AFRICAN JOURNAL OF SCIENCE, TECHNOLOGY AND SOCIAL SCIENCES

Journal website: https://journals.must.ac.ke

A Publication of Meru University of Science and Technology

A review of techniques for morphological analysis in natural language processing

Mutwiri George [1]*, Mutua Makau[1], Omamo Amos [1]

[1]Meru University of Science and Technology, Meru, Kenya.

ARTICLE INFO

ABSTRACT

KEY WORDS

Natural Language Processing (NLP)

Morphological segmentation

Low-resourced language

Lemmatization

Part-of-speech tagging,

Word segmentation

Segmentation

Natural language is a crucial tool to facilitate communication in our day-to-day activities. This can be achieved either in text or speech forms. Natural language processing (NLP) involves making computers understand and process natural language. NLP has enhanced the way humans interact with computers, from having computers use speech to talk to humans as well as having computers translate human speech. Apart from speech, computers also create and understand sentences in natural language in a process called morphological analysis. Morphological analysis is an important part in natural language processing, being applied as a preprocessing step in most NLP tasks. Morphological analysis consists of four subtasks, that is, lemmatization, part-of-speech (POS) tagging, word segmentation and stemming. In this paper, we explore in detail each of these tasks of morphological analysis. We then evaluate the techniques used in this NLP field. Finally, we give a summary of the results of each of these techniques.

* Corresponding George Muthee Email: mutwirigeorge3@gmail.com

https://doi.org/10.58506/ajstss.v1i2.11

AFRICAN JOURNAL OF SCIENCE, TECHNOLOGY AND SOCIAL SCIENCES ISSN :2958-0560

https://journals.must.ac.ke © 2022 The Authors. Published by Meru University of Science and Technology

Introduction

Morphological analysis is the study of how words in a language are formed by combining morphemes (Ding et al., 2019; Premjith et al., 2018; Yambao & Cheng, 2020). It is a subfield in semantic analysis containing the following subfields: morphological segmentation, lemmatization, POS tagging and stemming.

Tasks in morphological analysis

Morphological segmentation is an NLP task that involves of dissecting words into their constituent morphemes (Liu et al., 2021; Wang et al., 2019; Yang et al., 2019). This is an important NLP task that alleviates out-of-vocabulary and data sparsity problems (Liu et al., 2021). Lemmatization is the process of converting an inflected word, like talked, to its citation form, which is talk (Malaviya et al., 2019) using a lemmatizer (Ingólfsdóttir et al., 2019). This task aims at reducing a given word to a form that represents its entry in a dictionary (Chary et al., 2019). POS tagging aims at assigning each word a unique tag label that indicates its syntactic role in a sentence. Such labels include plural, verb and noun. The tags are necessary to identify the grammatical function a word plays in the sentence (Ayana, 2015). POS tagging is similar to chunking (Kudo et al., 2002). Two approaches could be used to perform POS tagging (Tukur et al., 2020): rule-based approach, where rules of language are handwritten and manually verified; and corpus-based or feature-engineering approach, which is performed from a training dataset that acts as the knowledge resource. Stemming is the processing of reducing a word into its stem which is not necessarily its morphological root (Koirala & Shakya, 2020). Stemming reduces a word ending to its root without adhering to morphological rules (Ingólfsdóttir et al., 2019). Stemming is a subtask of lemmatization when the concern is only to remove the suffix (Patel & Patel, 2019).

Techniques for morphological analysis

These are various techniques that have been applied in the field of morphological analysis of various natural languages. These techniques will be analyzed for each task. Depending on the focus of the technique, different results have been achieved across different test cases.

a) Morphological segmentation

Machine Learning

Three neural models that treat chunks as the basic unit for labeling in a sequence labeling problem were proposed by (Zhai et al., 2017). The first model is a bidirectional LSTM. The second model is also a bidirectional LSTM for encoding and sentence representation. The second model improves the performance of the first. The third model was a greedy encoder-decoder-pointer framework for segmentation and it too improved the first and the second models. The CoNLL 2000 shared task dataset was used as the data source. The third model performed better than the first two, including the baseline model, achieving a 94.72 F1 score. The challenge faced during the experiment was that the models I and II failed to consistently improve on the final F1 score.

An unsupervised morphological segmentation dataset created by the University of Pennsylvania and Linguistic Data Consortium for the DARPA LORELEI Program, containing about 2000 tokens for morphological segmentation for each of 9 resource-poor languages and root information for 7 languages in this category was proposed by (Mott et al., 2020). The 7 languages include: Akan (2048 tokens), Hindi (2028 tokens), Hungarian (2027 tokens), Indonesian (2035 tokens), Russian (2050 tokens), Spanish (2050 tokens), Swahili (2023 tokens), Tagalog (2001 tokens) and Tamil (2028 tokens). Annotation was conducted in two phases: a first pass done in 2018, and a quality control done in 2019. Four systems were evaluated on the dataset: Morfessor (Creutz & Lagus, 2007), MorphoChain (Narasimhan et al., 2015), ILP and ParaMa(Xu et al., 2018). On Swahili, MorphoChain outperformed the three other counterparts with an F1 measure of 0.4306, closely followed by Morfessor with 0.4320. This can be explained by Morfessor's poor performance on Bantu morphology (Pauw & de Schryver, 2008). A challenge with this model is that it does not distinguish between inflectional and derivational morphology

A method of segmenting both Chinese and Japanese using both word and character-level information was presented by (Nakagawa, 2004). The datasets used for Chinese word segmentation are the Academia Sinica corpus, the Hong Kong City University corpus and the Beijing University corpus. The dataset used for Japanese segmentation was the RWCP corpus. For comparison, Bakeoff-1, Bakeoff-2, Bakeoff-3, Maximum Matching and Character Tagging systems were used for Chinese segmentation while ChaSen, Maximum Matching and Character Tagging systems were used for Japanese. The model achieved the F-scores on Chinese segmentation: 0.972 on the Academia Sinica corpus, 0.950 on the Hong Kong City University corpus and 0.954 on the Beijing University corpus. These results were better than those posted by the benchmark systems. The model achieved an F-score of 0.993 on Japanese segmentation, better than the closer benchmark system: the ChaSen with 0.991. The strength of the model is that it is able to perform better than the benchmark systems for unknown words than for known words.

