Alma Mater Studiorum · Università di Bologna

Scuola di Scienze
Corso di Laurea in Informatica

Analisi della posizione dei riferimenti bibliografici nelle pubblicazioni scientifiche: modello e implementazione

Relatore:
Dott. Angelo Di Iorio

Presentata da:
Federico Giubaldo

Anno Accademico 2015-16
Sessione II

Contenuti

List of Figures

Capitolo 1

Introduzione

Questa tesi è incentrata sulla ricerca e l'implementazione di criteri per l'analisi dei modi in cui, gli autori in ambiente accademico, posizionano i riferimenti bibliografici nel testo. Il lavoro è nato dall'intuizione, confermata da alcune analisi preliminari, che esistono comportamenti differenti per quanto riguarda il posizionamento delle citazioni bibliografiche al variare delle discipline o delle riviste.
Con il lavoro descritto dalla tesi corrente si vogliono analizzare, in dettaglio, le differenze comportamentali al fine di estrapolare, ad esempio: informazioni relative alle caratteristiche specifiche per le diverse discipline oppure per le differenti riviste, o ancora, le caratteristiche di specifici autori.
Questi risultati sono il punto di partenza che ha dato il via allo studio. Infatti i risultati, sopra discussi, possono essere la base di analisi più ampie, ottenute facendo dei confronti tra di essi; ad esempio si può pensare a delle analisi che evidenzino le differenze tra le riviste che trattano la stessa disciplina oppure, banalmente, tra le diverse discipline.
A questo punto è chiaro che non esiste una modalità unica e standard per posizionare i riferimenti bibliografici ma che, al variari delle discipline e delle riviste trattate, cambiano i comportamenti. Tali comportamenti potrebbero essere dettati dalle necessità dell'argomento trattato o, dalle linee guida definite da una certa rivista o, ancora, dalla personalità dell'autore.
Quindi, definiamo i due obiettivi principali dello studio svolto nel modo seguente: affrontare, in modo sistematico, un argomento che, dalle informazioni in nostro possesso, non è stato trattato prima; ottenere distribuzioni riguardanti i riferimenti bibliografici nelle pubblicazioni scientifiche che permettano di estrapolare i trend di riviste e discipline.
Nei seguenti capitolo sarà descritta la logica e l'implementazione di Diribia, applicazione proposta come soluzione al secondo obiettivo descritto sopra. Gli articoli scientifici utilizzati per la fase di sviluppo e di valutazione dell'applicazione appartengono all'archivio PMC [17], il quale mette a disposizione un numero elevato di pubblicazioni scientifiche nell'ambito biomedico e biologico.

Diribia è formata dalle seguenti componenti: convertitore, analizzatore dei riferimenti bibliografici, analizzatore dei titoli, generatore dei grafici.
Il convertitore si occupa della trasformazione degli articoli con formato XML e specifico DTD in articoli XML con una particolare struttura che chiameremo: formato PLUS. Gli articoli in questo formato saranno utilizzati dagli analizzatori. In tal modo, gli analizzatori non dovranno gestire vari DTD bensì un formato unico e ad hoc.
L'analizzatore dei riferimenti bibliografici mette a disposizione diversi tipi di analisi, ognuna delle quali è caratterizzata da una specifica divisione logica che divide l'articolo in N parti logiche. Queste parti logiche saranno il discriminante tra le tipologie di analisi e quindi, tra le distribuzioni risultanti.
L'obiettivo delle diverse divisioni logiche è capire se i riferimenti bibliografici, ad esempio, si trovano all'inizio, nella parte centrale o alla fine del testo. Per catturare questa intuizione si può considerare l'articolo come un flusso di testo, ovvero una sequenza di caratteri senza considerarne la struttura, e dividerlo in parte di uguale lunghezza. Tale divisione logica che caratterizza una tipologia di analisi, può essere integrata con un'altra che, ad esempio, tiene conto della divisione in sezioni. Tale divisione potrebbe prevedere la scelta di un insieme di sezioni, identificandole dal titolo, ed eseguire l'analisi dei riferimenti su di esse; questo darà dei risultati completamente differenti rispetto l'analisi sul flusso di testo. Alle due analisi descritte se ne può aggiungere una terza che potrebbe trattare la divisione dell'articolo in insiemi di sezioni, in tal modo potremmo identificare la distribuzione dei riferimenti sulle sezioni, dove ogni sezioni è caratterizzata solamente dalla sua posizione nell'articolo e non dal titolo. Infine si può pensare pensare, ancora, ad analisi che mirano ad estrarre informazioni sul modo in cui i riferimenti bibliografici sono presentati nel testo, ovvero se sono principalmente citati singolarmente o se si preferisce citare un numero N di riferimenti bibliografici insieme.
L'analizzatore dei titoli è uno strumento necessario nel momento in cui abbiamo un insieme di articoli da analizzare. Questo permette di ottenere i titoli delle sezioni più rilevanti, ovvero che superano una certa percentuale di presenza all'interno dell'insieme analizzato. I titoli risultanti saranno utili per una tipologia di analisi la cui divisione logica si basa sulle sezioni caratterizzate dai titoli.
Il generatore dei grafici permette l'esportazione dei dati, ottenuti dall'analizzatore dei riferimenti bibliografici, in pagine HTML che contengono i grafici delle distribuzioni e altre informazioni di contorno. Le schermate grafiche prodotte da questa componente rendono intuitivi i risultati e permettono una visione complessiva dell'analisi effettuata.
La tesi è strutturata nel seguente modo:

