

Somstem: A Hybrid Stemmer for the Somali Language

Abdisalam Mahamed Badel*

*School of Information and Software Engineering
University of Electronic Science and Technology of China
Chengdu, China
202214090105@std.uestc.edu.cn*

*Corresponding author

Wenxin Tai

*School of Information and Software Engineering
University of Electronic Science and Technology of China
Chengdu, China
wxtai@std.uestc.edu.cn*

Xovee Xu

*School of Information and Software Engineering
University of Electronic Science and Technology of China
Chengdu, China
xovee@std.uestc.edu.cn*

Abdikadir Maktal Siraad

*Department of Computer Science
Jigjiga University
Jigjiga, Somali Region, Ethiopia
eng.layaab@gmail.com*

Abstract—Somali, a language characterized by rich morphological inflection, poses challenges for natural language processing (NLP) tasks such as machine translation and information retrieval, primarily due to its extensive use of affixation. Currently, no robust stemming solution exists for Somali. In this study, we introduce Somstem, a novel hybrid stemming approach that integrates linguistic expertise with machine learning techniques. Somstem comprises a high-precision, rule-based component containing over 700 rules derived from an in-depth analysis of Somali morphology, combined with a fine-tuned mT5-small model trained on a newly created dataset of 23,181 word–stem pairs. By utilizing the precision of handcrafted linguistic rules and the generalization capabilities of machine learning, this hybrid architecture addresses the limitations inherent in purely rule-based or data-driven methods. We evaluated Somstem on a challenging dataset of 1,650 unseen word–stem pairs. Empirical results demonstrate that Somstem outperforms conventional machine learning methods and state-of-the-art large language models (LLMs), achieving a precision of 82.45%, recall of 79.70%, and an F1-score of 79.84%. Furthermore, detailed analyses show that Somstem consistently captures morphological patterns across various parts-of-speech (POS) categories.

Index Terms—Stemming, Morphology, Somali, Low-resource

I. INTRODUCTION

Natural Language Processing (NLP) relies heavily on accurate representations of language. However, the inherent morphological complexity of many languages, such as Somali, presents significant challenges [1]. This complexity arises from rich inflectional patterns, where one word can have numerous diverse forms. Stemming a crucial preprocessing step in many NLP pipelines [2], aims to reduce words to their common root form [3]. It is commonly defined as a method for conflating various inflected forms of a word into a single representation [4]. This process plays a vital role in numerous downstream applications, including machine translation and

information retrieval [1], [5], and can significantly enhance their effectiveness [6], [7]. Despite its importance, to the best of our knowledge, no robust and publicly available stemmer currently exists for the Somali language. To address this gap, we introduce Somstem, a novel hybrid stemmer specifically designed for Somali. Somstem integrates a carefully crafted rule-based system, which features over 700 unique rules derived from extensive morphological analysis, with a fine-tuned mT5-small model. This hybrid architecture combines linguistic expertise with deep learning and effectively overcomes the limitations of purely rule-based or data-driven approaches. As part of this work, we annotated a dataset of 25,467 Somali words with their corresponding stems, thereby creating a valuable resource for the Somali NLP community. The dataset includes 23,181 word–stem pairs used for fine-tuning the mT5-small model, a challenging set of 1,650 word–stem pairs used for testing, and 636 words used to assess Somstem’s morphological sensitivity. The rule-based component of Somstem is organized into five distinct categories and is capable of recursively removing multiple suffixes from a single word to ensure accurate and efficient stem extraction. For example, the Somali word *macalimiin* “teachers” is accurately stemmed to *macalin* “teacher”. Beyond affix removal, the system also applies suffix replacement rules. To further enhance the rule-based system, the mT5-small model processes unseen words or words not effectively handled by the handcrafted rules, which improves the overall accuracy and coverage of the system. Additionally, we explore the potential of large language models (LLMs) in supporting stemming for low-resource languages like Somali, providing insights into the future of stemming methodologies.