MorphAGram (Eskander et al., 2020) was presented as a publicly accessible unsupervised framework for morphological segmentation based on 'Adaptor Grammars (AG)' and a previous work (Eskander et al., 2016). The model is evaluated on 12 languages and it performed well. Their adapter grammars consisted of probabilistic context free grammars and a caching model. Datasets used for the experiments include 50,000 words from the Morpho Challenge competition (German, Finnish, Turkish and English), 50,000 words of the Georgian Wikipedia (Georgian), 50,000 words of the Arabic PATB corpus (Arabic) and a 2132 -word dataset drawn from (Kann et al., 2018) (Wixarika, Mexicanero, Yorem Nokki and Nahuatl). Text segmentation could be either transductive (where a word needs to be in the learner's vocabulary first) or inductive (where a word does not need to be in the learner's vocabulary). The research concluded that inductive text segmentation had no improvement in performance. (Creutz & Lagus, 2007) and MorphoChain (Narasimhan et al., 2014) were chosen as the baselines. When Boundary Precision Recall metric is used, the model outperforms Morfessor (Creutz & Lagus, 2007) and MorphoChain on all languages. On these metrics, language independence reduces error rates from Morfessor by 26.0% and MorphoChain by 38.0%. However, Morfessor is not well equipped to handle Bantu morphology (Pauw & de Schryver, 2008).

In their novel morphological segmentation work for Tigrinya language, (Tedla & Yamamoto, 2018) combine CRFs with LSTMS for detecting morpheme boundaries. Begin (B), Inside (I), Outside (O), Single (S) and End (E) labels are used to annotate morphemes in order to mark morpheme boundaries. A window size of 5 characters was used for word embeddings. 10-fold cross validation is performed on a 45,127-token corpus. The BIE tagging strategy achieved the highest F1 score (94.67), followed by the BIES strategy (94.59), then by BIO strategy (90.11) and lastly the BIOES strategy (88.39). Their experiment shows that LSTMs performed better than their CRF counterpart, with both outperformed by bidirectional LSTMs. The corpus size was small, which contributed to the poor performance of the BIOES strategy which requires more details.

A bidirectional LSTM model is presented by (Almuhareb et al., 2019) for the word segmentation of Arabic with data sourced from the Arabic Treebank. Their character embeddings used a window size of 5. The model was trained on a 48 million token dataset. Word segmentation without rewriting achieved an F1 score of 97.65%, but was outperformed when rewriting was used, which improved the F1 score by 4.03%. More training epochs were needed to improve accuracy for tokens in the dataset that appeared less frequently and even then, the model failed to learn the least frequently occurring label.

In their work on morphological segmentation for Persian, (Ansari et al., 2019) use supervised methods trained on a well labelled manual corpus. In their experiment, the bidirectional LSTM model outperformed other models with an F score of 90.53, closely followed by the unidirectional LSTM at 88.80. The k-Nearest Neighbor model outperformed all other models in predicting boundaries.

Chinese word segmentation as a tagging problem based on word-internal positions was presented by (Xue, 2003). Tagging is based on maximum entropies. The experiment was branched into two: the first comprising a maximum matching method and serving as the baseline; and the second comprising the maximum entropy model. The dataset used for the experiment was the Xinhua newswire section of the Penn Chinese Treebank. Training data consisted of 237,791 words while the test set consisted of 12,598 words. The maximum entropy model achieved better results than the maximum matching method for the segmentation task, achieving an F-score of 94.98%. When the test set had no new words, the maximum matching method achieved an F-score of 95.15% compared to a score of 89.77% when the test set contained new words. The model was also capable of segmenting personal names, achieving a recall of 86.86%. The notable challenge with this segmentation approach is that it was not able to accurately segment foreign personal names.

A morphological analyzer incorporated in an English to Swahili, Russian and Hebrew phrase-based machine translation model was proposed by (Chahuneau et al., 2013). The model first identifies a stem bearing meaning on the target language and later selects the appropriate inflection using a discriminative model. The translations are generated in short phrases called synthetic phrases, according to rule extraction techniques (Chiang, 2007). Only Russian segmentation was based on supervised methods. The assumption on unsupervised method was to decompose a word into prefixes, a stem and suffixes. A regular grammar was developed to model possible morphemes in the morphologically resource-rich languages, in which a word comprised of a set of prefixes, a stem and a set of suffixes. Inflections are predicted using a stochastic gradient descent function that make the most of the conditional log-likelihood of the source language sentence feature pairs. A Conditional Random Field (CRF) tagger is used on the source language, which is trained on the Penn Treebank's section 02-21 and additionally, the TurboParser for dependency parsing, trained also on the Penn Treebank. The Global Voices project and the Helsinki Corpus of Swahili were chosen as the Swahili data sets. The synthetic phrases model outperformed the class-based language model on all test cases. The English-to-Swahili translation task outperformed other tasks, achieving a BLEU score of about 19.0, followed by Hebrew at about 17.6 and lastly, Russian at about 16.2. The strength of the model is that 1) translation is context-based, 2) it does not require language-specific engineering, and 3) it is workable with syntax- or phrase-based decoder without modification. Also, the model is able to generate unseen inflections (Botha & Blunsom, 2014).The weakness with the model is that the intrinsic inflectional dataset for evaluation was noisy, owing to errors in word alignments, with accuracy on predicting Swahili inflection being 78.2%, higher than Russian (71.2%) but lower than Hebrew (85.5%).

