

**Plateforme Web pour le traitement automatique des
langues: Analyse morphologique.**

**Introduction à l'étiquetage morphosyntaxique, la détection
de la langue et les correcteurs orthographiques.**

Application sur le kabyle.

Tiġerġert Web i i usemsel awurman n tutlayin: Tasleqt tasnalyant.

**Tazwert yer ucrad asnalyan aseddasan, tifin n tulayt, d
yimsejtiyen n tira.**

Asemres yef teqbaylit.

Asemsel awurman n tutlayin: Tasleqt tasnalyant
Traitement automatique des langues: Analyse morphologique

Agbur

Avertissement	6
Préparation de l'environnement d'exécution	7
Organisation des fichiers des dossiers inclus:	8
Résumé	10
Agzul	11
Ahric I: Asemsel awurman n tutlayt	13
Tazwert yer usemsel awurman n tutlayin	13
Tiyula n usemres n usemsel awurman n tutlayin	13
Tiyula & yifecka n tesleqt	14
Assay gar tigzi n tmacint d tesnilest	14
Assay gar tigzi n tmacint, almad awurman, almad Iqayen	14
Assay gar unagraw n tira akked talya/taseddast	15
Ahric II: Acrađ asnalyan aseddasan	18
Iswi n ucrađ asnalyan aseddasan	18
Tagrumma n yisefka isnalyanen iseddasanen	20
Tasleqt n tegramma n yisefka isnalyanen iseddasanen	21
Tagrumma n tecrađ tisnalyanin tiseddasanin	23
Talyiwin n wawal & taseddast n tefyirt	24
Acrađ asnalyan aseddasan s Keras	25
Betju asnalyan aseddasan	25
Alguritm n betju asnalyan aseddasan:	25
Ticrađ tisnalyanin tiseddasanin yettwacqedcen	26
Betju n tegramma n yisefka i ulmad s tallelt	26
Asbadu n talyiwin n wawal	26
Tukksa n tecrađ seg tefyirt	27
Betju n tefyar yettwacerden yer snat n tegrammiwin	27
Tiririt n yisefka yer talya n yimdanen	27
Asbadu n uzeṭṭa n yinurunen	27
Asleymu n uzeṭṭa n yinurunen	28
Aktazel n tneyruft	28
Asemres n tneyruft tasnalyant taseddasant n keras	30
Acrađ asnayal aseddasan s usemres n CRF (Scikit-learn)	32
Talya n yisefka n tegramma n tecrađ akked ulmad	32
Asleymu n yisefka	33

Asem sel awurman n tut layin: Tasleqt tasnalyant
Traitement automatique des langues: Analyse morphologique

Alguritm Ibfgs	33
Alguritm I2SGD	34
Alguritm AP.....	35
Alguritm PA.....	36
Alguritm arow.....	36
Asemres n tneyruft n ulmad asnalyan aseddasan s CRF	37
Tawuri n tizdit deg tseddast n tutayt taqbaylit	40
Aħriċ III: Tifin tawurmant n tutlayt.....	42
Tazwert.....	42
Asader n tegrummiwin n yisefka seg Tatoeba	42
Asem sel uzwir n tegrummiwin n yisefka.....	42
Aheggi n tegramma n tefyar i usekyed	43
Tasleqt n yisekkilen - unigram:.....	44
Tasleqt n yisekkilen s Bigrams (amsedfer n 2 n yisekkilen).....	46
Asleymu n tegramma n yisefka i tifin n tutlayt.	46
Tifin n tutlayt n tefyirt	47
Asismel n tefyar s tutlayt.....	47
Aktazel n tneyrufin: Tiseddi n ucrad.....	48
Aħriċ III: Imseyytiyen n tira i teqbaylit	50
Tazwert.....	50
Aheggi n twennaqt n tneflit	50
Talya d usiley n wawalen deg teqbaylit.....	51
Tayessa n yimawalen Hunspell.....	51
Afaylu n umawal:.....	51
Afaylu n yilugan.....	51
Tayessa n ufaylu n yilugan :.....	54
Taseddast n yilugan Hunspell.....	55
Asiley n talyiwin s tmerna n yiwsilen, tukksa ney abeddel n yisekkilen.....	55
Yiwen n wuzwir.....	55
Yiwen n wudfir.....	56
Yiwen n wuzwir d yiwen n wudfir.....	58
Aħriċ V: Tiyertyert Web	65
Tazwert.....	65
Aheggi n twennaqt n usemres.....	65
Asbeddi deg Linux Ubuntu 20.04 LTS	65
Asbeddin Postgres	65

Asem sel awurman n tutlayin: Tasleqt tasnalyant
Traitement automatique des langues: Analyse morphologique

Asbeddi n wkhtmltopdf.....	65
Tuqqna yer tyeryert:.....	65
Umuyen n ugrudem n tyeryert.....	67
Umuy “Tasnalya taseddast”	67
Umuy “Aseyti n tira”.....	68
Umuy “Tifin n tutlayt”	69
Umuy “Tasleqt n tjerrumt”	69
Umuy tawila:	70
Amek ara nernu tafyirt?	70
Amek ara ternuq ticraq tisnalyanin tiseddasanin i tefyirt?	72
Aħriċ n taggara:	75
Awal n taggara.....	75
Timsisyal	76
Timsirin yef ulmad awurman.....	77
Ifecka & tutlayin n usihel.....	77
Amawal.....	78
Tamaziyt – Tafransist.....	78
Tafransist – Tamaziyt.....	79
Tagrumma n tecraq tisnalyanin tiseddasanin	82

Avertissement

Ce document est principalement destiné aux étudiants des départements d'informatique et de langue et culture amazighe et toute autre personne initiée à la linguistique informatique et souhaitant s'investir dans le domaine du traitement automatique des langues appliqué aux langues amazighes. Pour les informaticiens, ce sera un outil pédagogique pour les introduire dans le monde du TAL appliqué à la langue kabyle et particulièrement l'apprentissage automatique. Il sera aussi un outil pour aider les étudiants des départements de langue et culture amazighe pour les introduire au monde de la linguistique des corpus et à prendre conscience de l'importance des corpus linguistiques structurés dans l'intégration numérique des langues amazighes dont le kabyle en appliquant ces algorithmes sur leur propres corpus afin de produire des modèles de plus en plus précis.

Nous invitons le lecteur à consulter le lexique inséré à la fin du document afin de mieux appréhender le sujet. La terminologie employée est issue de plusieurs documents et manuels traitant de lexique kabyle/amazigh et publiés depuis plus de quatre décennies. Référez-vous aussi aux différentes références bibliographiques indiquées en bas de ce présent document et ayant aidé dans la réalisation de ce présent travail.

Ce travail a pris plusieurs mois et couvre plusieurs domaines du TAL. Dans cette étude, nous présenterons uniquement une partie traitant de morphologie et de syntaxe. Ce travail ne prend en charge que la variante kabyle pour des raisons de syntaxe et de morphologie mais aussi de corpus en guise de matière. Les algorithmes insérés ne sont applicables que sur des corpus et textes de langue kabyle, transcrits en caractère latin et respectant les règles syntaxiques enseignées dans les écoles et les universités en Algérie. En dehors du trait d'union utilisé pour accompagner les affixes, seul l'alphabet officiel du kabyle et la ponctuation sont pris en charge. Toutefois, il est possible d'aménager le corpus des affixes, l'algorithme de segmentation morphosyntaxique, et l'ensemble des caractéristiques morphologique décrites et présentées dans la présente étude et les appliquer sur un autre corpus d'une autre langue amazighe mais transcrive uniquement en caractère latin.

Le corpus morphosyntaxique employé et les algorithmes utilisés sont inclus avec le présent document dans des fichiers séparés. Une explication est donnée sur le contenu des dossiers.

Pour des raisons de tailles de fichiers, les fichiers de données pour l'apprentissage de la détection des langues et les fichiers de modèle d'apprentissage morphosyntaxique générés ne seront pas inclus. Les fichiers de données peuvent être téléchargés via un programme que nous avons inséré. Les fichiers de modèles (morphosyntaxique ou modèle de langue) peuvent aussi être générés.

Préparation de l'environnement d'exécution

Les algorithmes présentés dans ce document sont écrits en Python. Le choix de ce langage est du principalement la disponibilité de bibliothèques libres de calcul numérique et d'apprentissage automatique. Python est un langage multiplateformes. Il s'exécute sur Windows, Linux, MacOS et autres. Ci dessous les avantages majeurs qu'offre le language Python:

Bibliothèques étendues: Python dispose nativement de bibliothèques traitant des expressions régulières, la génération de documentation, les tests unitaires, les navigateurs Web, le filetage, les bases de données, le CGI, le courrier électronique, la manipulation d'images, ...etc. Ce qui évite d'écrire du code long.

Extensibilité : Il est possible de faire appel à des bibliothèques écrites dans d'autres langages comme le C.

Intégrabilité : Il est possible d'intégrer du code C++ dans un programme Python.

Intelligence artificielle et IoT (Internet Of Things): La plus part des frameworks traitant d'apprentissage automatique et d'IoT sont disponibles en Python. Nous citons : TensorFlow, Keras, Pytorch et autres.

Opensource : Python est libre. La plus part des éditeurs de ce langage sont aussi disponibles en Opensource.

Pour pouvoir exécuter les programmes inclus dans la présente étude pour l'étiquetage morphosyntaxique basé sur keras/Tensorflow, et CRF-sklearnsuite il faudra préparer l'environnement. Les instructions ci-dessous sont décrites pour une installation sous Windows 7 (ou plus) 64 bits, disposant de 8 GO de RAM et d'un espace libre de 20GO. Il est préférable de disposer d'une carte GPU pour augmenter le nombre de couches et d'epochs du réseau de neurones lors de l'entraînement des données et la génération du modèle morphosyntaxique avec Keras/Tensorflow. Pour les autres programmes, aucune restriction de version n'est requise.

- 1- Disposer d'un environnement 64 bits (Windows 7 ou 10 64 bits)
- 2- Allez sur le site <https://www.python.org/downloads/release/python-360/> et télécharger Python 3.6 64 bits. Lors de l'installation, incluez l'outil pip. Pip est un outil qui nous servira à installer d'autres bibliothèques.
- 3- Tensorflow : Lancez la ligne de commande et tapez la commande **pip install tensorflow==1.5.0**. Tensorflow sera utilisé comme backend pour entraîner les réseaux de neurones.
- 4- Keras : Lancez la ligne de commande et tapez la commande **pip install keras==2.2.4**. Keras sera utilisé pour définir la structure du réseau de neurones.
- 5- Matplotlib: Lancez la ligne de commande et tapez **pip install matplotlib== 3.3.3** , Matplotlib sera utilisé pour visualiser les données (test des modèles et autres statistiques d'analyses du corpus)
- 6- sklearn: Lancez la ligne de commande et tapez **pip install sklearn== 0.19.0** , sklearn sera utilisé pour vectoriser les données, les transformer en entiers, et les décoder lors de l'étiquetage.
- 7- numpy: Lancez la ligne de commande et tapez **pip install numpy== 1.19.4**, numpy sera utilisé pour définir les tensors (vecteurs).
- 8- sklearn_crfsuite : Lancez la ligne de commande et tapez **pip install sklearn_crfsuite** : Utilisé pour générer des modèles basé sur CRF sklearn_crfsuite
- 9- Autres: **pip install pylab networkx**

Asem sel awurman n tutlayin: Tasledt tasnalayant
Traitement automatique des langues: Analyse morphologique

Organisation des fichiers des dossiers inclus:

Les différents programmes et corpus produits dans le cadre de cette étude sont opensource.

Le dossier **Keras** contenant l'étiquetage morphosyntaxique se basant sur Keras contient des programmes et des fichiers de données. Le modèle entraîné et l'encodage des étiquettes peuvent être générés en exécutant le fichier Model_generation.py. Ci-dessous la description du contenu du dépôt:

- 1- corpus-kab.txt : Contient le corpus d'apprentissage. C'est un ensemble de phrases de langue kabyle manuellement étiqueté en utilisant l'ensemble des étiquettes décrites dans le fichier tagrumma_n_tecraq.xlsx.
- 2- tagrumma_n_tecraq.xlsx : Décrit l'ensemble des étiquettes morphosyntaxiques. Cet ensemble est établi à partir des documents traitant de langue kabyle. Pour des raisons de précision, la classe des verbes est traitée par aspect.
- 3- Corpus_stat_analysis.py : Permet de fournir quelques analyses sur le corpus morphosyntaxique manuellement étiqueté.
- 4- postag_graph.py : Programme Python qui permet de représenter graphiquement la succession des classes grammaticales dans un corpus morphosyntaxique.
- 5- affixescolles.txt : Corpus des affixes en kabyle établi sur la base des documents traitant de syntaxe publiés et les pratiques syntaxiques en Kabylie (romans, journaux..).
- 6- morpho_syntax_segmentation.py : Programme Python qui permet d'appliquer la segmentation morphosyntaxique sur la langue kabyle. Il utilise le corpus des affixes imqimen_uzwiren_udfiren.txt et les règles syntaxiques du kabyle.
- 7- Model_generation.py: Programme Python qui permet d'entraîner le corpus morphosyntaxique sur le réseau de neurone défini.
- 8- tagger.py : Programme Python qui permet d'étiqueter de nouvelles phrases.
- 9- brut_text.txt : Contient du texte brut de langue kabyle. Une phrase par ligne. Il est utilisé pour tester la segmentation traité par le programme morpho_syntax_segmentation.py
- 10- tokenized_text.txt : Phrases segmentées générées à partir du fichier brut_text.txt en exécutant le programme morpho_syntax_segmentation.py

Le dossier PostagCRF contient les programmes et autres fichiers traitant d'étiquetage morphosyntaxique se basant sur CRF scikit-learn :

1. lbfgs.py: Programmes d'apprentissage générant le modèle morphosyntaxique en utilisant l'algorithme Gradient descent using the L-BFGS method
2. l2sgd.py: Programmes d'apprentissage générant le modèle morphosyntaxique en utilisant l'algorithme Stochastic Gradient Descent with L2 regularization term
3. ap.py : Programmes d'apprentissage générant le modèle morphosyntaxique en utilisant l'algorithme Averaged Perceptron

Asem sel awurman n tutlayin: Tasleđt tasnalijant
Traitement automatique des langues: Analyse morphologique

4. pa.py : Programmes d'apprentissage générant le modèle morphosyntaxique en utilisant l'algorithme Passive Aggressive (PA)
5. arrow.py: Programmes d'apprentissage générant le modèle morphosyntaxique en utilisant l'algorithme Adaptive Regularization Of Weight Vector (AROW)
6. Fichiers: lbfgs-tagger, l2sgd-tagger.py, ap-tagger.py, pa-tagger.py, arow-tagger sont des programmes qui permettent de faire la segmentation morphosyntaxique, utilisent les modèles générés par les programmes d'entraînement et génèrent des phrases étiquetées.
7. Fichiers model-lbfgs.joblib, model-l2sgd.joblib, model-ap.joblib, model-pa.joblib, model-arow.joblib sont les modèles générés à partir de l'entraînement sur le corpus. Il sont générés respectivement par les programmes d'entraînement, lbfgs, 2lsgd, ap, pa et arow des fichiers 1, 2, 3, 4 et 5.
8. corpus-kab.txt : Contient le corpus d'apprentissage. C'est un ensemble de phrase de langue kabyle manuellement étiqueté en utilisant l'ensemble des étiquettes décrites dans le fichier tagrumma_n_tecrad.xlsx.
- 11- brut_text.txt : Contient du texte brut de langue kabyle. Une phrase par ligne. Il est utilisé pour tester la segmentation traitée par le programme d'étiquetage.
9. tokenized_text.txt : Phrases segmentées.
10. Tagged_text : Fichier qui contient les phrases étiquetées générées par les programmes d'étiquetage.
11. affixescolles.txt : Corpus des affixes en kabyle établi sur la base des documents traitant de syntaxe publiés et les pratiques syntaxiques du kabyle (romans, journaux..).

Le dossier **Lang_detection** contient l'ensemble des programmes et fichiers traitant de détection de la langue:

1. download.py: contient le programme permettant de télécharger des fichiers de langues à partir du corpus Tatoeba. A noter que la langue kabyle est présente sur Tatoeba avec un corpus de plus de 500 000 phrases. Après téléchargement, il faut décompresser ces fichiers.
2. Prepare_langues_files.py: Ce programme permet de traiter les fichiers précédent en gradant que les phrases. Les autres informations sont omises.
3. Prepare_learning_corpora.py: Programme qui génère le corpus d'apprentissage en un seul fichier en y insérant la langue.
4. Prepare_test_corpora.py: Programme qui permet de générer un corpus pour tester la classification.
5. generate_model.py: Ce programme permet de générer le modèle de détection de langue.
6. predict_sentence_language.py: Ce programme permet de faire la prédiction de langue d'une phrase données en entrée.
7. classify_sentences_by_language.py: Ce programme permet de classer des phrases d'un corpus par langue.
8. Fichiers de données: Ils sont obtenus en exécutant dans l'ordre les programmes 1,2,3 et 4. Pour le programme 1, une fois les fichiers téléchargés, il faut procéder à leur décompression.

Résumé

Le traitement automatique des langues (TAL) est un domaine de la linguistique informatique qui consiste à utiliser la machine pour traiter, comprendre et générer des données linguistiques.

L'étiquetage morphosyntaxique, et la détection de la langue font partie du traitement automatique des langues. L'étiquetage morphosyntaxique permet de déterminer la classe grammaticale d'un mot dans une phrase en utilisant des programmes informatiques. L'étiquetage morphosyntaxique est une étape importante dans plusieurs domaines du TAL comme le résumé des textes, les questions/réponses, la lemmatisation, la racinisation, l'extraction et la recherche d'information dans les textes...etc.

La détection automatique des langues quant à lui, permet de reconnaître, extraire et classer des textes par langue pour leur appliquer des traitements adéquats, car sur le plan morphologique, les traitements diffèrent d'une langue à une autre. Le contenu numérique aujourd'hui est de plus en plus gros, la recherche et la classification de l'information en ligne ne pourra plus se faire de façon manuelle. C'est pour cela qu'un traitement permettant de détecter la langue est important et appliquer ensuite les traitements morphologiques adéquats tel que la lemmatisation, la dérivation, la racinisation...

L'enseignement et le passage de tamazight de l'oral à l'écrit étant relativement récents ont eu raison de l'absence de corpus morphosyntaxiques libres comparativement à d'autres langues. Ce qui nous a poussé à construire notre propre ensemble d'étiquettes morphosyntaxiques et un corpus de phrases manuellement étiquetées. Un projet de production de corpus de phrases libres de langue kabyle a aussi été initié depuis plus de 5 ans et qui nous a servi de matière de réaliser une partie de ce travail. Ces corpus nécessitent une revue continue pour les améliorer que ce soit en matière pour des raisons de diversité des formes ou du vocabulaire.

Dans cette étude, nous présenterons des concepts généraux liés au traitement automatique des langues, l'impact du système d'écriture sur la morphologie et la syntaxe, l'ensemble des étiquettes employées pour l'étiquetage, l'analyse statistique du corpus morphosyntaxique utilisé pour l'apprentissage automatique, et on termine par la démarche, l'ensemble des prétraitements et traitements nécessaires appliqués au corpus employé pour générer et utiliser des modèles d'étiquetage morphosyntaxique et détection de langue basé sur l'apprentissage profond et autres techniques.

La dernière partie, présente l'interface de la plateforme web. Cette interface permet de collecter divers types de corpus et donne accès à plusieurs traitements dont l'analyse morphosyntaxique, l'analyse morphologique et la détection de la langue.

Clés : Etiquetage morphosyntaxique, apprentissage profond, intelligence artificielle, linguistique, détection de langue, unigram, bigram, CRF, Tamazight, kabyle, Tensorflow, *Traitement automatique des langues, keras, Python, Matplotlib*.

Agzul

Asemsel awurman n tutlayin d tayult n tesnilest tasenselkamt i issemrasen aselkim i usemsel, gezzu, akked uslali n yisefka isnilsanen.

Acraç asnalyan aseddasan d yiwen uħric seg yiħriċen n usemsel awurman n tutlayin i d-ilehhun s usismel n wawalen n tefyar yer yismilen inejrumen s usemres n wahilen isenselkamen. Acraç asnalyan aseddasan d amecwar meqqren deg ddeqs n tyula i issemrasen asemsel awurman n tutlayt am usegzel n yiđri森, isteqsiyen d tririyin, tukksa d unadi n telyut deg yiđri森...atg.

Deg tezrawt-a, ad d-nsissen sumata amawal i icudden yer usemsel awurman n tutlayin, assay gar unagraw n tira akked tesnalya d tseddast, tagrumma n tecraç tisnalyanin tiseddasanin i yettwasqedcen i ucraç, tasleqt n tegrumma n yisefka isnalyanen iseddasanen i yettwasmersen i ulmad awurman, sakin, ad tt-nfak s tarrayt, asemsel uzwir,d usemsel i ilaġen i uslali d usemres n tneyrufin n ucraç asnalyan aseddasan yebnan yef ulmad alqayan i tutlayt taqbaylit.

Tifin tawurmant n tutlayin seg tama-is, yettaeqal tutlayt, itekkes-d iđri森, dayen yessismil iđri森 s tutlayt akken ad nizmir ad nsemres fell-asen isemsal i iwulmen, acku seg tama n tesnalya, isemsal mgaraden seg tutlayt yer tayed. Agbur umdin ass-a yettnerni s waṭas, anadi n telyut srid deg Internet akked usismel-is mačhi dayen ara yettwaxedmen s ufus. Xef waya i ilaġ usemsel n tifin n tutlayin akken ad nizmir ad nsemres isemsal isnalyanen i ilaġen am unadi nwawalen iherfiyen, asuddem, akked unadi n yiżuran n wawalen...

Asemed akked tira n tmaziyt d tarmit i yebdan iseggasen-a ineggura. Xef waya ur nettaf ara isefka isnalyanen iseddasanen ney tigrummiwin-nniđen n yisefka isnilsanen. Dayen i aġ-yegħġan nhegga-d tagrumma-nney n yisefka isnalyanen iseddasanen akked tegruma n tefyar n tutlayt tamaiyt (taqbaylit) akken ad d-nexdem tazrawt-a. Tagrumma n tefyar n tutlayt tamaiyt d asenfar i yebdan ugar n 5 n yiseggasen aya. Tigrummiwin-a n yisefka mazal-itent meżżejjit ama si tama n talyiwin tiseddasanin ney amdan n wawalen i nsemres deg-sent.

Tisura: *Acraç asnalyan aseddasan, almad lqayen, tigzi n tmacint, tasnilest, tifin n tutlayt, unigram, bigram, CRF, taqbaylit, Tensorflow, asemsel awurman n tutlayt, keras, Python, Matplotlib.*

Aḥric I : Asemsel awurman n tutlayt

Aħric I: Asemsel awurman n tutlay

Tazwert yer usemsel awurman n tutlayin

Asemsel awurman n tutlayt d tayult n tigzi n tmacint i d-ilehhun deg tesleqt, gezzu akked uslali ney asnulfu n tutlayt n umdan ama d iđrisen ney d tayect. Aya yettwaxdam i ferru n ddeqs n wuguren, maca dayen i usnerni n temygawt gar umdan d tmacint. Tayult-a, tessawal i ddeqs n tussniwin-nniđen. Sumata, llan tlata n yiħriċen igejdanen ideg asemsel awurman n tultlayin iggar-d iman-is:

Aeqal n tayect:

Aħric-a ileħħu-d deg temsal n temsiselt akked tesniselt. Dagi, aselkim ad yizmir ad d-iyer adrīs, ney ad d-yaru adrīs i yesla s tutlayt n umdan. Aeqal n tayect yettwasemras deg waṭas n yifecka am GPS.

Gezzu n tutlayt n umdan:

Aselkim ad yizmir ad yegzu inumak n wawalen akked tefyar akken ad d-isumer tiririt, ad isuqel seg tutlayt yer tayed, ad isegzel iđrisen, ad d-isuffey talijut tagejdant seg yiđrisen ...atg.

Asnulfu n tutlayt n umdan:

Aselkim ad yizmir ad isewzel iđrisen, ney ad yessufey kra n telyut ney tikta tigejdanin seg-sen. Yal asemsel ileħħu-d deg yiħet n tayult n tesnilest.

