



IMT Atlantique
Bretagne-Pays de la Loire
École Mines-Télécom

Recette du projet



SPEECH QUALITY
ASSESSMENT DOCKER

SOMMAIRE



IMT Atlantique
Bretagne-Pays de la Loire
École Mines-Télécom

- 1. Cahier des charges**
 - 1.1 Besoin
 - 1.2 Réponse : NISQA
- 2. Solutions**
 - 2.1 Serveur Debian
 - 2.2 Docker
 - 2.3 Command Line Interface
- 3. Organisation**
 - 3.1 Déroulement du projet et problèmes
- 4. Résultats**
 - 4.1 Analyse des résultats
 - 4.2 Comparaison avec PESQ
- 5. La suite du projet**
 - 5.1 Amélioration de la CI
 - 5.2 Optimisation de la taille du Docker
- 6. Conclusion**





CHAPITRE 1

Cahier des charges




IMT Atlantique
Bretagne-Pays de la Loire
École Mines-Télécom

1.1 Besoin

-  Surveiller en temps réel la qualité des communication VoIP sur le réseau Orange
-  Recherche de pannes
-  Correction en temps réel
-  Remplacer l'évaluation actuelle de qualité (narrowband) pour l'adapter à tous les types de communication (voix HD et au-delà)

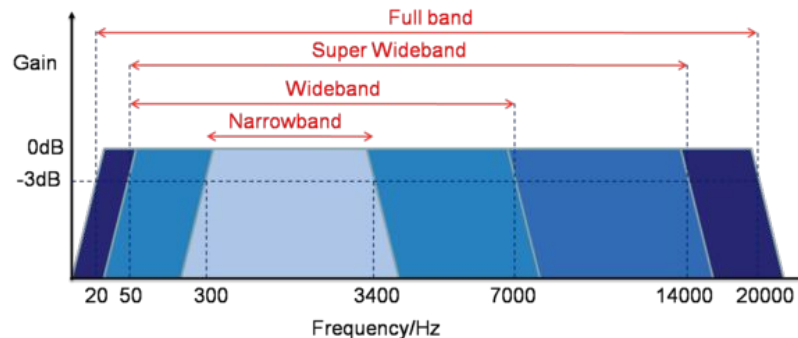


 NISQA modèle d'algorithme Deep Learning pour l'évaluation de la qualité audio

Le modèle a déjà été développé et optimisé, l'aspect Deep Learning de NISQA ne rentre pas dans notre projet

