

ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ

Speech to text Technologies for People with Deafness and hearing Loss

Ησαΐας Κωνσταντίνος

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

ΥΠΕΥΘΥΝΟΣ

Κωνσταντίνος Κολομβάτσος

Λαμία Φεβρουάριος έτος 2025



SPEECH TO TEXT TECHNOLOGIES FOR PEOPLE WITH DEAFNESS AND HEARING LOSS

ΗΣΑΙΑΣ ΚΩΝΣΤΑΝΤΊΝΟΣ

ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

ΥΠΕΥΘΥΝΟΣ

Κωνσταντίνος Κολομβάτσος

Λαμία Φεβρουάριος έτος 2025



SCHOOL OF SCIENCE

DEPARTMENT OF COMPUTER SCIENCE & TELECOMMUNICATIONS

Speech to text Technologies for People with Deafness and hearing Loss

Isaias Konstantinos

FINAL THESIS

ADVISOR

Konstantinos Kolomvatsos

Lamia February year 2025

|  |
| --- |
| *«Με ατομική μου ευθύνη και γνωρίζοντας τις κυρώσεις (1), που προβλέπονται από της διατάξεις της παρ. 6 του άρθρου 22 του Ν. 1599/1986, δηλώνω ότι:* |
| *1.    Δεν παραθέτω κομμάτια βιβλίων ή άρθρων ή εργασιών άλλων αυτολεξεί****χωρίς να τα περικλείω σε εισαγωγικά****και χωρίς να αναφέρω το συγγραφέα, τη χρονολογία, τη σελίδα. Η αυτολεξεί παράθεση χωρίς εισαγωγικά χωρίς αναφορά στην πηγή, είναι λογοκλοπή. Πέραν της αυτολεξεί παράθεσης, λογοκλοπή θεωρείται και η παράφραση εδαφίων από έργα άλλων, συμπεριλαμβανομένων και έργων συμφοιτητών μου, καθώς και η παράθεση στοιχείων που άλλοι συνέλεξαν ή επεξεργάσθηκαν, χωρίς αναφορά στην πηγή. Αναφέρω πάντοτε με πληρότητα την πηγή κάτω από τον πίνακα ή σχέδιο, όπως στα παραθέματα.*  *2.    Δέχομαι ότι η αυτολεξεί****παράθεση χωρίς εισαγωγικά****, ακόμα κι αν συνοδεύεται από αναφορά στην πηγή σε κάποιο άλλο σημείο του κειμένου ή στο τέλος του, είναι αντιγραφή. Η αναφορά στην πηγή στο τέλος π.χ. μιας παραγράφου ή μιας σελίδας, δεν δικαιολογεί συρραφή εδαφίων έργου άλλου συγγραφέα, έστω και παραφρασμένων, και παρουσίασή τους ως δική μου εργασία.*  *3.    Δέχομαι ότι υπάρχει επίσης περιορισμός στο μέγεθος και στη συχνότητα των παραθεμάτων που μπορώ να εντάξω στην εργασία μου εντός εισαγωγικών. Κάθε μεγάλο παράθεμα (π.χ. σε πίνακα ή πλαίσιο, κλπ), προϋποθέτει ειδικές ρυθμίσεις, και όταν δημοσιεύεται προϋποθέτει την άδεια του συγγραφέα ή του εκδότη. Το ίδιο και οι πίνακες και τα σχέδια*  *4. Δέχομαι όλες τις συνέπειες σε περίπτωση λογοκλοπής ή αντιγραφής.* |

Ημερομηνία: ……/..…/20……

Ο – Η Δηλ.

*(1)*   *«Όποιος εν γνώσει του δηλώνει ψευδή γεγονότα ή αρνείται ή αποκρύπτει τα αληθινά με έγγραφη υπεύθυνη δήλωση*

*του άρθρου 8 παρ. 4 Ν. 1599/1986 τιμωρείται με φυλάκιση τουλάχιστον τριών μηνών. Εάν ο υπαίτιος αυτών των πράξεων*

*σκόπευε να προσπορίσει στον εαυτόν του ή σε άλλον περιουσιακό όφελος βλάπτοντας τρίτον ή σκόπευε να βλάψει άλλον, τιμωρείται με κάθειρξη μέχρι 10 ετών.»*

### ΠΕΡΙΛΗΨΗ

Η συγκεκριμένη πτυχιακή εργασία αφορά την ανάπτυξη ενός προγράμματος για την “μετάφραση” κειμένου από τον χρήστη είτε από την αναγνώριση της φωνής του είτε ( για μεγαλύτερη ευκολία και ακρίβεια) από εισαγωγή μέσο πληκτρολογίου. Ο συγκεκριμένος μεταφραστής θα έχει την δυνατότητα να αντιστοιχεί κάθε γράμμα του χρήστη με ένα συγκεκριμένο σύμβολο στην νοηματική γλώσσα. Ο μεταφραστής έχει την δυνατότητα αναγνώρισης φωνής στα Αγγλικά και Ελληνικά. Η συγκεκριμένη εφαρμογή εκτός από την αναγνώριση και αντιστοίχιση γραμμάτων με σύμβολα νοηματικής, έχει και την δυνατότητα(Μόνο στην αγγλική γλώσσα) να απλοποιεί ως ένα βαθμό και το κείμενο που δίνει ο χρήστης σαν εισαγωγή όταν μιλάει. Το συγκεκριμένο χαρακτηριστικό της εφαρμογής υλοποιείται μέσο μοντέλου μηχανικής μάθησης το οποίο έχει “προπονηθεί” σε ‘ένα μεγάλο σύνολο δεδομένων με απλές προτάσεις και το απλοποιημένο περιεχόμενό τους. Συνοψίζοντας όλα τα παραπάνω υλοποιούνται από την χρήση τεχνολογιών όπως αναγνώριση φωνής, chatbot(μόνο για εκφώνηση έτοιμων διαλόγων) , βιβλιοθήκη επεξεργασίας και προβολής εικόνων, βιβλιοθήκη για την καταγραφή ημερομηνιών ( χρησιμοποιείται στο csv για την σωστή και οργανωμένη καταγραφή δεδομένων), βιβλιοθήκη για την συμπερίληψη ANN(Advanced Neural Networks) όπως το t5-small και BART για NLP ( Natural Language Processing ) που ευθύνονται για την απλοποίηση κειμένου στην εφαρμογή μας.

### ABSTRACT

This thesis project involves the development of a program for “translating” text from the user either through voice recognition or (for greater ease and accuracy) through keyboard input. This translator will have the capability to map each letter entered by the user to a specific symbol in sign language. The translator can recognize speech in both English and Greek. In addition to recognizing and mapping letters to sign language symbols, the application also has the capability (only in English) to simplify, to some extent, the text provided by the user during speech input. This specific feature of the application is implemented through a machine learning model that has been “trained” on a large dataset of simple sentences and their simplified content. In summary, all of the above functionalities are achieved using technologies such as voice recognition, a chatbot (only for rendering pre-defined dialogues), a library for image processing and displaying it, a library for date recording (which is used in the CSV for proper and organized data logging), and a library for the inclusion of advanced neural networks (ANN) such as T5-small and BART for natural language processing (NLP) which are responsible for the text simplification.

