Projet de traitement de données massives

Les données

Pour ce projet, vous aurez accès à des données collectées avec l'API Graph de Facebook. Il s'agit de publications postées par le journal Le Soleil ainsi que les commentaires que les utilisateurs FB ont écrits en réponse. Les données sont en trois fichiers. Le fichier *Post.csv* contient les publications du journal. Chaque tuple a les attributs suivants :

Attribut	Type de données	Description
attachments.data	JSON	Le contenu joint avec la publication.
created_time	Timestamp	Horodatage de la publication.
id	String	Un numéro unique pour la publication. Cet attribut est référencé par le PostID des commentaires.
mainTopic	String	Le sujet primaire de la publication, assigné par Le Soleil.
message	String	Le message de la publication.
permalink_url	URL	Un lien vers la publication.
secondTopic	String	Le sujet secondaire de la publication, assigné par Le Soleil.
shares	Int	Le nombre de partages de la publication.

Les deux autres fichiers, *Comments.csv* et *Test.csv*, contienent les commentaires des utilisateurs. Chaque tuple a les attributs suivants :

Attribut	Type de données	Description
IDENTITY_ATTACK	Float	Le score pour six catégories de toxicité pour ce commentaire, tel qu'assigné par l'outil Perspective (https://perspectiveapi.com/).
INSULT	Float	
PROFANITY	Float	
SEVERE_TOXIC	Float	
THREAT	Float	
TOXICITY	Float	
comment_count	Int	Le nombre de commentaires en réponse à celui-ci. Cet attribut est masqué pour les données tests.
created_time	Timestamp	Horodatage du commentaire .
id	String	Un numéro unique pour le commentaire
like_count	Int	Le nombre de likes du commentaire.
message	String	Le contenu du commentaire.
parent	String	Le id du commentaire parent, auquel celui-ci répond. Cet attribut est masqué pour les données tests.
postID	String	Le id de la publication sous laquelle le commentaire a été posté.

Le défi

Votre projet consiste à prédire quels commentaires des utilisateurs recevront des réponses. Quoique les données que vous avez permettent de connaitre le nombre exact de réponses, pour ce défi vous n'avez qu'à faire une prédiction binaire, pour savoir si le commentaire aura au moins une réponse ou non.

Questionnaire (1%)

Un questionnaire est disponible sur le site web du cours. Il pose une série de questions qui vous pousseront à explorer les données, la documentation des données, et Pandas (la librairie Python de traitement de données que je vous recommande d'utiliser pour ce projet). Ce questionnaire constitue donc un point de départ pour lancer votre travail. Il s'agit de la seule composante individuelle dans votre note de projet.

Question de ne pas vous stresser, ce questionnaire n'a pas de durée limite, et vous pouvez le faire un nombre illimité de tentatives. Le meilleur résultat sera conservé. Vous avez bien entendu le droit de consulter des ressources en ligne (particulièrement la documentation de Pandas) et de collaborer avec d'autres étudiants (particulièrement ceux de votre équipe).

Vous avez jusqu'à la date de remise du premier rapport pour compléter le questionnaire.

Premier rapport : Analyse et prétraitement des données (10%)

L'objectif de la première partie du projet est de vous familiariser avec les données avec lesquelles vous allez travailler.

Analysez vos données et leurs propriétés statistiques. Portez attention autant aux valeurs normales qu'aux cas problématiques, comme la présence de bruit, le fléau de dimensionnalité, les informations manquantes, le déséquilibre des classes, les valeurs aberrantes, etc. Discutez de vos observations. L'objectif ici n'est pas de faire une grande liste de statistiques sur les données, mais d'en tirer des leçons pour guider la réalisation du projet. (3 points)

Prévoyez les attributs que vous allez utiliser pour votre algorithme de traitement de données. Il y a une immense variété d'attributs qui peuvent être obtenus de ces données, incluant des attributs linguistiques (ex. : le contenu des commentaires), des attributs numériques (ex. : les scores Perspective), des attributs graphes (ex. : les connexions entre les publications, les commentaires parents et les commentaires enfants), et des méta-attributs (ex. : l'horodatage et le nombre de likes des commentaires). Vous pouvez également enrichir les données, tant avec le contenu des publications originales que des sources externes (des outils de traitement du langage par exemple). Vous devez prévoir les premiers attributs sur lesquels vous allez vous concentrer (il vous est bien entendu possible d'en rajouter à n'importe quel moment au cours de la session). Justifiez votre choix d'attributs initiaux. (3 points)

