Strategies to Improve Low-Resource Agglutinative Languages Morphological Inflection

Gulinigeer Abudouwaili[1,2], **Wayit Abliz**[1,2,*], **Kahaerjiang Abiderexiti**[1,2], **Aishan Wumaier**[1,2] and **Nian Yi** [1,2]
[1]School of Computer Science and Technology, Xinjiang University
[2]Laboratory of Multi-Language Information Technology, Xinjiang University
wayit@xju.edu.cn

Abstract

Morphological inflection is a crucial task in the field of morphology and is typically considered a sequence transduction task. In recent years, it has received substantial attention from researchers and made significant progress. Models have achieved impressive performance levels for both high- and low-resource languages. However, when the distribution of instances in the training dataset changes, or novel lemma or feature labels are predicted, the model's accuracy declines. In agglutinative languages, morphological inflection involves phonological phenomena while generating new words, which can alter the syllable patterns at the boundary between the lemma and the suffixes. This paper proposes four strategies for low-resource agglutinative languages to enhance the model's generalization ability. Firstly, a convolution module extracts syllable-like units from lemmas, allowing the model to learn syllable features. Secondly, the lemma and feature labels are represented separately in the input, and the position encoding of the feature labels is marked so that the model learns the order between suffixes and labels. Thirdly, the model recognizes the common substrings in lemmas through two special characters and copies them into words. Finally, combined with syllable features, we improve the data augmentation method. A series of experiments show that the proposed model in this paper is superior to other baseline models.

1 Introduction

Morphological inflection generates a word form given a lemma and target morpho-syntactic descriptions (MSDs) (Wiemerslage et al., 2023). For example, give the word 'dog' and the MSD labels 'N;PL', and to generate the word 'dogs'. Similar to morphological analysis (Toleu et al., 2022) and morphological segmentation (Batsuren et al., 2022a), morphological inflection is a fundamental task in natural language processing (NLP). It plays a crucial role in various downstream applications such as dependency parsing (Muñoz-Ortiz et al., 2022), machine translation (Tamchyna et al., 2017; Liu and Hulden, 2021; Xu and Carpuat, 2021), and others. Researchers have shown increasing interest in morphological inflection in recent years, and the research methods have evolved from traditional linguistic knowledge-based finite-state transducers (FSTs) to sequence-to-sequence frameworks (Xu and Carpuat, 2021). The construction of relevant datasets (Batsuren et al., 2022b) and the advancement of research approaches (Wu et al., 2021) have significantly reduced the difficulty of morphological inflection, but new challenges have also emerged.

The model achieves high accuracy when both the lemma and feature set are attested in the training set. However, when lemma or feature sets are unattested in training, or in cases similar to the "wug test" (Liu and Hulden, 2022), the model's accuracy begins to decline (Kodner et al., 2022), even in high-resource languages. Because the dataset of low-resource languages is too small, training neural network models can result in label bias, where the model tends to output characters commonly seen in the training set (Anastasopoulos and Neubig, 2019). It is very effective to augment training data in low-resource with a data hallucination approach (Liu and Hulden, 2022). Anastasopoulos and Neubig (2019) proposed a data augmentation based on characters, while Liu and Hulden (2022) argue that data hallucination based on strings or syllables approach (such as 2-gram, 3-gram, 4-gram, etc.) is more effective than character-based. This is because character-based hallucination breaks the original syllabic structure of words. Additionally, in sequence-to-sequence models (seq2seq), the input usually includes both the lemma and MSDs. When the lemma and MSDs are lengthy, it cannot be guaranteed that each label will impact every character. In the agglutinative language morpholog-

*:Corresponding author

Proceedings of the 27th Conference on Computational Natural Language Learning (CoNLL), pages 508–520
December 6–7, 2023.

ical task, MSD affects the beginning and ending of the word, with very few influences on the internal structure of the word, as shown in the following example in Kyrgyz:

кошуу   кошып жатасың

Figure 1: An example in Kyrgyz

In the example, the lemma is on the left, and the word is on the right. We divide the word into two parts: red is the stem, and blue is the suffix, and the stem is a part of the lemma. During model predictions, errors can occur not only in the suffix but also in the stem. Furthermore, when the lemma is connected to the suffixes, there may be character substitution, insertion, and deletion. There are too many uncertainties regarding which characters undergo each type of transformation (Kodner et al., 2022). These uncertainties can also change syllable categories at the connection points. All these problems make low-resource agglutinative language morphological inflection more challenging.

Therefore, based on the above problems and considering the characteristics of agglutinative language syllables, this paper proposed four strategies to address them. The first strategy aims to reduce the impact of agglutinative language phonetic variations by incorporating a convolution module in the model's encoder. This module extracts syllabic features (like n-grams). The second strategy, inspired by the work of Yang et al. (2022), adds reversed token embeddings and positional encodings in the encoder's input. Additionally, label positions are marked, enabling the model to learn the correspondence between suffixes and labels and the impact of labels on each character. The third strategy aims to alleviate errors in the stem. In the encoder, special characters are added to the beginning and ending of the lemma's stem. In the decoder, each character of the lemma is marked to indicate whether it should be copied. The fourth strategy is to avoid breaking the syllable categories of lemmas and words during data augmentation. Letter type (sound: vowel or consonant) is determined when randomly sampling. If the letter being replaced is a vowel, it is substituted with another vowel in the language; a consonant is replaced with another consonant. We evaluate our model on five low-resource agglutinative languages, Kazakh, Kirgiz, Tatar, Uyghur, and Uzbek, in Unimorph. The experiments show that the performance of the model proposed in this paper is superior to that of other comparable models. The baseline model (baseline-neural model) with data hallucination and three strategies have improved the overall accuracy of the model by 9.54% and 4.17%, respectively. In summary, our main contributions are as follows:

- Improved the existing data hallucination approach to generate fake data that adheres more closely to the language rules.
- Proposed three strategies to improve the model's accuracy by addressing issues in morphological inflection and considering the characteristics of agglutinative languages. Firstly, incorporating reversed token embeddings and positional encoding at the input, representing lemma and MSDs separately. Secondly, a convolution module for learning syllable features in agglutinative languages is added to the encoder. Finally, two types of labels are employed to enable the model to identify common substrings and learn to copy them.
- The proposed strategies were validated through experiments on Kazakh, Kyrgyz, Tatar, Uyghur, and Uzbek languages in the UniMorph dataset, and the results demonstrated the effectiveness of the proposed strategies.

