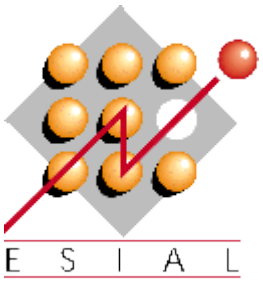


ESIAL

–

Observatoire Astronomique de Strasbourg

Projet Industriel
Rapport intermédiaire

Recherche de noms d'objets astronomiques dans les textes

Thomas Bouton
Benoît Lang
Benoit Burdy

Encadrants : Mr Tomczak (CRAN)
Mme Lesteven (Observatoire Astronomique de Strasbourg)

Sommaire

Introduction

Ce document est destiné à faire un bilan du projet industriel de 3ème année ESIAL, 2 mois après son début. Ce rapport contiendra donc une présentation du sujet et de son contexte, le cahier des charges établi au début du projet, l'état d'avancement actuel du projet, puis une partie consacrée à la description des tâches restantes.

La partie concernant le travail effectué reprendra tous les points techniques abordés, méthodes de travail, raisonnements ... et s'adressera donc à un public plus averti.

Nous ferons également figurer dans ce document l'ensemble des problèmes rencontrés et à venir.

1. Présentation du Projet

1.1 Observatoire Astronomique de Strasbourg

Situé en plein centre de la ville, au sein du campus universitaire, l'observatoire astronomique de Strasbourg se consacre principalement à la recherche et aux formations dans le domaine de l'astronomie. Pour n'en citer que quelques exemples, l'Observatoire intervient dans les enseignements de licence, mastere et. ; poursuit actuellement des recherches sur la physique stellaire ou l'astrophysique des hautes énergies etc.

L'Observatoire de Strasbourg est également responsable du Centre de données astronomiques (CDS) qui est en charge de:

- collecte, identification, analyse, archivage et diffusion des données sur les objets astronomiques ;
- bibliographie des objets astronomiques ;
- banque de données SIMBAD; serveur ViziR; atlas d'images du ciel ALADIN.

Ce dernier point, et notamment la base de données SIMBAD nous intéresse particulièrement puisqu'elle est à l'origine du projet industriel proposé à l'ESIAL.

Cette base de données rassemble l'ensemble des objets astronomiques connus à ce jour. Pour chaque objets, on trouvera par exemple ses coordonnées dans l'espace, le nom de la personne qui l'a découvert etc. et surtout, l'ensemble des noms que cet objet peut prendre dans la littérature.

1.2 Contexte

Notre projet s'inscrit donc dans le cadre de la base de données SIMBAD. L'une des applications de SIMBAD au sein de l'Observatoire de Strasbourg et de créer des liens entre des textes au format électronique et cette base de données. Ainsi, pour chaque nouvel article paru sur le web, le texte est analysé, chacun des noms d'objets astronomiques et relevé puis un lien hypertexte est créé entre le nom détecté et son entrée correspondante dans SIMBAD.

C'est sur ce point précis que le projet à été proposé à l'ESIAL : l'objectif étant de rendre automatique la recherche des noms d'objets astronomiques, ainsi que la création du lien entre cet objet et la base SIMBAD.

2. Cahier des Charges

2.1 Objectifs

L'objectif principal de ce projet est de détecter et de baliser, automatiquement, l'ensemble des noms d'objets astronomiques cités dans des textes scientifiques.

La base de connaissance utilisée par cette détection est le dictionnaire de la nomenclature des objets célestes qui contient toute l'information concernant la désignation des objets astronomiques (acronymes – appartenance à une liste d'observations- et les formats d'écriture). Ce dictionnaire est consultable en ligne : http://vizier.u-strasbg.fr/cgi-bin/Dic. Des listes supplémentaires (constellations, NAMES, lettres grecques) compléteront cette connaissance.

Les textes à traiter sont disponibles dans divers formats : ASCII, HTML, PDF, PostScript, LateX. L'outil à développer doit convertir ces documents en un format électronique manipulable afin d'analyser ces textes pour extraire et baliser les noms d'objets sur la base de données astronomiques SIMBAD.

Cet outil de reconnaissance sera utilisé dans différentes applications. Le taux de reconnaissance des noms d'objets sera plus ou moins strictes en fonction de ces applications, la vitesse du traitement pourra elle aussi varier, le balisage des noms détectés pourra être différent.

Une première application à ce travail sera l'implémentation d'une IHM visant à aider les documentalistes dans leur travail d'indexation.