Rule based approaches

Morphological segmentation is incorporated in the Abu-MaTran project systems to the English-to-Finnish language pair (Sánchez et al., 2016). Segmentation and deep learning address the data scarcity problem and the Finnish complex morphology. The morphological segmentation applied was rule-based. The Moses toolkit (Koehn et al., 2007) was used to preprocess training corpora. The corpora used for the experiments included the newsdev2015, newstest2015 and an SMT translated corpora of Finnish to English. The research concluded that rule-based morphological segmentation improved quality for both neural machine translation and statistical machine translation. The research also concluded that neural machine translation achieves better results than statistical machine translation. The disadvantage of this experiment is it took at least 5 days to train the models.

A morphological analyzer is incorporated in an English to Swahili, Russian and Hebrew phrase-based machine translation model (Chahuneau et al., 2013). The model first identifies a stem bearing meaning on the target language and later selects the appropriate inflection using a discriminative model. The translations are generated in short phrases called synthetic phrases, according to rule extraction techniques (Chiang, 2007). Only Russian segmentation was based on supervised methods. The assumption on unsupervised method was to decompose a word into prefixes, a stem and suffixes. A regular grammar was developed to model possible morphemes in the morphologically rich languages, where a word comprised of a set of prefixes, a stem and a set of suffixes. Inflections are predicted using a stochastic gradient descent function that maximizes the conditional log-likelihood of the source language sentence feature pairs.

fsm2 finite state method for the automatic analysis of Runyakitara nouns is presented by (Katushemererwe & Issue, 2010). All noun lexemes in the language were built into fsm2. Nouns were extracted from a Runyakitara dictionary and manually coded into noun sub-classes. The model comprised of modules/files, that is, a special symbol file, a noun grammar file and a replacement rule file. All three comprised the finite state transducer. The symbol specification file contained a mapping between human readable symbols and integers representing these symbols in the system. The noun grammar file contained quasi context free grammars. The replace rules are applied to enforce grammatical forms of nouns, like replacing ‘u’ with ‘w’ whenever ‘m’ occurs to the left of ‘u’ and either ‘a’ or ‘o’ or ‘i’ to its right. Further, the replacement rules could modify the noun by deletion, substitution or insertion of symbols. The system was evaluated on a dataset extracted from a weekly newspaper and an orthography reference book, both in a different language. The system achieved a precision of 80% on 4472 words and a recall of 80% on the 5599-word corpus.

Runyagram, a formal system for the morphological segmentation of Runyakitara verbs based on the fsm2 interpreter is presented by (Fridah & Thomas, 2010). Just like their similar model for nouns (Katushemererwe & Issue, 2010), Runyagram finite state transducer comprised of a special symbol file, a grammar file and a replacement rule file containing about 34 rules. The verb grammar is defined according to the number of morphemes a verb can take, from minimum to maximum. The grammar is converted into an unweighted finite-state acceptor by converting rules into directed graphs. The grammar contained about 330 rules was thus converted into a finite-state acceptor containing about 1200 transitions and about 800 states. The system was tested against 3971 verbs from an orthography reference book and a dictionary of another Bantu language. The system scored a recall of 86% and a precision of 82%.

A Setswana tokenizer based on two transducers and a finite-based morphological analyzer was presented by (Pretorius & Pretorius, 2009). The system is majorly inclined to disjunctive orthography. Morphotactics were developed on the lexc tool of the Xerox finite state tools, while morphological alternations were modeled in the xsft tool. The contents of the lexc tool and xsft tool are combined into a finite state transducer, which was considered the morphological analyzer. Errors in their tokenizer output were presented to humans for examination. 547 Setswana orthographic words were obtained to evaluate the system. The benchmark of the evaluation was a hand-tokenized text by an expert. The tokenizer was able to tokenize 95 orthographic verbs out of 111 tokens drawn from the initial test set and that contained more than one orthographic word. Their results proved that overall length of the input tokens improved the general tokenization. The overall F-score of the system was 0.93. The strength of the system was that it could tokenize more input words. The weakness of the system is that the morphological analyzer was underdeveloped and that it was unable to perform tokenization based on the context of the tokens.

A finite state morphological analyzer for the Ekegusii verbs was presented by (Elwell, 2008). The model is based on morphemes and implemented in Xerox finite state tools. All finite and non-finite forms were captured using only one regular expression. The morphosyntax of the verb is realized by specifying a set of morphemes that occupy each morpheme slot. The challenge with the system is that it poorly handled imbrication, which arises when there is a widened range of verbal roots. The evaluation results of the systems are unavailable.

A rule-based model for stemming Nepali text was presented by (Koirala & Shakya, 2020). A manually annotated corpus was extracted from online news portals. The corpus consisted of 4383 articles with 118,056 unique words. To classify news topics, 1400 news articles drawn from sports, global, politics, economy, literature, society and technology was extracted from Nepali news website and subdivided into 70% training set and 30% test set. The research concluded that stemmed classification outperformed the non-stemmed counterpart, with an F1-score difference of 0.02 and a significantly reduced vocabulary size of features.

b) Lemmatization

Machine Learning

Lematus, a system that context-based lemmatization using and encoder and decoder was presented by (Bergmanis & Goldwater, 2018). The system does not use a morphological tagger (Malaviya et al., 2019). The model is based on Nematus (Sennrich et al., 2017) neural machine translation toolkit. The benchmark systems for the experiment are Morfette (Chrupała et al., 2008), Lemming (Müller et al., 2015) and a context-sensitive lemmatizer based on two bidirectional gated recurrent neural networks (Chakrabarty et al., 2017). The dataset for the experiment was the Universal Dependency Treebank v2.0 dataset with 20 languages. The model achieves a 94.9% accuracy, outperforming the benchmarks with the closest model achieving 94.1%. However, a challenge with the model is that not relying on morphological tags make the system unrealistic as morpho-syntactic annotation must be available on corpora that have been annotated with token-level lemmata (Malaviya et al., 2019).

