ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

Επεξεργασία Φυσικής Γλώσσας

Απαλλακτική εργασία Εμβόλιμης εξεταστικής Δεκεμβρίου 2025

Παραδοτέο 3 – Δομημένη αναφορά

ΣΤΑΥΡΟΣ ΠΑΠΟΥΤΣΗΣ
Π21193

ΠΕΙΡΑΙΑΣ
06 Δεκεμβρίου 2025

Περιεχόμενα

Εισαγωγή **4**

Μεθοδολογία **4**

Μεθοδολογία Παραδοτέου 1 4

Μεθοδολογία Παραδοτέου 2 4

Πειράματα και Αποτελέσματα **5**

Συζήτηση **5**

Συμπεράσματα **5**

Βιβλιογραφία **5**

Εισαγωγή

Η παρούσα εργασία αφορά την επεξεργασία φυσικής γλώσσας (NLP) και περιλαμβάνει τρία βασικά στάδια: (α) ανάλυση και ανακατασκευή δύο κειμένων με διαφορετικές μεθοδολογίες, (β) τη σημασιολογική σύγκριση των παραγόμενων εκδοχών και (γ) τη συνολική τεκμηρίωση και αξιολόγηση της διαδικασίας. Στο πρώτο παραδοτέο υλοποιήθηκαν ένας κανονιστικός αναλυτής και τρία αυτόματα pipelines ανακατασκευής. Στο δεύτερο παραδοτέο πραγματοποιήθηκε σύγκριση των εκδοχών μέσω embeddings και cosine similarity. Η αναφορά αυτή συνοψίζει τη μεθοδολογία που ακολουθήθηκε σε κάθε στάδιο, παρουσιάζει τα αποτελέσματα και αξιολογεί τα pipelines ως προς την απόδοσή τους.

Μεθοδολογία

Μεθοδολογία Παραδοτέου 1

Παραδοτέο 1Α

Για το παραδοτέο 1Α η αρχιτεκτονική της εργασίας βασίζεται σε ξεχωριστά τμήματα που συνθέτουν την ολοκληρωμένη μορφή της, ένα main file, και 3 ακόμα.

Main.py

Η main συνάρτηση είναι υπεύθυνη για την τροφοδότηση με δεδομένα στα υπόλοιπα files. Εισάγει τις κύριες συναρτήσεις των φακέλων, δημιουργεί τα paths για να διαβάσει τις συμβολοσειρές από τα υπόλοιπα αρχεία και μέσω της συνάρτησης load_sentence_from_file() μπορεί να διαβάσει τις προτάσεις από 2 αρχεία και να τα στείλει στις υπόλοιπες συναρτήσεις.

Preprocessing.py

Το πρώτο στάδιο πριν την επεξεργασία των 2 προτάσεων είναι η προ-επεξεργασία κατά την οποία γίνεται ένας καθαρισμός κειμένου. Πρέπει να πραγματοποιηθεί γιατί το αρχικό κείμενο είναι μη δομημένο και θορυβώδες άρα απαιτεί κανονικοποίηση πριν οποιαδήποτε επεξεργασία. Το πρώτο βήμα της κανονικοποίησης είναι η αφαίρεση συντομεύσεων (π.χ. Don’t -> Do not) ώστε να έχουμε ολόκληρες λέξεις. Έπειτα όλοι οι χαρακτήρες μετατρέπονται σε πεζά για να εξαλειφθούν τεχνητές διαφορές μεταξύ λέξεων που διαφέρουν μόνο ως προς