2 Related Work

In recent years, the development of morphological inflection has significantly been promoted by the Sigmorphon shared tasks (Kodner et al., 2022; Vylomova et al., 2020; Pimentel et al., 2021). Research on morphological inflection mainly focuses on rule-based (such as FST) (Xu and Carpuat, 2021; Merzhevich et al., 2022), statistical (Liu and Mao, 2016), and neural network-based models (Wu et al., 2021; Liu and Hulden, 2020; Singer and Kann, 2020). Additionally, data augmentation (Anastasopoulos and Neubig, 2019; Silfverberg et al., 2017) can also improve the performance of models in low-resource languages. Seq2seq models, such as RNN+attention (Wiemerslage et al., 2018) or Transformer (Yang et al., 2022; Merzhevich et al., 2022; Elsner and Court, 2022), have become popular framework for morphological inflection in recent years. The lemma and tags are usually input together in this framework, and the model generates the inflected word. For example, given the input 'dog+N+PL', the output should be 'dogs' (Wu et al., 2021). Based on the Transformer, Wu et al. (2021) modified position encoding of MSDs in the input sequence to 0 and added embedding type to distinguish between characters and features. This modification makes the model more suitable for morphological inflection. Transformer can achieve high accuracy in high-resource or simple conditions where both the lemma and MSD have attested in the training dataset. However, training high-accuracy models in low-resource or complex situations where the lemma or tags are unattested in the training dataset is challenging. Through experimental analysis (Liu and Hulden, 2022), it has been found that for some languages, there is a portion of the generated word where the lemma and feature tags correspond to the common strings. Therefore, improving the model's ability to copy characters can enhance its performance. Singer and Kann (2020) proposed a pointer generator Transformer, which uses a copying mechanism to generate a character probability distribution. This model achieved a 4.46% improvement over the vanilla Transformer in low-resource languages. Wehrli et al. (2022) proposed a character-level neural transducer that operates over traditional edit actions based on their previous work (Makarov and Clematide, 2020). They optimized the training procedure using mini-batches and only relied on the teacher-forcing approach, i.e., using gold labels rather than what was predicted during the training phase. Morphosyntactic features were treated individually, and their embeddings were summed. Anastasopoulos and Neubig (2019) proposed a two-step attention decoding structure and augmented the dataset through data hallucination. Firstly, they identified the "stem" (the common part when comparing lemma and word, where there is one or several stems) based on the lemma-word pairs in the dataset. Then, they randomly replaced the string in the stem, except for the first and last strings. Yang et al. (2022) suggested that in morphological inflection, only forward distances are usually encoded while ignoring backward distances. Therefore, they added reverse positional encoding based on the char-Transformer model. Firstly, they trained the model using standard backpropagation and teacher forcing based on the data augmentation proposed by Anastasopoulos and Neubig (2019), saving the best model on the validation set. Then, they further trained the model using student forcing. Finally, this model achieved an accuracy improvement of 9.6% and 8.6% compared to the baseline model in low-resource and high-resource scenarios. Merzhevich et al. (2022) proposed two models in the Sigmorphon 2022 shared task: a neural network-based model and an FST-based model. The FST model outperformed the neural network-based model in specific languages. This indicates that for endangered languages or low-resource scenarios, data-driven methods are still immature and rely on linguistic rules. Although FST models achieve higher accuracy in specific languages, collecting or annotating linguistic rules is costly and time-consuming. Thus, building a high-performance model using existing data resources is crucial. Therefore, this paper focuses on five low-resource agglutinative languages. Based on the baseline model - Transformer, four strategies are proposed to improve the model's accuracy and robustness by incorporating morphological features of agglutinative languages.

3 Approaches

In this section, we describe our strategies for the inflection task.

3.1 Feature extraction

In agglutinative languages, when generating a new word, the connection between lemma and suffixes can result in character additions, deletions, and substitutions due to the influence of the pronunciation of surrounding characters, which is called phonological phenomena. This phenomena change the syllable structure of lemma. In this paper, we hypothesize that syllable features are important in agglutinative morphological inflection, in addition to character features and contextual features. The multi-head attention mechanism in Transformer extracts character and contextual features, but it is not sure whether syllable-like features are also extracted, such as n-gram. Therefore, this paper extract character contextual features through a convolution module to reduce manual labeling, simulating the process of extracting n-gram or syllable features. Specifically, we introduce convolutional blocks into the encoder (Vaswani et al., 2017) of the Transformer to extract syllable features, as shown in Figure 2.

Given a sequence $W = \{c_1, c_2, \ldots, c_n\}$, c_i embedding is represented as $x_i \in \mathbb{R}^{d_{model}}$, where

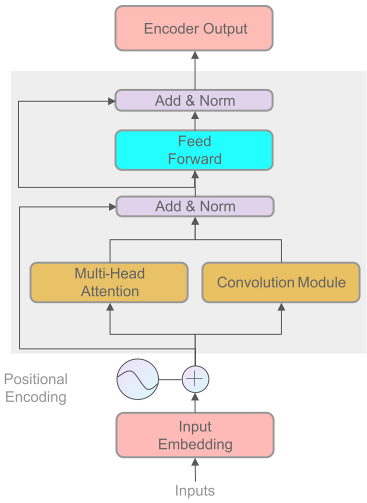

Figure 2: The Transformer encoder

d_{model} represents the dimension of the vectors. The word embeddings are separately fed into the multi-head self-attention module and the convolution module in the encoder. When inputted into the multi-head self-attention module, the input vectors are linearly transformed to obtain Q, K, V vectors of the same dimension as X. Then, the attention scores for the ith head are computed as shown in formulas 1-2:

$$\mathbf{Attention}\,(Q,K,V)_i = \mathbf{softmax}(\frac{Q_i K_i^T}{\sqrt{d_k}})V_i \quad i = 1,...,n \tag{1}$$

$$\mathbf{MultiHead}(Q,K,V) = \mathbf{Concat}(\mathbf{Attention_1},...,\mathbf{Attention_n})W^O \tag{2}$$

where $W^O \in \mathbb{R}^{hd \times d_{model}}$, $d = d_{model} = 256$, the number of heads h=4. When inputted into the convolution module, this paper utilizes depthwise separable convolution to reduce the number of parameters in the model. It combines depthwise convolution and pointwise convolution, as shown in formulas 3-5:

$$P = \sigma(W_a X^T) \tag{3}$$

where W_a is pointwise convolution, $W_a \in \mathbb{R}^{d_{model} \times d_{model}}$. $\sigma(\cdot)$ indicates the GLU activation function.