Amedya: acrad asnalyan aseddasan yelha-d kan deg tesleqt n talja akked tseddast, ama deg wawalen ney deg tifyar.

Tiyula n usemres n usemsel awurman n tutlayin

Asemsel awurman n tutlayin yebda yef rebea n wanawen:

Taseddast	Tasnamka	Inaw	Tayect
Anadi n wawalen iħerfiyen	Tasnamka tamawalant	Asegzel awurman	Adris yer tayect
Bettu asnalyan aseddasan	Tasuqqilt tawurmant	Tukksa n wawalen/tifyar	Tayect yer uđris
Tilisa n tefyar	Anadi n tyawsilin	tigejdanin seg yiđrisen	
Anadi n yizurani n wawalen	yettusemman	Tasleqt n yinaw	
Tasleqt n tseddast	Aslali awurman n yiđrisen	Tasleqt n yiħulfan	
Tasleqt n tunqiqin	Aeqal n yisekkilen		
	Tasleqt n tririyn		

Tiyula & yifecka n tesleqt

Ifecka n tesleqt bdan yef xemsa n taggayin. Yal taggayt tlehhu-d deg yiwen uswir n tesnilest. Tafelwit ddaw-a temmal-d tiyula n tesleqt d yifecka-is seg uswir afessas yer uswir azayan:

Aswir n tesleqt	Tayult tasnilsant	Taggayt	Ifecka n tesleqt
1	Tamsiselt/tasniseit	Imesli, afunim, tuniqqt, tasusrut	Ifecka n ueqal d uslali n tayect, ifecka n beetu n tayect.
2	Tasnalya / Tasleqt tamawalant	Awalen, iwshilen, Tawsit tanejrumt, amdan anejrum, addad n yisem, Taseftit, asuddem.	Ifecka n beetu asnalyan Ifecka n unadi n wawalen iherfiyen Ifecka n unadi n yizuran n wawalen Ifecka n unadi n tlisa gar tefyar Ifecka n ucrad asnalyan aseddasan.
3	Taseddast	Tifyar, asumer, awal	Imasladen n tseddast & tjerrumt
4	Tasnamka	Inumak n wawalen d tefyar	Ifecka n tukksa n ubnubek n yinumak n wawalen
5	Tapragmatit	Anaw n yinaw, ihulfan	Imasladen n yihulfan, ...

Assay gar tigzi n tmacint d tesnilest

Asemsel awurman n tutlayt d tayult i d-yekkan seg usdukkel n tesnilest d tigzi n tmacint.

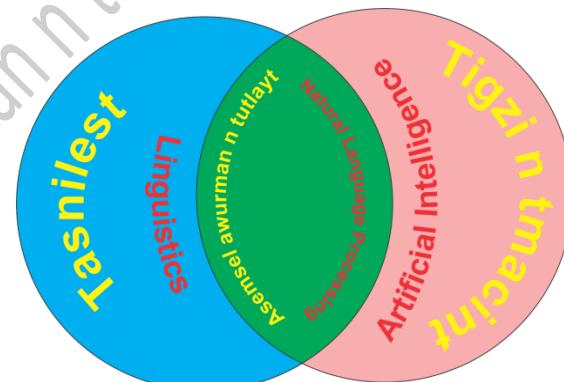
D asemres n yilugan n tesnilest s useqdec n uselkim, i d-yesnulfan tayult n usemsel awurman n tutlayt. Aya yegber merra tiyula n tesnilest: Seg yimesla arma d inumak n wawalen, tifyar, d yiqrissen.

Tigzi n tmacint d tagrumma n tetiknikin i yebnan yef tsenselkint. Tesseqdac ladja tussniwin am tusnakt.

Assay gar tigzi n tmacint, almad awurman, almad lqayen

Tigzi n tmacint d ahric amaynut meqqren deg tsenselkint. Tigzi n tmacint d asemres n usihel asenselkam deg yinurar n tussniwin ama i tesleqt n yisefka, ney anadi n tifrat i ddeqs n wuguren i icudden yer tudert n umdan. Ahric-a, tikwal, yessemras titiknikin n usleymu n yisefka.

D wagi umi qqaren almad awurman. Ticki tatiknikt tessemras izedwan n yinurunen, ahric-a qqaren-as almad lqayen.



Assay gar unagraw n tira akked talya/taseddast

Anagraw n tira yur-s assay meqqren akked tseddast. Talya n yisekkilen tetħettim talya n wawal. Ma newwi amedya n unagraw n tira n taerabt, wagi ur yegbir ara talya n usekkil meqqren/meżziyen am netta am unagraw n tira n tfinay. Rnu yur-s, qrib yal asekkil deg taerabt yesea rebea n talyiwin: Talya tilellit, talya n tazwara, talya n tlemmast, talya n taggara.

Amedya: Assekil ب



Ugur-a yella-d imi isekkilen-is ttemyentaden deg tira (agglutinant).

Rnu yer waya, anagraw n tira n taerabt ur yegbir ara tiyra. Xef waya i seqdacen 4 n yifeskilen akken ad εiwenen deg tukksa n ubnubek amsislan/asnislan ney tukksa n ubnubek asnalyan aseddasan: "Fetha", "demma", "kesra", akked "ssukun". Tussda deg taerab tettwaru s ufeskil n "cedda". Deg unagraw n tira n taerabt, yella dayen usemres n "tanwin". Aya yettwaru s tyugiwin n "Fetha", "Demma", akked "Kesra". Ma nsuddes ifeskilen akked talyiwin n yiwen n usekkil kan am ߲, lehsab ad yawed yer wugar n 50 n talyiwin i yiwen kan n usekkil. Deg uhric n titiknulujiyn timaynutin, yal asekkil tlaq-as tengalt Unicode. Dayen ara yerren asemsel awurman n tutlay yettwarun s unagraw n taerabt, d tamsalt i iweeren nezzeb am ucrad asnalyan aseddasan, betu asnalyst aseddasan, anadi n wawalen iherfiyen ney izuran n wawalen, aeqal d uslali n tayect..atg..

Amedya: betṭu asnalyan aseddasan deg taerabt



Uguren-a i d-yettisin deg tutlayin yettwarun s unagraw n tira n taerabt, ur d-kkin ara seg tseddast, maca kkan-d seg talya n usekkil.

Ma nezzi yer nagraw n tira s tfinay, yef wakken ur yegbir ara talya n usekkil meqqren, ddeqs n yisemsal i icudden yer tesleqt n yisem am unadi n tyawsiwin yettwasemman deg yiqrissen, ur ttizmiren ara ad ttwaxedmen s usemres n yilguritmen.

Amedya: Tifyar ddaw-a gebrent abnubek i d-yekkan seg talya n usekkil.

Tafyirt “ .Θ%ΕΖ.Ε% :ΙΙΞΧΙ +ΞΞΞ:ΧΨΥ.” izmer
ad d-tmudd:

- aseqqamu unnig n timmuzya, ney
 - Aseqqamu Unnig n Timmuzya

Tafyirt, “Ա.Խ.Կ ։ԿՀԱ.Ծ ։ԾԾ。” izmer ad tmudd:

- Walay ayilas ass-a, ney
 - Walay Ayilas ass-

Deg umedya amezwaru, s usemres n unagraw n tira aliñini, ad naru: Aseqqamu Unnig n Timmuzya. Wagi d tuddsia i yettusemman s yisem-a. Maca s tifinay ur nettizmir ad nzer aya ma nga kan tasleqt n talya n usekkil.

Deg umedya wis sin, s tira s usemres n unagraw n tira n tfinay, ur nettizmir ara ad nsemgired gar yisem n umdan (isem uzzig) d win n uyersiw (isem amagnu).

Tuget n tsekla i yettwarun ass-a, d tin yezdin yef tdamsa, tussna, tatiknulujit, ...atg. Xef waya i nettaf deg-sen ddeqs n wawalen uzzigen ney tiqawisin yettwasemman am yismawen n yimdanen, tuddsiwin, ticrad, ifarisen, imedqan d wiyaq. Ma yella ur nettizmir ara ad nsemgired s talya gar yismawen uzzigen d yismawen imugna, tasleqt tasnalyant taseddasant n yiqrissen yettwarun s yingrawen-a (Tifinay, Taerabt) d ayen ara ietlen i ferru ama s tmerna n yisekkilen imaynuten d talyiwin-nniñen i yisekkilen, abeddel deg tseddast, ney s ueggi n tegrummiwin timeqqrarin n yisefka isnilsanen am tegrummiwin n yismawen uzzigen d tyawsiwin yettwasemman.

Anagraw n tira alaçini ney agraylan, d anagraw yessifsusen ddeqs n tlufa i icudden yer usemsel awurman n tutlayt. Tuget n wuguren fran yakan i tutlayin yettwarun s unagraw-a. Dayen ara yeggan asemres-nsen yef tutlayt taqbaylit fesus, maca yeqqim-d kan ad d-nheggi tigrummiwin n yisefka isnilsanen iseddasanen i usleymu n yilguritmen, aslali d usemres n tneyrufin-a deg wannar n umadal umdin am tsuqilt tawurmant ney aeqal n tayect.

Aḥric II: Acrad asnalyan aseddasan

Aħric II: Acrad asnalyan aseddasan

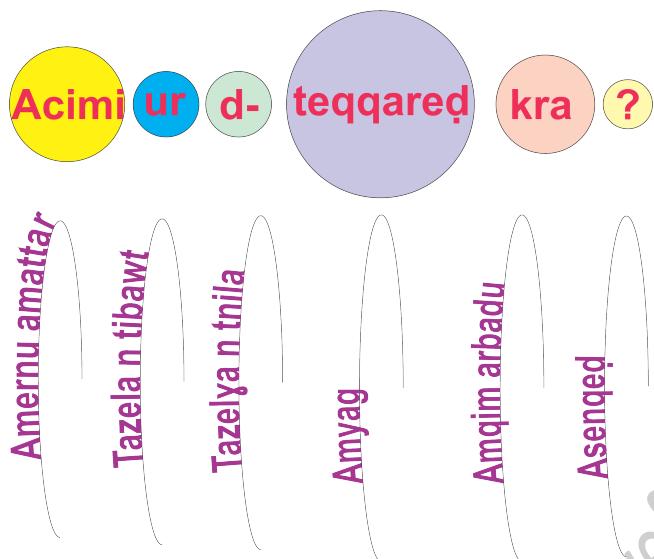
Iswi n ucrad asnalyan aseddasan

Denys Amesniles (Dionysius Thrax) n Aleksandrya (100 n yiseggasen uqbel talalit n Sidna Eisa), nej ahat d wayed aħħal aya, yura adlis yef tjerrumt n Tegrigit, yessegzel-d deg-s tussna tasnilsant n tallitis. Axeddim-a imudd-d aħas n tesnawalt tasnilsant n wass-a, gar-asen awalen am tseddast, tadiftungit, aklitik akked tserwest. ad naf dayen aglam n tmanyā n yiħriċen n ummeslay nej ismilen inejrumen: Isem, Amyag, amqim, tanzejt, amernu, tazelya n tuqqna, amayun, akked umagrad. Xas wamma imussnawen iqburn mudden-d ismal-nsen inejrumen akka am Aristote, d tmanyā-nni n yismilen n Thrax i yuvalen d lsas n usegħlem n tuget n tutlayin n Turuft i 2000 n yiseggasen yer zdat.

Schoolhouse Rock d yiwen n wahil ayerfan n usegħmi deg tiliżri deg Marikan deg yiseggasen n 1970. Aħric-nni n tjerrumt yegħiġi tuyac yef 8 n yismilen inejrumen. Xas akken 8 nni n yismilen n Thrax beddlen yef wayen i nessen ass-a, maca qqimen ugar n 2000 n yiseggasen ttwaseqdacen. D ajen i d-yemmalen arkad deg tutlayin i ttimeslayen yimdanen. Kra n yismilen inejrumen-nni den (nej tagħġi tis-seddasan) laqen, acku rennun-d aħas n yisallen yef wawal akked wawalen i t-id-iqurben.

Ticki necređ iferdisen (awalen) n tefyirt, iswi d tifin n usmil anejrum n yal awal. Ma neżra awal ma yella d isem nej d amyag, ad nzer d acu ara d-yekken zdat-s nej deffir-s (amyag ur t-id-tzewwir ara tenzejt, isem ur t-id-zewwir ara tzelya n wurmir), ajen ara yeġġen aċraq ASNALYAN aseddasan d tigejjid deg tesleqt.

Amedya 1: Acimi ur d-teqqareq kra?



Maca, iswi-nnidен n ucrađ asnalyan aseddasan, d tukksa n ubnubek asnalyan aseddasan ticki gdañ wawalen.

Amedya 2:

Acu ara iniy i wid ara ixelsen ayen ur ččin ara?

Deg umedya-a, awal "ara" yettbeddil asmil anejrum.



Di tazwara yusa-d d tazelya n wurmir aħerfi, deg tlemmast d tazelya n umayun urmir, deg taggara d tazelya n tibawt tuđfirt.

Amedya 3: D netta i yennan i baba-s.

Deg umedya-a, awal "i" yusa-d d amassay deg tazwara, d tanzejt deg taggara.



Acraç asnalyan aseddasan yessishil ahettec deg yiðrisen imi nezra asmil-is anejrum. Ticki nessen asmil anejrum n wawal, anadi n yisuddimen, awal aherfi ney azar n wawal ad yifus acku talyiwin n usudem n yimyagen d yismawen ttwassnent.

Tagrumma n yisefka isnalyanen iseddasanen

Deg tezrawt-a, nsemres ugar n 1633 n tefyar. Tifyar-a d tukkisin i d-yettwakksen seg waṭas n yiybula yemgaraden am yiymisen, ungalen, iðrisen n tuyac, iżedwan n tmetti, d yidilisen-nniđen yuran s teqbayli. Gar yiybula-a, izmer ad d-nadder: "La Dépêche De Kabylie", "Tiyremt", idlisen n Emer Mezdad, Hmed Nekkar d wiyađ. Yal awal ney d aferdis ilelli deg tefyirt, teṭṭafar-it-id tecredt-is ney asmil-is anejrum. Aħric-a ur d-ileħħu ara deg twuri n wawalen, maca deg usmil-is kan anejrum.

Amedya:

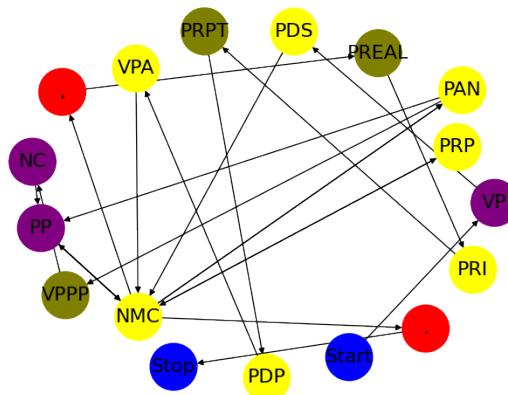
Yefka/VP -as/PSV -t/PSV i/PRP yemma/NMC -s/PAN ./.

Yerna/VP -d/PDS ussan/NMC -agi/PAN ieddan/VPPP yiwen/NC n/PP udlis/NMC n/PP tmaziyt/NMC ./, d/PREAL ayen/PRI ara/PRPT d-/PDP yernun/VPA abayur/NMC i/PRP tutlayt/NMC -agi/PAN n/PP tyemmat/NMC ./.

Anida:

- VP: Amyag izri
- PSV: Amqim udfir n umyag.
- PRP: Tanzejt
- NMC: Isem amagnu.
- PAN: Amqim awsil n yisem
- VPPP: Amayun n yizri ibaw
- PRI: Amassay
- PP: Tazelya n wayla
- PDP: Tazelya n tnila tuzwirt
- PREAL:
- PDS: Tazelya n tnila tuđfirt.

Xas wama tutlayt n umdan dayen iweeren nezzeħ ticki tasleqt tleħħu-d deg wawalen d yimesla s timmad-nsen, maca deg tilawt llant tneyrufin yeffren anida tasleqt tezmer ad tmudd kra n telyut i yeffren yef yimdanen. Azejt-a ddaw-a, yeskan-d amsedfer n yismilen inejrumen n snat tefyar i d-nmudd nnig-a:



Tasleqt n tegrumma n yisefka isnalyanen iseddasanen

Amdan n tefyar: 1633

```
amdan_n_tefyar=0
for tafyirt in open("corpus-kab.txt",encoding='utf-8'):
    amdan_n_tefyar+=1
print ("Amdan n tefyar: ",amdan_n_tefyar)
```

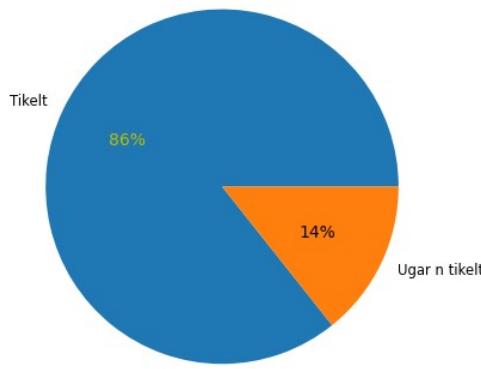
Amdan n wawalen: 18383, D amdan n yiferdisen igebren ula d asenqed.

```
awalen_n_tegrumma = []
for tafyirt in open("corpus-kab.txt",encoding='utf-8'):
    awalen=tafyirt.split()

    for i in awalen:
        j=i.split('/')
        awalen_n_tegrumma.append(j[0])
print ("Amdan n tegrumma: ",len(awalen_n_tegrumma))
```

Amdan n wawalen yemgaraden: 4717

14% n wawalen ney iferdisen isnalyanen ttuyalen-d ugar n tikelt.



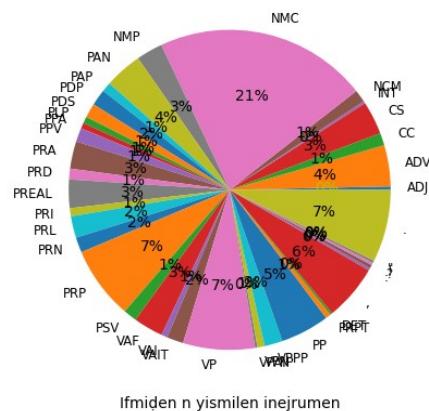
Tasleqt n udris n ulmad: afmiđi n wawalen i d-yettuyalen ugar n tikelt

```
awalen_yemgaraden = []
ticradj=[]
for tafyirt in open("corpus-kab.txt",encoding='utf-8'):
    awalen=tafyirt.split()
    for i in awalen:
        j=i.split('/')
        if (j[0] not in awalen_yemgaraden):
            awalen_yemgaraden.append(j[0])
```

```
print ("Amdan n wawalen yemgaraden: ",len(awalen_yemgaraden))
```

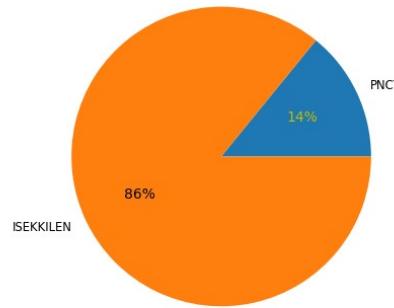
Afmiđi n yismilen inejrumen deg tegrumma n yisefka:

Tuget n wawalen deg tegrumma d ismawen akked yimyagen.



Afmiđi n usenqed/isekkilen:

Asenqed yewwed azal n 14% deg tegrumma n yisefka. 86% d isekkilen n ugemmay n teqbaylit.

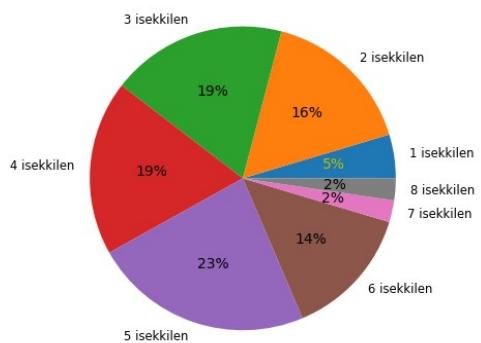


Ifmiden n yisekkilen d yizamulen n usenqed di teqbaylit

Afmiđi n wawalen s teysi:

Tget n wawalen n tefyar i nsedec deg tegrumma-a n yisefka, gebren gar 3 d 5 n yisekkilen.

Afmiđi n yimyagen s tmeżri:

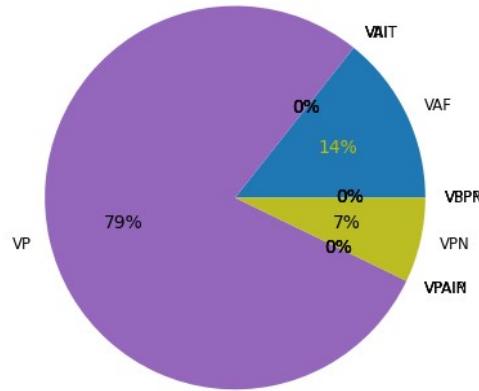


Ifniđen n teyzi n wawalen deg uđris n ulmad

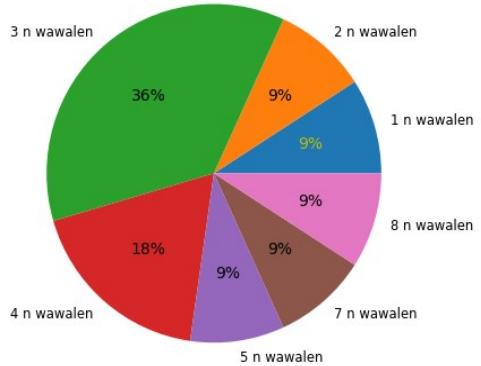
Afmiđi n tefyar s umđan n wawalen:

Tuget n tefyar deg uđris n ulmad i nseqdec deg tezrawt-a, gebrent gar 3 d 5 n yiferdisen isnalyanen illeliyen. Deg tesleđt tasnalyant taseddasant, tiseddi n tneyrufin tettali ticki tifyar wizzilit (gar 3 d 10 n yiferdisen).

Tasleđt n yiđrisen n teqbaylit temmal-d d akken tuget n yimyagen deg teqbaylit titlin deg tmeżri n wumir aħerfi ney izri.



Ifniđen n yimyagen s tmeżri deg uđris n ulmad



Ifniđen n teyzi n tifyar deg uđris n ulmad

Tagrumma n tecraq tisnalyanin tiseddasanin

Tagrumma n tecraq tisnalyanin tiseddasanin deg tuget d ismilen inejrumen. Ticraq deg tutayt taqbaylit zemrent ad ilin d tigi:

- Isem
- Amyag
- Arbib
- Amernu
- Tanzejt
- Amqim
- Tazelya

..atg.

Maca tikwal akken acraq asnalyan aseddasan ad yeddu s telqay, issefk ad yili kra n usiyef ney asilqey deg tecraq-a.

Amedya:

1. Amyag:

Izmer ad nebdu asmil n umyag yer tmeziyin-is meṛra. Ihi ad ay-d-ffyen 9 n tecraq i talyiwin n tmeziyin n umyag:

- Urmir anaq
- urmir
- Izri
- Amayun...

2. Isem:

- Isem amagnu
- Isem n umdan
- Isem uzzig & tyawsilin yettusemman...

3. Tazelya/Tisyunin:

- Tazelya/tisyunt n tuqqna
- Tazelya n tnilla -tuđfirt, tuzwirt)
- Tazelya n tibawt
- Tazelya n wurmir aherfi...