Prévoyez comment vous allez traiter les données d'un point de vue pratique. C'est-à-dire premièrement les algorithmes que vous allez implémenter et comment ils vont manipuler ces données, mais aussi l'optimisation de ceux-ci afin de pouvoir traiter la quantité massive de données disponible pour ce projet de manière efficace. (2 points)

Discutez également de la procédure de tests que vous envisagez. Vous ne pouvez pas tester votre système avec les commentaires dans le fichier test, puisque vous ne savez pas lesquels ont obtenus des réponses[1]. Vous devez donc prévoir votre propre procédure de tests afin de savoir si chaque variation de votre solution que vous implémentez améliore ou non vos prédictions, et ainsi guider votre travail de développement. (2 points)

Ce rapport est dû le 20 mars 2024. Les rapports en retard recevront automatiquement la note de 0.

Deuxième rapport : traitement des données (10%)

Pour ce rapport, vous devez présenter les algorithmes de traitement de données que vous avez implémentés, leur fonctionnement et les résultats que vous avez obtenu. Je m'attends que vous ayez un processus de développement itératif : implémentez un système, testez-le, découvrez ses points faibles, et raffinez-le en conséquences (en ajoutant des attributs, en modifiant l'entraînement, en corrigeant l'algorithme, etc.).

Décrivez chacun des algorithmes que vous avez choisi d'implanter et/ou les variations de ceux-ci. Décrivez, d'un point de vue technique, comment ils fonctionnent et les composantes clés. Le but ici n'est pas de répéter les notions de base des algorithmes que je vous ai enseigné au cours de la session, mais plutôt d'expliquer comment vous avez adapté et utilisé ces algorithmes pour ce projet, et comment vous les avez variés d'une expérience à l'autre. Justifiez vos choix pour les décisions de design et d'implémentation que vous avez pris. (2 points)

Décrivez également les tests que vous avez faits. Pour chaque test, présentez les résultats attendus et les résultats obtenus. Présentez des statistiques pertinentes (taux de succès, précision, rappel, temps moyen de calcul, complexité algorithmique, etc.). Discutez des leçons que vous avez prises de chaque test et comment elles ont guidé votre travail. (2 points)

Présentez des études de cas (i.e. des commentaires) comme exemples spécifiques du fonctionnement de votre algorithme. Décrivez autant les cas qui fonctionnent bien que ceux pour lesquels le test échoue, et discutez des raisons pour cette différence. (2 point)

Présentez la version finale de votre système, avec une attention particulière aux données et au traitement de données. Quels attributs des données sont utilisés par votre algorithme, quels

[1] Puisqu'il s'agit de commentaires publics sur Facebook, vous pouvez les retrouver en ligne et observer vous-même lesquels ont obtenu des réponses. Mais ce serait de faire un sur-apprentissage de votre système pour le corpus de test, ce qui est une mauvaise pratique et sera pénalisé.

attributs ont une valeur prédictive plus importante, et pourquoi? Quel prétraitement a été nécessaire pour bien exploiter les données? (3 points)

Offrez une rétrospective sur le projet. Comparé à vos réflexions au début de la session, en quoi avez-vous eu raison, et quelles surprises avez-vous eu en chemin? Si le projet était à refaire, que feriez-vous différemment? (1 point)

Ce rapport est dû le 17 avril 2024. Les rapports en retard recevront automatiquement la note de 0.

Évaluation des résultats (2%)

En même temps que le deuxième rapport, vous devez soumettre votre prédiction des commentaires qui obtiendront des réponses dans le fichier test.

Soumettez votre prédiction sous la forme suivante d'un fichier texte sans mise en forme. Chaque ligne de ce fichier est le id d'un commentaire ayant eu des réponses (les commentaires n'ayant pas eu de réponses ne sont pas inclus). Aucune autre information n'a besoin d'être inclue. L'ordre des commentaires n'est pas important. Un exemple de fichier de soumission est inclus sur le site web du projet.

Votre évaluation sera l'exactitude de vos résultats excluant les vrais négatifs (VN) :

$$Eval = \frac{VP}{VP + FP + FN}$$

Évaluation des rapports

Les rapports sont limités à 12 pages, incluant les figures, tableaux et références, doivent suivre le format de la conférence Canadian AI, disponible ici :

https://www.caiac.ca/en/conferences/canadianai-2021/call-papers

Les rapports seront remis en-ligne à travers le site web du cours. Une seule soumission par équipe. Les rapports doivent être écrits en Word ou LaTeX (pas de rapports écrits à la main) et soumis en format PDF (pas de fichiers Word ou texte).

