Report NLP Assignment-1

Akash Manna, 2024801008

February 2026

1 Task-1

Table 1: Analysis of Tokenization Methods: English

Lang	Tokenizer	Category	Example Snippet	Justification
En	Whitespace	Sensible	[’easily’, ’obtained’, ’online’, ...], [’The’, ’following’, ’steps’, ... ’success’, ’.’], [’Let’, ’us’, ’look’,... ’affair’, ’.’]	Clean splitting on spaces; preserves whole words and semantic integrity.
	Whitespace	Not Sensible	[’Saturday’, ’Dec’, ’1’, ’–’, ’10’, ...], [’”’, ’Fisherman’,... ’ass’, ’-’, ’-’, ’stand’, ’back’, ’!’, ...], [’L’, ’.’, ’is’,..., ’is’]	Poor handling of dates and time ranges; symbols like ‘–‘ are isolated excessively.
En	Regex	Sensible	[”I’d”, ’write’, ’them’, ...], [’Close’, ’to’, ’this’, ’horn’, ...] [’The’, ’card’, ’is’, ’to’, ...]	Semantics preserved; punctuation like apostrophes handled within context effectively.
	Regex	Not Sensible	[’A’, ’testamur’, ’(’,], [’This’, ’feature’, ... ’$’, ’500’, ’to’, ’$’, ’1’, ’,’, ’500’, ’.’], [’@’, ’proseimprint’, ’@’, ...]	Over-segmentation of currency (‘$‘, ‘500‘) and punctuation clusters; breaks logical entities.
En	BPE	Sensible	[’e’, ’at’, ’en</w>’, ’h’, ...], [’N’, ’a’, ..., ’the</w>’] [’un’, ... ’been</w>’, ’pri’, ’v’, ’il’, ’e’, ’g’, ’ed</w>’, ’to</w>’].	(Per input label): Functional, but exhibits extreme over-segmentation bordering on character-level tokens. ”previleged” being 5 pieces
	BPE	Not Sensible	[’str’, ’u’, ’gg’, ’le,</w>’, ...], [’j’, ’us’, ’ti’, ’fi’, ’ab’, ’le,</w>’,...], [’re’, ’li’,’gi’, ’ous’, ...]	Inconsistent fragmentation; common roots (‘struggle‘) are broken into non-semantic chunks.

