



Institut européen

des métiers de la **traduction** | IEMT

Université de Strasbourg

# Web, corpus, traduction : exploitations

## AntConc, expressions régulières

**Enzo Doyen**

2025 - M1

# Connaissez-vous ELIZA ?

EEEEEE	LL	IIII	ZZZZZZZ	AAAAA
EE	LL	II	ZZ	AA AA
EEEE	LL	II	ZZZ	AAAAAAA
EE	LL	II	ZZ	AA AA
EEEEEE	LLLLLL	IIII	ZZZZZZZ	AA AA

# Connaissez-vous ELIZA ?

```
EEEEEE LL      IIII ZZZZZZZZ AAAAAA
EE      LL      II      ZZ      AA      AA
EEEEEE LL      II      ZZZ      AAAAAAA
EE      LL      II      ZZ      AA      AA
EEEEEE LLLLLL IIII ZZZZZZZZ AA      AA
```

Premier « agent conversationnel » développé entre 1964 et 1967 par Joseph Weizenbaum.

# Connaissez-vous ELIZA ?

Premier « agent conversationnel » développé entre 1964 et 1967 par Joseph Weizenbaum.

Recherche publiée dans Communications of the Association for Computing Machinery:

*Weizenbaum, J. (1966). ELIZA—a computer program for the study of natural language communication between man and machine*

# Test de Turing

Test de la capacité d'une machine à imiter une conversation humaine.