- il capitolo 2 descrive gli studi esistenti che riguardano l'analisi dei riferimenti;
- il capitolo 3 descrive tutte le analisi possibili discutendo i concetti di base senza scendere nei dettagli tecnici;
- il capitolo 4 descrive le varie componenti che compongono Diribia, e le loro inter-

azioni;

- il capitolo 5 descrive il sistema dal punto di vista progettuale e tratta l'implementazione delle parti più complesse;
- il capitolo 6 descrive i risultati ottenuti e le conclusioni derivate;
- il capitolo 7 descrive i possibili sviluppo futuri.

1.1 Glossario dei termini

E' importante definire correttamente le terminologie di base, ricorrenti in questa tesi, che utilizzeremo per la trattazione dei problemi, delle soluzioni e per la descrizione di Diribia.
Con i termini "riferimenti" e "riferimenti bibliografici" indichiamo i riferimenti a risorse esterne che popolano le citazioni poste nel testo, non le stringhe bibliografiche elencate, solitamente, nella sezione finale degli articoli. Le risorse esterne possono essere risorse web oppure articoli pubblicati, questa differenza è poco rilevante ai fini dello studio poiché verranno trattare entrambi nel medesimo modo.
In modo analogo ai riferimenti, con i termini "citazioni" e "citazioni bibliografiche" indichiamo la citazioni nel testo di uno o più riferimenti bibliografici insieme. Fondamentale è capire la relazione tra riferimenti bibliografici e citazioni bibliografiche: una citazione è un insieme con cardinalità uno o superiore di riferimenti mentre, uno o più riferimenti possono essere contenuti nella medesima citazione.
A meno che non sia precisato, non ci riferiremo mai a riferimenti interni al testo come, ad esempio, i riferimenti a figure o tabelle.
Infine, con "distribuzione" dei riferimenti bibligrafici intendiamo la distribuzione delle posizioni dei riferimenti nel testo, essa, a seconda della divisione fatta sull'articoli esaminato, produrranno risultati differenti.

Capitolo 2

Analisi delle citazioni: stato dell'arte

Nel capitolo seguente verranno analizzati gli studi riguardanti i riferimenti bibliografici per chiarire come possono essere usati a seconda degli obiettivi che si vogliono raggiungere.

2.1 Indicatori sul numero di citazioni

L'analisi delle citazioni è un argomento molto discusso e trattato da tempo. Tra i primi lavori troviamo quelli di Garfield [9, 10] dove viene esposto il concetto di citation index applicato agli articoli scientifici e di impact factor, pensato per valutare l'influenza, nell'ambiente scientifico, di un articolo o di una rivista. In un altro studio [8], Garfield descrive il Science Citation Index (SCI), un citation index per gli articoli appartenenti alla letteratura scientifica. In questo lavoro viene descritto, anche, come ottenere il Science Citation Index, naturalmente con la tecnologia esistente in quel periodo. Successivamente nascono altri citation index, specifici per alcune discipline come, ad esempio, il Social Sciences Citation Index (SSCI) per gli articoli nell'ambito delle scienze sociali. L'obiettivo principale dei diversi citation index è permettere l'estrapolazione di informazioni relative ad articoli, riviste o autori attraverso un indice che tiene conto delle citazioni fatte nei lavori pubblicati successivamente a quello di nostro interesse.
Uno studio riguardante la valutazione di indici per il confronto tra differenti autori o riviste è quello di J. E. Hirsch [11]. In questo lavoro è descritto l'indice di Hirsch o h-index. Tale indice permette di valutare la "fama" accademica di una rivista o di un autore identificando un valore N, tale che un autore ha indice N se almeno N lavori tra quelli che ha pubblicato sono stati citati almeno N volte ciascuno.