Somstem represents a significant advancement in Somali NLP by providing a robust and reliable stemming solution. We are releasing both the software and dataset to encourage

further research in this area, particularly for Somali and other low-resource languages with closely related morphological structures. Our contributions are as follows:

- 1) We develop the first publicly available stemmer for the Somali language which addresses an important gap in the NLP toolkit for this low-resourced language.
- 2) We create a standardized benchmark dataset for Somali stemming that serves as a valuable resource for the research community.
- 3) We conduct experiments and compare Somstem with other conventional machine learning methods integrated with rule-based algorithm, as well as with state-of-the-art LLMs.

II. RELATED WORK

Stemming is a well-established and widely applied technique in NLP, particularly for languages such as English [8]. One of the earliest stemming algorithms employed a single-pass method to remove the longest suffixes from words [9]. Among the most notable techniques is Porter's stemmer [10], which remains highly effective and widely used for English text processing [8]. This algorithm operates in five iterative steps, applying specific transformations until predefined conditions are met. Several rule-based stemmers have also been developed for other languages. For instance, [11] introduced a postfix-based approach for Telugu, while [12] developed a rule-based stemmer for an agglutinative language, effectively handling suffix removal. Beyond purely rule-based methods, hybrid stemming algorithms have also been explored. Examples include those developed for Kokborok [13] and Kananda [14]. Tagging algorithms can further enhance stemming performance. For instance, [15] demonstrated that part-of-speech tagging contributes to more accurate stem extraction in Dutch. These diverse methods show how stemming techniques are adapted to the unique linguistic challenges of different languages. Despite these advances, a dedicated stemmer for Somali has not yet been developed. Thus, we propose a hybrid stemmer for the Somali language.

III. METHODOLOGY

Here we discuss how we developed this algorithm.

A. Somstem

The proposed Somstem algorithm consists of six steps: **Step 1:** Preserves the original form and meaning of proper nouns, such as personal names, country names, and place names. **Step 2:** This is a recursive step that checks words for multiple suffixes. **Step 3:** Performed after the recursive process has been completed. It addresses the fact that, in the Somali language, word stemming involves not only removing substrings but also replacing parts of words. **Step 4:** In this step, we recognize that some consonants may remain after suffixes are removed. These residual consonants do not contribute to the meaning of the words and can cause stem mismatches for the same word when derived from different parts of speech. **Step 5:** Used to remove prefixes in Somali.

The prefixes are identified based on their repeating patterns at the beginning of a word (i.e., two or more characters that appear more than once at the start of words). **Step 6:** After Step 5, the output of the rule-based algorithm is passed to a fine-tuned mT5-small model, which takes two inputs: the stem produced by the rule-based algorithm and the original word. Further details can be found in Algorithm 1 and Fig. 1.

Algorithm 1 Somali Stemmer

```

1: exception ← exceptions
2: suffix ← suffixes
3: prefix ← prefixes
4: if word in exception then
5:   return word
6: end if
7: while word has suffix do
8:   if suffix in suffixes then
9:     remove suffix
10:  else
11:    if word meets listed rules then
12:      replace part of the word
13:    end if
14:    if word ends with two consecutive consonants then
15:      Remove the second consonant counting from
the right
16:    end if
17:    if word starts with prefix + prefix[0] then
18:      remove prefix
19:    end if
20:    pass (original word, stem) to mT5
21:  end if
22:  if original word ≠ rule-based stem or word in excep-
tions then
23:    return word
24:  end if
25:  generate stem
26: end while

```

B. Data collection and annotation

To create an important resource for the Somali language and establish a baseline for future research, we conducted a benchmarking process by annotating a dataset of 24,831 inflected and derived words with their corresponding stems. This dataset comprises 23,181 training words and 1,650 unseen test words. The ground truth/test data were collected from a variety of sources, including literature [16], online news articles [17], government documents [18], and educational textbooks.¹ In contrast, the training data were primarily derived from dictionaries, including [19] and [20]. The 23,181 dictionary words were verified by language experts and native speakers. Additionally, the ground truth/test data were annotated by four individuals, each tasked with identifying the stems of 1,650 words. In addition to the training and test datasets, we

¹We also included texts and words collected from books provided by the Somali Region Education Bureau.

