PART I: Study of Corpus

Objective: Research and analyze three corpora (Brown, Susanne, and Penn Treebank) based on publicly available descriptions.

Brown Corpus

The Brown Corpus is one of the first electronic corpora of modern English. It contains approximately one million words of American English text, compiled from works published in 1961 [1]. It uses a set of lexical labels (POS) that has been widely adopted and adapted by other projects.

Susanne Corpus

The corpus SUSANNE (Surface and Underlying Structural ANalyses of Naturalistic English) is an annotated version of a portion of the Brown Corpus. It offers detailed syntactic tagging in addition to lexical tagging [2]. Although it is smaller than the Brown Corpus, it provides a more in-depth linguistic analysis.

Penn Treebank

The Penn Treebank is a corpus of American English with syntactic and speech annotations. It is known for its use in the development and evaluation of parsers. It contains approximately 4.5 million words of English text, including the Brown Corpus and other texts [3].

Comparative

Aspect	Brown Corpus	Susanne Corpus	Penn Treebank
Type of labeling	Lexicon (POS)	Lexicon and syntactic	Lexicon and syntactic
Corpus size	~1 million words (500 files of ~2000 words)	64 files of ~2000 words	~4.5 million words
Label Set Size	87 POS Labels	Extensive set of labels	36 POS labels
Topics included	Varied (15 genres)	4 genre (Brown A,G,J,N)	Varied, including financial news
Origin of the texts	Works published in 1961	Works published in 1961	Texts of Brown, Wall Street Journal, etc.

Brown vs Susanne Comparative Analysis

The Brown Corpus is more appropriate for extracting statistical information on lexical labels and consecutive pairs due to its larger size and diversity of texts. Although the Susanne Corpus offers a more detailed analysis, its smaller size could limit the statistical significance of the results for this specific purpose.

Citations:

[1] https://en.wikipedia.org/wiki/Brown_Corpus#cite_ref-1

[2] https://www.grsampson.net/SueDoc.html

[3] https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=7952a55fe309491318bae178e383ff42be547b9a#:~:text=In%20this%20paper%2C%20we%20review,million%20words%20of%20American%20English.

PART II: Comparison of statistical labelers

Objective: Compare at least two statistical Part-of-Speech (POS) taggers

This research is publicly available at
https://github.com/alvaro-francisco-gil/text-mining/tree/main/01_pos_tagging

Models

Given that many comparisons of typical POS tagger models have been conducted and all classical models have been benchmarked on multiple datasets, I wanted to take a step in a different direction. I aim to benchmark Mixtral-8x7B-Instruct-v0.1 [1], a language model fine-tuned to perform instructions but not explicitly trained for POS tagging. I also want to compare this model with one specifically trained for POS tagging: bert-base-multilingual-cased-pos-english [2], a BERT model fine-tuned with the Penn Treebank for POS tagging. This comparison will allow us to explore the differences between the two models.

Test Text

When selecting a test text for the models, I needed to eliminate the possibility of using any text from the Wall Street Journal or Penn Treebank, as these were part of the BERT training dataset and could result in data leakage. For this reason, I chose to use UDPOS [3], a dataset based on the Universal Dependencies framework, which provides multilingual POS tagging annotations

Tagger Descriptions

The test dataset we are using has a different set of tags compared to the training data and the model. Because of this, we needed to create a mapping between the two label sets. When the model predicts any of the Penn Treebank labels, they must be translated to the UDPOS tag set, which contains fewer categories. The mapping is as follows:

Penn Treebank Tag	UDPOS Tag	Explanation
O	X	"Other" or undefined category.
``	PUNCT	Opening quotation mark.
,	PUNCT	Comma.
:	PUNCT	Colon or ellipsis.
.	PUNCT	Sentence-final punctuation (e.g., period, exclamation mark).
''	PUNCT	Closing quotation mark.
$	SYM	Currency symbol.
#	SYM	Symbol (e.g., hashtag).
CC	CCONJ	Coordinating conjunction (e.g., "and", "or").
CD	NUM	Cardinal number (e.g., "one", "2").
DT	DET	Determiner (e.g., "the", "a").
EX	PRON	Existential "there".
FW	X	Foreign word.

