

PROJET TECHNIQUE : Détection du cyber- harcèlement sur Twitter

Lisa Giordani
Adrien Grimberg
Erwan Tinen-Touolac
Oscar Jozon

Introduction

Présentation du projet

Présentation du système

Mise en relations des acteurs du système

Résultats du projet

Conclusion

Recherches documentaires

- **Etude d'articles scientifiques**
- **Interview de Serena Villata**
- **Etude de la loi sur le cyber-harcèlement**
- **Détermination des critères du cyber-harcèlement**

Exigences du projet

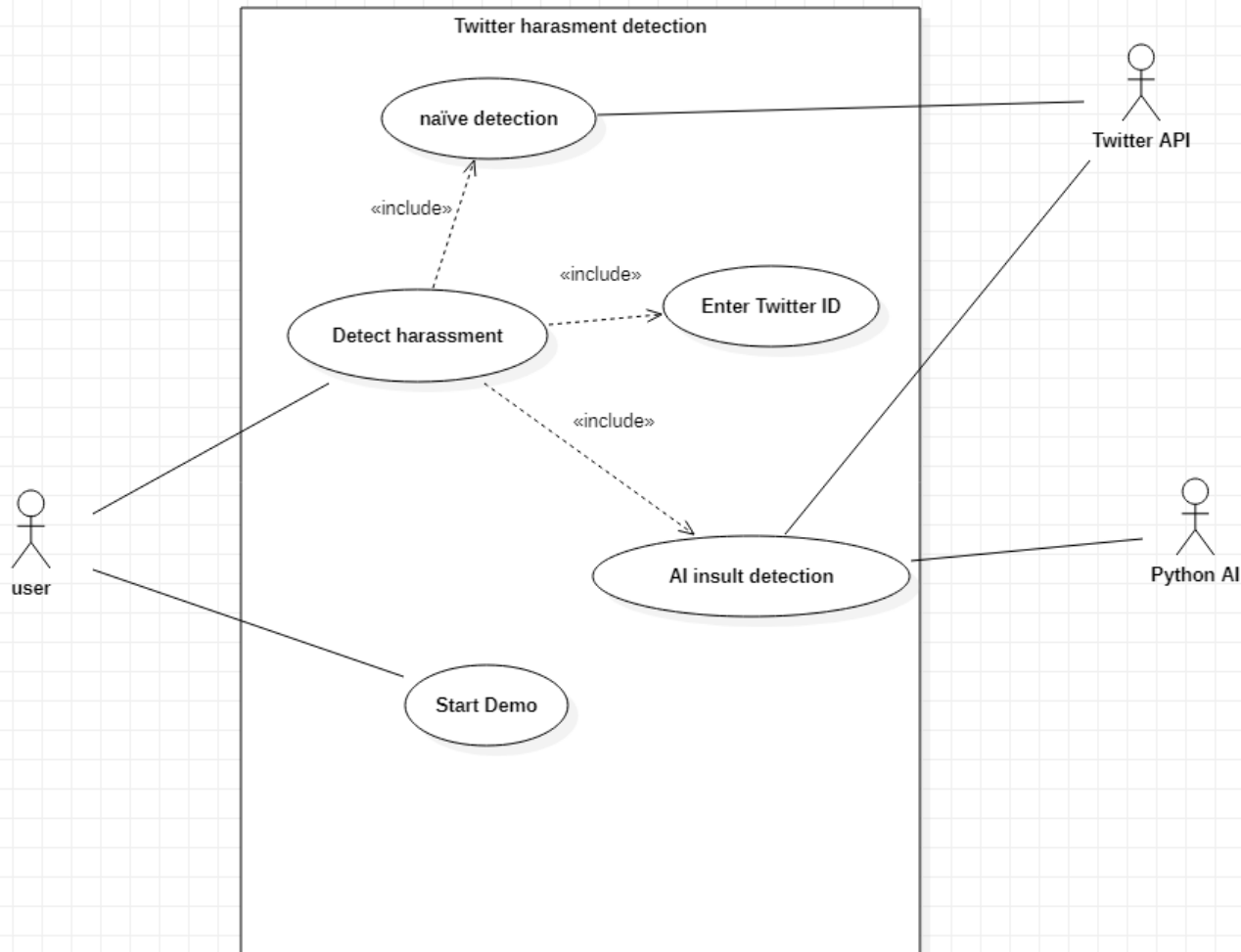
Exigences fonctionnelles :