La majorité des points du rapport seront donnés sur l'analyse et la discussion de votre système et de vos résultats. Il est donc important (pour vous) d'écrire une analyse approfondie et scientifique. La question centrale n'est donc pas « qu'est-ce qui se produit », mais « pourquoi est-ce que ça se produit » et « qu'est-ce qu'on peut y faire ». Vous ne devez pas simplement écrire un algorithme et générer des résultats. Vous devez être en mesure de justifier vos décisions qui ont mené à votre algorithme, et expliquer pourquoi il a généré ces résultats.

Un exemple peut clarifier les choses. Supposons que vos tests démontrent que votre algorithme génère un grand nombre de faux positifs (de commentaires sans réponse qui sont prédit d'avoir des réponses). Vous pouvez rapporter ce résultat de plusieurs manières :

- « Notre algorithme surestime le nombre de commentaires qui auront des réponses. » Ceci n'est pas une analyse, mais simplement une observation des faits. Les points donnés seront minimaux.
- « Notre algorithme surestime le nombre de commentaires qui auront des réponses. Ceci est probablement dû au fait qu'il surévalue le nombre de réponses que certains commentaires auront. » Ceci est l'inverse d'une analyse utile. Je ne donne pas de points, et je me réserve le droit de rire de vous.
- « Notre algorithme surestime le nombre de commentaires qui auront des réponses. Ceci est probablement dû au fait qu'il prédit que tout commentaire contenant un nom propre aura une réponse, alors qu'en réalité ce sont seulement les commentaires contenant le nom propre d'une personnalité publique qui obtiennent des réponses. » Vous avez identifié et analysé le problème et découvert sa source, bien joué! Vous avez des points.
- « Notre algorithme surestime le nombre de commentaires qui auront des réponses. Ceci est probablement dû au fait qu'il prédit que tout commentaire contenant un nom propre aura une réponse, alors qu'en réalité ce sont seulement les commentaires contenant le nom propre d'une personnalité publique qui obtiennent des réponses. Une solution possible serait d'incorporer une liste de noms de personnalités publiques. » Non seulement vous avez découvert la source du problème, mais vous l'avez comprise assez bien pour proposer une solution, c'est fantastique. Vous aurez une bonne note.
- « Notre algorithme surestime le nombre de commentaires qui auront des réponses. Ceci est probablement dû au fait qu'il prédit que tout commentaire contenant un nom propre aura une réponse, alors qu'en réalité ce sont seulement les commentaires contenant le nom propre d'une personnalité publique qui obtiennent des réponses. Une solution possible serait d'incorporer une liste de noms de personnalités publiques, mais cette solution sera peu flexible. Une meilleure option serait d'extraire les noms propres présents de l'article original afin de les reconnaitre dans les commentaires. C'est l'option que nous avons choisie d'appliquer. » Vous avez identifié le problème, vous l'avez analysé pour trouver sa source, puis vous avez exploré plusieurs pistes de solutions et justifié votre choix d'une en particulier. C'est parfait. 100%.

Notez finalement que jusqu'à 10% des points d'un rapport peuvent être enlevés en pénalité pour un rapport de mauvaise qualité. Ceci inclut particulièrement une abondance de fautes d'orthographe et de grammaire, des figures mal préparées (ou dessinées à la main), les rapports écrits à la main, le non-respect du format et de la longueur maximale, et les textes incohérents.

Équipes

Le projet doit être réalisé en équipes de 3 étudiants. La note sera donnée pour l'équipe, et non par individu, sauf dans des situations jugées extrêmes par moi-même. Choisissez bien vos coéquipiers.

Plagiat

Le plagiat est une offense académique sérieuse. Tout étudiant qui tente de soumettre un travail qui n'est pas le sien sera pénalisé. Ceci inclut de copier le travail ou rapport d'un autre étudiant du cours ou un système trouvé ailleurs. Un étudiant coupable de plagiat recevra automatiquement la note de

zéro pour le projet entier (c'est-à-dire toutes les parties) et s'exposera à d'autres sanctions telles que décidées par l'Université.

