Trip Type Classification

Apostolos Vakalos

2021

Εισαγωγή

Η παρούσα εργασία βασίζεται σε έναν παλιό διαγωνισμό του Kaggle με στόχο την εκτίμηση των καταναλωτικών συνηθειών των πελατών της Walmart. Συγκεκριμένα καλούμαστε να αξιοποιήσουμε τα δεδομένα ενός dataset και να προβλέψουμε τον τύπο επίσκεψης ενός πελάτη σε κάποιο κατάστημα. Έχουμε λοιπόν ένα πρόβλημα classification πολλών κλάσεων.

Επεξήγηση Dataset

Το dataset αποτελέιται από 7 στήλες

TripType: Ο αριθμητικά κωδικοποιημένος τύπος επίσκεψης ενός πελάτη – Ground Truth, Prediction Goal

VisitNumber: Μοναδικό id για μια συγκεκριμένη επίσκεψη από έναν συγκεκριμένο πελάτη.

Weekday: Η ημέρα της εβδομάδας που πραγματοποιήθηκε η επίσκεψη.

Upc: (Universal product code) Ο μοναδικός κωδικός του προϊόντος που αγοράστηκε ή επιστράφηκε.

ScanCount: Η ποσότητα ενός συγκεκριμένου προϊόντος που αγοράστηκε ή επιστράφηκε.

DepartmentDescription: Σύντομη περιγραφή του τμήματος που ανήκε το προϊόν.

FinelineNumber: Αριθμητικά κωδικοποιημένη κατηγορία πιο συγκεκριμένη από το τμήμα.

Προ-επεξεργασία

Αρχικά εντοπίστηκαν οι διπλότυπες γραμμές και οι γραμμές με ελλιπή πληροφορία (Null, NaN).

Κάποιες γραμμές διαγράφηκαν ως διπλότυπες ή λόγω μεγάλης έλλειψης πληροφορίας. Στις γραμμές με μερική έλλειψη πληροφορίας συμπληρώθηκαν οι άγνωστες τιμές με τις πιο συχνές τιμές ανάλογα με την περίπτωση.

Το feature FinelineNumber κρίθηκε αρκετά ασαφές στην πρωταρχική του μορφή για αυτό συμπεριλήφθηκε μόνο η πιο γενική κατηγοριοποίηση από το DepartmentDescription.

Επειδή τα δεδομένα είναι οργανωμένα ανά προϊόντα και εμείς θέλουμε να κάνουμε την πρόβλεψη του τύπου ανά επίσκεψη, θα πρέπει να οργανώσουμε εκ νέου την πληροφορία σε ένα νέο dataset όπου κάθε γραμμή θα αντιστοιχεί σε μία ξεχωριστή επίσκεψη. Έτσι, οι στήλες του νέου dataset είναι:

VisitNumber: Ίδιο με το αρχικό dataset.

Weekday : Η μέρα της επίσκεψης αλλά αριθμητικά κωδικοποιημένη.

Items Count: Το πλήθος των αντικειμένων που αγοράστηκαν στη συγκεκριμένη επίσκεψη

Unique Items: Πόσα ξεχωριστά προϊόντα αγοράστηκαν ή επεστράφησαν.

Επιπλέον, η ενσωμάτωση της πληροφορίας για το τμήμα των προϊόντων ανά γραμμή απαιτεί την προσθήκη ισάριθμων στηλών με όλα τα πιθανά departments των προϊόντων που αγοράστηκαν ή επεστράφησαν, με τιμή το πλήθος των προϊόντων που αντιστοιχούν στο εκάστοτε τμήμα.

Μπορεί να αυξάνουμε τη διάσταση των δεδομένων μας αλλά με αυτό τον τρόπο αποφεύγουμε την σημαντική απώλεια πληροφορίας και βοηθάμε τα μοντέλα ταξινόμησης να αποδώσουν καλύτερα.