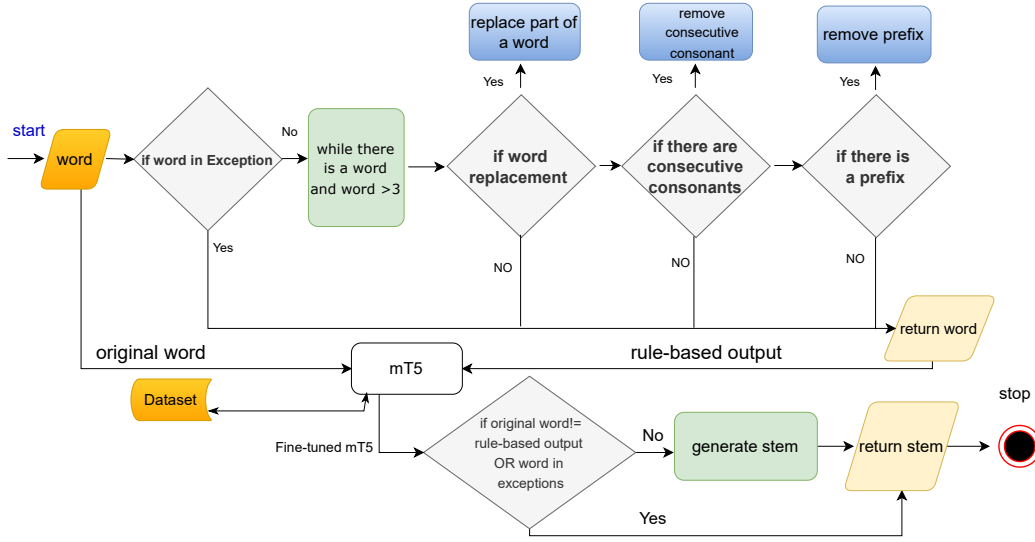


Fig. 1. Algorithm Flowchart.

also annotated 636 parts-of-speech test words derived from dictionaries, excluding numbers and certain proper nouns, to evaluate the stemmer's sensitivity to various morphological categories. These excluded categories were constructed separately by native speakers of the language, meaning they were not sourced from dictionaries.

IV. EXPERIMENTS

To develop **Somstem**, we implemented a high-precision rule-based algorithm with over 700 crafted rules and integrated it with a fine-tuned mT5-small model [21], a multilingual pre-trained language model (PLM) comprising 300 million parameters. The mT5-small model supports over 101 languages, including Somali. Its prior training on Somali data further enhances its applicability to our use case. We fine-tuned the mT5-small model using a learning rate of $5e-5$, a batch size of 8, 3 epochs, and a weight decay of 0.01. The experiments were conducted using PyTorch and the Huggingface Transformers library [22]. Words are initially stemmed using the rule-based algorithm, while the mT5-small model is applied in cases where the rule-based algorithm fails to produce a stem.

Dataset. We used 23,181 samples as a training set to fine-tune the mT5-small model, while a challenging dataset of 1,650 samples was used for testing. Furthermore, we evaluated Somstem using an additional set of 636 samples representing various parts-of-speech in the Somali language.

Evaluation metrics. To assess the performance of Somstem, we utilized standard evaluation metrics such as precision, recall, accuracy, and F1-score.

