



Frédérique Rey - Sophie Robert-Hayek

Humboldt Universität - Sorbonne Universität

Project ANR SHERBET

Computational stemmatology

Using artificial intelligence to reconstruct the stemma of the Hebrew Bible

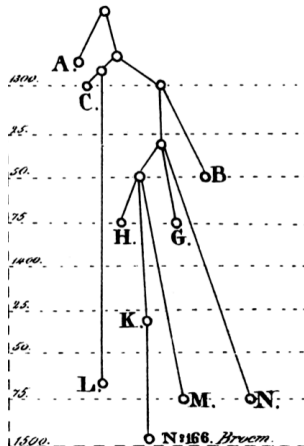
- 1 What is Stemmatology**
- 2 Computational stemmatology**
 - Distance based algorithms
 - Probability algorithms
- 3 Problem 1 – Benchmarking of stemmatology algorithms**
- 4 Problem 2 – How to define a distance between manuscripts**
- 5 Problem 3 – Text Alignment and Automatic Analysis of Variants**
- 6 Conclusion**

What is Stemmatology

What is Stemmatology?



Schema Cognationis Codicum manusc.



Stemmatology

Stemmatology is the science aimed at reconstructing the genealogical tree (stemma) of the various manuscripts of a given textual tradition.

It aims:

- to “reconstruct” an archetype in reconstructionist (lachmanian) perspectives;
- to understand the process of transmission and transformation of a textual tradition over the centuries.

Figure: Stemma codicum by

Problem: The Transmission of Harry Potter



[A] Mr. and Mrs. Potter,
of number four, Privet Drive

[B] Mr. and Mrs. Dursley,
of number four, Private Drive

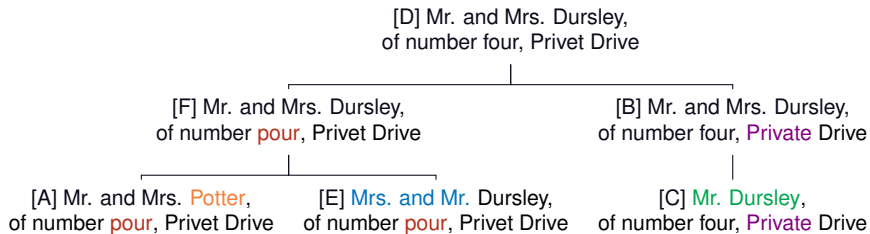
[C] Mr. Dursley,
of number four, Private Drive

[D] Mr. and Mrs. Dursley,
of number four, Privet Drive

[E] Mrs. and Mr. Dursley,
of number four, Privet Drive

[F] Mr. and Mrs. Dursley,
of number four, Privet Drive

Solution: The Transmission of Harry Potter



The issue of manuscript genealogy was first raised in the 19th century and has been at the heart of textual criticism ever since.

Approaches to solving this problem have fallen into two main categories:

1. **Qualitative Methods:** K. Lachmann and Paul Maas, based on common errors
2. **Quantitative Methods:** D. Quentin and D. Froger, based on statistical data

Can artificial intelligence help us reconstruct the stemma of Hebrew Bible manuscripts?



Computational stemmatology

Computational stemmatology

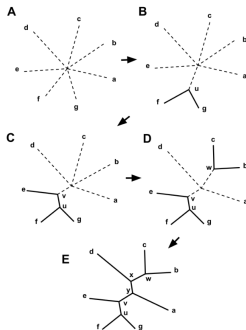
Computational stemmatology uses **computational techniques and algorithms** to reconstruct the evolutionary relationships between the witnesses.

The improvements in computational algorithms and computer speed have led to the development of **automatic algorithms** inspired from **biology**.

Many algorithms have been designed over the last 50 years:

- Distance based algorithms;
- Probabilistic based algorithms.

- Define a distance **matrix** between manuscripts;
- Iteratively group together the closest manuscripts;



Example algorithms

UPGMA (**upgma**) and Neighbor Joining (**neighbor_joining**).

- Define a probability model of transition between manuscripts : probability of going from manuscript P_i to manuscript P_j given the **variants**;
- Select the tree that is the **most likely true** given the data.

Example algorithms

Bayesian Inference (**drummond_bayesian_2015**), Maximum Likelihood trees (**max_likelihood**) ...

Problem 1 – Benchmarking of stemmatology algorithms

Benchmarking

Benchmarking refers **to the process of evaluating the performance** of a new model, algorithm, or technique by comparing it against established and standardized datasets, metrics, or existing models.

To perform benchmarking we need:

1. A golden standard (**ground truth**): a tradition where we know the true stemma;
2. A set of stemmatology algorithms to compare;
3. A metric to compare the different results between them.

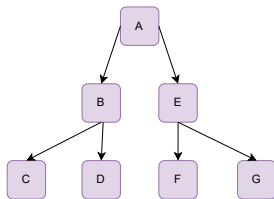
Benchmarking of stemmatology algorithms



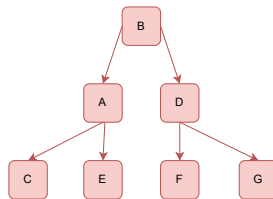
Set of manuscripts



True tree



Reconstructed tree



Distance score

Two possible approaches:

- Use “real” **handwritten benchmarking data** or traditions with known ground truth.
- Use **computer generated traditions** that imitate observed traditions and simulate variants over time.

StemmaBench is a Python library that allows you to quickly generate an artificial textual tradition given:

- An **input text**;
- A **configuration file**.