- Choisir la méthode de détection des tweets insultants (IA ou algorithme naïf)
- Utiliser l'algorithme naïf
- Utiliser l'algorithme de deep learning
- Récupérer les tweets insultants adressés à un compte Twitter
- S'informer au sujet du cyber-harcèlement

Exigences non fonctionnelles :

- Utilisabilité
- Efficacité
- Confidentialité

Diagramme des cas d'utilisations



Algorithmes de détection du cyber-harcèlement

- **Algorithme naïf :**

Détection des tweets contenant au moins une insulte couramment utilisée

- **Algorithme de deep learning :**

Détecte les tweets pouvant témoigner d'un harcèlement grâce à entraînement sur une base de données de 100 000 tweets annotés

Acteurs du système

Twitter et son API

- Flux de tweets réduit : 100 maximum par requête
- Accès limité dans le temps : 8 jours à partir du jour de la requête
- Modération : faible nombre tweets insultants

Base de données MySQL

- Stockage des informations saisies par l'utilisateur via l'interface
- Stockage de données utiles au fonctionnement des algorithmes
- Stockage des résultats des algorithmes

Serveur web local

- Interface utilisateur

Introduction

Présentation du projet

Présentation du système

Mise en relations des acteurs du système

Résultats du projet

Conclusion

Algorithme naïf

```
/**
 * @Route("/generateNaif", name="main_generateNaif", methods={"GET"})
 */
public function generateNaif()
{
    $entityManager = $this->getDoctrine()->getManager();

    //////////////// Suppression des tweets de la table tweets
    $tweetController = new TweetController();
    $tweetController->deleteAll($entityManager);

    //////////////// Suppression des statistiques de la table statistics
    $statisticsController = new StatisticsController();
    $statisticsController->deleteAll($entityManager);

    //////////////// Récupération de l'id de la victime présumée
    $webRequestController = new WebRequestController();
    $recipientId = $webRequestController->getLastRecipientId($entityManager);

    //////////////// Connection à l'API Twitter
    $settings = array(
        'oauth_access_token' => "1308765160467828736-rJvp6amtRT8pRtxZ5kKPk1TM9ZIBXN",
        'oauth_access_token_secret' => "ELdg6TVrMd6FKX2TJK9XslV7jLp9Y33JKYn0eRQ84DuPc",
        'consumer_key' => "sadLxTJ4NjuGK8xnEbpcMFx2w",
        'consumer_secret' => "ijqDVtK6RWV6TNNRE2LVescq4XNjki5lM3MRVf8WtBDDCZldc7"
    );
    $url= "https://api.twitter.com/1.1/search/tweets.json";
    $twitterAPIconnection = new TwitterAPIConnection($settings, $url);
```

