Co-funded by the
Erasmus+ Programme
of the European Union

EULALIA

European Latin Linguistic Assessment

Erasmus+ Strategic Partnership for Higher Education (2019-2022)

(2019-1-IT02-KA203-062286)

https://site.unibo.it/eulalia/en

O1: European Latin Language Certification – Basic Level

Methodological and Pedagogical tools

LEXICON

(French Version: 31.05.2021)

Project Coordinator:

Alma Mater Studiorum – University of Bologna (Italy)

Project Partners:

University of Köln (Germany)

Catholic University of the Sacred Heart – Milan (Italy)

University of Rouen (France)

University of Salamanca (Spain)

University of Uppsala (Sweden)

Un nouveau « lexique de base »

Guido Milanese*

1) État de l'art

Le projet d'une certification linguistique, comme celle que met en place Eulalia, requiert un lexique de base. L'offre dans ce domaine est large : voir par ex. les outils très développés offerts par le Perseus Project[1] ou encore la liste fournie par le Dickinson College (équipe menée par Christopher Francese)[2]. Cependant, la liste du Dickinson College est limitée à 1000 mots, alors que le programme Eulalia requiert un plus grand nombre de mots pour les niveaux de certification plus avancés. De plus, ces listes ne fournissant pas de renseignements sur les sources utilisées pour les établir, les membres du projet Eulalia ont décidé de fournir aux futurs étudiants une liste préparée à partir de textes latins classiques, c'est-à-dire les textes qui font partie du véritable « canon » utilisé dans les écoles européennes.

Pour tous ces raisons, l'ancien et vénérable *Vocabulaire de base du latin*, édité en 1984 par une équipe de l'Université de Besançon (A.R.E.L.A.B.), qui s'est servi du *Dictionnaire fréquentiel* édité par le groupe liégeois LASLA en 1981, peut encore servir de fondement fiable pour construire un nouveau vocabulaire de base adapté aux besoins de l'enseignement du latin dans les écoles et les universités européennes. Les avantages de ce travail sont :

1. Le nombre de mots : les 1600 mots proposés sont compatibles avec le programme Eulalia
2. Une liste claire des textes latins utilisés pour préparer le lexique :

> « Le corpus analysé dans cet ouvrage comprend, en partie ou en totalité, les œuvres de Catulle, César, Cicéron, Horace, Juvénal, Ovide, Perse, Properce, Quinte-Curce, Salluste, Sénèque, Tacite, Tibulle, Tite-Live, Virgile, Vitruve »[3]

Aucune liste n'est bien sûr parfaite et l'absence d'auteurs comme Térence ou Lucrèce peut être regrettée. Néanmoins, afin de préparer une liste « neutre » de mots latins de la fin de la République et du début de l'Empire, le choix des auteurs est adéquat. L'auteur de cet article estime que le latin médiéval peut fournir aux enseignants une grande quantité d'ouvrages très efficaces pour l'enseignement du latin, mais, dans ce cas également, le lexique de base peut être facilement intégré.

2) Scan et OCR du lexique de Besançon

---

[1] EULALIA – European Latin Linguistic Assessment
https://www.perseus.tufts.edu/hopper/help/vocab

[2] http://dcc.dickinson.edu/vocab/core-vocabulary

[3] G. Cauquil & J.Y. Guillaumin, *Vocabulaire de base du latin (alphabétique, fréquentiel, étymologique)*, A.R.E.L.A.B., Besançon 1984, p. 4.

Le lexique franco-belge a été publié sous forme de livre[4]. Malheureusement, la qualité de l'impression originale est celle d'un livre imprimé dans les années 1980 avec les dispositifs de l'époque. La liste de mots a été numérisée à l'aide d'un HP OfficeJet Pro 7720 standard, et l'OCR a été effectuée par le programme OCR bien connu Tesseract, avec l'option latin[5]. La liste a été corrigée manuellement mais, à ma grande surprise, même la première sortie Tesseract était déjà très bonne.