*A. M. Turing (1950). Computing Machinery and Intelligence. Mind, 59, 433-460.*

# Exemple de conversation avec ELIZA

User: Men are all alike

**ELIZA: IN WHAT WAY**

User: They're always bugging us about something or other

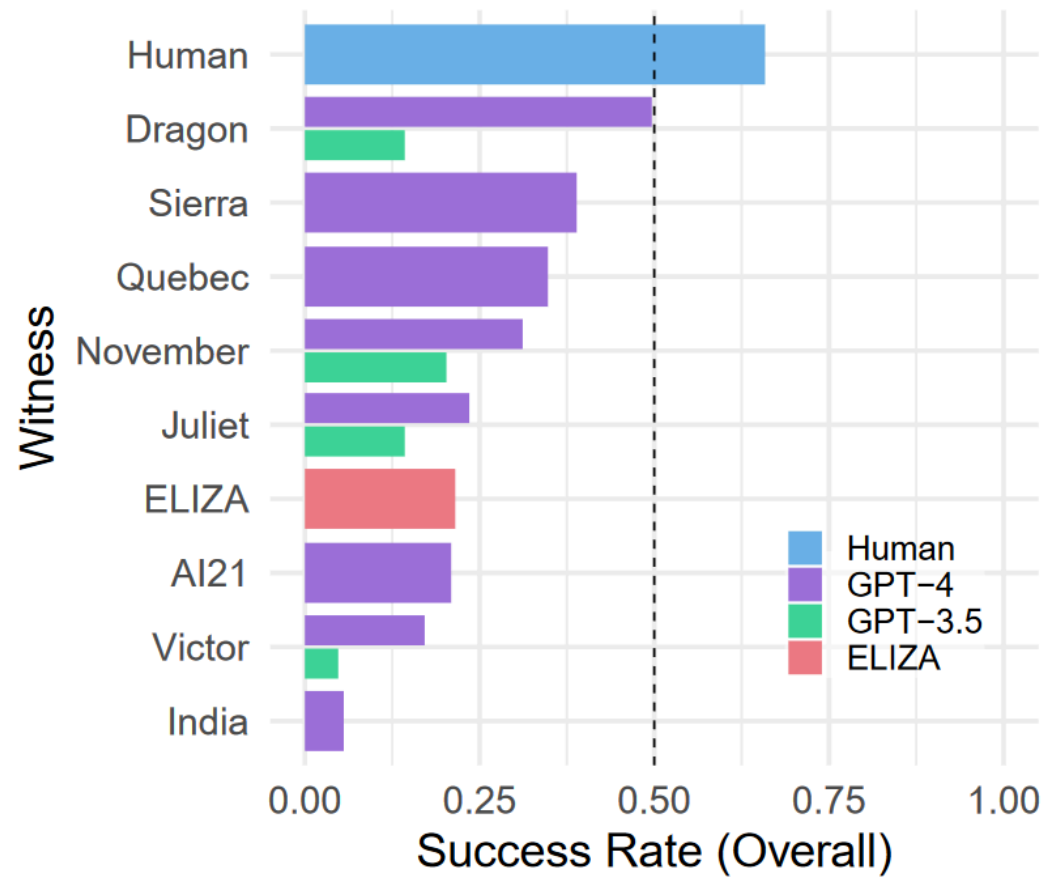
**ELIZA: CAN YOU THINK OF A SPECIFIC EXAMPLE**

User: Well, my boyfriend made me come here

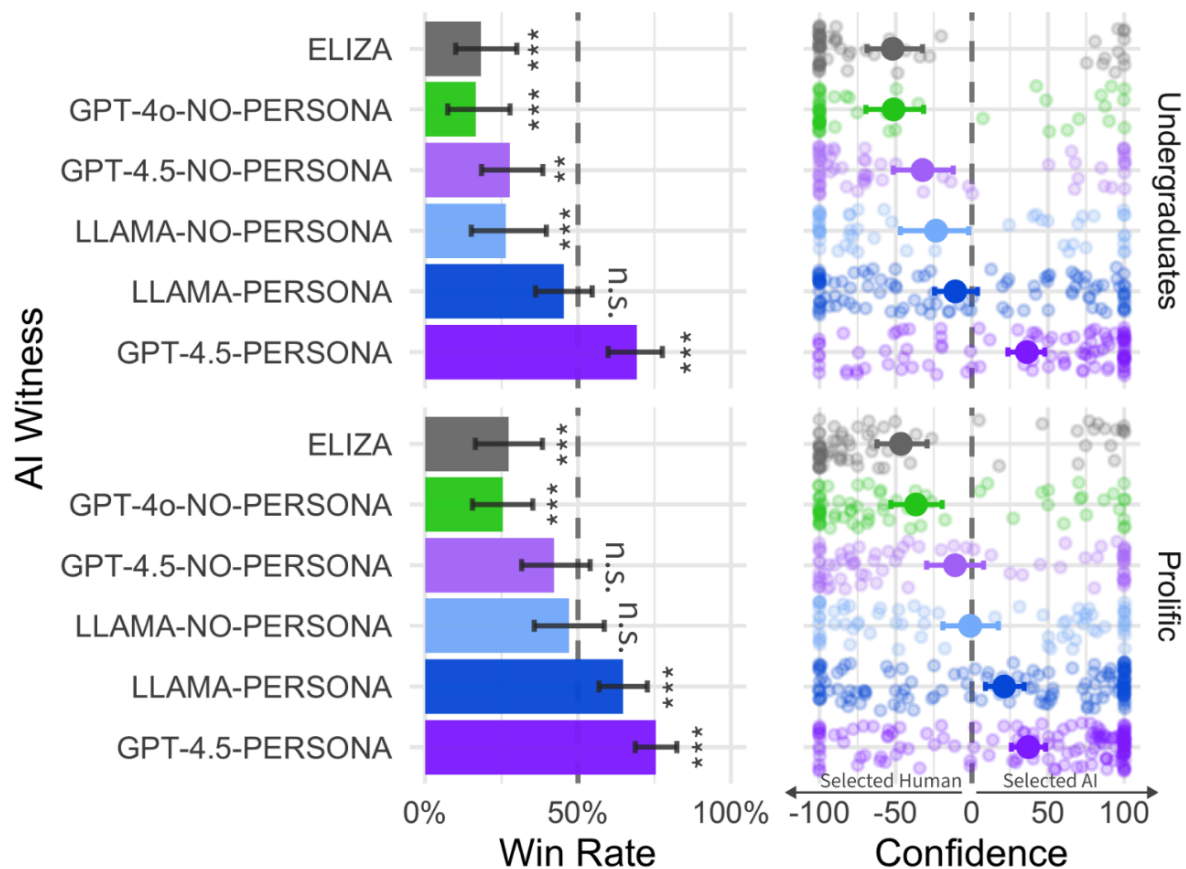
**ELIZA: YOUR BOYFRIEND MADE YOU COME HERE**

User: He says I'm depressed much of the time

**ELIZA: DO YOU THINK YOU ARE DEPRESSED MUCH OF THE TIME**



Jones et Bergen (2024), *Does GPT-4 pass the Turing test?* arXiv:2310.20216v2



Jones et Bergen (2025), *Large Language Models Pass the Turing Test*  
 arXiv:2503.23674



# Expressions régulières

ELIZA fonctionne (en partie) avec des expressions régulières.

