Morfetik : Une ressource lexicale morphologique extensible et modulaire pour le français

Jaime Arias*, Othman Boudarga*, Aude Grezka*

*CNRS, Université Sorbonne Paris Nord, LIPN, F-93430 Villetaneuse, France
{arias,boudarga,grezka}@lipn.univ-paris13.fr

Résumé. Les ressources lexicales morphologiques, décrivant la structure interne des mots et leurs formes fléchies, sont essentielles pour le traitement automatique des langues (TAL) et la linguistique computationnelle.

Nous présentons MORFETIK, une ressource lexicale open-source complète pour le français, capable de générer et d'identifier automatiquement toutes les formes fléchies des mots (noms, verbes, adjectifs, locutions, etc.). Il offre une couverture large du lexique contemporain et spécialisé, une architecture extensible et modulaire, et une intégration aisée avec des ressources externes.

De même, nous illustrons son utilisation à travers deux études de cas et détaillons son architecture, montrant comment sa modularité et son interopérabilité facilitent l'analyse de corpus, et le développement d'applications TAL.

1 Introduction

Une *ressource lexicale* est une base de données linguistique structurée qui rassemble des informations sur les mots d'une langue, leurs formes, leurs significations et leurs relations. Elle constitue un élément essentiel pour la recherche en traitement automatique des langues (TAL), en linguistique computationnelle et en technologies du langage. Selon leur conception et leurs objectifs, les ressources lexicales peuvent décrire différents aspects du lexique : la *morphologie* (formes et flexions des mots), la *syntaxe* (régimes et constructions), la *sémantique* (sens et relations lexicales) ou encore la *pragmatique*.

Dans cet article, nous présentons MORFETIK, une *ressource lexicale morphologique* complète et modulaire pour le français, conçue pour générer, structurer et exploiter automatiquement les formes fléchies du lexique (noms, adjectifs, déterminants, pronoms, verbes, adverbes, prépositions, conjonctions, interjections, locutions, etc.). Elle permet d'obtenir, pour n'importe quel mot français, l'ensemble de ses formes (pluriel des noms, féminin et pluriel des adjectifs, formes conjuguées des verbes, etc.), ou bien, réciproquement, d'identifier le mot (la forme de base, le "*lemme*") correspondant à n'importe quelle forme fléchie.

MORFETIK est une ressource clé pour l'analyse, la recherche et le développement d'applications en TAL qui offre :

— une couverture lexicale très large du français contemporain et spécialisé (médecine, minéralogie, etc.), plus de 240.000 lemmes ;

— un recensement lexical, réalisé par des linguistes à partir de nombreuses sources lexicographiques (*e.g.,* le Petit et le Grand Robert, GDEL, le Trésor de la langue française, Bescherelle, etc.) garantissant la précision et la fiabilité des informations contenues dans la ressource ;
— une génération automatique et systématique des formes fléchies (réduisant les erreurs manuelles) ;
— une structuration normalisée et exploitable par des outils TAL modernes ; et
— un ancrage linguistique solide, prenant en compte les variations, les formes rares et les défectivités.

En tant qu'application open-source, MORFETIK se distingue par sa modularité et son ouverture. Son architecture, composée d'une API et d'un frontend indépendants, permet leur évolution séparée et assure une interopérabilité avec d'autres clients (*e.g.,* un pipeline TAL, ou une interface en ligne de commande (CLI)). De même, il permet une intégration facile de ressources externes, telles que FranceTerme, Neoveille ou Wiktionnaire. Enfin, il est extensible, permettant l'enrichissement et la mise à jour continue des données, librement accessibles et téléchargeables. Il constitue ainsi une brique linguistique fondamentale pour la lemmatisation et l'étiquetage morpho-syntaxique ; l'analyse automatique de corpus ; la création ou l'enrichissement d'autres ressources lexicographiques ; et l'enseignement assisté par ordinateur.

Dans le domaine du TAL, on retrouve plusieurs ressources lexicales morphologiques comme le LEFFF (Sagot et al., 2006) ou encore le GLÀFF (Hathout et al., 2014), fondées sur des approches théoriques variées et offrant des fonctionnalités distinctes. Ces ressources ont servi de base à plusieurs outils de traitement automatique du français, notamment pour la lemmatisation, l'étiquetage morpho-syntaxique et la reconnaissance d'entités nommées. Plus récemment, des initiatives comme Neoveille (Cartier, 2017) ont permis d'enrichir la description lexicale à partir de données textuelles dynamiques, en y intégrant des métadonnées linguistiques, temporelles et contextuelles afin de suivre l'évolution du lexique et l'émergence de nouvelles formes.