Table of Contents

[ΠΕΡΙΛΗΨΗ i](#_Toc24624054)

[ABSTRACT iii](#_Toc24624055)

[ΚΕΦΑΛΑΙΟ 1 ΕΙΣΑΓΩΓΉ ΚΑΙ ΚΊΝΗΤΡΟ 1](#_Toc24624056)

**1.1** Λιγα λόγια για την νοηματική γλώσσα [1](#_Toc24624057)

**1.2** σκοπος και κινητρο της πτυχιακης εργασιας [1](#_Toc24624058)

**1.3** Εκμαθηση γραμμάτων στη νοηματική

**1.4** Εισαγωγή ατόμων στη Νοηματική γλωσσα(Με ή χωρις ακουστικά προβληματα)

[ΚΕΦΑΛΑΙΟ 2 Μοντελα μηχανικης μαθησης 2](#_Toc24624059)

**2.1** bart-base [2](#_Toc24624060)

**2.2** t5-small [2](#_Toc24624061)

**2.3** 3ο μοντέλο μηχανικής μαθησης

**2.4** Συγκριτική αξιολόγηση μεταξύ των μοντέλων

[ΚΕΦΑΛΑΙΟ 3 SPEECH RECOGNITION……… 3](#_Toc24624062)

3.1 speech recognition βιβλιοθηκη [3](#_Toc24624063)

3.2 Αναγνωριση ομιλιας με recognize\_google

3.3 λειτουργια recognizer  [3](#_Toc24624064)

[ΚΕΦΑΛΑΙΟ 4 προτεινομενο συστημα……… 4](#_Toc24624065)

4.1 γλωσσα προγραμματισμου και πακετα [4](#_Toc24624066)

4.2 αναλυση λειτουργιας του προγραμματος

4.3 διαρθωση κωδικα

4.4 εκπαιδευση μοντελων bart και t5-small [4](#_Toc24624067)

[ΚΕΦΑΛΑΙΟ 5 πειραματικη αποτιμηση 5](#_Toc24624068)

5.1 ελεγχος αποθηκευση στο csv

5.2 δοκιμες speech\_recognition σε δυο γλωσσες

5.3 Ελεγχος λειτουργιας switch/stop

5.4

5.5 Δοκιμή αντιστοιχησης εικόνων με γραμματα

5.6 Ελεγχος αναγνωρισης χαρακτηρων, αριθμων και συμβολων

5.7 Δοκιμη προβολης εικονων (ταχυτητα και ομαλοτητα)

5.8 Προτασεις απο χρηστη ASL για βελτιωση εφαρμογης

[κεφαλαιο 6 συμπερασματα και μελλοντικες προεκτασεις……………….](#_Toc24624069)

6.1 δυσκολιες στην υλοποιηση του προγραμματος

6.2 προβληματα ευρεσης λεξιλογίου νοηματικης γλωσσας

6.3 αναπτυξη της νοηματικης γλωσσας

6.4 Μελλοντικες προεκτασεις: χρηση νεων τεχνολογιων(animation bot)

6.5 προοπτικοη για προγραμμα αναγνωρισης νοηματων απο βιντεο

6.6 Δημιουργια μοντελου που συγκρινει βιντεο με κειμενα

[ΒΙΒΛΙΟΓΡΑΦΙΑ 6](#_Toc24624069)

# ΚΕΦΑΛΑΙΟ 1

# Εισαγωγή και κίνητρο

## **1.1 Λίγα λόγια για τη νοηματική γλώσσα**

Η νοηματική γλώσσα είναι μια προσπάθεια του ανθρώπου να εκφραστεί μέσο νοημάτων με τας χέρια του, με την στάση του σώματος του αλλά και τις εκφράσεις του προσώπου του για να μπορέσει να εκφράσει κάποια συναισθήματα ή καταστάσεις χωρίς να μιλήσει. Η νοηματική γλώσσα διαθέτει λεκτικές και συντακτικές δομές για να εκφράσει οποιοδήποτε αφηρημένη έννοια, και καθημερινές ανάγκες μας ( καλημέρα, χαίρω πολύ, πεινάω κ.α. ). Οι γλωσσολόγοι διακρίνουν τις φυσικές νοηματικές γλώσσες από άλλα συστήματα που προέρχονται από αυτές , όπως είναι οι τεχνητοί ανθρώπινοι κώδικες (αλγόριθμοι, γλώσσες προγραμματισμού κ.α. ), η ομιλούμενες γλώσσες, η οικιακή, η “βρεφική” και νοήματα που αποκτούνται από την βρεφική ηλικία.

Έτσι όπως υπάρχουν πολλές ομιλούμενες γλώσσες υπάρχουν και πολλές νοηματικές γλώσσες . Για παράδειγμα, η ASL ( American sign language ) είναι διαφορετική από την BSL ( British sign language) και τίς άλλες υπάρχουσες γλώσσες. [1]

Σύμφωνα με τον ethnologue υπάρχουν περισσότερες από 130 νοηματικές γλώσσες παγκοσμίως. Αυτές οι γλώσσες χρησιμοποιούνται από διάφορες κοινότητες κωφών και ημί-κωφων και ποικίλλουν από χώρα σε χώρα καθώς και σε διαφορετικές περιοχές εντός των ίδιων χωρών.[2]

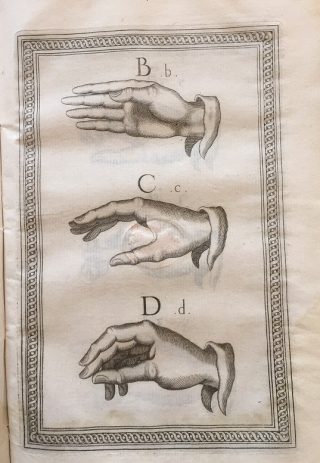
Σύμφωνα με τον Π.Ο.Υ (παγκόσμιος οργανισμός υγείας) περισσότερο από το 5% του παγκόσμιου πληθυσμού ή περίπου 430 εκατομμύρια άνθρωποι έχουν μόνιμη βλάβη στο ακουστικό τους σύστημα, είτε υπάρχει εκ γενετής ή μετά από ατύχημα. Μέχρι το 2050 εκτιμάται ότι πάνω από 700 εκατομμύρια άνθρωποι θα έχουν προβλήματα ακοής.[3]

Να σημειωθεί ότι η νοηματική γλώσσα δεν θα πρέπει να συσχετιστεί με την γλώσσα του σώματος , που θεωρείται ένα είδος μη λεκτικής επικοινωνίας.[4]

Πρώτες καταγεγραμμένες αναφορές για την νοηματική γλώσσα μπορούμε να βρούμε στο βιβλίο του Πλάτωνα “Κρατύλο” από συνομιλίες του με τον Σωκράτη.[5] Σε αυτό το σημείο να αναφέρω ότι τα άτομα με προβλήματα ακοής βρίσκονταν στο περιθώριο, διότι εκείνη την εποχή επικρατούσε η άποψη ότι ή γλώσσα μπορεί να μαθευτεί μόνο ακούγοντας τον προφορικό λόγο. Για παράδειγμα στην ρώμη σύμφωνα με το ρωμαϊκό δίκαιο επικρατούσε η άποψη ότι οι άνθρωποι που γεννιούνται κωφοί δεν είχαν ίσα δικαιώματα με τους υπόλοιπους πολίτες καθώς θεωρούταν ότι “δεν καταλάβαιναν τίποτα”.