À la fin du cours, vous pourrez vous aussi (théoriquement) créer votre propre ELIZA.

# Expressions régulières : fonctionnement

- ♦ Outil pour rechercher et manipuler des chaînes de caractères
- ♦ Permet notamment de trouver des séquences qui suivent un même motif :
  - les mots ayant un certain préfixe/suffixe ;
  - les mots contenant une certaine séquence ;
  - d'autres types de chaînes de caractères (adresses e-mail, numéros de téléphone, etc.)

# Expressions régulières : applications pratiques

- ♦ Recherches complexes rapides dans un texte ou un ensemble de corpus ;
- ♦ Remplacement de chaînes de caractères ;
- ♦ Outil extrêmement répandu et puissant, disponible dans tous les outils de TAO (MemoQ, Trados, OmegaT, Smartcat...).

WHENEVER I LEARN A  
NEW SKILL I CONCOCT  
ELABORATE FANTASY  
SCENARIOS WHERE IT  
LETS ME SAVE THE DAY.

OH NO! THE KILLER  
MUST HAVE FOLLOWED  
HER ON VACATION!



BUT TO FIND THEM WE'D HAVE TO SEARCH  
THROUGH 200 MB OF EMAILS LOOKING FOR  
SOMETHING FORMATTED LIKE AN ADDRESS!



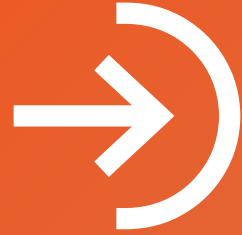
IT'S HOPELESS!

EVERYBODY STAND BACK.



I KNOW REGULAR  
EXPRESSIONS.





# I. Introduction aux expressions régulières

# Expressions régulières : principes de base

En général, chaque caractère se reconnaît lui-même.

Exemple : **/e/**



semaine

# Expressions régulières : principes de base

Exemple : `/em/`

**s****maine**

# Expressions régulières : principes de base

Différence majuscules/minuscules.

Exemple : `/T/`

**T**out



# Expressions régulières : séquences spéciales

Il existe des symboles utilisés pour trouver des groupes de plusieurs caractères.

**\d** : n'importe quel chiffre

**Windows 11**

# Expressions régulières : séquences spéciales

Il existe des symboles utilisés pour trouver des groupes de plusieurs caractères.

**\s** : n'importe quelle espace (TAB, retour à la ligne inclus)

**Windows 11**

# Expressions régulières : séquences spéciales

Il existe des symboles utilisés pour trouver des groupes de plusieurs caractères.

**\w** : n'importe quelle lettre (majuscule/minuscule, chiffre ou \_)

!@)a \_ 3 \A

# Expressions régulières : séquences spéciales

Il existe des symboles utilisés pour trouver des groupes de plusieurs caractères.

- (point) : tous les caractères (sauf retour à la ligne)

**Windows 11**

# Expressions régulières : inversion des séquences spéciales

Les séquences spéciales peuvent être inverser pour trouver ce qui ne correspond **pas** à un certain motif.

**\D** : n'importe quel caractère qui n'est pas un chiffre

**\S** : n'importe quel caractère qui n'est pas une espace

**\W** : n'importe quel caractère qui n'est pas une lettre, un chiffre ou un \_

# Expressions régulières : ensembles de caractères

Les ensembles de caractères permettent de chercher un caractère parmi plusieurs. On utilise les crochets `[]`.

Exemple : `[aeiou]`



Windows 11

# Expressions régulières : ensembles de caractères

Les ensembles de caractères permettent de chercher un caractère parmi plusieurs. On utilise les crochets `[]`.

**Exemple** : `[0123456789]` (équivalent de `\d`)

Windows 11

# Expressions régulières : ensembles de caractères

Les ensembles de caractères permettent de chercher un caractère parmi plusieurs. On utilise les crochets `[]`.

**Exemple** : `[0-9]` (équivalent de `\d` ou `[0123456789]`)

Windows 11



# Expressions régulières : ensembles de caractères

Les ensembles de caractères permettent de chercher un caractère parmi plusieurs. On utilise les crochets **[ ]**.

**Exemple** : **[2-9]** – rien n'est trouvé car il n'y a pas de chiffre entre 2 et 9 !

# Windows 11

# Expressions régulières : ensembles de caractères

**[a-z]** : n'importe quelle lettre minuscule

**[A-Z]** : n'importe quelle lettre majuscule

**[a-zA-Z]** : n'importe quelle lettre, minuscule ou majuscule

**[a-zA-Z0-9]** : n'importe quel lettre ou chiffre

# Expressions régulières : négation

Les crochets peuvent aussi être utilisés pour exclure des ensembles de caractères.