A contextual neural model for lemmatization was presented by (Malaviya et al., 2019). The model employs morphological tagging (assigning words their POS tags and more morphological information ((Yildiz & Tantuğ, 2019))) to provide the summary of the context of the word in the sentence. The input of the lemmatizer is the output from the morphological tagger. The Universal Dependencies Treebanks was the data source for the experiments. The lemmatizer is a 2-layer bidirectional LSTM encoder and a 1-layer bidirectional LSTM decoder consisting of 400 hidden units. The baseline systems for the experiment include: Lematus (Bergmanis & Goldwater, 2018), UD-Pipe (Straka & Straková, 2017), Lemming (Müller et al., 2015) and Morfette (Chrupała et al., 2008). The experiments show that morphological taggers improve the general performance of lemmatizers for the task of lemmatization. The overall accuracy of the proposed model is 95.42%, better than Lematus at about 95.05% with all models tested across 20 languages.

A sequence-to-sequence lemmatizer is presented by (Celano, 2020) for the closed EvaLatin shared task. The lemmatizer was implemented in Keras and training spanned 10 epochs. Lemmatization relied on POS tags generated from LightGBM. These POS tags serve to disambiguate word forms. The model's accuracy on the development set and test set is 99.82% and 97.63% respectively. This model, however, could not lemmatize Arabic numbers.

Rule based approaches

A rule based approach to Sinhala lemmatization is presented by (Nandathilaka et al., 2018). Their model relied on a POS tagger to detect the part of speech of a word before lemmatization was performed. Roots were manually annotated based on their role in the sentences. These sentences were derived from social media text. A total of 30 rules were created to guide lemmatization of nouns. The model was tested on 300 words obtained from Facebook, achieving an accuracy of 73.33%. The weaknesses of this model are that it did not rely on a formal lexicon and that its accuracy depended on how correctly the POS tagger was configured.

A lemmatizer for Gujarati text based on a stemmer is presented by (Patel & Patel, 2019). The dataset was manually created and contained both the stem and lemma of a word. The model allows new words to be added to the dictionary. Wrong stems can be manually handpicked and rectified. The model accurately performed stemming on 98.33% of the total 2097 words tested. 239 new words were added to the dictionary. The weakness of the system is that a derived stem could also be a lemma of another word bearing a different meaning. The model could also give erroneous output if a certain inflection is also a part of another totally different inflection since it uses shortest-affix-match technique to locate affixes. This makes only one inflection to be removed leaving a part of the other inflection unremoved since it has been distorted by the removal of the first, making this distorted inflection appear as part of the stem. The strength of the model lies in the predefined format of the vocabulary, which greatly boosts its results.

A rule-based lemmatizer or Kannada is proposed by (Prathibha & Padma, 2016). A manual dictionary of both verb and noun roots was created. The model relied on the longest-affix-match technique to locate affixes within a word. This lemmatizer automatically updates the dictionary with new lemma. Affixes are also manually collected. The weaknesses of the model are: errors could arise if affixes were absent from the vocabulary, rule violation, misspelt input, and overfitting arising from longest-affix-match. This model could also perform poorly on input with multiple suffixes. Since the model was tested on four datasets, it achieved an average overall accuracy of 93.50%.

A rule based lemmatizer for Punjabi is presented by (Puri, 2018). The model relies on the Synonym replacement algorithm to obtain the lemma of a word based on predefined rules. The model works by looking up the shortest synonym of an input word from a dictionary. The weakness of this model lies in its reliance to named entity recognition when lookup. The model also relies on a list of suffixes to guide it on what affixes to strip from a word. The model achieved an F score of 86 when tested on 10 articles containing a total of 3979 words. Other weaknesses of the model are that there were a few words in the database and that the model did not consider the context and part of speech of an input word.

c) Part-of-Speech (POS) tagging

Machine Learning

Bi-directional long short-term memory models have been used with traditional POS taggers across 22 languages and data sizes by (Plank et al., 2016). In their experiment, they used three taggers: the TNT, CRF tagger and the bidirectional LSTM tagger. The source of data was the Universal Dependencies project v1.2, with languages chosen being at least 60,000 tokens. The TNT performed better on the 22 languages than the CRF. However, the LSTM tagger performed better than the traditional taggers on 3 languages and RNNs. The multi-task bidirectional LSTM performed best on 12 languages and successfully predicted POS tags for out of vocabulary tokens. This was made possible by auxiliary loss function that enhanced the performance on rare words. However, the performance of the bi-LSTM is curtailed by the presence of noise, and more so, higher rates of noise.