το casing. Η αφαίρεση σημείων στίξης και ειδικών χαρακτήρων πραγματοποιείται για την μείωση του θορύβου καθώς αυτά δεν συμβάλουν στην σημασιολογική ανάλυση των λέξεων. Υπάρχει σε αυτό το σημείο ένα μικρό bug όπου δημιουργούνται διπλά κενά κατά την αφαίρεση οπότε πρέπει να αντικατασταθούν τα διπλά κενά με ένα μόνο κενό για να έχουμε ένα ομοιόμορφο κείμενο. Στη συνέχεια έχουμε tokenization με τη χρήση του NLTK. Έπειτα πρέπει να καταλάβει το πρόγραμμα τι μέρος του λόγου είναι κάθε λέξη. Αυτό επιτυγχάνεται με τη χρήση του POS tagging (Part-Of-Speech tagging) ώστε να αποδοθούν γραμματικές ετικέτες σε κάθε token, παρέχοντας βασική συντακτική πληροφορία που είναι απαραίτητη για μεταγενέστερα στάδια ανάλυσης. Τέλος εφαρμόζεται lemmatization (lemma) λημματοποίηση με σκοπό τη μετατροπή λέξεων στη βασική τους μορφή, λαμβάνοντας υπόψη τη γραμματική τους κατηγορία ως μέρος του λόγου, ώστε να διατηρείται το νόημα ενώ μειώνεται η μορφολογική ποικιλία. Με αυτό το τρόπο έχει επιτευχθεί το πρώτο στάδιο στο pipeline και η δημιουργία ενός νέου string.

Syntactical_analysis.py

Αφού έχουμε πάρει από το preprocessing το νέο string σε αυτό το στάδιο πρέπει να το ανασυντάξουμε με βάση τα tokens. Η συντακτική ανάλυση υλοποιείται σε ένα επιφανειακό επίπεδο χωρίς ένα πλήρες συντακτικό parsing αλλά σε επιφανειακή μορφοσυντακτική πληροφορία. Θεμέλιο της διαδικασίας είναι τα POS tags από πριν που είναι κλειδί για την κατανόηση της δομής μιας πρότασης.Η αναγνώριση ονοματικών φράσεων (noun phrases) που πραγματοποιείται, βασίζεται σε απλά POS patterns που αντιστοιχεί στο chunking (που παρουσιάζεται ως μια τυπική μορφή shallow parsing στο βιβλίο [1]). Οι ρηματικές ομάδες (verb groups) εντοπίζονται από τις ετικέτες και διαχωρίζονται σε κύρια και βοηθητικά ρήματα, πρακτική που ευθυγραμμίζεται με την POS ανάλυση ρηματικών δομών. Η εξαγωγή της δομής SVO (Subject-Verb-Object) χρησιμοποιείται ως λειτουργική αναπαράσταση της πρότασης, και αποτελεί βασική δομή αναπαράστασης και πληροφοριακής ανάλυσης χωρίς πλήρη γραμματική. Οι προσθετικές φράσεις αντιμετωπίζονται ως ανεξάρτητες συντακτικές μονάδες (IN + NP) σύμφωνα με τη διάκριση φραστικών δομών που γίνεται στο πλαίσιο της επιφανειακής συντακτικής ανάλυσης. Ο εντοπισμός δευτερεύουσων προτάσεων βασίζεται σε ρητές υποδεέστερες συνδετικές λέξεις, τεχνική που αναγνωρίζεται ως απλή αλλά αποδεκτή μορφή συντακτικής τμηματοποίησης πριν το πλήρες parsing. Η αναδιάταξη της πρότασης γίνεται με κανόνες βασισμένους στη συντακτική πληροφορία (POS και φράσεις) και όχι στην σημασιολογία. Δεν χρησιμοποιείται Context-Free Grammars, dependency trees ή semantic role labeling καθώς είναι πιο σύνθετα στάδια συντακτικής και σημασιολογικής ανάλυσης. Η

συνολική προσέγγιση είναι rule-based, ερμηνεύσιμη και αναπαράξιμη, βασικό πλεονέκτημα ενός πρώιμου σταδίου της NLP ανάλυσης.