**Exemple** : `[^\d]` (tout ce qui n'est pas un chiffre)

**w****i****n****d****o****w****s****|****1****1**

# Expressions régulières : négation

Les crochets peuvent aussi être utilisés pour exclure des ensembles de caractères.

**Exemple** : `[^aeiou]` (tout ce qui n'est pas une voyelle)



Windows

# Expressions régulières : ancres

Les ancres spécifient la position d'un motif dans une chaîne de caractères.

**^** : début de la chaîne

**\$** : fin de la chaîne

# Expressions régulières : ancres

Les ancres spécifient la position d'un motif dans une chaîne de caractères.

Exemple : `^ka`



katakana

# Expressions régulières : ancres

Les ancres spécifient la position d'un motif dans une chaîne de caractères.

Exemple : **k\$**



knack

# Expressions régulières : frontières de mots

Une alternative aux ancres sont les frontières de mots : `\b`.

Exemple : `\bka`



katakana



# Expressions régulières : frontières de mots

Une alternative aux ancres sont les frontières de mots : `\b`.

Exemple : `\bchat\b`

le **chat** et le chaton

# Expressions régulières : échappement des caractères spéciaux

Et si je veux trouver un point, un crochet ou un signe dollar dans mon texte ?  
On peut échapper le caractère avec un antislash \.

Exemple : \.

M.Darcy

# Expressions régulières : échappement des caractères spéciaux

Et si je veux trouver un point, un crochet ou un signe dollar dans mon texte ?  
On peut échapper le caractère avec un antislash \.

Exemple : \\$

Coute 20 \$

# Expressions régulières : quantifieurs

Les quantifieurs permettent de spécifier le nombre de fois qu'un motif doit apparaître.

**{n}** : exactement n fois | **{n,m}** : entre n et m fois

**?** : 0 ou 1 fois

**\*** : 0 ou plusieurs fois

**+** : 1 ou plusieurs fois

# Expressions régulières : quantifieurs

Les quantifieurs permettent de spécifier le nombre de fois qu'un motif doit apparaître.

**Exemple :** `l{2}`



aller

# Expressions régulières : quantifieurs

Les quantifieurs permettent de spécifier le nombre de fois qu'un motif doit apparaître.

**Exemple** :  $\{1,2\}$

duquel aller

# Expressions régulières : quantifieurs

**Attention** : les quantifieurs ne s'appliquent qu'au caractère (ou au groupe) qui les précède.

**Exemple** : `char{1,2}iot`

**chariot**

**charriot**

# Expressions régulières : quantifieurs

**?** : 0 ou 1 fois (équivalent : **{0,1}**)

**Exemple** : **sons?**

bison  
bisons



# Expressions régulières : quantifieurs

**+** : 1 ou plusieurs fois (équivalent : **{1,}**)

**Exemple** : **ab.+**

**ab**

**abcdefg**

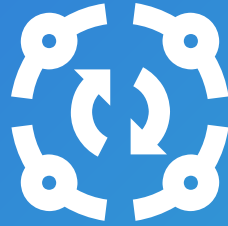
# Expressions régulières : quantifieurs

\* : 0 ou plusieurs fois (équivalent : {0,})

Exemple : **ab.\***

**ab**

**abcdefg**



## II. Expressions régulières : groupes, références et remplacements

# Expressions régulières : groupes et références

Les groupes peuvent être utilisés pour indiquer un choix entre plusieurs motifs avec la barre verticale |. On utilise les parenthèses **()**.

**Exemple** : **[LI](e|a)**

# Expressions régulières : groupes et références

Les groupes peuvent être utilisés pour indiquer un choix entre plusieurs motifs avec la barre verticale |. On utilise les parenthèses **()**.

**Exemple** : **[LI](e|a)**

**Le chat et la souris**

# Expressions régulières : groupes et références

Les groupes permettent également de capturer des motifs pour les réutiliser plus tard, et utiliser ces mêmes motifs dans le cadre d'un remplacement.

**Exemple :** (ab)

# Expressions régulières : groupes et références

Une fois un groupe défini, on peut le réutiliser dans le cadre d'un remplacement avec `\1`, `\2`, etc. (ou, selon les outils, `$1`, `$2`, etc.).