Au niveau international, d'autres initiatives comme UniMorph (Batsuren et al., 2022) ou UDLexicons (Sagot, 2018) ont cherché à normaliser la représentation morphologique des langues à grande échelle, en proposant des formats unifiés pour la description des paradigmes flexionnels et des catégories grammaticales. Dans ce contexte, MORFETIK s'inscrit dans la continuité de ces travaux tout en apportant plusieurs contributions. La ressource repose sur un moteur de flexion capable de générer les formes fléchies à partir de règles linguistiques formalisées. Présenté initialement par Buvet et al. (2009), la plateforme visait à constituer un dictionnaire morphologique exhaustif de la langue française. Plus tard, Mathieu-Colas et al. (2015) en ont proposé une mise à jour importante, enrichissant la base de données et en évaluant sa couverture sur de grands corpus contemporains tels que Wikipedia, FrWac et Le Monde. Grâce à sa modularité et à une architecture logicielle ouverte, MORFETIK favorise aujourd'hui la réutilisation, l'intégration et l'extension des données dans différents environnements de TAL.

L'essor récent des modèles massifs (LLM) a transformé le paysage du TAL, mais la morphologie demeure un domaine où les approches neuronales rencontrent encore des limites bien documentées : erreurs d'accord, régularisations abusives, mauvaise gestion des formes rares, ambiguïtés mal résolues, ou encore sur-génération de formes inexistantes. Ces phénomènes montrent l'importance persistante de ressources morphologiques explicites pour garantir la cohérence linguistique et interprétable des applications. Dans ce contexte, MORFETIK occupe

une place stratégique en fournissant un inventaire morphologique exhaustif, structuré, contrôlé par des linguistes et entièrement interopérable grâce à des formats normalisés et à une API stable. MORFETIK est conçu pour être utilisé dans des pipelines hybrides modernes, où les modèles neuronaux bénéficient de l'appui de ressources symboliques afin d'améliorer la précision, la robustesse et l'explicabilité : génération de cohorte d'analyses morphologiques pour guider un modèle, vérification automatique des accords produits par un LLM, filtrage des sorties morphologiquement invalides, supervision faible pour la construction de datasets annotés, etc. Ainsi, loin de s'opposer aux approches actuelles, MORFETIK propose une complémentarité forte qui renforce la qualité et la fiabilité des systèmes neuronaux contemporains.

Ce document est structuré de la manière suivante : la Section 2 est consacrée à l'analyse des deux études de cas. Cette analyse permet de cerner les problématiques principales et de justifier l'importance de MORFETIK. La Section 3 expose ensuite l'architecture de l'application, en détaillant sa structure interne, les technologies employées et les interactions entre les différents composants. Enfin, la Section 4 présente les conclusions et perspectives de ce travail, en mettant en avant les contributions apportées et les pistes d'amélioration envisageables pour des travaux futurs.

2 Études de cas

Cette section présente deux études de cas illustrant l'utilisation de MORFETIK dans des contextes distincts. Ces études mettent en évidence les principaux atouts de l'application.

2.1 Recherche d'un terme

L'interface web de MORFETIK permet à l'utilisateur de rechercher un terme souhaité. Une fois la requête soumise, le système interroge la base de données et affiche les résultats correspondants de manière structurée. Comme l'illustre la Figure 1, l'interface offre plusieurs options de filtrage, notamment la recherche *stricte*, *sensible à la casse* et *sensible aux accents*, afin de permettre une exploration linguistique plus précise. Chaque résultat est en outre associé à plusieurs *ressources externes* de référence, telles que *FranceTerme*, *Neoveille* ou *Wiktionary*, facilitant ainsi l'accès à des informations complémentaires et à des contextes d'usage variés.

