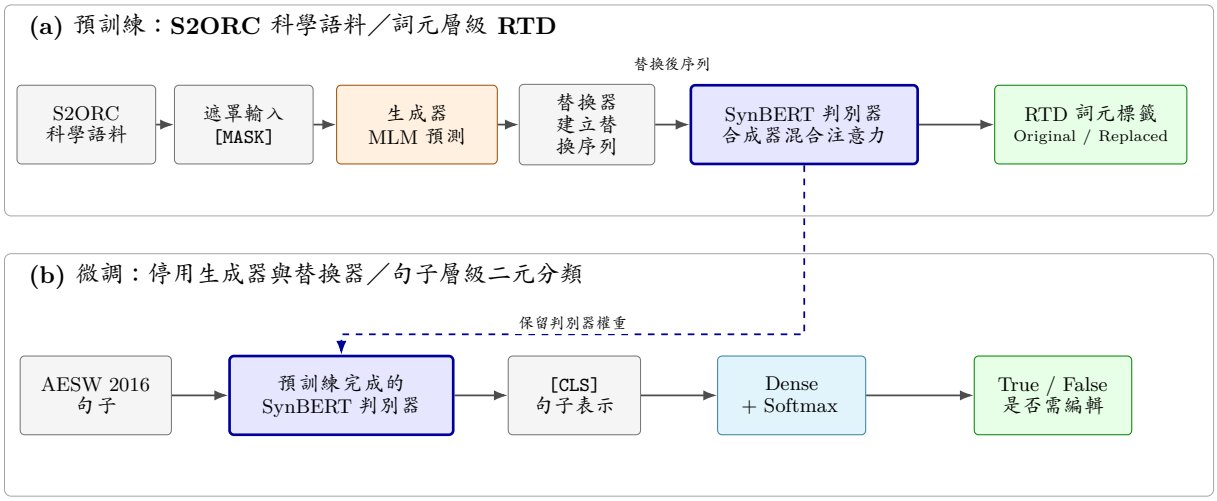

圖 1: SynBERT 的預訓練與微調架構。模型以 ELECTRA 的生成器—判別器框架進行替換詞元偵測預訓練，並於微調階段保留導入合成器混合注意力的判別器，以進行句子層級的二元分類。

規則或修正訊號，尚未形成一致的科學文本表示方式。

預訓練語言模型普及後，GED 研究進一步轉向上下文表示與更細緻的錯誤標籤。EMC-NN 的實驗結果顯示，多來源語言表示、全域序列資訊與局部特徵在科學語言編輯偵測中具有互補性 (Lee et al., 2021b)。Yuan et al. (2021) 採用 ELECTRA (Clark et al., 2020) 架構建立高效的二元 GED 模型，並將輸出擴充爲不同粒度的多類別錯誤標記，以支援後續修正模型。研究發現基於 BERT (Devlin et al., 2019) 的語法錯誤偵測器只需約 5–10% 的標註資料，便可達到非預訓練方法使用完整資料時的表現，顯示遮罩語言模型已內含可轉移的語法知識 (Nagata et al., 2022)。另有研究以主詞動詞一致性錯誤爲探針，比較 BERT (Devlin et al., 2019)、XLNet (Yang et al., 2019)、GPT-2 (Radford et al., 2019)、RoBERTa (Liu et al., 2019) 與 ELECTRA (Clark et al., 2020) 的逐層表示，發現不同模型與訓練資料對句法錯誤的可辨識性仍有明顯差異 (Davis et al., 2022)。

近期 GED 亦逐漸重視多語與低資源情境。MultiGED-2023 將詞元層級 GED 擴展至五種語言，建立跨語言的共同評估平台 (Volodina et al., 2023)；Ngo et al. (2023) 以字元式 LSTM 與多語預訓練模型進行實驗，結果顯示，子詞及字元表示可在多語錯誤偵測中互補；Lopez Latouche et al. (2024) 利用多語預訓練模型進行零樣本跨語遷移，爲缺乏人工標註的目標語言生成合成錯誤資料；Gralinski et al. (2025) 則將 GED 視爲文字異常偵測，提出不需標註資料的非監督式方法。這些研究分別處理標註成本、跨語遷移及非監督學習，但主要評估對象仍是一般或學習者文本。

綜合上述文獻，尚未有方法在 AESW 句子層級任務上，同時系統性結合科學領域預訓練、輸入相關的全域注意力、局部動態卷積與輸入無關的結構注意力。因此，我們提出 SynBERT 模型以科學語料強化領域表示，並以合成器混合注意力整合不同感受範圍的訊息，回應科學語言編輯偵測中的領域差異與局部及全域建模需求。

