Avaliando Abordagens do Estado da Arte para Reconhecimento de Textos Artísticos

Experiência Criativa: Projeto Transformador II
Bacharelado em Ciência da Computação – PUCPR

Turma U – Equipe 7
Lucas Azevedo Dias, Henrique Anderle Schulz, Rafaela de Miranda, Guilherme de Lara Peres, Pedro Lucas Bittencourt
{dias.azevedo, henrique.schulz, r.miranda2, guilherme.peres, pedro.bittencourt}@pucpr.edu.br

I. Introdução

Figura 1. Exemplos de imagens que compõem o *dataset* WordArt-V1.5 [1]

Textos artísticos consistem em textos decorados por artistas com propósito extremamente estético podendo apresentar variedades grandes de formas, cores e vários tipos de distorções como rotação, deformação, substituição de letras por objetos, sombras etc. Dessa forma, diferentemente de textos convencionais, esses tipos de textos beiram o limiar do compreensível pelo foco na estética e nas mensagens subliminares conforme 1 [2].

Pode-se destacar que o reconhecimento de textos artísticos (ATR) é uma área intimamente relacionada ao reconhecimento de textos em cena (STR), uma vez que ambas têm como foco o reconhecimento de textos dispersos em cenários complexos. No entanto, elas divergem de modo que o ATR apresenta uma maior variabilidade devido ao seu foco artístico, os textos em cena tendem a ser mais previsíveis, geralmente dispostos de forma horizontal e utilizando fontes padronizadas, como ilustrado na Figura 2 [2] [3].

Com efeito, é possível dizer que existe um potencial de melhoria da área de reconhecimento de textos caso o ATR seja desenvolvido devido a sua dificuldade intrínseca de generalização dos modelos se comparado à outros tipos de

(a) Exemplos de imagem de ATR retirada do *dataset* WordArt-V1.5 [1].

(b) Exemplos de imagem de STR retirada do *dataset* Union14M [4].

Figura 2. Comparação entre uma imagem de ATR e uma imagem de STR.

reconhecimento. Para tanto, foi realizado uma competição em 2024 da *International Conference on Document Analysis and Recognition* (ICDAR) sobre a área com 33 participantes, onde o foco era a exploração do estado da arte e a criação de novas soluções. Dessa forma, incentivando novas pesquisas na área de textos artísticos e de reconhecimento de textos como um todo [1].

O objetivo do presente artigo é explorar o estado do conhecimento atual para o ATR e criar uma solução customizada competitiva com as apresentadas na competição da ICDAR. Contudo, é importante salientar que a regra de proibição do uso de pesos pré-treinados para modelos não será seguida devido a implicações de tempo e de recursos limitados.

Na competição, foi proposto e utilizado o conjunto de dados WordArt-V1.5, uma expansão do *dataset* WordArt-V1 originário da competição de 2022 da *European Conference on Computer Vision* (ECCV). Assim, o conjunto da dados é composto por uma grande variedade de imagens coletadas em pôsteres, em cartões de felicitação, em capas, em *outdoors* etc. e que foram divididas em 6.000 imagens para Treino, 3.000 para Teste A e 3.000 para Teste B. Onde, o Teste A foi utilizado como primeira etapa para selecionar os dez melhores para a avaliação do Teste B [1].

Para a solução proposta neste artigo, foram utilizados os conjuntos do *dataset* de Treino, de Teste A e de Teste B, destinados, respectivamente, ao treinamento, à validação e ao teste da solução proposta.

As Figuras 3 e 4 mostram análises estatísticas sobre o *dataset*. Assim, as distribuições dos tamanhos e das frequências

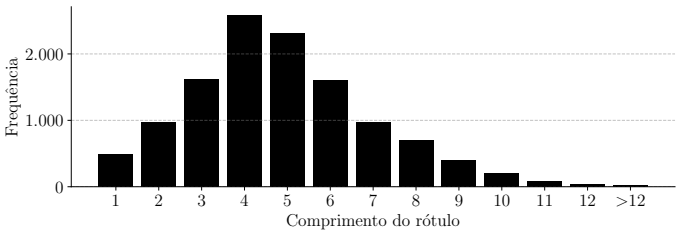

Figura 3. Gráfico evidenciando a frequência dos comprimentos de rótulos no *dataset* WordArt-V1.5. Elaborado pelos autores com base em Xie *et al.* [1].

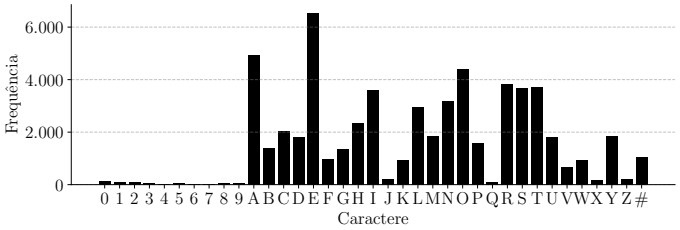

Figura 4. Gráfico evidenciando a frequência dos caracteres de rótulos no *dataset* WordArt-V1.5 com todos os símbolos combinados na classe "#". Elaborado pelos autores com base em Xie *et al.* [1].

dos caracteres se aproximam das distribuições encontradas no *corpus* da língua inglesa [1].

Como parâmetro de avaliação das soluções na competição foi utilizado a acurácia de reconhecimento de palavras (WRA), a qual foi definida como:

$$W_{RA} = \frac{W_r}{W}, \tag{1}$$

onde W é a quantidade total de palavras e W_r representa as palavras corretamente reconhecidas ignorando símbolos e a capitalização das letras [1].