A	B	C	D	E
Language	Tokenizer	Category	Example	Justification
Mn	WhiteSpace	Sensible	['БА', '-', '8', '.', '1', 'Барилга', 'байгууламжийн', 'засвар', ',', 'засал', 'чимэглэл', ',', 'интерьер', ',', 'орчны', 'ажил'] ['Урам', 'зориг', 'орохын', 'тулд', 'цоглог', 'хөгжим', 'сонс', '.', 'Аль', 'болох', 'бөмбөрийн', 'хэмнэл', 'түргэн', 'хүчтэй', 'беаттэй', 'дуу', 'сонс', '.'] ['Шинжлэх', 'ухаан']	Correctly identifies full words in Cyrillic; maintains semantic units
Mn	WhiteSpace	Not Sensible	['2017', 'оны', '10', 'сарын', '22', '-', '23', 'өдөр', 'Увс', 'аймгийн', 'Улаангом', 'хотод', 'зохион', 'байгуулагдсан', '""', 'ДЭЛХИЙН', 'ЦООХОР', 'ИРВЭС', 'ХАМГААЛАХ', 'ӨДӨРТ', '""', 'манай', 'аймгийн', 'ЕБС', '-', 'ын', 'ЭКО', 'клубын', '60', 'сурагч', '6', 'багш', 'нарын', 'хамт', '""', 'Ирвэсний', 'эх', 'нутаг', '2017', '""', 'нэрийн', 'доор', 'оролцож', 'багаараа', '2', '-', 'р', 'байр', 'эзэлэж', ',', 'дараагийн', 'жил', 'манай', 'аймагт', 'зохион', 'байгуулахаар', 'болов', '.'](Over fragmentation) ['-', 'Идэвхжүүлэлт', '-', 'Паал', 'Морк', 'хэвлэл', 'мэдээллийн', 'хэрэгслэлээр', 'суртачлах', ',', 'зар', 'сурталчилгаа', ',', 'веб', 'сайт', 'болон', 'олон', 'нийтийн', 'мэдээллийн', 'хэрэгсэл', ',', 'идэвхжүүлэлтийн', 'арга', 'хэрэгслийг', 'хэрхэн', 'ашиглах', 'дадлага', 'ажил'] ['УИХ', '-', 'ын', 'гишүүн', 'Л', '.', 'Энхболдтой', 'холбоотой', 'гурван', 'ч', 'компани', 'ЖДҮХС', '-', 'гаас', 'зээл', 'авчээ']("L" character and "." are seperated.)	Fails on suffixes and acronyms containing hyphens, splitting them into meaningless distinct tokens.
Mn	Regex	Sensible	['Цагдаагийн', 'ажилтнууд', 'статистик', ',', 'мэдээллийн', 'сургалт', 'хийж', 'байна'] ['Х', '.', 'Чойбалсан', 'нэг', 'удаа', 'мэргэн', 'хүнээс', 'хувь', 'заяаныхаа', 'төөргийг', 'асуухдаа', 'энхрий', 'хайрт', 'намынхаа', 'ирээдүйг', 'ч', 'бас', 'сонирхсон', 'гэдэг', '.', 'Өнөө', 'хүн', '""', 'Энэ', 'нам', 'чинь', '90', 'настай', 'л', 'юм', 'байна', '""', 'гэж', 'хэлж', 'байсан', 'гэнэ', 'лээ', '.'] ['Зөөврийн', 'тоглоомын', 'талбай']	Standard sentence structure is maintained; words are kept intact
Mn	Regex	Not Sensible	['Билгийн', 'тооллын', 'XVII', 'жарны', '""', 'Тийн', 'унжлагат', '""', 'шороон', 'нохой', 'жилийн', 'хаврын', 'тэргүүн', 'сарын', 'шинийн', 'нэгний', 'өглөөний', '09', ':', '00', 'цагт', 'төрийн', 'золголт', 'эхэлнэ', '.', 'Төрийн', 'ордон', 'дахь', 'Төрийн', 'ёслол', 'хүндэтгэлийн', 'гэрт', 'УИХ', '-', 'ын', 'дарга', ',', 'Ерөнхий', 'сайд', 'нар', 'Ерөнхийлөгчид', 'золгох', 'юм', '.']("\"" are seperate.) ['Ерөнхий', 'сайд', 'У', '.', 'Хүрэлсүхийг', 'огцруулах', 'саналыг', 'албан', 'ёсоор', 'өргөн', 'барилаа'](Single chars) ['Post', 'subject', ':', 'Re', ':', 'Монголчуудын', 'овогуудын', 'угсаа', 'гарал'](Both English and Monglean mixed.)	Incorrectly splits initials and honorifics; separates quotation marks (`"`) into isolated tokens.