3 SynBERT 方法

3.1 系統架構

我們採用 ELECTRA (Clark et al., 2020) 的生成器—判別器預訓練架構，並將判別器作爲下游句子分類模型的主要編碼器。模型包含詞嵌入層、生成器、替換器、導入合成器混合注意力的判別器，以及微調階段使用的分類器。圖 1上半部對應預訓練流程。由左至右，科學語料先進入生成器：輸入句子的部分詞元被遮罩，生成器依上下文執行遮罩語言模型（Masked Language Modeling, MLM），預測遮罩位置的候選詞元；替換器再以預測結果建立新的輸入序列。該序列同時保留原始詞元與生成器替換的詞元，接著送入導入合成器混合注意力的判別器。判別器以替換詞元偵測（Replaced Token Detection, RTD）逐一判斷各位置屬於 Original 或 Replaced。圖 1下半部則對應微調流程：將 AESW 2016 句子輸入預訓練完成的判別器以取得語意表示，再經由 Dense 與 softmax 分類器輸出 True 或 False，分別表示句子需要或不需要語言編輯。

3.2 輸入表示與詞嵌入

在圖 1的生成器與判別器中，輸入文字先切分爲詞段（word pieces），並於序列前後加入 `[CLS]` 與 `[SEP]`。令輸入序列爲 $X = [x_1, x_2, \ldots, x_n]$，token embedding 依各詞段的索引取得向量 T；position embedding 則以詞段在句中的絕對位置產生位置向量 P。兩者逐位置相加後得到 W，作爲編碼器第一層的輸入。此表示同時保留詞彙身分與序列位置資訊：

$$T = \mathrm{TokenEmbedding}(X), \qquad W = T + P \tag{1}$$

以輸入句“he is drinking water”爲例，句子會表示爲 `[CLS]`、he、is、drink、`##ing`、water、`[SEP]`。其中 drinking 被切成 drink 與 `##ing` 兩個詞段，各詞段分別查得 token embedding；位置 0 至 6 則各自對應一個 position embedding。兩組向量在相同位置相加後，`[CLS]`、一般詞段及 `[SEP]` 都具有自己的 W_i。這些向量不直接代表最終判斷，而是提供後續多層編碼器逐層整合上下文的起始表示。經詞嵌入取得的 W 會依模型規模通過多層修改後的 Transformer 編碼器。每一層先計算合成器混合注意力（詳見第 3.5 節式 (11)），再將注意力輸出與原輸入進行殘差相加及層正規化；其後通過由兩層 Dense 組成的前饋神經網路，第一層使用 ReLU，最後再次執行殘差相加與層正規化。若以前一層輸出作爲下一層輸入，單層計算可表示爲：

$$Z = \mathrm{LayerNorm}(X + \mathrm{SMA}(X)) \tag{2}$$

$$\begin{aligned} Z' &= W_2 \max(0, W_1 Z + b_1) + b_2, \\ Y &= \mathrm{LayerNorm}(Z + Z') \end{aligned} \tag{3}$$

3.3 生成器、替換器與判別器

預訓練時，生成器接收含有遮罩位置的序列 x_{masked}，利用其編碼器輸出的上下文特徵，估計每個遮罩位置在詞庫中各候選詞段的條件機率。替換器依此機率選取預測詞段 $\hat{x}$，並將其填回遮罩位置 m，形成 $x_{\text{generated}}$。若填入詞段與原始詞段相同，該位置仍標記爲 Original；若兩者不同，則標記爲 Replaced。此步驟將生成器的語言建模結果轉換成可供判別器學習的逐詞元監督訊號：

$$x_{\text{generated}} = \mathrm{REPLACE}(x, m, \hat{x}) \tag{4}$$

$$\hat{x}_i \sim p_G(x_i \mid x_{\text{masked}}), \quad i \in m \tag{5}$$

生成器對遮罩位置的每個候選詞段計算分數，再以 softmax 取得詞庫上的條件機率。分數同時使用候選詞段的嵌入 $e(x_t)$ 與生成器在位置 t 的上下文特徵 $h^G(x)_t$，因此預測結果取決於遮罩前後的語境。替換器只處理遮罩集合 m 中的位置，其他詞元保持不變；這使 $x_{\text{generated}}$ 與原始句子高度相似，但可能含有語境上合理、實際卻非原詞的替換詞元，形成比任意擾動更具辨識難度的訓練樣本。

例如輸入 `[CLS]` he `[MASK]` drink `##ing` water `[SEP]` 時，若生成器預測機率最高的詞段爲 is，替換器便將 is 填入遮罩位置。若原始句子該位置也是 is，標籤爲 Original；若原始詞爲 was，雖然生成後句子在形式上仍可能通順，該位置仍須標記爲 Replaced。因此監督標籤由生成結果與原始詞元是否相同決定，而不是由句子表面是否合乎文法決定。