Tabela I
CLASSIFICAÇÃO DA COMPETIÇÃO ICDAR 2024 EM ATR (TESTE B) [1]

Posição	Equipe	WRA (%)
1	Ocr For WordArt	91,07
2	ViettelAI-OCR	90,77
3	Let Me See	89,77
4	iPad_OCR	89,27

Considerando esse parâmetro, a classificação das melhores soluções de acordo com o teste B pode ser vista na Tabela I. Deste modo, a equipe **Ocr For WordArt** com WRA de 91.07% utilizou um *ensemble* de quatro modelos, sendo dois supervisionados (LVP [5] e MGP-STR [6]), um semissupervisionado baseado na arquitetura *Mean-Teacher* [7] e um autossupervisionado de autoria própria (*Symmetric Superimposition Model*). Como estratégia de inferência do *ensemble*, utilizaram uma estratégia de votação baseada no maior produto cumulativo dos *logits* de cada letra. Porém, para casos onde a palavra predita por um modelo coincidisse com a de outro, era realizado a soma das pontuações desses modelos [1].

A segunda colocação, equipe **ViettelAI-OCR** (90.77%), escolheu um ensemble com alguns modelos baseados em transformador de visão (ViT), dentre eles, o de maior destaque foi o modelo Sequência Autorregressiva Permutada (PARSeq) [8], o qual é baseado em modelo de linguagem (LM) autorregressivo (AR), além do ViT. Dessa forma, o time realizou um treinamento massivo com aprendizado autossupervisionado aplicando o método de destilação de caractere em caractere (CCD) [9] com objetivo de segregar detalhadamente o plano de fundo dos caracteres e obter uma maior generalização dos modelos [1].

A terceira equipe, **Let Me See** (89.77%), empregou o modelo Reconhecimento com *Autoencoder* com Máscara (MAERec) [4] com perda contrastiva de caracteres (CC Loss) [2] para minimizar o impacto da variedade de fontes e de estilos dos dados artísticos [1].

A quarta equipe, **iPad_OCR** (89.27%), criou um *ensemble* usando os modelos PARSeq [8], MAERec [4] e CornerTransformer [2]. Cada modelo foi treinado individualmente e, em seguida, aplicou-se uma estratégia de votação para determinar a predição final. Entretanto, quando todas as predições divergiam entre si, optou-se por priorizar o resultado do modelo PARSeq, uma vez que ele apresentou o melhor desempenho individual [1].

Mais adiante, serão abordados trabalhos relacionados do estado da arte, bem como a metodologia adotada para o desenvolvimento da solução proposta, suas principais escolhas de implementação e os resultados finais obtidos em comparação com as equipes previamente apresentadas.

II. TRABALHOS RELACIONADOS

Nesta etapa, serão discutidos os principais artigos pertinentes para a compreensão do estado da arte de ATR. Assim, o objetivo é contextualizar as abordagens existentes, identificar tendências de pesquisa e destacar lacunas encontradas.

Tabela II
COMPARAÇÃO ENTRE MODELOS DO ESTADO DA ARTE COM DATASETS DE BENCHMARK

Modelo	Acurácia (%)				
	IIIT5k	SVT	IC15	SVTP	U14M-B[a]
ViTSTR [10]	88,4	87,7	72,6	81,8	-
CornerTransformer [2]	95,9	94,6	86,3	91,5	-
PARSeq [8]	99,1	97,9	89,6	95,7	84,3[b]
MAERec [4]	98,5	97,8	89,5	94,4	85,2
SVTRv2 [11]	99,2	98,0	91,1	99,0	86,1

[a] Refere-se a Union14M-Benchmark.
[b] Segundo Du *et al.* [11], devido a não existência do *dataset* Union14M-Benchmark à época de publicação do artigo de Bautista e Atienza [8].

Para colaborar com a compreensão dos resultados obtidos por cada modelo do estado da arte, será disponibilizada a Tabela II, a qual contém a performance de cada modelo medida em acurácia para vários *datasets* de *benchmark* de STR: (i) IIIT5k [12], (ii) SVT [13], (iii) IC13 [14], (iv) IC15 [15], (v) e SVTP [16]. Vale frisar que ambos, os artigos abordados e os modelos na Tabela II, foram ordenados de acordo com seu tempo de publicação.

A. ViTSTR

O modelo Transformador de Visão para Reconhecimento de Textos em Cena (ViTSTR) proposto por Atienza [10]

foi um dos primeiros a abordar o STR apenas utilizando ViT e se tornaram uma referência para modelos futuros que beberiam da mesma fonte. Deste modo, o ViTSTR inovou ao utilizar uma arquitetura de estágio único proporcionada pelo ViT, indo em contrapartida aos modelos anteriores baseados em redes neurais convolucionais (CNNs) para extração de características.

No método proposto, a imagem do texto é segmentada em pequenos blocos não sobrepostos, que são processados pelo ViT para extrair representações visuais relevantes. A partir dessas representações, o modelo realiza a predição dos caracteres da palavra de forma paralela, dispensando o uso de redes recorrentes, módulos de retificação ou mecanismos de atenção adicionais comumente empregados em abordagens tradicionais. Essa estrutura simplificada permite que o sistema opere em uma única etapa principal, reduzindo significativamente a complexidade computacional sem comprometer o desempenho [10].