Αυτή η προκατάληψη ενάντια στα κωφά άτομα άρχισε να εξαλείφεται την περίοδο της αναγέννησης.[6]

Μάλιστα το πρώτο άτομο που δημιούργησε μια “επίσημη” νοηματική γλώσσα είναι ο Ισπανός μοναχός Pedro Ponce de Leon τον 16ο αιώνα μΧ. Στη συνέχεια ένας άλλος Ισπανός κληρικός ο Juan Pablo Bonet κυκλοφόρησε το 1620 το πρώτο έργο για την εκμάθηση νοηματικής γλώσσας σε ανθρώπους με προβλήματα ακοής (Είναι από τα πρώτα έργα που έχουμε καταγεγραμμένα). Παρακάτω είναι μερικές από τις σελίδες του βιβλίου του Pedro.[7]



**Εικόνα 1** [**https://shorturl.at/qtZB5**](https://shorturl.at/qtZB5) **Εικόνα 2** [**https://shorturl.at/zxkDn**](https://shorturl.at/zxkDn)

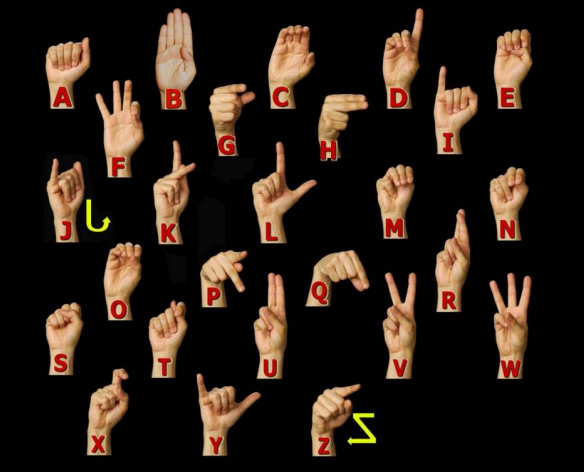
Η ASL( American sign language ) έχει ρίζες από την Αμερικανική σχολή για κωφούς που ιδρύθηκε στο Hartford, Connecticut το 1817[8]. Η συγκεκριμένη σχολή ιδρύθηκε από τον απόφοιτο του Yale Thomas Hopkins Gallaudet[9] και εκείνη την εποχή θεωρούταν σχολή για “χαζούς”.

Στην νοηματική γλώσσα ASL για να γίνει σωστά η εκτέλεση των λέξεων είναι απαραίτητες η παρακάτω 5 παράμετροι.[10]

1. Σχηματισμός χεριού
2. Προσανατολισμός παλάμης
3. κίνηση του χεριού
4. Τοποθεσία χεριού
5. Εκφράσεις σώματος (NMS)

**Σχηματισμός χεριού**

Σχηματισμοί χεριών αποτελούνται από το αλφάβητο που υπάρχει με της κινήσεις των χεριών σε διάφορες παραλλαγές σχημάτων όπως στην εικόνα παραπάνω.

****

**Εικόνα 3.** [**https://shorturl.at/iMmTu**](https://shorturl.at/iMmTu)

**Προσανατολισμός Παλάμης**

Ο σχηματισμός αναφέρεται σε ποια κατεύθυνση βλέπει η παλάμη για ένα συγκεκριμένο νοηματικό σήμα.

* Παλάμη προς τα έξω
* Παλάμη προς τα μέσα
* Παλάμη σε οριζόντια θέση
* Παλάμη βλέπεις προς τα Δεξιά/Αριστερά
* Παλάμη βλέπεις προς παλάμη
* Παλάμη βλέπει προς τα Πάνω/Κάτω

**Κίνηση του χεριού**

Ένα νόημα μπορεί να διακριθεί με κάποιες από τις παραπάνω αναριθμηζόμενες

κινήσεις.

* Κυκλική κίνηση
* Πάνω και κάτω
* Προς τα μπροστά
* Προς τα πίσω
* χτύπημα
* Μπροστά και πίσω
* Διάφορα κουνήματα

**Τοποθεσία χεριού**

Η τοποθεσία είναι ο χώρος στον οποίο λαμβάνει χώρα το νόημα που κάνει ο άνθρωπος σε σχέση με την τοποθεσία του σώματος του . Παραπάνω είναι όλες οι τοποθεσίες που μπορούν να γίνουν τα νοήματα, να σημειωθεί ότι ένα νόημα μπορεί να αρχίσει για παράδειγμα από το κεφάλι και να ολοκληρωθεί στο στήθος.

* Πιγούνι
* Ώμος
* Μπροστά από το σώμα
* Αριστερά και δεξιά του σώματος
* Μέτωπο

**Εκφράσεις σώματος**

Η παραπάνω εκφράσεις σώματος αναφέρονται ως συμπληρωματικές για να υπάρχει μεγαλύτερη κατανόηση σε νοήματα.

* Κούνημα κεφαλιού ( Καταφατικά/αρνητικά)
* Κούνημα φρυδιών
* Κούνημα μύτης
* Νοήματα με τα μάτια
* Χρήση των χειλιών

## **1.2 Σκοπός και κίνητρο της πτυχιακής εργασίας**

Ο σκοπός της συγκεκριμένης πτυχιακής είναι η εξοικείωση ανθρώπων χωρίς νοηματικά προβλήματα με το αλφάβητο της ASL και GSL. Είναι μια απλή προσπάθεια της τριβής ανθρώπων που δεν έχουν ξαναδεί νοήματα μέσο της ομιλίας τούς ή από την είσοδο στο πληκτρολόγιο στο πρόγραμμα που έχω υλοποιήσει. Περαιτέρω θα ήθελα μελλοντικά να ασχοληθώ με την αλληλεπίδραση προγραμμάτων χρήστη-υπολογιστή αλλά κυρίως με την κατανόηση της Νοηματικής και δημιουργία μιας πιο εξελιγμένης εφαρμογής.