2.2 Phases

Après avoir pris connaissance du sujet, nous avons décidé de le découper en 5 phases, non nécessairement chronologiques :

2.2.1 Détection du type de fichier et conversion en Ascii

Les sources des textes pouvant être différentes (pdf, html …), nous réaliserons un traitement préalable qui fera une conversion du texte sous forme Ascii. Ce traitement permettra de ne faire la recherche que sur un type de format, quelle que soit la source.

Les publications scientifiques à étudier contiennent du texte, mais aussi des tables de données, des figures, graphes, images. Nous limitons le travail à l'analyse des textes, des légendes, et si le temps le permet, aux tables.

2.2.2 Traitement du dictionnaire

Il faudra créer un module permettant d'extraire l'ensemble des acronymes et des formats contenus dans le dictionnaire de nomenclature, puis de convertir ces formats en expressions régulières en vue de la recherche. Ce module doit tenir compte de la mise à jour hebdomadaire du dictionnaire.

2.2.3 Recherche des noms d'objets

Nous envisageons d'implémenter deux algorithmes pour la recherche (ou le même avec des options de restriction) :

- un permettant de détecter et de baliser les noms d'objets avec rigueur afin d'éviter des ancres qui pointent vers SIMBAD mais sur un objet inexistant.
- un permettant un taux de reconnaissance très large afin de détecter tous les noms dans le but d'indexer les bases de données (une validation par un documentaliste étant obligatoire).

En ce qui concerne l'algorithme de recherche, nous nous baserons sur la méthode préalablement utilisée par Mme Lesteven : extraction des acronymes et formats du dictionnaire, conversion de ces informations en expressions régulières et recherche de ces expressions dans les textes afin de baliser les noms reconnus. Ces détections sont affinées par une analyse du contexte direct (mots placés avant et après).

Il faudra envisager la possibilité de détecter de nouveaux noms d'objets, dans ce cas, il faudra étudier le contexte de la phrase et/ou la syntaxe.

Cependant, nous travaillerons en parallèle sur un autre type d'algorithme, basé uniquement sur le contexte et/ou la syntaxe, et faisant donc abstraction (tout ou en partie) du dictionnaire et des listes supplémentaires.

2.2.4 Balisage des noms d'objets détectés

Le résultat de la recherche sera une liste de noms qu'il suffira de rechercher dans le document original afin de les baliser (mettre une ancre vers la base de données SIMBAD ou vers un logiciel de mise à jour). La forme originale du texte doit-être maintenue.

2.2.5 Interface

S'il nous reste encore un peu de temps en fin de projet, nous développerons une Interface Homme-Machine (IHM) permettant une utilisation, la plus intuitive possible, de cet outil de recherche dans le but d'aider les documentalistes dans leur travail d'indexation.

2.3 Difficultés prévisibles – Sources d'erreurs

Après analyse du sujet, nous avons établi une liste des principales sources d'erreurs ou de difficultés prévisibles, il s'agit de :

- Recherche des noms d'objets
  - L'algorithme devra être le plus rapide possible, malgré la grande quantité d'informations à traiter
  - La recherche devra prendre en compte de très nombreux cas spéciaux, exceptions.. Et devra donc prendre en compte le contexte des phrases
  - La faisabilité d'un algorithme basé uniquement sur le contexte ?

- Types de fichiers
  - La liste des types de fichiers à prendre en compte n'est pas encore établie, nous nous efforcerons dans un premier temps de traiter les fichiers PDF et HTML.

Les principales difficultés résideront donc dans l'écriture d'un algorithme de recherche fiable et rapide et dans la conversion des fichiers.

2.4 Notes sur la gestion du projet

Un rapport hebdomadaire, même informel, est demandé pour faciliter l'avancement et le suivi du projet. Pour cela on utilisera le Wiki du CDS, disponible à l'adresse suivante : http://cds.u-strasbg.fr/twiki/bin/view/Stages/Esial.