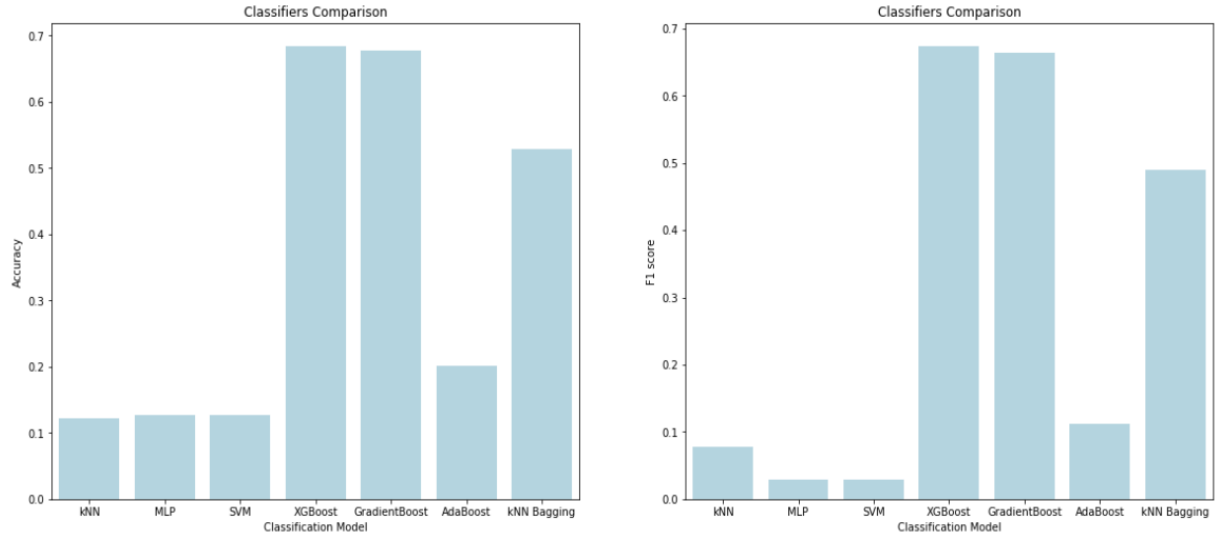
Μοντελοποίηση

Αρχικά έγινε χρήση του πιο απλού μοντέλου ταξινόμησης του K-Nearest Neighbors. Επίσης έγινε έρευνα και σε επίπεδο ensemble μεθόδων με τον Bagging Classifier που συνδυάζει αδύναμους KNN Classifiers.

Μετά δοκιμάστηκε το νευρωνικό δίκτυο MLP με δύο κρυμμένα επίπεδα και 200 νευρώνες/επίπεδο και ο SVM.

Η ανάλυση προχώρησε σε εξερεύνηση της απόδοσης των ensemble μεθόδων GradientBoost, AdaBoost και XGBoost.

Τα αποτελέσματα για την ακρίβεια και το F1 Score φαίνονται στην παρακάτω εικόνα:



Παρατηρήσεις:

- Παρατηρούμε ότι ο K-Nearest Neighbors δεν είναι αποτελεσματικός στη συγκεκριμένη εφαρμογή αδυνατώντας να βρει σωστά ομοιότητες ανάμεσα στα σημεία των δεδομένων. Η απόδοσή του όμως βελτιώνεται αρκετά μέσω του Bagging Classifier σαν αποτέλεσμα της αξιοποίησης των διαφορετικών υποσυνόλων χαρακτηριστικών που εξετάζονται με αποτέλεσμα να έχουμε συνολικά καλύτερη απόδοση.

- Ο SVM αδυνατεί επίσης να φέρει κατάλληλα διαχωριστικά όρια ανάμεσα στις κατηγορίες πιθανότατα λόγω της ασυμμετρίας των κατηγοριών.

- Το νευρωνικό δίκτυο φαίνεται να έχει ελαφρώς καλύτερη απόδοση αλλά όχι αρκετή ώστε να είναι αποδεκτή.

- Οι ensemble μέθοδοι φαίνεται να είναι περισσότερο κατάλληλες για τα συγκεκριμένα δεδομένα εκτελώντας βελτιστοποίηση επιμέρους αδύναμων εκτιμητών με αποτέλεσμα μια αρκετά καλύτερη απόδοση συνολικά. Η μέθοδος Ada Boost έχει αρκετά χαμηλή απόδοση συγκριτικά με τις υπόλοιπες ensemble methods. Η τεχνική Gradient Boosting φαίνεται αρκετά αποδοτική όπως και η πιο εξελιγμένη Extreme Gradient Boosting που πετυχαίνει τη συνολικά καλύτερη απόδοση (accuracy & f1) από όλα τα μοντέλα που εξετάστηκαν και με αρκετά μεγάλη ταχύτητα σε σχέση με τα υπόλοιπα μοντέλα.

Σύνοψη

Η μέγιστη επίδοση που επιτεύχθηκε δεν ήταν αρκετά υψηλή αντικειμενικά και αυτό πιθανόν να οφείλεται στα παρακάτω:

- Προ-επεξεργασία των δεδομένων
- Ανομοιόμορφη κατανομή των δεδομένων εισόδου στις κλάσεις. Αυτός είναι ένας σημαντικός παράγοντας που επηρεάζει τα περισσότερα μοντέλα.
- Έλλειψη οργανωμένου Hyperparameter Tuning.