A. Baseline methods

We compare *Somstem* with the following baseline methods: (1) Rule-based algorithm integrated with Hidden Markov Models (HMM) [23]: A statistical approach commonly employed for stemming and tagging tasks. (2) Morfessor [24]:

A statistical toolkit designed for morphological segmentation across multiple languages. (3) ChatGPT [25]: Specifically, we used GPT-3.5-turbo and GPT-4o-mini via the OpenAI API. (4) DeepSeek-V3 [26]: A recent language model based on a mixture-of-expert architecture. (5) Rule-based algorithm integrated with two fine-tuned variants of BERT: roberta-base [27] and mbert-base-cased [28], both recognized for their robustness and efficacy in language understanding tasks.

V. RESULT ANALYSIS

In this section, we discuss the results, restricting our report to those obtained from the test dataset.

Comparison with conventional methods. Table I presents the performance of Somstem in comparison with conventional methods. The results show that Somstem outperforms all competing models, achieving precision, recall, and F1-scores of 82.45%, 79.70%, and 79.84%, respectively. Morfessor recorded the lowest scores, underscoring the limitations of purely statistical approaches and highlighting the advantages of hybrid systems that incorporate rule-based methods. Furthermore, as illustrated in Fig. 2b, Somstem achieves a higher accuracy score of 80%.

TABLE I
PERFORMANCE COMPARISON OF *Somstem* WITH OTHER CONVENTIONAL MODELS. BOLD INDICATES THE BEST PERFORMANCE, WHILE THE UNDERLINED INDICATES THE SECOND-BEST SCORES (%). (HMM + RULES = HIDDEN MARKOV MODEL INTEGRATED WITH OUR RULE-BASED ALGORITHM).

Model	Precision	Recall	F ₁ -score
Morfessor	21.13	22.48	21.39
HMM+Rules	80.27	72.67	74.91
Somstem	82.45	79.70	79.84

Comparison with LLMs. To evaluate LLMs in low-resource stemming, we employed a 3-shot learning paradigm using a

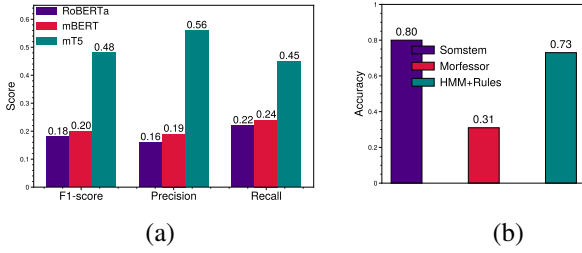


Fig. 2. The impact of the rule-based algorithm on the performance of fine-tuned PLMs (a) and a comparison of accuracy between Somstem and two conventional methods (b).

prompt template adapted from [6]. DeepSeek-V3 and GPT-4o-mini achieved comparable F1-scores of 52.92% and 52.09%, respectively (Table II), with DeepSeek-V3 performing slightly better. While these LLMs demonstrate potential, Somstem significantly outperforms them, likely due to the integration of the rule-based system. As shown in Fig. 3b, DeepSeek-V3 achieves the highest accuracy among the three LLMs in the 3-shot setting. Table V shows sample stemmed words.

TABLE II

COMPARISON OF *Somstem* WITH THE LLMs ON THE TEST DATASET (%).

Model	Precision	Recall	F_1 -score
gpt-3.5-turbo	49.42	31.61	35.37
gpt-4o-mini	64.21	47.87	52.09
DeepSeek-V3	<u>66.38</u>	<u>48.16</u>	<u>52.92</u>
Somstem	82.45	79.70	79.84

TABLE III

PERFORMANCE COMPARISON OF *Somstem* WITH OTHER FINE-TUNED PLMS INTEGRATED WITH THE RULE-BASED ALGORITHM (%).