 Signaux audio super wideband jusqu'à 16 kHz

 Open Source, permettant un déploiement plus large



gabrielmittag/
NISQA



NISQA - Non-Intrusive Speech Quality and TTS
Naturalness Assessment



1
Contributor



0
Issues



279
Stars



64
Forks



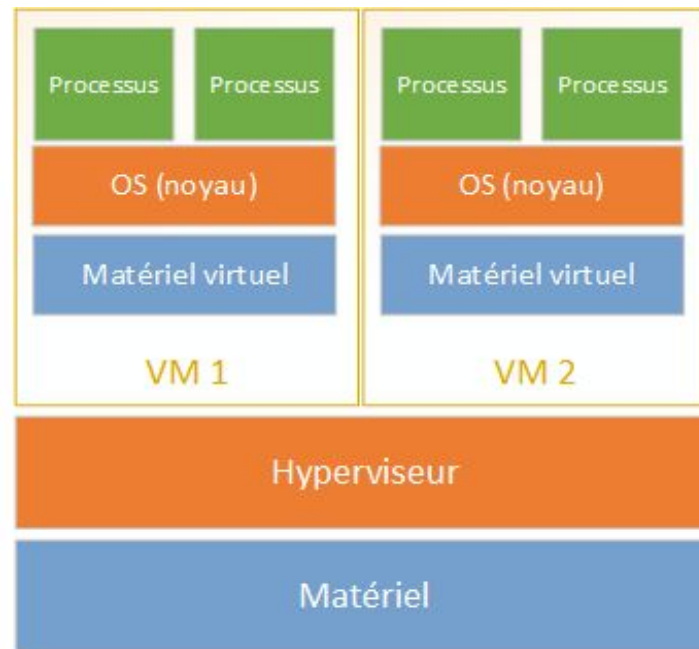
CHAPITRE 2

Solutions

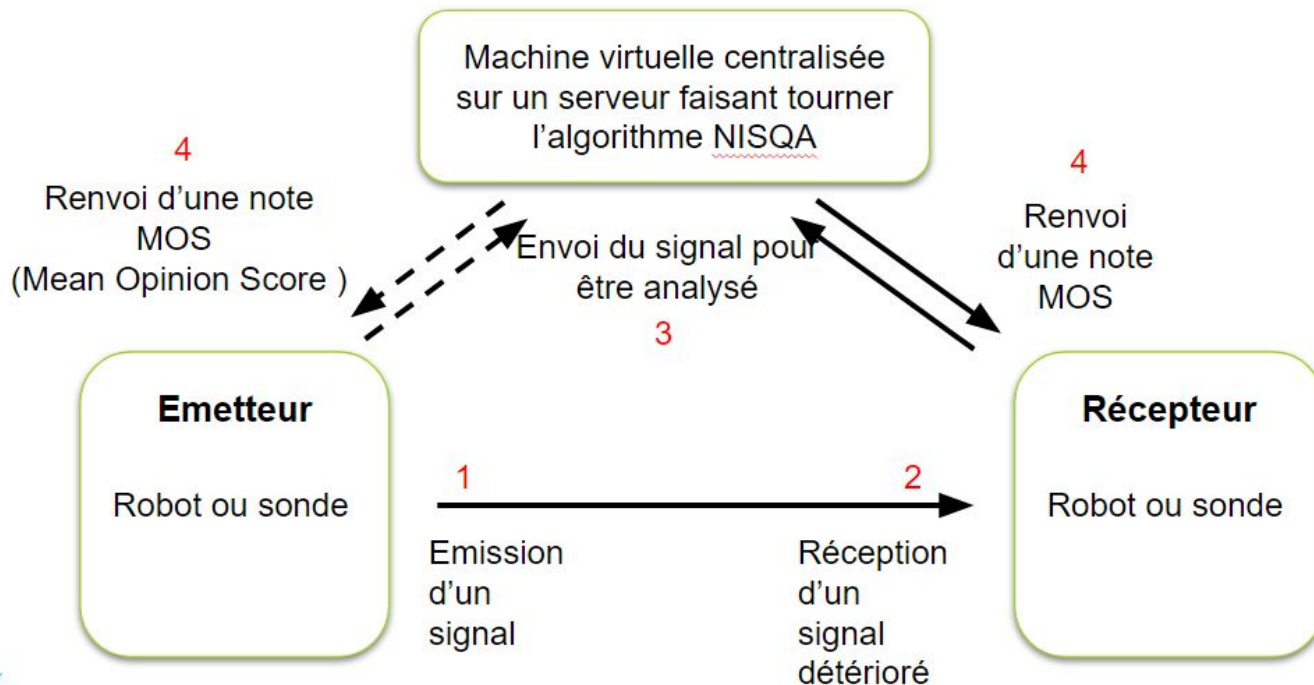



IMT Atlantique
Bretagne-Pays de la Loire
École Mines-Télécom

- 🎯 Implémenter NiSQA sous forme de logiciel debian
- 🎯 Installation dans une VM
- 🎯 Avoir une unité centralisée qui gère les flux de données.
- 🎯 Pas de changement de l'architecture du système déjà existant
- 🎯 Amélioration de la qualité de l'analyse
- 🎯 Cette solution est la plus facile à mettre en place pour un utilisateur déjà habitué aux process de Orange.