2.2 Citazioni Web

Grazie alla rapida espansione e alla facilità con cui è possibile reperire delle risorse, il web ha cambiato radicalmente il sistema di comunicazione all'interno della comunità scientifica. Al giorno d'oggi le citazioni web, ovvero le citazioni di risorse reperibili tramite un certo URL (Uniform Resource Locator), sono molto frequenti. Questo ha dato il via ad un nuovo filone di studi, relativo ai vantaggi, svantaggi e ai problemi che portano con sé le citazioni web.
Tra i problemi principali vi è quello della persistenza che non è garantita infatti, individui o organizzazioni possono abbandonare le pagine web, spegnere i server o, semplicemente, rinominare le risorse per renderla inaccessibile.
Lo studio affrontato da Lawrence [13] si occupa di analizzare le citazioni web al fine di ottenere statistiche riguardanti le risorse non più reperibili.
Un altro lavoro interessante è quello compiuto da McCown [14]. I risultati ottenuti da questo studio hanno evidenziato come gli URLs più propensi ad essere inaccessibili sono quelli che puntano a risorse:

- con particolari domini di primo livello;
- che usano porte non standard;
- che puntano a risorse con estensioni deprecate.

Un terzo studio riguardante le citazioni web [20] tratta l'analisi delle quantità e dell'accessibilità degli URL utilizzati all'interno degli abstract nelle pubblicazioni scientifiche appartenenti al database MEDLINE. La scelta degli abstract non è banale poiché, avendo un ruolo rilevanti nelle pubblicazione scientifiche, anche gli URL al proprio interno avranno un ruolo fondamentale e quindi, l'inaccessibilità diventa un problema da risolvere.
Anche lo studio [21] tratta le citazioni web però, spostando il focus dalla trattazione dei problemi di persistenza e di affidabilità verso le distribuzioni delle risorse web. In questo lavoro sono presentati grafici interessanti che rappresentano le distribuzioni delle risorse web citate all'interno degli articoli scientifici. Tra le distribuzioni più rilevanti che sono trattate da questo studio, troviamo quelle sui domini di primo livello e sulle estensioni.

2.3 Parsing delle stringhe bibliografiche

Nella comunità scientifica vi sono molti lavori riguardanti le varie metodologie utilizzate per il parsing e la classificazione delle stringhe bibliografiche. Prima di procedere, però, è bene definire due terminologie di base:

- con "stringa bibliografica" viene identificato un elemento appartenente alla lista dei riferimenti che si trova, solitamente, in una sezione alla fine di un documento.

Una stringa bibliografica contiene tutte le informazioni relative ad una risorsa citata(nome autore, titolo, anno pubblicazione, etc);

- con "citazione" viene identificato il riferimento posto nel testo relative ad una o più stringhe bibliografiche.

L'obiettivo di questo filone di studi è l'analisi delle stringhe bibliografiche, la loro divisione in token o elementi semplici e, infine, l'etichettatura di questi ultimi per attribuirgli un valore semantico e classificarli.
Diversi lavori trattano il parsing delle stringhe bibliografiche tra cui [2,15,22,23]. Questi lavori presentano una caratteristica comune, ovvero l'utilizzo dei modelli CRFs al fine di segmentare ed etichettare le stringhe bibliografiche, invece un discriminante è la diversa classificazione dei token dovuta alle diverse classi di etichette utilizzate.