Αυτό που με ώθησε και ήθελα να ασχοληθώ με το συγκεκριμένο θέμα είναι το συναισθηματικό κομμάτι, καθώς με μια απλή εφαρμογή μπορείς να κάνεις πιο εύκολη την ζωή κάποιου με κάποιες γραμμές κώδικα. Πέρα από την ευκολία της μετάφρασης των νοημάτων μπορείς να διευκολύνεις και κάποιον που θέλει να κάνει μια αρχική εισαγωγή στην νοηματική γλώσσα.

Η συγκεκριμένη εφαρμογή για εμένα δεν απευθύνεται σε άτομα που γνωρίζουν ήδη την Νοηματική γλώσσα αλλά σε άτομα που σε μια μεγαλύτερη ή και μικρότερη ηλικία θέλουν να βοηθηθούν στην εκμάθηση της . Κανείς δεν ξέρει πότε θα γνωρίσει κάποιον σημαντικό άνθρωπο που θα είναι κωφός και θα χρειαστεί να επικοινωνήσει μαζί του μέσο νοημάτων , οι περισσότεροι από εμάς θα θέλαμε να γνωρίζουμε την γλώσσα του για να μπορούμε να εμβαθύνουμε την σχέση μας με το συγκεκριμένο άτομο.

Στην τελική αυτός είναι ο σκοπός της εφαρμογής, μια μικρή εισαγωγή στην νοηματική γλώσσα και το να φέρουμε τους ανθρώπους πιο κοντά.

## **1.3 Εκμάθηση γραμμάτων στην νοηματική**

Η εκμάθηση του αλφαβήτου της Νοηματικής γλώσσας είναι σημαντική για την ανάπτυξη επικοινωνίας με βαρήκοα και κωφά άτομα. Η εκμάθηση είναι πολύ πιο αποδοτική με άτομα που την γνωρίζουν και μιλάνε, αλλά δεν έχουν όλοι την ευκαιρία να έχουν στον κύκλο τους ένα άτομο που να μιλάει και να χρησιμοποιεί την νοηματική γλώσσα. Για αυτό τον λόγο με το συγκεκριμένο πρόγραμμα ο μέσος χρήστης θα έχει την δυνατότητα να μιλάει με το πρόγραμμα ή να γράφει τι θέλει να πει και με τον καιρό να αρχίσει να μαθαίνει το κάθε νόημα σε ποιο γράμμα αντιστοιχεί.

Παρακάτω είναι το αλφάβητο της ASL.

Εικόνα που περιέχει πεταλούδα

Περιγραφή που δημιουργήθηκε αυτόματα

**Εικόνα 4** [**https://shorturl.at/pyVqo**](https://shorturl.at/pyVqo)

## **1.4 Εισαγωγή ατόμων στη Νοηματική γλώσσα( με ή χωρίς ακουστικά προβλήματα)**

Πέρα από την βοήθεια στην εκμάθηση της νοηματικής το πρόγραμμα βοηθάει και στην εισαγωγή ατόμων που δεν έχουν καμιά επαφή με την Νοηματική γλώσσα στα σύμβολα που χρησιμοποιούνται. Μπορεί να υπάρξουν περιπτώσεις οι οποίες δεν έχουν ξαναδεί κανένα σύμβολο.

Ακόμα η νοηματική προσφέρει μια πολυδιάστατη επικοινωνία στον άνθρωπο.

Ενισχύει την ικανότητα επικοινωνίας του είτε έχει κάποιο ακουστικό πρόβλημα ή

απλά την χρησιμοποιεί ως εναλλακτικό μέσο επικοινωνίας, επιπλέον συμβάλει και στην κοινωνική του συνοχή με άτομα της κωφής κοινότητας προάγοντας την συμπερίληψη και την κατανόηση διαφορετικών πολιτισμών.

Πέρα από όλα αυτά ανοίγονται και επαγγελματικής φύσεως ευκαιρίες διότι η εκμάθηση της νοηματικής γλώσσας μπορεί να προσφέρει εργασία σε όλους τους επαγγελματικούς χώρους όπου υπάρχει η κοινότητα των κωφών.[11]

Τέλος έχουν δείξει έρευνες ότι με την εκμάθηση της νοηματικής δίνεται η δυνατότητα ανάπτυξης γλωσσικών αλλά και γνωστικών δεξιοτήτων σε άτομα όλων των ηλικιών και πως αυτά τα άτομα έχουν βελτιωμένη νοητική ευελιξία, επίλυση καθημερινών προβλημάτων και αποκτούν δημιουργική σκέψη. [12]

# ΚΕΦΑΛΑΙΟ 2 Μοντέλα μηχανικής μάθησης

## 2.1 Text-simplification

Η απλοποίηση κειμένου είναι μια διαδικασία που αποσκοπεί στη μείωση της πολυπλοκότητας του λόγου, ώστε το κείμενο να γίνει πιο κατανοητό, ιδιαίτερα από άτομα με περιορισμένες αναγνωστικές δεξιότητες ή γλωσσική επάρκεια. Στην περίπτωση του προγράμματος που υλοποιώ θέλω να προσθέσω την απλοποίηση κειμένου για την διευκόλυνση του χρήστη στην μετάφραση της νοηματικής γλώσσας. Με λίγα λόγια το σκεπτικό είναι ότι είτε από είσοδο μικροφώνου ή από το πληκτρολόγιο ο χρήστης θα μπορεί να δίνει την πρόταση που θέλει για μετάφραση και το πρόγραμμα με βάση τα δεδομένα που έχει εκπαιδευθεί θα υλοποιεί μια απλοποίηση του κειμένου που στην συνέχεια θα μεταφράζεται από το πρόγραμμα μου.

Η απλοποίηση κειμένου συνδέεται με τον κλάδο του **Natural Language processing** **(NLP)** όπου είναι ένας τομέας της επιστήμης υπολογιστών και ιδιαίτερα της τεχνητής νοημοσύνης. Η **NLP** ουσιαστικά επιτρέπει στους υπολογιστές να κατανοούν, να επεξεργάζονται και να αναπαράγουν την ανθρώπινη γλώσσα Ουσιαστικά με το **NLP** γίνεται η καταγραφή κειμένων από διαφορετικές πηγές όπως το πληκτρολόγιο η το μικρόφωνο και να εφαρμοστούν αυτόματα μέθοδοι απλοποίησης ή παραπάνω επεξεργασία από εκπαιδευμένα μοντέλα για να γίνει στην τελική η μετάφραση στην νοηματική γλώσσα με τα κατάλληλα νοήματα.[13]

Ξεκινώντας η απλοποίηση κειμένου αποτελείται από την **Λεξιλογική**, **Συντακτική** και **Νοηματική** απλοποίηση[14].