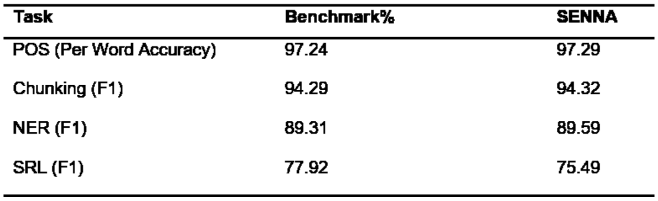
The Target Preserved Adversarial Neural Network (TPANN) to perform POS tagging for Twitter was presented by (Gui et al., 2017). The POS tagger is based on the bidirectional LSTM, an adversarial network and autoencoder. Their feature extraction component relies on a CNN for character embedding feature extraction. The POS tagger is a feed-forward classifier with a SoftMax layer. The datasets used to support the experiments ranged from labeled out-of-domain, labeled in-domain and unlabeled in-domain data. The out-of-domain data comprised of the Wall Street Journal data extracted from the Penn Treebank v3. This set was applied for training POS tagging. The labeled in-domain data was extracted from three benchmarks for comparison with their proposed method: RIT-Twitter, NPSCHAT, ARK-Twitter. This data was applied to further train and evaluate the POS tagger. The unlabeled data was constructed at a large scale from Twitter through its application programming interface. The model achieved 94.1% accuracy when evaluated on NPSChat, better than 90.8% accuracy achieved on a previous work. When evaluated on the RIT-Twitter, the model achieved 90.92% accuracy.

Transformation-based learning for chunking is applied by (Ramshaw & Marcus, 1999). The model applies Brill's POS tagger (Brill, 1992) to assign chunk tags to each word based on its POS tag. The experiments relied on data sourced from the Wall Street Journal section of the Penn Treebank. 50,000 words were used in each test set. The experiments subdivide chunking into two subtasks: the baseNP and the partitioning chunk tasks. The model proves that not relying on POS tags for chunking improved the baseNP subtask by 1% and the partitioning subtask by 5%, implying that the baseNP subtask is better favored by reference to actual words. Further, the models improved in accuracy if more words were supplied for training, achieving 90.5% and 83.5% precision on baseNP and partitioning subtasks respectively.

Semantic/Syntactic Extraction Using a Neural Network Architecture (SENNA), a model that relied on feed forward neural network and word embeddings for NLP tasks like POS tagging, NER, semantic role labelling and chunking was presented by (Collobert et al., 2011). The model achieved a best F1-score of 94.32%, 0.03% higher than the benchmark system.

An unsupervised algorithm to identify verb arguments with POS tagging as the only annotation requirement is proposed by (Abend et al., 2009). MXPOST (Ratnaparkhi, 1996) and decision tree-based tagger (Schmid, 1994) were used to extract POS tags for English and Spanish respectively. The experiment sourced data from the PropBank English corpus. Training data consisted of 207 sentences with 132 distinct verbs. The test data comprised of 6007 sentences and 1008 distinct verbs. The Spanish branch of the experiment sourced data from Spanish Wikipedia, resulting in 200 sentences with 313 verb instances for training and 848 sentences with 1279 verb instances for testing. On English test data, the model achieved an F1 score of 59.14% when using clause detection compared to the 57.35% score of the baseline system. On Spanish, the model achieved an F1 score of 23.87% when using collocation maximum F-score compared to 21.62% score of the baseline system.

SENNA, a model that relied on feed forward neural network and word embeddings for NLP tasks like POS tagging, NER, semantic role labelling and chunking was proposed by (Collobert et al., 2011). The model achieved the following results:



Task	Benchmark%	SENNA
POS (Per Word Accuracy)	97.24	97.29
Chunking (F1)	94.29	94.32
NER (F1)	89.31	89.59
SRL (F1)	77.92	75.49

Table 1: *SENNA-s performance on POS, Chunking, NER, and SRL*

Statistical approach

A POS tagger that incorporates Hidden Markov models and the Unigram was presented by (Tukur et al., 2019). The purpose of HMMs is to assign tags to a sentence based on its context, while the Unigram assigns POS tags on a per word basis. The sentences in the corpus were split into bigrams using a Hidden Markov Model-based sentence analyzer. The system accurately tagged 20% of the words in the corpus. This figure is considerably low considering that the corpus had words drawn from a native website. However, this is commendable as it is the first POS tagger for the Hausa language.

A technique for tagging parts of speech in Hausa using Hidden Markov Models was proposed by (Tukur et al., 2020). A corpus of Hausa words was used as the knowledge resource. The performance of the system was tested using 187 Hausa words that were presented to a Hausa expert for verification. The system accurately tagged 76.795% of the words. This is better than the 20% accuracy achieved in the first POS tagger for Hausa language (Tukur et al., 2019). The challenge the experiment faced was that the corpus lacked enough words and that it couldn't correctly tag all the words, with conjunctions being the least correctly tagged at 50%.

A POS tagger based on Hidden Markov Models, implementing he Viterbi algorithm for optimization was developed by (Mamo & Meshesha, 2011). To analyze the performance of the model, the corpus was divided into nine folds for training and the remaining onefold for testing. Each test set contained about 146 words. The bigram algorithm recorded a 91.97% accuracy while the unigram algorithm correctly tagged 87.5% of the words.

Conditional random field are applied to capture code switched pattern sequences to tag words extracted from social media text with accurate POS information (Ghosh et al., 2016). The targeted code-switched languages are Bengali, Hindi and Tamil on mostly-English text. The dataset comprised of utterances from each of the languages to English, resulting in a total of 44,908 utterances for training and 27,028 utterances for testing. POS tagging was performed using two taggers: first the Stanford POS tagger, and later the Conditional Random (CRF) Field tagger for language identification. Compared to the Stanford model, the CRF performed better on code switches from each language to English. The CRF model achieved the following accuracies for code-switches to English: Bengali = 75.22%, Hindi = 73.2%, Tamil = 64.83%.

A POS tagger for Bengali using CRF is proposed by (Ekbal, 2007). 26 POS tags were used for the experiment. The CRF method was chosen as it worked better than Hidden Markov model (HMM) for languages that lack large annotated corpora. The POS tagger comprised of context word features, word suffix, word prefix, and named entity recognition. The model was trained using 72,341 words, with the training corpus sourced from NLPAI_Contest06 and SPSAL2007 data. 20,000 wordforms were presented to the tagger during testing. The model recorded 86.4% accuracy when the bare CRF was used alone. However, the accuracy improved to 90.3% when the CRF was combined with Named Entity Recognition (NER), the Bengali lexicon and unknown words. This CRF model thus achieves better results than (Ghosh et al., 2016), even though their CRF was meant for a different purpose.