## Expressions régulières : groupes et références

**Dupont Jean**  
**Martin Alice**

On peut facilement renverser l'ordre nom-prénom en prénom-nom grâce aux expressions régulières.



## Expressions régulières : groupes et références

**Dupont Jean**  
**Martin Alice**

Soit l'expression régulière : `(\w+)\s+(\w+)`

Et le remplacement : `\2 \1`

# Expressions régulières : groupes et références

Soit l'expression régulière : `(\w+)\s+(\w+)`

Et le remplacement : `\2 \1`

**Jean Dupont**  
**Alice Martin**

# Expressions régulières : groupes et références

Autre exemple : imaginons que nous voulions remplacer les dates au format **jj/mm/aaaa** par le format **aaaa-mm-jj**.

On peut faire un groupe pour chaque partie de la date et les réutiliser dans le remplacement.

# Expressions régulières : groupes et références

❓ **Question :** Quelle pourrait être l'expression régulière pour récupérer les dates au format jj/mm/aaaa, et récupérer individuellement chaque partie de la date ?

(Par exemple : 07/10/2024)

# Expressions régulières : groupes et références

Remplacement de dates : 07/10/2024

**Expression régulière** : `(\d{2})/(\d{2})/(\d{4})`

# Expressions régulières : groupes et références

Remplacement de dates : 07/10/2024

Expression régulière : `(\d{2})/(\d{2})/(\d{4})`

`(\d{2})` / `(\d{2})` / `(\d{4})`

07

10

2024

# Expressions régulières : groupes et références

Remplacement de dates : 07/10/2024

❓ **Question :** Quelle pourrait être l'expression de remplacement pour obtenir le format aaaa-mm-jj ?

Pour rappel, les groupes sont référencés par \1, \2, etc.

# Expressions régulières : groupes et références

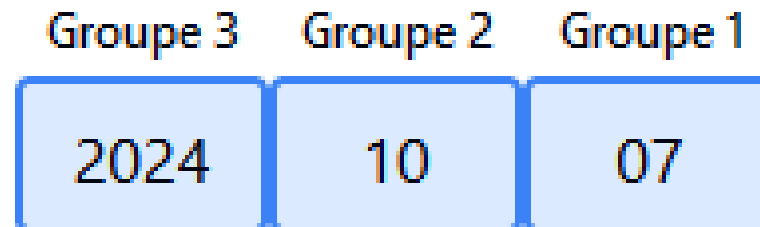
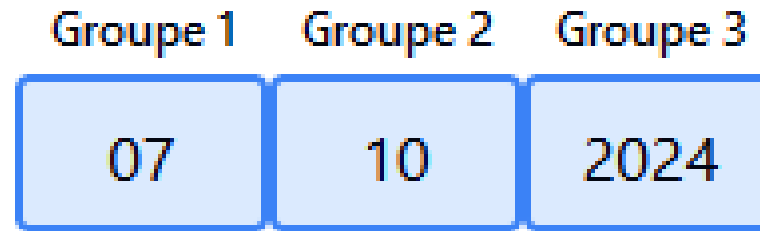
Remplacement de dates : 07/10/2024

Remplacement : \3-\2-\1

Résultat : 2024-10-07



# Expressions régulières : groupes et références





**C'est fini ! Ouf !**

# Exemple de conversation avec ELIZA

User: Men are all alike

**ELIZA: IN WHAT WAY**

User: They're always bugging us about something or other

**ELIZA: CAN YOU THINK OF A SPECIFIC EXAMPLE**

User: Well, my boyfriend made me come here

**ELIZA: YOUR BOYFRIEND MADE YOU COME HERE**

User: He says I'm depressed much of the time

**ELIZA: DO YOU THINK YOU ARE DEPRESSED MUCH OF THE TIME**

# Implémentation partielle d'ELIZA

s/. \* I'm (depressed|sad) . \*/I AM SORRY TO HEAR YOU ARE  
\1/

s/. \* I'm (depressed|sad) . \*/WHY DO YOU THINK YOU ARE \1/

s/. \* all . \*/IN WHAT WAY/

s/. \* always . \*/CAN YOU THINK OF A SPECIFIC EXAMPLE/

# Ressources utiles et exercices

Des ressources utiles pour approfondir les expressions régulières sont disponibles sur la page Moodle, avec d'autres exemples que ceux vus dans le cours.

Vous pouvez maintenant pratiquer avec les exercices disponibles sur Moodle.