La Figure 1 illustre la recherche du terme "avions" mettant en évidence la capacité de MORFETIK à gérer les phénomènes d'ambiguïté morphologique caractéristiques du français. En effet, le système renvoie deux analyses distinctes correspondant à des catégories grammaticales différentes. D'une part, "avions" est identifié comme un nom masculin pluriel, forme fléchie du lemme *avion*. D'autre part, il est également reconnu comme une forme verbale conjuguée, correspondant à la première personne du pluriel de l'imparfait de l'indicatif du verbe *avoir*. Cette double analyse illustre la manière dont la ressource encode et distingue les informations de catégorie lexicale, de flexion et de fonction morphosyntaxique, tout en maintenant une cohérence entre les différents modules lexicaux.

2.2 Intégration avec un pipeline TAL : le cas de ChêneTAL

MORFETIK présente une interface de programmation (API) rigoureusement définie, facilitant son interopérabilité avec des services externes. Cette API, permet une communication

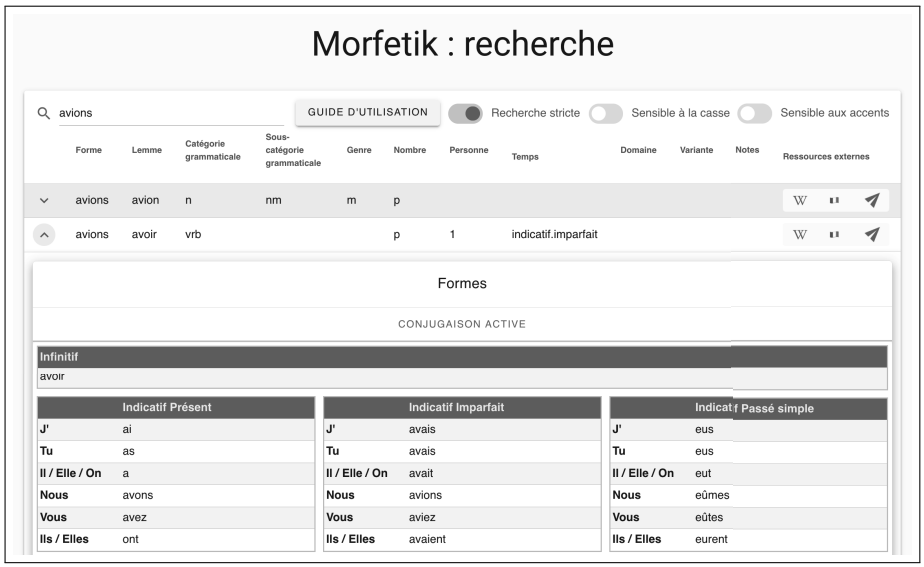

FIG. 1 – *Interface web de* MORFETIK *illustrant le processus de recherche d'un terme*

structurée et fiable entre MORFETIK et d'autres plateformes. À titre d'exemple, la plateforme ChêneTAL (*i.e.,* une plateforme d'expérimentation sur des outils TAL et d'IA) peut exploiter cette interface afin d'accéder aux fonctionnalités offertes par MORFETIK et d'intégrer ses données dans ses propres processus applicatifs.

Pour illustrer le mode d'accès aux ressources offertes par l'API de MORFETIK, considérons la requête permettant de récupérer l'ensemble des verbes disponibles dans le système.

GET	**/verbs**	Récupérer tous les verbes

Cette opération repose sur la méthode `GET` du protocole HTTP, qui vise à interroger une ressource sans en modifier l'état. L'exemple ci-dessous présente la requête envoyée à l'API, ainsi que la réponse correspondante en format JSON :

```bash
> curl https://tal.lipn.univ-paris13.fr/morfetik2/api/verbs

[
  {
    "id": "019a4a37-cf8d-762e-a0f2-b5101661d7d9",
    "value": "avoir",
    "codeId": "019a4a37-cf7d-77de-9a4a-661e213fc062",
    "category": "VERB",
    "notes": "",
    "rare": false,
    "domain": "",
    "subcategory": null
  },
```

```
  ...
]
```

3 Architecture

Cette section présente l'architecture de MORFETIK et décrit les principes fondamentaux ainsi que les avantages en termes de flexibilité, de maintenabilité et d'évolutivité.

MORFETIK repose sur le paradigme de l'architecture "*ports et adaptateurs*" ou "*hexagonale*" (Cockburn et de Paz, 2024), un modèle qui sépare clairement la logique métier des technologies externes. La logique métier définit les règles et comportements essentiels de l'application sans se soucier de la manière dont les données sont stockées, affichées ou échangées. Les interactions avec le monde extérieur passent par des *ports*, qui représentent des interfaces abstraites. Ces ports sont concrétisés par des *adaptateurs*, qui traduisent les interactions abstraites de la logique métier en opérations concrètes avec les composants externes, tels que les bases de données ou les interfaces utilisateur.