判別器以 $x_{\text{generated}}$ 爲輸入，經詞嵌入與修改後編碼器取得每個位置 t 的上下文特徵 $h^D(x)_t$，再以線性轉換及 sigmoid 估計該位置是否被替換。相較於只對遮罩位置計算預測，RTD 會對輸入序列的每個詞元給出 Original ／ Replaced 判斷，因此判別器可從整個序列學習詞彙與上下文是否一致。其逐詞元輸出爲：

$$D(x, t) = \mathrm{sigmoid}\Big(w^\top h^D(x)_t\Big) \tag{6}$$

此處的分類器與下游句子分類器用途不同。預訓練分類器對序列中的每個位置 t 分別套用同一組權重 w，sigmoid 輸出該詞元爲替換詞元的分數；生成器的輸出只負責建立訓練序列與標籤，眞正保留到下游任務的是判別器。藉由逐位置比較原始與替換詞元，判別器在進入 AESW 2016 微調前，已先學習科學語料中詞元與上下文的相容關係。

3.4 微調與句子層級預測

完成科學語料預訓練後，生成器與替換器不再用於下游任務；圖 1下半部僅保留判別器，並以 AESW 2016 的句子及其是否需要編輯的二元標籤（True ／ False）進行微調。判別器將整句轉換爲語意表示 x，分類器再以 Dense 層進行線性轉換，並透過 softmax 取得兩類機率。模型選取機率較高的標籤作爲最終預測，其中 True 表示句子需要進一步語言編輯，False 表示不需要編輯：

$$\mathrm{Prediction} = \mathrm{softmax}(\mathrm{Dense}(x)) \tag{7}$$

因此，圖 1的兩個階段具有不同的預測粒度與學習目標：預訓練階段在詞元層級辨識替換位置，使判別器先從科學語料學習術語、句法與上下文一致性；微調階段則將相同的判別器轉

換爲句子層級分類器，直接對應 AESW 2016 的 Boolean Decision 任務。這個流程使領域特化表示與最終的科學語言編輯偵測目標相互銜接。

3.5 合成器混合注意力

合成器混合注意力由三種運算機制組成，分別爲 self-attention、span-based dynamic convolution 與 random synthesized attention。

Self-attention 計算 Query 與 Key 的相似度，可捕捉由輸入內容決定的全域依賴；span-based dynamic convolution 以局部卷積關注相鄰片段，補強局部語序與短距離依賴；random synthesized attention 則透過可訓練的隨機初始化矩陣形成與輸入無關的注意力圖，提供另一種全域結構偏好。

令輸入序列爲 A，經線性投影得到 Q、K 與 V。自注意力先計算 Query 與 Key 的相似度，再以 d_k 的平方根縮放並經 softmax 正規化，最後乘上 V：

$$\mathrm{SA}(Q,K,V)=\mathrm{softmax}\left(\frac{QK^{\top}}{\sqrt{d_k}}\right)V \tag{8}$$

其中 Q、K 與 V 均由同一輸入 A 經不同線性投影取得。$QK^{\top}$ 的每個元素表示一個 Query 與序列中各 Key 的相似程度；除以 $\sqrt{d_k}$ 可避免內積值隨維度增大，softmax 再將各列轉換爲機率分布。該分布乘上 V 後，每個位置便能依輸入內容彙整序列其他位置的資訊，因此自注意力負責提供輸入相關的全域語意。

區間動態卷積先以深度分離卷積產生 K_s，再由 $Q\odot K_s$ 建立輸入相關的局部卷積核。令 α_i 表示位置 i 經 softmax 正規化後的動態卷積核；其中 LConv 表示以位置 i 爲中心的輕量卷積，W_f 爲線性轉換，SDConv_i 爲位置 i 的區間動態卷積輸出：

$$\begin{aligned}\alpha_i &= \mathrm{softmax}(W_f[Q\odot K_s]_i)\,,\\ \mathrm{SDConv}_i &= \mathrm{LConv}_i(V,\alpha_i)\end{aligned} \tag{9}$$

區間動態卷積源自 ConvBERT (Jiang et al., 2020) 對注意力頭的觀察：部分自注意力頭主要集中於鄰近詞元，不必檢視完整序列即可學習局部關係。因此，此分支先以深度分離卷積由 A 產生 K_s，並將 Q 與 K_s 逐元素相乘；W_f 將結果映射爲位置 i 的動態卷積核，經 softmax 正規化後，由 LConv 在局部視窗內對 V 加權。相較於全域自注意力，這個分支明確建模相鄰片段及短距離語序，但單獨使用時取得全域資訊的能力較弱。