Introduction

Présentation du projet

Présentation du système

Mise en relations des acteurs du système

Résultats du projet

Conclusion

Algorithme naïf

```
////////// Création des 8 requêtes vers l'API Twitter correspondant aux tweets reçus au cours des 8 derniers jours
$NBTweetMax = 100;

$twitterAPIrequestController = new TwitterAPIRequestController();
$requests = $twitterAPIrequestController->get8Requests($NBTweetMax, $recipientId);

////////// Création de l'instance de Statistics associée à cette recherche
$stats = new Statistics($recipientId, date('Y-m-d'));

////////// Obtention des résultats des 8 requêtes vers l'API Twitter pour les 8 derniers jours
$twitterAPIcontroller = new TwitterAPIController();
$tweets = $twitterAPIcontroller->get8Results($twitterAPIconnection, $requests);

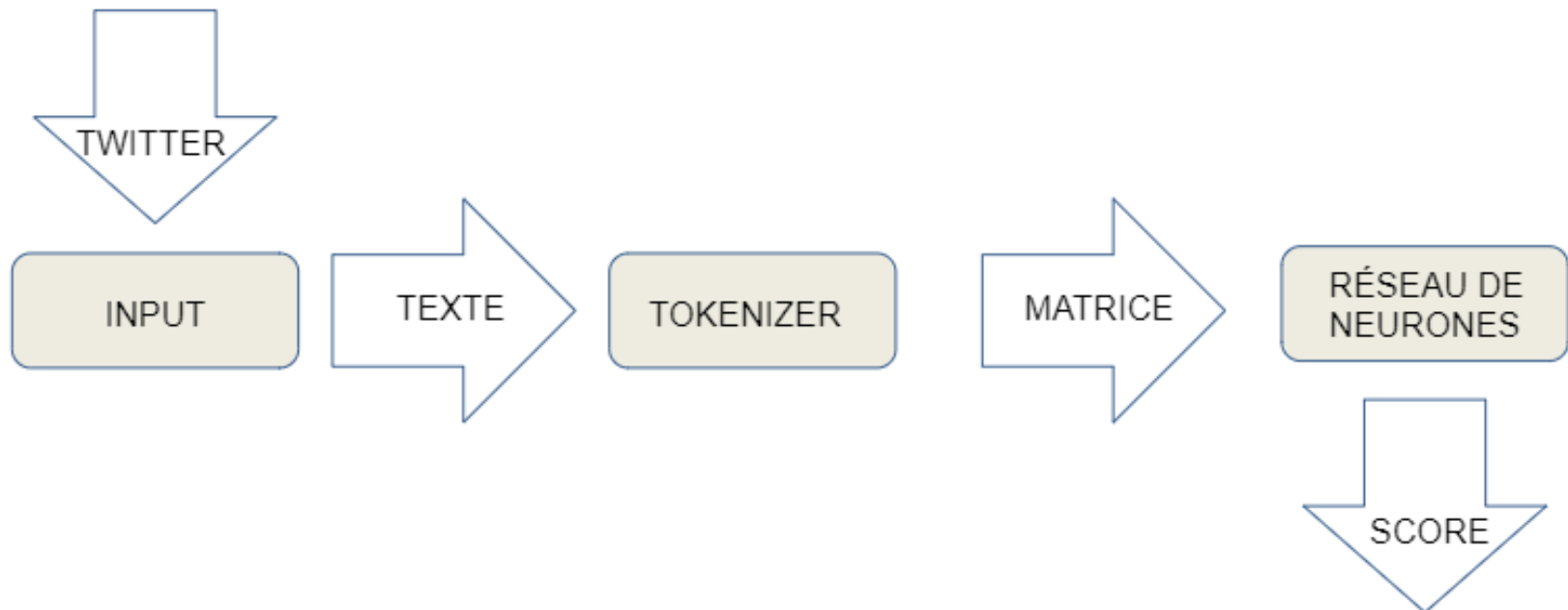
////////// Collecte les insultes de la table insults
$insultController = new InsultController();
$insults = $insultController->getInsults($entityManager);

////////// Détection et stockage des tweets possédant au moins une insulte
$bullyingController = new BullyingController();
$bullyingController->getBullying($tweets, $recipientId, $insults, array(), $entityManager, $stats);

////////// Mise à jour des statistiques et stockage dans la table statistics
$statisticsController = new StatisticsController();
$statisticsController->update($stats, $entityManager);

return $this->redirectToRoute('view_result');
}
```