Penn Treebank Tag	UDPOS Tag	Explanation
IN	ADP	Preposition or subordinating conjunction (e.g., "in", "that").
JJ	ADJ	Adjective (e.g., "big").
JJR	ADJ	Comparative adjective (e.g., "bigger").
JJS	ADJ	Superlative adjective (e.g., "biggest").
-LRB-	PUNCT	Left round bracket "("
LS	X	List item marker.
MD	AUX	Modal auxiliary verb (e.g., "can", "should").
NN	NOUN	Singular noun (e.g., "cat").
NNP	PROPN	Proper noun, singular (e.g., "John").
NNPS	PROPN	Proper noun, plural (e.g., "Americans").
NNS	NOUN	Plural noun (e.g., "cats").
PDT	DET	Predeterminer (e.g., "all the").
POS	PART	Possessive ending (e.g., "'s").
PRP	PRON	Personal pronoun (e.g., "I", "you").

Penn Treebank Tag	UDPOS Tag	Explanation
PRP$	PRON	Possessive pronoun (e.g., "my", "your").
RB	ADV	Adverb (e.g., "quickly").
RBR	ADV	Comparative adverb (e.g., "faster").
RBS	ADV	Superlative adverb (e.g., "fastest").
RP	PART	Particle (e.g., "up" in "give up").
-RRB-	PUNCT	Right round bracket ")".
SYM	SYM	Symbol.
TO	PART	Infinitival marker ("to" in infinitives).
UH	INTJ	Interjection (e.g., "oh", "wow").
VB	VERB	Base form of verb (e.g., "run").
VBD	VERB	Past tense verb (e.g., "ran").
VBG	VERB	Gerund or present participle verb (e.g., "running").
VBN	VERB	Past participle verb (e.g., "run" in "has run").
VBP	VERB	Non-3rd person singular present verb (e.g., "run" in "I run").

Penn Treebank Tag	UDPOS Tag	Explanation
VBZ	VERB	3rd person singular present verb (e.g., "runs" in "he runs").
WDT	DET	Wh-determiner (e.g., "which", "that").
WP	PRON	Wh-pronoun (e.g., "who", "what").
WP$	PRON	Possessive wh-pronoun ("whose").
WRB	ADV	Wh-adverb ("where", "when", etc.).

Example result

Let's illustrate an example of both models before checking the overall metrics:

Example Sentence: ['What', 'if', 'Google', 'Morphed', 'Into', 'GoogleOS', '?']

Ground Truth: ['PRON', 'SCONJ', 'PROPN', 'VERB', 'ADP', 'PROPN', 'PUNCT']

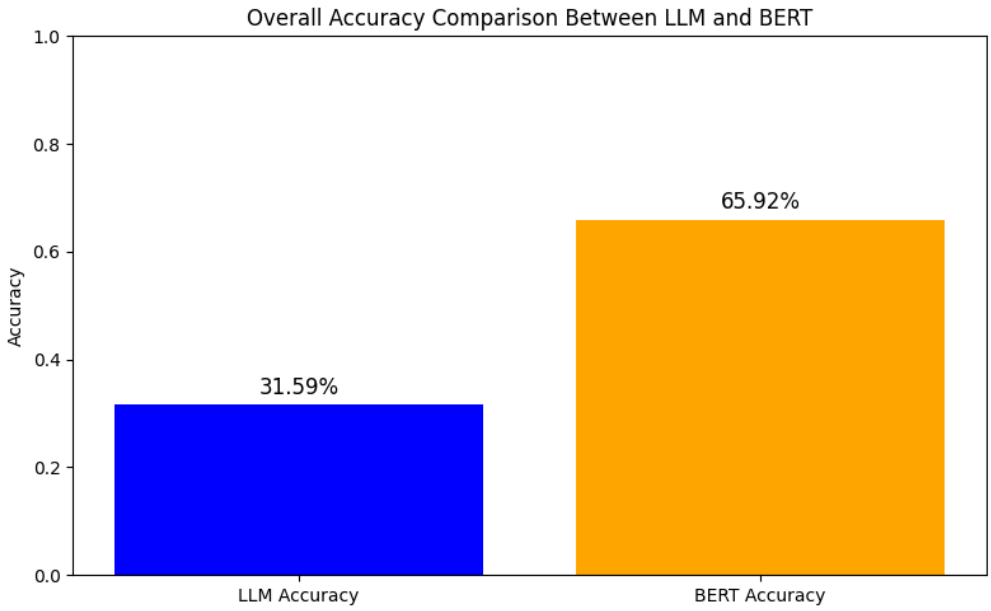
Bert Prediction: ['PRON', 'ADP', 'PROPN', 'VERB', 'ADP', 'PROPN', 'PUNCT']

