# Lab 2 : Task 1 - Lemmatization

Idrissa Dicko Tyler Marino Simon Khan

January 23, 2026

## Results

We evaluate three lemmatization approaches:

- **MFL token → lemma**: Most-frequent-lemma baseline learned from training data.
- **MFL (token, PoS) → lemma**: The same approach, incorporating PoS tags to reduce lexical ambiguity.
- **spaCy baseline**: Lemmatization produced by the `fr_core_news_sm` model.

### Standard Test Set Evaluation

| Model | Overall Accuracy |
|---|---|
| MFL token → lemma | 0.9531 |
| MFL (token, PoS) → lemma | **0.9653** |
| spaCy baseline | 0.8150 |

Table 1: Overall accuracy on the standard test set.

Both proposed approaches significantly outperform the spaCy lemmatizer on the standard set, with an improvement of more than **15 accuracy points**.

To better understand model behavior, we distinguish between words seen during training (*known words*) and unseen words (*unknown words*):

| Model | Known accuracy | Unknown accuracy |
|---|---|---|
| MFL token → lemma | 0.9694 | 0.6988 |
| MFL (token, PoS) → lemma | **0.9824** | 0.6988 |
| spaCy baseline | 0.8427 | 0.6988 |

Table 2: Accuracy breakdown for known and unknown tokens (Standard Set).

### Historical Evaluation (Gallica)

We further evaluated the models on the Gallica dataset to test robustness against domain shift and archaic spellings.

| Model | Known Acc. | Unknown Acc. | Overall Acc. |
|---|---|---|---|
| MFL token → lemma | 0.8231 | 0.0844 | 0.4812 |
| MFL (token, PoS) → lemma | **0.8427** | 0.0844 | **0.4917** |
| spaCy baseline | 0.8414 | 0.0844 | 0.4910 |

Table 3: Performance on the Gallica historical dataset.

The results show a severe drop in overall accuracy (≈ 49%) across all models. While known-token accuracy remains relatively high (≈ 84%), the unknown-token accuracy collapses to 8.44% due to the massive rate of Out-Of-Vocabulary (OOV) archaic spellings which neither the dictionary nor spaCy could resolve.

## Discussion

### Baseline: Most-Frequent Lemma

The simplicity of the most-frequent-lemma approach is a strength when the training and testing domains match. It consistently achieves a high accuracy of 96.9% on known tokens. However, the main limitation of this baseline is its poor performance on out-of-vocabulary tokens, particularly in domain-shift scenarios (like Gallica) where simple dictionary lookups and modern fallbacks are insufficient.

### Impact of PoS Tags

Including PoS tags provides a clear benefit for known words. By leveraging part-of-speech information, the accuracy improves from 96.9% to 98.2%. This demonstrates that additional grammatical context significantly improves the disambiguation of homographic forms (e.g., distinguishing "s" as a noun vs. a verb).

## Justification for Non-Deep Learning Approach

We deliberately chose a statistical approach over Deep Learning (DL) for this task for the following reasons:

1. **Lexical Determinism**: In standard French, lemmatization is highly deterministic. The vast majority of tokens map to a single lemma. Even ambiguous forms can be almost perfectly resolved with a simple Part-of-Speech tag. A lookup table captures these mappings instantly, whereas a neural network would require extensive training to essentially "memorize" a dictionary.

2. **Domain Matching**: The training and testing datasets come from the same domain. Deep Learning excels at generalization to unseen distributions, but in this task, the test set vocabulary overlaps heavily with the training set. Since the dictionary model already achieves ≈ 96.5% accuracy, the generalization power of a neural network would yield diminishing returns.

3. **Efficiency and Complexity**: A dictionary model trains in less than a second on a standard CPU and offers $O(1)$ inference speed. In contrast, Sequence-to-Sequence models require significant training time, hyperparameter tuning, and expensive matrix operations for inference, making them disproportionately costly for this specific problem.

4. **Handling Unknown Words**: While character-level neural networks theoretically handle OOV words better by learning morphological patterns, the engineering effort required to

tune a custom RNN to beat a pre-trained industrial fallback (like spaCy) is often disproportionate to the gain. Our hybrid approach (Dictionary + spaCy fallback) already provides a competitive baseline.