Algorithme de deep learning



Introduction

Présentation du projet

Présentation du système

Mise en relations des acteurs du système

Résultats du projet

Conclusion

Interface utilisateur

PAGE D'ACCUEIL

LOI SUR LE CYBER-HARCÈLEMENT

ALGORITHME NAÏF

INTELLIGENCE ARTIFICIELLE

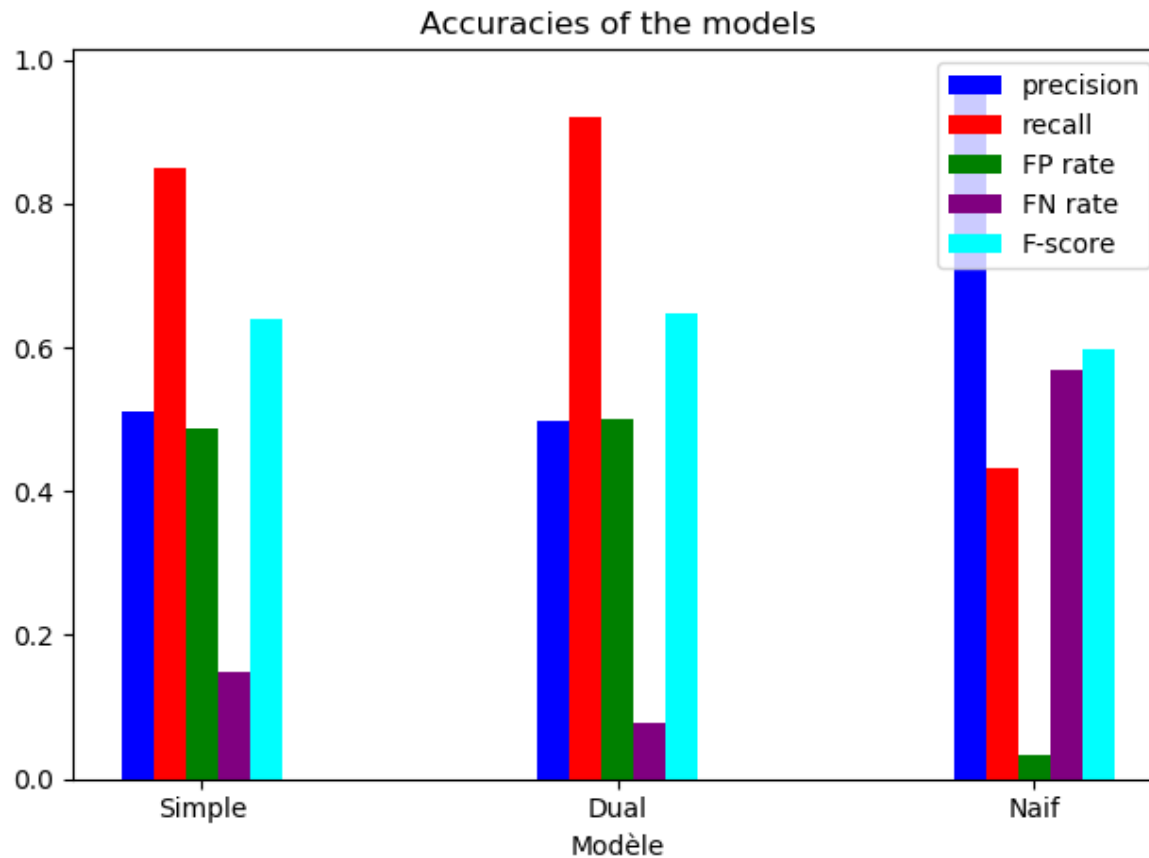
FAQ

CONTACTS

e-usticia

Détection du cyber-harcèlement sur Twitter

Comparaison des algorithmes



Points forts et faibles des algorithmes

	Naïf	Deep learning
Insulte couramment utilisée	oui	oui
Insulte à double sens utilisée dans son sens non insultant (ex : chien)	non	oui
Insulte couramment utilisée non dirigée vers la présumée victime	non	peu probable, car voué à être un cas marginal
Sous-entendu	non	oui
Ironie	non	oui
Humour	non	dépend largement de la taille du dataset (cas non marginal, mais rare)

Le projet remplit les objectifs fixés

- 2 algorithmes de détection du cyber-harcèlement : naïf et deep learning
- Mis en relation avec l'API Twitter, les bases de données MySQL et le serveur web
- Interface utilisateur

→ Pistes d'améliorations

Merci
de votre attention

Bibliographie (1)

- [1] Ying CHEN, Yilu ZHOU, Sencun ZHU et Heng XU. 2012. Detecting Offensive Language in Social Media to Protect Adolescent Online Safety, ASE/IEEE International Conference on Social Computing et ASE/IEEE International Conference on Privacy, Security, Risk and Trust
- [2] Michele Corazza, Stefano Menini, Elena Cabrio, Sara Tonelli, et Serena Villata. 2019. A Multilingual Evaluation for Online Hate Speech Detection. ACM Trans. Internet Technol. 1, 1, Article 1 (January 2019), 22 pages. <https://doi.org/10.1145/3377323>
- [3] Mai Ibrahim, Marwan Torki et Nagwa El-Makky. 2018. Imbalanced Toxic Comments Classification using Data Augmentation and Deep Learning. 17th IEEE International Conference on Machine Learning and Applications
- [4] Site du service public sur le cyberharcèlement : <https://www.service-public.fr/particuliers/vosdroits/F32239>
- [5] Article 222-33-2-2 du code pénal : https://www.legifrance.gouv.fr/codes/article_lc/LEGIARTI000037289658/2018-08-06

Bibliographie (2)