$$D = (W_c(\mathbf{Concat}(W_b^1 P, \ldots, W_b^i P))^T + b) \quad i = 1,...,5 \tag{4}$$

where W_b^i is depthwise convolution, $W_b^i \in \mathbb{R}^{d_{model} \times d_{model}}$, W_c represents a linear layer used to reduce data dimension, $W_c \in \mathbb{R}^{md_{model} \times d_{model}}$, m represents how many convolutions are used, and b is the model parameter.

$$\mathbf{ConvFeat} = W_d \sigma(D^T) \tag{5}$$

where W_d is pointwise convolution, $W_d \in \mathbb{R}^{d_{model} \times d_{model}}$. $\sigma(\cdot)$ indicates Swish activation function. Therefore, the final feature output is shown in formula 6:

$$\mathbf{FinalFeat} = \mathbf{MultiHead} + \mathbf{ConvFeat^T} \tag{6}$$

3.2 Model input

In morphological inflection, the MSDs are added to the lemma and input into the model together. Therefore, the model treats MSDs as special characters. However, we want the MSDs to constrain the lemma rather than become part of the lemma. Thus, Wu et al. (2021) set the positional encoding of MSDs to 0 and only start counting the positions for characters. They add a special token to indicate whether a symbol is a word character or an MSD. Additionally, Yang et al. (2022) argue that in morphological inflection, it is important to encode the distance from the beginning of the input string and encode the distance to the end of the string. So, they proposed reverse positional encoding, where the final positional encoding is obtained by concatenating forward and reverse positional encodings.

Both of the above approaches do not learn the positional encoding for MSD. However, we believe that MSDs correspond to suffixes. As suffixes have a specific order, MSDs also have an order. Therefore, this paper handles lemma and MSD embeddings separately, without including any type of embeddings. The model input is shown in Figure 3. Given a sequence of length n (excluding MSD), where x_i represents the word embedding of the i-th character, f_i represents the forward sinusoidal positional encoding of the i-th character. Thus, the token embedding and positional encoding of the i-th character are formulated as shown in Equation 7-8, and the final embedding representation is shown in Equation 9.

$$C_i = \mathbf{concat}(x_i, x_{n-i+1}) \tag{7}$$

$$P_i = \mathbf{concat}(f_i, f_{n-i+1}) \tag{8}$$

$$E_i = C_i + P_i \tag{9}$$

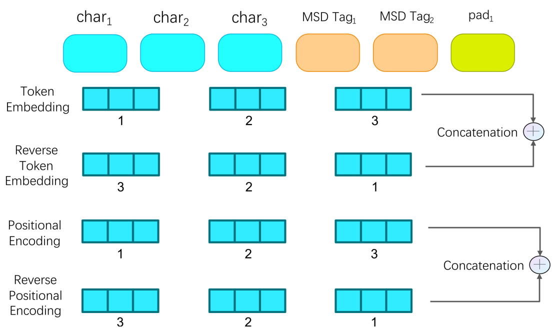

Figure 3: The model's input

3.3 Finding Common Substrings

We divide words into two parts: stem and suffixes. In neural network-based morphological inflection, errors can occur in the suffixes and the stem. Therefore, improving the model's ability to copy the stem accurately can enhance the overall accuracy. In this paper, the stem in the word is identified by comparing it with the lemma, and the "$" symbol is added to the beginning and ending of the stem to indicate that it is the same part. Additionally, an extra character token is introduced in the input of the transformer decoder to indicate whether the character is a part of stem, as shown in Figure 4. The model is trained using teacher-forcing, and during testing, a greedy search with a width of 5 is applied.

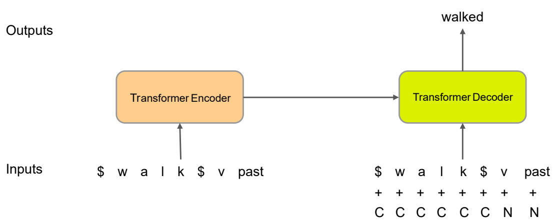

Figure 4: The Encoder-Decoder input

3.4 Data hallucination

In agglutinative language morphological inflection, we found that the main focus is on suffixes. In other words, suffixes are added, deleted, and substituted. In the data augmentation approach proposed by Anastasopoulos and Neubig (2019), the stem containing at least three or more characters is selected, and random replacement is performed on the middle characters of the stem (excluding the first and last characters) while maintaining the overall length of the stem. The data augmentation

Algorithm 1 Data hallucination (DH)

Input: labeled data
Output: fake data

D = labeled data
for each $i \in [0, len(D)]$ **do**
  line= D[i]
  lemma, word, label = getparts(line)
  comstr= getcommon(lemma, word)
  achar=getrandom(0: len(comstr)-1)
  if achar is Vowel **then**
    new_char=getrandom(VowelsList)
  else
    new_char=getrandom(ConsonantsList)
  end if
  new_comstr=replace(comstr,achar,newchar,1)
  Add_Hallucinate_Dictionary(comstr,new comstr)
  Add_Word_Dictionary(word)
end for
for each $i \in [0, len(D)]$ **do**
  line= D[i]
  lemma, word, label = getparts(line)
  comstr= getcommon(lemma,word)
  new_comstr= getFromHallu_Dict(comstr)
  new_lemma=replace(lemma,comstr, new_comstr,1)
  new_word= replace(word,comstr,_comstr,1)
  while new_word in Word_Dictionary **do**
    new_comstr =Regenerate_new_comstr()
    new_word =replace (word, comstr, new_comstr, 1)
  end while
  Add_Word_Dictionary (new_word)
  new_line= makeNewDataLine(new_lemma, new_word, label)
  Add_FakeData(new_line)
end for
return FakeData

approach proposed in this paper, language features are incorporated to improve the rules of random replacement. During each sampling, only one letter is replaced, and the category of the original letter (consonant or vowel) is determined before replacement. A randomly sampled character of the same type is then used for replacement. It is worth noting that there are cases in the dataset where two characters together represent a single sound, such as "ch", "sh" and so on. When encountering the replacement of such characters, this paper combines and replaces them with another character of the same type, which may alter the length of the word. In this paper, 10,000 fake examples were generated for each language through data augmentation. The pseudocode for the data augmentation is shown in Algorithm 1.