3) Taggage morphologique

La liste finale fournie par le lexique de Liège-Besançon est une simple liste de mots et de leur fréquence, avec quelques notes ajoutées ici et là. Afin d'utiliser cette liste pour une véritable liste du « lexique de base », il était évidemment nécessaire d'ajouter un taggage morphologique adapté. Pour cela, j'ai utilisé Treetagger, le programme morphologique conçu par Helmut Schmid dans le projet TC de l'Institute for Computational Linguistics de l'Université de Stuttgart et maintenant hébergé par l'Université de Munich en Allemagne[6]. L'exécution de Treetagger sur une liste de mots est une approche sujette aux erreurs : le programme ne peut pas effectuer d'analyse contextuelle, et les entrées, comme par exemple *quam*, sont évidemment équivoques[7]. L'analyse morphologique proposée par Treetagger a été révisée manuellement ; environ 15% des balises étaient fausses, comme prévu. Les notes A.R.E.L.A.B. ont été utilisées pour corriger le taggage Treetagger, chaque fois qu'il était différent de la catégorie morphologique offerte par la note A.R.E.L.A.B.

4) Le problème des « mots-outils »

Lors d'une réunion de l'équipe Eulalia, Mélanie Lucciano (Université de Rouen-Normandie) a remarqué qu'une liste de mots basée purement sur leur fréquence ne convenait pas pour servir d'outil pour l'enseignement d'une langue car les mots-fonctions (« function words », « mots-outils », « morfemi grammaticali », « parole vuote » ...) sont très courants dans toutes les langues et figurent au rang le plus élevé d'une liste de fréquences. En conséquence, si nous supposons une liste de 400 mots pour un niveau A1, les étudiants apprendraient une grande quantité de conjonctions et de prépositions, au détriment des verbes, des noms, etc.

Les classes morphologiques listées par Treetagger sont les suivantes :

- ADJ
- ADJ:NUM
- ADV

---

[4] La simple liste de mots a été réimprimée dans des travaux plus récents, comme Paolo Lamagna, *Il lessico latino di base*, Bompiani, Milan, 1999.

[5] À propos de Tesseract, voir Guido Milanese, *Filologia, letteratura, computer. Idee e strumenti per l'informatica umanistica*, Vita e Pensiero, Milan, 2020, p. 103-107, et la documentation en ligne à l'adresse : https://github.com/tesseract-ocr. Le programme est FOSS (Free and OpenSource Software) et tourne sur un système Linux Mint 20.1.

[6] Voir https://www.cis.uni-muenchen.de/~schmid/tools/TreeTagger/, et https://www.cis.lmu.de/~schmid/ pour une bibliographie récente.

[7] Il n'y a pas de réelle différence lorsqu'on utilise la création plus récente de Schmid, appelée « RNNTagger » : voir https://www.cis.lmu.de/~schmid/tools/RNNTagger/.

- CC
- CS
- DET
- DIMOS
- ENCL
- ESSE
- EXCL
- FW
- INDEF
- INT
- N
- POSS
- PREP
- PRON
- REL
- V

Puisqu'il s'agit de l'enseignement de la langue et que le but de l'adaptation de la liste de mots originale d'A.R.E.L.A.B. est d'éviter un nombre trop important de mots-outils, je proposerais de considérer comme « mots-outils » les catégories suivantes : conjonctions, prépositions et clitiques.

Un simple programme SNOBOL4 appliqué à la liste A.R.E.L.A.B. permet d'obtenir les résultats suivants[8]:

- Sur la totalité de la liste (1635 mots), les « mots-outils » sont 78, c'est-à-dire 4.1260 %
- Sur les premiers 600 mots, les « mots-outils » sont 55, c'est-à-dire 9.100 %
- Sur les premiers 400 mots, les « mots-outils » sont 45, c'est-à-dire 11.100 %

De plus, 57% des « mots-outils » figurent parmi les 400 premiers mots, et près de 70% parmi les 600 premiers mots. En conséquence, une liste des mots établie par fréquence n'est pas adaptée pour tester la compétence linguistique des étudiants, car parmi les 400/600 premières positions il y a trop de mots-outils. Le but de cette liste révisée est d'équilibrer la répartition entre mots-outils et « mots-sémantiques », en déplaçant certains mots-sémantiques situés plus bas dans la liste (c'est-à-dire les mots après les n.°400 et n.°600) afin d'obtenir un pourcentage plus équilibré entre les mots-sémantiques et les mots-outils. En d'autres termes, les 400 et 600 premiers mots de la liste doivent comporter un pourcentage de mots-outils similaire à celui de l'ensemble de la liste.