2.4 Semantic Publishing

I recenti studi sul semantic publishing trattano l'analisi dei riferimenti bibliografici mediate ontologie e tecniche tipiche del semantic web. Il semantic publishing si occupa di migliorare l'interattività e l'usabilità nelle pubblicazioni scientifiche mediante l'uso di standard web moderni, compreso l'uso di ontologie per codificare la semantica sotto forma di metadata RDF machine-readable [18].
Il lavore di David Shotton [19] tratta le potenzialità e i cambiamenti connessi al semantic publishing. Shotton ipotizza che il semantic publishing porterà sostanziali benefici alla comunicazione scientifica nel lontano 2009.
Un lavoro interessante in quest'ottica [4], descrive la panoramica delle ontologie più rilevanti, ad oggi, disponibili. In seguito, identifica le quelle necessarie per la descrizione di tutti gli aspetti dei riferimenti bibliografici (contesto, riferimento nel testo, lista dei riferimenti bibliografici, etc.) e infine, propone uno strumento che mira ad aumentare l'utilizzo della semantica nel mondo della comunicazione scientifica. Tale strumento, REnhancer, prende in input una lista di stringhe bibliografiche testuali e produce un insieme di triple RDF conforme con le ontologie scelte. Lo studio tratta i problemi relativi alla scelta delle ontologie e descrive, accuratamente, le due ontologie scelte ed utilizzate da REnhancer. Un'altro studio [3] presenta e tratta la descrizione di un algoritmo, chiamato CiTaLo. Tale strumento si pone l'obiettivo di dedurre automaticamente la funzione delle citazioni. Tale studio nasce dalla necessità di comprendere le ragioni alla base di una citazione, esse possono essere molteplici, ad esempio: offrire informazioni di background, per esporre idee e metodi precedentemente trattati da altri autori o, anche, criticare o rifiuta un lavore precedente. Come per lo studio precedente, quest'ultimo si basa su un'ontologia, scelta tra quelle che si occupano della descrizione della natura delle citazioni nelle pubblicazioni scientifiche, CiTO [16].

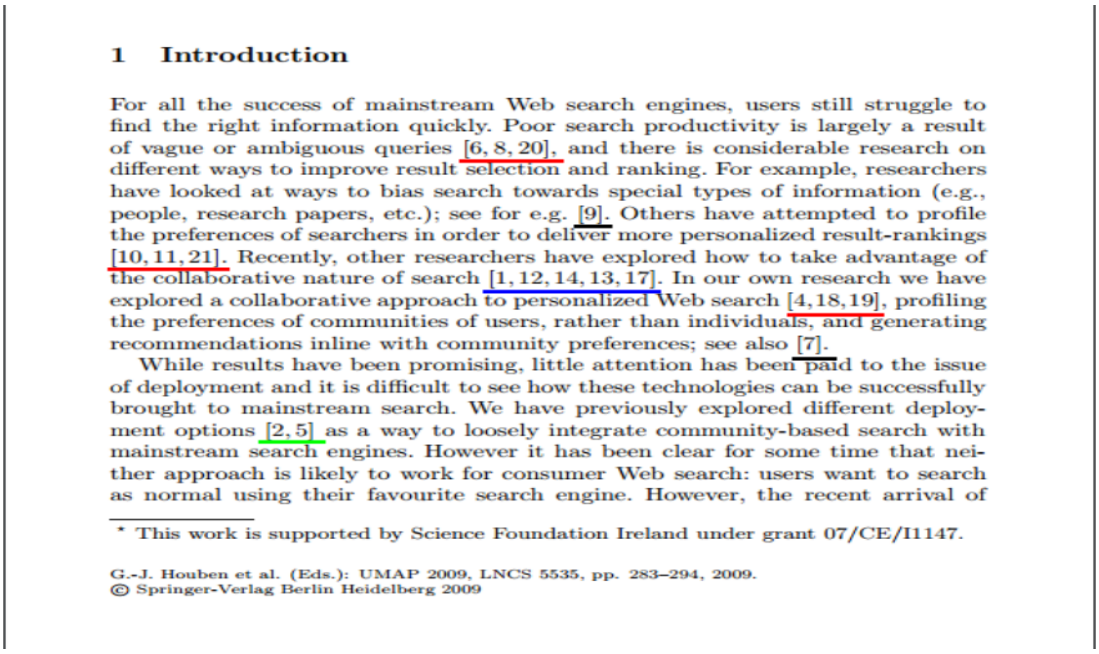
Capitolo 3

Diribia: Applicazione per l'analisi dei riferimenti bibliografici

Nel capitolo corrente verrà descritto il problema che sta alla base dello studio affrontato, le possibili soluzione e, infine, la soluzione utilizzata per risolverlo.