* Η **Λεξιλογική** απλοποίηση αφορά την αναγνώριση σύνθετων λέξεων, όπως σπάνιων, τεχνικών ή αφηρημένων, και την αντικατάστασή τους με απλούστερα και πιο κατανοητά συνώνυμα. Εναλλακτικά, το κείμενο μπορεί να εμπλουτιστεί με ορισμούς, εικόνες ή βίντεο για να διευκολύνει την κατανόηση. Κατά τη διαδικασία αυτή, η αποσαφήνιση παίζει σημαντικό ρόλο, καθώς απαιτείται η επιλογή της πιο κατάλληλης έννοιας από μια λίστα συνωνύμων. Ωστόσο, η εξάρτηση από την πιο συχνή έννοια μιας λέξης μπορεί να οδηγήσει σε προβλήματα που χρειάζονται περαιτέρω έρευνα.
* Η **Συντακτική** απλοποίηση στοχεύει στη μείωση της πολυπλοκότητας των προτάσεων, όπως παθητικές φράσεις, μεγάλες προτάσεις και σχετικές προτάσεις. Αυτό επιτυγχάνεται μέσω αναδιάταξης, διαχωρισμού ή απλοποίησης των γραμματικών δομών, με στόχο την εξάλειψη περιττών πληροφοριών.
* Η **Νοηματική** απλοποίηση του λόγου διασφαλίζει ότι καμία ουσιαστική πληροφορία δεν χάνεται κατά τη λεξιλογική ή συντακτική απλοποίηση. Αυτό το βήμα επικεντρώνεται στη διατήρηση της συνοχής του κειμένου, αναλύοντας τις αναφορές και αντικαθιστώντας επαναλαμβανόμενες οντότητες ή απλοποιώντας τις ονοματικές φράσεις.[15][16]

## 2.2 BART (Bidirectional and Auto-Regressive Transformers)

 Το **BART (Bidirectional and Auto-Regressive Transformers)** είναι ένας αυτόματος κωδικοποιητής αποθορυβοποίησής (**Denoising autoencoder**) που έχει υλοποιηθεί με ένα μοντέλο σχέσεων (sequence-to-sequence model) που μπορεί να χρησιμοποιηθεί σε ένα μεγάλο φάσμα εργασιών. Με λίγα λόγια είναι ένας συνδυασμός της αρχιτεκτονικής του **BERT (Bidirectional encoder)** και του **GPT (Left to Right Decoder)**.

**Εικόνα 5** [**https://shorturl.at/dNSPc**](https://shorturl.at/dNSPc)

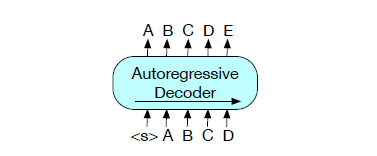
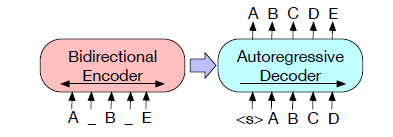
Το **BART** έχει σχεδιαστεί από το Facebook AI καιείναι αρκετά αποδοτικό, όταν ρυθμιστεί στις κατάλληλες παραμέτρους και δεδομένα, για να επιλύσει εργασίες **NLP (Natural language processing)**, όπως η δημιουργία κειμένου αλλά και εργασίες λογικής. Στην συγκεκριμένη περίπτωση θα το χρειαστούμε για να επιλύσουμε **εργασίες λογικής** δηλαδή περίληψη κειμένου (Text-summarization), αλλά αρχικά ας απεικονίσουμε το πώς είναι η βασική αρχιτεκτονική του μοντέλου.

Όπως προαναφέρθηκα το **BART** μπορεί να αναφερθεί και ως ένας συνδυασμός παρόμοιας αρχιτεκτονικής του **BERT+GPT** τα οποία είναι παρόμοια μοντέλα.

* Εικόνα που περιέχει κείμενο, γραμματοσειρά, σχεδίαση

  Περιγραφή που δημιουργήθηκε αυτόματαΤο μοντέλο **BERT** λειτουργεί με την **αντικατάσταση τυχαίων tokens** με **ειδικές masks ([MASK] Element)** και κωδικοποιεί το κείμενο αμφίδρομα, δηλαδή λαμβάνοντας υπόψη το συμφραζόμενο και πριν και μετά το κάθε token. Ωστόσο τα **tokens** που δεν υπάρχουν προβλέπονται ανεξάρτητα, έτσι το BERT δεν είναι κατάλληλο για δημιουργία νέου κειμένου.[18]

**Εικόνα 7[17].**

* Το μοντέλο **GPT**, τα tokens προβλέπονται **auto-regressively**, κάτι που επιτρέπει τη χρήση του για δημιουργία κειμένου. Ωστόσο το μοντέλο μπορεί να βασιστεί μόνο στα συμφραζόμενα που βρίσκονται στα αριστερά του κάθε token, γεγονός που σημαίνει ότι δεν μπορεί να κατανοήσει πλήρως αμφίδρομες αλληλεπιδράσεις στο κείμενο.[19] **Εικόνα 8[17].**
* Στο μοντέλο **BART**, τα δεδομένα εισόδου στον κωδικοποιητή (**Encoder**) δεν χρειάζεται να ταιριάζουν με αυτά που παράγει ο αποκωδικοποιητής (**Decoder**), δίνοντας τη δυνατότητα για

**Εικόνα 9[17].**

αυθαίρετες/τυχαίες μετατροπές θορύβου. Για παράδειγμα, ένα έγγραφο μπορεί να αλλοιωθεί αντικαθιστώντας τμήματα κειμένου με **mask symbols**. Αυτό το αλλοιωμένο έγγραφο κωδικοποιείται (**Encoded)** με ένα αμφίδρομο μοντέλο, και στη συνέχεια υπολογίζεται η πιθανότητα του αρχικού κειμένου μέσω ενός (**Auto-regressive Decoder)**. Στη φάση της βελτιστοποίησης, ένα μη αλλοιωμένο έγγραφο δίνεται και στον κωδικοποιητή (**Encoder)** και στον αποκωδικοποιητή (**Decoder)**, και χρησιμοποιούνται οι τελικές αναπαραστάσεις από την κρυφή κατάσταση (**Hidden state)** του αποκωδικοποιητή (**Decoder**).[20]

Ας δούμε και λίγο πιο αναλυτικά τον transformer που χρησιμοποιείται και τον κώδικα για αυτόν. Η παρακάτω φωτογραφία είναι το βασικό πλαίσιο για το μοντέλο μας.