隨機合成注意力不使用 Query-Key 相似度，而是學習與輸入序列長度相應、但不由輸入詞元內容決定的矩陣 R：

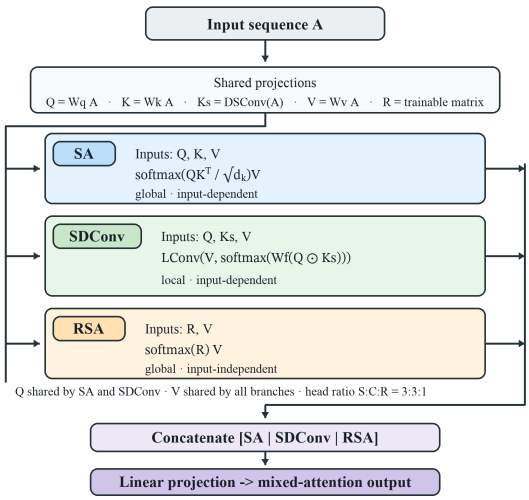

圖 2: 合成器混合注意力架構。SA 與 SDConv 共用 Query，三個分支共用 Value；各分支輸出串接並經線性轉換。在 SynBERT-base 中，3:3:1 分別對應 6、6、2 個注意力頭。

$$\mathrm{RSA}(R,V)=\mathrm{softmax}(R)V \tag{10}$$

隨機合成注意力以隨機初始化且可訓練的矩陣 R 取代 $QK^{\top}$。R 的大小與輸入序列相應，但其數值不由當前詞元內容計算；訓練過程直接學習位置之間的注意力模式。R 經 softmax 後與共用的 V 相乘，使此分支能補充輸入無關的全域結構觀點，並與另外兩個依賴輸入內容的分支形成互補。

三個分支完成運算後，其輸出先進行串接，再經線性轉換形成合成器混合注意力的輸出：

$$\mathrm{SMA}=\mathrm{Concat}(\mathrm{SA},\mathrm{SDConv},\mathrm{RSA}) \tag{11}$$

三個分支並非建立彼此獨立的完整投影。如圖 2 所示，SA 與 SDConv 共用 Query，SA、SDConv 與 RSA 則共用 Value；各注意力頭完成運算後，輸出沿特徵維度串接，再經線性投影回到編碼器所需維度。這個設計在同一層中保留自注意力的全域語意、動態卷積的局部依賴，以及隨機合成注意力的全域結構偏好。

注意力頭比例決定三種資訊在串接結果中的容量。SynBERT-base 的每個注意力頭維度皆爲 64；最終的 3:3:1 配置分別分配 6 個 SA 頭、6 個 SDConv 頭與 2 個 RSA 頭，共 14 個頭，因而形成 896 維隱藏表示。此配置保留較多輸入相關的全域與局部建模容量，同時以較少的 RSA 頭提供輸入無關的結構訊息。

	無修改句子			需語言編輯句子		
領域	訓練集	發展集	測試集	訓練集	發展集	測試集
數學	218,585 (30.2%)	27,784 (30.8%)	28,347 (32.8%)	135,344 (28.5%)	16,971 (29.0%)	16,183 (28.0%)
物理學	169,160 (23.4%)	23,290 (25.8%)	19,203 (22.2%)	122,742 (25.8%)	15,704 (26.9%)	15,962 (27.7%)
工程學	145,013 (20.0%)	17,309 (19.2%)	17,722 (20.5%)	99,505 (20.9%)	11,633 (19.9%)	12,676 (21.9%)
電腦科學	103,368 (14.3%)	12,234 (13.5%)	11,694 (13.5%)	61,104 (12.8%)	7,719 (13.2%)	6,799 (11.7%)
統計學	42,390 (5.8%)	5,283 (5.8%)	4,475 (5.1%)	27,749 (5.8%)	3,321 (5.6%)	2,854 (4.9%)
經濟與管理	25,677 (3.5%)	2,582 (2.8%)	2,646 (3.0%)	12,041 (2.5%)	1,387 (2.3%)	1,434 (2.4%)
天文物理學	8,492 (1.1%)	588 (0.6%)	858 (1.0%)	8,138 (1.7%)	796 (1.3%)	818 (1.4%)
化學	7,697 (1.0%)	831 (0.9%)	1,063 (1.2%)	5,880 (1.2%)	728 (1.2%)	775 (1.3%)
人文科學	2,358 (0.3%)	205 (0.2%)	176 (0.2%)	1,697 (0.3%)	113 (0.2%)	119 (0.2%)
總計	722,740 (100%)	90,106 (100%)	86,184 (100%)	474,200 (100%)	58,372 (100%)	57,620 (100%)