Grammatical_correction.py

Το στάδιο grammatical_correction υλοποιείται ως μια επιφανειακή κανονικοποίηση κειμένου (surface-level grammatical normalization) και όχι ως μια πλήρη γραμματική ανάλυση, γραμματική ή σημασιολογική ανάλυση. Τοποθετείται μετά την συντακτική ανακατασκευή, ακολουθώντας μια λογική post-processing για βελτίωση της εξόδου. Η ορθογραφική διόρθωση εφαρμόζεται μέσω κανόνων (μείωση μορφολογικών παραλλαγών) και όχι από στατιστικά ή μαθησιακά μοντέλα, γεγονός που την καθιστά κανονικοποίηση (όπως περιγράφεται στο βιβλίο [2]). Οι κανόνες αξιοποιούν τις ετικέτες POS ως μέρος του λόγου για καθαρισμό και εξομάλυνση αδόκιμων ακολουθιών λέξεων. Επίσης υπάρχουν κανόνες για την αφαίρεση διπλότυπων λέξεων, πλεονασμών και “ορφανών” ουσιαστικών. Τέλος το στάδιο του post-processing (του προγράμματος) περιλαμβάνει την τελευταία μορφοποίηση του τελικού κειμένου όπως αφαίρεση κενών, σημεία στίξης, πρώτο γράμμα κεφαλαίο.

Παραδοτέο 1Β

Για το Παραδοτέο 1Β ζητείται στην εκφώνηση η δημιουργία 3 διαφορετικών pipeline για την επεξεργασία των 2 κειμένων. Η αρχιτεκτονική που ακολουθήθηκε είναι παρόμοια με το πρώτο ερώτημα με την μόνη διαφορά ότι εδώ η main μπορεί να διαβάσει και “γράψει” τα κείμενα σε ξεχωριστά txt αρχεία που βρίσκονται σε διαφορετικούς υποφακέλους, ώστε να αποθηκεύσουμε τα αποτελέσματα. Επίσης η main συνάρτηση που περιέχουν τα pipelines είναι παρόμοια: τυπώνει το αρχικό κείμενο και το ανακατασκευασμένο και ανακατευθύνει για την συνάρτηση που είναι υπεύθυνη για την δημιουργία του reconstructed οπότε δεν θα αναλυθεί.

Οι τρεις προσεγγίσεις επιλέχθηκαν ώστε να καλύψουν διαφορετικά επίπεδα NLP επεξεργασίας: επιφανειακή γλωσσική ανάλυση (TextBlob), σημασιολογική αναπαράσταση μέσω διανυσμάτων (embeddings) και βαθιά συμφραζόμενη ανακατασκευή μέσω transformer. Με τον τρόπο αυτό, το ίδιο πρόβλημα αντιμετωπίζεται με διαφορετικές υπολογιστικές φιλοσοφίες, επιτρέποντας ουσιαστική σύγκριση των μεθόδων.

Pipeline 1: TextBlob Library

Το πρώτο αυτόματο χρησιμοποιεί την βιβλιοθήκη TextBlob που αναφέρεται πολύ συχνά στο βιβλίο [2]. Η συνάρτηση που κατευθύνει τις υπόλοιπες για την δημιουργία του

ανακατασκευασμένου κειμένου είναι η reconstruct_text_with_textblob, δημιουργεί το αντικείμενο textblob και στη συνέχεια με αυτόματο sentence segmentation μηχανισμό της βιβλιοθήκης εφαρμόζει ανακατασκευή ανά πρόταση ώστε να διατηρείται η τοπική συνοχή του κειμένου. Πριν την επιστροφή του στην main οι προτάσεις ενώνονται μεταξύ τους μαζί με κενά ώστε να έχουμε ένα ενιαίο κείμενο. Όπως αναφέρθηκε, η υλοποίηση γίνεται σε κάθε μεμονωμένη πρόταση. Στην _reconstruct_sentence() εφαρμόζεται ορθογραφική διόρθωση από το TextBlob, εξάγονται τα γλωσσικά χαρακτηριστικά και πραγματοποιείται επιφανειακή αναδιοργάνωση της πρότασης με βάση τα στατιστικά γλωσσικά χαρακτηριστικά. Στην _reorganize_by_pos υλοποιείται επιφανειακή ομαδοποίηση με βάση ετικέτες POS από το TextBlob, κατηγοριοποιούνται τα tokens, με στόχο την απλοποίηση και εξομάλυνση της πρότασης χωρίς μια πλήρης συντακτική αναδόμηση. Η αναδιοργάνωση λειτουργεί ως heuristic sentence simplification, σύμφωνη με επιφανειακές τεχνικές NLP που περιγράφονται στο βιβλίο [1]. Στο τέλος εφαρμόζεται ένα string-level post-processing στο ανακατασκευασμένο κείμενο ομοίως με το Α ερώτημα.

