UNIVERSITY OF ATHENS
DEPARTMENT OF INFORMATICS AND TELECOMMUNICATIONS

Deep Learning for NLP

Student name: *<Andreas-Symeon Frantzis>*
sdi: <sdi2100273>

Course: *Artificial Intelligence II (M138, M226, M262, M325)*
Semester: *Fall Semester 2023*

Contents

1. Abstract

Σε αυτήν την αναφορά, παρουσιάζεται ένα μοντέλο ανάλυσης συναισθήματος για tweets. Ο στόχος της εργασίας είναι η ταξινόμηση των tweets ως θετικά (1) ή αρνητικά (0), βασισμένη στο περιεχόμενό τους.
Για την υλοποίηση, χρησιμοποιήθηκε Logistic Regression ως ταξινομητής και TF-IDF ως τη μοναδική μέθοδο εξαγωγής χαρακτηριστικών. Η προσέγγισή μας περιλαμβάνει τα εξής βασικά βήματα:
Ανάλυση Δεδομένων (EDA) για την κατανόηση της δομής και της κατανομής του συνόλου δεδομένων.
Προεπεξεργασία Κειμένου, που περιλαμβάνει μετατροπή σε πεζά, αφαίρεση URL, hashtags και ειδικών χαρακτήρων.
Εξαγωγή Χαρακτηριστικών μέσω του TF-IDF vectorizer για τη μετατροπή των κειμένων σε αριθμητικές αναπαραστάσεις.
Εκπαίδευση και Αξιολόγηση Μοντέλου, όπου εκπαιδεύουμε ένα Logistic Regression μοντέλο και το αξιολογούμε χρησιμοποιώντας accuracy, precision, recall και F1-score.
Τα αποτελέσματα δείχνουν ότι το μοντέλο μας διακρίνει αποτελεσματικά τα θετικά και αρνητικά συναισθήματα, επιτυγχάνοντας ικανοποιητική απόδοση στο validation set.

2. Data processing and analysis

2.1. Pre-processing

Αυτό το στάδιο της εργασίας είναι ίσως και το σημαντικότερο βήμα για την βελτίωση της απόδοσης του μοντέλου. Επέλεξα τα συγκρεκριμένα βήματα:

Μετατροπή σε πεζά (lowercasing): Όλα τα κείμενα μετατράπηκαν σε πεζά γράμματα για να διασφαλιστεί η συνοχή στη μοντελοποίηση.
Αφαίρεση συνδέσμων (URLs): Οι διευθύνσεις ιστού (http://, www.) αφαιρέθηκαν, καθώς δεν παρέχουν χρήσιμη πληροφορία για την ανάλυση συναισθήματος.
Αφαίρεση χρηστών και hashtags (@user, hashtag): Οι αναφορές σε χρήστες και hashtags αφαιρέθηκαν, καθώς δεν συμβάλλουν άμεσα στον εντοπισμό του συναισθήματος.
Αφαίρεση ειδικών χαρακτήρων και σημείων στίξης: Διαγράψαμε σύμβολα όπως !, ?, . και αριθμούς, ώστε να διατηρηθεί μόνο το αλφαβητικό περιεχόμενο.
Tokenization: Χρησιμοποιήσαμε την word tokenize της nltk για τον διαχωρισμό των προτάσεων σε λέξεις.

2.2. Analysis

Για την κατανόηση των δεδομένων, πραγματοποιήθηκαν οι εξής αναλύσεις:
Κατανομή των ετικετών (Label Distribution):
Εξετάσαμε πόσα tweets ανήκουν σε κάθε κατηγορία (θετικά / αρνητικά) και οπτικοποιήσαμε την κατανομή τους με barplot.
Στατιστικά κειμένων:
Αναλύσαμε το μήκος των tweets και τον αριθμό λέξεων πριν την προεπεξεργασία. Χρησιμοποιήσαμε ιστογράμματα (histograms) για να δούμε την κατανομή τους. WordCloud Token Frequency:
Δημιουργήσαμε WordClouds για να δούμε τις πιο συχνές λέξεις στα θετικά και αρνητικά

tweets.
Υπολογίσαμε τη συχνότητα εμφάνισης των πιο κοινών λέξεων με Seaborn barplots.