3.1 Problema

"Dove sono posizionati e come sono rappresentati i riferimenti bibliografici nelle pubblicazioni scientifiche?"
Per rispondere a questa domanda bisogna pensare a delle metodiche che hanno l'obiettivo di individuare dove sono posizionati i riferimenti bibliografici e, in che modo essi sono rappresentati. Per "rappresentati" intendiamo se i riferimenti vengono citati singolarmente o insieme(citazione multipla). La figura 3.1 mostra una sezione di un articolo scientifico dove sono evidenti i diversi modi con cui i riferimenti bibliografici possono essere citati; vi sono citazioni di un riferimento(sottolineati con il colore nero), citazioni di cinque riferimenti insieme(colore blu) e altre varianti evidenziate da differenti colori.
Invece la figura 3.2 mostra una parte della sezione "Introduzione" presa da un'altro articolo scientifico. In tale figura possiamo notare dei riferimenti bibliografici sintatticamente differenti ma che hanno lo stesso significato dei riferimenti visti in precedenza poiché, per rispondere alla domanda che ci siamo posti, questa differenza sintattica non è rilevante. L'esempio ci mostra che la maggior parte dei riferimenti bibliografici presenti sono citati singolarmente, eccetto dei casi in cui troviamo citazioni multiple contenenti due riferimenti ognuna.



1 Introduction

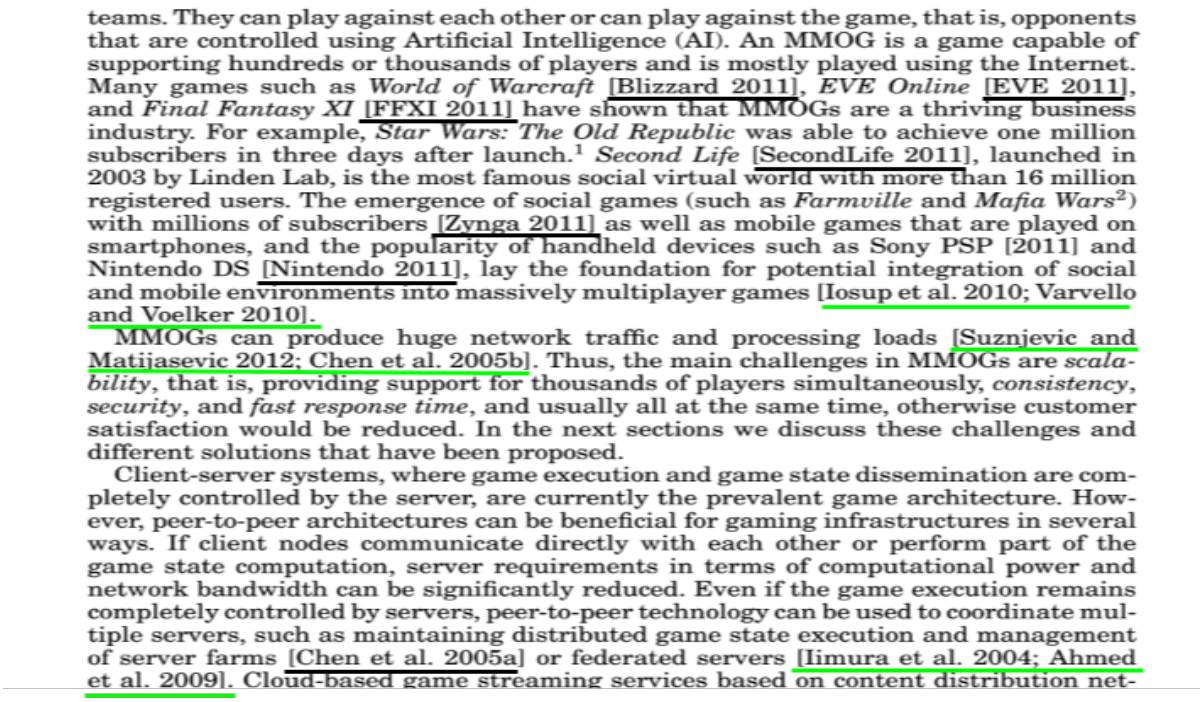
For all the success of mainstream Web search engines, users still struggle to find the right information quickly. Poor search productivity is largely a result of vague or ambiguous queries [6, 8, 20], and there is considerable research on different ways to improve result selection and ranking. For example, researchers have looked at ways to bias search towards special types of information (e.g., people, research papers, etc.); see for e.g. [9]. Others have attempted to profile the preferences of searchers in order to deliver more personalized result-rankings [10, 11, 21]. Recently, other researchers have explored how to take advantage of the collaborative nature of search [1, 12, 14, 13, 17]. In our own research we have explored a collaborative approach to personalized Web search [4, 18, 19], profiling the preferences of communities of users, rather than individuals, and generating recommendations inline with community preferences; see also [7].

While results have been promising, little attention has been paid to the issue of deployment and it is difficult to see how these technologies can be successfully brought to mainstream search. We have previously explored different deployment options [2, 5] as a way to loosely integrate community-based search with mainstream search engines. However it has been clear for some time that neither approach is likely to work for consumer Web search: users want to search as normal using their favourite search engine. However, the recent arrival of

⋆ This work is supported by Science Foundation Ireland under grant 07/CE/I1147.