- [6] Trello du système : <https://trello.com/invite/b/F8o8Lmt1/a85029bfbec6246af31bcb321d85ed42/protech-syst%C3%A8me>
- [7] Trello des algorithmes : <https://trello.com/invite/b/r1JqjKb1/cf78e9a55cfef1e51bfe7abeaca8f1be/protech-algorithmes>
- [8] Base de données d'insultes trouvée sur Wiktionnaire : https://fr.wiktionary.org/wiki/Cat%C3%A9gorie:Insultes_en_fran%C3%A7ais
- [9] Base de données d'insultes annotée par catégorie d'insultes, Hurtlex : <https://github.com/valeribasile/hurtlex>
- [10] Base de données de tweets annotée sur laquelle l'intelligence artificielle a été entraînée : <https://github.com/Momotoculteur/Harassment-detector/tree/master/dataset>
- [11] Documentation utile pour programmer en PHP : <https://www.php.net/>
- [12] Documentation pour utiliser l'API Twitter : <https://developer.twitter.com/en/docs>
- [13] Documentation pour TwitterAPIExchange : <https://github.com/J7mbo/twitter-api-php>



Annexe 1 : Trello - système

Board: ProTech - Système | Private Team | Team Visible | AG ET G O Invite | Calendar | Butler | Show Menu

Idées

- Utilisateur = personne harcelée ?
- Catégoriser les insultes
- Pouvoir modifier la classification des insultes
- Score pour les insultes
- Etudier les API de Facebook et Instagram
- Corriger les fautes d'orthographe et de frappe

A faire

- Diagramme des cas d'utilisation
- Présentation pour la soutenance

En cours

- Création de l'interface utilisateur
- Trouver comment mettre à disposition le projet pour le client/tuteur
- Rédaction du dossier technique
- Rédaction d'une documentation pour le code
- Etudier l'interaction Php-Python et inclure l'IA au système

Vérification

- Compléter les informations qui seront disponibles sur le site web (onglets loi sur le cyber-harcèlement, FAQ, contacts)

Terminé

Date	Activités	Travail de terrain	Travail de terrain
8 sept 2020	Début de projet, L2B	Démarage projet	
1 octobre 2020	Réunion avec les directeurs de la faculté de sciences et de technologies (STI) pour la mise en place de la plateforme d'insultes	En accord avec le tuteur, direct et indirect, les membres du jury de la soutenance ont été informés de la mise en place de la plateforme d'insultes	Réunion avec les directeurs de la faculté de sciences et de technologies (STI) pour la mise en place de la plateforme d'insultes
1 janvier 2021	Présentation des résultats de la soutenance	Présentation des résultats de la soutenance	Présentation des résultats de la soutenance
1 janvier 2021	Rapport scientifique et technique	Rapport scientifique et technique	Rapport scientifique et technique

Dossier pour l'évaluation du 6 octobre

Justice sur le cyberharcèlement

Lire articles

Mail réunion tuteur 2

Création et data cleaning de la BDD insultes

Créations des BDD pour afficher les résultats des algorithmes

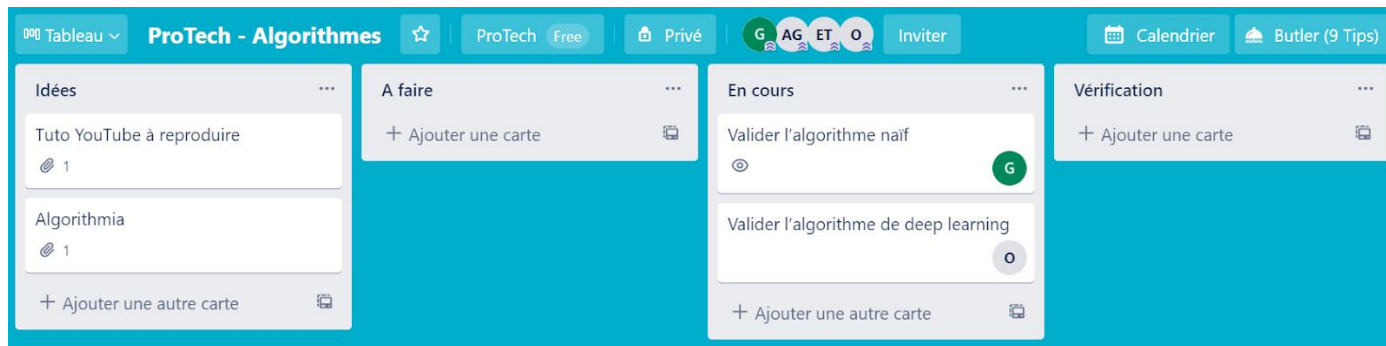
Création de la BDD servant à récupérer les informations de la requête (identifiant Twitter)

Création de la BDD servant à récupérer les informations de la requête (identifiant Twitter)

Connexion du système aux BDD MySQL

Création de la BDD de tweets annotés sur laquelle faire la démo et la validation des algorithmes

Annexe 2 : Trello - algorithmes



The screenshot shows a Trello board titled "ProTech - Algorithmes". The board is organized into four columns:

- Idées**: Contains two cards: "Tuto YouTube à reproduire" and "Algorithmia".
- A faire**: Contains one card: "+ Ajouter une carte".
- En cours**: Contains two cards: "Valider l'algorithme naïf" and "Valider l'algorithme de deep learning".
- Vérification**: Contains one card: "+ Ajouter une carte".