LLM Prediction: ['PRON', 'SCONJ', 'PROPN', 'VERB', 'ADP', 'PROPN']

As observed, the LLM produced one word less than expected. In this case, this is counted as an error. The comparison between the ground truth and the model's predictions is performed sequentially, word by word. In this example, both BERT and the LLM achieved a score of 6/7: BERT made one mistake, while the LLM failed to analyze the final token ('?').

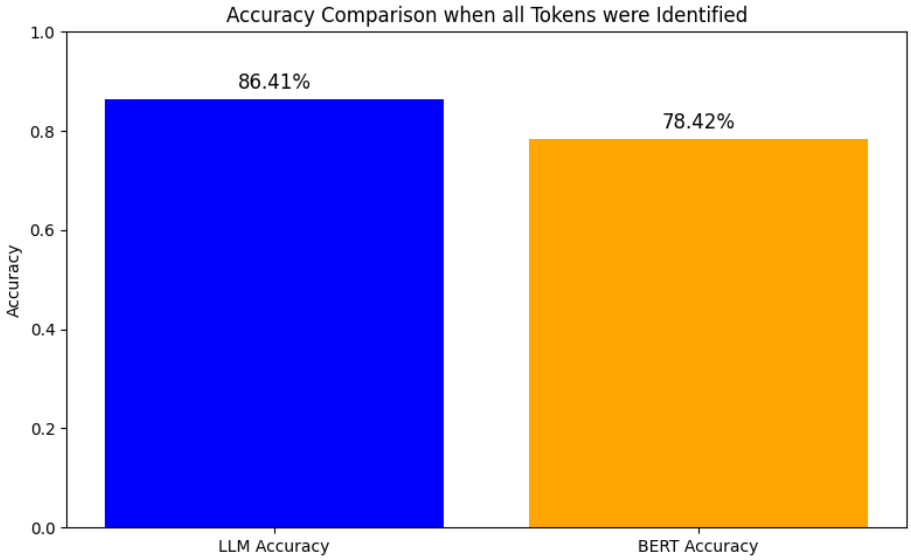
Overall result

A total of 1,136 sentences of the test set were analyzed by both models, containing 14,632 tags in total. Considering that this is essentially a classification task with 16 possible outcomes for each decision, both results are significantly higher than the random baseline of 6.25%. However, given the nature of these models, we would expect even better performance. To investigate further, we analyzed some examples in detail.