Grâce à cette approche, MORFETIK est hautement modulaire et extensible. Différents types de clients peuvent consommer les mêmes fonctionnalités métiers sans modification du cœur du code. Par exemple, au-delà de l'interface graphique actuelle, il serait simple d'ajouter une interface en ligne de commande (CLI) ou un script pour de *scrapping*. Cette flexibilité favorise la réutilisation du code et la cohérence fonctionnelle entre plusieurs points d'accès.

L'un des autres avantages majeurs de cette architecture réside dans sa résilience face aux évolutions technologiques. Comme la logique métier ne dépend d'aucune technologie spécifique, il devient possible de remplacer ou de faire évoluer des composants techniques sans impact majeur sur le reste du système. Par exemple, l'application utilise actuellement PostgreSQL comme système de gestion de base de données, mais il est facile d'intégrer un autre moteur de base de données, ou d'expérimenter différentes configurations afin d'optimiser les performances et la rapidité d'exécution des requêtes.

La Figure 2 illustre l'architecture logicielle actuelle de MORFETIK. Cette architecture est composée de deux grandes parties :

1. **Morfetik UI :** Le frontend, ou interface utilisateur (UI), représente la couche avec laquelle les utilisateurs interagissent directement. Les requêtes des utilisateurs sont envoyées au backend sous forme de requêtes HTTP, et les réponses sont retournées en format JSON. Il est développé avec le framework Vue.js, qui permet de concevoir des applications web modernes, dynamiques et ergonomiques en s'appuyant sur des technologies standards du web telles que HTML, CSS et JavaScript. Par conséquent, l'utilisation de MORFETIK ne requiert aucune installation préalable : l'utilisateur peut y accéder directement à partir de n'importe quel navigateur web.
2. **Morfetik API :** Le backend, ou API de l'application, a pour rôle de consulter la ressource, de traiter les informations et de retourner les résultats de manière structurée au frontend. Il est développé avec le framework AdonisJS et il est organisé en couches suivant les principes de l'architecture hexagonale, à savoir :
   — Les *adaptateurs entrants* gèrent les interactions avec le monde extérieur. Dans notre cas, il s'agit des *contrôleurs* qui reçoivent les requêtes HTTP du frontend, les valide

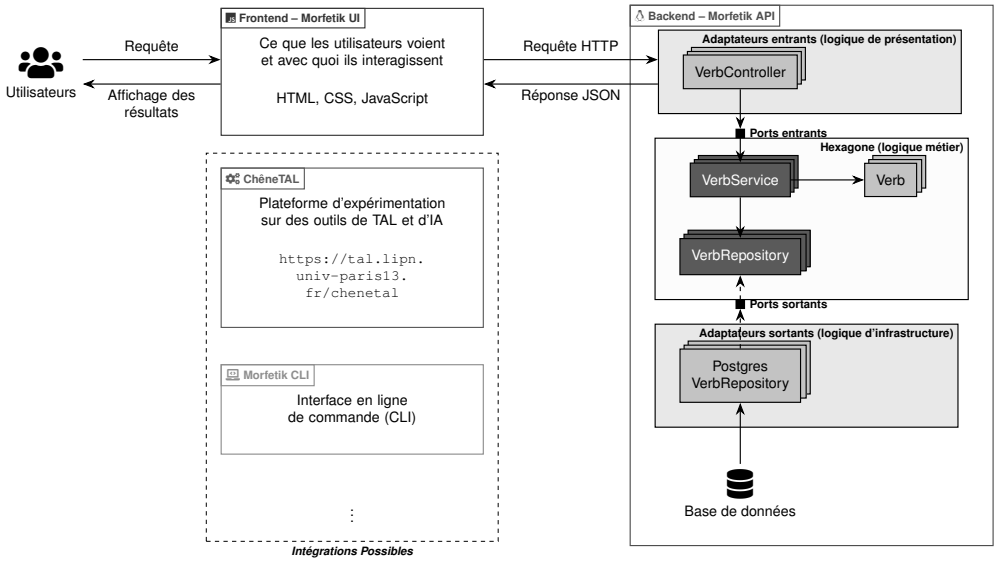

FIG. 2 – *Architecture logicielle de* MORFETIK

et délègue leur traitement aux services internes appropriés. Par exemple, le contrôleur `VerbController` expose des endpoints HTTP permettant d'invoquer les opérations de création, de consultation, de mise à jour et de suppression (CRUD) sur les verbes.