Each card has a small icon in the bottom right corner, likely representing a status or priority. The board also features a top navigation bar with various tools and a sidebar on the right.



The screenshot shows a Trello board titled "Terminé". The board contains a list of completed tasks:

- BD insultes V1
- Interview Serena et Elena
- Recherche Langage pour l'API (Symfony, Java, C, php)
- Étudier l'API Twitter
- [Pour la fin] Essayer de clean le JSON, ou renvoyer sous format différent text et nom d'utilisateur et url du tweet.
- Implémenter un algorithmes naïf
- Implémenter un algorithme de deep learning

Each task has a small icon in the bottom right corner, likely representing a status or priority. The board also features a top navigation bar with various tools and a sidebar on the right.

Annexe 3 : Formules utilisées lors de la validation

	1	0
true	VP	FP
false	FN	VN

$FP\ rate = FP / (VP + FP)$

$FN\ rate = FN / (VP + FN)$

$recall = VP / (VP + FN)$

$precision = VP / (VP + FP)$

$F\text{-score} = 2 * precision * recall / (precision + recall)$

Formules utilisées pour calculer les statistiques

Annexe 4 : Benchmarking

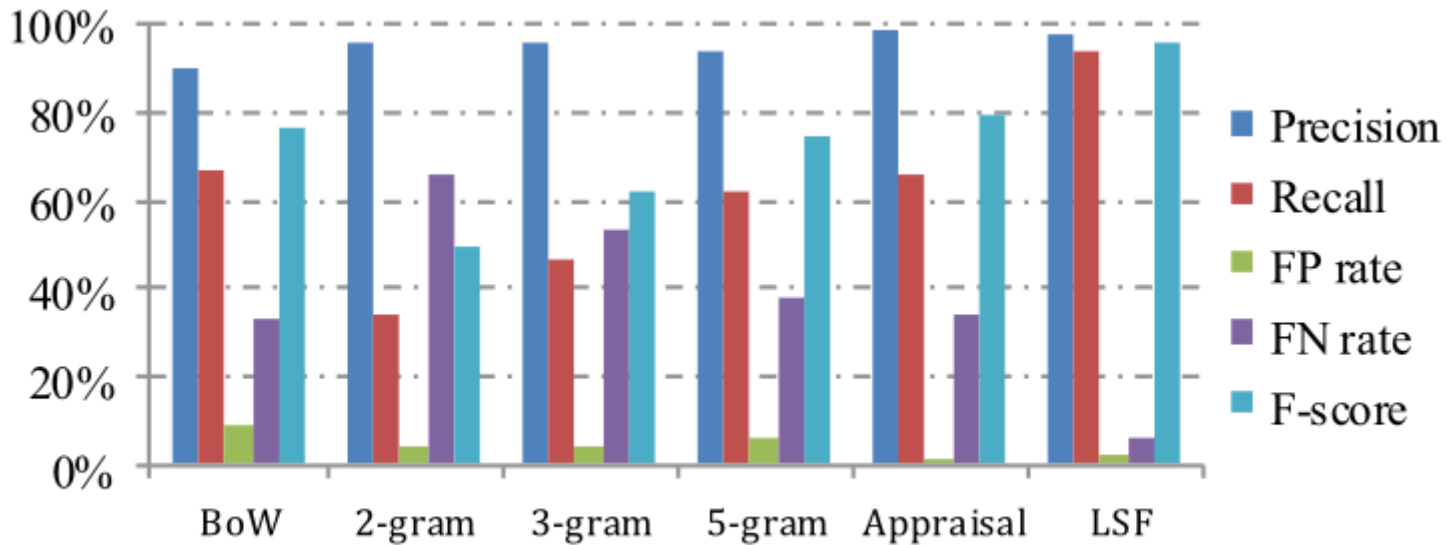


Figure 2. Accuracies of sentence level offensiveness detection

Detecting Offensive Language in Social Media to Protect Adolescent Online Safety,
Ying CHEN, Yilu ZHOU, Sencun ZHU et Heng XU. 2012