Conseils

- Il y a plusieurs solutions possibles pour ce projet. Quoiqu'un algorithme de classification supervisé qui associe chaque commentaire à une classe (avec réponse ou sans réponse) semble naturel, on pourrait aussi faire un partitionnement de l'espace de commentaires, ou encore créer un algorithme de recommandation qui suggère des commentaires attractifs aux réponses. Il y a aussi une structure et une temporalité aux données, alors vous pourriez les traiter comme un graphe de commentaires ou un flux de commentaires. Tous ces sujets sont couverts dans des cours à venir, et le matériel du cours est disponible pour vous permettre de prendre de l'avance et d'explorer ces pistes de solution.
- Essayez plusieurs de vos idées et parlez-en dans vos rapports. Décrivez quelle est l'idée, pourquoi vous pensez que c'est intéressant à essayer (qu'est-ce que vous voulez découvrir ou que pensez-vous va arriver), et quel est le résultat obtenu (est-ce celui que vous attendiez, et sinon pourquoi). À force de réfléchir et d'expérimenter, vous trouverez une bonne solution. Et ce n'est pas mauvais que plusieurs de vos idées ne fonctionnent pas; c'est la nature même de la recherche! De plus, ça justifie que la version finale de votre système est la meilleure, et non simplement la première que vous avez essayé. Pour l'évaluation, je donne des points pour les explorations intéressantes (à condition qu'elles soient bien présentées, justifiées, et analysées, bien entendu). Je ne donnerai pas de points pour des idées farfelues ou mal présentées. Mais par contre je n'enlèverai jamais de points pour avoir essayé quelque chose. Et en contrepartie, si vous ne décrivez pas vos idées et expériences dans votre rapport, je ne peux pas vous donner de points du tout.
- Considérez les extrêmes logiques de vos idées. Par exemple, si augmenter le poids d'un attribut améliore les résultats, pourquoi ne pas l'augmenter encore plus, ou ne conserver que cette variable? Ce sera rarement le bon choix, mais d'explorer le comportement de votre système dans les cas extrêmes peut souvent aider à mieux comprendre le problème et à développer une nouvelle intuition pour sa solution.
- Justifiez votre analyse avec des démonstrations mathématiques lorsque possible.
- Ne soumettez pas un copier-coller de votre code au complet dans votre rapport. Expliquez comment votre algorithme fonctionne en utilisant des descriptions du processus et des étapes, la logique du système, des formules mathématiques, et du pseudo-code.
- Je suis disponible durant mes heures de bureau pour vous aider en discutant de votre projet, des difficultés que vous rencontrez, et en suggérant des idées et des pistes. Je n'ai pas la solution du projet. Et je ne vais pas déboguer votre code pour vous.

Prix Pierre Ardouin

« *Depuis plusieurs années, le Département d'informatique et de génie logiciel a mis en place un concours récompensant l'équipe qui produit le meilleur TP/projet dans le cadre d'un cours. Ces travaux de session ont l'envergure d'un mini-projet qui est admissible par rapport aux normes fixées par le département. À*

la suite des évaluations des travaux, l'enseignant du cours détermine l'équipe gagnante; chaque membre de l'équipe gagnante reçoit alors un certificat-cadeau d'une valeur de 50$ ainsi qu'une attestation remise par le département à la fin de l'année universitaire.

De plus, le Département d'informatique et de génie logiciel a mis en place une prix élite, appelée Prix « Pierre Ardouin », qui vise à récompenser le meilleur projet, tous cours participants confondus. Deux principaux critères guident le choix des évaluateurs dans l'identification du lauréat: l'excellence du travail (par rapport à ce qui est demandé dans l'énoncé) et l'aspect créativité/innovation. Il est actuellement prévu un certificat-cadeau de 200$ pour récompenser chaque membre de l'équipe « élite » gagnante (pour un maximum de 1000$ pour toute l'équipe). Aussi, le département veille à publier l'information sur le site web.

Notez que depuis l'automne 2019, le concours récompense les meilleurs travaux d'une année (été, automne et hiver) selon que les cours participants sont de 1ère, de 2e année, de 3e/4e année, ou de 2e/3e cycle; ainsi, pour chacune de ces catégories de cours, un prix sera remis aux finalistes ainsi qu'aux lauréats du prix Pierre Ardouin (donc, potentiellement 4 équipes lauréates par année).

À la fin du mois de mai de chaque année universitaire, le département organise une cérémonie pour honorer les finalistes et les lauréats du prix « Pierre Ardouin » de l'année qui vient de s'écouler (été, automne et hiver), et pour leur remettre une attestation. Les attestations peuvent être transmises par courriel également à ceux et celles qui ne seraient pas présents à la cérémonie. »