4 Experiments

4.1 Data and evaluation

This paper defines training data with fewer than 7000 instances as low-resource. The experimental data for Kyrgyz (kir), Tatar (tat), Uyghur (uig), and Uzbek (uzb) languages are sourced from UniMorph (Batsuren et al., 2022b), while the Kazakh (kaz) dataset is obtained from the Sigmorphon2022 shared task. The dataset consists of three columns: lemma, word form, and label. The statistics of the dataset are shown in Table 1:

Lang.	Train	Test	Development
Kaz	7000	1994	998
Kir	3879	1109	556
Tat	5481	1567	784
Uig	5675	1668	835
Uzb	7000	1988	998

Table 1: Dataset statistics.

To test the model's morphological inflection ability for lemmas and MSDs that have been unattested in the training set, we ensured that a portion of the lemmas and morphological features were unseen in the training and test sets during data partitioning. Following (Kodner et al., 2022), the overlap types for each example in the validation and test sets can be categorized into the following four types. The statistical information on different overlap types in the validation and test sets are shown in Table 2:

Both overlap: Both the lemma and feature set of a training pair are attested in the training set (but not together in the same triple)

Lemma overlap: A test pair's lemma is attested in training, but its feature set is novel

Feature overlap: A test pair's feature set is attested in training, but its lemma is novel

Neither overlap: A test pair is entirely unattested in training. Both its lemma and features are novel.

This paper evaluates the model performance using accuracy (ACC) and calculates the accuracy for different overlap types using the evaluation script [1] from SIGMORPHON2022 shared task 0.

4.2 Baseline models and hyperparameters

This paper selects the rule-based (baseline-nonneural), neural (baseline-neural) CLUZH models from SIGMORPHON2022 shared task 0 and a data hallucination approach. The rule-based model is used for shared tasks from 2020, while the neural model is based on the vanilla transformer proposed by Vaswani et al. (2017). The CLUZH is a system submitted by the CLUZH team to SIGMORPHON2022 shared task 0, a character-level neural transducer (Wehrli et al., 2022). The proposed improvements in this paper are modifications made to the vanilla transformer. In addition to these two baseline models, we incorporate the data augmentation method proposed by Anastasopoulos and Neubig (2019) in the neural-based experiments.

We train our models with four layers in the encoder and decoder, each containing four attention heads. The embedding size is 256, and the hidden layer size is 1024. We use the Adam optimizer with an initial learning rate of 0.001. In the baseline comparison experiments, the batch size is 256; in the data Hallucination comparison experiments, the batch size is 64.

4.3 Experimental results

In the paper, we conducted two sets of comparative experiments to demonstrate the effectiveness of the proposed strategies. In the first set of experiments, we incorporated the improvements proposed in Sections 3.1, 3.2, and 3.3 into the vanilla Transformer and compared the results to the baseline model. The experimental results are shown in Table 3. In the second set of experiments, we compared the data augmentation method proposed by Anastasopoulos and Neubig (2019) with the data augmentation method proposed in this paper. The experimental results are presented in Table 5. A detailed description of the comparative experiment is provided in Appendix A.

The experimental results in Table 3 show that three strategies proposed in this paper outperform the baseline model on test set. Compared to the baseline-nonneural model, the overall accuracy is

[1] https://github.com/sigmorphon/2022InflectionST/blob/main/evaluation/evaluate.py

Lang.	Development					Test				
	Total	Both	Lemma	Feature	Neither	Total	Both	Lemma	Feature	Neither
Kaz	998	412	563	13	10	1994	966	992	28	8
Kir	556	138	237	160	21	1109	303	483	272	51
Tat	784	776	0	8	0	1567	1551	0	16	0
Uig	835	206	312	274	43	1668	427	601	562	78
Uzb	998	793	79	121	5	1988	1540	159	281	8

Table 2: Statistics of four kinds of overlaps

Lang	Baseline-nonneural		Baseline-neural		CLUZH		Our model	
	dev	test	dev	test	dev	test	dev	test
Kaz	37.58	42.88	68.04	65.55	55.41	55.42	**69.64**	**68.81**
Kir	46.40	44.91	66.01	71.87	**78.06**	76.47	74.10	**81.24**
Tat	76.02	77.15	95.41	95.72	**97.07**	97.00	96.43	**97.26**
Uig	50.30	51.50	77.25	76.80	77.61	77.28	**83.35**	**83.75**
Uzb	89.17	88.03	91.68	91.05	**96.99**	**96.53**	92.18	92.96
Total	60.86	62.11	80.41	80.41	80.65	80.24	**83.41**	**84.58**

Table 3: Comparison experimental results of baseline models

Partition	CLUZH	Flexica	OSU	TüM Main	UBC	Our model
overall	58.38	34.20	49.20	53.61	65.75	**68.81**
both	96.17	67.70	**98.76**	89.96	97.52	97.72
lemma	20.87	0.81	0.00	17.44	34.38	**40.22**
features	**100.00**	71.43	96.43	96.43	92.86	96.43
neither	0.00	0.00	0.00	0.00	25.00	25.00

Table 4: Experimental results of Kazakh in Sigmarphon 2022 shared task

improved by 22.55% and 22.47%, while compared to the baseline-neural model, the improvement is 3.00% and 4.17%, respectively. Compared with the CLUZH, it has increased by 2.76% and 4.34%, respectively. There have been significant improvements in test sets for all languages except Uzbek. This indicates that the proposed methods are effective for low-resource agglutinative languages. It's worth noting that although the rule-based approach has the lowest accuracy, it achieves an accuracy of 88.03% on the Uzbek language test set, while the neural model only reaches 91.05% and 92.96%. The improvement is not as significant compared to other languages. Similarly, there are interesting findings in the case of Kazakh. The neural network improves accuracy compared to the rule-based method, but the improvement is not significant. Through analysis, it was found that this may be related to three factors in the dataset: 1) the distribution of lemmas and features, 2) the frequency of phonological phenomena occurrences.