Example



Input text: Extract from *The Lion, The Witch and The Wardrobe*, by C.S. Lewis *Once there were four children whose names were Peter, Susan, Edmund and Lucy.*

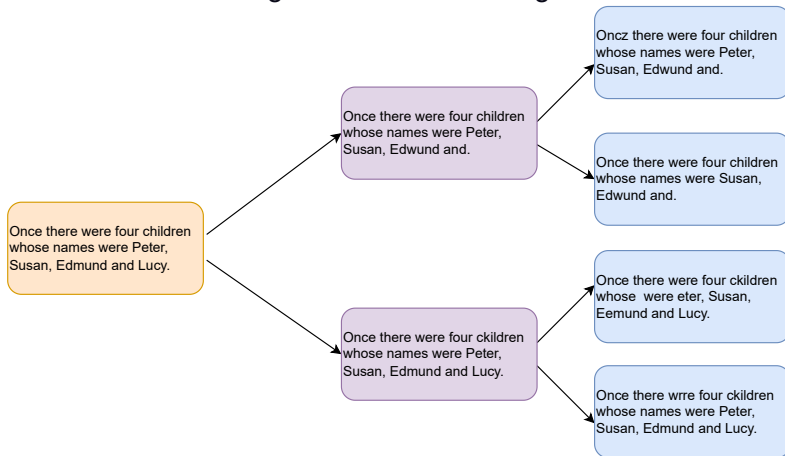
Configuration file:

```
meta:
  language: eng

variants:
  words:
    synonym:
      law: Bernouilli
      rate: 0.01
    misspell:
      law: Bernouilli
      rate: 0.1
    omit:
      law: Bernouilli
      rate: 0.05
  sentences:
    duplicate:
      args:
        nbr_words: 2
      law: Bernouilli
      rate: 0.1

stemma:
  depth: 2
  width:
    law: Uniform
    min: 2
    max: 4
```

StemmaBench will generate the following artificial tradition:



- **Easy install** using Python;
- **Fast** and **easy** tradition generation;
- **Effortless** result visualization and manipulation;
- **Flexible scribal modelization**: possibility to define your own probabilities for:
 - ▶ Misspells
 - ▶ Synonym insertions
 - ▶ Word omission
 - ▶ Word repetition
 - ▶ Word order change

Problem 2 – How to define a distance between manuscripts

How to define the distance between two texts



Example:

- **MS A** - The lion devours all the belongings in the house
- **MS B** - The tiger devours all the belongings in the house
- **MS C** - The bison devours all the belongings in the house
- **MS D** - All the belongings in the house were devoured by the Lion

What is the distance between AB, AC, and AD?

- Levenshtein distance: A and C are close (but semantically far)
- Semantic distance: A and B are close (but far in Levenshtein distance)
- A and D are semantically equivalent but syntactically inverted only attention captures their equivalence.

How to define the distance between two texts



Example:

- **MS A** - The lion devours all the belongings in the house
- **MS B** - The tiger devours all the belongings in the house
- **MS C** - The bison devours all the belongings in the house
- **MS D** - All the belongings in the house were devoured by the Lion

Characterize the variants and assess their likelihood of occurrence

- lion → tiger: synonymic variation
- lion → bison: orthographic variation (letter substitution and addition)
- etc.

How to define the distance between two texts



We need to define a distance that weights Levenshtein distance, semantic distance, attention-based methods, and variant probability

Problem 3 – Text Alignment and Automatic Analysis of Variants

Analysing the distance between two manuscripts requires a perfect alignment to identify variations, a process known as *collation*.

Automating this task has been one of the earliest areas of interest for applying computational methods to the humanities.

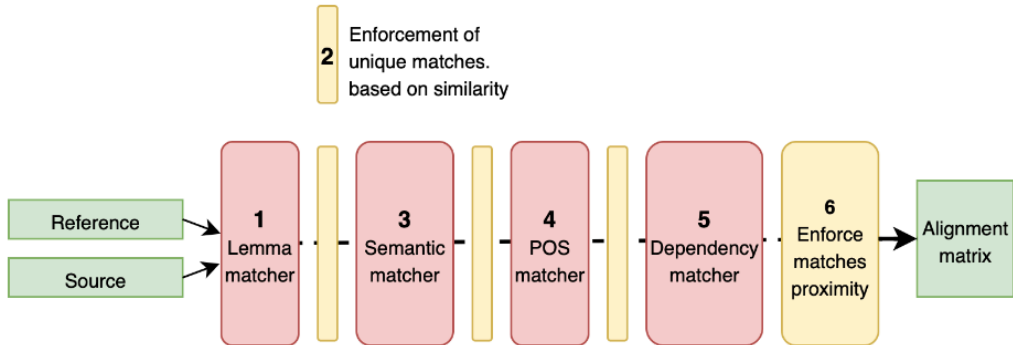
Most current approaches rely on sequence alignment algorithms, originally developed for bioinformatics. However, recent advances in mono and multilingual neural alignment, capable of modeling semantic, syntactic, and grammatical relationships, remain largely underutilized in this field.

ALMA : Automated Alignment of Ancient Texts using Linguistic and Semantic Analysis



We propose a new pipeline that integrates lexical, grammatical, and semantic analysis to perform automatic textual alignment and generate a collation.

Figure 2: Schematic representation of the workflow of the ALMA algorithm



Conclusion

A three-step pipeline

Alignment

Collation of witnesses

ALMA



Distance

Levenshtein + Semantic
+ Attention + Variant
probability

Novel distance metric



Phylogenetics

Stemma reconstruction

StemmaBench

These tools do not replace the philologist they allow large-scale analysis of what philologists already do manually on a handful of manuscripts.

Next steps

- Extend to **multilingual traditions** (Hebrew, Greek, Syriac)
- Fine-tune models on **ancient language corpora**
- Apply to **Ben Sira** and the Hebrew Bible