G.-J. Houben et al. (Eds.): UMAP 2009, LNCS 5535, pp. 283–294, 2009.

Figura 3.1: Esempio 1 - parte della sezione "Introduzione" che presenta diverse citazioni



teams. They can play against each other or can play against the game, that is, opponents that are controlled using Artificial Intelligence (AI). An MMOG is a game capable of supporting hundreds or thousands of players and is mostly played using the Internet. Many games such as *World of Warcraft* [Blizzard 2011], *EVE Online* [EVE 2011], and *Final Fantasy XI* [FFXI 2011] have shown that MMOGs are a thriving business industry. For example, *Star Wars: The Old Republic* was able to achieve one million subscribers in three days after launch.[1] *Second Life* [SecondLife 2011], launched in 2003 by Linden Lab, is the most famous social virtual world with more than 16 million registered users. The emergence of social games (such as *Farmville* and *Mafia Wars*[2]) with millions of subscribers [Zynga 2011] as well as mobile games that are played on smartphones, and the popularity of handheld devices such as Sony PSP [2011] and Nintendo DS [Nintendo 2011], lay the foundation for potential integration of social and mobile environments into massively multiplayer games [Iosup et al. 2010; Varvello and Voelker 2010].

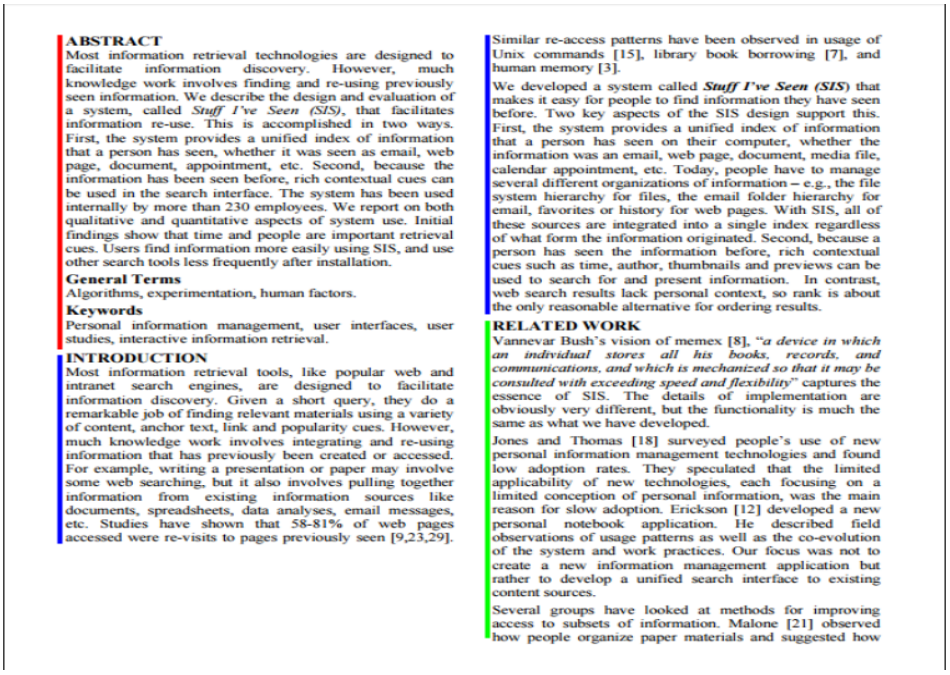
MMOGs can produce huge network traffic and processing loads [Suznjevic and Matijasevic 2012; Chen et al. 2005b]. Thus, the main challenges in MMOGs are *scalability*, that is, providing support for thousands of players simultaneously, *consistency*, *security*, and *fast response time*, and usually all at the same time, otherwise customer satisfaction would be reduced. In the next sections we discuss these challenges and different solutions that have been proposed.