表 1: AESW 2016 各領域資料的詳細統計

超參數	ELECTRA-small	ELECTRA-base	ConvBERT-small	ConvBERT-base	SynBERT-base
編碼器層數	12	12	12	12	12
隱藏層維度	256	768	256	768	896
詞嵌入維度	128	768	128	768	768
生成器／判別器大小比例	1/4	1/3	1/4	1/3	1/3
注意力頭數	4	12	4	12	14
每個注意力頭維度	64	64	64	64	64
學習率	3×10^{-4}	2×10^{-4}	3×10^{-4}	2×10^{-4}	2×10^{-4}
Dropout	0.1	0.1	0.1	0.1	0.1
批次大小	128	256	128	256	64
輸入序列長度	128	512	128	128	128
訓練步數	1,000,000	1,000,000	1,000,000	1,000,000	1,000,000

表 2: 預訓練完整超參數設定

4 實驗設定

4.1 資料集

本文採用 AESW 2016 的科學英文寫作評估資料集 (Daudaravicius et al., 2016)，資料由 Springer 與 VTex 的專業英文編輯者，從 2006 至 2013 年出版的 9,919 篇科學論文整理而成，涵蓋九個科學領域，並以數學及物理爲主。表 1 列出九個領域在訓練、發展與測試集中的無修改與需語言編輯句子數量及其分布比例。爲保護個人資訊，作者、機構、引文、網址與數學公式等內容以代稱取代；需要刪除與插入的片段分別以 `<del>` 與 `<ins>` 標記。本文使用資料所提供的句子標籤：True 表示句子需要語言編輯，False 表示不需要。

4.2 預訓練設定

考量 AESW 2016 涵蓋多個科學領域，本文選用 S2ORC 資料 (Lo et al., 2020) 中與任務較相關的電腦科學、經濟學、工程學、數學及物理學文獻作爲預訓練語料，共約 234 萬篇論文。實驗使用 DGX-1 的單張 V100 GPU（32 GB 記憶體），模型初始設定參照 ELECTRA (Clark et al., 2020) 與 ConvBERT (Jiang et al., 2020)，再配合合成器混合注意力調整。SynBERT-base 使用 12 層編碼器、14 個注意力頭，每個頭的維度爲 64，隱藏層維度爲 896，詞嵌入維度爲 768，生成器大小爲判別器的三分之一。預訓練的學習率爲 2×10^{-4}、dropout 爲 0.1、批次大小爲 64、輸入序列長度爲 128，並以 RTD 目標訓練一百萬步。表 2

模型	語料	Precision	Recall	F1
ELECTRA-small	BERT 語料	61.77	68.74	65.07
ELECTRA-small	S2ORC	62.41	69.88	65.93
ConvBERT-small	OpenWebText	63.22	68.64	65.82
ConvBERT-small	S2ORC	69.08	64.04	66.47

表 3: 不同預訓練語料的發展集結果

模型	S:C:R	Precision	Recall	F1
ConvBERT	1:1:0	67.44	64.69	66.04
SynBERT	1:1:1	67.38	63.21	65.23
SynBERT	3:3:1	67.01	65.49	66.24

表 4: 合成器混合注意力比例的發展集結果

模型	Precision	Recall	F1
Knowlet	62.41	36.85	46.34
HITS	37.65	**94.80**	53.89
UW–Stanford	41.45	82.01	55.07
ISWD	44.82	72.79	55.48
NTNU–YZU	50.25	77.85	61.08
HU	54.44	74.13	62.78
BERT-base	**65.63**	61.67	63.59
EMC-NN	53.78	78.03	63.67
SciBERT-base	65.56	62.34	63.91
RoBERTa-base	62.95	65.61	64.25
ELECTRA-base	63.79	65.36	64.56
ConvBERT-base	64.19	65.30	64.74
SynBERT-base	64.62	65.91	**65.26**

表 5: AESW 2016 測試集模型比較結果

列出實驗所採用之 ELECTRA、ConvBERT 與 SynBERT 各模型規模的完整預訓練超參數。

4.3 訓練流程與微調設定

整體實驗依三個步驟進行。第一步以 ELECTRA-small 與 ConvBERT-small 比較通用語料及 S2ORC 預訓練結果，確認科學語料的領域特化效果。第二步使用 S2ORC 與 base 規模模型比較注意力頭比例；爲降低比例分析的訓練成本，此階段將預訓練步數設爲二十萬步，分別測試 1:1:1（4、4、4 個頭）及 3:3:1（6、6、2 個頭）。第三步依前兩步結果選用 S2ORC 與 3:3:1 配置，完成一百萬步預訓練，再於 AESW 2016 上進行微調與測試。