Εικόνα που περιέχει κείμενο, διάγραμμα, στιγμιότυπο οθόνης, Σχέδιο

Περιγραφή που δημιουργήθηκε αυτόματα

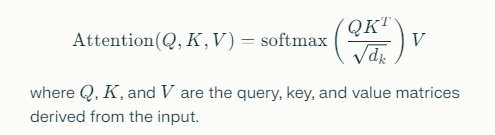
**Εικόνα 10** [**https://nlp.seas.harvard.edu/annotated-transformer/**](https://nlp.seas.harvard.edu/annotated-transformer/)**[23]**

Στην παραπάνω εικόνα βλέπουμε την αρχιτεκτονική κωδικοποιητή-αποκωδικοποιητή **(Encoder-Decoder) Transformer** είναι ένα πρωτοποριακό πλαίσιο στον τομέα του **Deep learning**, ειδικά ανεπτυγμένο για επεξεργασία διαδοχικών εργασιών δεδομένων, όπως **automatic translation**, η **text summarization** που χρειαζόμαστε στην περίπτωσή μας.[22]

Ας ξεκινήσουμε με μια περιεκτική ανάλυση του αριστερού μέρους του transformer μας τον κωδικοποιητή (**Encoder)** που είναι υπεύθυνος για την επεξεργασία της ακολουθίας (**sequence).**

**Multi-Head Self-Attention**

Επιτρέπει στον κωδικοποιητή να εστιάζει σε διαφορετικά μέρη της ακολουθίας εισόδου (**input-sequence).** Με λίγα λόγια στα **Self-Attention** layers όλα τα **keys,values** και queries προέρχονται από το ίδιο σημείο, οπού σε αυτή την περίπτωση είναι το προηγούμενο **layer** στον κωδικοποιητή (**Encoder)** και έτσι κάθε θέση στον κωδικοποιητή μπορεί να λάβει υπόψιν το προηγούμενο **layer** στον κωδικοποιητή (**Encoder)** **.** Στην παρακάτω class **“MultiHeadedAttention”** με τα δοθέντα δεδομένα υπολογίζει το **Attention** σε διάφορα **Head** παράλληλα. Ο παρακάτω είναι ο μαθηματικός τύπος και μετά ο κώδικας.



**Εικόνα 11** [**https://nlp.seas.harvard.edu/annotated-transformer/**](https://nlp.seas.harvard.edu/annotated-transformer/)**[23]**

Στην συνέχεια μπορούμε να δούμε και σε κώδικα πώς πραγματοποιείται αλλά και σε ένα απλό διάγραμμα που βλέπουμε που χρησιμοποιούνται οι τιμές αυτές.

**Εικόνα που περιέχει κείμενο, στιγμιότυπο οθόνης, γραμματοσειρά, αριθμός

Περιγραφή που δημιουργήθηκε αυτόματα**

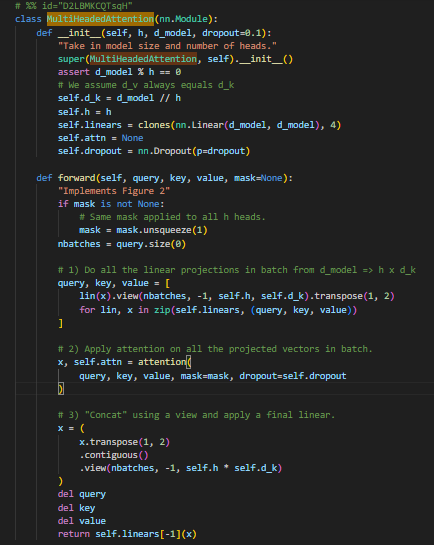
**Εικόνα 12** [**https://nlp.seas.harvard.edu/annotated-transformer/[23**](https://nlp.seas.harvard.edu/annotated-transformer/%5b23)**]**

**Εικόνα που περιέχει κείμενο, στιγμιότυπο οθόνης, γραμματοσειρά

Περιγραφή που δημιουργήθηκε αυτόματα**

**Εικόνα 12.1** [**https://github.com/ikoctac/annotated-transformer/[22**](https://github.com/ikoctac/annotated-transformer/%5b22)**]**

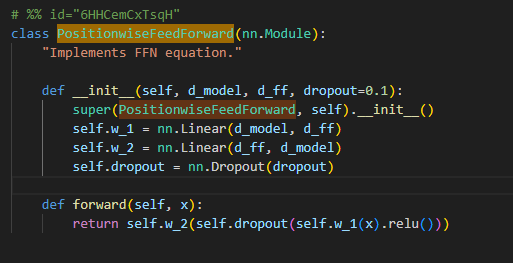
Η **MultiHeadedAttention** class που χρησιμοποιεί και το attention του παραπάνω κώδικα.

****

**Εικόνα 13** [**https://github.com/ikoctac/annotated-transformer**](https://github.com/ikoctac/annotated-transformer)**[22]**

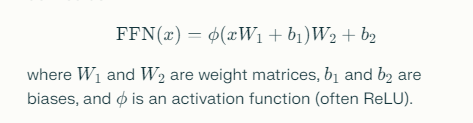
**Feed-Forward N****etwork (FFN)**

Εφόσον ολοκληρωθεί το πρώτο κομμάτι και υπολογισθεί το **Attention,** τα δεδομένα περνάνε μέσω ενός διπλού **Fully-Connected Network Layer.** Συγκεκριμένα το μοντέλο χρησιμοποιεί **Learned embeddings[24]** για την μετατροπή των **input-tokens** και **output-tokes** σε **Vectors χ** διαστάσεων. Επίσης χρησιμοποιούνται οι συναρτήσεις **Learned-linear-transformation** και **SoftMax** για να με μετατραπεί η έξοδος του αποκωδικοποιητή στα επόμενα **Predicted-Token-Probabilities.** Στο παραπάνω μοντέλο τα **Two-embedding-layers** μοιράζονται τα ίδια **weight-matrix** και την **pre-SoftMax linear transformation[24].** Τέλος στα **embedding layers** πολλαπλασιάζουμε τα **weights** με

****

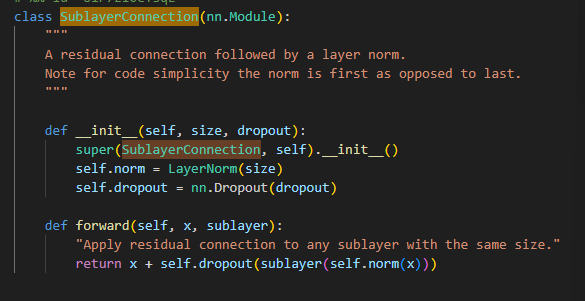
**Εικόνα 13** [**https://github.com/ikoctac/annotated-transformer [22**](https://github.com/ikoctac/annotated-transformer%20%5b22)**]**

Ο παραπάνω κώδικας υπολογίζει την μαθηματική φόρμουλα για το **FFN (Feed-Forward-Network).**

****

**Εικόνα 14** [**https://nlp.seas.harvard.edu/annotated-transformer/**](https://nlp.seas.harvard.edu/annotated-transformer/)**[22]**

**Add &** **Norm**

****Οι υπολειπόμενες συνδέσεις μεταφέρουν την είσοδο με την έξοδο του **sublayer** και στην συνέχεια εφαρμόζουν **Layer-Normalization** στην αντίστοιχη class **LayerNorm.**