Deg tezrawt-a, nsemres 48 n tecraq tisnalyanin tiseddasanin. Xas wama ddeqs n telqay i mazal, maca tuget n yismilen inejrumen ttwabedren-d deg tebdert ddaw-a:

1. NMC: Isem amagnu (Nom commun)
2. NMP: Isem uzzig (Nom propre)
3. NC: Amdan s yisekkilen (Cardinal lettre)
4. NCM: Amdan s yizwilen (Cardinal Chiffre)
5. ABR: Asewzel/Axewfal (Acronyme/Abbréviation)
6. EML: Tansa imayl (Email)
7. DOM: Isem n tayult (Nom de domaine)
8. \$: Azamul n tedrimt (Valeur monétaire)

9. VAI: urmir anaq (aoriste impératif)
10. VAF: urmir imal (aoriste futur)
11. VP: izri (précédent)
12. VPN: izri ibaw (précédent négatif)
13. VPA: amayun n wurmir (participe de l'aoriste)
14. VPPP: amayun n yizri ilbaw (participe du précédent positif)
15. VPPN: amayun n yizri ibaw (participe du précédent négatif)
16. VII: anaq ussid (impératif intensif)
17. VAIT: urmir ussid (aoriste intensif)
18. VPAIP: amayun n wurmir ussid ilaw (participe de l'aoriste intensif positif)
19. VPAIN: amayun n wurmir ussid ibaw (participe de l'aoriste intensif négatif)
20. MADJ: irriben s umserbib (Morphème adjectivant)
21. ADJ: arbib (adjectif)
22. ADV: amernu (adverbe)
23. PRL: amassay (Pronom relatif)
24. PDP: Tazelya n tnila tuzwirt (particule de direction préfixe)
25. PDS: tazelya n tnila tuđfirt (particule de direction suffixe)
26. PRN: tazelya n tibawt (particule de négation)
27. PREAL: tazelya n tilawt (particule de réalité)
28. CC: Tasyunt n tuqqna (conjonction de coordination)
29. CS: Tasyunt n usentel (conjonction de subordination)
30. PAF: Tazelya n wurmir aherfi (particule de l'aoriste futur)
31. PAI: Tazelya n wurmir ussid (particule de l'aoriste intensif)
32. PLP: Tazelya n usiwel (Particule d'appel)
33. PPAF: Tazelya n umayun n wurmir aherfi (Particule du participe de l'aoriste)
34. PRP: tanzejt (préposition)
35. PRA: imqimen udwanen ilelliyan (pronoms personnels libres)
36. PPAN: imqimen n wayla iwšilen n yisem (Pronoms possessifs affixes de noms)
37. PDAN: Imqimen imeskaranen iwšilen n yisem d umernu (Pronoms démonstratifs affixes de noms ou d'adverbe)
38. PIAN: imqimen irbuda iwšilen n yisem (Pronoms indéfinis affixes de noms ou d'adverbe)
39. PPV: imqimen uwireen usriden n umyag (Pronoms préfixe de verbes)
40. PSV: imqimen udfiren usriden n umyag (Pronoms suffixes de verbes)
41. PAP: imqimen iwšilen n tenzejt (Pronoms affixes de prépositions)
42. PPA: imqimen n wayla (Pronoms possessifs autonomes)
43. PRI: imqimen irbuda (Pronoms indéfinis)
44. PRD: imqimen imeskaranen (Pronoms démonstratifs)
45. PAA: Imqimen iwšilen n yimerna (Pronom affixes d'adverbe)
46. INT: imqimen imattaren (pronoms interrogatifs)
47. ADINT: imerna imattaren (adverbes interrogatifs)
48. INJ: Tazelya n ubhat (Interjection)
49. PP: Tazelya n wayla.

Talyiwin n wawal & taseddast n tefyirt

Akken ad d-naf asmil anejrum n wawal deg tefyirt n tutlayt taqbaylit, tlaq tesleđt tasnalýant taseddasant. Gar talyiwin-a ad d-naf:

1. Awal s timmad-is.
2. Ma yezga-d wawal deg tazwara n tefyirt.
3. Ma yezga-d deg taggara n tefyirt.
4. Ma ibeddu wawal s usekkil meqqren.
5. Ma yura i meṛra s usekkil meqqren.
6. Ma yura i meṛra s usekkil mezziyen.
7. 1 usekkil seg tazwara n wawal.
8. 2 isekkilen seg tazwara n wawal.
9. 3 isekkilen seg tazwara n wawal.
10. 4 isekkilen seg tazwara n wawal.
11. 5 isekkilen seg tazwara n wawal.
12. 6 isekkilen seg tazwara n wawal.
13. 1 usekkil seg taggara n wawal.
14. 2 isekkilen seg taggara n wawal.
15. 3 isekkilen seg taggara n wawal.
16. 4 isekkilen seg taggara n wawal.
17. 5 isekkilen seg taggara n wawal.
18. Awal uzwir amenzu (i d-yezgan uqbel).
19. Awal uzwir wis sin (i d-yezgan uqbel n wawal uzwir menzu).
20. Awal uđfir amezu (i d-yezgan mbeđd).
21. Awal uđfir wis sin (i d-yezgan mbeđd n wawal ufir amenzu).
22. Ma yegber kan izwilen.
23. Ma yegber asekkil meqqren daxel-is
24. Ma yegber asekkil ur nelli ara deg ugemmay ney deg usenqed(@,\$,%,& ...atg).

Amedya:

"Tiziri tettwabder-d deg tesdawsit ass-a."

Deg tefyirt nnig-a, tasleđt n wawal "tettwabder", ad tefk:

- Awal s timmad-is: **tettwabder**
- Ma yezga-d wawal deg tazwara n tefyirt: **False**
- Ma yezga-d deg taggara n tefyirt: **False**
- Ma ibeddu wawal s usekkil meqqren: **False**
- Ma yura i meṛra s usekkil meqqren: **False**
- Ma yura i meṛra s usekkil mezziyen: **True**
- 1 usekkil seg tazwara n wawal: **t**
- 2 isekkilen seg tazwara n wawal: **te**
- 3 isekkilen seg tazwara n wawal: **tet**
- 4 isekkilen seg tazwara n wawal: **tett**
- 5 isekkilen seg tazwara n wawal: **tettw**
- 6 isekkilen seg tazwara n wawal: **tettwa**
- 1 usekkil seg taggara n wawal: **r**
- 2 isekkilen seg taggara n wawal: **er**
- 3 isekkilen seg taggara n wawal: **der**
- 4 isekkilen seg taggara n wawal: **bder**
- 5 isekkilen seg taggara n wawal: **abder**
- Awal uzwir amenzu (i d-yezgan uqbel): **Tiziri**
- Awal uzwir wis sin (i d-yezgan uqbel n wawal uzwir menzu): **False**
- Awal uđfir amezu (i d-yezgan mbeđd): **-d**
- Awal uđfir wis sin (i d-yezgan mbeđd n wawal ufir amenzu): **deg**
- Ma yegber kan izwilen: **False**
- Ma yegber asekkil meqqren daxel-is: **False**
- Ma yegber asekkil ur nelli ara deg ugemmay ney deg usenqed(@,\$,%,& ...atg): **False**

Acrad asnalyan aseddasan s Keras

Betṭu asnalyan aseddasan

Tabadut

Betṭu asnalyan aseddasan, d asemsel awurman uzwir n tutlayt ara yeğgen aselkim ad yebdu tifyar yer wawalen ilelliyen akken ad yizmir ad d-yaf asmil anejrum n yal awal ney ayen iwumi neqqar: acrad asnalyan aseddasan.

Deg uswir-a n tesleqt, betṭu asnalyan aseddasan ilehhu-d kan deg yiwerdisen ilelliyen. Aswir-a ur d-ilehhu ara deg yimataren

Alguritm n betṭu asnalyan aseddasan:

Alguritm-a yettwaru s Python 3.6. Deg tazwara ad as-nefk tifyar yettwrun s teqbaylit s uqader n yilugan n tira. Ticki yettwaselkem, ad d-yerr afa tifyar maca s wawalen yebdan. Alguritm-a yessemras tagrumma n yiwsilen n yimyagen, ismawen, tinzay d yimerma. Iwsilen-a ttwarun s tizdit ama d iwṣilen uzwiren ney iwṣilen uđfiren.

```
uđfiren=uzwiren=[]
Asenqed=['...',' ',';','?','!',':',"'",'(',')',
'*','_','.',',[','']','{','}',">'','>','+'+
','=','"']
for i in open("imqimen_uzwiren_udđfiren.txt",
"r",encoding='utf-8'):
    a=i.replace("\ufffd","",).replace("\n","",).strip()
    if (a[len(a)-1]=="-"):
        uzwiren.append(str(a).lower())
    else:
        udđfiren.append(str(a).lower())
def bđu_awal(awal,udđfiren,uzwiren):
    amurfim=awal[0:awal.find('-')+1]
    awal_yebđan=''
    if (amurfim in uzwiren):
        awal=awal[awal.find(
            '-')+1:len(awal)]
        awal_yebđan=awal_yebđan+' '+amurfim
        while awal.find('-')>=0:
            amurfim=awal[0:awal.find(
                '-')+1]
            awal=awal[awal.find(
                '-')+1:len(awal)]
            awal_yebđan=awal_yebđan+
            '+amurfim
            awal_yebđan=awal_yebđan+' '+awal
    else:
        amurfim=awal[0:awal.find('-')]
        awal_yebđan=awal_yebđan+' '+amurfim
        awal=awal[awal.find(
            '-')+1:len(awal)]
        while awal.find('-')>=0:
            amurfim=awal[0:awal.find(
                '-')]
```

udmawanen n yimyagen, ney anaw, amđan n waddad n yismawen. Deg teqbaylit, awalen bđan deg tefyar s usemres n usekkil ilem ney tizdit.

Amedya:



```
awal_yebđan=awal_yebđan+' '+-
'+amurfim
awal=awal[awal.find(
    '-')+1:len(awal)]
if ('-'+awal in udđfiren):
    awal_yebđan=awal_yebđan+' '+-
    '+awal
else:
    awal_yebđan=awal_yebđan
return awal_yebđan
# tawuri-a tbetṭu tafyirt
def bđu_tafyirt(tafyirt,udđfiren,uzwiren):
    a= tafyirt.split()
    tafyirt1=""
    for i in a: #awalen
        if(i.find('<')<0):
            tafyirt1=tafyirt1+' '+i
        else:
            awals=bđu_awal(i,udđfiren,uzwiren)
            tafyirt1=tafyirt1+' '+awals
            tafyirt1=tafyirt1.strip()
    return tafyirt1
f= open("tokenized_text.txt","w+",
encoding='utf-8')
g=open("brut_text.txt",encoding='utf-8')
for adur in g:
    for i in Asenqed:
        adur=adur.replace(i,' '+i+
        '').replace("\ufffd","",)
izirig=bđu_tafyirt(adur,udđfiren,uzwiren)
izirig=izirig.replace(" "," ")
f.write(izirig+'\n')
f.close()
g.close()
```

Ticraq tisnalyanin tiseddasanin yettwacqedcen

Ticraq ad d-ttwakksent seg tegramma n tefyar
 nnig-a:

```
ticraq = set([
    tacredt for tafyirt in tifyar
    for _, tacredt in tafyirt
])

{'VPPP', 'PADV', 'PRL', '!', 'INT', '.', 
'PRP', 'PPV', 'VAIT', 'PAINT', '-', ';', 
'PRD', 'PRPT', 'NMP', 'CC', 'PAN', '(', 
'PREAL', 'VPAIN', 'VPN', ')', 'INTR', 'VP', 
'VAI', 'PDP', 'PAD', '"', ':', 'CS', 'PRI', 
'VS', 'NCM', '?', 'VAF', 'PRN', 'PPA', 
'VPPN', ',', 'PAP', 'PSV', 'VPAIP', 'NMC.', 
'VII', '$', 'PDS', 'PP', 'DET', 'ADV', 
'PAPR', 'VPA', 'PRA', 'NMC', 'PLP', 'INJ', 
'ADJ'}
```

Betju n tegramma n yisefka i ulmad s tallelt

Deg usleymu n yisefka isnalyanen iseddassanen s usemres n uzeṭṭa n yinurunen (Aselmed s tallelt), ilaq tagrumma n yisefka ad tebdu yef 3 n tegrummiwin timeżyanin. Maca s leħsab.

- Tagrumma n yisefka n usleymu:** D tukkist n yisefka ara yettwasqedcen deg usnulfu n tneyruft,
- Tagrumma n yisefka n usentem:** D tukkist n yisefka ara yettwasqedcen deg ušeggem n yiġewwaren n umsismel, amedja: afra n umdān n tayunin n uzeṭṭa n yinurunen,
- Tagrumma n yisefka n usekyed:** D tukkist n yisefka ara yettwasqedcen deg i uktazel n umsismel.



S leħsab: 60% n yisefka i usleymu, 20% n yisefka i ušeggem n yiġewwaren n usleymu, 20% n yisefka i uktazel n tseddi n tneyruft.

```
train_test_cutoff = int(.80 * len(sentences))

training_sentences =
sentences[:train_test_cutoff]
testing_sentences =
sentences[train_test_cutoff:]

train_val_cutoff = int(.25 * len(training_sentences))
validation_sentences =
training_sentences[:train_val_cutoff]
training_sentences =
training_sentences[train_val_cutoff:]
```

Asbadu n taliwin n wawal

Akken i d-nessegza deg usebter wis 21, yal awal deg tegramma n yisefka isnilsanen iseddassanen ad yettwasled.

Tawwuri ddaw-a tattarra-d 21 n yisallen yef yal awal.

Izmer ad nesnerni tagrumma-a s usemres n taliwin-nniđen. Maca skud yettnerni umdān n taliwin, skud aselkim yettawi ugar n wakud akken ad yesled meṛṛa awalen yellan deg uđris n ulmad.

```
def add_basic_features(awalen_n_tefyirt,
amatar):

    awal = awalen_n_tefyirt[amatar]
    return {
        'nb_terms': len(awalen_n_tefyirt),
        'term': awal,
        'is_first': amatar == 0,
        'is_last': amatar ==
len(awalen_n_tefyirt) - 1,
        'is_capitalized': awal[0].upper() ==
term[0],
        'is_all_caps': awal.upper() ==
term,
        'is_all_lower': awal.lower() ==
term,
        'is_numeric':
awalen_n_tefyirt[amatar].isdigit(),
        'prev_word': '' if amata == 0 else
awalen_n_tefyirt[amatar - 1],
        'prev2_word': '' if amatar == 1
else awalen_n_tefyirt[amatar - 2],
```

```

    'next_word': '' if amatar ==
len(awalen_n_tefyirt) - 1 else
awalen_n_tefyirt[amatar + 1],
    #'next2_word': '' if amatar ==
len(awalen_n_tefyirt) - 2 else
awalen_n_tefyirt[amatar + 1],
    'prefix-1':
awalen_n_tefyirt[amatar][0],
    'prefix-2':
awalen_n_tefyirt[amatar][:2],
    'prefix-3':
awalen_n_tefyirt[amatar][:3],
    'prefix-4':
awalen_n_tefyirt[amatar][:4],
    'prefix-5':
awalen_n_tefyirt[amatar][:5],
    'suffix-1':
awalen_n_tefyirt[amatar][-1],
    'suffix-2':
awalen_n_tefyirt[amatar][-2:],
    'suffix-3':
awalen_n_tefyirt[amatar][-3:],
    'suffix-4':
awalen_n_tefyirt[amatar][-4:],

}

```

Tukksa n tecrad seg tefyirt

Tawuri ddaw-a tettikkes meṛra ticrad tisnalyanin tiseddasanin (asmil anejrum) seg tafyirt.

```

def untag(tefyirt_yettwacerden):
    return [w for w, _ in
tefyirt_yettwacerden]

```

Betṭu n tefyar yettwacerden yer snat n tegrummiwin

Tawuri ddaw-a tbetṭu tifyar yettwacerden yef snat n tegrummiwin. Yal tafyirt iqubel-itt-id uzrir n tecrad-is:

- Tagrumma n tefyar war ticrad
- Tagrumma n tecrad

```

def transform_to_dataset(tagged_sentences):
    X, y = [], []
    for pos_tags in tagged_sentences:
        for amatar, (term, class_) in
enumerate(pos_tags):
            # Add basic NLP features for
            # each sentence term

```

```

X.append(add_basic_features(untag(pos_tags),
, amatar))
y.append(class_)
return X, y
###  

X_train, y_train =
transform_to_dataset(training_sentences)

X_test, y_test =
transform_to_dataset(testing_sentences)
X_val, y_val =
transform_to_dataset(validation_sentences)

```

Tiririt n yisefka yer talya n yimdanen

```

label_encoder = LabelEncoder()
label_encoder.fit(y_train + y_test + y_val)
dump(label_encoder,
open('label_encoder.pkl', 'wb'))
X_train =
dict_vectorizer.transform(X_train)
print(len(X_train[1]))

X_test = dict_vectorizer.transform(X_test)
X_val = dict_vectorizer.transform(X_val)

label_encoder = LabelEncoder()
label_encoder.fit(y_train + y_test + y_val)

y_train = label_encoder.transform(y_train)
y_test = label_encoder.transform(y_test)
y_val = label_encoder.transform(y_val)

y_train = np_utils.to_categorical(y_train)
y_test = np_utils.to_categorical(y_test)
y_val = np_utils.to_categorical(y_val)

```

Asbadu n uzeṭṭa n yinurunen

Tayessa n uzeṭṭa n yinurunen yettwasqedcen tegber 4 n tissiwin

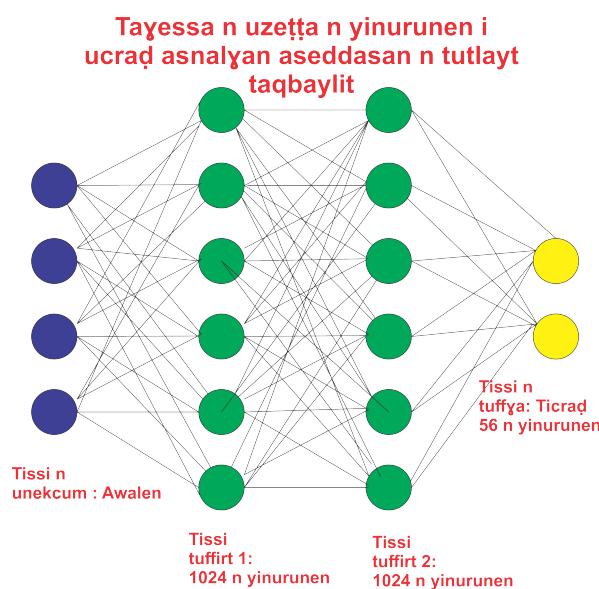
- 1- Tissi n unekcum
- 2- Tissi tuffirt tamezwarut s 1024 n yinurunen
- 3- Tissi tuffirt tis snat s 1024 n yinurunen
- 4- Tissi n tuffyā s 56 n yinurunen

Dagi nsemres 2 n tissiwin yeffren. Izmer ad nsemres ugār, maca skud yettnerni umđan n tissiwin yeffren, skud aselkim ilemmmed ugār, maca ad yawi ddeqs n wakud deg usleymu n yisefka.

Ad d-nger tamawt d akken dagi nsemres tawuri **relu** i urmad n tissiwin n unekcum d

tid yeffren. Tawuri softmax tettwaseqdec i urmad n tissi n tuffya.

Ugar n yisallen yef yizedwa n yinurunen, ddu yer wumuy n temsisyal akked temsirin srid i nefka yer taggara.



```
def build_model(input_dim, hidden_neurons,
output_dim):
    model = Sequential([
        Dense(hidden_neurons,
input_dim=input_dim),
        Activation('relu'),
        Dropout(0.2),
        Dense(hidden_neurons),
        Activation('relu'),
        Dense(hidden_neurons),
        Activation('relu'),
        #Dropout(0.2),
        Dense(output_dim,
activation='softmax')
    ])

    model.compile(loss='categorical_crossentropy',
y', optimizer='adam', metrics=['accuracy'])
    return model

model_params = {
    'build_fn': build_model,
    'input_dim': X_train.shape[1],
    'hidden_neurons': 1024,
    'output_dim': y_train.shape[1],
    'epochs': 10,
    'batch_size': 1024,
    'verbose': 1,
    'validation_data': (X_val, y_val),
    'shuffle': True
}
```

Skud yettnerni umdan n yinurunen deg tissiwin yeffren akked umdan n #epoch (achal n tikkal ara ieawed uselkim i ulmad), aselkim ilemmmed ugar. Maca ad yawi atas n wakud akken ad d-yessufey taneyruft n ulmad. Xef waya, yessefk aselkim ara yesleymun isefka n ulmad, ad yili n tsutwin-a tineggura, anagar ma yegber takarda GPU ney TPU.

Asleymu n üzətə n yinurunen

Akken aselkim ad yegzu talyiwin n wawalen d tseddast n tefyar, ilaq azetə ad ieiddi merrə yef tefyar n tegrumma n ulmad.

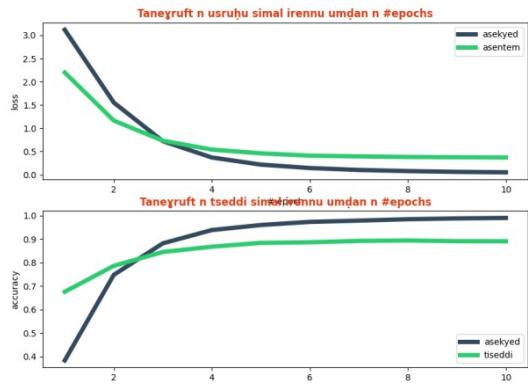
```
model=build_model(X_train.shape[1],1024,
y_train.shape[1])
dump(model, open('model.clf', 'wb'))

clf = KerasClassifier(**model_params)
hist = clf.fit(X_train, y_train)
```

Aktazel n tneyruft

Tiseddi n ucraq tettnerni ticki tıqten yisefka ney ticki ittuqet umdan n tissiwin n üzətə n yinurunen. Maca, tikwal ma yella tagrumma n yisefka ur tegbir ara ddeqs n talyiwin n tefyar, aya izmer ur d-yettak ara igmaq yelhan. Almad lqayen ney tigzi n tmacint bnan yef tegrummiwin n yisefka meqqren. Ma nserwes gar teyzi n udris n ulmad i nsemres deg tezrawt-a akked wid yettwasemrasen i titlayin-nniñen teglizit, ad naf d akken adrıs n ulmad n teglizit yugar win n teqbaylit s wazal n 1000 n tikal.

Tasnilest n yisefka isnilsanen mazal ur tuy ara azar deg yigedza ney tisadawin i isselmaden tutlayin timaziyinyef waya ur d-nufi ara tigrummiwin n yisefka ara nsemres deg yinadiyen-a i d-nger, gar-asen tazrawt-a.



```
def plot_model_performance(train_loss,
train_acc, train_val_loss, train_val_acc):
    blue= '#34495E'
    green = '#2ECC71'
    orange = '#E23B13'

    fig, (ax1, ax2) = plt.subplots(2,
figsize=(10, 8))
    ax1.plot(range(1, len(train_loss) + 1),
train_loss, blue, linewidth=5,
label='asekyed')
    ax1.plot(range(1, len(train_val_loss) +
1), train_val_loss, green, linewidth=5,
label='asentem')
    ax1.set_xlabel('# epoch')
    ax1.set_ylabel('loss')
```

```
    ax1.tick_params('y')
    ax1.legend(loc='upper right',
shadow=False)
    ax1.set_title('Asruḥu n tseddi s
#epochs', color=orange, fontweight='bold')

    ax2.plot(range(1, len(train_acc) + 1),
train_acc, blue, linewidth=5,
label='asekyed')
    ax2.plot(range(1, len(train_val_acc) +
1), train_val_acc, green, linewidth=5,
label='tiseddi')
    ax2.set_xlabel('# epoch')
    ax2.set_ylabel('tiseddi')
    ax2.tick_params('y')
    ax2.legend(loc='lower right',
shadow=False)
    ax2.set_title('Tiseddi s #epochs',
color=orange, fontweight='bold')
    plt.show()

plot_model_performance(
    train_loss=hist.history.get('loss',
[]),
    train_acc=hist.history.get('acc', []),
    train_val_loss=hist.history.get('val_loss',
[]),
    train_val_acc=hist.history.get('val_acc',
[]))
```

Asemres n tneyruft tasnalyant taseddasant n keras

Deg ulgruritm ddaw-a, ad nsemres taneyruft tasnalyant taseddasant i d-nesnulfa nnig-a akken ad necreñ tifyar timaynutin.