Apesar dos avanços apresentados, Atienza [10] também aponta limitações e desafios abertos relacionados ao desempenho do ViTSTR. Um dos principais problemas observados refere-se às falhas de reconhecimento em textos com alta irregularidade, como aqueles dispostos em curvas, com fortes oclusões, baixa resolução ou distorções extremas. Nessas situações, o modelo demonstra dificuldade em preservar a acurácia, além de apresentar confusões frequentes entre caracteres visualmente semelhantes.

B. CornerTransformer

Considerando as limitações dos modelos de STR da época quando aplicados ao cenário mais complexo do ATR, Xie *et al.* [2] propuseram uma abordagem inovadora voltada especificamente para esse desafio. Mais especificamente, os modelos anteriores apresentavam desempenho restrito em textos com orientação irregular, formas arbitrárias e fontes não padronizadas, sendo majoritariamente eficazes apenas em textos horizontais e bem estruturados. Para superar essas limitações, os autores introduziram o CornerTransformer, uma arquitetura que integra um *encoder* guiado por contornos e um *decoder* baseado em ViT.

O *encoder* de contornos é projetado para capturar características estruturais explícitas das formas de texto por meio da detecção de mapas de contorno, as quais fornecem informações geométricas mais precisas sobre o formato das letras. Assim, esses mapas são então fundidos, através um mecanismo de *cross-attention*, com as representações visuais extraídas no *backbone*. Portanto, enriquecendo o fluxo de informação e auxiliando o *decoder* na reconstrução sequencial e na predição mais robusta de textos com curvaturas e distorções variadas. Com essa abordagem, o CornerTransformer alcançou resultados de estado da arte em vários conjuntos de dados de texto em forma livre e demonstrou a eficácia da incorporação explícita de indícios estruturais no reconhecimento de texto arbitrário [2].

Contudo, Xie *et al.* [2] ressaltam a dependência do modelo na qualidade dos mapas de contornos e a dificuldade em imagens com caracteres juntos, sobrepostos ou com símbolos semelhantes a caracteres.

C. PARSeq

Bautista e Atienza [8], para o problema de STR, propuseram o modelo PARSeq baseado em LM AR, o qual apresenta uma nova abordagem ao usar permutação de modelos de linguagem (PLM) em seu treinamento. Dessa forma, a PLM surge como um melhoramento ao uso de LMs ARs, os quais são limitados: (i) pelo viés de unidirecionalidade advindo da predição baseada em caracteres anteriores (ii) e pela restrição de decodificação monotônica, ou seja, o modelo fica incapaz de usar o contexto em outras direções para prever o caractere. Portanto, Bautista e Atienza [8], ao usarem a PLM, conseguem criar um modelo com maior capacidade de generalização e de ciência de contexto.

A arquitetura do PARSeq segue um paradigma *encoder-decoder*, no qual o codificador baseado em ViT extrai características visuais da imagem e o decodificador visual-linguístico, processa se baseando no contexto visual e linguístico para gerar a sequência textual final [8].

No entanto, como o número de permutações possíveis cresce exponencialmente com o comprimento da sequência, a execução completa de todas as combinações no treinamento é inviável. Para contornar essa limitação, o PARSeq adota uma seleção amostral de permutações, mantendo o equilíbrio entre diversidade e custo computacional [8].

D. MAERec

Jiang *et al.* [4], em seu trabalho, atribuíram a má performance dos modelos do estado da arte em dados reais aos *datasets* de *benchmark* da época, os quais não mais representavam os desafios encontrados em situações do mundo real. Portanto, os autores propuseram um novo conjunto maciço, Union14M, tanto para treinamento quanto para validação dos modelos em casos reais. Porém, os mesmos apresentam um solução capaz de superar os modelos nestes dados mais representativos, a qual foi denominada de MAERec.

O modelo MAERec utiliza um *backbone* baseado em ViT com um *decoder* AR. Assim, sua principal inovação é o pré-treinamento autossupervisionado com *autoencoder* com máscaras (MAE), onde blocos não-rotulados de imagens são mascarados para forçar o modelo a aprender com representações grosseiras da estrutura do texto a partir de um contexto limitado. Portanto, apenas após o estágio de pré-treinamento, é que o modelo efetivamente passa pelo ajuste fino com dados rotulados [4].

Todavia, embora o modelo MAERec tenha alcançado uma acurácia de 85,2% no conjunto Union14M-Benchmark — superando significativamente os demais modelos de estado da arte nesse cenário com 10,6% de ganhos sobre o segundo melhor — Jiang *et al.* [4] destacam que o problema do STR ainda está longe de ser resolvido.

E. SVTRv2

Du *et al.* [11] identificaram no estado da arte que modelos baseados em classificação temporal conexionista (CTC), anteriormente colocados de lado em favor do uso de *encoders* baseados em ViT, poderiam utilizar de novos métodos para confrontar eles e endereçar o problema de número máximo de caracteres preditos. Para tal, os autores utilizaram módulos de correção focados em transformar a imagem em um formato menos irregular; porém, ao contrário de outros modelos do estado da arte que utilizaram tamanhos fixos de imagem para os módulos, os autores criaram um módulo inicial denominado redimensionamento multi-tamanho (MSR). Dessa maneira, o MSR efetua redimensionamentos de acordo com as proporções da imagem para remover distorções indesejadas.

Na etapa de extração de características, Du *et al.* [11] utilizaram, além das camadas convolucionais, blocos de combinação de informações, os quais agregam contextos locais ou globais. Para os contextos locais, foram usados convoluções e, para contextos globais, foram usados autoatenções multi-cabeças (MHSAs).