Machine virtuelle contenant
l'algorithme NiSQA



 Installation du .deb et des paquets de façon basique.

```
> sudo pip3 install -r requirements.txt  
> sudo apt-get install ./nisqa.deb
```

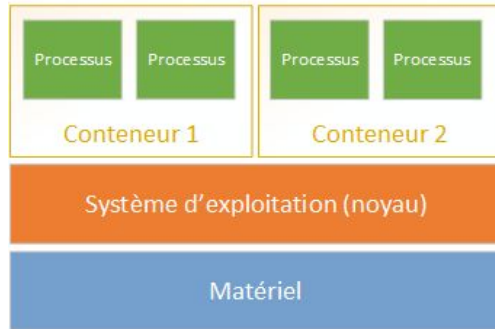
 Choix de connexion ou non au serveur

```
> sudo su - nisqa -c "rwd up"  
> sudo su - nisqa -c "rwd down"
```

 Envoi de l'audio à évaluer

```
> curl -H "Content-type: audio/vnd.wave" --data-binary  
@- http://127.0.0.1:8080/cgi-bin/nisqa.cgi < audio.wav
```

🎯 Pour plus de flexibilité et de portabilité de la solution apportée on utilise la conteneurisation via Docker.



Conteneur
Docker avec
l'algorithme
NISQA



```
1 FROM python:3.9-slim as python_base
2 COPY . /env/nisqa
3 RUN python3 -m venv /env/venv
4 # Make sure we use the virtualenv:
5 ENV PATH="/env/venv/bin:$PATH"
6 RUN python3 -m pip install /env/nisqa/lib_py/* \
7     && rm -rf /env/nisqa/lib_py
8
9
10 FROM python:3.9-slim
11 COPY --from=python_base /env /
12 ENV PATH="/venv/bin:$PATH"
13
14 RUN apt-get update \
15     && apt-get install -y ./nisqa/nisqa.deb \
16     && rm -rf ./nisqa/lib_deb \
17     && apt-get clean
18
19 EXPOSE 8080
20 CMD ["/nisqa/launch.sh"]
21 ENTRYPOINT ["/bin/bash"]
```

Extrait du Dockerfile

 Création de l'image

```
> sudo docker build . -t=nisqa
```

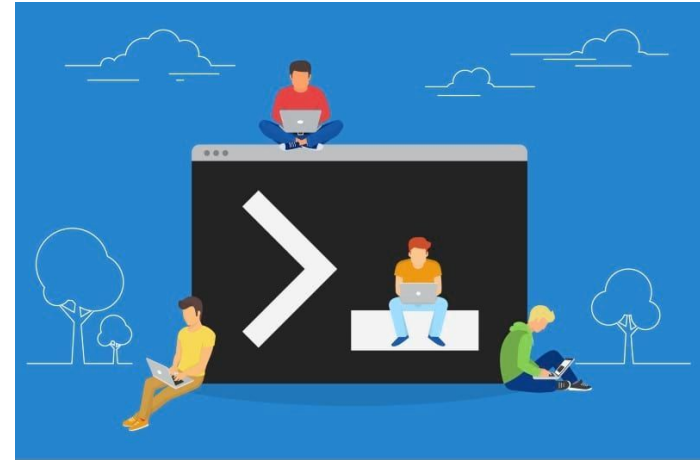
 Lancement du conteneur

```
> sudo docker run -d -p 8080:8080 --name nisqa nisqa
```

 Envoi de l'audio à évaluer

```
> curl -H "Content-type: audio/vnd.wave" --data-binary  
@- http://127.0.0.1:8080/cgi-bin/nisqa.cgi < audio.wav
```

- 🎯 Utilisation directe de NISQA
- 🎯 Architecture de logiciel debian
- 🎯 Simple et rapide d'utilisation





Installation du .deb et des paquets
de façon basique.

```
> sudo pip3 install -r requirements.txt  
> sudo apt-get install ./nisqacli.deb
```



Mesure simple et adaptable selon le
format

```
> su - nisqa -c "nisqacli" < audio.wav
```