La liste a été révisée en déplaçant certains mots-outils vers le bas de la liste, en veillant toutefois à ce que les mots-outils essentiels (comme *quia* ou *si*) soient répertoriés dans les listes pour les

[8] Bien qu'il s'agisse d'un langage de programme « de niche », snobol4 est bien maintenu, surtout grâce aux efforts de Phil Budne : voir https://www.snobol4.org/. La version la plus récente de ce langage a été rendue publique en décembre 2020. Sur ce langage de programmation particulièrement adapté à l'analyse textuelle, voir G. Milanese, *Filologia, letteratura, computer*, *ibid.*, p. 241-244. Snobol4 tourne sous Linux, Windows, et OSX.

débutants. Ce travail a été effectué avec une approche assistée par ordinateur, c'est-à-dire en vérifiant l'équilibre des mots étape par étape.

La liste révisée présente alors les pourcentages suivants :

- Sur les premiers 600 mots, les « mots-outils » sont 46, c'est-à-dire 7.4%
- Sur les premiers 400 mots, les « mots-outils » sont 30, c'est-à-dire 7.2%

Même si ces pourcentages ne reflètent pas exactement les données et les caractéristiques de l'ensemble de la liste, cela me semble un compromis raisonnable – bien que discutable, bien sûr, et ouvert à d'autres améliorations – à mi-chemin entre l'exactitude et les besoins pratiques d'enseignement.

5) Résultat final

Le résultat final a été produit sous la forme d'une liste csv (Comma Separated Values), qui peut être lue par des programmes standard comme LibreOffice Calc ou MSOffice Excel. J'ai ajouté un simple header :

N, VOX, MORPHO, NOTAE, FREQ
Où N= nombre, VOX= le mot de la liste, MORPHO= le tag morphologique, NOTAE= la note originale du *Vocabulaire* de Liège–Besançon, et FREQ la fréquence du mot.

Par exemple :
16, QUIS, PRON, interr., 4555

L'information donnée par le *Vocabulaire* est ajoutée pour spécifier le type de pronom auquel appartient *quis*. Lorsque l'information donnée par le *Vocabulaire* est la même que le tag morphologique, elle n'est pas indiquée.
Pour rendre la liste plus lisible, un fichier PDF a été produit à partir de la liste CSV, en utilisant Pandoc[9].

6) Listes à partir de la liste

Puisque le but du Lexicon est de fournir aux enseignants et aux étudiants des outils pratiques et faciles à utiliser, j'ai réorganisé les listes sous différentes formes. La première étape, évidente, consistait à imprimer les listes des 400 et 600 premiers mots.

6.1 Listes alphabétiques

[9] Voir https://pandoc.org.

Toutes les listes (c'est-à-dire la liste entière et les listes des « 400 mots » et des « 600 mots ») ont été triées par ordre alphabétique, afin que l'enseignant ou l'étudiant repère un mot donné en parcourant la liste par ordre alphabétique. La liste alphabétique complète a également été transformée en fichier PDF.

6.2 Listes morphologiques

Toutes les listes (c'est-à-dire la liste entière et les listes des « 400 mots » et des « 600 mots ») ont été triées suivant un ordre morphologique, afin que l'enseignant ou l'étudiant repère un mot donné en parcourant la liste en suivant les catégories morphologiques. La liste morphologique complète a également été transformée en fichier PDF.