Rule based approaches

A POS tagger is incorporated in a grammar checker for Oromo by (Tesfaye, 2011). The grammar checker is rule based, with 123 rules initially in place to enhance its function. The model achieved 88.89% accuracy from thesis text belonging to students. However, the model faced the following challenges: incorrect word stems, assignment of wrong POS tags and that rules were few.

Hardware accelerators are implemented to improve POS tagging by converting rules to regular expressions (Sadredini et al., 2018). POS tagging rules are trained through the Brill's tagger. Training data is sourced from the Penn Treebank and Brown corpora. Since rules entered in these hardware accelerators are learned from the result of performing POS tagging on other systems, it is not guaranteed that the output will be accurate for certain rules. The strength of the system is that since POS tagging is performed on hardware accelerators, the overall tagging is faster. This is because these hardware accelerators can process more than one rule at a time. This results in recorded improvement in performance 2600 and 1914 times for the Automata Processor and Field Programmable Gate Arrays hardware accelerators respectively.

An Indonesian POS tagger is presented by (Purnamasari & Suwardi, 2018). This model relies on a dictionary to lookup tokenized input before performing stemming of the input. POS tags on input are determined by a matching entry's POS tag in the dictionary. This model achieves an average accuracy of 87.4% on an average of 2099 words. The strength of this system lies in its lack of hand engineered morphological rules. However, the system performs poorly when input belongs to more than one part of speech.

A Welsh POS tagger is presented in (Neale et al., 2019). This POS tagger is based on a dictionary and requires few rules and annotated data. The training set for the model was a corpus comprising 611 sentences. The tagger achieved an accuracy of 94.5% on 14,876 tokens contained in the corpus. The tagger could not perform well for words belonging to more than one part of speech.

POS tagging of Romanized Sindhi words performed using an online Python program is presented in (Sodhar et al., 2019). The model was tested on 352 words extracted from 100 Romanized Sindhi. The model correctly tagged 309 words (87.78%). The algorithm is brute force, evidenced by its approach in assigning a POS tag whereby each input must be assigned a tag regardless. The data set for this experiment was rather small.

References

Abend, O., Reichart, R., & Rappoport, A. (2009). Unsupervised argument identification for Semantic Role Labeling. ACL-IJCNLP 2009 - Joint Conf. of the 47th Annual Meeting of the Association for Computational Linguistics and 4th Int. Joint Conf. on Natural Language Processing of the AFNLP, Proceedings of the Conf. https://doi.org/10.3115/1687878.1687884

Almuhareb, A., Alsanie, W., & Al-Thubaity, A. (2019). Arabic Word Segmentation With Long Short-Term Memory Neural Networks and Word Embedding. IEEE Access. https://doi.org/10.1109/ACCESS.2019.2893460

Ansari, E., Žabokrtský, Z., Mahmoudi, M., Haghdoost, H., & Vidra, J. (2019). Supervised morphological segmentation using rich annotated lexicon. International Conference Recent Advances in Natural Language Processing, RANLP. https://doi.org/10.26615/978-954-452-056-4_007

Ayana, A. (2015). Improving Brill's tagger lexical and transformation rule for Afaan Oromo language. PeerJ PrePrints, 3, 1–11. https://doi.org/10.7287/peerj.preprints.1225

Bergmanis, T., & Goldwater, S. (2018). Context sensitive neural lemmatization with lematus. NAACL HLT 2018 - 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies - Proceedings of the Conference. https://doi.org/10.18653/v1/n18-1126

Botha, J. A., & Blunsom, P. (2014). Compositional morphology for word representations and language modelling. 31st International Conference on Machine Learning, ICML 2014.

Brill, E. (1992). A simple rule-based part of speech tagger. https://doi.org/10.3115/974499.974526

Celano, G. G. A. (2020). A Gradient Boosting-{S}eq2{S}eq System for {L}atin {POS} Tagging and Lemmatization. Proceedings of LT4HALA 2020 - 1st Workshop on Language Technologies for Historical and Ancient Languages.

Chahuneau, V., Schlinger, E., Smith, N. A., & Dyer, C. (2013). Translating into morphologically rich languages with synthetic phrases. EMNLP 2013 - 2013 Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference.

Chakrabarty, A., Pandit, O. A., & Garain, U. (2017). Context sensitive lemmatization using two successive bidirectional gated recurrent networks. ACL 2017 - 55th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference (Long Papers). https://doi.org/10.18653/v1/P17-1136

Chary, M., Parikh, S., Manini, A. F., Boyer, E. W., & Radeos, M. (2019). A review of natural language processing in medical education. Western Journal of Emergency Medicine. https://doi.org/10.5811/westjem.2018.11.39725

Chiang, D. (2007). Hierarchical phrase-based translation. Computational Linguistics. https://doi.org/10.1162/coli.2007.33.2.201

Chrupała, G., Dinu, G., & van Genabith, J. (2008). Learning morphology with Morfette. Proceedings of the 6th International Conference on Language Resources and Evaluation, LREC 2008.

Collobert, Ronan, Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., & Kuksa, P. (2011). Natural language processing (almost) from scratch. Journal of Machine Learning Research.

Creutz, M., & Lagus, K. (2007). Unsupervised models for morpheme segmentation and morphology learning. ACM Transactions on Speech and Language Processing. https://doi.org/10.1145/1217098.1217101

Ding, C., Aye, H. T. Z., Pa, W. P., Nwet, K. T., Soe, K. M., Utiyama, M., & Sumita, E. (2019). Towards Burmese (Myanmar) morphological analysis: Syllable-based Tokenization and Part-of-speech Tagging. ACM Transactions on Asian and Low-Resource Language Information Processing. https://doi.org/10.1145/3325885

Ekbal, A. (2007). Bengali Part of Speech Tagging using Conditional Random Field. Proceedings of Seventh ….