CHAPITRE 3

Organisation



IMT Atlantique
Bretagne-Pays de la Loire
École Mines-Télécom

CHAPITRE 3 : Organisation

3.1 Déroulement du projet et problèmes

16

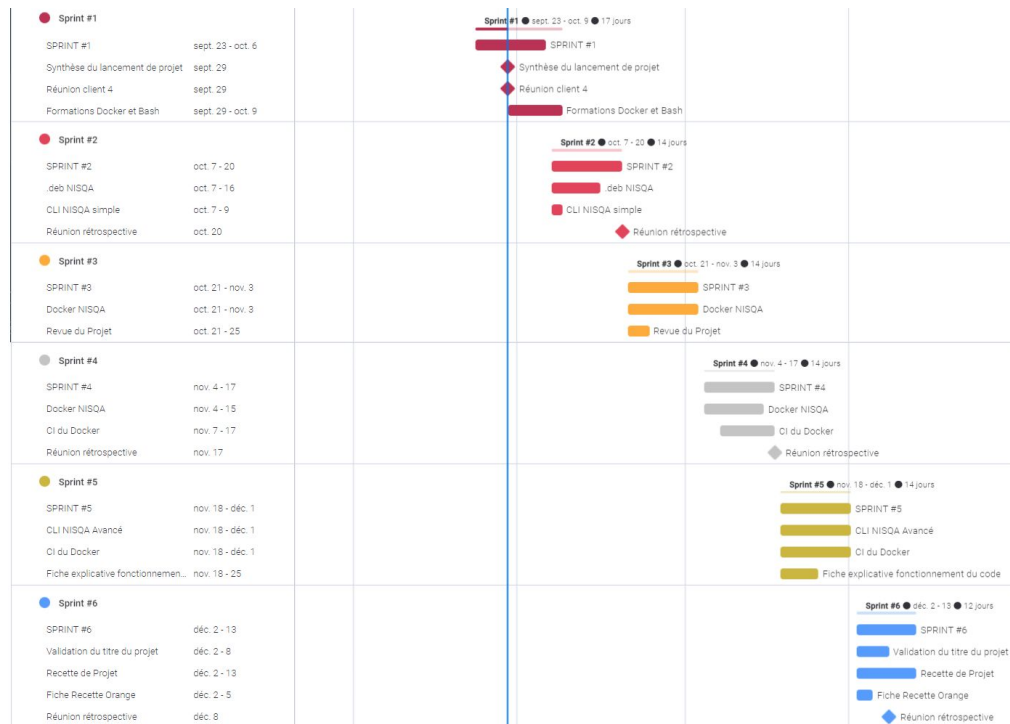




Diagramme de Gantt prévisionnel

 Formations internes et travail de développement en équipe

 Beaucoup de réunions et de contacts avec les clients qui ont été d'une grande aide

 Relation avec des personnes qui sont extérieures au projet pour nous aider

 Jalons plus ou moins respectés (cohérence des formations et du développement)

 Problème de la réduction de la taille du livrable Docker qui nécessite d'aller au delà de nos compétences et qui nous a pris du temps

CHAPITRE 4

Résultats







IMT Atlantique
Bretagne-Pays de la Loire
École Mines-Télécom

audio	MOS	Noisiness	Discontinuity	Coloration	Loudness	Description
F1S1.Co1	5.150409	4.817146	4.87177	4.702571	4.831268	référence haute, spectre complet sans codage
F1S2.Co2	2.764512	2.513579	4.70238	4.164081	3.836166	ajout de bruit de fond
F1S2.Co8	1.672902	4.20311	1.292345	3.776736	4.116467	20\% de paquets IP perdus
F1S2.C26	4.649351	4.513099	4.741065	4.046252	4.496233	codage de la voix HD en 3G
F1S3.C15	3.80061	4.142939	4.202243	2.986036	3.579714	codage du GSM
F1S4.Co4	3.486153	4.722647	4.715476	4.24715	1.821873	signal atténué de 20 dB
F1S5.Co5	3.193204	4.001684	4.506642	1.973342	3.656659	spectre ramené à 500-2500 Hz