In addition to the comparative experiments with the baseline model mentioned above, this paper also compared the experimental results of systems such as CLUZH, Flexica, OSU, TüM Main, and UBC on Kazakh in the Sigmorphon 2022 shared task (Kodner et al., 2022). The experimental results are presented in Table 4.

From the experimental results on the Kazakh dataset in Table 4, it is observed that the model achieves higher accuracy when both the lemma and the feature are attested in the training set or only the feature is attested in the training set. On the contrary, the model's accuracy is relatively low when only the lemma is attested, or neither of them is attested in the training set. This is one of the reasons for the lower accuracy in Kazakh. Therefore, we consider that in some languages, the phonological phenomena that occur in word differ with different sets of labels, and important morphological variations are rarely learned through overlaped lemmas. This leads to the lower accuracy of the model in the case of lemma overlap. The data hallucination seems to improve the model's robustness by increasing the variety of lemmas. But in reality, it enables the model to learn the relationship between the labels and suffixes through the overlap of MSD. This phenomenon can also be observed in the experimental results in Appendix A.2, where there is an improvement in accuracy on lemma overlap for languages other than Tatar.

Lang	Baseline-Neural		Baseline-Neural+hall		Baseline-Neural+our hall	
	dev	test	dev	test	dev	test
Kaz	62.12	61.89	63.83	61.69	**68.44**	**66.40**
Kir	64.75	70.51	**84.89**	**87.92**	83.99	87.29
Tat	92.22	92.92	93.24	92.79	**94.90**	**95.28**
Uig	74.61	72.00	**94.01**	**93.05**	91.14	92.63
Uzb	89.87	87.68	**95.69**	**95.62**	94.38	94.57
Total	77.27	77.06	85.83	85.42	**86.24**	**86.60**

Table 5: The results of comparison experimental based on hallucinations

Therefore, to further improve the model's accuracy, this paper investigates the technique of data hallucination. From Table 5, it is observed that data hallucination has a significant impact on

Model	Overall		Both		lemma		features		neither	
	Dev	Test	Dev	Test	Dev	Test	Dev	Test	Dev	Test
Baseline	80.41	80.41	96.56	96.93	66.84	62.55	48.79	52.03	39.74	37.24
+Feature extraction	81.94	82.56	96.65	96.99	68.18	66.58	55.73	58.50	47.44	44.83
+Model input	82.35	82.75	**96.77**	**97.41**	67.34	63.89	59.20	63.07	52.56	46.90
+Common substrings	80.74	81.46	96.00	96.37	67.84	65.10	51.22	56.43	41.03	41.38
+++	**83.41**	**84.58**	96.22	96.93	**70.11**	**67.61**	**62.85**	**70.15**	**56.41**	**53.79**

Table 6: The experimental results of ablation study. '+Feature extraction' means adding feature extraction module to the baseline.'+Model input' means adding model input module to the baseline. '+Common Substrings' means adding finding common substrings module to the baseline."+++" means adding all three modules to the baseline.

all languages. Compared to the baseline-neural model, the proposed approach in this paper shows improvements of 8.97% and 9.54% on the validation set and test set, respectively. Compared with the method proposed by Anastasopoulos and Neubig (2019) (baseline-neural+hall), it has increased by 0.41% and 1.18%, respectively. On Kyrgyz, Uyghur, and Uzbek, comparable to the baseline-neural+hall model, there is not much difference between the performance. Through analysis of the experiments in Appendix A.2, it is found that baseline-neural+our hall model slightly outperforms baseline-neural+hall in both overlap and lemma overlap, but underperforms baseline-neural+hall in feature overlap and neither overlap.

4.4 Experimental analysis

To further validate the impact of the three strategies on model performance, this paper conducted a set of ablation experiments, and the results are shown in Table 6. From the overall results, it can be seen that each strategy contributes to improving the model's accuracy. When the baseline model is added with the feature extraction module, the accuracy is improved by 1.53% and 2.15% on the validation set and test set, respectively. Adding the model input module improves the accuracy by 1.94% and 2.34%. Incorporating the common substring enhances the accuracy by 0.33% and 1.05%. Finally, when all three strategies are combined, the accuracy is improved by 3.00% and 4.17%. In simple scenarios where both lemma and features are attested, the model achieves an accuracy of over 96.00%. However, the model's accuracy is relatively low in complex scenarios where only one or neither of them are attested. The three strategies proposed in this paper show improvements in lemma overlap, feature overlap, and neither overlap compared to the baseline model. The accuracy on the validation set and test set is increased by 3.27%, 5.06%, 14.06%, 18.12%, 16.67%, and 16.55%, respectively.

Through error analysis, it was discovered that phonological phenomena in agglutinative languages are also a major source of errors. When the lemma is connected to suffixes, there are many uncertainties, such as: 1) which phonological phenomena will occur; 2) which character will change; 3) which character should be generated. Therefore, errors may arise in insertion, deletion, and substitution operations. In addition to errors caused by phonological phenomena, this paper also found that when the lemma contains repeated characters (regardless of whether they are consecutive), the generated word often omits some characters. This phenomenon exists in the baseline model and the proposed method, as demonstrated by the examples in Kazakh and Uyghur languages below. Positional encoding is considered a possible factor contributing to such errors.