Model	Precision	Recall	F_1 -score
RoBERTa+Rules	79.53	73.15	74.83
mBERT+Rules	<u>79.98</u>	<u>73.70</u>	<u>75.33</u>
Somstem	82.45	79.70	79.84

Impact of fine-tuned pretrained language models. To evaluate the strength of the mT5-small model in addressing our problem, we compared it with two other models that are also integrated with the rule-based algorithm. Specifically,

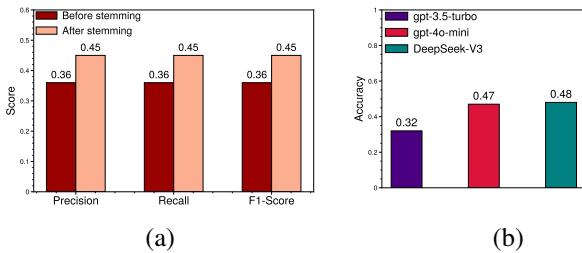


Fig. 3. Impact of Somstem on Information Retrieval (a) and (b) accuracy comparison of different LLMs.

we examined two fine-tuned variants of BERT: roberta and mbert, both widely recognized for their robustness and efficacy in language understanding tasks. As shown in Table III, *Somstem* demonstrates exceptional performance.

Morphological sensitivity. Given the importance of parts-of-speech tagging in NLP, we also assessed the robustness of Somstem across various parts-of-speech categories as detailed in Table IV. The evaluation revealed that adjectives and numbers were handled most effectively, while proper nouns exhibited the lowest performance.

The impact of the rule-based system on performance. We removed the rule-based algorithm from mT5-small to evaluate its impact and compared the results with other models that also lacked rule-based components. As shown in Fig. 2a, mT5-small outperformed all other models. This is evidence that the fine-tuned mT5 model can help our rule-based algorithm by handling unseen words that are out of the scope of the rules.

TABLE IV

THE PERFORMANCE OF *Somstem* IS EVALUATED ACROSS VARIOUS SOMALI PARTS-OF-SPEECH CATEGORIES (%). EACH CATEGORY IS ASSESSED USING A SAMPLE OF 100 RANDOMLY SELECTED DICTIONARY WORDS, WHILE THE NUMBER CATEGORY IS EVALUATED BASED ON 36 DISTINCT REPRESENTATIONS OF NUMBER WORDS.

Parts-of-speech category	Precision	Recall	F_1 -score
ADJECTIVE	83.33	83.00	82.93
VERB	78.00	78.01	77.67
NOUN	72.74	72.73	72.73
PROPER NOUN	65.00	65.00	65.00
ADVERB	76.00	78.00	76.67
PROPONOUN	76.02	74.00	73.87
NUMBER	100.00	100.00	100.00

TABLE V

SAMPLE STEMMED WORDS.

Word	Rule Applied	Stem	Translation
Warka	Remove <i>ka</i>	war	news
Xarunta	Remove <i>ta</i>	xarun	station
Maamulid	Remove <i>id</i>	maamul	manage

A. Impact of Somstem on information retrieval

We evaluated the effectiveness of the algorithm for information retrieval using the top 20k rows from [29]. We constructed a set of relevant queries, we then, used a BM25 retriever [30]. The results demonstrated that applying the stemmer improved performance, increasing recall from 36% to 45% and precision from 36% to 45% (see Figure 3a). This improvement was observed when retrieving the top K (K@10) relevant documents, using both original words and stemmed words as query expansion terms.

B. Index compression factor

We selected the top 20k rows from a Somali corpus [29], to evaluate the effectiveness of the proposed stemmer in reducing the index size of words. The initial selection comprised 702,834 words, including 58,285 unique stems. After applying

the stemmer, the corpus size decreased to 692,159 words, with 25,575 unique stems remaining. These results indicate a significant reduction in index size.

C. Computational efficiency

Using our machine (Windows 10, 32GB RAM, NVIDIA GeForce RTX 3080 Ti Laptop GPU), we evaluated Somstem's computational efficiency. Processing 500 words required just 3.9 seconds, while 1,650 words were processed in 9.6 seconds.