Amedya:

Tafyirt	Açrad n wawalen
Ad as-tent-id-awiy.	Ad/PAF as-/PPV tent-/PPV id-/PDP awiy/VAF ./.
Ay abrid ttun medden.	Ay/PLP abrid/NMC ttun/VP medden/NMC ./.
Yemyi-d lehcic deg later-ik.	Yemyi/VP -d/PDS lehcic/NMC di/PRP later/NMC -ik/PAN ./.
Aeudiw n baba d jeddi, rniy rrekba n deffir!	Aeudiw/VAI n/PP baba/NMC d/PREAL jeddi/NMC ,/, rniy/VP rrekba/NMC n/PP deffir/NMC !/!
Wwiy-d 4582 n tfunasin.	Wwiy/VP -d/PDS 4582/NC ,/, n/PP tfunasin/NMC ./.
Axxam, ibennu yef llsas.	Axxam/NMC ,/, ibennu/VAIT yef/PRP llsas/NMC ./.
Llsas-nney nekni d ayen nnan d wayen gan yimezwura-nney.	Llsas/NMC -nney/PAN nekni/PPA d/CC ayen/PRD nnan/VP d/PRP wayen/PRI gan/VP yimezwura/NMC -nney/PAN ./.
D win i ay-ilaq ad d-nejmee ass-a.	D/PREAL win/PRD i/PRL ay-/PPV ilaç/VP ad/PAF d-/PDP nejmee/VPPP ass/NMC -a/PAN ./.
Awerrat iherzen ayen i as-d-yekkan syur Iwaldin-is, yelha; win yernan s ayen i as-d-yeğga baba-s, yif-it	Awerrat/NMC iherzen/VPPP ayen/PRD i/PRL as-/PPV d-/PDP yekkan/VPPP syur/PRP Iwaldin/NMC -is/PAN ,/, yelha/VP ;/; win/PRD yernan/VPPP s/PRP ayen/PRD i/PRL as-/PPV d-/PDP yeğga/VP baba/NMC -s/PAN ,/, yif/VP -it/PSV ./.

Tamawt: Tifyar nnig-a ttwacerdent s uselkim. Izmer ad ilint tuccdiwin. Asegzi n tecrađ nefka-t-id deg taggara n tezrawt-a.

```
from sklearn.feature_extraction import
DictVectorizer
from sklearn.preprocessing import
LabelEncoder
from keras.models import load_model
import numpy as np
from pickle import load
def add_basic_features(sentence_terms,
index):
    term = sentence_terms[index]
    return {
        'nb_terms': len(sentence_terms),
        'term': term,
        'is_first': index == 0,
        'is_last': index ==
len(sentence_terms) - 1,
        'is_capitalized': term[0].upper() ==
term[0],
        'is_all_caps': term.upper() ==
term,
        'is_all_lower': term.lower() ==
term,
        'is_numeric':
sentence_terms[index].isdigit(),
        'prev_word': '' if index == 0 else
sentence_terms[index - 1],
```

```
'prev2_word': '' if index == 1 else
sentence_terms[index - 2],
        'next_word': '' if index ==
len(sentence_terms) - 1 else
sentence_terms[index + 1],
        'prefix-1':
sentence_terms[index][0],
        'prefix-2':
sentence_terms[index][:2],
        'prefix-3':
sentence_terms[index][:3],
        'prefix-4':
sentence_terms[index][:4],
        'prefix-5':
sentence_terms[index][:5],
        'suffix-1': sentence_terms[index][-1],
        'suffix-2': sentence_terms[index][-2:],
        'suffix-3': sentence_terms[index][-3:],
        'suffix-4': sentence_terms[index][-4:],
    }
def transform_to_dataset(sentences):
    X = []
```

```
for sentence in sentences:  
    for index, word in  
enumerate(sentence) :  
X.append(add_basic_features(sentence,  
index))  
    return X  
sentences=[['('Awi'), ('-  
d'), ('aman'), ('.')], [('Awi'), ('-as'), ('-  
t'), ('-  
id'), ('.'), ('Zemren'), ('i'), ('twuya'), ('qer  
rihen'), ('.')]]  
X_train = transform_to_dataset(sentences)  
# Fit our DictVectorizer with our set of  
features  
dict_vectorizer =  
DictVectorizer(sparse=False)  
dict_vectorizer.fit(X_train)  
# Convert dict features to vectors  
X_train =  
dict_vectorizer.transform(X_train)  
# load model the kerras model  
loaded_model = load(open('model.clf',  
'rb'))  
#load the input dim  
for l in loaded_model.layers:  
    inputs= l.input_shape[1]  
  
break  
#add remaining dim for the sentences to be  
taged  
a=np.zeros((X_train.shape[0], inputs-  
X_train.shape[1]))  
X_train=np.concatenate((X_train, a),axis=1)  
# make prediction  
predictions =  
loaded_model.predict_classes(X_train,verbos  
e=1)  
  
#lod the tag labels  
label_encoder = LabelEncoder()  
label_encoder =  
load(open('label_encoder.pkl', 'rb'))  
  
#print words and theur tags  
  
words=[]  
for i in sentences:  
    for j in i:  
        words.append(j)  
print(words)  
  
print(label_encoder.inverse_transform(predi  
ction)
```

Acrad asnayal aseddasan s usemres n CRF (Scikit-learn)

CRF d yiwt n tarrayt yettwasemrasen deg tayult n tesledo n yisefka yesean ticrad yettemsedfare am deg tseddast, tajerrumt. Tarrayt-a tettwasemras dayen deg tyula-nnidien mačči kan deg tesnilest.

Ilguritmen n ddaw-a semrasen yiwt n tarrayt i ulmad, maca mgaraden yilguritmen. Yal algoritm yesea iyewwaren-is. Ilguritmen-a ttwasbedden yakan deg scikit-learn. Twaseqdacen deg yal talya n yisefka anida ttemsedfare wanawen ney ticrad n yisefka. Ilguritmen-a semrasen meṛṛa yiwt n talya n yisefka n ulmad.

Tzemred ad tafeq meṛṛa ilguritmen-a d tegrumma n yisefka n ulmad akked yisefka-nnidien deg tegrumma n yifuyla i d-yeddan deg tezrawt-a.

Talya n yisefka n tegrumma n tecrad akked ulmad

Tagrumma n tecrad akked yisefka isnalyanen iseddasanen i nsemres yakan deg ulgirtm n Keras nniga, d tin ara nseqdec dayen deg yilguritmen-a. Ilguritmen n beṭtu asnalyan asaedasan d wid dayen ara yeqqimen akken llan.

Algoritm n ddaw-a d win ara yettwasqedcen deg yal algoritm n usleymu n yisefka ama s tarrayt **Ibtgs, Isgd, ap, pa ney arow**.

```
from sklearn_crfsuite import CRF
from nltk.tag.util import untag
from sklearn_crfsuite import metrics

tagged_sentences=[]
#Construction du texte global à apartir du
corpus étiqueté
first=0
taille2=0
for ligne in open("corpus-
kab.txt",encoding='utf-8'):
    taille=0
    if (first!=0):
        sentence=[]
        line=ligne.split()
        taille=len(line)
        for i in line:
            j=i.split('/')
            couple=(j[0],j[1])
            sentence.append(couple)
        taille2=taille+taille
        tagged_sentences.append(sentence)
    first=1

print("Amdan n tefyar: ",
len(tagged_sentences))
print("Amdan n yiferdisen: ", taille2)

#Définition des caractéristiques d'un mot
def features(sentence, index):
    return {
        'word': sentence[index],      # Awal
s timmad-is
        'is_first': index == 0,       # Ma
yezga-d deg tazwar n tefyirt
        'is_last': index == len(sentence) - 1, # Ma yezgma-d deg taggar n tefyirt
        'is_capitalized': sentence[index][0].upper() ==
sentence[index][0], # MA ibeddu s usekkil
meqqren
        'is_all_caps': sentence[index].upper() == sentence[index],
# Ma yura meṛṛa s usekkil meqqren
        'is_all_lower': sentence[index].lower() == sentence[index],
# ma yura meṛṛa s usekkil mezzien
        'prefix-1': sentence[index][0], #1
usekkil uzwir
        'prefix-2': sentence[index][:2], #2
isekkilen uzwiren
        'prefix-3': sentence[index][:3], #3
isekkilen uzwiren
        'prefix-4': sentence[index][:4], #4
isekkilen uzwiren
        'prefix-5': sentence[index][:5], #4
isekkilen uzwiren tettecmumuḥend (aoriste
intensif)
        'suffix-1': sentence[index][-1], #1
usekkil udfir
        'suffix-2': sentence[index][-2:], #2
isekkilen udfire
        'suffix-3': sentence[index][-3:], #3
isekkilen udfire
        'suffix-4': sentence[index][-4:], #2
isekkilen udfire
        'prev_word': '' if index == 0 else
sentence[index - 1], #awal uzwir
```

```

    'next_word': '' if index ==
len(sentence) - 1 else sentence[index + 1],
#awal udfir

    'is_numeric':
sentence[index].isdigit(), #ma yegber kan
izwilen
    'capitals_inside':
sentence[index][1:].lower() !=
sentence[index][1:] #ma yegber asekkil
meqqren daxel-is
}

#transformation du corpus x: contient les
tokens et y les tags
def transform_to_dataset(tagged_sentences):
    X, y = [], []
    for tagged in tagged_sentences:
        X.append([features(untag(tagged)),
index] for index in range(len(tagged)))
        y.append([tag for _, tag in
tagged])
    return X, y

total=int(len(tagged_sentences)*0.80)
#print(total)
#train=tagged_sentences

X1_train, y1_train =
transform_to_dataset(tagged_sentences[:total])
X_test, y_test =
transform_to_dataset(tagged_sentences[total:])

```

Merṛa ilguritmen n ucrađ semrasen yiwt n tarray n beṭtu asnalyan aseddasan. D isem n ufatlu i igebrēn taneyruft n ulmad i yettbeddilen.

Asleymu n yisefka

Xemsa n yilguritmen n CRF sklearn_crfsuite, yal wa d acu n tarrayt i yessexdam, ssawađen gezzun taseddast n tefyar d talya n wawalen.

Alguritm lbfgs

Alguritm-a yessemras lbfgs ney s teglizit (Limited-Memory Broyden-Fletcher-Goldfarb-Shanno) i usleymu n yisefka.

```

#déclaration du modèle suivant l'algorithme
lbfgs
model = CRF(
    algorithm='lbfgs',
    min_freq=2, #min des occurrences des
features au dessous du min on ignore
    all_possible_states=True,
    all_possible_transitions=True,

    c1=0.1,
    c2=0.01,
    max_iterations=100,# lbfgs - unlimited;
l2sgd - 1000; ap - 100; pa - 100; arow -
100.
    num_memories=6,
    epsilon=1e-5,# ap, arow, lbfgs, pa
    period=10, # The duration of iterations
to test the stopping criterion. l2sgd,
lbfgs
    delta=1e-5, # l2sgd, lbfgs
    linesearch= 'MoreThuente',#L-BFGS
values MoreThuente Backtracking
StrongBacktracking
    max_linesearch=20,# The maximum number
of trials for the line search algorithm.
lbfgs
    #calibration_eta=0.1, # l2sgd
)
#entraînement
model.fit(X1_train, y1_train)

labels = list(model.classes_)
Ponctuation= [ ';', '(', ')', '-',
'.', ',', '!', '?', '"', ':', '$', ',']
for i in Ponctuation:
    labels.remove(i)
print(labels)
print("amdan n tecrađ : ",len(labels))

y_pred = model.predict(X_test)

#Affichage des métriques
metrics.flat_f1_score(y_test, y_pred,
average='weighted',
labels=labels)

sorted_labels = sorted(
    labels,
    key=lambda name: (name[1:], name[0]))
)

print(metrics.flat_classification_report(
    y_test, y_pred, labels=sorted_labels,
    digits=3
))

```

```
#affichage des transitions

from collections import Counter

def print_transitions(trans_features):
    for (label_from, label_to), weight in trans_features:
        print("%-6s -> %-7s %.6f" %
(label_from, label_to, weight))

print("Transitions les plus courantes:")

print_transitions(Counter(model.transition_features_).most_common(20))

print("\nTransitions les moins courantes:")
print_transitions(Counter(model.transition_features_).most_common()[-20:])

# Sauvegarde du modèle

from joblib import dump, load
dump(model, 'model_lbfsgs.joblib')
```

Algiritm l2SGD

Algiritm-a yessemras L2SGD ney s (Stochastic Gradient Descent with L2 regularization term) i usleymu n yisefka.

```
model = CRF(
    algorithm='l2sgd',
    min_freq=2, #min des occurrences des features au dessous du min on ignore
    all_possible_states=True,
    all_possible_transitions=True,
    #c1=0.1,
    c2=0.01,
    max_iterations=100,# lbfsgs - unlimited;
    l2sgd - 1000; ap - 100; pa - 100; arow -
    100.
    #num_memories=6,
    #epsilon=1e-5,# ap, arow, lbfsgs, pa
    period=10, # The duration of iterations to test the stopping criterion. l2sgd,
    lbfsgs
    delta=1e-5, # l2sgd, lbfsgs
    #linesearch= 'MoreThuente',#L-BFGS values MoreThuente Backtracking
    StrongBacktracking
    #max_linesearch=20,# The maximum number of trials for the line search algorithm.
    lbfsgs
    #calibration_eta=0.1, # l2sgd
)
#entraînement
model.fit(X1_train, y1_train)

labels = list(model.classes_)
#labels.remove('0')
Ponctuation= [ ';', '(', ')', '-',
',', '.', '!', '?', '!', ':', '$', ',', '']
for i in Ponctuation:
    labels.remove(i)
print(labels)
print("amdan n tecrad : ",len(labels))

y_pred = model.predict(X_test)
metrics.flat_f1_score(y_test, y_pred,
                      average='weighted',
                      labels=labels)

sorted_labels = sorted(
    labels,
    key=lambda name: (name[1:], name[0])
)

print(metrics.flat_classification_report(
    y_test, y_pred, labels=sorted_labels,
    digits=3
))

from collections import Counter

def print_transitions(trans_features):
    for (label_from, label_to), weight in trans_features:
        print("%-6s -> %-7s %.6f" %
(label_from, label_to, weight))

print("Transitions les plus courantes:")
print_transitions(Counter(model.transition_features_).most_common(20))

print("\nTransitions les moins courantes:")
print_transitions(Counter(model.transition_features_).most_common()[-20:])

# Sauvegarde du modèle

from joblib import dump
dump(model, 'model-l2sgd.joblib')
```

Alguritm AP

Alguritm-a yessemras AP ney s (Averaged Perceptron) i usleymu n yisefka.

```

model = CRF(
    algorithm='ap',

    min_freq=2, #min des occurences des
features au dessous du min on ignore
    all_possible_states=True,
    all_possible_transitions=True,

    #c1=0.1,
    #c2=0.01,
    max_iterations=100,# lbfsgs - unlimited;
l2sgd - 1000; ap - 100; pa - 100; arow -
100.
    #num_memories=6,
    epsilon=1e-5,# ap, arow, lbfsgs, pa
    #period=10, # The duration of
iterations to test the stopping criterion.
l2sgd, lbfsgs
    #delta=1e-5, # l2sgd, lbfsgs
    #linesearch= 'MoreThuente',#L-BFGS
values MoreThuente Backtracking
StrongBacktracking
    #max_linesearch=20,# The maximum number
of trials for the line search algorithm.
lbfgs
    #calibration_eta=0.1, # l2sgd
)
#entraînement
model.fit(X1_train, y1_train)

labels = list(model.classes_)
#labels.remove('0')
Ponctuations= [ ';', '(', ')', '-',
',', '.', '!', '?', '"', ':', '$', ',', '']
for i in Ponctuation:
    labels.remove(i)
print(labels)
print("amdan n tecrad : ",len(labels))

y_pred = model.predict(X_test)
metrics.flat_f1_score(y_test, y_pred,
average='weighted',
labels=labels)

sorted_labels = sorted(
    labels,
    key=lambda name: (name[1:], name[0]))
)

print(metrics.flat_classification_report(
    y_test, y_pred, labels=sorted_labels,
digits=3
))

#print (model.score(X1_train, y1_train))
# Sauvegarde du modèle

from joblib import dump
dump(model, 'model-ap.joblib')

from collections import Counter

def print_transitions(trans_features):
    for (label_from, label_to), weight in
trans_features:
        print("%-6s -> %-7s %.6f" %
(label_from, label_to, weight))

print("Top likely transitions:")
print_transitions(Counter(model.transition_
features_).most_common(20))

print("\nTop unlikely transitions:")
print_transitions(Counter(model.transition_
features_).most_common()[-20:])

```

Algûrîtm PA

Algûrîtm-a yessemras PA ney s (Passive Aggressive) i usleymu n yisefka.

```

model = CRF(
    algorithm='pa',

    min_freq=2, #min des occurences des
features au dessous du min on ignore
    all_possible_states=True,
    all_possible_transitions=True,

    #c1=0.1,
    #c2=0.01,
    max_iterations=100,# lbfsgs - unlimited;
l2sgd - 1000; ap - 100; pa - 100; arow -
100.
    #num_memories=6,
    epsilon=1e-5,# ap, arow, lbfsgs, pa
    #period=10, # The duration of
iterations to test the stopping criterion.
l2sgd, lbfsgs
    #delta=1e-5, # l2sgd, lbfsgs
    #linesearch= 'MoreThuente',#L-BFGS
values MoreThuente Backtracking
StrongBacktracking
    #max_linesearch=20,# The maximum number
of trials for the line search algorithm.
lbfsgs
    #calibration_eta=0.1, # l2sgd
)
#entraînement
model.fit(X1_train, y1_train)

labels = list(model.classes_)
#labels.remove('0')
Ponctuation= [ ';', '(', ')', '-',
'.', '!', '?', '"', ':', '$', ',']
for i in Ponctuation:
    labels.remove(i)
print(labels)
print("amdan n tecraq : ",len(labels))

```

```

y_pred = model.predict(X_test)
metrics.flat_f1_score(y_test, y_pred,
average='weighted',
labels=labels)

sorted_labels = sorted(
    labels,
    key=lambda name: (name[1:], name[0]))
)

print(metrics.flat_classification_report(
    y_test, y_pred, labels=sorted_labels,
    digits=3
))

#print (model.score(X1_train, y1_train))
# Sauvegarde du modèle

from joblib import dump
dump(model, 'model-pa.joblib')

from collections import Counter

def print_transitions(trans_features):
    for (label_from, label_to), weight in
trans_features:
        print("%-6s -> %-7s %.6f" %
(label_from, label_to, weight))

print("Top likely transitions:")
print_transitions(Counter(model.transition_
features_).most_common(20))

print("\nTop unlikely transitions:")
print_transitions(Counter(model.transition_
features_).most_common()[-20:])

```

Algûrîtm arow

Algûrîtm –a yessemras AROW ney s (Adaptive Regularization Of Weight Vector) i usleymu n yisefka.

```

model = CRF(
    algorithm='arow',
    min_freq=2, #min des occurences des
features au dessous du min on ignore
    all_possible_states=True,
    all_possible_transitions=True,

    #c1=0.1,
    #c2=0.01,
    max_iterations=100,# lbfsgs - unlimited;
l2sgd - 1000; ap - 100; pa - 100; arow -
100.
    #num_memories=6,
    epsilon=1e-5,# ap, arow, lbfsgs, pa
    #period=10, # The duration of
iterations to test the stopping criterion.
l2sgd, lbfsgs
    #delta=1e-5, # l2sgd, lbfsgs
    #linesearch= 'MoreThuente',#L-BFGS
values MoreThuente Backtracking
StrongBacktracking
    #max_linesearch=20,# The maximum number
of trials for the line search algorithm.
lbfsgs
    #calibration_eta=0.1, # l2sgd
)
#entraînement
model.fit(X1_train, y1_train)

labels = list(model.classes_)
#labels.remove('0')
Ponctuation= [ ';', '(', ')', '-',
'.', '!', '?', '"', ':', '$', ',']
for i in Ponctuation:

```

```

    labels.remove(i)
print(labels)
print("amdan n tecraq : ",len(labels))

y_pred = model.predict(X_test)
metrics.flat_f1_score(y_test, y_pred,
                      average='weighted',
                      labels=labels)

sorted_labels = sorted(
    labels,
    key=lambda name: (name[1:], name[0])
)

print(metrics.flat_classification_report(
    y_test, y_pred, labels=sorted_labels,
    digits=3
))

#print (model.score(X1_train, y1_train))

```

Asemres n tneyruft n ulmad asnalyan aseddasan s CRF

Alguritm n ucraq asnalyan aseddasan s usemres n CRF scikit-learn d yiwen-is akken tebyu tella tarrayt n ulguritm ama s lbtgs, l2sgd, ap, pa ney arow.

D isem kan n ufatlu i igebrén taneyruft ara ibeddlen. Deg yilguritmen n usleymu n yisefa i d-nge nnig-a, ismawen n yifuyal n tneyrufin dwig:

- Lbfgs: model-lbfgs.joblib
- L2SGD: model-l2sgd.joblib
- AP: model-ap.joblib
- PA: model-pa.joblib
- Arow: model-arow.joblib

Am wakken i t-id-nefka yakan deg usleymu s ulguritm n ulmad lqayen Keras deg usebter wis 21, d talyiwin-nni n wawal i nsemres yakan, ara nseqdec dayen deg ulguritm-a:

- Awal s timmad-is
- Ma yebda s usekkil amenzu
- ...atg.

```

# Sauvegarde du modèle
from joblib import dump
dump(model, 'model-arow.joblib')

from collections import Counter

def print_transitions(trans_features):
    for (label_from, label_to), weight in trans_features:
        print("%-6s -> %-7s %0.6f" %
              (label_from, label_to, weight))

print("Top likely transitions:")
print_transitions(Counter(model.transition_
features_).most_common(20))

print("\nTop unlikely transitions:")
print_transitions(Counter(model.transition_
features_).most_common()[-20:])

```



```

from sklearn_crfsuite import CRF
adris=[]
amenzu=0
udfiren=[]
uzwiren=[]
Asenqed=['...', ',', ';', '?', '!', ':', '"', '(', ')', '*', '_', '.', '[', ']', '{', '}', '<', '>', '+', '=', '\"', "'"]
for i in open("affixescolles.txt",encoding='utf-8'):
    a=i
    a=a.replace("\ufeff","");
    a=a.replace("\n","");
    if (a[len(a)-1]=="-"):
        uzwiren.append(str(a))
    else:
        udfiren.append(str(a))