әрекеттес	N;GEN;SG		külümsirimek	V;PROG;SG;1;PST	
Baseline	әрекетестің	×	Baseline	kölüriwatattim	×
Our model	әрекеттестің	√	Our model	külüriwatattim	×
Gold	әрекеттестің		Gold	külümsirewatattim	

Figure 5: Error analysis

5 Conclusion

This paper addressed the challenges of low-resource agglutinative language morphological inflection and proposed four strategies. Firstly, to tackle the main issue of limited training data in low-resource settings, a data hallucination approach that incorporates syllable features is introduced. A syllable feature extraction module is added to the encoder, enabling the model to learn the context and transformation of characters through syllables. Secondly, the lemma and MSDs are separately encoded at the encoder's input. Reversed token embeddings and positional encoding are also incorporated to establish correlations between labels and generated suffixes. Lastly, the model's ability to copy common parts of lemmas is enhanced by marking common substrings at the encoder-decoder. Experimental results demonstrate that the proposed strategies effectively alleviate the issues caused by data scarcity or agglutinative language features, and all strategies lead to improvements in model accuracy, outperforming other comparative models. This paper initially explores the agglutinative language morphological inflection model in low-resource scenarios. In future research, we will continue optimizing the model's ability to learn positional encoding and extract syllable features, further enhancing its generalization capabilities.

Limitations

Although the strategies proposed in this paper have achieved good experimental results in different types of overlap, the accuracy is not very high for overlaps other than "both overlap," especially in "neither overlap." Of course, the task is also challenging. Through analyzing the experimental results, it is found that positional encoding is crucial in morphological inflection tasks. When the same characters appear in the lemma, there are still cases where other characters are omitted in the word. This paper has conducted further research based on previous studies, there is still a lot of room for improvement.

Acknowledgements

We gratefully thank the anonymous reviewers for their insightful comments. This work is supported by the National Natural Science Foundation of China (grant numbers 62166044) and Doctoral Research Initiation Fund of Xinjiang University (grant numbers 620323008).

References

Antonios Anastasopoulos and Graham Neubig. 2019. Pushing the limits of low-resource morphological inflection. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 984–996, Hong Kong, China. Association for Computational Linguistics.

Khuyagbaatar Batsuren, Gábor Bella, Aryaman Arora, Viktor Martinovic, Kyle Gorman, Zdeněk Žabokrtský, Amarsanaa Ganbold, Šárka Dohnalová, Magda Ševčíková, Kateřina Pelegrinová, Fausto Giunchiglia, Ryan Cotterell, and Ekaterina Vylomova. 2022a. The SIGMORPHON 2022 shared task on morpheme segmentation. In *Proceedings of the 19th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 103–116, Seattle, Washington. Association for Computational Linguistics.

Khuyagbaatar Batsuren, Omer Goldman, Salam Khalifa, Nizar Habash, Witold Kieraś, Gábor Bella, Brian Leonard, Garrett Nicolai, Kyle Gorman, Yustinus Ghanggo Ate, Maria Ryskina, Sabrina Mielke, Elena Budianskaya, Charbel El-Khaissi, Tiago Pimentel, Michael Gasser, William Abbott Lane, Mohit Raj, Matt Coler, Jaime Rafael Montoya Samame, Delio Siticonatzi Camaiteri, Esaú Zumaeta Rojas, Didier López Francis, Arturo Oncevay, Juan López Bautista, Gema Celeste Silva Villegas, Lucas Torroba Hennigen, Adam Ek, David Guriel, Peter Dirix, Jean-Philippe Bernardy, Andrey Scherbakov, Aziyana Bayyr-ool, Antonios Anastasopoulos, Roberto Zariquiey, Karina Sheifer, Sofya Ganieva, Hilaria Cruz, Ritván Karahóǧa, Stella Markantonatou, George Pavlidis, Matvey Plugaryov, Elena Klyachko, Ali Salehi, Candy Angulo, Jatayu Baxi, Andrew Krizhanovsky, Natalia Krizhanovskaya, Elizabeth Salesky, Clara Vania, Sardana Ivanova, Jennifer White, Rowan Hall Maudslay, Josef Valvoda, Ran Zmigrod, Paula Czarnowska, Irene Nikkarinen, Aelita Salchak, Brijesh Bhatt, Christopher Straughn, Zoey Liu, Jonathan North Washington, Yuval Pinter, Duygu Ataman, Marcin Wolinski, Totok Suhardijanto, Anna Yablonskaya, Niklas Stoehr, Hossep Dolatian, Zahroh Nuriah, Shyam Ratan, Francis M. Tyers, Edoardo M. Ponti, Grant Aiton, Aryaman Arora, Richard J. Hatcher, Ritesh Kumar, Jeremiah Young, Daria Rodionova, Anastasia Yemelina, Taras Andrushko, Igor Marchenko, Polina Mashkovtseva, Alexandra Serova, Emily Prud'hommeaux, Maria Nepomniashchaya, Fausto Giunchiglia, Eleanor Chodroff, Mans Hulden, Miikka Silfverberg, Arya D. McCarthy, David Yarowsky, Ryan Cotterell, Reut Tsarfaty, and Ekaterina Vylomova. 2022b. UniMorph 4.0: Universal Morphology. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 840–855, Marseille, France. European Language Resources Association.

Micha Elsner and Sara Court. 2022. OSU at SigMorphon 2022: Analogical inflection with rule features. In *Proceedings of the 19th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 220–225, Seattle, Washington. Association for Computational Linguistics.

Jordan Kodner, Salam Khalifa, Khuyagbaatar Batsuren, Hossep Dolatian, Ryan Cotterell, Faruk Akkus, Antonios Anastasopoulos, Taras Andrushko, Aryaman Arora, Nona Atanalov, Gábor Bella, Elena Budianskaya, Yustinus Ghanggo Ate, Omer Goldman, David Guriel, Simon Guriel, Silvia Guriel-Agiashvili, Witold Kieraś, Andrew Krizhanovsky, Natalia Krizhanovsky, Igor Marchenko, Magdalena Markowska, Polina Mashkovtseva, Maria Nepomniashchaya, Daria Rodionova, Karina Scheifer, Alexandra Sorova, Anastasia Yemelina, Jeremiah Young, and Ekaterina Vylomova. 2022. SIGMORPHON–UniMorph 2022 shared task 0: Generalization and typologically diverse morphological inflection. In *Proceedings of the 19th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 176–203, Seattle, Washington. Association for Computational Linguistics.