Mn	BPE	Sensible	['З', 'амын</w>', 'хөдөл', 'г', 'өө', 'ний</w>', 'ши', 'нэ', 'чил', 'сэн</w>', 'най', 'р', 'уул', 'га</w>', 'ир', 'эх</w>', 'сарын</w>', '1', '-', 'нээс</w>', 'м', 'өр', 'д', 'ө', 'ж</w>', 'эхэл', 'нэ.</w>', 'Ж', 'ил', 'д</w>', 'үйл', 'дэг', 'дэ', 'ж</w>', 'буй</w>', '5', '0', '00</w>', 'гаруй</w>', 'ос', 'ол</w>', 'түү', 'ний</w>', 'ул', 'м', 'аас</w>', 'ам', 'и', 'а</w>', 'ал', 'даж</w>', 'буй</w>', '6', '00', '-', '6', '5', '0</w>', 'хүн</w>', 'д', 'үр', 'мийн</w>', 'бу', 'с</w>', 'үйл</w>', 'ажиллагааны</w>', 'үр</w>', 'даг', 'ав', 'ар</w>', 'тул', '</w>', 'з', 'айл', 'ш', 'гүй</w>', 'м', 'өр', 'д', 'өх', 'өөс</w>', 'өөр</w>', 'ар', 'г', 'аг', 'үй', '.</w>'](IDK) ['10', '.</w>', 'Ж', 'аг', 'с', 'аал', 'т</w>', 'бат', 'лах</w>', 'тухай</w>', 'Б', '.Б', 'А', 'Т', 'З', 'О', 'Р', 'И', 'Г</w>'](Random) ['7', '0</w>', 'дугаар</w>', 'цэ', 'цэр', 'лэг</w>'](['7', '0', 'number', 'number', 'number', 'number'])	Sub-word units are generated, though the first character is strangely isolated.
Mn	BPE	Not Sensible	['Т', 'үү', 'н', 'ч', 'лэн</w>', '19', '9', '9</w>', 'онд</w>', 'Г', 'алын</w>', 'аюул', 'гүй</w>', 'байдл', 'ын</w>', 'хуулийн</w>', 'төс', 'лийг</w>', 'боловср', 'уулан</w>', 'УИХ', '-', 'аар</w>', 'бат', 'л', 'уул', 'сн', 'аар</w>', 'Г', 'ал</w>', 'т', 'үй', 'м', 'эр', 'тэй</w>', 'тэм', 'цэ', 'х</w>', 'байгууллаг', 'ын</w>', 'үйл</w>', 'ажиллагааны</w>', 'эрх</w>', 'зүйн</w>', 'ор', 'чин</w>', 'бүр', 'дэ', 'ж,</w>', 'о', 'й</w>', 'х', 'ээ', 'рийн</w>', 'болон</w>', 'о', 'б', 'ъ', 'ек', 'тын</w>', 'т', 'үй', 'м', 'р', 'ээс</w>', 'ур', 'ь', 'дчил', 'ан</w>', 'сэр', 'г', 'ий', 'лэ', 'х,</w>', 'ун', 'тр', 'аа', 'х,</w>', 'ш', 'алт', 'гаа', 'н</w>', 'нөхц', 'өл', 'ийг</w>', 'тог', 'тоо', 'х,</w>', 'г', 'ал</w>', 'т', 'үй', 'м', 'рийн</w>', 'хэрэг</w>', 'бүр', 'тгэ', 'х</w>', 'аж', 'лыг</w>', 'хариуц', 'ан</w>', 'Х', 'уул', 'ь</w>', 'зүйн</w>', 'сай', 'дын</w>', 'эрх', 'лэх</w>', 'асуудл', 'ын</w>', 'хүрээ', 'нд</w>', 'Засгийн</w>', 'газрын</w>', 'хэрэгж', 'үүлэ', 'гч</w>', 'аг', 'ен', 'тл', 'аг</w>', 'Г', 'ал</w>', 'т', 'үй', 'м', 'эр', 'тэй</w>', 'тэм', 'цэ', 'х</w>', 'газар</w>', 'бол', 'г', 'он</w>', 'зохион</w>', 'байгуул', 'жээ.</w>'](Over fragmentation) ['М', 'ар', 'к', 'е', 'т', 'ин', 'г,</w>', 'судал', 'гааны</w>', 'мэргэ', 'жил', 'тэ', 'н</w>'] ['Ч', 'е', 'хийн</w>', 'ви', 'з</w>', 'мэд', 'үүл', 'гийн</w>', 'төв', '</w>', 'Улаанбаатар</w>', 'хо', 'то', 'д</w>', 'н', 'ээ', 'г', 'дл', 'ээ</w>', '1', '9</w>', 'цагийн</w>', 'өмнө</w>'](Whole sentence doesn't make any sense).	Extreme over-fragmentation; almost every character is a token, defeating the purpose of BPE compression.

