

Phase 1 – Analyse du besoin et étude bibliographique

Ce document rassemble la première phase du projet : compréhension du besoin, identification des facteurs influents sur la consommation des véhicules électriques (VE), définition des cas d'usage, et synthèse des approches et modèles existants (étude bibliographique). Il sert de base pour les phases suivantes (données, modélisation, validation, prototype).

1. Contexte et objectifs

Objectif général : - Comprendre les leviers et contraintes affectant la consommation énergétique des VE et définir des cas d'usage exploitables par des modèles IA / systèmes de recommandation.

Objectifs de la phase 1 : - Lister et prioriser les facteurs influençant la consommation. - Définir clairement les cas d'usage (priorités fonctionnelles). - Réaliser une revue bibliographique ciblée (modèles prédictifs, optimisation d'itinéraires, systèmes de recommandation, usages de GenAI). - Proposer métriques d'évaluation, sources de données et un plan d'étude / livrables.

2. Facteurs influents sur la consommation d'un VE

À collecter / prendre en compte dans les modèles :

- Profil cinématique
- Vitesse instantanée et moyenne
- Accélérations / décélérations (profil en fréquence et amplitude)
- Stop & go (nombre d'arrêts / reprises)
- Topographie et géométrie de la route
- Dénivelé (pente) et profil d'altitude
- Courbes, rayon (impact sur vitesse)
- Conditions ambiantes
- Température extérieure
- Vent (direction/intensité) si disponible
- Pluie/neige (adhérence)
- Confort & usages embarqués
- Chauffage / climatisation / ventilation (HVAC)
- Accessoires (phares, radio, etc.)
- Propriétés du véhicule
- Masse totale (charge, passagers)
- Aérodynamique (C_x), surface frontale
- Résistance au roulement (pneus, pression)
- Capacités batterie, état de santé (SoH), SoC initial
- Rendement régénération/freinage
- Infrastructure et trafic
- Type de route (urbain, périurbain, autoroute)
- Conditions de trafic (flux, embouteillages)
- Limitations de vitesse
- Style de conduite / comportement du conducteur
- Agressivité (fréquence d'accélérations => indicateurs)
- Anticipation (conduite douce vs brutale)

3. Cas d'usage (Use cases)

1. Prédition de la consommation énergétique
2. Prévision d'énergie (kWh) pour un trajet donné (sur trajectoire connue ou profil attendu).

3. Estimation de SoC restant en fonction d'un plan de route.
 4. Planification de trajets optimisés en énergie
 5. Itinéraire économisant l'énergie (plutôt que le temps ou la distance).
 6. Itinéraire multi-critère (temps vs énergie vs sécurité vs points de recharge).
 7. Intégration des bornes de recharge (choix des arrêts, SOC cible).
 8. Recommandations de conduite personnalisées (en temps réel ou a posteriori)
 9. Conseils pour réduire la consommation (vitesse, accélération, plages de récupération).
 10. Feedback personnalisé (score éco-conduite, actions correctives).
 11. Génération de scénarios et explications (usage GenAI)
 12. Aide à la décision basée sur explications naturelles (pourquoi un itinéraire est plus économique).
 13. Chatbot d'assistance pour l'utilisateur (explication des recommandations).
-

4. Panorama des approches

4.1 Approches physiques / heuristiques

- Modèles physiques basés sur la dynamique du véhicule : bilan puissance = résistance aérodynamique + roulement + pente + accélération + pertes (moteur, conversion).
- Avantage : explicables, robustes pour extrapolation.
- Limite : nécessite paramètres physiques précis et données détaillées (masse, Cx, rendement batterie).

4.2 Approches statistiques / machine learning classiques

- Régressions linéaires / polynomiales, SVR.
- Arbres : Random Forest, Gradient Boosting (XGBoost, LightGBM, CatBoost).
- k-NN, modèles basés sur histogrammes.
- Avantages : simples à mettre en œuvre, bons baselines; tolèrent bruit.
- Limites : capturent mal dépendances temporelles complexes ou séquences.

4.3 Approches séquentielles / deep learning

- RNN: LSTM, GRU — pour séries temporelles (profil vitesse -> consommation).
- TCN (Temporal Convolutional Networks), Transformer pour séries temporelles.
- Architectures hybrides (CNN + LSTM) pour extraire motifs de vitesse/accélération.
- Avantages : puissants pour captures de dynamiques et long context.
- Limites : besoin données volumineuses, risque d'overfitting.

4.4 Modèles graphiques et spatiaux

- Graph Neural Networks (GNN) pour intégrer topologie routière (nœuds segments).
- Usage pour prédire dépendance spatiale (ex : conditions de route adjacentes).

4.5 Approches "grey-box" / physiques + ML

- Combinaison de modèle physique avec correction ML (residual learning).
- Permettent meilleure généralisation et interprétabilité.