— Le cœur de l'application regroupe la *logique métier* proprement dite, structurée autour de *services* et *d'entités*. Les services définissent les règles métier et orchestrent les interactions entre les différents ports, tout en demeurant indépendant de toute technologie ou infrastructure spécifique. À titre d'exemple, nous présentons uniquement le service `VerbService` et l'entité `Verb`, qui incarnent le modèle applicatif pour les verbes. Néanmoins, cette logique s'étend à l'ensemble des catégories morphologiques couvertes par MORFETIK.

— Les *adaptateurs sortants* assurent la communication entre la couche métier et les infrastructures externes, notamment les systèmes de persistance. Par exemple, l'adaptateur `PostgresVerbRepository` implémente le port `VerbRepository`, offrant au service métier un accès abstrait et unifié aux données. Cette séparation entre la logique métier et les détails techniques du stockage favorise un faible couplage, une meilleure testabilité et facilite le remplacement ou l'évolution des composants d'infrastructure sans impacter la logique métier.

Il est important de souligner que la Figure 2 illustre de manière simplifiée un contrôleur (`VerbController`), un service (`VerbService`) et une entité métier (`Verb`). Cette représentation est illustrative : dans la réalité, l'architecture couvre l'ensemble des éléments lexicaux, chaque composant étant défini de manière générique et modulaire pour permettre l'ajout de nouveaux services, entités ou adaptateurs sans modifier les couches existantes.

MORFETIK est doté d'un moteur de flexion (voir Figure 3). Il assure la génération des

formes fléchies à partir d'un lemme et de son code morphologique. Chaque lemme est catégorisé par sa catégorie grammaticale (*e.g.,* verbe, nom, etc.) et par un code qui spécifie les règles morphologiques (*i.e.,* radical et terminaison). Ce sont les deux informations nécessaires à la génération de forme. À partir de cette combinaison, le moteur de flexion applique les règles morphologiques sur le lemme afin de produire l'ensemble des formes. Il s'agit de conjugaison pour les verbes, et de flexions nominales et adjectivales pour les noms et adjectifs. Ainsi, le moteur de flexion de MORFETIK permet de dériver automatiquement toutes les formes correctes d'un lemme grâce au code morphologique qui lui est associé, garantissant donc la cohérence linguistique.

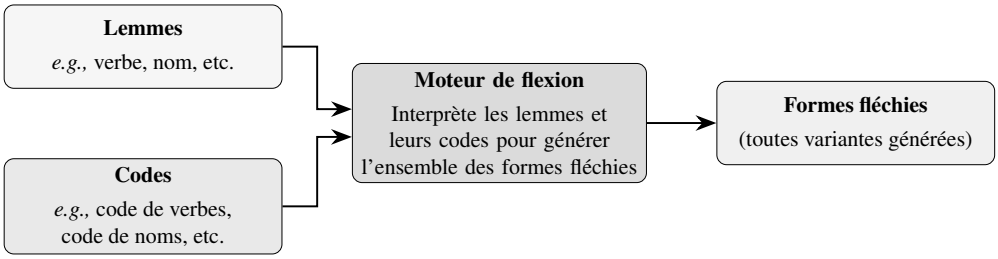

FIG. 3 – *Moteur de flexion de* MORFETIK

4 Conclusion et perspectives

Dans cet article, nous avons présenté MORFETIK, une application open-source permettant de consulter une ressource lexicale. Elle est évolutive, indépendante des technologies sous-jacentes et facilement extensible grâce à son architecture. Nous avons également illustré son fonctionnement à travers deux études de cas, qui mettent en évidence sa robustesse et sa pertinence, ainsi que son apport pour la gestion et la consultation de données lexicales.

Pour les travaux futurs, plusieurs axes d'amélioration sont envisagés : optimiser les performances des requêtes pour obtenir les résultats plus rapidement ; effectuer des benchmarks de différents moteurs de base de données afin de sélectionner celui le plus adapté à notre cas d'usage, ainsi que des benchmarks comparatifs de performance pour situer notre approche par rapport à l'existant ; enrichir l'interface graphique en intégrant les retours d'expérience ; et développer un plugin web permettant d'ajouter plus facilement de nouveaux éléments lexicaux.