Figure 1: Analysis of Tokenization Methods: Mongolian

2 Task-2: Language models

2.1 Perplexity Score

Table 2: Perplexity score of 9 Languagu models

Tokenizer	Smoothing	Perplexity Score
Whitespace	None	66771.900
	Witten-Bell	3911.900
	Kneser-Ney	37846.629
Regex	None	72964.946
	Witten-Bell	4619.654
	Kneser-Ney	43083.070
BPE	None	33893.713
	Witten-Bell	278.490
	Kneser-Ney	9463.389

2.2 Comparative study

- Smoothing Impact: None → MLE (crashes on unseen) → Witten–Bell (light discount) → Kneser–Ney (continuation-aware, best).
- Tokenization Impact: Whitespace (coarse, fast) → Regex (syntax-aware) → BPE (morphological, generalizable).
- Best Overall: BPE + Kneser–Ney (lowest perplexity, fluent, robust). Fastest Baseline: Whitespace + None (but poor quality).
- Syntax-Critical: Regex + Witten-Bell (dialogue, punctuated text).

2.3 Examples:

Table 3: Comparison of Language Models: Tokenization, Smoothing, and Performance

Tokenization	Smoothing	Sparsity Handling	Key Strength	Key Weakness
Whitespace	None (MLE)	Zero-probability crashes	Simplicity, speed	Repetition on unseen n-grams
	Witten–Bell	Reserves mass for novel events	Better generalization	Still favors seen n-grams
	Kneser–Ney	Continuation probability	Semantic coherence, lowest perplexity	Computational cost
Regex	None (MLE)	Zero-probability crashes	Syntax preservation	Extreme sparsity, broken structure
	Witten–Bell	Reserves mass for punct. pairs	Syntax-aware smoothing	Moderate perplexity
	Kneser–Ney	Continuation-aware	Best syntax handling	Slowest training
BPE	None (MLE)	Morphological fallback	Handles novel words	Overfits subword patterns
	Witten–Bell	Subword generalization + smoothing	Morphological robustness	Moderate for unseen forms
	Kneser–Ney	Continuation + subword	Best overall: low perplexity, fluent	Highest complexity

Table 4: Analysis of Language Model Failure Modes and Root Causes

Model	Primary Failure Mode	Root Cause
WS + None	Repetition loops on rare contexts	Zero-probability MLE crash $\rightarrow$ argmax locks on high-freq token
WS + Witten-Bell	Still vulnerable to sparse n-grams	Discount $T/(count + T)$ is too weak ($\sim$10–20% reserved mass)
WS + Kneser-Ney	Only fails on unseen context diversity	Continuation probability cannot invent contexts not present in training data
Regex + None	CATASTROPHIC punctuation repetition	Explicit tokenization $\rightarrow$ sparse (word, punct.) pairs $\rightarrow$ MLE collapse
Regex + Witten-Bell	Punctuation context bias remains	Witten-Bell discount treats word-word and word-punct. pairs equally
Regex + Kneser-Ney	Nearly never fails (except ultra-rare cases)	Continuation probability deprioritizes punctuation; semantic recovery via backoff
BPE + None	Rare subword n-gram crashes	BPE reduces OOV (Out-Of-Vocabulary) but does not fix zero-probability n-grams
BPE + Witten-Bell	Subword fragmentation drift	Merges can vary; context learned on different fragmentation leads to mismatch
BPE + Kneser-Ney	Minimal failure (only cross-domain/adversarial)	Morphology + continuation probability + recursive backoff = trifecta defense

Table 5: Scenarios where the model performs well

Model	Best Scenario
WS + None	Exact phrase match present in training data
WS + Witten-Bell	Moderate-frequency phrases in a balanced domain
WS + Kneser-Ney	Any standard input, handling both rare and common contexts effectively
Regex + None	Unpunctuated simple phrases (behaves like whitespace)
Regex + Witten-Bell	Semi-rare phrases without heavy punctuation
Regex + Kneser-Ney	Punctuation-heavy phrases (e.g., titles, lists)
BPE + None	Morphologically common words where subwords are frequent
BPE + Witten-Bell	Rare words with clear morphological structure (prefixes/suffixes)
BPE + Kneser-Ney	Any input (rare, punctuation-heavy, or Out-Of-Vocabulary)