Elwell, R. (2008). Finite State Methods for Bantu Verb Morphology. Computational Linguistics for Less-Studied Languages, X, 56–67.

Eskander, R., Callejas, F., Nichols, E., Klavans, J., & Muresan, S. (2020). MorphAGram: Evaluation and Framework for Unsupervised Morphological Segmentation. Aclweb.Org.

Eskander, R., Rambow, O., & Yang, T. (2016). Extending the use of adaptor grammars for unsupervised morphological segmentation of unseen languages. COLING 2016 - 26th International Conference on Computational Linguistics, Proceedings of COLING 2016: Technical Papers.

Fridah, K., & Thomas, H. (2010). Finite State Methods in Morphological Analysis of Runyakitara Verbs. Nordic Journal of African Studies.

Ghosh, S., Ghosh, S., & Das, D. (2016). Part-of-speech Tagging of Code-Mixed Social Media Text. https://doi.org/10.18653/v1/w16-5811

Gui, Tao, Zhang, Q., Huang, H., Peng, M., & Huang, X. (2017). Part-of-speech tagging for twitter with adversarial neural networks. EMNLP 2017 - Conference on Empirical Methods in Natural Language Processing, Proceedings. https://doi.org/10.18653/v1/d17-1256

Ingólfsdóttir, S. L., Loftsson, H., Daðason, J. F., & Bjarnadóttir, K. (2019). Nefnir: A high accuracy lemmatizer for Icelandic. http://arxiv.org/abs/1907.11907

Kann, K., Mager, M., Meza-Ruiz, I., & Schütze, H. (2018). Fortification of neural morphological segmentation models for polysynthetic minimal-resource languages. NAACL HLT 2018 - 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies - Proceedings of the Conference. https://doi.org/10.18653/v1/n18-1005

Katushemererwe, F., & Issue, S. (2010). Fsm2 and the Morphological Analysis of Bantu Nouns – First Experiences from Runyakitara. 4(1), 58–69.

Koehn, P., Zens, R., Dyer, C., Bojar, O., Constantin, A., Herbst, E., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., & Moran, C. (2007). Moses: open source toolkit for statistical machine translation. Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions - ACL '07. https://doi.org/10.3115/1557769.1557821

Koirala, P., & Shakya, A. (2020). A Nepali Rule Based Stemmer and its performance on different NLP applications. http://arxiv.org/abs/2002.09901

Kudo, Taku, & Matsumoto, Y. (2002). Chunking with Support Vector Machines. Journal of Natural Language Processing. https://doi.org/10.5715/jnlp.9.5_3

Liu, Su, X., Zhang, H., Gao, G., & Bao, F. (2021). Incorporating Inner-word and Out-word Features for Mongolian Morphological Segmentation. https://doi.org/10.18653/v1/2020.coling-main.408

Malaviya, C., Wu, S., & Cotterell, R. (2019). A simple joint model for improved contextual neural lemmatization. NAACL HLT 2019 - 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies - Proceedings of the Conference.

Mamo, G., & Meshesha, M. (2011). Parts of Speech Tagging for Afaan Oromo. International Journal of Advanced Computer Science and Applications. https://doi.org/10.14569/specialissue.2011.010301

Mott, J., Bies, A., Strassel, S., Kodner, J., Richter, C., Xu, H., & Marcus, M. (2020). Morphological Segmentation for Low Resource Languages. May, 3996–4002.

Müller, T., Cotterell, R., Fraser, A., & Schütze, H. (2015). Joint lemmatization and morphological tagging with LEMMING. Conference Proceedings - EMNLP 2015: Conference on Empirical Methods in

Natural Language Processing. https://doi.org/10.18653/v1/d15-1272

Nakagawa, T. (2004). Chinese and Japanese word segmentation using word-level and character-level information. https://doi.org/10.3115/1220355.1220422

Nandathilaka, M., Ahangama, S., & Thilini Weerasuriya, G. (2018). A Rule-based Lemmatizing Approach for Sinhala Language. 2018 3rd International Conference on Information Technology Research, ICITR 2018. https://doi.org/10.1109/ICITR.2018.8736134

Narasimhan, K., Barzilay, R., & Jaakkola, T. (2015). An Unsupervised Method for Uncovering Morphological Chains. Transactions of the Association for Computational Linguistics. https://doi.org/10.1162/tacl_a_00130

Narasimhan, K., Karakos, D., Schwartz, R., Tsakalidis, S., & Barzilay, R. (2014). Morphological segmentation for keyword spotting. EMNLP 2014 - 2014 Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference. https://doi.org/10.3115/v1/d14-1095

Neale, S., Donnelly, K., Watkins, G., & Knight, D. (2019). Leveraging lexical resources and constraint grammar for rule-based part-of-speech tagging in Welsh. LREC 2018 - 11th International Conference on Language Resources and Evaluation.

Patel, H., & Patel, B. (2019). Stemmatizer—Stemmer-based Lemmatizer for Gujarati Text. Advances in Intelligent Systems and Computing. https://doi.org/10.1007/978-981-13-2285-3_78

Pauw, G., & de Schryver, G. M. (2008). Improving the computational morphological analysis of a Swahili corpus for lexicographic purposes. Lexikos.