VI. CONCLUSION

In this paper, we present Somstem, the first publicly available stemmer for Somali, a low-resource language. To develop this tool, we identified over 700 morphological affixes in Somali and designed a rule-based algorithm integrated with a fine-tuned mT5-small model. Additionally, we constructed a dataset of 25,467 annotated word-stem pairs to fine-tune the model and evaluate its performance. Comprehensive evaluations demonstrate that Somstem outperforms both traditional machine learning approaches and state-of-the-art LLMs, achieving an F_1 -score of 79.84% while exhibiting strong morphological sensitivity. Somstem is well-suited for practical applications, such as search engines, machine translation, and information retrieval systems. To the best of our knowledge, it is the first publicly available tool of its kind for the Somali language, representing a significant milestone in advancing NLP for Somali. As future work, we recommend adding additional rules to the rule-based algorithm.

REFERENCES

- [1] H. B. Patil, A. S. Patil, Mars: "A rule-based stemmer for morphologically rich language marathi", in: 2017 International Conference on Computer, Communications and Electronics (Comptelix), 2017, pp. 580–584.
- [2] H. Alshalabi, S. Tiun, N. Omar, E. abdulwahab Anaam, Y. Saif, "Bpr algorithm: New broken plural rules for an arabic stemmer", *Egyptian Informatics Journal* 23 (3) (2022) 363–371. doi:https://doi.org/10.1016/j.eij.2022.02.006.
- [3] A. Bimba, N. Idris, N. Khamis, N. F. Noor, "Stemming Hausa text: using affix-stripping rules and reference lookup", *Language Resources and Evaluation* 50 (3) (2016) 687–703. doi:10.1007/s10579-015-9311-x.
- [4] F. N. Flores, V. P. Moreira, C. A. Heuser, "Assessing the impact of stemming accuracy on information retrieval", in: *Computational Processing of the Portuguese Language*, Springer Berlin Heidelberg, Berlin, Heidelberg, 2010, pp. 11–20.
- [5] K. Kariyawasam, S. Senanayake, P. S. Haddela, "A rule based stemmer for sinhala language", in: 2019 14th Conference on Industrial and Information Systems (ICIIS), 2019, pp. 326–331.
- [6] S. Wang, S. Zhuang, G. Zuccon, "Large language models based stemming for information retrieval: Promises, pitfalls and failures", In *Proc. of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '24*, Association for Computing Machinery, New York, NY, USA, 2024, p.2492–2496.
- [7] J. Singh, V. Gupta, "A systematic review of text stemming techniques", *Artificial Intelligence Review* 48 (2) (2017) 157–217. doi:10.1007/s10462-016-9498-2.
- [8] D. Bijal, S. Sanket, "Overview of stemming algorithms for indian and non-indian languages", *ArXiv abs/1404.2878* (2014).
- [9] J. B. Lovins, "Development of a stemming algorithm", *Mechanical Translation and Computational Linguistics* 11 (1-2) (1968) 22–31, published March and June 1968.
- [10] M. F. Porter, "An algorithm for suffix stripping", *Program: electronic library and information systems* 14 (3) (1980) 130–137. doi:10.1108/eb046814.
- [11] M. V. Raju, M. Sreenivasulu, "A Lightweight Stemmer for Telugu Language", 4th International Conference on Inventive Research in Computing Applications, ICIRCA 2022 - Proc. (Icirca) (2022) 1385–1388. doi:10.1109/ICIRCA54612.2022.9985623.
- [12] N. Sultanova, K. Kozhakhmet, R. Jantayev, A. Botbayeva, "Stemming algorithm for kazakh language using rule-based approach", 2019 15th International Conference on Electronics, Computer and Computation, ICECCO 2019 (Icecco) (2019) 3–6. doi:10.1109/ICECCO48375.2019.9043253.