```

```

def tokenize_awal(awal,udfiren,uzwiren):
    a=''
    amurfim=awal[0:awal.find('-')+1]
    awal_yebđan=''
    if (amurfim in uzwiren):
        awal=awal[awal.find(
        '-')+1:len(awal)]
        awal_yebđan=awal_yebđan+' '+amurfim
        while awal.find('-')>=0:
            amurfim=awal[0:awal.find(
            '-')+1]
            awal=awal[awal.find(
            '-')+1:len(awal)]
            awal_yebđan=awal_yebđan+' '+amurfim
            awal_yebđan=awal_yebđan+' '+awal
        else:
            amurfim=awal[0:awal.find(
            '-')]
            awal_yebđan=awal_yebđan+' '+amurfim
            awal=awal[awal.find(
            '-')+1:len(awal)]
            while awal.find('-')>=0:
                amurfim=awal[0:awal.find(
                '-')]
                awal_yebđan=awal_yebđan+' '+-
                '+amurfim
                awal=awal[awal.find(
                '-')+1:len(awal)]
                if ('-'+awal in udfiren):
                    awal_yebđan=awal_yebđan+' '+-
                    '+awal
                else:
                    awal_yebđan=awal_yebđan
    return awal_yebđan
def tokenize(sentence,udfiren,uzwiren):
    a=sentence.split()
    tafyirt1=""
    for i in a: #mots
        if(i.find('<')<0):
            tafyirt1=tafyirt1+' '+i
        else:
            awals=tokenize_awal(i,udfiren,uzwiren)
            tafyirt1=tafyirt1+' '+awals
            tafyirt1=tafyirt1.strip()
            return tafyirt1

def features(sentence, index):
    return {
        'word': sentence[index],      # Awal
        'timmad-is': s
        'is_first': index == 0,       # Ma
        'yezga-d deg tazwar n tefyirt'
        'is_last': index == len(sentence) - 1, # Ma yezgma-d deg taggar n tefyirt
        'is_capitalized':
        sentence[index][0].upper() == sentence[index][0], # MA ibeddu s usekkil
        'meqqren': meqqren
    }

    'is_all_caps':
    sentence[index].upper() == sentence[index],
    # Ma yura merr'a s usekkil meqqren
    'is_all_lower':
    sentence[index].lower() == sentence[index],
    # ma yura merr'a s usekkil mezziyen
    'prefix-1': sentence[index][0], #1
    'usekkil uzwir'
    'prefix-2': sentence[index][:2], #2
    'isekkilen uzwiren'
    'prefix-3': sentence[index][:3], #3
    'isekkilen uzwiren'
    'prefix-4': sentence[index][:4], #4
    'isekkilen uzwiren'
    'prefix-5': sentence[index][:5], #5
    'isekkilen uzwiren tettecmumuħend (aoriste intensif)
    'suffix-1': sentence[index][-1], #1
    'usekkil udfir'
    'suffix-2': sentence[index][-2:], #2
    'isekkilen udfiren'
    'suffix-3': sentence[index][-3:], #3
    'isekkilen udfiren'
    'suffix-4': sentence[index][-4:], #4
    'isekkilen udfiren'
    'suffix-5': sentence[index][-5:], #5
    'isekkilen udfiren
    'prev_word': '' if index == 0 else
    sentence[index - 1], #awal uzwir
    'prev1_word': '' if index == 1 else
    sentence[index - 2], #awal uzwir
    'next_word': '' if index ==
    len(sentence) - 1 else sentence[index + 1],
    #awal udfir
    'is_numeric':
    sentence[index].isdigit(), #ma yegber kan
    'izwilien
    'capitals_inside':
    sentence[index][1:].lower() !=
    sentence[index][1:] #ma yegber asekkil
    'meqqren daxel-is
    }

model = CRF()
def pos_tag(sentence,model):
    try:
        sentence_features =
        [features(sentence, index) for index in
        range(len(sentence))]
    except:
        return (sentence)
    return list(zip(sentence,
    model.predict([sentence_features])[0]))

from joblib import dump, load
#dump(model, 'lodel.joblib')
#print(metrics.flat_accuracy_score(y_test,
y_pred))
clf = load('model-arow.joblib')

f=
open("tokenized_text.txt","w+",encoding='ut
f-8')

```

```
h=
open("tagged_text.txt","w+",encoding='utf-
8')
g=open("brut_text.txt",encoding='utf-8')
for adur in g:
    #print (adur)
    for i in Asenqed:
        if i=='.':
            adur=adur.replace(i,' '+i+' ')
        else:
            adur=adur.replace(i,' '+i+' ')
adur=adur.replace("\ufeff","");
ligne=tokenize(adur,udfiren,uzwiren)
ligne=ligne.replace(" "," ")
ligne=ligne
izirig=""
izirig1=ligne

f.write(izirig1+'\n')
#print (izirig1.split(" "))
xx=pos_tag(izirig1.split(" "),clf)
for u in xx:
    try:
        yy=u[0]+'/'+u[1]
        izirig=izirig+yy+" "
    except:
        print (izirig1+"$$$$")
        f.close()
        g.close()
        h.close()
        exit()

    #print(xx)
    h.write(izirig+'\n')

    #t.sleep(0.5)
    #print ("tafyirt: ",ligne, "bettu:
",izirig1, "acrad :",izirig)

    f.close()
    g.close()
    h.close()
    print ("Yedda akken iwata!! Adris
yettwacred. ")
```

Tawuri n tizdit deg tseddast n tutayt taqbaylit

Asemres n usemsel awurman n tutlayin yefteq baylit

Aḥric III Tifin tawurmant n tutlayt

Aħric III: Tifin tawurmant n tutlay

Tazwert

Agħbur ass-a deg tallit n umađal umdin yettnerni s waṭas ladja deg Internet. Anadi n telyut d yisallen srid deg Internet mačċi dayen ara yexxem umdan s ufuś ladja ticki imdanen ttinadu tiybula n yisefka meqqren. Xef waya i ilaq usemsel ara yessifsusen anadi n telyut ney isallen yettwarun s tutlayt n umdan, maca dayen asismel n yiđri森 s tutlayt.

Tifin tawurmant n tutlayin d yiwen n taġult n usemsel awurman n tutlayin i d-ilehhun s tesledo̱t n yiđri森 akken ad d-naf tutlayt ideg ttwarun.

Tikwal izmer ad d-naf ddeqs n tutayin deg yiwen n warrat ney asebter web ney taffa n yisefka.



Asader n tegrummiwin n yisefka seg Tatoeba

Algoritmu ddaw-a ad d-yessader ifuyla n tegrummiwin n tefyar n 15 tulayin-a seg **Tatoeba: Taqbaylit, taglizit, tafransist, taṭelyanit, tabaskit, takaṭalant, tapurtugit, taspanit, talmanit, tahulandit, taswidit, tasṭunit, tasirbit, taṭurkit, akked thungarit**. Tzemrem ad ternum tutlayin-nniđen maca s useqdec n tengalt-nsent akken i d-ttwabedrent deg **ISO 639-3**. Ifuyla-a ad d-adren s talya “**bz2**”.

```
import wget
#wali tutlayin yellan deg tatoeba akken ad tessadre tigrummiwin n yisefka. Rnu tangalt n
tutlayt-a ddaw-a
tutlayin=['kab','eng','fra','ita','eus','cat','por','spa','deu','nld','swe','est','srp','tur',
'hun']#tingalin iso 639-3
for i in tutlayin:
    wget.download('https://downloads.tatoeba.org/exports/per_language/' +str(i)+
    '/'+str(i)+ '_sentences.tsv.bz2')
```

Asemsel uzwir n tegrummiwin n yisefka

Di tazwara, ticki udren-d yifuyla n tegrummiwin n yisefka nnig-a, semres Winrar ney Winzip akken ad ten-id-terred yer talya anida nezmer ad ten-nsemres. Ifuyla-a ttasen-d s talya **TSV**.

Yal afaylu n tutlayt yegħber iduren. Yal adur yegħber uṭtan, tangalt n tutlayt akked tefyirt. Talata-a n yiferdisen, b'dan s usekkil n trigla (tabulation). Yessefk ihi ad d-nekkes kan seg-s tifyar melba ma yedda-d wuṭtan ney tangalt n tutlayt.

6718492	kab	Ad t-ddmey.	➔ Ad t-ddmey.
6718488	kab	Ayyer tseqsayed?	➔ Ayyer tseqsayed?
6718483	kab	Tom ad yeqqim.	➔ Tom ad yeqqim.

```
tutlayin=['kab','eng','fra','ita','eus','cat','por','spa','deu','nld','swe','est','srp','tur','hun']

for tutlayt in tutlayin:
    afaylu_arewway= open(tutlayt+"_sentences.tsv",encoding='utf-8')
    afaylu_zeddigen=open(tutlayt+"_sentences.txt","w+",encoding='utf-8')

    for adur in afaylu_arewway:
        adur=adur.replace("\ufffd","");
        azalen = adur.split("\t")
        afaylu_zeddigen.write(azalen[2])
    afaylu_arewway.close()
    afaylu_zeddigen.close()
```

Ticki heggan yifuya n yal tutlayt, ad ten-nesdukkan. Deg umedya-a, nekkes-d 1000 n tefyar seg yal tutlayt. Wid yebyan asleymu s wugar n 1000 n tefyar, issefk kan ad ibeddel amdan-a. Afaylu "LanguageDetection.csv" yegber akk tukkisin n tefyar l d-yettwakken seg yifuya nnig-a.

```
limit=1000
tutlayin=['kab','eng','fra','ita','eus','cat','por','spa','deu','nld','swe','est','srp','tur','hun']
i=0
afaylu_yemmden= open("LanguageDetection.csv","w+",encoding='utf-8')
afaylu_yemmden.write("sentence"\t"language"\r")
for tutlayt in tutlayin:
    i=0
    afaylu_zeddigen=open(tutlayt+"_sentences.txt",encoding='utf-8')
    for sentence in afaylu_zeddigen:
        afaylu_yemmden.write(sentence.replace("\n","")+"\t"+tutlayt+"\r")
        i=i+1
        if limit!=0 and i>=limit:
            afaylu_zeddigen.close()
            break
    afaylu_zeddigen.close()
afaylu_yemmden.close()
```

Aheggi n tegramma n tefyar i usekyed

Ad d-nekkes tukkist seg tegramma i d-nessader yakan. Tifyar n tikkist-a ur yessefk ara ad ilint deg tegramma n ulmad. Tukkist-a tegber 10 n tefyar deg yal tutlayt. Beddel amdan-a ma tebyiq ugar ney drus.

```
limit=10
from_=5000
tutlayin=['kab','eng','fra','ita','eus','cat','por','spa','deu','nld','swe','est','srp','tur','hun']
i=0
afaylu_yemmden= open("Brut.csv","w+",encoding='utf-8')
for tutlayt in tutlayin:
    i=0
    afaylu_zeddigen=open(tutlayt+"_sentences.txt",encoding='utf-8')
    for sentence in afaylu_zeddigen:
        #print (sentence)
        if i>5000:
            afaylu_yemmden.write(sentence.replace("\n","").replace("\r","","")+"\r")
```

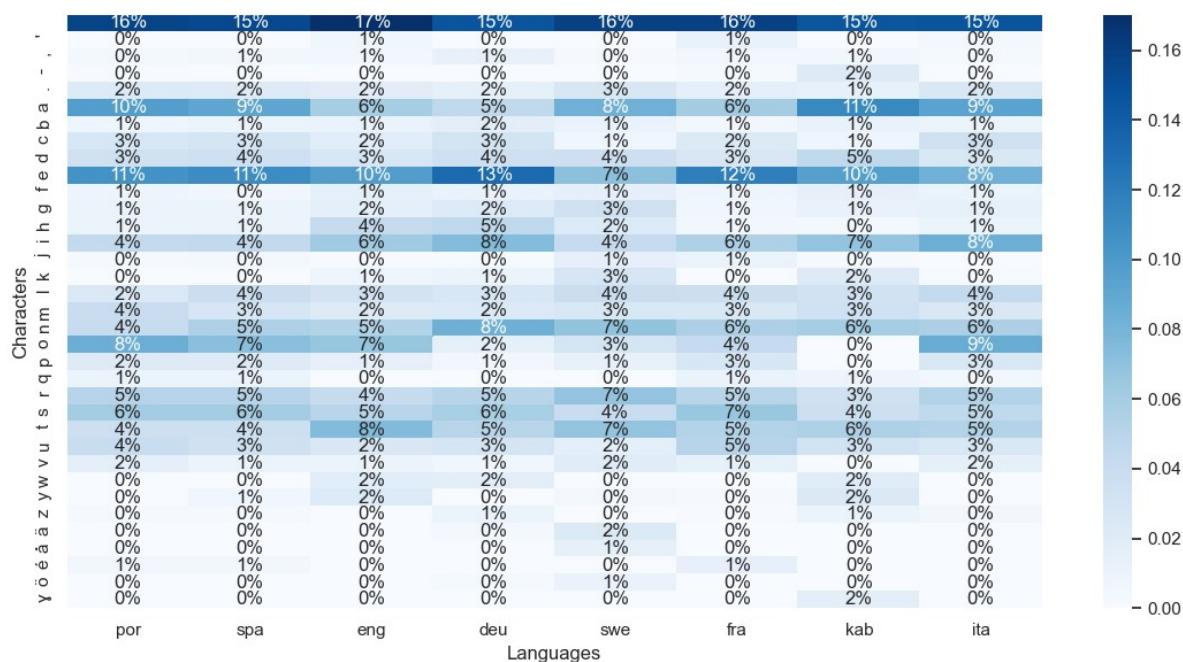
```
i=i+1
if limit!=0 and i>=limit+from_:
    afaylu_zeddigen.close()
    break

afaylu_zeddigen.close()
afaylu_yemmden.close()
```

Tasleđt n yisekkilen - unigram:

Di tesleđt n yisekkilen s unigram (yiwen n usekkil), yettban-d d akken tutlayin yettwarun s unargraw n tira alaṭini mqarabent meṛṛa. Xef waya, anaw-a n tesleđt ur yezmir ara ad yeęqel akken iwata tutlayt. Xef waya i tlaq tesleđt yef yiwen kan n usekkil ur temir ara ad teęqel tutlayt.

1. por 64
2. spa 55
3. eng 50
4. deu 57
5. swe 49
6. fra 66
7. kab 51
8. ita 49



```

import pandas as pd
from sklearn.feature_extraction.text import
CountVectorizer
import seaborn as sn
import matplotlib.pyplot as plt

raw = pd.read_csv("LanguageDetection.csv",
sep='\t')
languages = set(raw['language'])
import numpy as np
from sklearn.model_selection import
train_test_split

X=raw['sentence']
y=raw['language']

X_train, X_test, y_train, y_test =
train_test_split(X, y, test_size=0.2,
random_state=42)
unigramVectorizer =
CountVectorizer(analyzer='char',
ngram_range=(1,1))
X_unigram_train_raw =
unigramVectorizer.fit_transform(X_train)
X_unigram_test_raw =
unigramVectorizer.transform(X_test)

unigramFeatures =
unigramVectorizer.get_feature_names()

def train_lang_dict(X_raw_counts, y_train):
    lang_dict = {}
    for i in range(len(y_train)):
        lang = y_train[i]
        v = np.array(X_raw_counts[i])
        if not lang in lang_dict:
            lang_dict[lang] = v
        else:
            lang_dict[lang] += v

    # to relative
    for lang in lang_dict:
        v = lang_dict[lang]
        lang_dict[lang] = v / np.sum(v)

    return lang_dict
language_dict_unigram =
train_lang_dict(X_unigram_train_raw.toarray
(), y_train.values)
def getRelevantCharsPerLanguage(features,
language_dict, significance=1e-30):
    relevantCharsPerLanguage = {}
    for lang in languages:
        chars
        chars = []
        relevantCharsPerLanguage[lang] =
        v = language_dict[lang]
        for i in range(len(v)):
            if v[i] > significance:
                chars.append(features[i])
        return relevantCharsPerLanguage

relevantCharsPerLanguage =
getRelevantCharsPerLanguage(unigramFeatures
, language_dict_unigram)
europeanLanguages = ['por', 'spa', 'eng',
'deu', 'swe', 'fra', 'kab', 'ita']
for lang in europeanLanguages:
    print(lang,
len(relevantCharsPerLanguage[lang])))

relevantChars_OnePercent =
getRelevantCharsPerLanguage(unigramFeatures
, language_dict_unigram, 0.009)

# collect and sort chars
europeanCharacters = []
for lang in europeanLanguages:
    europeanCharacters +=
relevantChars_OnePercent[lang]
europeanCharacters =
list(set(europeanCharacters))
europeanCharacters.sort()

# Construirees données
indices = [unigramFeatures.index(f) for f
in europeanCharacters]
data = []
for lang in europeanLanguages:
    data.append(language_dict_unigram[lang][ind
ices])

#build dataframe
df = pd.DataFrame(np.array(data).T,
columns=europeanLanguages,
index=europeanCharacters)
df.index.name = 'Characters'
df.columns.name = 'Languages'
# Affichage de la distribution unigram
sn.set(font_scale=0.8) # for label size
sn.set(rc={'figure.figsize':(10, 10)})
sn.heatmap(df, cmap="Blues", annot=True,
annot_kws={"size": 12}, fmt='%.0%')# font
size
plt.show()

```

Tasledoñt n yisekkilen s Bigrams (amsedfer n 2 n yisekkilen)

D tasledoñt-a ara nsemres i ueqal n tutlayt. Asemsel-a, ad ay-d-yefk talya n umsedfer n tyugiwin n yisekkien deg uđris. Xas wama tutlayin-a i nesled seqdacen yiwen n unagraw n tira, maca tiyugiwin n yisekkilen mgaradent akken itemsedfareñ.

```
bigramVectorizer = CountVectorizer(analyzer='char', ngram_range=(2,2))

X_bigram_raw = bigramVectorizer.fit_transform(X_train)

bigramFeatures = bigramVectorizer.get_feature_names()

print('Number of bigrams', len(bigramFeatures))

# top bigrams (>1%) for Spanish, Italian (Latin), English, Dutch, Chinese, Japanese, Korean
language_dict_bigram = train_lang_dict(X_bigram_raw.toarray(), y_train.values)

relevantCharsPerLanguage = getRelevantCharsPerLanguage(bigramFeatures, language_dict_bigram,
significance=1e-2)

for lang in europeanLanguages:

    print(lang, " has ", relevantCharsPerLanguage[lang], ' top gigrams >1%')
```

Ddaw-a, tasledoñt n umsedfer n tyugiwin n yisekkilen I d-yettuyalen s wañas, temmal-d amgired gar tutlayin xas akken ttwarunt s yiwen n unagraw n tira:

- por has ['a', 'c', 'd', 'e', 'p', 'a', 'ar', 'de', 'e', 'er', 'es', 'm', 'o', 'os', 'r', 'ra', 's', 'te', 'u']
- spa has ['a', 'c', 'd', 'e', 'l', 'm', 'p', 'a', 'ar', 'as', 'de', 'e', 'en', 'er', 'es', 'la', 'n', 'o', 'qu', 's', 'ue']
- eng has ['a', 'i', 's', 't', 'w', 'an', 'd', 'e', 'er', 'he', 'in', 'n', 'o', 'on', 'ou', 're', 's', 't', 'th']
- deu has ['d', 'e', 'i', 'm', 's', 'w', 'ch', 'de', 'e', 'ei', 'en', 'er', 'ge', 'h', 'ic', 'ie', 'in', 'n', 'r', 's', 'st', 't', 'te']
- swe has ['d', 'h', 'm', 's', 'a', 'ag', 'an', 'ar', 'de', 'e', 'en', 'er', 'et', 'g', 'in', 'ja', 'n', 'r', 't', 'te', 'tt', 'är']
- fra has ['c', 'd', 'l', 'p', 's', 'ai', 'de', 'e', 'en', 'es', 'le', 'n', 'nt', 'on', 'ou', 'qu', 're', 's', 't']
- kab has ['a', 'd', 'i', 'n', 't', 'y', 'a', 'an', 'ar', 'd', 'en', 'i', 'n', 'te', 'ye']
- ita has ['a', 'c', 'd', 'p', 's', 'a', 'an', 'ar', 'e', 'er', 'i', 'n', 'no', 'o', 'on', 're', 'ta', 'to']

Amsegfer n tyugiwin n yisekkilen d tarrayt I izemren ad taf tutlayt n uđris, maca tiseddi tikwal ur meqqret ara. Yef waya, ad nsemres algoritm n Naïve-Bayes n scikit-learn akken ad nesnerni tiseddi.

Asleymu n tegrumma n yisefka i tifin n tutlayt.

Algoritm ddaw-a ilemmmed seg uđris i as-nefka sakin ad d-yesnulfu taneyruft n ueqal n tutlayt I igebrén 15 n tutlayin (tid yellan deg tegrumma n yisefka n ulmad). Tarrayt-a tebna yef ulguritmus **Naive-bayes**.

```
import pandas as pd
import numpy as np
import re
import seaborn as sns
import matplotlib.pyplot as plt
import warnings
from sklearn.preprocessing import LabelEncoder
```

```
from sklearn.model_selection import train_test_split
from sklearn.naive_bayes import MultinomialNB
from sklearn.metrics import accuracy_score, confusion_matrix, classification_report
from sklearn.feature_extraction.text import CountVectorizer
from pickle import dump

data = pd.read_csv("LanguageDetection.csv", sep='\t')

X = data["sentence"]
y = data["language"]
le = LabelEncoder()
y = le.fit_transform(y)
dump(le, open('modelle.pkl', 'wb'))
data_list = []
for text in X:
    text = re.sub(r'[@#$(),;~`0-9]', ' ', text)
    text = re.sub(r'[]', ' ', text)
    # converting the text to lower case
    text = text.lower()
    # appending to data_list
    data_list.append(text)

cv = CountVectorizer()
X = cv.fit_transform(data_list).toarray()
dump(cv, open('modelcv.pkl', 'wb'))
x_train, x_test, y_train, y_test = train_test_split(X, y, test_size = 0.20)
model = MultinomialNB()
model.fit(x_train, y_train)
dump(model, open('model.pkl', 'wb'))
y_pred = model.predict(x_test)

ac = accuracy_score(y_test, y_pred)
print (ac)
```

Tifin n tutlayt n tefyirt

Alguritm dda-w, yettarra-d tutlayt n tefyirt i as-nefka. Dagi nefka-as tayirt s teqbaylit. Maca taneyruft i as-nefka tezmer kan ad taf tifyar n tutlayin i as-nselmed. Dagi 15 kan n tutlayin.

```
from sklearn.preprocessing import LabelEncoder
from sklearn.naive_bayes import MultinomialNB
from sklearn.feature_extraction.text import CountVectorizer
from pickle import load
le = LabelEncoder()
le = load(open('modelle.pkl', 'rb'))
cv = CountVectorizer()
cv = load(open('modelcv.pkl', 'rb'))
model = MultinomialNB()
model = load(open('model.pkl', 'rb'))
def predict(text,cv,model,le):
    x = cv.transform([text]).toarray() # converting text to bag of words model (Vector)
    lang = model.predict(x) # predicting the language
    lang = le.inverse_transform(lang) # finding the language corresponding the the predicted value
    print("tutlay n :",text," d :",lang[0]) # printing the language

predict ("Wagi d argaz ameqqranc.",cv,model,le)
```

Asismel n tefyar s tutlayt

Deg ulguritm ddaw-a, ad neered ad nsismel tiyar yellan deg ufaylu Brut.csv. Afaylu-a, yegber tifyar deg 15 n tutlayin. Alguritm-a ad yessemres taneyruft **model.pk** i d-ihegga seg ulmad n yiðrisen I as-nefka deg ulguritm yzerin.

```

from sklearn.preprocessing import LabelEncoder
from sklearn.naive_bayes import MultinomialNB
from sklearn.feature_extraction.text import CountVectorizer
from pickle import load

le = LabelEncoder()
le = load(open('modelle.pkl', 'rb'))

cv = CountVectorizer()
cv = load(open('modelcv.pkl', 'rb'))

model = MultinomialNB()
model = load(open('model.pkl', 'rb'))

tutlayin=['kab','eng','fra','ita','eus','cat','por','spa','deu','nld','swe','est','srp','tur',
'hun']
tifyar=[]
for i in tutlayin:
    tifyar.append([])
a=[]

def asismel (afaylu, model,cv,le):
    afaylu_araway=open(afaylu,encoding='utf-8')
    for sentence in afaylu_araway:
        sentence=sentence.replace('\n','')
        x = cv.transform([sentence]).toarray() # converting text to bag of words model
    (Vector)
        lang = model.predict(x) # predicting the language
        lang = le.inverse_transform(lang) # finding the language corresponding to the predicted value
        a=tifyar[tutlayin.index(lang)]
        a.append(sentence)
        tifyar[tutlayin.index(lang)]=a

afaylu="Brut.csv"
asismel (afaylu,model,cv,le)

for i in tutlayin:
    print (i, ':')
    for j in tifyar[tutlayin.index(i)]:
        print (j)

```

Aktazel n tneyrufin: Tiseddi n ucrad

Ticki yemmed usleymu n yisefka, izmer ad d-nesken tiseddi n tneyruft akka:

```
print (model.score(X1_train, y1_train))
```

Tafelwit ddaw-a temmal-d tiseddi n tneyrufin i d-yeffyen seg usemres n yiwt n tegrumma n yisefka isnalyanen n tutlayt taqbaylit i ulmad awurman. Yettban d akken algoritm **pa** i d-yettarran tiseddi meqqren.

Algoritm	Lbfgs	I2sgd	ap	Pa	Arow
Tiseddi	0.9866861935827453	0.9802955665024631	0.9897483690587139	0.9899480761549727	0.992943682598855
Algoritm pa yettarra-d tiseddi n 98.99%. Itusemma yef 100 n wawlen, aselkim ad yecced deg tecredt n yiwen kan n wawal.					