Quelques résultats obtenues avec NISQA

Reference	Degraded	PESQ	Reference NISQA	Degegraded NISQA	Difference Reference - Degraded	Difference PESQ-NISQA
or105.wav	dg105.wav	1.84394	3.450691	1.495162	1.95553	0.34878
or109.wav	dg109.wav	3.09075	3.417776	1.828903	1.58887	1.26185
or114.wav	dg114.wav	1.75777	3.495709	1.357655	2.13805	0.40011
or129.wav	dg129.wav	2.36606	3.545906	1.521932	2.02397	0.84413
or134.wav	dg134.wav	1.97845	3.593792	1.44308	2.15071	0.53537
or137.wav	dg137.wav	3.77944	3.547771	2.858148	0.68962	0.92129
or145.wav	dg145.wav	2.84601	3.483449	2.534688	0.94876	0.31132
or149.wav	dg149.wav	2.20702	2.941261	1.341206	1.60006	0.86581
or152.wav	dg152.wav	2.48674	3.127168	2.060962	1.06621	0.42578
or154.wav	dg154.wav	2.38495	3.616052	1.189305	2.42675	1.19564

Quelques résultats PESQ et NISQA

NISQA	PESQ
 Les notes peuvent être légèrement supérieurs à 5 et légèrement inférieurs à 1	 Notes strictement comprises entre 1 et 5
 Notes MOS en général plus basse car il n'y a pas de comparaison avec le fichier de référence	 Notes MOS en général plus haute

CHAPITRE 5

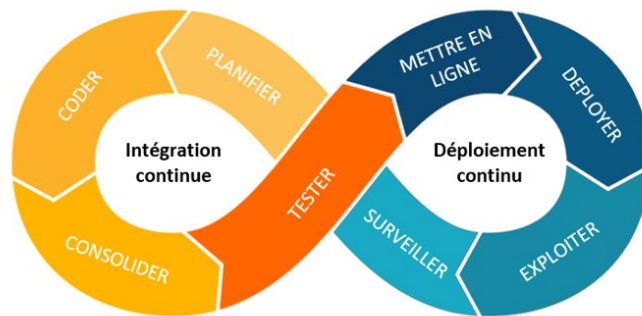
La suite du projet



IMT Atlantique
Bretagne-Pays de la Loire
École Mines-Télécom

🎯 Améliorer la sécurité du docker (créer un USER) et la sécurité des tests réalisés

🎯 Avoir une intégration continue semblable à celle des autres projets d'Orange



5.2 Optimisation de la taille du Docker

Problème : Taille des fichiers



Solutions : refaire l'architecture dans un autre langage plus léger

 Réécrire les fonctions pytorch bloquantes pour Onnx



Permet d'exporter le modèle pytorch en un modèle Onnx plus léger.

 Refonte de l'architecture sur Tensorflow



Utilisation d'un Docker avec une image de Tensorflow, lite réécriture de l'architecture sur Tensorflow lite, utilisation des poids.

 Refonte de l'architecture en C++








Utilisation d'un Docker avec image C/C++, réécriture de l'architecture avec Darknet, utilisation des poids pour inférence.

CHAPITRE 6

Conclusion



IMT Atlantique
Bretagne-Pays de la Loire
École Mines-Télécom

-  Logiciel Debian
-  Docker
-  Ligne de Commande
-  Guides d'utilisation
-  Fiche recette



6.2 Bilan

🎯 Beaucoup de nouvelles compétences et d'outils utilisés

🎯 Découverte du déroulement d'un projet en entreprise et en grande équipe

🎯 Très bonne expérience d'échanges élèves/entreprise

🎯 Satisfaction d'avoir produit un service utile



Merci



IMT Atlantique
Bretagne-Pays de la Loire
École Mines-Télécom