Pipeline 2: NLTK + Embeddings

Στη δεύτερη προσέγγιση, η ανακατασκευή κειμένου βασίστηκε στη χρήση προεκπαιδευμένων word embeddings, με στόχο τη σημασιολογική επεξεργασία του κειμένου. Το αρχικό κείμενο διασπάστηκε σε προτάσεις και λέξεις, ενώ εφαρμόστηκε Part-of-Speech tagging ώστε να εντοπιστούν οι λέξεις περιεχομένου (ουσιαστικά, ρήματα, επίθετα και επιρρήματα). Για κάθε λέξη περιεχομένου αναζητήθηκαν σημασιολογικά παρόμοιες λέξεις στον διανυσματικό χώρο των embeddings, χρησιμοποιώντας μέτρο σημασιολογικής ομοιότητας. Η αντικατάσταση πραγματοποιήθηκε μόνο όταν η ομοιότητα ξεπερνούσε προκαθορισμένο κατώφλι, ώστε να διατηρείται το αρχικό νόημα. Η τελική ανακατασκευή προέκυψε από την επανασύνθεση των λέξεων σε προτάσεις, χωρίς την εφαρμογή ρητών συντακτικών ή γραμματικών κανόνων.

Pipeline 3: Transformer

Στην τρίτη προσέγγιση, η ανακατασκευή κειμένου διατυπώθηκε ως πρόβλημα text-to-text μετασχηματισμού και υλοποιήθηκε μέσω προεκπαιδευμένου encoder–decoder transformer μοντέλου. Το αρχικό κείμενο δόθηκε απευθείας ως είσοδος στο μοντέλο, το οποίο επεξεργάστηκε την πληροφορία χρησιμοποιώντας μηχανισμό προσοχής, επιτρέποντας την κατανόηση του πλήρους συμφραζομένου κάθε πρότασης. Ο encoder παρήγαγε συμφραζόμενες αναπαραστάσεις του κειμένου, ενώ ο decoder δημιούργησε νέα εκδοχή του κειμένου token-by-token, με στόχο τη βελτίωση της γραμματικής ορθότητας, της συνοχής και της σαφήνειας. Η προσέγγιση δεν βασίστηκε σε χειροκίνητους κανόνες ή επιφανειακές αντικαταστάσεις λέξεων,

αλλά σε ενδογενώς μαθημένες γλωσσικές σχέσεις, ενώ εφαρμόστηκε μόνο ελάχιστο post-processing για μορφοποίηση της τελικής εξόδου.