Posteriormente, são aplicados o módulo de reorganização de características (FRM) e o módulo de orientação semântica (SGM) simultaneamente. Assim, o FRM realiza o ajuste do alinhamento do texto e o SGM adiciona o contexto linguístico para dedução de letras com base em suas vizinhas [11].

F. Síntese

Após a revisão do estado da arte, observa-se uma ampla diversidade de abordagens aplicadas aos problemas de STR e ATR. Diante disso, uma direção promissora consiste em combinar diferentes métodos, de modo a explorar as melhores características de cada abordagem e, assim, potencializar o desempenho geral das soluções.

III. Metodologia

Nesta seção, será detalhadamente apresentada a solução proposta, acompanhada de uma exposição clara do raciocínio e dos critérios que fundamentam as escolhas realizadas para sua concepção. Serão discutidos os princípios, métodos e justificativas que orientaram cada etapa da proposta, permitindo ao leitor compreender não apenas o resultado final, mas também o processo de análise e de tomada de decisão que levaram até ela.

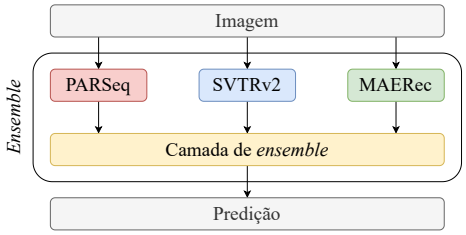

Figura 5. Diagrama da solução de *ensemble* proposta. Nela, pode ser averiguado o fluxo dos dados dentro do *ensemble* que passa pelos modelos individualmente e são combinados para gerar a predição final através da camada de *ensemble*.

A. Considerações iniciais

Inicialmente, foi decido realizar um *ensemble* de modelos por votação devido a dois principais fatores: (i) a dificuldade de generalização do *dataset* pela quantidade de fontes e de estilos encontrados nele (ii) e a alta taxa de utilização desta técnica pelas soluções campeãs da competição [1].

Tabela III
Resultados do ajuste fino dos modelos do ensemble no Teste A

Modelo	WRA (%)
SVTRv2 [11]	85,80
PARSeq [8]	83,97
MAERec [4]	83,40
ViTSTR [10]	78,80
CornerTransformer [2]	73,80

Após isso, para decidir os modelos que participariam do *ensemble*, foram estudados os melhores resultados encontrados na competição abordada e foram pesquisados dentro do estado da arte os modelos mais promissores. Assim, foram separados cinco modelos promissores para serem testados com seus pesos pré-treinados sobre o *dataset*. Finalmente, conforme os resultados da Tabela III, foi possível observar que os três melhores modelos formam um grupo coeso graças à proximidade de WRA dentre eles, enquanto os modelos restantes ficam consideravelmente atrás. Portanto, foram selecionados os modelos PARSeq [8], MAERec [4] e *Single Visual Text Recognition* (SVTRv2) [11].

Contudo, para determinar o grau de ganhos possíveis através de uma fusão destes modelos, foi realizado uma análise utilizando a interseção dos erros dos modelos para averiguar a taxa de sobreposição destes. Para tal análise, foi utilizado o conjunto do Teste A com 3.000 imagens.

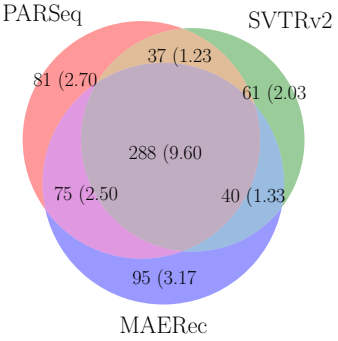

Figura 6. Diagrama de Venn com interseção dos erros de cada modelo usando pesos pré-treinados sobre o conjunto do Teste A com 3.000 imagens.

Conforme a Figura 6, é possível averiguar que, de um total de 677 erros possíveis (22,56%), 288 (9,60%) são comuns a todos os modelos, ou seja, apenas 42,54% de taxa de compartilhamento de erros. Ademais, considerando os erros comuns a todos os modelos, pode-se calcular que a acurácia teórica ideal do modelo seria de 90,40%. Assim, foi possível validar a viabilidade de um *ensemble* destes modelos comparando com os resultados dos outros competidores sobre o Teste A em consonância com a Tabela IV.

Tabela IV
MELHORES RESULTADOS DA COMPETIÇÃO ICDAR 2024 EM ATR NO TESTE A [1]

Posição	Equipe	WRA (%)
1	cyrilsterling	89,73
2	Ocr For WordArt	89,70
3	zichengli	89,67
4	ViettelAI-OCR	89,60
5	Let Me See	89,47
6	buftzwlsb	89,33
7	ML-LTU	88,70
8	iPad_OCR	87,50
9	MILKTEA-LOVE	85,20
10	Vision_Alpha	83,40

Após estes entendimentos, o foco se projetou a: (i) realizar o ajuste fino dos pesos pré-treinados dos modelos (ii) e construir a etapa de *ensemble* por votação.

B. Ajuste fino dos modelos

É essencial enfatizar que todos os modelos tiveram seu ajuste fino feito sobre o conjunto de Treino do *dataset* com uso do conjunto de Teste A para validação dos resultados.

Para o ajuste fino do modelo SVTRv2, foram realizadas diversas operações de aumento de dados *online*. Especificamente, aplicou-se rotações aleatórias de até 10 graus, translações de até 15% do tamanho da imagem, e variações de escala no intervalo de 0,85 a 1,15, além de um cisalhamento de até 10 graus. Complementarmente, foram inseridas perturbações de natureza visual, incluindo desfoque, variações de brilho, contraste e saturação de até 30%, além da adição de ruído aleatório.