**Aḥric IV : Imseytiyen n tira.
Tazwert yer umawal umdīn i
tutlayt taqbaylit.**

Aħric III: Imsejtyen n tira i teqbaylit

Tazwert

Imsejtyen n tira ttaken-d tallelt i yimdanen akken ad arun war tuccdiwin. Maca dayen ttwasemrasen deg ddeqs n yisemsal iwurmanen i yesseytayen tira uqbel asemres n wawalen ama deg unadi, asismel, ney deg tyula-nniđen.

Deg uħriċ-a ad d-nemmeslay yef Hunspell, yiwt n tarrayt i yettwasemrasen deg tneflit n yimsejtyen n tira deg uselkim ney deg tiliyirin.

Hunspell d tagrumma n yifecka n tneflit n yimsejtyen n tira, dayen d amaslađ n talya n wawalen. Yettwasemras deg yiminigen ilelliġen am Firefox d Chrome, akked yifecka am LibreOffice d OpenOffice, inagrawen n wammud am Linux d macOS. Hunspell yebna yef wawalen iħerfiyen ansa i d-yessiliż awalen-nniđen s usemres n yilugan n usilej.

Aheggi n twennaqt n tneflit

Tarrayt-a, nhegga-tt-id i wid yebyan ad sneflin imsejti n tira i tutlayt taqbaylit. Tawennaqt-a terza wid yesseqdacenanagraw n wammud Ubuntu. Ddaw-a, ad tafem amek ara theggim tawennaqt.

1. Selkem **apt-get install -y hunspell** akken ad tesbeddeħ hunspell, afecku n tnefli d usekyed n yimsejti.
2. Ddu yer GitHubdeg tansa-a <https://github.com/mozillakab/hunspell-kab>: Sader-d talata-a n yifula. Ifuyla kab.dic akked kab.aff d imedyaten kan:
 - a. Kab.dic afaylu n umawal, afaylu-a yeğber awalen s talya taħerfit d wuṭṭunen n yilugan i yellan deg ufaylu n yilugan.
 - b. Kab.aff afaylu n yilugan n usiley
 - c. Wordforms.sh , afecku n tesleħt n usiley n talyiwin n wawal
3. Selkem **sudo wordforms kab.aff kab.dic azemmur** akken ad twaliż talyiwin i d-yekkan seg wawal **azemmur**.

```
odoo@odoo:/home/kab$ sudo wordforms kab.aff kab.dic azemmur
azemmur
izemmran
tazemmurt
tzemmrin
tzemmurin
tzemmurt
tizemmrin
tizemmurin
vizemmranc
```
4. Selkem **sudo unmunch kab.dic kab.aff** akken ad twaliż ilugan n usiley n wawalen merra i yellan deg ufaylu n yilugan n usiley **kab.aff**

Talya d usiley n wawalen deg teqbaylit

Taqbaylit d tutlayt anida talya n wawal tettbeddil s waṭas ama yef sebba n yizwiren d yiḍfiren i nrennu ney deg ubeddel n kra n yisekkilen deg tazwara, talemmast neş taggara n wawal. Aya yerza ladya ismawen, irbiben akked yimyagen. Taqbaylit d tutlayt timneteqd, timleywit.

Ismawen d yirbiben di teqbaylit ttbeddilen talya s ubeddel n umdan, tawsit akked waddad. Ma d amyag, yettbeddil talya deg tsefsit s tmera n yimataren udmawanen akked tmezri.

Deg yismawen, ma netṭef ameda n wawal “**amyar**”, asiley ad d-yefk awalen: **tamyart, temyart, imyaren, yemyaren, yimyaren, timyarin, temyarin**. Deg umedya-a, awal amyar yur-s

Tayessa n yimawalen Hunspell

Hunspell yesra sin n yifuya akken ad yizmir ad yesseyti tira: afyalu n umawal i igebrén awalen d tecraq yettwasemrasen fell-as, akked ufaylu n yiwsilen i d-yemmalen amek ttwasemrasent tecraq-a yef wawalen. Llan yifuya-nniđen i iṭṭafaren tawennađt anida ara yettwasemres yimseyti-a am deg LibreOffice, OpenOffice, Firefox, Chrome...atg. Ugar nyisallen ad ten-id-nefk deg taggara n uhric-a.

Afaylu n umawal:

Afaylu-a yettili-d s uziżef “**.dic**”. Yegber umuy n wawalen deg talya-nsen taħerfit akken i ten-nettaf deg yimawalen. Yal awal deg yiwen n udur ney izirig.

Deg yizirig amenu ad naf amdan n wawalen i yellan deg umawal. Awal izmer ad t-id-yedfer uzamul “/” akked tecraq ara nsemres fell-as. Ticraq-a mmalen-d amek i iderru usiley n wawalen-nniđen ara d-yeħken seg wawal-a. Asiley-a yetṭafar asmil anejrum n wawal. Ma d isem ney arbib, ad nsemres ilugan n umdan, addad amaruz akked tewsit. Ma d amyag, ad ffyent seg-s meṛra talyiwin-is i d-ittekken seg tsefsit. Ticki awal ur t-id-ṭṭafarent ara ticraq, aya yemmal-d d akken ulac isuddam ney talyiwin-nniđen i izemren ad d-kkent seg wawal-a. Aya yerza imqimen, tizelyiwin, Ismawen uzzigen (imdanen, imukan, tuddsiwin..), imerna, tisyunin.

Amedya n ufatlu n umawal:

3
aman
argaz/ABCD
tala/KLM

Deg umedya nnig-a, uṭṭun **3** yemmal-d achal n wawalen iħerfiyen I yellan deg umawal.

- **3** d amdan n wawalen
- **ABCD** d ilugan n usiley, dagi awal **argaz** yur-s **4** n ilugan n usiley: **A,B,C,D**
- **KLM** d ilugan n usiley, dagi awal **tala** yur-s **2** n ilugan n usiley: **K, L, M**

Afaylu n yilugan

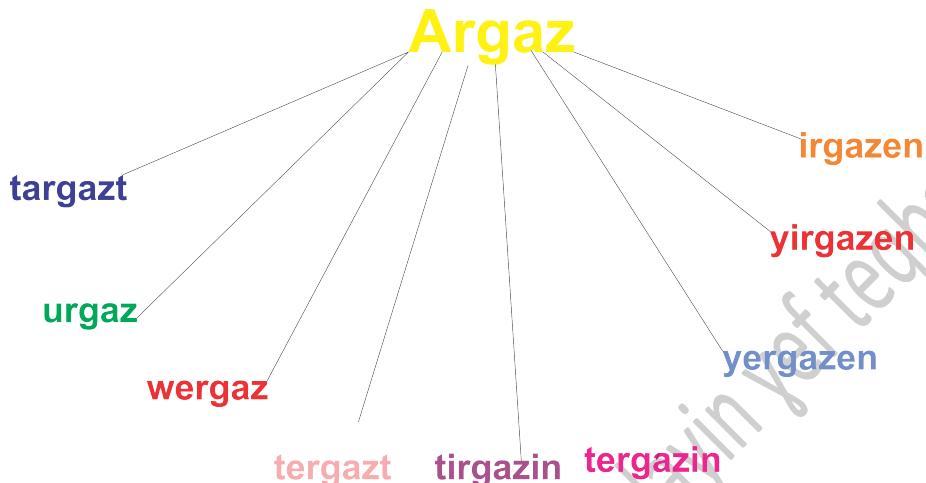
Afaylu n yilugan yegber umuy n yilugan i nezmer ad nsemres yef wawalen yellan deg ufatlu n umawal akken ad d-nsiley seg-s meṛra talyiwin-nniđen i icudden yer umdan, tawsit, addad ma d isem, ney talyiwin i icudden yer tmezri ma yella awal d amyag.

Amedya 1: Isem

Seg wawal “**argaz**” izmer ad d-nsiley 9 n wawalen ma nmuel yer umdan, addad akked tewsit avku d isem.

Dya ad d-ffyen wigi: **urgaz**, **wargaz**, **targazt**, **tergazt**, **irgazen**, **yirgazen**, **yerazen**, **tergazin**.

Yal talya n wawal, issek ad nseddu ilugan n yiwişilen uzwiren d yiwişilen udfireni izemren ad netdən deg wawal **argaz**.



Deg umedya-a, newwi snat n talyiwin n waddad amaruz.

- Argaz-> urgaz , **alugen** : nekkes **a**, nerra **u** deg umdiq-is (uzwir)
- Argaz-> wargaz , **alugen** : nekkes **a**, nerra **we** deg umdiq-is (uzwir)
- Argaz-> targazt , **alugen** : nerna **t** deg tazwara (awsil uzwir) , nerna **t** deg taggara (udfir)
- Argaz-> tergazt , **alugen** : nekkes **a**, nerna **te** deg tazwara (uzwir) , nerna **t** deg taggara (udfir)
- Argaz->irgazen, **alugen**: nekkes **a**, nerra **i** di tazwara (udfir) , nerna **en** deg taggara (udfir)
- Argaz->yirgazen, **alugen** : nekkes **a** deg tazwara, nerra **yi** (uzwir), nerna **en** deg taggara (udfir)
- Argaz->yerazen, **alugen** : nekkes **a** deg tazwara, nerra **ye** (uzwir),nerna **en** deg taggara(udfir)
- Argaz-> tirgazin, **alugen** : nekkes **a** seg tazwara, nerra **ti** (uzwir), nerna **in** deg taggara (udfir).
- Argaz-> tergazin. **alugen**: nekkes **a** seg tazwara, nerra **te** (uzwir), nerna **in** deg taggara (udfir).

Sumata, talyiwin n wawalen i d-itekken seg yisem n talya taherfit d tigi:

5. Unti/amalay n wasuf ilelli (2 n talyiwin)
6. Unti/amalay n wasuf amaruz (2 n talyiwin)
7. Unti/amalay n usget ilelli (2 n talyiwin)
8. Unti/amalay n usget amaruz (2 n talyiwin)

Awalen i ittafareni ilugan n usiley am wawal **argaz**, ad seun merrä ilugan n usiley-is deg ufatlu n umawal.

Amedya 2 amyag

Seg wawal **ddem**, nezmer ad d-nsiley 40 n talyiwin yemgaraden

ddmet	neddem	yeddim	ddiment
ddmemt	teddmem	iddim	yeddmen
ddmey	teddmemt	teddim	iddmen
teddmed	ddmen	neddim	teddmet
yeddem	ddment	teddimem	teddmey
iddem	ddimey	teddimemt	tteddmed
teddem	teddimed	ddimen	iteddem

tteddem	iteddem	yettaddam	ttaddamen
nteddem	ttaddam	ittaddam	ttaddament
tteddmem	ttaddamet	tettaddam	yettaddamen
tteddmemt	ttaddamemt	nettaddam	ittaddamen
teddmen	ttaddamey	tettaddamem	
teddment	tettaddamed	tettaddamemt	

Tayessa n ufaylu n yilugan :

Ddaw-a, ad d-nefk ilugan n usiley n talyiwin i wawalen argaz,axxam, ugur, iri.

- Awal argaz ad d-kkent seg-s talyiwin-a: wargaz, wergaz, urgaz, tergazin, irgazen, targatz, tirgazin, yirgazen, yergazen
- Awal axxam ad d-kkent seg-s talyiwin-a: wexxam, uxxam, texxamin, ixxamen, taxxamt, tixxamin, yixxamen,yexxamen
- Awal ugur ad kken-t segès talyiwin-a: wugur , uguren, wuguren
- Awal imi ad d-kkent seg-s talyiwin-a: yiri, iran, yiran.

Afyalu n umawal:

11
argaz/ABCGILPR
isli/ABCGILPR
axxam/ACGILPR
ugur/DS
iri/EUV
ifri/EUV
amyar/GACIPRL
azagur/GACIPRL
abexsis/GACIPRL
akermus/GAIPRL
aman/B

Deg ufaylu-a llan 11 n wawalen:

- Argaz: yettafar 9 n yilugan n usiley
- Isli : yettafar 9 n yilugan n usiley
- Axxam : yettafar 7 n yilugan n usiley
- Ugur : yettafar 2 n yilugan n usiley
- Iri : yettafar 3 n yilugan n usiley
- Ifri : yettafar 3 n yilugan n usiley
- Amyar: yettafar 6 n yilugan n usiley
- Azagur : yettafar 6 n yilugan n usiley
- Abexsis : yettafar 6 n yilugan n usiley
- Akermus : yettafar 6 n yilugan n usiley
- Aman : yettafar 1 n ulugen n usiley

Afaylu n yilugan n usiley:

```
SET UTF-8
#addad amaruz n wasuf amalay...
PFX A Y 1
PFX A a u a #a->u
PFX B Y 1
PFX B a wa a # a->wa
PFX C Y 1
PFX C a we a #a->we
PFX D Y 1
PFX D u wu u #u->wu
PFX E Y 1
PFX E i yi i #i->yi
CIRCUMFIX X
###addad amaruz n usget unti
PFX Q Y 1
PFX Q a te/X a
###
SFX R Y 1
SFX R 0 in/QX .
#unti
PFX F Y 1
PFX F 0 t/X .
SFX G Y 1
SFX G 0 t/FQX .
###unti asget
PFX H Y 1
PFX H a ti/X a
SFX I Y 1
```

```

SFX I 0 in/QHX .
##
###asget amalay
PFX K Y 1
PFX K a i/X a
SFX L Y 1
SFX L 0 en/KX .
SFX S Y 1
SFX S 0 en .
##
###addad amaruz n usget amalay
PFX M Y 1
PFX M a yi/X a

PFX O Y 1
PFX O a ye/X a
SFX P Y 1
SFX P 0 en/MOX .
##
###addad amaruz n usget amalay
SFX V Y 1
SFX V i an i
PFX T Y 1
PFX T i yi/X i
SFX U Y 1
SFX U i an/TX i

```

Taseddast n yilugan Hunspell

Asiley n talyiwin s tmerna n yiwsilen, tukksa ney abeddel n yisekkilen

Hunspell yessefrak kan timerna, abeddel ney tukksa n yizwiren d yiñfiren. Hunspell ur issefrak ara talyiwin i d-tekken seg ubeddel deg tlemmast n wawal am usget agensan n kra n yismawen ney abeddel n yisekkilen deg kra n yimyagen. Sumata, Hunspell yessiliy-d kan talyiwin yebnan yef tmerna n yizwiren d yiñfiren.

Yiwen n wuzwir

Alugen-a irennu kan yiwen n wuzwir. Yettwasemras ladya deg usiley n waddad amaruz n umalay asuf n yismawen akked kra n talyiwin n umyag:

Amedya:

Uṭun	Asiley	Afaylu n yilugan Hunspell	Talya	Afaylu n umawal Hunspell
1	Argaz->Urgaz	PFX A Y 1 PFX A a u a	Asiley n waddad amaruz n umalay asuf	argaz/A amyar/A axxam/A
2	Ečč->Nečč, Tečč, Yečč	PFX B Y 3 PFX B 0 t . PFX B 0 n . PFX B 0 y .	Asiley n 3 n talyiwin n kra n yimyagen deg tmezri n wurmir aherfi	ečč/B awi/B els/B ali/B

Asegzi:

Alugen1:

argaz/A, amyar/A, axxam/A



PFX A Y 1
PFX A a u a



urgaz, umyar, uxxam

PFX A Y 1 : PFX yemmal-d d akken alugen-a d azwir. A d isem n ulugen. Y 1 d amdan n wudmawen n usiley

PFX A a u a: PFX yemmal-d d akken asiley-a s yessemras azwir. A d isem n ulugen, a d asekkil ara nbeddel deg tazwara, u d asekkil ara nerr deg umdiq n usekkil n tazwara, a yemmal-d d akken awal ara d-nsiley ilaq ad yebdu s usekkil "a".

Deg ufatlu n umawal awalen **argaz**, **amyar**, **axxam** tafaren alugen n usiley A. Imseyti ad d-yessiley awalen **urgaz**, **umyar**, **uxxam**.

Alugen 2:

ečč/B, awi/B, els/B, ali/B



PFX B Y 3

PFX B 0 t .

PFX B 0 n .

PFX B 0 y .



**tečč, nečč, yečč, tawi,
nawi, yawi, tels, nels,
yels, tali, nali, yali**

PFX B Y 3 : PFX yemmal-d d akken alugen-a d azwir. B d isem n ulugen. Y 3 d amdan n wudmawen n usiley

PFX B 0 t .: PFX yemmal-d d akken akken asiley-a yessemras azwir. B d isem n ulugen, 0 yemmal-d d akken ad nernu asekkil yer tazwara n wawal, t d asekkil ara nernu deg tazwara, . (**taneqqidt**) temmal-d akken awal ara d-nsiley izmer ad yebdu s yal asekkil.

PFX B 0 n .: PFX yemmal-d d akken asiley-a yessemras azwir. B d isem n ulugen, 0 yemmal-d d akken ad nernu asekkil yer tazwara n wawal, n d asekkil ara nernu deg tazwara, . (**taneqqidt**) temmal-d d akken awal ara d-nsiley izmer ad yebdu s yal asekkil.

PFX B 0 y .: PFX yemmal-d d akken asiley-a yessemras azwir. B d isem n ulugen, 0 yemmal-d d akken ad nernu asekkil deg tazwara n wawal, y d asekkil ara nernu deg tazwara, . (**taneqqitt**) temmal-d d akken awal ara d-nsiley izmer ad yebdu s yal asekkil.

Deg ufatlu n umawal awalen **ečč**, **awi**, **els**, **ali** tafaren alugen n usiley B. Imseyti ad d-yessiley awalen **tečč**, **nečč**, **yečč**, **tawi**, **nawi**, **yawi**, **tels**, **nels**, **yels**, **tali**, **nali**, **yali**. Wigi d kra n talyiwin n umyag deg wurmir aherfi.

Yiwen n wuđfir

Alugan-a lrennu kan yiwen n wuđfir. Yettwasemras deg kra n talyiwin n usget n yismawen, akked kra n talyiwin n yimyagen

Amedya:

Uṭṭun n ulugen	Asiley	Afaylu n yilugan Hunspell	Talşa	Afaylu n umawal Hunspell
1	Ifri-> ifran	SFX C Y 1 SFX C i an i	Asiley n usget amalay n kra n yismawen	ifri/C imyi/C izli/C
2	Awi->Awiy, Awin, Awint	SFX D Y 3 SFX D 0 γ i SFX D 0 n i SFX D 0 nt i	Asiley n 3 n talyiwin n kra n yimyagen deg tmezri n wurmir ahifi	awi/D ali/D argu/D ili/D

Asegzi:

Alugen1:

ifri/C, imyi/C, izli/C


SFX C Y 1
SFX C i an i


ifran, imyan, izlan

SFX C Y 1 : SFX yemmal-d d akken alugen-a d adfir. C d isem n ulugen. Y 1 d amdan n wudmawen n usiley

SFX C i an i : SFX yemmal-d d akken asiley-a s usemres n wudfir. C d isem n ulugen, i d asekkil ara nbeddel deg taggara n wawal, an d iseikkilen ara nerr deg umdinq n usekkil n taggara, i yemmal-d d akken awal ara d-nsiley ilaq ad yebdu s usekkil “i”.

Deg ufatlu n umawal awalen **ifri, imyi, izli** ttafareen alugen n usiley **C**. Imseyti ad d-yessiley awalen **ifran, imyan, izlan**.

Alugen 2:

**awi/D, ali/D
argu/D, ili/D**


SFX D Y 3
SFX D 0 γ [iu]
SFX D 0 n [iu]
SFX D 0 nt [iu]


**awiy, awin, awint, aliγ,
alin, alint, arguγ, argun,
argunt, iliγ, ilin, ilint**

SFX D Y 3: yemmal-d d akken alugen-a d aðfiir. **D** d isem n ulugen. **Y 3** d amðan n wudmawen n usiley

SFX D 0 y [iu] : **SFX** yemmal-d d akken asiley-a yessemras aðfir. **D** d isem n ulugen, **0** yemmal-d d akken ad nernu asekkil ney isekkilen yer deffir, **y** d asekkil ara nernu yer taggara n wawal, **i** yemmal-d d akken awal ara d-nsiley ilaq ad yeffak s usekkil i ney u.

SFX D 0 n [iu] : **SFX** yemmal-d d akken asiley-a yessemras aðfir. **D** d isem n ulugen, **0** yemmal-d d akken ad nernu asekkil ney isekkilen yer deffir, **n** d asekkil ara nernu yer taggara n wawal, **i** yemmal-d d akken awal ara d-nsiley ilaq ad yeffak s usekkil i ney u.

SFX D 0 nt [iu]: **SFX** yemmal-d d akken asiley-a yessemras aðfir. **D** d isem n ulugen, **0** yemmal-d d akken ad nernu asekkil ney isekkilen yer deffir, **nt** d asekkil ara nernu yer taggara n wawal, **i** yemmal-d d akken awal ara d-nsiley ilaq ad yeffak s usekkil i ney u.

Deg ufatlu n umawal awalen **awi, ali, argu, ili** tñafaren alugen n usiley **D**. Imseyti ad d-yessiley awalen **awiy, awin, awint, aliy, alin, alint, arguy, argun, argunt, iliy, ilin, ilint**. Wigi d kra n talyiwin n umyag deg wurmir aherfi.

Yiwen n wuzwir d yiwen n wudfir

Alugen-a irennu (ney itekkes, ney ittbeddil) i wawal yiwen n uzwir d yiwen n wudfir. Talyiwin-a ad tent-naf deg usiley n usget, unti, addad amaruz n wunti akked usget amalay ney unti n yismawen d yirbiben, maca dayen deg ddeqs n talyiwin n yimyagen.

Amedya:

Uttun n ulugen	Asiley	Afaylu n yilugan Hunspell	Talya	Afaylu n umawal Hunspell
1	Axxam -> Taxxamt, Ixxamen, Yixxamen, Tixxamin, Texxamin	CIRCUMFIX X PFX E Y 1 PFX E O t/X a SFX F Y 1 SFX F O t/EX .	Asiley n wunti	axxam/FHJ amyar/FHJ zniq/FHJ
2		PFX G Y 2 PFX G a i/X a PFX G a yi/X a SFX H Y 1 SFX H O en/GX .	Asiley n usget amaray d waddad amaruz n usget amalay	
3		PFX I Y 2 PFX I a te/X a PFX I a ti/X a SFX J Y 1 SFX J O in/IX .	Asiley n usget unti d waddad amaruz n usget unti	
4	Ali-> talid, talim, talimt ntu-> tentuð, tentum, tentumt Sentu-> tsentuð, tsentum, tsentumt	PFX K Y 3 PFX K O t/X [aieu] PFX K O te/X [^auei][^auei] PFX K O t/X [^auei][auei] SFX L Y 3	Asiley n kra n talyiwin n kra n yimyagen deg tmezri n wurmir aherfi	awi/L ini/L argu/L ili/L sentu/L ntu/L

SFX L 0 q/KX [iu]
SFX L 0 mt/KX [iu]
SFX L 0 m/KX [iu]

Asemres n usemsel awurman n tutlayin yefteq baylit

Asegzi :

Deg yilugan nnig-a, talyiwin ara d-nsiley gebren azwir d wudfir.

CIRCUMFIX X : Yettwaseqdac ticki asiley yessemras sin ney ugar n yiwsilen, **X** d azamul i d-yemmalen d akken awsil-a ad yettwasemres deg ulugen-nniđen n usiley.

Alugen 1:

axxam/F, amyar/F, azniq/F



CIRCUMFIX X
PFX E Y 1
PFX E 0 t/X a
SFX F Y 1
SFX F 0 t/EX .



taxxamt, tamgart, tazniqt

PFX E Y 1 : **PFX** yemmal-d d akken alugen-a d azwir. **E** d isem n ulugen. **Y 1** d amđan n wudmawen n usiley

PFX E 0 t/X a : **PFX** yemmal-d d akken asiley-a s yessemras azwir. **E** d isem n ulugen, **0** yemmal-d d akken ulac asekkil ara nbeddel deg tazwara, **t** d asekkil ara nernu deg tazwara, **X** yemmal-d d akken alugen-a ad yettwasemres deg ulugen-nniđen, **a** yemmal-d d akken awal ara d-nsiley ilaq ad yebdu s usekkil “**a**”.