Μεθοδολογία Παραδοτέου 2

Πειράματα και Αποτελέσματα

Παραδοτέο 1Α

Για την αξιολόγηση της προτεινόμενης προσέγγισης επιλέχθηκαν δύο προτάσεις που παρουσίαζαν γραμματικές ασυνέπειες και μειωμένη σαφήνεια. Σε κάθε πρόταση εφαρμόστηκε αυτόματη διαδικασία ανακατασκευής μέσω ενός rule-based pipeline, σχεδιασμένου να λειτουργεί χωρίς τη χρήση προεκπαιδευμένων μοντέλων ή σημασιολογικών αναπαραστάσεων. Η διαδικασία περιλάμβανε αρχικά στάδιο προεπεξεργασίας κειμένου, το οποίο κάλυπτε την τμηματοποίηση (tokenization) και την κανονικοποίηση (normalization) των λέξεων. Στη συνέχεια πραγματοποιήθηκε Part-of-Speech tagging, ενώ η τελική ανακατασκευή βασίστηκε στην αναδιοργάνωση της πρότασης σύμφωνα με απλά συντακτικά πρότυπα. Η όλη διαδικασία εφαρμόστηκε ανεξάρτητα για κάθε πρόταση, χωρίς αλληλεξάρτηση ή μεταφορά πληροφορίας μεταξύ των παραδειγμάτων.

Παραδοτέο 1Β – TextBlob

Στο πρώτο pipeline, το αρχικό κείμενο υποβλήθηκε σε αυτόματη επεξεργασία μέσω βιβλιοθήκης NLP που συνδυάζει στατιστικές και rule-based τεχνικές. Η διαδικασία περιλάμβανε λειτουργίες ορθογραφικής διόρθωσης, Part-of-Speech tagging και εξαγωγής noun phrases. Η ανακατασκευή των προτάσεων πραγματοποιήθηκε χωρίς τη χρήση χειροκίνητα ορισμένων κανόνων ή εκπαίδευσης μοντέλου, βασιζόμενη αποκλειστικά στις ενσωματωμένες δυνατότητες της βιβλιοθήκης.

Παραδοτέο 1Β – Embeddings

Στο δεύτερο pipeline, το κείμενο αναλύθηκε σε επίπεδο λέξεων και επισημάνθηκε με Part-of-Speech tags. Για τη σημασιολογική αναπαράσταση των λέξεων χρησιμοποιήθηκαν προεκπαιδευμένα word embeddings. Η διαδικασία ανακατασκευής περιλάμβανε

αντικατάσταση λέξεων περιεχομένου με σημασιολογικά συγγενείς λέξεις, βάσει προκαθορισμένου κατωφλίου ομοιότητας. Δεν εφαρμόστηκαν γραμματικοί ή συντακτικοί κανόνες κατά τη φάση της ανακατασκευής.

Παραδοτέο 1Β – Transformer

Στο τρίτο pipeline, το αρχικό κείμενο εισήχθη απευθείας σε pretrained encoder–decoder transformer μοντέλο. Η διαδικασία ανακατασκευής διατυπώθηκε ως πρόβλημα text-to-text generation, επιτρέποντας στο μοντέλο να παράγει νέα διατύπωση της πρότασης. Ο μηχανισμός προσοχής του transformer επέτρεψε την επεξεργασία του πλήρους συμφραζομένου κάθε πρότασης, χωρίς την εφαρμογή ρητών γραμματικών ή συντακτικών κανόνων.

Συζήτηση

Σύγκριση rule-based και στατιστικών προσεγγίσεων

Η rule-based προσέγγιση που εφαρμόστηκε στο Παραδοτέο 1Α οδήγησε σε αισθητή βελτίωση της γραμματικής μορφής των προτάσεων, ωστόσο τα αποτελέσματά της περιορίστηκαν από την απουσία μηχανισμών σημασιολογικής κατανόησης. Η αποτελεσματικότητα της μεθόδου εξαρτήθηκε σε μεγάλο βαθμό από τη δομή και την ποιότητα της αρχικής πρότασης, γεγονός που την καθιστά ευαίσθητη σε περιπτώσεις ασάφειας ή συντακτικής πολυπλοκότητας. Οι rule-based τεχνικές αποδείχθηκαν κατάλληλες κυρίως για απλές και σχετικά καλοδιατυπωμένες προτάσεις, ενώ παρουσίασαν σαφείς αδυναμίες κατά την επεξεργασία πιο σύνθετων ή αδόμητων γλωσσικών σχημάτων.