微調時以訓練集更新句子分類模型，並以發展集選擇超參數；經多次微調實驗後，採用學習率 0.0001、批次大小 64，並訓練 3 個輪次，最後僅在測試集上報告模型比較結果。

上述程序將模型選擇與最終評估區隔：語料來源及注意力比例皆以發展集 F1 分數進行比較，確定完整 SynBERT 架構後，才使用完整預訓練模型在 AESW 2016 測試集上與共享任務系統及預訓練語言模型基線比較。

5 實驗結果與分析

5.1 領域語料效果

爲驗證領域語料的影響，我們先以 ELECTRA-small 與 ConvBERT-small，比較通用語料和 S2ORC 科學語料的預訓練效果。發展集的實驗數據如表 3 所示，ELECTRA-small 使用 S2ORC 後，F1 分數由 65.07 提升至 65.93；ConvBERT-small 使用 S2ORC 後，F1 分數則由 65.82 提升至 66.47。對兩種模型而言，使用科學論文語料均取得較高的 F1 分數，顯示領域特化預訓練有助於模型學習更符合 AESW 2016 任務的語言表示。

5.2 注意力比例效果

合成器混合注意力的比例實驗，使用預訓練 20 萬步的 base 規模模型進行。發展集的實驗數據如表 4 所示，當三種機制以 1:1:1 的比例平均配置時，SynBERT 的 F1 分數爲 65.23，低於 ConvBERT 的 66.04。此結果顯示，輸入無關的 random synthesized attention 若占比過高，可能削弱模型對輸入語意與局部語序的建模能力。當比例調整爲 3:3:1 時，SynBERT 的 F1 分數提升至 66.24，高於 ConvBERT。這說明 random synthesized attention 較適合作爲補充性的全域結構訊號，不宜與 self-attention 及 span-based dynamic convolution 等量配置。

5.3 基線模型比較

表 5 比較 SynBERT-base、AESW 2016 六個共享任務系統、EMC-NN，以及多個預訓練語言模型基線。六個共享任務系統的結果取自 Daudaravicius et al. (2016)；BERT 與 EMC-NN 的結果取自 Lee et al. (2021b)，其餘預訓練模型則依本文相同的 AESW 2016 設定評估 (Beltagy et al., 2019; Liu et al., 2019; Clark et al., 2020; Jiang et al., 2020)。

HITS 的召回率最高（94.80），但精確率僅 37.65，顯示其判斷較偏向高召回；BERT 的精確率最高（65.63），EMC-NN 則以多通道集成將 F1 提升至 63.67。完成預訓練與微調後，SynBERT-base 在 AESW 2016 測試集上取得 64.62 的精確率、65.91 的召回率，以及 65.26 的 F1 分數，相較 HU 與 EMC-NN 分別提高 2.48 與 1.59 個百分點，顯示在精確率與召回率之間取得較佳平衡。

5.4 錯誤分析

我們從 SynBERT-base 在 AESW 2016 測試集中的錯誤分類樣本隨機抽取 1,000 句進行人工分析，並將錯誤歸納爲標點符號錯誤

類別	常見現象與原始例句
標點符號	標點符號缺漏、多餘或誤用。 EIT<del> </del><ins>-</ins>wave trains detected thus far are always accompanied by funnel QFPs, but not vice versa.
詞彙	大小寫、拼字或用詞錯誤。 ~~each~~<ins>Each</ins> cycle of _MATH_ is homotopic to the corresponding cycle of _MATH_.
文法	關係代名詞、介詞或動詞時態錯誤。 _CITE_ ~~describe~~<ins>has described</ins> such calculations for a variety of solar activity indices.
句法結構	語序不自然或含多餘片段。 ~~Events~~<ins>However, events</ins> are not ubiquitously distributed in the active region<del> though</del>.