Posteriormente, foi aplicada uma regularização espacial de apagamentos aleatórios, a qual consiste em apagar aleatoriamente regiões da imagem com uma probabilidade de, neste caso, 50%, com objetivo de simular oclusões ou perdas parciais de informação comumente encontradas em dados reais. Além disso, as áreas apagadas variam entre 2% e 10% da área total da imagem, com uma relação de aspecto entre 0,3 e 3,3.

O ajuste fino do modelo PARSeq usou uma estratégia de aumento de dados com um conjunto de operações de transformação que incluem rotações de até 15 graus, cisalhamentos horizontais e verticais com intensidades máximas de 90% e de 20%, respectivamente, e translações horizontais e verticais de até 10% e de até 30% do tamanho da imagem. Assim, apenas três operações são escolhidas por imagem com foco em aumentar a diversidade das transformações.

No caso do modelo MAERec, no processo de aumento de dados de seu ajuste fino, foram aplicadas transformações aleatórias com probabilidade de 50%, consistindo em rotações aleatórias de até 15 graus, transformações afins com variações de translação de até 30%, escala entre 0,5 a 2,0 do tamanho da imagem, cisalhamento (entre –45°e 45°) e, ainda, distorções de perspectiva com fator de deformação de 0,5.

Ademais, foram aplicadas com uma probabilidade adicional de 25%, a transformação de redimensionamento piramidal e outras operações baseadas na biblioteca Albumentations, como adição de ruído gaussiano com variância de 20 e desfoque de movimento com limite em 7. Além disso, havia uma probabilidade de mais 25% para aplicação de uma variação no brilho, na saturação e no contraste de 50% e de matiz de 10%.

Tabela V
RESULTADOS DO AJUSTE FINO DOS MODELOS DO ENSEMBLE NO TESTE A

Modelo	Pré-treinado (WRA %)	Ajuste fino (WRA %)
SVTRv2 [11]	85,80	86,23
PARSeq [8]	83,97	84,53
MAERec [4]	83,40	84,53

Através dessas técnicas foi possível chegar aos resultados apresentados na Tabela V. Em especial, é importante destacar os resultados do SVTRv2, o qual se consolidou como melhor modelo do *ensemble*.

C. Ensemble por votação

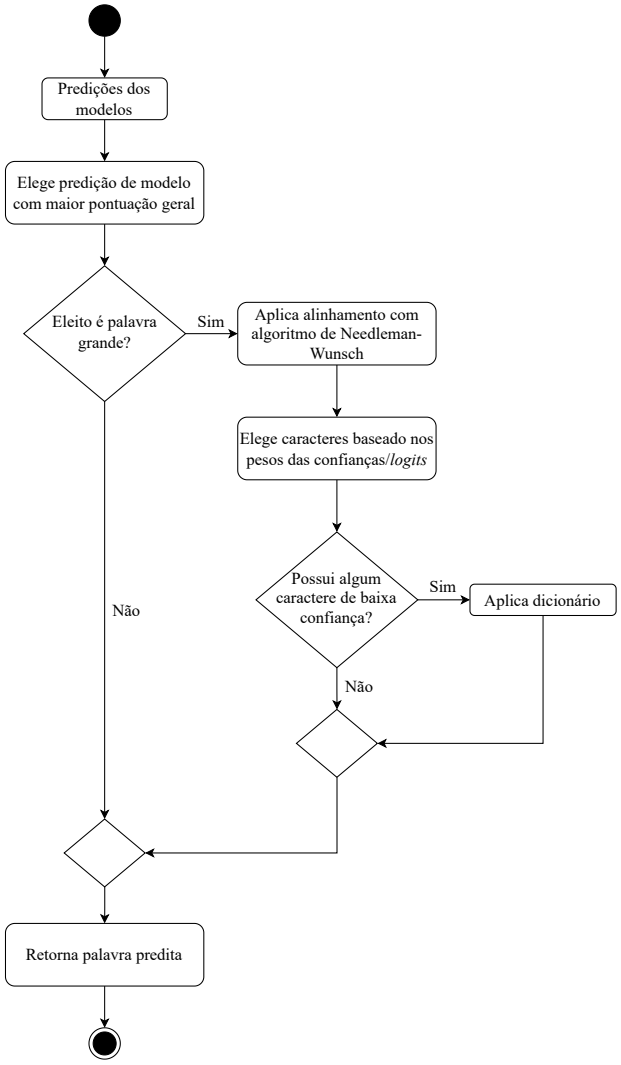

Figura 7. Diagrama de atividades representando lógica interna da camada final de agregação das predições dos modelos.

Para a camada de *ensemble* das predições dos modelos, foi utilizado inicialmente uma eleição dentre as predições dos modelos baseando-se na maior pontuação obtida. Assim, para o cálculo desta pontuação, realizou-se a separação das posições que possuem caracteres divergentes e, considerando apenas o *logit* deles, foi-se obtido a pontuação pelo valor mínimo encontrado neste vetor resultante. Contudo, caso não haja caracteres divergentes, usa-se o vetor de *logits* completo para o cálculo a fim de obter o resultado de maior confiança.

Desta maneira, a lógica aplicada para selecionar uma predição se baseia em tentar encontrar a de menor incerteza, onde um baixo valor de *logit* em um caractere de confusão dentre as predições dos modelos gera uma redução em seu valor para o *ensemble*.