SFX F Y 1 : **SFX** yemmal-d d akken alugen-a d adfir. **F** d isem n ulugen. **Y 1** d amđan n wudmawen n usiley

SFX F 0 t/EX . : **SFX** yemmal-d d akken asiley-a yessemras adfir. **F** d isem n ulugen, **0** yemmal-d d akken ad nernu asekkil ney isekkil yer taggara n wawal, **t** d asekkil ara nernu yer taggara n wawal, **EX** yemmal-d d akken alugen **E (azwir)** ad yettwasemres akked ulugen **F(adfir)**, . (taneqqiđt) temmal-d d akken alugen-a ad yeddu i yal awal akken yebyu yella usekkil n taggara.

Alugen 2 :

axxam/H, amyar/H, azniq/H



PFX G Y 2
PFX G a i/X a
PFX G a yi/X a
SFX H Y 1
SFX H 0 en/GX .



**ixxamen,yixxamen, imyaren,
yimyaren, izniqen, yizniqen**

PFX G Y 2 : **PFX** yemmal-d d akken alugen-a d azwir. **G** d isem n ulugen. **Y 2** d amđan n wudmawen n usiley

PFX G a i/X a: **PFX** yemmal-d d akken asiley-a s yessemras azwir. **G** d isem n ulugen, **a** yemmal-d d akken ad nbeddel asekkil **a** deg tazwara, **i** d asekkil ara nernu deg tazwara, **X** yemmal-d d akken

alugen-a ad yettwasemres deg ulugen-nniđen, **a** yemmal-d d akken awal ara d-nsiley ilaq ad yebdu s usekkil “**a**”.

PFX G a yi/X a: **PFX** yemmal-d d akken asiley-a s yessemras azwir. **G** d isem n ulugen, **a** yemmal-d d akken ad nbeddel asekkil **a** deg tazwara, **yi** d isekkilen ara nernu deg tazwara, **X** yemmal-d d akken alugen-a ad yettwasemres deg ulugen-nniđen, **a** yemmal-d d akken awal ara d-nsiley ilaq ad yebdu s usekkil “**a**”.

SFX H Y 1: **SFX** yemmal-d d akken alugen-a d adfiir. **H** d isem n ulugen. **Y 1** d amđan n wudmawen n usiley

SFX H 0 en/GX . : **SFX** yemmal-d d akken asiley-a yessemras adfir. **H** d isem n ulugen, **0** yemmal-d d akken ad nernu asekkil ney isekkilen yer taggara n wawal, **en** d isekkilen ara nernu yer taggara n wawal, **GX** yemmal-d d akken alugen **G (azwir)** ad yettwasemres akked ulugen **H(adfir)**, . (taneqqidt) temmal-d d akken alugen-a ad yeddu i yal awal akken yebyu yella usekkil n taggara.

Alugen 3 :

axxam/J, amyar/J, azniq/J



PFX I Y 2
PFX I a te/X a
PFX I a ti/X a
SFX J Y 1.
SFX J 0 in/IX .



texxamin,tixxamin, temyarin,
timyarin, tezniqin, tizniqin

PFX I Y 2: **PFX** yemmal-d d akken alugen-a d azwir. **I** d isem n ulugen. **Y 2** d amđan n wudmawen n usiley

PFX I a te/X a: **PFX** yemmal-d d akken asiley-a s yessemras azwir. **I** d isem n ulugen, **a** yemmal-d d akken ad nbeddel asekkil **a** deg tazwara, **te** d isekkilen ara nernu deg tazwara, **X** yemmal-d d akken alugen-a ad yettwasemres deg ulugen-nniđen, **a** yemmal-d d akken awal ara d-nsiley ilaq ad yebdu s usekkil “**a**”.

PFX I a ti/X a: **PFX** yemmal-d d akken asiley-a s yessemras azwir. **I** d isem n ulugen, **a** yemmal-d d akken ad nbeddel asekkil **a** deg tazwara, **ti** d isekkilen ara nernu deg tazwara, **X** yemmal-d d akken alugen-a ad yettwasemres deg ulugen-nniđen, **a** yemmal-d d akken awal ara d-nsiley ilaq ad yebdu s usekkil “**a**”.

SFX J Y 1: **SFX** yemmal-d d akken alugen-a d adfiir. **J** d isem n ulugen. **Y 1** d amđan n wudmawen n usiley

SFX J 0 in/IX . : **SFX** yemmal-d d akken asiley-a yessemras adfir. **J** d isem n ulugen, **0** yemmal-d d akken ad nernu asekkil ney isekkilen yer taggara n wawal, **in** d isekkilen ara nernu yer taggara n wawal, **IX** yemmal-d d akken alugen **I (azwir)** ad yettwasemres akked ulugen **J(adfir)**, . (taneqqidt) temmal-d d akken alugen-a ad yeddu i yal awal akken yebyu yella usekkil n taggara.

Deg ufaylu n umawal awalen **axxam, amyar, ayyul** ḥafaren ilugan n usiley **F,H,J**. Imseyti ad d-yessiley awalen **taxxamt, tamwart, tazniqt, ixxamen, imwaren, izniqen, yixxamen, yimwaren, yizniqen,**

tixxamin, timyarin, tizniqin, texxamin, temyarin, tezniqin. Wigi d asiley n wunti, addad amariz n wunti, asget amalay, asget unti, addad amaruz n usget unti, addad amaruz n usget amalay n kar n yismawen.

Alugen 4:

awi/L, ini/L, argu/L,
ili/L, sentu/L, ntu/L



PFX K Y 3
PFX K 0 t/X [aieu]
PFX K 0 te/X [^auei][^auei]
PFX K 0 t/X [^auei][auei]
SFX L Y 3
SFX L 0 d/KX [iu]
SFX L 0 mt/KX [iu]
SFX L 0 m/KX [iu]



tawid, tiniq, targud, tilid, tsentud, tentud,
tawim, tinim, targum, tilim, tsentum,
tentum, tawimt, tinimt, targumt, tilimt,
tsentumt, tentumt

PFX K Y 3: PFX yemmal-d d akken alugen-a d azwir. K d isem n ulugen. Y 2 d amdan n wudmawen n usiley.

PFX K 0 t/X [aieu] : PFX yemmal-d d akken asiley-a yessemras azwir. K d isem n ulugen, 0 yemmal-d d akken ulac asekkil ara nbreddel deg tazwara, t d asekkil ara nernu deg tazwara, X yemmal-d d akken alugen-a ad yettwasemres deg ulugen-nniqen, [aieu] yemmal-d d akken awal ara d-nsiley ilaq ad yebdu s yiwen gar yisekkilen “aieu”.

PFX K 0 te/X [^auei][^auei] : yemmal-d d akken asiley-a yessemras azwir. K d isem n ulugen, 0 yemmal-d d akken ulac asekkil ara nbreddel deg tazwara, te d asekkil ara nernu deg tazwara, X yemmal-d d akken alugen-a ad yettwasemres deg ulugen-nniqen, [^auei][^auei] yemmal-d d akken awal ara d-nsiley ur ilaq ad yebdu s yiwen gar yisekkilen “aieu”, dayen asekkil wis sin ur ilaq ara ad yili d yiwen gar yisekkilen “aieu”.

PFX K 0 t/X [^auei][auei] : yemmal-d d akken asiley-a yessemras azwir. K d isem n ulugen, 0 yemmal-d d akken ulac asekkil ara nbreddel deg tazwara, t d asekkil ara nernu deg tazwara, X yemmal-d d akken alugen-a ad yettwasemres deg ulugen-nniqen, [^auei][auei] yemmal-d d akken awal ara d-nsiley ur ilaq ad yebdu s yiwen gar yisekkilen “aieu”, dayen asekkil wis sin ur ilaq ad yili d yiwen gar yisekkilen “aieu”.

SFX L Y 3 : SFX yemmal-d d akken alugen-a d adfir. L d isem n ulugen. Y 3 d amdan n wudmawen n usiley

SFX L 0 d/KX [iu] : SFX yemmal-d d akken asiley-a yessemras adfir. L d isem n ulugen, 0 yemmal-d d akken ad nernu asekkil ney iseikkilen yer taggara n wawal, d d asekkil ara nernu yer taggara n wawal, KX yemmal-d d akken alugen K (azwir) ad yettwasemres akked ulugen L(adfir), [iu] yemmal-d d akken alugen-a ad yettwasemres yef wawalen yettfakkan s yisekkilen [iu].

SFX L 0 mt/KX [iu] : SFX yemmal-d d akken asiley-a yessemras adfir. L d isem n ulugen, 0 yemmal-d d akken ad nernu asekkil ney iseikkilen yer taggara n wawal, mt d asekkil ara nernu yer taggara n wawal, KX yemmal-d d akken alugen K (azwir) ad yettwasemres akked ulugen L(adfir), [iu] yemmal-d d akken alugen-a ad yettwasemres yef wawalen yettfakkan s yisekkilen [iu].

SFX L 0 m/KX [iu]: SFX yemmal-d d akken asiley-a yessemras ađfir. **L** d isem n ulugen, **0** yemmal-d d akken ad nernu asekkil ney isekkilen yer taggara n wawal, **m** d asekkil ara nernu yer taggara n wawal, **KX** yemmal-d d akken alugen **K (azwir)** ad yettwasemres akked ulugen **L(ađfir)**, **[iu]** yemmal-d d akken alugen-a ad yettwasemres ȳef wawalen yettfakkan s yisekkilen **[iu]**.

Deg uфaylu n umawal, awalen **awi**, **ini**, **argu**, **ili**, **santu**, **ntu** тѣafaren alugen n usiley **L**. Imseyti ad d-yessiley awalen **tawid**, **tiniд**, **targud**, **tilid**, **tsentuд**, **tentuд**, **tawim**, **tinim**, **targum**, **tilim**, **tsentum**, **tentum**, **tawimt**, **tinimt**, **targumt**, **tilimt**, **tsentumt**, **tentumt**. Wigi d asisiley n kra n tlayiwin n yimyagen deg tmezri n wurmir aherfi.

Aḥric V Tiġertyert Web

Aħric V: Tiġertyert Web

Tazwert

Akken ad nsifses asemres n yilguritmen i d-nefka nnig-a, nhegga-d tiġertyert Web i wid iran ad sqedcen allal-a ama i uheggi n tegrummiwin n yisefka ney i tesleqt tasnalyant. Tiġertyert-a tebna yef titiknuluijiyn Web am **HTML, CSS, JavaScript**. Ahilen ttwarun s **Python** s usemres n **Postgres** deg **Linux Ubuntu**.

Aheggi n twennaqt n usemres

Asbeddi deg Linux Ubuntu 20.04 LTS

- 1- Sbedd Linux Ubuntu 20.04 LTS
- 2- Asbeddi n temkerdiyin Python: Deg uzirig n tladna (ligne de commande) n Ubuntu, aru:
 - a. sudo apt update
 - b. sudo apt install git python3-pip build-essential wget python3-dev python3-venv python3-wheel libfreetype6-dev libxml2-dev libzip-dev libldap2-dev libsasl2-dev python3-setuptools node-less libjpeg-dev zlib1g-dev libpq-dev libxslt1-dev libldap2-dev libtiff5-dev libjpeg8-dev libopenjp2-7-dev liblcms2-dev libwebp-dev libharfbuzz-dev libfribidi-dev libxcb1-dev
- 3- Timerna n useqdac n unagraw: Deg uzirig n tladna n Ubuntu, aru:
 - a. sudo useradd -m -d /opt/tigherghert -U -r -s /bin/bash aseqdac

Asbeddin Postgres

Deg uzirig n tladna n Ubuntu, aru:

- 1- sudo apt install postgresql
- 2- sudo su - postgres -c "createuser -s aseqdac"

Asbeddi n wkhtmltopdf

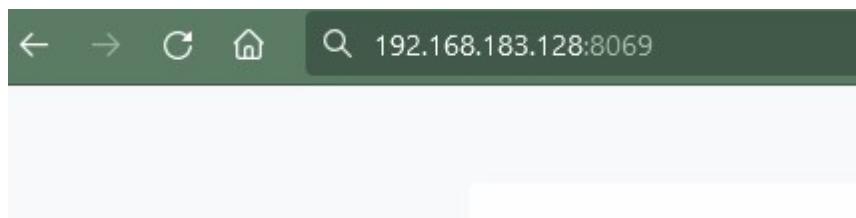
Deg uzirig n tladna n Ubuntu aru:

- 1- sudo wget https://github.com/wkhtmltopdf/wkhtmltopdf/releases/download/0.12.5/wkhtmltox_0.12.5-1.bionic_amd64.deb
- 2- sudo apt install ./wkhtmltox_0.12.5-1.bionic_amd64.deb

Tuqqna yer tyerġert:

Anekcum yer tyerġert web iteddu s usemres n yiminig Web ama d Firefox, Chrome, Edge ney wiyaq.

- 1- Sekcem tansa ney isem n taġult n tyerġert deg yiminig Web



2- Deg ugrudem, sekcem isem n useqdac d wawal uffir, syin akin tekki yef **Kcem**

Imayl

admin

Awal uffir

.....

Kcem

3- Agrudem yegber 5 n wumuyen

Tafyirt	Tafyirt yettwacerden
<input type="checkbox"/> Suref-iyi, wi k-ilan ?	Suref/VAI -iyi/PSV ./, wi/INT k-/PPV ilan/NMC ?/?
<input type="checkbox"/> Marie la tess Igazuz.	Marie/NMP la/NMP tess/VP Igazuz/NMC ./.
<input type="checkbox"/> Aql-iken wehd-nwen ?	Aql/VP -iken/PSV wehd/ADV -nwen/PADV ?/?
<input type="checkbox"/> Sami yella di Tahiti.	Sami/NMP yella/VP di/PRP Tahiti/NMC ./.
<input type="checkbox"/> Ruh si ger wallen-iw !	Ruh/VAI si/PRP ger/VP wallen/NMC -iw/PAN !!
<input type="checkbox"/> Uyalet yer deffir !	Uyalet/VAI yer/PRP deffir/NMC !!
<input type="checkbox"/> Yvaw ad nsüb sva	Yvaw/VAI ad/PAF nsüb/VAF sva/CS /

Umuyen n ugrudem n tyeryert

Umuy "Tasnalya taseddast"

Umuγ-a yegber inekcumen yer tegrumma n tefyar s tecraq ney tid ur nesei ara ticraq, akked d tineyrufin tisnalyanin tiseddasanin:

rrumt	Tasnalya/taseddast	Aseyti n tira	+
	Tifyar	Nnadi...	
	1 Tiyar ur nettwaqbel ara	▼ Imzize	
	2 Tifyar yettwaqeblen	en	
	Tagrumma n yisefka	/, wi/INT k-/PPV	
	3 Tagrumma n yisefka	tess/VP Igazuz/l	
	4 Tineyrufin		
?	Aql/VP -iken/PSV weħd/ADV -nwen/		
	Sami/NMP yella/VP di/PRP Tahiti/NI		
	Ruh/AI si/PRP ger/A/PR wawen/NMC		

Umuγ “Aseyti n tira”

Umuγ-a yegber anekcum yer wawalen, ilugan n usiley igejdanen, ilugan n usiley isnawanen, imawalen, talyiwin akked yigrawen n wawalen.

alya/Tajerrumt	Tasnalya/taseddast	Aseyti n tira	+
		<ul style="list-style-type: none"> 1 Awalen n umawal 2 llugan n usiley 3 Izirigen n yilugan n umawal 4 Amawal 5 Talyiwin 6 igraben 	
Anaw	D awṣil n uddes?		
Azwir	Uhu		
Azwir	Uhu		
Azwir	Ih		feminine beg
Adfir	Uhu		singular feme

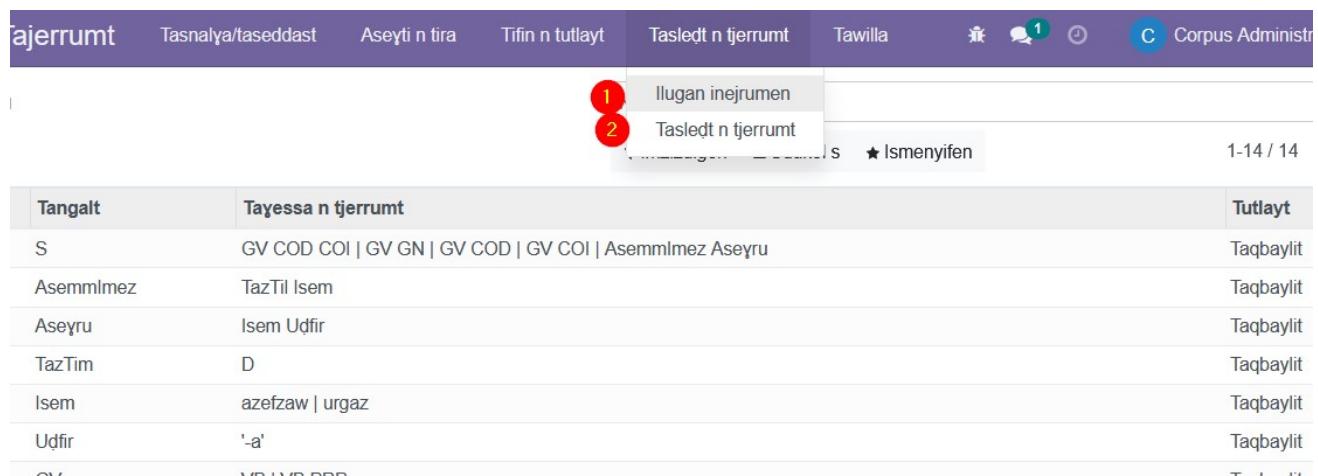
Umu “Tifin n tutlayt”

Umu-a yegber anekcum yer tneyrufin n tutlayt d tegrummiwin n tefyar yettwasemrasen i waya

Tasnalya/Tajerrumt	Tasnalya/taseddast	Aseyti n tira	Tifin n tutlayt	Tasleđt n tjerrumt	Tawilla
grumma n tefyar			<ul style="list-style-type: none"> 1 Tifin n tutlayt 2 Tagrumma n tefyar 		
				Jigen	Sdukel s
Isem n tegrumma	Amđan n tefyar	Teyzi tafellat n tefyirt...	Teyzi taddayt n tefyirt	Tutlayin	
Nine languages	1000	25	20	Taqbaylit Tafransist Taglizit Tapurtugit	
Kabyle	100000	10000	0	Taqbaylit	
Berber	600000	4000	0	Berber	
Tacehlhit	12000	500	0	Tacehlhit	
Finois	7000	40	0	Tafinit	
~	~	~	~	~	~

Umu “Tasleđt n tjerrumt”

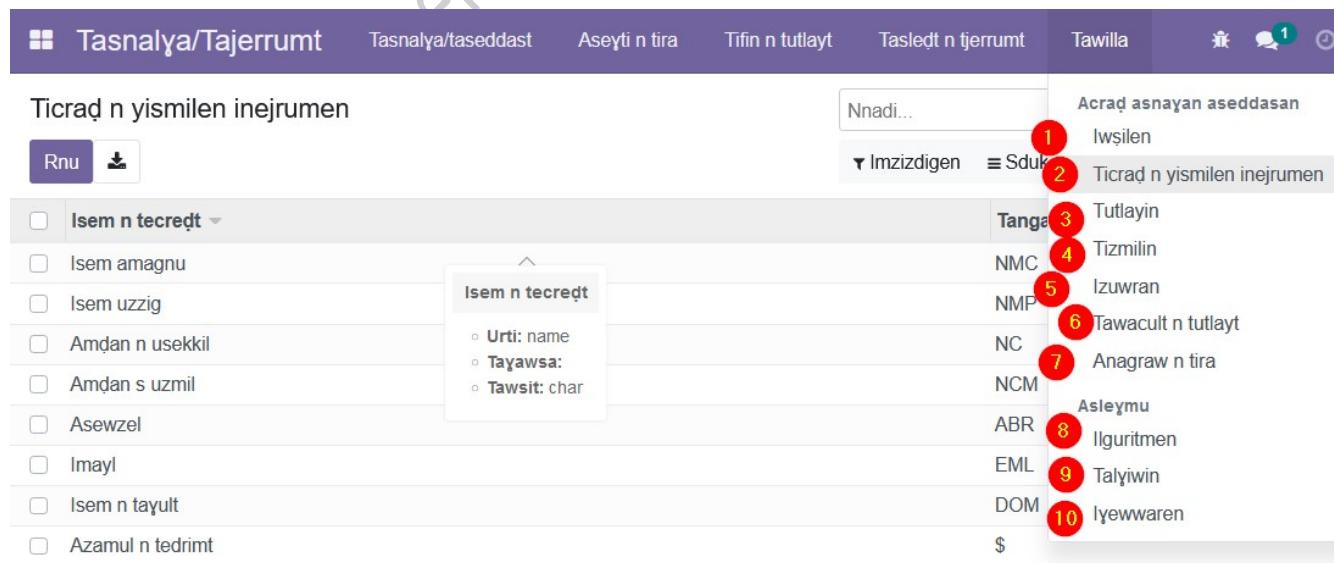
Umuy-a yegber anekcum yer yilugan inejrumen akked teslesleqt tanejrumt.



Tangalt	Tayessa n tjerrumt	Tutlayt
S	GV COD COI GV GN GV COD GV COI Asemmlmez Aseyru	Taqbaylit
Asemmlmez	TazTil Isem	Taqbaylit
Aseyru	Isem Udfir	Taqbaylit
TazTim	D	Taqbaylit
Isem	azefzaw urgaz	Taqbaylit
Udfir	'-a'	Taqbaylit

Umuy tawila:

Umuy-a yegber anekcum yer tmerna n yiwsilen, ticrad n yismilen inejrumen, tutlayin, tizmilin, tiwaculin n tultayin, izuran n wawalen, inagrawen n tira, ilguritmen, iżewwaren n yilguritmen, akked talyiwin n usleymu n ylgurtimen.



Nnadi...	Tangalt	Tutlayt
1	Iwşilen	Ticrad asnayan aseddasan
2	Ticrad n yismilen inejrumen	Iwşilen
3	Tutlayin	Tutlayin
4	Tizmilin	Tizmilin
5	Izuwran	Izuwran
6	Tawacult n tutlayt	Tawacult n tutlayt
7	Anagraw n tira	Anagraw n tira
8	Asleymu	Asleymu
9	Ilguritmen	Ilguritmen
10	Talyiwin	Talyiwin
\$	Iżewwaren	Iżewwaren

Amek ara nernu tafyirt?

The screenshot shows a software interface for morphological analysis. At the top, there are tabs: "Tasnalya/Tajerrumt" (highlighted with a red circle), "Tasnalya/taseddast" (highlighted with a red circle), "Aseyti n tira", and "Tifin n t". Below the tabs, the word "Tafyirt" is selected. On the left, a list of words is shown with checkboxes:

- Tafyirt (highlighted with a red circle)
- Marie la tess Igazuz.
- Aql-iken weħd-nwen ?
- Sami yella di Tahiti.
- Ruh si ger wallen-iw !
- Uyalet yer deffir !

To the right, a detailed analysis is provided for each word:

Analysis	Meaning
Tifyar	
2 Tiyar ur nettwaqbel ara	Tiyar ur nettwaqbel ara
Tifyar yettwaqeblen	Tifyar yettwaqeblen
Tagrumma n yisefka	Tagrumma n yisefka
Tineyrufin	Tineyrufin
yettwacerden	yettwacerden
NMP la/NMP tess/VP	NMP la/NMP tess/VP
-iken/PSV weħd/ADJ	-iken/PSV weħd/ADJ
Sami/NMP yella/VP di/PRP	Sami/NMP yella/VP di/PRP
Ruh/VAI si/PRP ger/VP wall	Ruh/VAI si/PRP ger/VP wall
Uyalet/VAI yer/PRP deffir/NI	Uyalet/VAI yer/PRP deffir/NI

Akken nwala deg tugna nnig-a :

- 1- Ddu yer wumuq “Tasnalya/Taseddast”
- 2- Tekki yef “Tufyar ur nettwaqbel ara”
- 3- Tekki yef tqeffalt “Rnu”

Ticki tsended yef tqeffalt “Rnu”, asfaylu ddaw-a ad d-iban akken ad ternuð tafyirt:

The screenshot shows a software interface for text entry and processing. At the top, there are buttons for "Kles" and "Sefsex". Below them, a navigation bar includes buttons for "Bdu", "Sal-d tayunin s tecrad", "Qbel", "Af-d tutlay n tefyirt", "Cred tafyirt-a", "Arewway", "Teyti", "Tettwacred", and "Tettwaqbel".