表 6: SynBERT 錯誤類別與範例

（45.0%）、詞彙錯誤（22.5%）、文法錯誤（21.0%）與句法結構錯誤（11.5%）四類。標點符號類包含符號缺漏、多餘或使用錯誤；詞彙類包含專有名詞或句首字母大小寫、拼字及用詞錯誤；文法類包含關係代名詞、介詞與動詞時態錯誤；句法結構類則包含語序不自然及多餘片段。表 6 列出各類別的實際例句，其中 <del> 與 <ins> 分別標示應刪除與插入的內容。這些結果顯示，模型雖能掌握多種語意與語法特徵，但面對細緻的局部編輯時仍容易產生誤判。

6 結論與未來工作

本文提出 SynBERT，將 ELECTRA 的替換詞元偵測預訓練框架應用於科學語言編輯，並使用 S2ORC 科學語料進行領域特化預訓練。模型的核心改良爲合成器混合注意力，並以 3:3:1 的比例整合自注意力、區間動態卷積與隨機合成注意力。SynBERT 在 AESW 2016 二元分類任務中取得 65.26 的 F1 分數，高於共享任務系統及多個預訓練語言模型基線。

未來研究可從三個方向延伸。第一，針對符號與句法結構錯誤，設計更細緻的輔助任務或資料增強策略。第二，將句子層級的二元分類擴展爲序列標記或生成式校正，以指出錯誤位置並提供修改建議。第三，將相同的領域特化流程應用於中文學術寫作或其他專業領域，以建立符合特定文體需求的語言編輯模型。

Acknowledgments

This study was partially supported by the National Science and Technology Council, Taiwan, under the grant NSTC 114-2221-E-A49-059-MY3. This work was also financially supported by the Co-creation Platform of the Industry Academia Innovation School, NYCU.

References

Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. SciBERT: a pretrained language model for scientific text. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, pages 3615–3620.

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

Vidas Daudaravicius, Rafael E. Banchs, Elena Volodina, and Courtney Napoles. 2016. A report on the automatic evaluation of scientific writing Shared Task. In *Proceedings of the 11th Workshop on Innovative Use of NLP for Building Educational Applications*, pages 53–62.

Christopher Davis, Christopher Bryant, Andrew Caines, Marek Rei, and Paula Buttery. 2022. Probing for targeted syntactic knowledge through grammatical error detection. In *Proceedings of the 26th Conference on Computational Natural Language Learning*, pages 360–373.

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, Volume 1 (Long and Short Papers)*, pages 4171–4186.

Dan Flickinger, Michael Goodman, and Woodley Packard. 2016. UW–Stanford system description for AESW 2016 Shared Task on grammatical error detection. In *Proceedings of the 11th Workshop on Innovative Use of NLP for Building Educational Applications*, pages 105–111.

Filip Gralinski, Ryszard Staruch, and Krzysztof Jurkiewicz. 2025. Oddballness: universal anomaly detection with language models. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 2683–2689.

Zihang Jiang, Weihao Yu, Daquan Zhou, Yunpeng Chen, Jiashi Feng, and Shuicheng Yan. 2020. ConvBERT: improving BERT with span-based dynamic convolution. In *Advances in Neural Information Processing Systems 33*, pages 12837–12848.

Masahiro Kaneko, Yuya Sakaizawa, and Mamoru Komachi. 2017. Grammatical error detection using error- and grammaticality-specific word embeddings. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing, Volume 1: Long Papers*, pages 40–48.

Sudhanshu Kasewa, Pontus Stenetorp, and Sebastian Riedel. 2018. Wronging a right: generating better errors to improve grammatical error detection. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4977–4983.

Lung-Hao Lee, Li-Ping Chang, Kuei-Ching Lee, Yuen-Hsien Tseng, and Hsin-Hsi Chen. 2013. Linguistic rules based Chinese error detection for second language learning. In *Work-in-Progress Poster Proceedings of the 21st International Conference on Computers in Education*, pages 27–29.

Lung-Hao Lee, Li-Ping Chang, and Yuen-Hsien Tseng. 2016a. Developing learner corpus annotation for Chinese grammatical errors. In *Proceedings of the 20th International Conference on Asian Language Processing*, pages 254–257.

Lung-Hao Lee, Man-Chen Hung, Chao-Yi Chen, Rou-An Chen, and Yuen-Hsien Tseng. 2021a. Chinese grammatical error detection using adversarial ELECTRA transformers. In *Proceedings of the 29th International Conference on Computers in Education*, pages 111–113.

Lung-Hao Lee, Bo-Lin Lin, Liang-Chih Yu, and Yuen-Hsien Tseng. 2016b. The NTNU–YZU system in the AESW Shared Task: automated evaluation of scientific writing using a convolutional neural network. In *Proceedings of the 11th Workshop on Innovative Use of NLP for Building Educational Applications*, pages 122–129.

Lung-Hao Lee, Bo-Lin Lin, Liang-Chih Yu, and Yuen-Hsien Tseng. 2017. Chinese grammatical error detection using a CNN-LSTM model. In *Proceedings of the 25th International Conference on Computers in Education*, pages 919–921.

Lung-Hao Lee, Yuen-Hsien Tseng, and Li-Ping Chang. 2018. Building a TOCFL learner corpus for Chinese grammatical error diagnosis. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation*, pages 2298–2302.