Em termos formais, seja um conjunto de modelos

$$M = \{m_1, m_2, \ldots, m_k\},$$

em que cada modelo m_k gera uma sequência de *logits* e de predições de caracteres, denotadas respectivamente por

$$\mathbf{l}_k = [l_{k,1}, l_{k,2}, \ldots, l_{k,n}] \quad \text{e} \quad \hat{\mathbf{y}}_k = [\hat{y}_{k,1}, \hat{y}_{k,2}, \ldots, \hat{y}_{k,n}].$$

Primeiramente, define-se o conjunto de posições com divergências entre as predições dos modelos como:

$$D = \{j \in \{1, \ldots, n\} \mid \exists\, p, q, \hat{y}_{p,j} \neq \hat{y}_{q,j}\}. \tag{2}$$

A pontuação S_k de cada modelo m_k é então calculada considerando o valor mínimo de *logit* nas posições divergentes. Caso não haja divergências, utiliza-se todo o vetor de *logits*:

$$S_k = \begin{cases} \min\limits_{j \in D} l_{k,j}, & \text{se } |D| > 0, \\ \min\limits_{j \in \{1,\ldots,n\}} l_{k,j}, & \text{caso contrário.} \end{cases} \tag{3}$$

A predição final do *ensemble* é então escolhida como aquela associada ao modelo com a maior pontuação S_k, isto é, o modelo com menor incerteza nas posições de divergência:

$$\hat{\mathbf{y}}_{\text{ens}} = \hat{\mathbf{y}}_{k^*}, \quad \text{onde} \quad k^* = \arg\max_k S_k. \tag{4}$$

Com efeito, o processo de decisão pode ser dado como:

$$\hat{\mathbf{y}}_{\text{ens}} = \hat{\mathbf{y}}_{\arg\max_k \left(\min_{j \in D^*} l_{k,j}\right)}, \quad D^* = \begin{cases} D, & |D| > 0, \\ \{1, \ldots, n\}, & |D| = 0. \end{cases} \tag{5}$$

Vale frisar que as pontuações são sempre normalizadas de acordo com a média de confiança por caractere de cada modelo.

Contudo, constatou-se que este processo, por mais que eficiente para palavras menores, apresentou deficiência no tratamento de termos mais longos — com 9 ou mais caracteres, cerca de 749 itens (6,24%) de todo *dataset* conforme Figura 3. Portanto, constatou-se o claro problema de aumento na chance de erro da predição de uma palavra devido à má predição de um único caractere, onde cada caractere adicional representa uma oportunidade a mais para que uma previsão incorreta ocorra.

Formalizando, se a probabilidade de erro em cada caractere é p, então a probabilidade de uma palavra inteira de comprimento n ser predita sem erro é $(1-p)^n$. Assim, à medida que n aumenta, $(1-p)^n$ diminui, demonstrando que as palavras mais longas têm uma chance significativamente maior de conter pelo menos um erro, o que explica a queda de desempenho do *ensemble* de modelos em palavras mais longas.

Pensando em tratar estes casos, aplicou-se o algoritmo de Needleman-Wunsch [17] [18] e um método de dicionário em duas etapas.

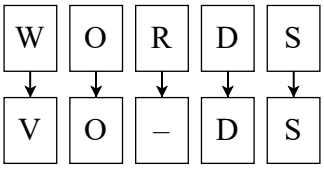

Figura 8. Exemplo de alinhamento de duas palavras com inserção de caractere de espaçamento na terceira posição para melhor comparação entre elas.

Na primeira etapa, o algoritmo de Needleman-Wunsch é usado para alinhar todas as predições dos modelos conforme Figura 8, as quais serão usadas para realizar uma eleição por caractere. Assim, a eleição é determinada pela maior confiança ajustada pela pontuação da predição (S_k) conforme a Equação 3.

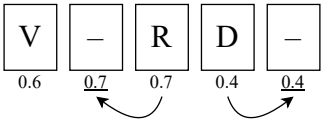

Figura 9. Exemplo de propagação da confiança pela palavra alinhada, onde um espaçamento ao meio recebe do próximo caractere e um ao final recebe do último caractere que não seja de espaçamento.

Para os casos em que o caractere sendo comparado é o caractere de espaçamento, será aplicado o *logit* do próximo caractere ou do último caractere — se não houver mais caracteres adiante — conforme Figura 9.

Posteriormente, essa nova predição, caso possua algum caractere com baixa confiança — inferior à 60% — passará pela etapa de dicionário.

Na etapa de dicionário, será calculado a palavra com maior similaridade de acordo com o cálculo da distância de Levenshtein aplicado à bigramas [19] dentre as palavras do dicionário `wlist_match3` [20]. Assim, caso a similaridade da palavra encontrada seja superior ao limiar de 70%, a predição terá seus caracteres substituídos, um a um, pelos caracteres da palavra similar, ignorando quaisquer símbolos presentes na predição ou caracteres adicionais além do seu tamanho original.

IV. Ambiente de execução

Para a execução dos experimentos, foi utilizado o Google Colab, configurado com uma GPU L4 com 22,5 GB de

memória dedicada e 53 GB de memória RAM do sistema. Ademais, os experimentos foram implementados em Python, utilizando a biblioteca PyTorch.