2.3. Data partitioning for train, test and validation

Για την εκπαίδευση και αξιολόγηση του μοντέλου, τα δεδομένα ειναι χωρισμένα σε training, validation και test sets.
Train Set (60/100): Χρησιμοποιείται για την εκπαίδευση του μοντέλου.
Validation Set (20/100): Βοηθά στη ρύθμιση υπερπαραμέτρων και στον έλεγχο υπερεκπαίδευσης.
Test Set (20/100): Χρησιμοποιείται για την τελική αξιολόγηση του μοντέλου.

2.4. Vectorization

Για τη μετατροπή των κειμένων σε χαρακτηριστικά κατάλληλα για το μοντέλο, χρησιμοποιήσαμε TF-IDF (Term Frequency - Inverse Document Frequency), που μετρά τη σημαντικότητα των λέξεων σε σχέση με το σύνολο των κειμένων.

3. Algorithms and Experiments

3.1. Experiments

Για την αντιμετώπιση του προβλήματος της ταξινόμησης συναισθήματος, ακολουθήσαμε μια προσέγγιση βασισμένη σε Logistic Regression με χαρακτηριστικά TF-IDF. Παρακάτω περιγράφουμε τα πειράματα που πραγματοποιήσαμε για τη βελτίωση της απόδοσης του μοντέλου.

Trial				Score
1				0.76975
2				0.78348

Table 1: Trials

Πρακτικά έγιναν και άλλες προσπάθειες για την βελτίωση της απόδοσης του μοντέλου αλλά στον παραπάνω πίνακα περιγράφονται αυτές που είχαν το σημαντικότερο ρόλο σε όλη την διεξαγωγή της άσκησης. Επίσης είναι αυτές με την μεγαλύτερη διαφορά στο score. Η ουσιαστική διαφορά που βελτίωσε το score απο την πρώτη στην δευτερη προσπάθεια είναι πως στην δέυτερη προσπάθεια, στο βήμα του preprocessing δεν αφαιρέθηκαν τα stopwords. Η μη αφαίρεση των stopwords φαίνεται να βοηθάει στον υπολογισμό του συναισθήματος και να πετυχένουμε καλύτερα αποτελέσματα στο τελικό στάδιο.

3.2. Optimization techniques

Για να βελτιώσουμε την απόδοση του μοντέλου μας, εφαρμόσαμε διάφορες τεχνικές βελτιστοποίησης. Παρακάτω περιγράφουμε τις βασικές στρατηγικές που ακολουθήσαμε:

1. Χρήση του LBFGS Optimizer

Το Logistic Regression εκπαιδεύτηκε με τον LBFGS (Limited-memory Broyden–Fletcher–Goldfarb–Sha optimizer, έναν αλγόριθμο βελτιστοποίησης gradient descent που είναι αποτελεσματικός για μεγάλα σύνολα δεδομένων.

2. Vectorization - TF-IDF

Για τη μετατροπή των κειμένων σε αριθμητικά χαρακτηριστικά, χρησιμοποιήσαμε τη μέθοδο TF-IDF (Term Frequency - Inverse Document Frequency), η οποία ενισχύει τις σημαντικές λέξεις και μειώνει τη σημασία κοινών λέξεων.

2. Καμπύλη Roc

Η ROC Curve (Receiver Operating Characteristic) μας επιτρέπει να αξιολογήσουμε την ικανότητα διάκρισης του μοντέλου. Η περιοχή κάτω από την καμπύλη (AUC-ROC score) μετρά την ποιότητα της ταξινόμησης.

4. Confusion Matrix

Χρησιμοποιήσαμε Confusion Matrix που μας επιτρέπει να αναλύσουμε ποια λάθη κάνει το μοντέλο και να πετύχουμε ένα καλύτερο αποτέλεσμα.

4. Results and Overall Analysis

4.1. Results Analysis

Όσο αναφορά την άσκηση και τα δεδομένα που είχαμε στην διάθεση μας, παρατήρησα κάποια πράγματα που μου έκανα εντύπωση:
1) Τρέχωντας local το όλο project τα αποτελέσματα που έπερνα ήταν διαφορετικά(καλύτερα) απο το Kaggle.
2) Την ουσιώδης διαφορά στις καλύτερες αποδόσεις την έκανε η αφαίρεση των stopwords.

4.1.1. Best trial. Η καλύτερη προσπάθει μου έχει score: 0.78348 και ο τρόπος με τον οποίο εκπαιδεύω το μοντέλου δεν ξεφεύγει απο τις διαφάνειες του φροντηστηρίου.

5. Bibliography

<Διαφάνειες φροντηστηρίου και παρακολούθηση:>
<TfidfVectorizer — scikit-learn 1.6.1 documentation:> <More about TfidfVectorizer>