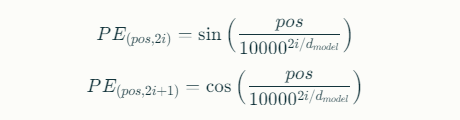
**Εικόνα 15.1** [**https://github.com/ikoctac/annotated-transformer [22**](https://github.com/ikoctac/annotated-transformer%20%5b22)**]**

**Εικόνα που περιέχει κείμενο, ηλεκτρονικές συσκευές, στιγμιότυπο οθόνης, οθόνη

Περιγραφή που δημιουργήθηκε αυτόματαΕικόνα 15.2** [**https://github.com/ikoctac/annotated-transformer**](https://github.com/ikoctac/annotated-transformer) **[22]**

**Positional** **Encoding**

Για να ενσωματωθούν οι πληροφορίες για την τοποθεσία στο μοντέλο προστίθενται επιπλέον κωδικοποιήσεις (**Encodings)** τοποθεσίας για την ενσωμάτωση του **embedding-input,** αυτό βοηθάει το μοντέλο να καταλαβαίνει την σειρά τον **tokens in sequences.**

****

**Εικόνα 16** [**https://nlp.seas.harvard.edu/annotated-transformer/**](https://nlp.seas.harvard.edu/annotated-transformer/) **[22]**

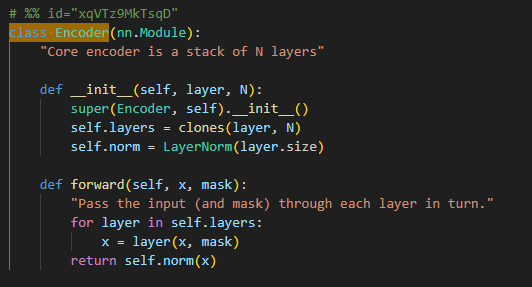
**Encoder Stacking**

Τέλος έχουμε το **Encoder stacking** το οποίο επαναλαμβάνει τα βήματα για το **Attention, FFN (Feed-Forward-Network)** και **Normalization** μέσα από πολλαπλά στρώματα (στο μοντέλο μας **BART-base** είναι 6 ενώ στο **BART-Large** είναι 12). Το **Stacking** επιτυγχάνεται με την class **“EncoderLayer”** η οποία βρίσκεται στην class **“Encoder”.**

Εικόνα που περιέχει κείμενο, στιγμιότυπο οθόνης, οθόνη, λογισμικό

Περιγραφή που δημιουργήθηκε αυτόματα

**Εικόνα 17** [**https://github.com/ikoctac/annotated-transformer [22**](https://github.com/ikoctac/annotated-transformer%20%5b22)**]**

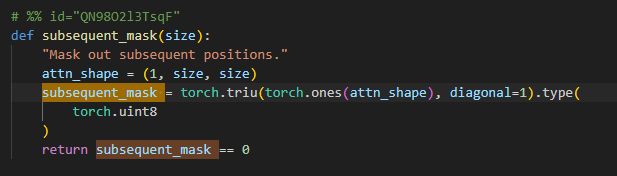
****

**Εικόνα 18** [**https://github.com/ikoctac/annotated-transformer [22**](https://github.com/ikoctac/annotated-transformer%20%5b22)**]**

Αυτά όσον αφορά το αριστερό κομμάτι του **Transformer,** στη συνέχεια έχουμε στο δεξιό κομμάτι τον αποκωδικοποιητή (**Decoder**) ο οποίος δημιουργεί μια ακολουθία εξόδου βασισμένη στην κωδικοποιημένη **(Encoded)** είσοδο και τα **tokens** που δημιουργήθηκαν προηγούμενος και αποτελείται από τα παρακάτω κομμάτια.

**Masked Multi-Head Self-Attention**

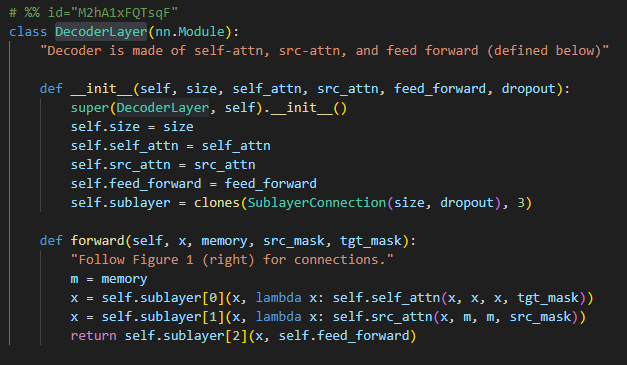
Όπως και στον κωδικοποιητή (**Encoder**) έτσι και εδώ έχουμε παρόμοιο πρώτο στρώμα είναι ίδιο με το **”**[**MultiHeadSelfAttention**](#MultiHeadSelfAttention)**”** μόνο που χρησιμοποιεί **masking** για να αποτραπεί η εστίαση σε μελλοντικά **tokens.** Έτσι στο **“DecodeLayer”** το πρώτα στρώμα που χρησιμοποιεί την **“**[**MultiHeadSelfAttention**](#MultiHeadSelfAttention)**”** μαζί με την **“subsequent\_mask”.**



**Εικόνα 19** [**https://github.com/ikoctac/annotated-transformer [22**](https://github.com/ikoctac/annotated-transformer%20%5b22)**]**

**Encoder-Decoder Attention**

Ο αποκωδικοποιητής **(Decoder)** ελέγχει και τα αποτελέσματα του κωδικοποιητή **(Encoder)** και ελέγχει και μαθαίνει από τις σχέσεις της εισόδου/εξόδου, και αυτό είναι το δεύτερο υπόστρωμα στον **“DecoderLayer”** και χρησιμοποιεί και αυτό την “[**MultiHeadSelfAttention**](#MultiHeadSelfAttention)**”.**

****

**Εικόνα 20** [**https://github.com/ikoctac/annotated-transformer [22**](https://github.com/ikoctac/annotated-transformer%20%5b22)**]**

**Feed-Forward Network (FFN)**

Ακολουθεί την ίδια λογική με τον κωδικοποιητή **(Encoder)** και χρησιμοποιεί την class **PositionwiseFeedForward (**[**FeedForwardNetwork**](#FeedForwardNetwork)**).**

**Add & Norm**

Παρόμοια με τον κωδικοποιητή **(Encoder)** έτσι και ο αποκωδικοποιητής **(Decoder) χρησιμοποιεί το ίδιο “**[**AddNorm**](#AddNorm)**”**

**Decoder Stacking**

Πάλι όπως και στον κωδικοποιητή **(Encoder)** παρόμοια και στον αποκωδικοποιητή **(Decoder)** αποτελείται από 6 στρώματα επαναλαμβάνοντας τα παραπάνω βήματα για την εξαγωγή αποτελεσμάτων μόνο που αυτή την φορά χρησιμοποιούμε το **“DecoderLayer”** το **“Decoder”** class.