V. Resultados e Discussão

PARSeq: cherry (0.94)
SVTRv2: chenny (0.89)
MAERec: champag (0.74)
ENSEMBLE: cherry (0.94)
LABEL: cherry

PARSeq: faherine (0.79)
SVTRv2: echevins (0.83)
MAERec: feberias (0.72)
ENSEMBLE: echevins (0.83)
LABEL: echevins

Figura 10. Exemplos de predições corretas pelo *ensemble* de modelos.

PARSeq: cne (0.83)
SVTRv2: cne (0.91)
MAERec: one (0.82)
ENSEMBLE: cne (0.91)
LABEL: one

PARSeq: h (0.24)
SVTRv2: h (0.90)
MAERec: tt (0.46)
ENSEMBLE: h (0.90)
LABEL: it

Figura 11. Exemplos de predições incorretas pelo *ensemble* de modelos.

Com a execução do modelo proposto sobre o conjunto do *dataset* de Teste B, foi obtido uma WRA de **89,90%**. Na sequência serão feitas análises comparando o resultado da solução com os modelos que compõem o *ensemble* e com os outros modelos da competição.

Para ilustrar a execução do modelo, pode ser conferido alguns exemplos de predições corretas na Figura 10 e algumas incorretas na Figura 11.

A. Comparação entre ensemble e modelos

Tabela VI
Comparação entre o resultado do ensemble e dos modelos individualmente com ajuste fino (Teste B)

Modelo	WRA (%)	Diferença (%)
SVTRv2 [11]	88,13	-1,77
PARSeq [8]	87,60	-2,30
MAERec [4]	87,50	-2,40
Ensemble	**89,90**	**-**

A combinação das previsões dos modelos ajustados, por meio da técnica de *ensemble*, resultou em um ganho adicional de desempenho. Dessa maneira, o *ensemble* proposto foi capaz de integrar os pontos fortes individuais de cada arquitetura de forma à mitigar erros pontuais e aumentar a robustez final das predições. Como evidenciado na Tabela VI, o *ensemble*, atingindo uma WRA de 89,90%, superou por 1,77% a melhor rede individual (SVTRv2 com 88,13%).

Tabela VII
Comparação entre o resultado do ensemble e da competição ICDAR 2024 em ATR (Teste B) [1]

Posição	Equipe	WRA (%)
1	Ocr For WordArt	91,07
2	ViettelAI-OCR	90,77
3	Let Me See	89,77
4	iPad_OCR	89,27
-	***Ensemble***	**89,90**

B. Comparação entre ensemble e estado da arte

Para avaliar a relevância do método proposto no contexto do estado da arte, foi comparado o resultado com os melhores da competição em relação ao conjunto final do *dataset* Teste B. Como pode ser observado na Tabela VII, o modelo proposto, com o WRA de 89,90%, equivaleria-se ao terceiro colocado do desafio. Assim, evidenciando a eficiência da solução e sua competitividade frente ao estado da arte atual na área de ATR.

C. Estudo de ablação

Tabela VIII
Efeitos do uso de alinhamento e de dicionário nos resultados (Teste B)

Método	WRA (%)	Ganhos (%)	Ganhos ajustados[a] (%)
Nenhum	89,67	-	-
NW	89,73	+0,1	+1,1
D	89,73	+0,1	+1,1
NW e D	89,90	+0,2	+3,8

[a] Ajustados com relação a quantidade de palavras com 9 ou mais letras.

Para entender os ganhos obtidos com o uso dos métodos de alinhamento com algoritmo de Needleman-Wunsch e dicionário, foi realizado um estudo de ablação, onde cada componente foi estudado individualmente. Contudo, como esses métodos são apenas aplicáveis a palavras com 9 ou mais letras — 184 ao todo no conjunto do Teste B, foi aplicado um ajuste nos ganhos para melhor representar eles.

Conforme Tabela VIII, é possível observar que houve ganhos reais (+3,8%) ao serem aplicados os métodos propostos no contexto de termos longos.

D. Estudo dos erros

Tabela IX
Classificação dos erros do ensemble sobre o conjunto do Teste B

Tipo de erro	Quantidade	(%)
Erro de modelo	155	51,2
Erro de rotulagem	100	33,0
Ambiguidade	33	10,9
Não legível	15	5,0

Buscando compreender a causa de erros do modelo final, foi efetuado uma classificação dos erros dele sobre o conjunto do Teste B — 303 ao todo — em quatro grupos: (i) erro do modelo, (ii) erro de rotulagem, (iii) erro por ambiguidade,

Figura 12. Exemplos de predições problemáticas pelo *ensemble* de modelos.

(iv) e erro por falta de legibilidade, conforme a Tabela IX e como pode ser visualizado na Figura 12.

Assim, foi constatado que há espaço para melhorias em, pelo menos, 51,2% dos erros. Onde muitos casos constituem problemas de compreensão de letras cursivas e de estilizações radicais. Contudo, essa análise também evidencia que parte considerável dos erros (48,8%) não estão associados diretamente à capacidade de generalização do modelo, mas sim a limitações externas seja da base de dados ou seja da própria interpretabilidade visual das amostras.

VI. Conclusões

O presente trabalho apresentou uma abordagem baseada em ensemble para o ATR, combinando os modelos SVTRv2, PARSeq e MAERec com uma camada de decisão por votação ponderada, complementada por alinhamento via algoritmo de Needleman–Wunsch e correção léxica por dicionário para termos longos. A solução proposta alcançou uma WRA de 89,90% sobre o conjunto de Teste B do *dataset* WordArt-V1.5, o que a posicionaria entre as três melhores soluções da competição ICDAR 2024. Esses resultados evidenciam a eficácia da integração de diferentes arquiteturas para aumentar a robustez e a precisão no reconhecimento de textos com grande variabilidade estética e estrutural.