Plank, B., Søgaard, A., & Goldberg, Y. (2016). Multilingual part-of-speech tagging with bidirectional long short-term memory models and auxiliary loss. 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016 - Short Papers. https://doi.org/10.18653/v1/p16-2067

Prathibha, R. J., & Padma, M. C. (2016). Design of rule based lemmatizer for Kannada inflectional words. 2015 International Conference on Emerging Research in Electronics, Computer Science and Technology, ICERECT 2015. https://doi.org/10.1109/ERECT.2015.7499024

Premjith, B., Soman, K. P., & Kumar, M. A. (2018). A deep learning approach for Malayalam morphological analysis at character level. Procedia Computer Science. https://doi.org/10.1016/j.procs.2018.05.058

Pretorius, & Bosch, S. E. (2003). Finite-state computational morphology: An analyzer prototype for Zulu. In Machine Translation. https://doi.org/10.1007/s10590-004-2477-4

Pretorius, & Pretorius, L. (2009). Setswana Tokenisation and Computational Verb Morphology : Facing the Challenge of a Disjunctive Orthography. Computational Linguistics.

Puri, R. (2018). A Rule based approach for lemmatisation of Punjabi text Documents. 27(63019), 216–224.

Purnamasari, K. K., & Suwardi, I. S. (2018). Rule-based Part of Speech Tagger for Indonesian Language. IOP Conference Series: Materials Science and Engineering. https://doi.org/10.1088/1757-899X/407/1/012151

Ramshaw, L. A., & Marcus, M. P. (1999). Text Chunking Using Transformation-Based Learning. https://doi.org/10.1007/978-94-017-2390-9_10

Ratnaparkhi, A. (1996). A Maximum Entropy Model for Part-of-Speech Tagging. In Proceedings of the Conference on Empirical Methods in Natural Language Processing.

Sadia, R., Rahman, M. A., & Seddiqui, M. H. (2019). N-gram Statistical Stemmer for Bangla Corpus. 2–6. http://arxiv.org/abs/1912.11612

Sadredini, E., Guo, D., Bo, C., Rahimi, R., Skadron, K., & Wang, H. (2018). A scalable solution for rule-based part-of-speech tagging on novel hardware accelerators. Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. https://doi.org/10.1145/3219819.3219889

Sánchez, Cartagena, V. M., & Toral, A. (2016). AbuMaTran at WMT 2016 Translation Task: Deep Learning, Morphological Segmentation and Tuning on Character Sequences. https://doi.org/10.18653/v1/w16-2322

Schmid, H. (1994). Probabilistic Part-of-Speech Tagging Using Decision Trees. Proceedings of the International Conference on New Methods in Language Processing.

Sennrich, R., Firat, O., Cho, K., Birch, A., Haddow, B., Hitschler, J., Junczys-Dowmunt, M., Läubli, S., Barone, A. V. M., Mokry, J., & Nădejde, M. (2017). Nematus: A toolkit for neural machine translation. 15th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2017 - Proceedings of the Software Demonstrations. https://doi.org/10.18653/v1/e17-3017

Sodhar, I. N., Jalbani, A. H., Channa, M. I., & Hakro, D. N. (2019). Parts of Speech Tagging of Romanized Sindhi Text by applying Rule Based Model. 19(11), 91–96. https://doi.org/10.13140/RG.2.2.35194.03524

Straka, M., & Straková, J. (2017). Tokenizing, POS tagging, lemmatizing and parsing UD 2.0 with UDPipe. CoNLL 2017 - SIGNLL Conference on Computational Natural Language Learning, Proceedings of the

CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies. https://doi.org/10.18653/v1/k17-3009

Tedla, Y., & Yamamoto, K. (2018). Morphological Segmentation with LSTM Neural Networks for Tigrinya. International Journal on Natural Language Computing. https://doi.org/10.5121/ijnlc.2018.7203

Tesfaye, D. (2011). A rule-based Afan Oromo Grammar Checker. International Journal of Advanced Computer Science and Applications, 2(8). https://doi.org/10.14569/ijacsa.2011.020823

Tukur, A., Umar, K., & Muhammad, A. S. (2019). Tagging part of speech in hausa sentences. 2019 15th International Conference on Electronics, Computer and Computation, ICECCO 2019, Icecco. https://doi.org/10.1109/ICECCO48375.2019.9043198

Tukur, A., Umar, K., & Sa, A. (2020). Parts-of-Speech Tagging of Hausa-Based Texts Using Hidden Markov Model. 6(2), 303–313.

Wang, Fam, R., Bao, F., Lepage, Y., & Gao, G. (2019). Neural Morphological Segmentation Model for Mongolian. Proceedings of the International Joint Conference on Neural Networks. https://doi.org/10.1109/IJCNN.2019.8852050

Xu, H., Marcus, M., Yang, C., & Ungar, L. (2018). Unsupervised morphology learning with statistical paradigms. Proceedings of COLING 2018, the 27th International Conference on Computational Linguistics.

Xue, N. (2003). Chinese Word Segmentation as Character Tagging. Computational Linguistics.

Yambao, & Cheng, C. (2020). Feedforward Approach to Sequential Morphological Analysis in the Tagalog Language. 2020 International Conference on Asian Language Processing, IALP 2020. https://doi.org/10.1109/IALP51396.2020.9310516

Yang, Y., Li, S., Zhang, Y., & Zhang, H. P. (2019). Point the Point: Uyghur Morphological Segmentation Using PointerNetwork with GRU. Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). https://doi.org/10.1007/978-3-030-32381-3_30

Yildiz, E., & Tantuğ, A. C. (2019). Morpheus: A Neural Network for Jointly Learning Contextual Lemmatization and Morphological Tagging. https://doi.org/10.18653/v1/w19-4205

Zhai, F., Potdar, S., Xiang, B., & Zhou, B. (2017). Neural models for sequence chunking. 31st AAAI Conference on Artificial Intelligence, AAAI 2017.