3. Gestion du projet

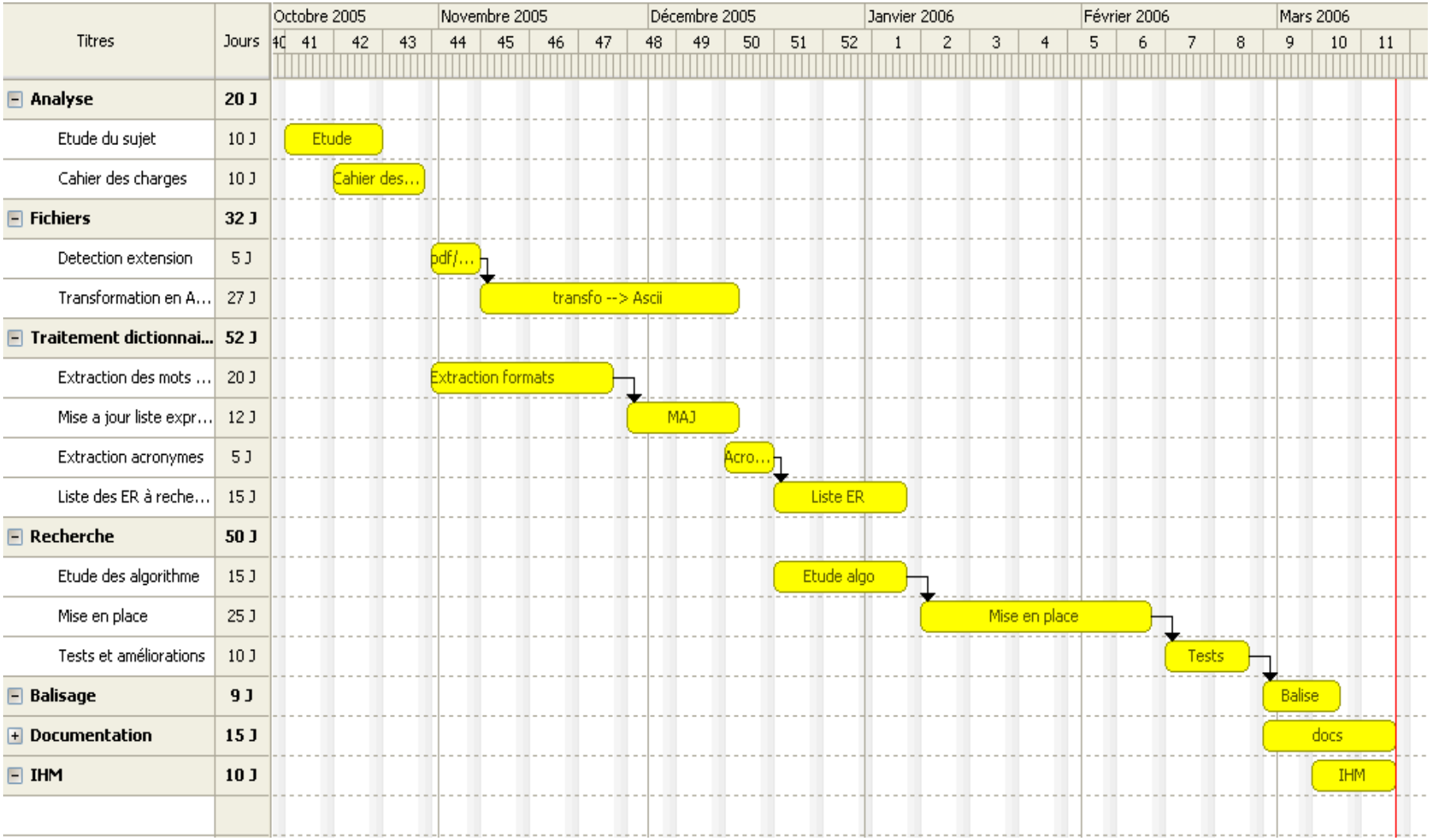
3.1 Responsabilités

Pour garantir une gestion optimale du projet, l'équipe à été répartie comme suit :

D'une manière générale, Thomas Bouton supervise l'ensemble des tâches concernant la transformation des fichiers en Ascii, ainsi que la partie concernant l'algorithme de recherche. Benoit Burdy étant en charge de la gestion du dictionnaire : extraction et traitements des informations, ainsi que d'interface graphique (si sa réalisation à lieue). Les aspects de communication entre les divers intervenants (encadrants industriel et universitaire, groupe de projet) seront gérés par Benoît Lang.

3.2 Planning prévisionnel

Après étude du projet, nous avons établi un premier planning prévisionnel (page suivante) , qui sera prochainement remplacé par un planning plus complet réalisé sous Microsoft Project.