Como perspectivas futuras, propõe-se investigar o limite de contribuição de métodos e de estratégias em fusões de modelos. Portanto, focando em compreender o grau máximo de contribuição que esses métodos podem agregar aos resultados de um *ensemble* e se há possibilidade de aumentar suas eficiências para atingir um grau mais próximo do limite teórico de acerto.

Referências

[1] X. Xie, L. Deng, Z. Zhang, Z. Wang, and Y. Liu, "Icdar 2024 competition on artistic text recognition," in *Document Analysis and Recognition - ICDAR 2024*, E. H. Barney Smith, M. Liwicki, and L. Peng, Eds. Cham: Springer Nature Switzerland, 2024, pp. 301–314.

[2] X. Xie, L. Fu, Z. Zhang, Z. Wang, and X. Bai, "Toward understanding wordart: Corner-guided transformer for scene text recognition," in *Computer Vision – ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXVIII*. Berlin, Heidelberg: Springer-Verlag, 2022, p. 303–321. [Online]. Available: https://doi.org/10.1007/978-3-031-19815-1_18

[3] Y. Du, Z. Chen, Y. Su, C. Jia, and Y.-G. Jiang, "Instruction-guided scene text recognition," 2025. [Online]. Available: https://arxiv.org/abs/2401.17851

[4] Q. Jiang, J. Wang, D. Peng, C. Liu, and L. Jin, "Revisiting scene text recognition: A data perspective," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2023, pp. 20 543–20 554.

[5] B. Zhang, H. Xie, Y. Wang, J. Xu, and Y. Zhang, "Linguistic more: Taking a further step toward efficient and accurate scene text recognition," 2023. [Online]. Available: https://arxiv.org/abs/2305.05140

[6] P. Wang, C. Da, and C. Yao, "Multi-granularity prediction for scene text recognition," in *Computer Vision – ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXVIII*. Berlin, Heidelberg: Springer-Verlag, 2022, p. 339–355. [Online]. Available: https://doi.org/10.1007/978-3-031-19815-1_20

[7] A. Tarvainen and H. Valpola, "Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results," *Advances in neural information processing systems*, vol. 30, 2017.

[8] D. Bautista and R. Atienza, "Scene text recognition with permuted autoregressive sequence models," in *Computer Vision – ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXVIII*. Berlin, Heidelberg: Springer-Verlag, 2022, p. 178–196. [Online]. Available: https://doi.org/10.1007/978-3-031-19815-1_11

[9] T. Guan, W. Shen, X. Yang, Q. Feng, Z. Jiang, and X. Yang, "Self-supervised character-to-character distillation for text recognition," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 19 473–19 484.

[10] R. Atienza, "Vision transformer for fast and efficient scene text recognition," 2021. [Online]. Available: https://arxiv.org/abs/2105.08582

[11] Y. Du, Z. Chen, H. Xie, C. Jia, and Y.-G. Jiang, "Svtrv2: Ctc beats encoder-decoder models in scene text recognition," 2025. [Online]. Available: https://arxiv.org/abs/2411.15858

[12] A. Mishra, K. Alahari, and C. Jawahar, "Scene text recognition using higher order language priors," in *BMVC - British Machine Vision Conference*. Surrey, United Kingdom: BMVA, Sep. 2012. [Online]. Available: https://inria.hal.science/hal-00818183

[13] K. Wang, B. Babenko, and S. Belongie, "End-to-end scene text recognition," in *2011 International Conference on Computer Vision*, 2011, pp. 1457–1464.

[14] D. Karatzas, F. Shafait, S. Uchida, M. Iwamura, L. Bigorda, S. Mestre, J. Mas, D. Mota, J. Almazan, and L. {De Las Heras}, "Icdar 2013 robust reading competition," *Proceedings of the International Conference on Document Analysis and Recognition, ICDAR*, pp. 1484–1493, 2013, copyright: Copyright 2013 Elsevier B.V., All rights reserved.; 12th International Conference on Document Analysis and Recognition, ICDAR 2013 ; Conference date: 25-08-2013 Through 28-08-2013.

[15] D. Karatzas, L. Gomez-Bigorda, A. Nicolaou, S. Ghosh, A. Bagdanov, M. Iwamura, J. Matas, L. Neumann, V. R. Chandrasekhar, S. Lu, F. Shafait, S. Uchida, and E. Valveny, "Icdar 2015 competition on robust reading," in *2015 13th International Conference on Document Analysis and Recognition (ICDAR)*, 2015, pp. 1156–1160.

[16] T. Phan, P. Shivakumara, S. Tian, and C. L. Tan, "Recognizing text with perspective distortion in natural scenes," pp. 569–576, 12 2013.

[17] A. Willis, D. Morse, A. Dil, D. King, D. Roberts, and C. Lyal, "Improving search in scanned documents: Looking for ocr mismatches," 01 2009.

[18] S. Eger, "Sequence alignment with arbitrary steps and further generalizations, with applications to alignments in linguistics," *Information Sciences*, vol. 237, pp. 287–304, 2013, prediction, Control and Diagnosis using Advanced Neural Computations. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0020025513001485

[19] G. Kondrak, "N-gram similarity and distance," in *String Processing and Information Retrieval*, M. Consens and G. Navarro, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, pp. 115–126.

[20] K. Vertanen. Big english word lists. Acessado em: 03/11/2025. [Online]. Available: https://www.keithv.com/software/wlist/