- [13] P. Debbarma, P. Hrangkhaw, S. Debbarma, "Modeling a Hybrid Stemmer for Kokborok", *Proceedings of 2023 International Conference on Intelligent Systems, Advanced Computing and Communication, ISACC 2023* (2023) 1–7doi:10.1109/ISACC56298.2023.10084267.
- [14] S. Bhat, "Statistical stemming for kannada", in: P. Bhattacharyya, M. G. A. Malik (Eds.), *Proc. of the 4th Workshop on South and Southeast Asian Natural Language Processing, Asian Federation of Natural Language Processing*, Nagoya Congress Center, Nagoya, Japan, 2013, pp. 25–33.
- [15] A. Jonker, C. de Ruijt, J. de Gruijl, "Bag & tag'em - a new Dutch stemmer", in: *Proc. of the Twelfth Language Resources and Evaluation Conference, European Language Resources Association, Marseille, France, 2020*, pp. 3868–3876.
- [16] aktabadda, "Diiwaanka gabayadii sayid maxamed cabdulle xasan", <https://maktabadda.com/2014/04/24/diiwaanka-gabayadii-sayid-maxamed-cabdulle-xasan/>, accessed: 12 April 2023 (2014).
- [17] Hiiraan Online, "Hiiraan online - somali news and information", <https://www.hiiraan.com/somali.aspx>, accessed: 13 April 2023.
- [18] iiraan Online, "Hagaha dastuurka federaalka soomaaliya", <https://www.hiiraan.ca/PDF/files/Dastuurka/GuidebookA4S%20280912web.pdf>, accessed : 12 April 2023.
- [19] A. Puglielli, C. C. Mansuur, "Qaamuuska af soomaaliga", <https://romatrepress.uniroma3.it/wp-content/uploads/2019/05/qaam-cama.pdf>, accessed 14 June 2023.
- [20] Y. C. Keenadiid, "Qaamuuska Af-Soomaaliga", <https://arcadia.sba.uniroma3.it/bitstream/2307/2021/1/DizionarioYaasiinC.Keenadiid.pdf>, accessed 9 June 2023.
- [21] L. Xue, N. Constant, A. Roberts, M. Kale, R. AlRfou, A. Siddhant, A. Barua, C. Raffel, "mT5: A massively multilingual pre-trained text-to-text transformer", *Proc. of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, 2021*, pp.483–498.
- [22] T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, et al. "Transformers: State-of-the-art natural language processing", In *Proc. of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, Association for Computational Linguistics, Online, 2020, pp. 38–45.
- [23] L. Rabiner, "A tutorial on hidden markov models and selected applications in speech recognition", *Proceedings of the IEEE* 77 (2) (1989) 257–286. doi:10.1109/5.18626.
- [24] P. Smit, S. Virpioja, S.-A. Grönroos, M. Kurimo, "Morfessor 2.0: Toolkit for statistical morphological segmentation", In *Proc. of the Demonstrations at the 14th Conference of the European Chapter of the Association for Computational Linguistics, Gothenburg, Sweden, 2014*, pp. 21–24.
- [25] OpenAI, "Chatgpt", accessed: 2024-11-07 (2022). <https://openai.com/api/>
- [26] DeepSeek Inc., "Deepseek api documentation", accessed: 2024-11-07 (2024). <https://api-docs.deepseek.com/>,
- [27] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, et al. "Roberta: A robustly optimized bert pretraining approach", (2019). arXiv:1907.11692. <https://arxiv.org/abs/1907.11692>
- [28] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding", In *Proc. of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics, Minneapolis, Minnesota, 2019*, pp. 4171–4186. doi:10.18653/v1/N19-1423.
- [29] A. Conneau, K. Khandelwal, N. Goyal, V. Chaudhary, G. Wenzek, F. Guzmán, et al. "Unsupervised cross-lingual representation learning at scale", In *Proc. 58th Annu. Meeting Assoc. Comput. Linguistics, Online, Jul. 2020*, pp. 8440–8451, doi: 10.18653/v1/2020.acl-main.747
- [30] A. Trotman, A. Puurula, and B. Burgess, "Improvements to BM25 and language models examined", In *Proc. 19th Australasian Doc. Comput. Symp. Melbourne, VIC, Australia, 2014*, pp. 58–65, doi: 10.1145/2682862.2682863.