Επίδραση της σημασιολογικής αναπαράστασης (Embeddings)

Η εισαγωγή word embeddings στην ανακατασκευή κειμένου επέτρεψε τη διατήρηση του νοήματος μέσω σημασιολογικής εγγύτητας μεταξύ λέξεων, ανεξάρτητα από τη γραμματική τους μορφή. Οι προτάσεις που προέκυψαν μέσω αυτής της προσέγγισης εμφάνισαν αυξημένη λεξιλογική ποικιλία σε σύγκριση με τις rule-based μεθόδους, γεγονός που υποδηλώνει μεγαλύτερη ευελιξία στη λεξική επιλογή. Παρ’ όλα αυτά, η απουσία μοντελοποίησης συμφραζομένων και συνολικής δομής οδήγησε σε ασυνέπειες στη ροή και στη συντακτική συνοχή των προτάσεων. Τα ευρήματα αυτά επιβεβαιώνουν ότι η σημασιολογική ομοιότητα σε επίπεδο λέξεων δεν επαρκεί από μόνη της για την ολοκληρωμένη ανακατασκευή προτάσεων.

Πλεονεκτήματα της transformer-based προσέγγισης

Η transformer-based προσέγγιση παρουσίασε τη μεγαλύτερη συνοχή και γραμματική ορθότητα μεταξύ των εξεταζόμενων μεθόδων. Ο μηχανισμός προσοχής επέτρεψε στο μοντέλο να λαμβάνει υπόψη μακρινές εξαρτήσεις και σύνθετα συμφραζόμενα, διευκολύνοντας την κατανόηση της συνολικής δομής της πρότασης. Η ανακατασκευή πραγματοποιήθηκε σε επίπεδο πρότασης και όχι μεμονωμένων λέξεων, γεγονός που οδήγησε σε πιο φυσική και συνεπή αναδιατύπωση. Τα αποτελέσματα υποδεικνύουν ότι η text-to-text προσέγγιση είναι ιδιαίτερα κατάλληλη για εργασίες αναδιατύπωσης και ανακατασκευής φυσικής γλώσσας.

Περιορισμοί της μελέτης

Η παρούσα μελέτη παρουσιάζει ορισμένους περιορισμούς που πρέπει να ληφθούν υπόψη κατά την ερμηνεία των αποτελεσμάτων. Το σύνολο των κειμένων που χρησιμοποιήθηκε ήταν περιορισμένο, γεγονός που δεν επιτρέπει τη γενίκευση των συμπερασμάτων. Επιπλέον, δεν πραγματοποιήθηκε εκπαίδευση ή fine-tuning των μοντέλων, ενώ η αξιολόγηση βασίστηκε κυρίως σε σημασιολογική ομοιότητα και ποιοτική παρατήρηση. Τέλος, οι παράμετροι των pipelines δεν βελτιστοποιήθηκαν εκτενώς, γεγονός που ενδέχεται να επηρέασε την απόδοση των μεθόδων.

Συμπεράσματα

Η σύγκριση των διαφορετικών προσεγγίσεων ανέδειξε μια σαφή μετάβαση από επιφανειακή σε συμφραζόμενη επεξεργασία φυσικής γλώσσας. Κάθε μέθοδος παρουσίασε συγκεκριμένα πλεονεκτήματα και περιορισμούς, ανάλογα με το επίπεδο ανάλυσης και πολυπλοκότητας που υποστηρίζει. Συνολικά, τα αποτελέσματα επιβεβαιώνουν ότι πιο σύνθετα μοντέλα οδηγούν σε ποιοτικότερη ανακατασκευή κειμένου, με το αντίστοιχο όμως αυξημένο υπολογιστικό κόστος.

Βιβλιογραφία

[1] A. Kulkarni and A. Shivananda, “Natural Language Processing Recipes: Unlocking Text Data with Machine Learning and Deep Learning using Python,” Apress, 2019.

[2] S. Bird, E. Klein, and E. Loper, “Natural Language Processing with Python,” O’Reilly Media, 2009.