Références

Batsuren, K., O. Goldman, S. Khalifa, N. Habash, W. Kieras, G. Bella, B. Leonard, G. Nicolai, K. Gorman, Y. G. Ate, M. Ryskina, S. J. Mielke, E. Budianskaya, C. El-Khaissi, T. Pimentel, M. Gasser, W. A. Lane, M. Raj, M. Coler, J. R. M. Samame, D. S. Camaiteri, E. Z. Rojas, D. L. Francis, A. Oncevay, J. L. Bautista, G. C. S. Villegas, L. T. Hennigen, A. Ek, D. Guriel, P. Dirix, J. Bernardy, A. Scherbakov, A. Bayyr-ool, A. Anastasopoulos, R. Zariquiey, K. Sheifer, S. Ganieva, H. Cruz, R. Karahóga, S. Markantonatou, G. Pavlidis,

M. Plugaryov, E. Klyachko, A. Salehi, C. Angulo, J. Baxi, A. Krizhanovsky, N. Krizhanovskaya, E. Salesky, C. Vania, S. Ivanova, J. C. White, R. H. Maudslay, J. Valvoda, R. Zmigrod, P. Czarnowska, I. Nikkarinen, A. Salchak, B. Bhatt, C. Straughn, Z. Liu, J. N. Washington, Y. Pinter, D. Ataman, M. Wolinski, T. Suhardijanto, A. Yablonskaya, N. Stoehr, H. Dolatian, Z. Nuriah, S. Ratan, F. M. Tyers, E. M. Ponti, G. Aiton, A. Arora, R. J. Hatcher, R. Kumar, J. Young, D. Rodionova, A. Yemelina, T. Andrushko, I. Marchenko, P. Mashkovtseva, A. Serova, E. Prud'hommeaux, M. Nepomniashchaya, F. Giunchiglia, E. Chodroff, M. Hulden, M. Silfverberg, A. D. McCarthy, D. Yarowsky, R. Cotterell, R. Tsarfaty, et E. Vylomova (2022). Unimorph 4.0 : Universal morphology. In *LREC*, pp. 840–855. European Language Resources Association.

Buvet, P., E. Cartier, F. Issac, Y. Madiouni, M. Mathieu-Colas, et S. Mejri (2009). Morfetik, ressource lexicale pour le TAL. In *TALN (Articles courts)*, pp. 217–226. ATALA.

Cartier, E. (2017). Neoveille, a web platform for neologism tracking. In *EACL (Software Demonstrations)*, pp. 95–98. Association for Computational Linguistics.

Cockburn, A. et J. M. G. de Paz (2024). *Hexagonal Architecture Explained : How the Ports & Adapters Architecture Simplifies Your Life, and How to Implement It*. Humans and Technology Press.

Hathout, N., F. Sajous, et B. Calderone (2014). Glàff, a large versatile french lexicon. In *LREC*, pp. 1007–1012. European Language Resources Association (ELRA).

Mathieu-Colas, M., E. Cartier, et A. Grezka (2015). Dictionnaires morphologiques du français contemporain : présentation de morfetik, éléments d'un modèle pour le TAL. In *TALN*, pp. 150–156. ATALA.

Sagot, B. (2018). A multilingual collection of conll-u-compatible morphological lexicons. In *LREC*. European Language Resources Association (ELRA).

Sagot, B., L. Clément, É. V. de la Clergerie, et P. Boullier (2006). The lefff 2 syntactic lexicon for french : architecture, acquisition, use. In *LREC*, pp. 1348–1351. European Language Resources Association (ELRA).

Summary

Morphological lexical resources, describing the internal structure of words and their inflected forms, are crucial for natural language processing (NLP) and computational linguistics.

We present MORFETIK, a comprehensive open-source lexical resource for French, capable of automatically generating and identifying all inflected forms of words (nouns, verbs, adjectives, phrases, etc.). It offers broad coverage of the contemporary and specialised lexicon, an extensible and modular architecture, and easy integration with external resources.

We also illustrate its use through two case studies and detail its architecture, showing how its modularity and interoperability facilitate corpus analysis and the development of NLP tools.