Client-server systems, where game execution and game state dissemination are completely controlled by the server, are currently the prevalent game architecture. However, peer-to-peer architectures can be beneficial for gaming infrastructures in several ways. If client nodes communicate directly with each other or perform part of the game state computation, server requirements in terms of computational power and network bandwidth can be significantly reduced. Even if the game execution remains completely controlled by servers, peer-to-peer technology can be used to coordinate multiple servers, such as maintaining distributed game state execution and management of server farms [Chen et al. 2005a] or federated servers [Iimura et al. 2004; Ahmed et al. 2009]. Cloud-based game streaming services based on content distribution net-

Figura 3.2: Esempio 2 - parte della sezione "Introduzione" che presenta diverse citazioni

3.2 Un modello per studiare la distribuzione dei riferimenti bibliografici nelle pubblicazioni scientifiche

Avendo ben chiaro il problema da risolvere, possiamo pensare a delle soluzioni ragionevoli per risolverlo.
Una prima soluzione, per rispondere alla domanda "dove sono posizionati i riferimenti bibliografici", potrebbe basarsi sul conteggio dei riferimenti posizionati in ogni singola sezione. Dividendo logicamente l'articolo in sezioni e valutando i riferimenti bibliografici presenti in ognuno di essi, otterremo una situazione simile a quella descritta dalla figura 3.3. Tale figura mostra ogni sezione evidenziata, lateralmente, da un colore differente.



ABSTRACT
Most information retrieval technologies are designed to facilitate information discovery. However, much knowledge work involves finding and re-using previously seen information. We describe the design and evaluation of a system, called *Stuff I've Seen (SIS)*, that facilitates information re-use. This is accomplished in two ways. First, the system provides a unified index of information that a person has seen, whether it was seen as email, web page, document, appointment, etc. Second, because the information has been seen before, rich contextual cues can be used in the search interface. The system has been used internally by more than 230 employees. We report on both qualitative and quantitative aspects of system use. Initial findings show that time and people are important retrieval cues. Users find information more easily using SIS, and use other search tools less frequently after installation.

General Terms
Algorithms, experimentation, human factors.

Keywords
Personal information management, user interfaces, user studies, interactive information retrieval.

INTRODUCTION
Most information retrieval tools, like popular web and intranet search engines, are designed to facilitate information discovery. Given a short query, they do a remarkable job of finding relevant materials using a variety of content, anchor text, link and popularity cues. However, much knowledge work involves integrating and re-using information that has previously been created or accessed. For example, writing a presentation or paper may involve some web searching, but it also involves pulling together information from existing information sources like documents, spreadsheets, data analyses, email messages, etc. Studies have shown that 58-81% of web pages accessed were re-visits to pages previously seen [9,23,29].

Similar re-access patterns have been observed in usage of Unix commands [15], library book borrowing [7], and human memory [3].

We developed a system called ***Stuff I've Seen (SIS)*** that makes it easy for people to find information they have seen before. Two key aspects of the SIS design support this. First, the system provides a unified index of information that a person has seen on their computer, whether the information was an email, web page, document, media file, calendar appointment, etc. Today, people have to manage several different organizations of information – e.g., the file system hierarchy for files, the email folder hierarchy for email, favorites or history for web pages. With SIS, all of these sources are integrated into a single index regardless of what form the information originated. Second, because a person has seen the information before, rich contextual cues such as time, author, thumbnails and previews can be used to search for and present information. In contrast, web search results lack personal context, so rank is about the only reasonable alternative for ordering results.

RELATED WORK
Vannevar Bush's vision of memex [8], "*a device in which an individual stores all his books, records, and communications, and which is mechanized so that it may be consulted with exceeding speed and flexibility*" captures the essence of SIS. The details of implementation are obviously very different, but the functionality is much the same as what we have developed.

Jones and Thomas [18] surveyed people's use of new personal information management technologies and found low adoption rates. They speculated that the limited applicability of new technologies, each focusing on a limited conception of personal information, was the main reason for slow adoption. Erickson [12] developed a new personal notebook application. He described field observations of usage patterns as well as the co-evolution of the system and work practices. Our focus was not to create a new information management application but rather to develop a unified search interface to existing content sources.

Several groups have looked at methods for improving access to subsets of information. Malone [21] observed how people organize paper materials and suggested how

Figura 3.3: Parte di un articolo scientifico partizionata in sezioni

Da questo esempio riusciamo ad ottenere una distribuzione dei riferimenti bibliografici; nella sezione "abstract" abbiamo 0 riferimenti, in "introduction" 6 e nella sezione "related work" ne troviamo 4.
Un seconda soluzione potrebbe essere quella di non considerare la struttura in sezioni ma considerare l'articolo come un flusso di testo e quindi osservare come si posizionano i riferimenti bibliografici se dividiamo l'articolo in un numero stabilito di parti, tutte col medesimo numero di caratteri.
La figura 3.4 mostra la seconda soluzione, appena descritta, applicata allo stesso articolo