Lung-Hao Lee, Yuh-Shyang Wang, Po-Chen Lin, Chih-Te Hung, and Yuen-Hsien Tseng. 2020. Multi-channel CNN–BiLSTM for Chinese grammatical error detection. In *Proceedings of the 28th International Conference on Computers in Education*, pages 558–560.

Lung-Hao Lee, Liang-Chih Yu, Kuei-Ching Lee, Yuen-Hsien Tseng, Li-Ping Chang, and Hsin-Hsi Chen. 2014. A sentence judgment system for grammatical error detection. In *Proceedings of the 25th International Conference on Computational Linguistics: System Demonstrations*, pages 67–70.

Lung-Hao Lee, Yuh-Shyang Wang, Chao-Yi Chen, and Liang-Chih Yu. 2021b. Ensemble multichannel neural networks for scientific language editing evaluation. *IEEE Access*, volume 9, pages 158540–158547.

Tzu-Mi Lin, Chao-Yi Chen, Lung-Hao Lee, and Yuen-Hsien Tseng. 2022. Evaluating the performance of Chinese multi-label grammatical error detection using deep neural networks. In *Proceedings of the 30th International Conference on Computers in Education*, pages 524–526.

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. *arXiv preprint arXiv:1907.11692*.

Kyle Lo, Lucy Lu Wang, Mark Neumann, Rodney Kinney, and Daniel S. Weld. 2020. S2ORC: the Semantic Scholar open research corpus. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4969–4983.

Gaetan Lopez Latouche, Marc-André Carbonneau, and Benjamin Swanson. 2024. Zero-shot cross-lingual transfer for synthetic data generation in grammatical error detection. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 3002–3016.

Liliana Mamani Sanchez and Hector-Hugo Franco-Penya. 2016. Combined tree kernel-based classifiers for assessing quality of scientific text. In *Proceedings of the 11th Workshop on Innovative Use of NLP for Building Educational Applications*, pages 223–228.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S. Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems 26*, pages 3111–3119.

Ryo Nagata, Manabu Kimura, and Kazuaki Hanawa. 2022. Exploring the capacity of a large-scale masked language model to recognize grammatical errors. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 4107–4118.

The Quyen Ngo, Thi Minh Huyen Nguyen, and Phuong Le-Hong. 2023. Two neural models for multilingual grammatical error detection. In *Proceedings of the 12th Workshop on NLP for Computer Assisted Language Learning*, pages 40–44.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. GloVe: global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pages 1532–1543.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI Technical Report*.

Marek Rei and Helen Yannakoudakis. 2016. Compositional sequence labeling models for error detection in learner writing. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, Volume 1: Long Papers*, pages 1181–1191.

Madeline Remse, Mohsen Mesgar, and Michael Strube. 2016. Feature-rich error detection in scientific writing using logistic regression. In *Proceedings of the 11th Workshop on Innovative Use of NLP for Building Educational Applications*, pages 162–171.

Allen Schmaltz, Yoon Kim, Alexander M. Rush, and Stuart M. Shieber. 2016. Sentence-level grammatical error identification as sequence-to-sequence correction. In *Proceedings of the 11th Workshop on Innovative Use of NLP for Building Educational Applications*, pages 242–251.

Yi Tay, Dara Bahri, Donald Metzler, Da-Cheng Juan, Zhe Zhao, and Che Zheng. 2021. Synthesizer: rethinking self-attention for transformer models. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 10183–10192.

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 30*, pages 5998–6008.

Elena Volodina, Christopher Bryant, Andrew Caines, Orphée De Clercq, Jennifer-Carmen Frey, Elizaveta Ershova, Alexandr Rosen, and Olga Vinogradova. 2023. MultiGED-2023 Shared Task at NLP4CALL: multilingual grammatical error detection. In *Proceedings of the 12th Workshop on NLP for Computer Assisted Language Learning*, pages 1–16.

René Witte and Bahar Sateli. 2016. Combining off-the-shelf grammar and spelling tools for the automatic evaluation of scientific writing (AESW) Shared Task 2016. In *Proceedings of the 11th Workshop on Innovative Use of NLP for Building Educational Applications*, pages 252–255.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Ruslan Salakhutdinov, and Quoc V. Le. 2019. XLNet: generalized autoregressive pretraining for language understanding. In *Advances in Neural Information Processing Systems 32*, pages 5754–5764.

Zheng Yuan, Shiva Taslimipoor, Christopher Davis, and Christopher Bryant. 2021. Multi-class grammatical error detection for correction: a tale of two systems. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 8722–8736.