Titres
Jours
Octobre 2005
Novembre 2005
Décembre 2005
Janvier 2006
Février 2006
Mars 2006
40 41 42 43 44 45 46 47 48 49 50 51 52 1 2 3 4 5 6 7 8 9 10 11
Analyse 20 J
Etude du sujet 10 J
Cahier des charges 10 J
Fichiers 32 J
Detection extension 5 J
Transformation en A... 27 J
Traitement dictionnai... 52 J
Extraction des mots ... 20 J
Mise a jour liste expr... 12 J
Extraction acronymes 5 J
Liste des ER à reche... 15 J
Recherche 50 J
Etude des algorithme 15 J
Mise en place 25 J
Tests et améliorations 10 J
Balisage 9 J
Documentation 15 J
IHM 10 J
Etude
Cahier des...
pdf/...
transfo --> Ascii
Extraction formats
MAJ
Acro...
Liste ER
Etude algo
Mise en place
Tests
Balise
docs
IHM

4. Travail effectué

4.1 Détection du type de fichier et transformation en ASCII

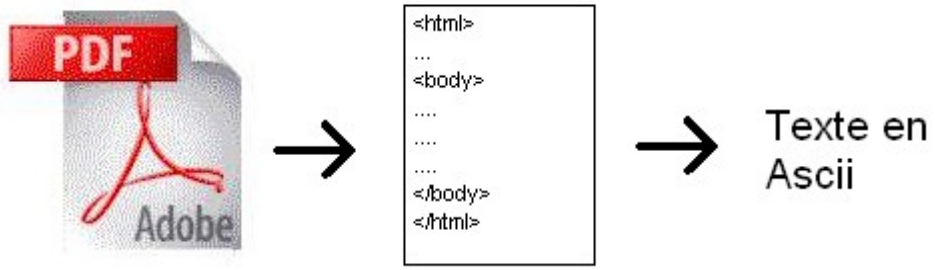
L'application devant être capable de traiter des fichier PDF, HTML, ASCII, LateX, PostScript, il était nécessaire de transformer les fichiers en un format donné. De cette manière, l'algorithme de recherche n'aura pas à se soucier du type du texte.

Notre choix s'est naturellement tourné vers ASCII, forme de texte la plus simple possible. Notre première tâche fût donc la transformation des texte PDF et HTML au format ASCII (le PostScript et le LateX seront traités plus tard, si besoin mais ne devraient à priori pas poser de problèmes),

Le traitement des fichiers HTML se fait par simple réécriture des textes compris entre les balises, en supprimant ces dernière. Il est possible d'effectuer un traitement particulier pour chaque balise, et donc de formater en sortie à notre convenance les tableaux, listes ... ainsi que de détecter les images.

Pour les fichiers au format PDF, nous utilisons un outil permettant la transformation d'un fichier PDF en HTML. Il suffit donc ensuite de traiter le fichier HTML avec la méthode précédente.

Cet outil est disponible aussi bien sous Windows que Linux et permet le traitement des images, tableaux etc...



Dans certains textes, il peut arriver qu'une image représente l'un des caractères d'un nom d'objet. Il serait possible d'intégrer un module de reconnaissance de caractères dans les images, cependant, de nombreux problèmes se posent : les logiciels de reconnaissance de caractères libres (gratuits) sont peu nombreux et peu efficace, les offres payantes sont plus lourdes à mettre en place et, par définition, plus coûteuse. De plus, il est peu probable de trouver un outil capable de traiter les lettres grecques. Le développement d'un tel module n'est pas envisageable dans le temps qui nous est imparti, et l'utilisation d'un logiciel existant ne donnera pas des résultat satisfaisants justifiant les temps de traitement supplémentaires engendrés.

Une solution envisageable serait la suivante : chaque image serait extraite du fichier puis proposée à l'utilisateur afin qu'il donne le caractère correspondant, avec bien sûr la possibilité de désactiver cette option et donc de ne pas traiter les images. L'avantage de cette solution est qu'elle est relativement simple à mettre en place et ne ralentira que très peu l'exécution de l'application, l'inconvénient étant l'obligation pour l'utilisateur d'être présent.

4.2 Extraction des informations du dictionnaire

Comme énoncé dans le cahier des charges, nous nous baserons sur l'algorithme de recherche de Mme Lesteven pour écrire le programme. Nous aurons donc besoin de l'ensemble des expressions régulières sous lesquelles peuvent se présenter les noms d'objets astronomiques.

Afin d'établir cette liste, il nous a donc fallu extraire les formats et les acronymes présents dans le dictionnaire. Notre programme permet donc l'extraction de l'ensemble des mots de chaque format.