7) L'interface graphique utilisateur (GUI) de la liste

Pour faciliter l'utilisation de ces listes, une interface utilisateur graphique (GUI) très simple a été développée par le présent auteur et est à la disposition des bêta-testeurs intéressés par ce projet.

8) Code des scripts snobol4 et bash

8.1 Le script snobol4

Le script snobol4 pour le calcul des pourcentages :

```
        P_type = break(',') len(1) break(',') len(1) (break(','). M_type)
        KL = 0
        KE = 0
loop            Line = input            :f(loop_end)
        KL = KL +1
        Line ? P_type
        ((leq(M_type,"CC")),   (leq(M_type,"CS")),    (leq(M_type,"ENCL")),
+     (leq(M_type,"PREP")))  (KE = KE +1) : (loop)
loop_end        Output = KE '/'  KL
        Extreme = KE  *100
        Percentage = Extreme / KL
        Modulus = Remdr(Extreme,KL)
        Terminal = Percentage '.'  Modulus
end
```

La première ligne définit un *pattern* : dans une ligne CSV donnée, le programme parcourt la ligne jusqu'à trouver une première virgule, puis un caractère, et répète ces mêmes étapes, et, à la troisième répétition, sauvegarde l'information en tant que type morphologique (dans M_type). Ensuite, on

initialise à zéro le compteur de lignes (KL) et le compteur de mots vides (KE)[10]. La boucle lit chaque ligne à partir de la sortie standard ; si le type morphologique est un mot vide, alors le compteur KE est incrémenté de 1. Après la boucle, des opérations mathématiques simples produisent les sorties désirées.

Le même script est exécuté sur les premiers 400/600 mots en ajoutant une limite :

```
lt (KL, 601)
```

Dans ce cas, la boucle se terminera à la 600ème ligne (lt= *less than*).

8.2 Les scripts bash

Les scripts bash utilisés pour organiser les lignes en ordre alphabétique ou morphologique sont des simples scripts de tri, comme l'exemple suivant :

```
sort -t,  -k3,3 -k2 frequentia-rev-1-400.csv\
> frequentia-rev-1-400-morph.csv
```

Cette commande trie la liste « 400 » en fonction du 3ème champ (le type morphologique) et ensuite 2ème champ (l'ordre alphabétique), écrivant la sortie dans un nouveau fichier (frequentia-rev-1-400-morph.csv). La virgule est utilisée comme champ séparateur ("-t," or "-t','").

8.3 Du CSV au PDF en utilisant Pandoc

Pandoc est un programme en ligne de commande très efficace pour transformer de très nombreux formats fréquemment utilisés (par exemple DOCX, LATEX, markdown) en d'autres formats[11]. La version la plus récente de Pandoc peut lire un fichier CSV et le transformer en d'autres formats, incluant le PDF (en faisait appel à TEX de manière non explicite).

Il s'agit d'une commande simple :

```
pandoc -f csv frequentia-format-rev-morph.csv\
  -s -o frequentia-format-rev-morph.pdf
```

Pandoc lit un fichier CSV (-f signifie « depuis ») et produit en sortie un fichier PDF. Des lignes de commande similaires s'appliquent à tous les autres fichiers.

---

[10] Cette étape n'est pas nécessaire mais l'initialisation des données rend le programme plus facile à lire et à comprendre.
[11] Voir https://pandoc.org/

Références bibliographiques

Cauquil, G. & Guillaumin, J. Y., *Vocabulaire de base du latin (alphabétique, fréquentiel, étymologique)*, A.R.E.L.A.B., Besançon, 1984.
Delatte, L., Evrard, Et., Govaerts, S., & Denooz, J., *Dictionnaire fréquentiel et index inverse de la langue latine*, L.A.S.L.A. (Laboratoire d'analyse statistique des langues anciennes), Université de Liège, Liège, 1981.
Lamagna, Paolo, *Il lessico latino di base*, Bompiani, Milano, 1999.
Milanese, Guido, *Filologia, letteratura, computer. Idee e strumenti per l'informatica umanistica*, Vita e Pensiero, Milano, 2020.