4.6 Optimisation d'itinéraires et routing économe en énergie

- Algorithmes classiques : Dijkstra, A*, variantes temps-dépendantes.
- Techniques avancées : Contraction Hierarchies, ALT, CH-Potentials pour accélérer la recherche.
- Multi-critère / multi-objectif : Pareto-optimal pour temps/énergie.
- Routing tenant compte du profil d'altitude, vitesse limite, trafic et consommation estimée (eco-routing).

4.7 Génération et usage de données synthétiques / GenAI

- Génération de profils routiers et de conduite synthétiques pour enrichir jeu de données.
 - Utilisation de LLM/GenAI pour :
 - Produire explications en langage naturel pour recommandations.
 - Aide à la génération d'interfaces conversationnelles (chatbot).
 - Synthèse et structuration automatique de rapports sur la consommation.
-

5. Données : sources, format attendu, enrichissements

Sources potentielles (à rechercher et valider) : - CAN bus (vitesse, RPM, courant/tension, SoC, HVAC states). - Traces GPS ou capteurs IMU (position, vitesse). - Profils altitude (DEM / SRTM) couplés à trajets (OpenStreetMap pour topologie). - Données météo (température, vent) historiques par timestamp/position. - Modèles trafic / données historiques (TMC, HERE, TomTom). - Jeux publics / rapports : rechercher NREL EV Project, TSDC (U.S. DOT Transportation Secure Data Center), jeux sur Kaggle (énergie / trajets), publications académiques partageant datasets, base de données CapGemini.

Format conseillé (par enregistrement de trajet) : -
Identifiant trajet, timestamps, lat/lon, altitude, vitesse,
acceleration, road_type, traffic_flag, SoC,
courant_batterie, tension_batterie, HVAC_on,
masse_estimée, température_ambiante.

Enrichissements possibles : - Calcul profil pente par
segments. - Extractions de features statistiques
(moyenne vitesse, énergie par km, nombre
d'accélérations > seuil). - Segmentation trajet (urbain /
interurbain / autoroute).

6. Métriques / critères d'évaluation

Pour prédiction de consommation : - MAE, RMSE (kWh
ou Wh/km) - MAPE (attention si petites valeurs) - R²
(explained variance) - Pour classification de risque ou
bins (ex : dépassement de seuil) : F1, AUC

Pour routing & économie d'énergie : - Économie d'énergie
absolue (%) vs itinéraire baseline - Impact sur temps de
trajet (trade-off) - Taux de réussite (arrivée sans
recharge) - Latence / complexité de calcul

Pour recommandations : - Acceptation utilisateur (A/B
testing), réduction mesurable de conso sur cohortes
pilotes - Interprétabilité / satisfaction (via
questionnaires ou logs)

7. Protocole expérimental proposé (validation)

1. Baseline simple : modèle physique + régression
linéaire sur features agrégées.
2. Baselines ML classiques : Random Forest /
XGBoost sur features extraites.
3. Approches séquentielles : LSTM / Transformer sur
séquences de vitesse/accélération.
4. Grey-box : physique + ML pour corriger résidus.

5. Évaluation croisée par trajets (leave-one-route-out / leave-one-vehicle-out) pour tester généralisation.
 6. Test d'itinéraires : comparer eco-routing vs shortest-time vs shortest-distance sur un ensemble de trajets réels/simulés (en fonction des jeux de données sélectionnées).
-

8. Outils & bibliothèques possibles

- Data processing : pandas, geopandas, shapely
 - Routage & cartes : OpenStreetMap, osmnx, OSRM, GraphHopper, HERE APIs (besoin licences)
 - Elevation / DEM : SRTM, AWS Terrain Tiles, Google Elevation API (besoin licences)
 - ML & DL : scikit-learn, XGBoost/LightGBM/ CatBoost, PyTorch / TensorFlow, PyTorch Lightning
 - GNN : PyTorch Geometric, DGL
 - Hyperopt / Optuna pour tuning
 - Evaluation & visualisation : mlflow, tensorboard, seaborn, folium/kepler
-

9. Exemples de références et mots-clés recherché

Mots-clés : - "electric vehicle energy consumption prediction" - "eco-routing", "energy-aware routing", "eco-driving recommendation" - "residual modeling vehicle energy", "grey-box vehicle model" - "LSTM energy consumption vehicles", "Transformer time series energy" - "genAI for recommendations", "explainable recommendations in transportation"

Sources / revues utiles : - IEEE Transactions on Intelligent Transportation Systems - Transportation Research Part C / D - Applied Energy, Energy - arXiv (pour derniers preprints) - Rapports NREL (National Renewable Energy Laboratory) - Communications de congrès : IV, ITS World Congress, KDD (appl. ML)

10. Points d'attention et risques

- Qualité / granularité des données : la présence/absence de variables (SoC, courant) conditionne les modèles possibles.
- Hétérogénéité des véhicules : un modèle formé sur un type peut mal généraliser à d'autres (nécessité d'adaptation/fine-tuning).
- Données privées / sensibles : gestion RGPD, anonymisation des traces.
- Trade-offs pratiques : itinéraire économisant énergie peut être plus long / moins sûr — nécessité d'intégrer contraintes multi-critères.