Ling Liu and Mans Hulden. 2020. Leveraging principal parts for morphological inflection. In *Proceedings of the 17th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 153–161, Online. Association for Computational Linguistics.

Ling Liu and Mans Hulden. 2021. Backtranslation in neural morphological inflection. In *Proceedings of the Second Workshop on Insights from Negative Results in NLP*, pages 81–88, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Ling Liu and Mans Hulden. 2022. Can a transformer pass the wug test? tuning copying bias in neural morphological inflection models. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 739–749, Dublin, Ireland. Association for Computational Linguistics.

Ling Liu and Lingshuang Jack Mao. 2016. Morphological reinflection with conditional random fields and unsupervised features. In *Proceedings of the 14th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 36–40, Berlin, Germany. Association for Computational Linguistics.

Peter Makarov and Simon Clematide. 2020. CLUZH at SIGMORPHON 2020 shared task on multilingual grapheme-to-phoneme conversion. In *Proceedings of the 17th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 171–176, Online. Association for Computational Linguistics.

Tatiana Merzhevich, Nkonye Gbadegoye, Leander Girrbach, Jingwen Li, and Ryan Soh-Eun Shim. 2022. SIGMORPHON 2022 task 0 submission description: Modelling morphological inflection with data-driven and rule-based approaches. In *Proceedings of the 19th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 204–211, Seattle, Washington. Association for Computational Linguistics.

Alberto Muñoz-Ortiz, Carlos Gómez-Rodríguez, and David Vilares. 2022. Cross-lingual inflection as a data augmentation method for parsing. In *Proceedings of the Third Workshop on Insights from Negative Results in NLP*, pages 54–61, Dublin, Ireland. Association for Computational Linguistics.

Tiago Pimentel, Maria Ryskina, Sabrina J. Mielke, Shijie Wu, Eleanor Chodroff, Brian Leonard, Garrett Nicolai, Yustinus Ghanggo Ate, Salam Khalifa, Nizar Habash, Charbel El-Khaissi, Omer Goldman, Michael Gasser, William Lane, Matt Coler, Arturo Oncevay, Jaime Rafael Montoya Samame, Gema Celeste Silva Villegas, Adam Ek, Jean-Philippe Bernardy, Andrey Shcherbakov, Aziyana Bayyr-ool, Karina Sheifer, Sofya Ganieva, Matvey Plugaryov, Elena Klyachko, Ali Salehi, Andrew Krizhanovsky, Natalia Krizhanovsky, Clara Vania, Sardana Ivanova, Aelita Salchak, Christopher Straughn, Zoey Liu, Jonathan North Washington, Duygu Ataman, Witold Kieraś, Marcin Woliński, Totok Suhardijanto, Niklas Stoehr, Zahroh Nuriah, Shyam Ratan, Francis M. Tyers, Edoardo M. Ponti, Grant Aiton, Richard J. Hatcher, Emily Prud'hommeaux, Ritesh Kumar, Mans Hulden, Botond Barta, Dorina Lakatos, Gábor Szolnok, Judit Ács, Mohit Raj, David Yarowsky, Ryan Cotterell, Ben Ambridge, and Ekaterina Vylomova. 2021. SIGMORPHON 2021 shared task on morphological reinflection: Generalization across languages. In *Proceedings of the 18th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 229–259, Online. Association for Computational Linguistics.

Miikka Silfverberg, Adam Wiemerslage, Ling Liu, and Lingshuang Jack Mao. 2017. Data augmentation for morphological reinflection. In *Proceedings of the CoNLL SIGMORPHON 2017 Shared Task: Universal Morphological Reinflection*, pages 90–99, Vancouver. Association for Computational Linguistics.

Assaf Singer and Katharina Kann. 2020. The NYU-CUBoulder systems for SIGMORPHON 2020 task 0 and task 2. In *Proceedings of the 17th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 90–98, Online. Association for Computational Linguistics.

Aleš Tamchyna, Marion Weller-Di Marco, and Alexander Fraser. 2017. Modeling target-side inflection in neural machine translation. In *Proceedings of the Second Conference on Machine Translation*, pages 32–42, Copenhagen, Denmark. Association for Computational Linguistics.

Alymzhan Toleu, Gulmira Tolegen, and Rustam Mussabayev. 2022. Language-independent approach for morphological disambiguation. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 5288–5297, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention Is All You Need. *arXiv e-prints*, page arXiv:1706.03762.

Ekaterina Vylomova, Jennifer White, Elizabeth Salesky, Sabrina J. Mielke, Shijie Wu, Edoardo Maria Ponti, Rowan Hall Maudslay, Ran Zmigrod, Josef Valvoda, Svetlana Toldova, Francis Tyers, Elena

Klyachko, Ilya Yegorov, Natalia Krizhanovsky, Paula Czarnowska, Irene Nikkarinen, Andrew Krizhanovsky, Tiago Pimentel, Lucas Torroba Hennigen, Christo Kirov, Garrett Nicolai, Adina Williams, Antonios Anastasopoulos, Hilaria Cruz, Eleanor Chodroff, Ryan Cotterell, Miikka Silfverberg, and Mans Hulden. 2020. SIGMORPHON 2020 shared task 0: Typologically diverse morphological inflection. In *Proceedings of the 17th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 1–39, Online. Association for Computational Linguistics.

Silvan Wehrli, Simon Clematide, and Peter Makarov. 2022. CLUZH at SIGMORPHON 2022 shared tasks on morpheme segmentation and inflection generation. In *Proceedings of the 19th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 212–219, Seattle, Washington. Association for Computational Linguistics.

Adam Wiemerslage, Miikka Silfverberg, and Mans Hulden. 2018. Phonological features for morphological inflection. In *Proceedings of the Fifteenth Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 161–166, Brussels, Belgium. Association for Computational Linguistics.

Adam Wiemerslage, Changbing Yang, Garrett Nicolai, Miikka Silfverberg, and Katharina Kann. 2023. An Investigation of Noise in Morphological Inflection. *arXiv e-prints*, page arXiv:2305.16581.

Shijie Wu, Ryan Cotterell, and Mans Hulden. 2021. Applying the transformer to character-level transduction. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 1901–1907, Online. Association for Computational Linguistics.