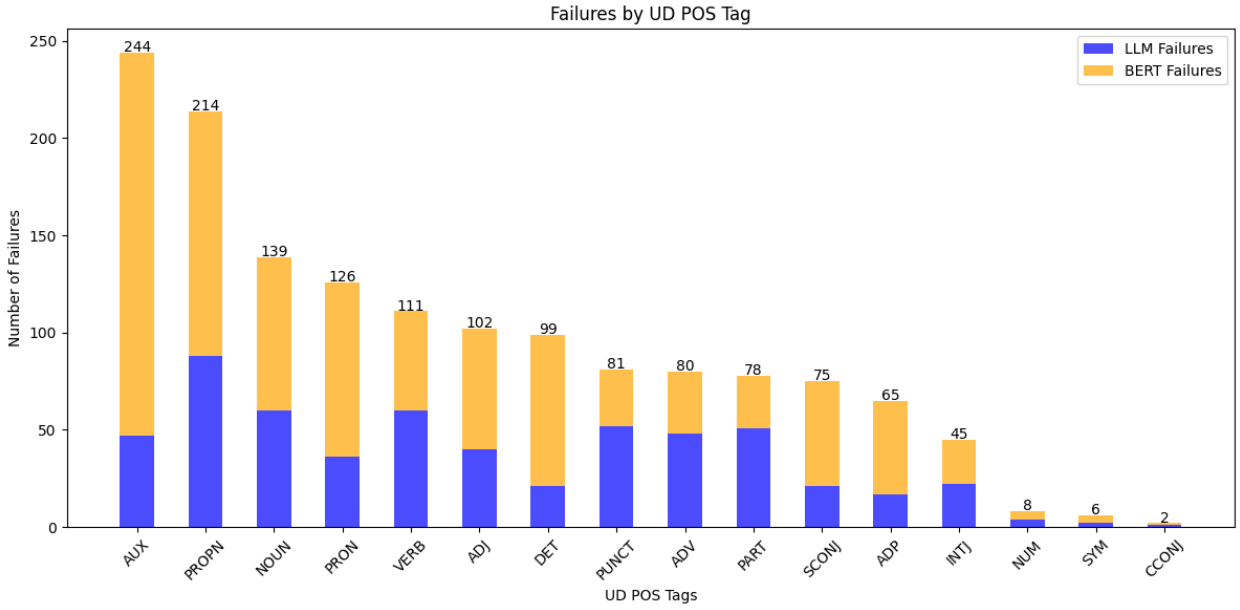


What we discovered was that, in many cases, both BERT and the LLM failed to correctly interpret the number of tokens in the sentence, as illustrated in the example result above. Consequently, when an extra or missing token was introduced, all subsequent predictions failed due to our evaluation relying on a rudimentary sequential approach.

To measure the models' performance without the inconvenience of token misinterpretation and sequential evaluation, we filtered the results to include only those sentences where the length of the predictions matched the ground truth. This filtering reduced the dataset to 617 sentences out of the initial 1136. The results for these filtered sentences are as follows:



We observed an improvement in performance for both models, particularly for the LLM, suggesting that it is more affected by word misidentification. To further analyze these results, we plotted a bar chart showing all tags and the number of times each model failed to classify a label correctly.



When visualizing the tag misclassifications, we noticed a prevalence of errors involving AUX tags. A possible explanation for this is that the mapping from PTB to UDPOS fails to differentiate auxiliary verbs (AUX) from main verbs (VERB). PTB tags like VBZ, VBP, or VBN are used for both auxiliaries and main verbs, so if BERT predicts these tags, the mapping may incorrectly assign VERB instead of AUX. This issue arises due to the simplistic nature of the mapping process, which lacks syntactic or contextual analysis.

This forced change in context caused by mapping labels could explain the decrease in BERT's performance compared to the original paper by the author (F1-score of 96.69), where the test set had labels identical to those used during training.

Conclussions

This modest research demonstrates that both models are capable of reasonably accurate POS tagging on a corpus that was not used during training. However, we identified that the primary challenge, particularly for the LLM, lies in correctly identifying the words in a sentence. This issue could be addressed through various approaches, such as improved parsing, re-prompting, or, in BERT's case, making individual predictions for each word.

For bert-base-multilingual-cased-pos-english, better token detection or fine-tuning on the specific test labels could significantly improve performance. Despite these challenges, achieving an accuracy of 0.78 under these conditions highlights the strong downstream capabilities of models fine-tuned from BERT. This suggests that even with tokenization issues, BERT retains robust performance for tasks like POS tagging.

Regarding Mixtral-8x7B-Instruct, the 0.86 accuracy achieved when tokens are properly identified underscores the potential of a multitask pretrained model that was never explicitly trained for this task. This aligns with the broader trend in AI of leveraging LLMs for diverse tasks, demonstrating their prolific potential for POS tagging and beyond. The choice of the 7B model specifically illustrates that even smaller models, which can run locally on personal hardware, are capable of performing general tasks effectively. This highlights exciting possibilities for automating tasks locally without large-scale infrastructure.

Citations:

[1] https://huggingface.co/QCRI/bert-base-multilingual-cased-pos-english

[2] https://huggingface.co/mistralai/Mixtral-8x7B-Instruct-v0.1

[3] https://universaldependencies.org/u/pos/