On the left, there are input fields for "Tafyirt" (highlighted with a red circle), "Tafyirt yettwacerden", "Amdan", and "Tutlayt". The "Tafyirt" field contains the text: "Ass-a, d ass amenu seg Yennayer." The "Tafyirt yettwacerden" field contains an empty line. The "Amdan" field contains the number "0". The "Tutlayt" field contains an empty dropdown menu.

Below these fields is a button labeled "Tayunin tisnalyanin". At the bottom, there are buttons for "Asmizz...", "Tayunt tasnalyant", "Tacrejt", "Ażar", and "Tizmilin". There is also a link "Add a line".

- 1- Aru tafyirt-ik sakın tekki yef tqeffalt “Sekles”

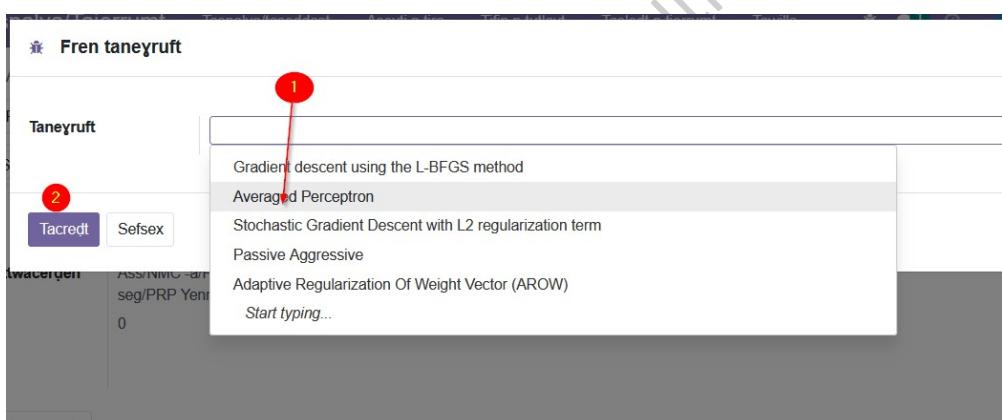
Amek ara ternud ticrad tisnalyanin tiseddasanin i tefyirt?

Tafyirt / Ass-a, d ass amenzu seg Yennayer.

Zreg	Rnu	Tigawt		
Bdu	Sali-d tayunin s tecrad	Qbel	Af-d tutlayt n tefyirt	Cred tafyirt-a
				1
Tafyirt	Ass-a, d ass amenzu seg Yennayer.			
Tafyirt yettwacerden	Ass/NMC -a/PAN /, d/PREAL ass/NMC amenzu/NMC seg/PRP Yennayer/NMP /.			
Amdan	0			
Tutlayt				

Tayunin tisnalyanin

- 1- Mi teskelseq tafyirt, ad d-tban tqeffalt “**Cred tafyirt-a**”, tekki fell-as. Asfaylu- ddaw-a ad d-iban. Akken ad tfernen algoritm i tebyid ad tesqedced.



- 1- Fren algoritm sakin tekki yef Tqeffalt “Tacreqt”

Dagi aselkim ad yefk tacreqt i yal tayunt tasnalyant i yellan deg tefyirt.

Deg tefyirt-a, aselkim yefka-d 9 n tayunin:

Tafyirt	Ass-a, d ass amenzu seg Yennayer.								
Tafyirt yettwacerden	Ass/NMC -a/PAN /, d/PREAL ass/NMC amenzu/NMC seg/PRP Yennayer/NMP /.								
Amdan	0								

- 1- Ass/NMC: “Ass” d isem amagnu
2- -a/PAN: “-a” d amqim awsil n yisem.
3- ,/: “,” d azamul n usenqed.

- 4- d/PREAL: “d” d tazelya n tilawt.
- 5- Ass/NOM: “ass” d isem amagnu.
- 6- Amenzu/NMC: “amenzu” d isem amagnu.
- 7- Seg/PRP: “seg” d tanzeyt.
- 8- Yennayer/NMP: “Yennayer” d isem uzzig.
- 9- ./.，“.” D azamul n usenqed.

Aḥric VI Taggarā

Aħric n taggara:

Awal n taggara

Tifin tawurmant n tutlayt akked ucrad asnalyan aseddasan d isemsal imezwura ara yeldin abrid i wayen-nniđen. Ur nettizmir ara ad d-nnadi isuddimen ney awal aħerfi/ażar n yiżiżen ma yella ur neżri ara awal-a n wanta tutlayt ney dayen ma d isem ney ala. Ur nettizmir ara ad neg tasleqt n tjerrumt mebla ma nga tasleqt n talya aked tseddast.

Titiknulujiyn timaynutin ass-a ur llint ara d żżerb i usnefli n tutlayin. Ric Tutulayin timaziżin zemrent ad farşent tagħnit seg waya akken ad keċment s telqay deg umādil umdin.

Deg tezrawt-a, nesken-d d akken aselmed awurman izmer ad yettwasemres deg tesleqt tasnalyant taseddasant. Maca maċči daya. Yal aswir n tesnilest, akken yesea ilugan-is isnilsanen, yesea dayen ifecka d tarrayin i as-iwulmen akken yeddu.

Ugur meqqren deg yiħriċen-a i icudden yer usemsel awurman n tutlayin, d lqella n yisefka isnilsanen, ama s talya n tira ney s talya n tayect. Igħaqda i d-yettmuddu uselkim yeldeñ tikwal, acku ađris n ulmad wezzil. Awal “wezzil” ɣur-s sin n yinumak: Ulac ddeqs n tefyar ney ulac ddeqs n talyiwin n tefyar. Issefk ađris n ulmad ad yegħber aħas n tefyar yemgaraden deg tseddast. Ma nseħħas aħas n tefyar n kra kan n talyiwin tiseddasanin , aselkim ur ilemmmed ara akken iwata.. Asemres n tigzi n tmacint deg tayult n tesnilest, dayen ara isseftin taqbaylit deg yinurar n tudert n umdan n yal ass am: Tsuqqilt, aeqal n tayect, anadi n wawalen, anadi n yiżri, asismel n yisefka ...atg.

Xas wamma deg tezrawt-a nessisen-d kan tarrayt n ucrad asnalyan aseddasan s usemres n ulmad lqayen n Keras, scikit-learn, akked tifin tawurmant n tutlayt maca ddeqs n tarrayin d yifecka-nniđen i yellan am Azure, Pytorch, d wiċċa zemren ad d-glun s tifrat i wuguren-nniđen n tesnisalest tasenselkamt.

Ddeqs n yiħriċen-nniđen n usemsel awurman n tutlayt i izemren ad ldin abrid i teqbaylit akken ad tressi iman-is deg umādil umdin, gar-asen: Anadi n wawalen iħerfiyen, asuddem d tseftit, asuddem, aeqal n tyawsilin yettusemman, aeqal n tayect, tasleqt n yinaw, ...atg.

Ilguritmen i d-nedfka deg tezrawt-a mmden i usleymu akked usemres n tnejrufin ara d-ssuġien. Yeqqim-d kan ad d-nheġgi tigrummiwin n yisefka isnilsanen ara nerr gar yiffasen n yimussnawen akken ad ikemmell unadi deg wuħriċ-a.

Akken ad tnerni tseddi n tnejrufin i itezzin yef tesnilest, ila qad tgħid tegrummiwin n yisefka isnilsanen deg yal tayult deg tesnilest imi tussna ass-a tebna yef yisefka.

Timsisyal

- Speech and language processing: An introduction to natural language processing, computational linguistics, and speech recognition – Daniel Jurafsky & James.
- Isuraz n usezdi d tenmezla taqrirant n tmaziyt: asnekwu d tesleqt – Zahir Meksem.
- Tajerrumt tatrart n teqbaylit – Kamal Nath Zerrad.
- Ilugan n tira n tmaziyt – K. Bouamara, B. Hamek, M. A. Amrouche, z. Meksem, A. Rabhi, M. Tidjet.
- Memento grammatical et orthographique du berbère, kabyle, chleuh et rifain – Kamal Nait Zerrad.
- Tajerrumt n tmaziyt tamirant, talyiwin – Kamal Nait Zerrad.
- Linguistique berbère, études de syntaxe et de diachronie – Salem Chaker.
- Un parler berbère d'Algérie (Kabylie), Syntaxe – Salem Chaker.
- Asegzawal Issin – Kamal Bouamara.
- Manuel de conjugaison kabyle – Kamal Nait Zerrad.
- Foundation of statistical Natural Language Processing – Christopher D. Manning & Hinrich Schütze.
- Machine Learning Mastery ith Python, undesrtand your data, create accurate models and work projects en-to-end – Jason Brownlee.
- Machine learning yearning, technical strategy for AI engineers, in the era of deep learning – Andrew NG.
- Fundamentals of deep learning – Nikhil Buduma & Nicholas Locascio
- Deep Learning with Python – Nikhil Ketkar.
- Machine learning and deep learning with Python, Scikit-learn and Tensorflow – Sebastian Raschka & Vahid Mirjalili.
- Learning Tensorflow
- Lexique de Mathématique, revue Tafsut, 1986 – Ramdane Achab, Hend Sadi, Layhem.
- Amawal n tmaziyt tatrart, 1976, Mouloud Maméri.
- Lexique d'informatique, 1998 – Saad Bouzefrane.
- Dictionnaire des sciences du language, 2005 – Mohand Mahrazi.
- Dictionnaire de langue Kabyle: Jean Marie Dallet.
- Tajerrumt n tmaziyt – tantala taqbaylit – Mouloud Maméri.
- Automatic Tagging of Arabic Text: From Raw Text to Base Phrase Chunks - Mona Diab, Kadri Hacioglu, Daniel Jurafsky - Stanford University, University of Colorado, Boulder -
- TensorFlow for Machine Intelligence, A Hands-On Introduction to Learning Algorithms - Sam Abrahams, Danijar Hafner, Erik Erwitt, Ariel Scarpinelli – Bleeding Edge Press -2016.
- Awfus n tutlayt tamaziyt – Boussad Kebir – HCA
- Aslugen n tira n tmaziyt – Actes- HCA – Décembre 2010.
- Tira n tmaziyt – Ramdane Achab -

Timsirin yef ulmad awurman

- Almad awurman s Pytorch: <https://classroom.udacity.com/courses/ud188>
- Almad awurman s tensorflow: <https://developers.google.com/machine-learning/crash-course/ml-intro?hl=fr>
- Almad awurman s Octave: <https://www.coursera.org/learn/machine-learning/>

Ifecka & tutlayin n usihel

Ahilen iyettwasqedcen deg tezrawt-a ttwasneflan s yifecka-a:

- python v 3.6 - <https://www.python.org/>
- keras v 2.2.4 - <https://keras.io/>
- tensorflow v 1.5.0 - <https://www.tensorflow.org/>
- matPlotLib v 3.3.3 - <https://matplotlib.org/>
- sklearn V 0.19.0 - <https://scikit-learn.org/>
- numpy v 1.19.4 - <https://numpy.org/>

Amawal

Tamaziyt – Tafransist

Tamaziyt	Tafransist
Abnubek	Ambiguité
Abnubek amsislan/asnislan	Ambiguité phonétique/phonologique
Abnubek asnalyan aseddasan	Ambigüité morphosyntaxique
Acraç asnalyan aseddasan	Etiquetage/Marquage morphosyntaxique
Adris n ulmad	Texte d'apprentissage
Aeqal n tayect	Reconnaissance vocale
Afecku/Ifecka	Outils
Afmiđi	Taux
Agbur umđin	Contenu numérique
Ayewwar/iyewwaren	Paramètres
Ahil asenselkam	Programme informatique
Aktazel	Evaluation/mesure
Alguritm	Algorithme
Alguritm n tifin	Algorithme de détection
Alguritm n ulmad	Algorithme d'apprentissage
Almad awurman	Apprentissage automatique
Almad Iqayen	Apprentissage profond
Amađal umđin	Monde numérique
Amsenselkam	Informaticien
Anadi n uzař n wawal	Racinisation
Anadi n wawalen iherfiyen	Lemmatisation
Anagraw n tira	Système d'écriture
Arrat	Document
Asefku (isefka)	Donnée(s)
Asegzel n yiđrisen	Résumé de textes
Asekkil ilem	Caractère blanc
Asekkil n trigla	Caractère de tabulation
Asemsel	Traitemet (données)
Asemsel awurman n tutlayt	Traitemet automatique des langues
Asenselkam	informatique (Adjectif)
Asihel asenselkam	Programmaton informatique
Asismel	Classification
Asiwel n temkađiyin	Appel de bibliothèques
Askan n yisefka	Visualisation des données
Asleymu	Entraînément
Asleymu n yilguritmen	Entrainement des algorithmes
Asmil anejrium	classe grammaticale
Asuddem	Dérivation

Azetta n yinurunen	Réseau de neurones
Betṭu asnalyan aseddasan	Segmentation morphosyntaxique
Betṭu n tunṭiqin	Syllabation
Gezzu	Compréhension
Imseyti n tira	Correcteur orthographique
Imsismel	Classificateur
Inaw	Discours
Isefka	Données
Snefli	Développer
Taffa n yisefka	Base de données
Tagrumma	Ensemble
Tagrumma n yisefka n usekyed	Ensemble de données de test
Tagrumma n yisefka n usentem	Ensemble de données de confirmation/validation
Tayult	Domaine
Talqiwın tiseddasanin	Formes syntaxiques
Tamsisyelt/Timsisyal	Référence(s)
Tamyigawt	Interaction
Taneflit	Développement
Taneyruft	Modèle
Tasenselkamt	Informatique (Nom)
Tasenselkamt tasnilsant	Informatique linguistique
Tasleđt n yisefka	Analyse des données
Tasleđt tamawalant	Analyse lexicale
Tasleđt tasnalylan	Analyse morphologique
Tasnilest tasenselkamt	Linguistique informatique
Tawuri n urmad	Fonction d'activation (réseau de neurones)
Tifin tawurmant n tutlayt	Détection automatique de la langue
Tiyawsiwin yettwasemman	Objets/Entités nommées
Tigzi	Intelligence
Tigzi n tmacint	Intelligence artificielle (intelligence de la machine)
Tiseddi	Précision
Tissi n tuffya	Couche de sortie (réseaux de neurones)
Tissi n unekcum	Couche d'entrées (réseau de neurones)
Tissi yeffren	Couche cachée (réseaux de neuronne)
Tizdit	Trait d'union
Tukksa n ubnubek	Désambigüisation
Tusnakt	Mathématiques (Nom)
Tussna n yisefka	Science de données
Tutlayin n usihel	Langages de programmation

Tafransist	Tamaziyt
Ambiguité	Abnubek
Ambiguité phonétique/phonologique	Abnubek amsislan/asnislan
Ambigüité morphosyntaxique	Abnubek asnalyan aseddasan
Etiquetage/Marquage morphosyntaxique	Acrad asnalyan aseddasan
Texte d'apprentissage	Ađris n ulmad
Reconnaissance vocale	Aeqal n tayect
Outils	Afecku/Ifecka
Taux	Afmiđi
Contenu numérique	Agbur umđin
Paramètres	Ayewwar/iyewwaren
Programme informatique	Ahil asenselkam
Evaluation/mesure	Aktazel
Algorithme	Alguritm
Algorithme de détection	Alguritm n tifin
Algorithme d'apprentissage	Alguritm n ulmad
Apprentissage automatique	Almad awurman
Apprentissage profond	Almad Iqayen
Monde numérique	Amađal umđin
Informaticien	Amsenselkam
Racinalisation	Anadi n üzər n wawal
Lemmatisation	Anadi n wawalen iherfiyen
Système d'écriture	Anagraw n tira
Document	Arrat
Donnée(s)	Asefku (isefka)
Résumé de textes	Asegzel n yiđrisen
Caractère blanc	Asekkil ilem
Caractère de tabulation	Asekkil n trigla
Traitemet (données)	Asemsel
Traitemet automatique des langues	Asemsel awurman n tutlayt
informatique (Adjectif)	Asenselkam
Programmaton informatique	Asihel asenselkam
Classification	Asismel
Appel de bibliothèques	Asiwel n temkađiyin
Visualisation des données	Askan n yisefka
Entraînement	Asleymu
Entrainement des algorithmes	Asleymu n yilguritmen
classe grammaticale	Asmil anejrium
Dérivation	Asuddem
Réseau de neurones	Azetta n yinurunen
Segmentation morphosyntaxique	Bet̄u asnalyan aseddasan
Syllabation	Bet̄u n tunqiñin
Compréhension	Gezzu

Correcteur orthographique	Imseyti n tira
Classificateur	Imsismel
Discours	Inaw
Données	Isefka
Développer	Snefli
Base de données	Taffa n yisefka
Ensemble	Tagrumma
Ensemble de données de test	Tagrumma n yisefka n usekyed
Ensemble de données de confirmation/validation	Tagrumma n yisefka n usentem
Domaine	Tayult
Formes syntaxiques	Talyiwin tiseddasanin
Référence(s)	Tamsisyelt/Timsisyal
Intéraction	Tamyigawt
Développement	Taneflit
Modèle	Taneyruft
Informatique (Nom)	Tasenselkимт
Informatique linguistique	Tasenselkимт tasnilsant
Analyse des données	Tasleđt n yisefka
Analyse lexicale	Tasleđt tamawalant
Analyse morphologique	Tasleđt tasnalyant
Linguistique informatique	Tasnilest tasenselkamt
Fonction d'activation (réseau de neurones)	Tawuri n urmad
Détection automatique de la langue	Tifin tawurmant n tutlayt
Objets/Entités nommées	Tiyawsiwın yettwasemman
Intelligence	Tigzi
Intelligence artificielle (intelligence de la machine)	Tigzi n tmacint
Précision	Tiseddi
Couche de sortie (réseaux de neurones)	Tissi n tuffya
Couche d'entrées (réseau de neurones)	Tissi n unekcum
Couche cachée (réseaux de neuronne)	Tissi yeffren
Trait d'union	Tizdit
Désambigüisation	Tukksa n ubnubek
Mathématiques (Nom)	Tusnakt
Science de données	Tussna n yisefka
Langages de programmation	Tutlayin n usihel

Tagrumma n tecrađ tisnalyanin tiseddasanin

Tafelwit ddaw-a tessegzay-d s wugar n telqayt tagrumma n tecrađ tisnalyanin teseddasanin i nsemres deg tezrawt-a.

Asmil anejrum	Tafransist	Tacredt	Amedya
Isem amagnu	Nom commun	NMC	ayrum
Isem uzzig	Nom propre	NMP	Bgayet
Amđan s yisekkilen	Cardinal lettre	NC	Yiwen
Amđan s yizwilen	Cardinal Chiffre	NCM	0392
Asewzel/Axefwal	Accronyme/Abbréviation	ABR	atg, AMLS
Tansa imayl	Email	EML	said77@gmail.com
Isem n tayult	Nom de domaine	DOM	imsidag.com
Azamul n tedrimt	Symbole monétaire	\$	USD, DZD, EURO...
Urmir anađ	aoriste impératif	VAI	Ečč
Urmir imal	aoriste futur	VAF	ad ččey
Izri	Prétérit	VP	Yečča
Izri ibaw	prétérit négatif	VPN	ur yečči ara
Amayun n wurmir	participe de l'aoriste	VPA	ara yeččen
Amayun n yizri ilbaw	participe du prétérit positif	VPPP	win yeččan
Amayun n yizri ibaw	participe du prétérit négatif	VPPN	Win ur nečči
Anađ ussid	impératif intensif	VII	Tett
Urmir ussid	aoriste intensif	VAI	tettey
Amayun n wurmir ussid ilaw	participe de l'aoriste intensif positif	VPAIP	win itetten
Amayun n wurmir ussid ibaw	participe de l'aoriste intensif négatif	VPAIN	win ur ntett ara
	Subjonctif	VSBJ	yas-d ma yeba.
Irbiben s umserbib	Morphème adjectivant	MADJ	Sut, mm, bu, tis, wis, at, wat, yir,
Arbib	Adjectif	ADJ	axxam azeggay, d amellal, argaz ayezfan
Amernu	Adverbe	ADV	iđelli, azekka
Amassay	Pronom relatif	PRL	i - d netta i d-yusan
Tazelya n tnila tuzwirt	particule de direction prefixe	PDP	d-, id-, n-, in-

Tazelya n tnila tuđfirt	particule de direction suffixe	PDS	-d,-id,-n, -in
Tazelya n tibawt	particule de négation	PRN	ur, ara
Tazelya n tilawt	particule de réalité	PREAL	d izem
Tasyunt n tuqqna	conjonction de coordination	CC	d, akked, ney, ihi, maca, meña
Tasyunt n usentel	conjonction de subordination	CS	ma, imi, amer
Tazelya n wormir aherfi	particule de l'aoriste futur	PAF	ad čcey, ara čcey,
Tazelya n wormir ussid	particule de l'aoriste intensif	PAI	la yessawal
Tazelya n usiwel	Particule d'appel	PLP	Ay argaz! a tameṭṭut
Tazelya n umayun n wormir	Particule du participe de l'aoriste	PPAF	ara iruhen, ara yettruḥun
Tanzeyt	Préposition	PRP	seg, i, yer, ar
Imqimen udmawanen ilelliyan	pronoms personnels libres	PRA	nekk, nekkini
Imqimen n wayla iwšilen n yisem	Pronoms possessifs affixes de noms	PPAN	axxam-iw, axxam-nwen
Imqimen imeskanen Iwšilen n yisem d umernu	Pronoms démonstratifs affixes de noms ou d'adverbe	PDAN	axxam-ihin, axxam-agı
Imqimen irbuda iwsilen n yisem	Pronoms indéfinis affixes de noms ou d'adverbe	PIAN	axxam-nniđen, xxam-nni
Imqimen uzwiren usriden n umyag	Pronoms préfixe de verbes	PPV	ad ak-fkey
Imqimen uđfiren usriden n umyag	Pronoms suffixes de verbes	PSV	fkiy-ak
Imqimen iwsilen n tenzeyt	Pronoms affixes de prépositions	PAP	seg-ney
Imqimen n wayla	Pronoms possessifs autonome	PPA	yinu, yinek, nnu, nnek, nwen
Imqimen irbuda	Pronoms indéfinis	PRI	wiyađ, tiyad, win, tid
Imqimen imeskanen	Pronoms démonstratifs	PRD	wagi, tagi, winna, tinna, widak, tidak, wihin, tihin, wihinna, thinna, wihid, tihid
Imqimen iwsilen n yimerna	Pronom affixes d'adverbe	PAA	idelli-nni
Imqimen imattaren	pronoms interrogatifs	INT	Anwa? Anta?

			Melmi?
Imerna imattaren	adverbes interrogatif	ADINT	yezra d anta-tt.
Tazela n ubhat	Interjection	INJ	Ah, abbuḥ, ahbuḥ, Ah, Uk, uf, Yah, Ayuh, ayhu,
Agaz	Point	.	
Tafryat	Virgule	,	
Agaz d tefrayt	Point virgule	;	
Sin wagazen	Deux points	:	
Agaz n usesten,n tuttra	Point d'interrogation	?	
Agaz n ubhat	Point d'exclamation	!	
	Simple guillemet/apostrophe	'	
Tuccar	Double guillemets	"	
Ajerrid (tizdit)	Trait d'union	-	
Timerna	Plus	+	
Itri	Etoile	*	
Iceew imedlen	Parenthèse fermante)	
Iceew yeldin	Parenthèse ouvrante	(
Akruci imedlen	Crochet fermant]	
Akruci yeldin	Crochet ouvrant	[
	Accolade ouvrante	{	
	Accolade fermante	}	
Aækaz yeknan s ayeffus	Slash	/	
Aækaz yeknan s azelmađ	Anti slash	\	
Ajerrid n uderrer	Souligné	—	
Tafesna	Degrès	°	
Uzmir sin	Puissance carrée	²	
Uzmir krad	Puissance cube	³	
Aækaz	Barre verticale		
Akafu	Chapeau	^	
Agazen n wagal	Points de suspension	...	