Weijia Xu and Marine Carpuat. 2021. Rule-based morphological inflection improves neural terminology translation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5902–5914, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Changbing Yang, Ruixin (Ray) Yang, Garrett Nicolai, and Miikka Silfverberg. 2022. Generalizing morphological inflection systems to unseen lemmas. In *Proceedings of the 19th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 226–235, Seattle, Washington. Association for Computational Linguistics.

A Detailed experimental results

A.1 Detailed comparison of experimental results with baseline models

Lang	Partition	Baseline-nonneural		Baseline-neural		CLUZH		Our Model	
		dev	test	dev	test	dev	test	dev	test
kaz	total acc	37.58	42.88	68.04	65.55	55.41	55.42	**69.64**	**68.81**
	both	87.86	85.61	97.57	97.10	95.15	93.06	**97.82**	**97.72**
	lemma	0.00	0.00	46.89	34.27	26.47	17.94	**49.56**	**40.22**
	feats	100.00	100.00	100.00	96.43	92.31	100.00	100.00	96.43
	neither	0.00	0.00	0.00	25.00	0.00	0.00	0.00	25.00
kir	total acc	46.40	44.91	66.01	71.87	**78.06**	76.47	74.10	**81.24**
	both	74.64	75.58	96.38	**99.67**	**97.10**	98.02	94.93	98.02
	lemma	0.00	0.00	71.73	74.74	58.65	54.66	**81.86**	**86.54**
	feats	**96.88**	**98.90**	34.38	44.49	93.13	95.96	46.88	60.29
	neither	0.00	0.00	42.86	25.49	57.14	**50.98**	57.14	43.14
tat	total acc	76.02	77.15	95.41	95.72	**97.07**	97.00	96.43	**97.26**
	both	75.90	77.11	95.49	95.68	**97.17**	96.97	96.52	**97.23**
	lemma	-	-	-	-	-	-	-	-
	feats	87.50	81.25	87.50	100.00	87.50	100.00	87.50	100.00
	neither	-	-	-	-	-	-	-	-
uig	total acc	50.30	51.50	77.25	76.80	77.61	77.28	**83.35**	**83.75**
	both	80.58	76.58	99.52	99.06	99.03	98.13	99.52	**99.30**
	lemma	0.00	0.00	91.67	**90.18**	58.97	54.91	91.67	90.02
	feats	**92.70**	**94.66**	48.54	50.36	87.59	88.61	63.87	68.68
	neither	0.00	0.00	48.84	42.31	46.51	53.85	**69.77**	**58.97**
uzb	total acc	89.17	88.03	91.68	91.05	**96.99**	**96.53**	92.18	92.96
	both	**97.23**	97.08	96.34	96.95	97.10	**97.53**	94.45	95.26
	lemma	0.00	0.00	96.20	**97.48**	**98.73**	96.23	96.20	96.23
	feats	**97.52**	90.75	60.33	55.52	95.04	**91.82**	76.03	78.29
	neither	0.00	0.00	25.00	75.00	**100.00**	75.00	50.00	**100.00**
total	total acc	60.86	62.11	80.41	80.41	80.65	80.24	**83.41**	**84.58**
	both	85.63	85.11	96.56	96.93	**96.95**	96.53	96.22	96.93
	lemma	0.00	0.00	66.84	62.55	46.18	41.39	**70.11**	**67.61**
	feats	**94.97**	**94.65**	48.79	52.03	90.80	91.54	62.85	70.15
	neither	0.00	0.00	39.74	37.24	46.15	51.03	**56.41**	**53.79**

Table 7: Detailed comparison of experimental results with baseline models

A.2 Detailed comparison of experimental results between two data hallucination

Lang	Partition	Baseline-neural		Baseline-neural-hall		Our Model-hall	
		dev	test	dev	test	dev	test
kaz	total acc	62.12	61.89	63.83	61.69	**68.44**	**66.40**
	both	91.51	92.65	97.09	96.27	**98.30**	**97.62**
	lemma	41.21	31.65	39.79	27.62	**47.07**	**35.48**
	feats	84.62	85.71	100.00	92.86	100.00	**96.43**
	neither	0.00	12.25	0.00	0.00	0.00	**25.00**
kir	total acc	64.75	70.51	**84.89**	**87.92**	83.99	87.29
	both	95.65	98.02	96.38	97.69	**97.10**	**99.01**
	lemma	68.78	72.05	79.75	84.06	**83.54**	**87.37**
	feats	34.38	45.96	**83.13**	**87.50**	76.25	77.21
	neither	47.62	23.53	**80.95**	68.63	61.91	**70.59**
tat	total acc	92.22	92.92	93.24	92.79	**94.90**	**95.28**
	both	92.27	92.84	93.30	92.71	**94.97**	**95.23**
	lemma	-	-	-	-	-	-
	feats	87.50	100.00	87.50	100.00	87.50	100.00
	neither	-	-	-	-	-	-
uig	total acc	74.61	72.00	**94.01**	**93.05**	91.14	92.63
	both	98.54	97.42	98.54	98.83	**99.52**	**99.30**
	lemma	91.35	89.19	91.67	89.85	**94.87**	**94.18**
	feats	42.70	39.50	**94.53**	**92.71**	81.75	87.72
	neither	41.86	34.62	**86.05**	**88.46**	83.72	79.49
uzb	total acc	89.87	87.68	**95.69**	**95.62**	94.38	94.57
	both	95.84	96.49	95.97	96.82	**96.60**	**97.47**
	lemma	97.47	98.74	97.47	98.74	**98.73**	**99.37**
	feats	47.93	34.52	**92.56**	**87.54**	78.51	76.51
	neither	25.00	37.50	**100.00**	**87.50**	50.00	75.00
total	total acc	77.27	77.06	85.83	85.42	**86.24**	**86.60**
	both	94.11	94.72	95.53	95.61	**96.65**	**97.03**
	lemma	63.56	60.63	65.16	61.61	**70.28**	**67.03**
	feats	43.06	41.76	**90.97**	**90.34**	80.04	82.92
	neither	37.18	29.66	**74.36**	**76.55**	65.39	73.10

Table 8: Detailed comparison of experimental results between two data hallucination