Un mot d'un format est une série de lettres ; chaque lettre correspond à une expression régulière. Par exemple, le mot de format `aaa` correspond à l'expression régulière suivante :

`[A-Z]\{1,3\}`

Les formats peuvent être constitués de plusieurs mots séparés par un espace ou par d'autres caractères (. ; + ; / ; \ ;[;] ;{ ;} …). Exemples de formats complets :

```
UGC NNNN NNN
HHMM+DDA
JHHMMSS.ss+DDMMSS.s
```

Nous avons donc programmé une classe JAVA nommée RechercheFormats cherchant tous les noms de formats dans le fichier du dictionnaire et de les écrire dans un fichier texte (liste_mots). Ainsi, à partir des formats énoncés précédemment, on obtiendrait la liste suivante :

```
DDA
DDMMSS
JHHMMSS
HHMM
NNN
NNNN
UGC
s
ss
```

Chaque mot doit ensuite être traduit en expression régulière. Pour cette tâche, nous nous servons de la liste existante des équivalent mot/expression régulière. Le programme met uniquement de coté les nouveaux mots qu'ils n'a pu traduire en vue de les proposer à l'utilisateur.

Une automatisation de la traduction des mots en expression régulière semble extrêmement difficile, en effet, certain format peuvent avoir plusieurs interprétations, par exemple N peut correspondre à un chiffre en 0 et 9 ou bien à la direction Nord. Le choix entre les deux se fait à l'aide d'un texte explicatif présent dans le dictionnaire. Nous nous contenterons donc pour le moment de proposer les nouveaux mots, accompagné du texte explicatif, à l'utilisateur lors de la mise à jour du dictionnaire.

5. Prochaines étapes

5.1 Traitement du dictionnaire – suite

Bien que très avancé, le traitement des informations du dictionnaire n'est pas tout à fait terminé. En effet, il nous reste encore à extraire l'ensemble des acronymes. De plus, nous devons comparer nos résultats avec ceux obtenus par le programme de Mme Lesteven sur un dictionnaire identique.

5.2 Algorithme de recherche

L'écriture de l'algorithme de recherche sera sans doute la partie la plus longue et la plus difficile du projet. Nous ne nous somme que très peu penché sur le problème pour le moment. Dans un premier temps, nous analyserons l'algorithme utilisé dans le précédent programme, afin de déterminer si des améliorations sont possibles.

Il est fortement probable que nous ayons à écrire plusieurs algorithmes, afin de répondre aux exigences du sujet, à savoir un taux de reconnaissance variable suivant les applications.

En parallèle, nous travaillons sur un tout autre type d'algorithme, basé uniquement sur le contexte du texte. Il ne permettra probablement un taux de reconnaissance maximale, mais sera très certainement plus rapide que la méthode classique qui utilise le dictionnaire.

Pour cet algorithme, nous avons déjà récupéré des cours de master traitant du sujet, nous contacterons les chercheurs concernés ultérieurement.

5.3 IHM (Interface Homme-Machine)

Si le temps le permet en fin de projet, nous réaliseront une interface graphique complète et ergonomique afin d'utiliser au mieux l'application développée.

5.4 Documentation technique

Nous accompagnerons notre rapport final d'une documentation technique détaillée : JavaDoc complète, description des algorithme utilisés etc ... afin de permettre une réutilisation rapide de notre travail.

Conclusion

Ce projet s'avère complet, concret et ambitieux, pour ne citer que trois adjectifs.

Complet tout d'abord puisqu'il aborde de nombreux points : parsage de texte, algorithme de recherche, transformation d'informations et développement d'interface graphique, entre autres. Le projet mettra en oeuvre bon nombre de compétences acquises à l'ESIAL, et abordera des thèmes nouveaux.

Concret ensuite, puisque l'outil devenu nécessaire, voire indispensable à l'analyse rapide de nombreux article sera utilisé quotidiennement par les analystes de l'Observatoire Astronomique de Strasbourg.

Ambitieux enfin, car pour finaliser l'outil en conformité avec les attentes de l'industriel, des développements et amélioration ultérieures seront sans doute nécessaires. Tout d'abord pour la partie reconnaissance de caractère dans les images qu'il est peu probable que nous traitions en totalité. Une automatisation de la génération des expressions régulières est également envisageable, quoique rendue difficile par les multiples traductions pour un mot donné. Enfin, l'interface que nous proposerons ne sera probablement pas aussi complète que celle souhaitée par l'industriel, faute de temps.

Une bonne organisation de l'équipe à permit de respecter largement le planning fixé, mais la partie la plus importante du projet reste à faire : le module de recherche proprement dit.