Εικόνα που περιέχει κείμενο, ηλεκτρονικές συσκευές, στιγμιότυπο οθόνης, λογισμικό

Περιγραφή που δημιουργήθηκε αυτόματα

**Εικόνα 21** [**https://github.com/ikoctac/annotated-transformer [22**](https://github.com/ikoctac/annotated-transformer%20%5b22)**]**

**Εικόνα που περιέχει κείμενο, στιγμιότυπο οθόνης, γραμματοσειρά, λογισμικό

Περιγραφή που δημιουργήθηκε αυτόματα**

**Εικόνα 22** [**https://github.com/ikoctac/annotated-transformer**](https://github.com/ikoctac/annotated-transformer) **[22]**

**Final Linear & SoftMax**

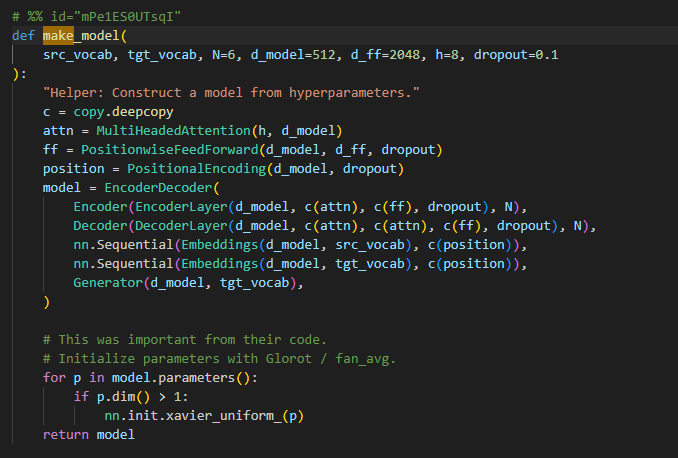
Τέλος η τελική έξοδος από τον αποκωδικοποιητή μετατρέπεται σε μια πιθανότητα πάνω στα δεδομένα που έχει προπονηθεί το μοντέλο και εξάγει ένα αποτέλεσμα με την class **“Generator”.**

**Εικόνα που περιέχει κείμενο, στιγμιότυπο οθόνης, γραμματοσειρά

Περιγραφή που δημιουργήθηκε αυτόματα**

**Εικόνα 23** [**https://github.com/ikoctac/annotated-transformer [22**](https://github.com/ikoctac/annotated-transformer%20%5b22)**]**

Και τέλος η συνάρτηση που δημιουργεί το μοντέλο μας από μια συνάρτηση με όλες της Hyperparameters.



**Εικόνα 24** [**https://github.com/ikoctac/annotated-transformer [22**](https://github.com/ikoctac/annotated-transformer%20%5b22)**]**

**Pre-training BART**

Εφόσον έχουμε τον κώδικα για το πρόγραμμα μας έχει έρθει η ώρα για το **pre-training** του **BART.** Η εκπαίδευση του μοντέλου γίνεται μέσο του **Document corrupting** και της ανακατασκευής αυτών προσπαθώντας να βελτιστοποιήσει την ανακατασκευή αυτών. Το μοντέλο σε αντίθεση με άλλες **de-noising** προσεγγίσεις που λειτουργούν σε συγκεκριμένα **noising schemes** μας επιτρέπει να χρησιμοποιήσουμε οτιδήποτε τύπου δεδομένων χωρίς περιορισμούς.

Οι παρακάτω μετασχηματισμοί που υφίστανται στο **BART** ευθύνονται για την δημιουργία θορύβου **(Noising)** στο αρχείο μας.

1. Token Masking
2. Token Deletion
3. Text Infilling
4. Sentence permutation
5. Document Rotation

Στο παρακάτω σχήμα μπορούμε να δούμε και τι κάνουν στην εισαγωγή κειμένου.

Εικόνα που περιέχει κείμενο, στιγμιότυπο οθόνης, γραμματοσειρά, αριθμός

Περιγραφή που δημιουργήθηκε αυτόματα

**Εικόνα 25 Transformations for noising[20]**

**Token Masking**

Όπως και στο **BERT** γίνεται τυχαίο sampling στα δεδομένα και γίνεται αντικατάσταση με στοιχεία [**MASK].**

**Token Deletion**

Τυχαία **tokens** διαγράφονται από την εισαγωγή των δεδομένων, σε αντίθεση με το **token masking** σε αυτή την περίπτωση πρέπει να ορίσει το μοντέλο ποιες θα είναι αυτές οι θέσεις που θα διαγραφούν στην εισαγωγή δεδομένων.

**Text Infilling**

Η διαδικασία **Text Infilling (**Συμπλήρωση κειμένου) περιλαμβάνει την επιλογή τμημάτων κειμένου με μήκη που προέρχονται από την κατανομή **Poisson,** όπου κάθε τμήμα αντικαθίσταται από ένα Token[MASK]. Αυτή η μέθοδος είναι εμπνευσμένη από το **span-BERT.** Εν κατακλείδι ο σκοπός της συμπλήρωσης κενού είναι να εκπαιδευτή το μοντέλο και να μπορεί να βρίσκει πόσα **tokens** λείπουν από κάθε τμήμα των δεδομένων.

**Sentence Permutation**

Η διαδικασία περιλαμβάνει τη διάσπαση ενός εγγράφου σε προτάσεις, που προσδιορίζονται από τις τελείες. Στη συνέχεια ,γίνεται τυχαίο ανακάτεμα στις προτάσεις δημιουργώντας μια νέα σειρά που θα χρησιμοποιηθεί για τον σκοπό της εκπαίδευσης ανάλυσης κειμένου στην περίπτωσή μας.

**Document Rotation**

H διαδικασία περιλαμβάνει την τυχαία επιλογή ενός **token** από το έγγραφο και γίνεται το **document rotation** ώστε να ξεκινάει με αυτό το **token.** Με αυτή την μέθοδο έχει σαν στόχο να εκπαιδεύσει το μοντέλο να αναγνωρίζει που βρίσκεται η αρχή και με αυτόν τον τρόπο να το κάνει καλύτερο στο να κατανοεί την δομή αλλά και την οργάνωση των δεδομένων.

## 2.3 BART (Bidirectional and Auto-Regressive Transformers)

# ΚΕΦΑΛΑΙΟ 3 ………

## (Υποκεφάλαιο 3.1)

### (Ενότητα 3.1.α)

# ΚΕΦΑΛΑΙΟ 4 ………

## (Υποκεφάλαιο 4.1)

### (Ενότητα 4.1.α)

# ΚΕΦΑΛΑΙΟ 5 Συμπεράσματα

# ΒΙΒΛΙΟΓΡΑΦΙΑ

Εδώ θα προστεθεί όλη η βιβλιογραφία