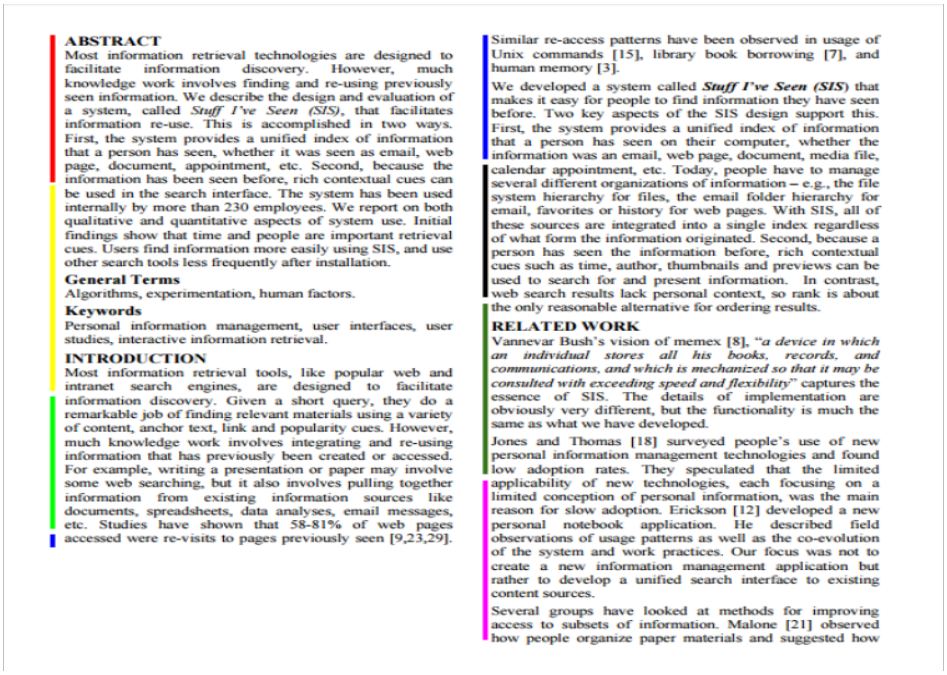


Figura 3.4: Parte di un articolo scientifico partizionata in 7 fette

utilizzato nell'esempio precedente. Con la divisione del testo in sette parti, le informazioni che riusciamo ad ottenere sono del tutto differenti rispetto a quelle estrapolate dalla figura 3.3. Le prime tre parti non presentano riferimenti bibliografici, nella parte caratterizzata dal colore blu troviamo 6 riferimenti, nella parte nera 0, nella parte verde 2 ed infine nella parte evidenziata con il colore viola 2.
Il totale dei riferimenti bibliografici presenti nei due esempi è ovviamente lo stesso ma, i due approcci utilizzati producono due distribuzioni differenti e, allo stesso tempo, ragionevoli.
I due casi, appena discussi, mettono in evidenza due aspetti importanti:

- le distribuzioni dei riferimenti bibliografici sono caratterizzati dal modo in cui viene diviso l'articolo (divisione logica);
- non esiste un'unica divisione logica corretta bensì, a seconda degli interessi, può essere più adatta una divisione piuttosto che un'altra.

Diribia è un'applicazione server-side che analizza le pubblicazioni scientifiche e mette a disposizione dell'utente un insieme di analisi sui riferimenti bibliografici che riguardano, non solo la loro distribuzione sul testo, ma anche il modo in cui essi sono citati all'interno dell'articolo.
Prima di introdurre queste tipologie di analisi bisogna, però, definire il concetto di divisione logica che sarà il discriminante principale, come visto nelle figure 3.3 e 3.4. Per

divisione logica si intende come l'articolo viene spezzettato, cioè diviso in parti che chiameremo parti logiche. Le divisioni logiche possono tener conto, o meno, della struttura dell'articolo.
Le analisi riguardanti le distribuzioni dei riferimenti bibliografici, quindi, hanno come oggetto le parti logiche in cui l'articolo è diviso; queste parti logiche sono determinate dalla divisione logica, specifica in ogni tipologia di analisi.
L'analisi relativa agli indici di aggregazione, invece, si occupa di descrivere in che modo i riferimenti sono citati all'interno degli articoli.
La ricerca e la descrizione delle divisioni logiche adottate è avvenuta per fasi successive, osservando le distribuzioni ottenute su campioni significativi di articoli e mettendo in evidenza le possibili varianti che avrebbero potuto portare differenti risultati.
La figura 3.5 mostra un semplice schema che evidenzia le analisi implementate in Diribia.

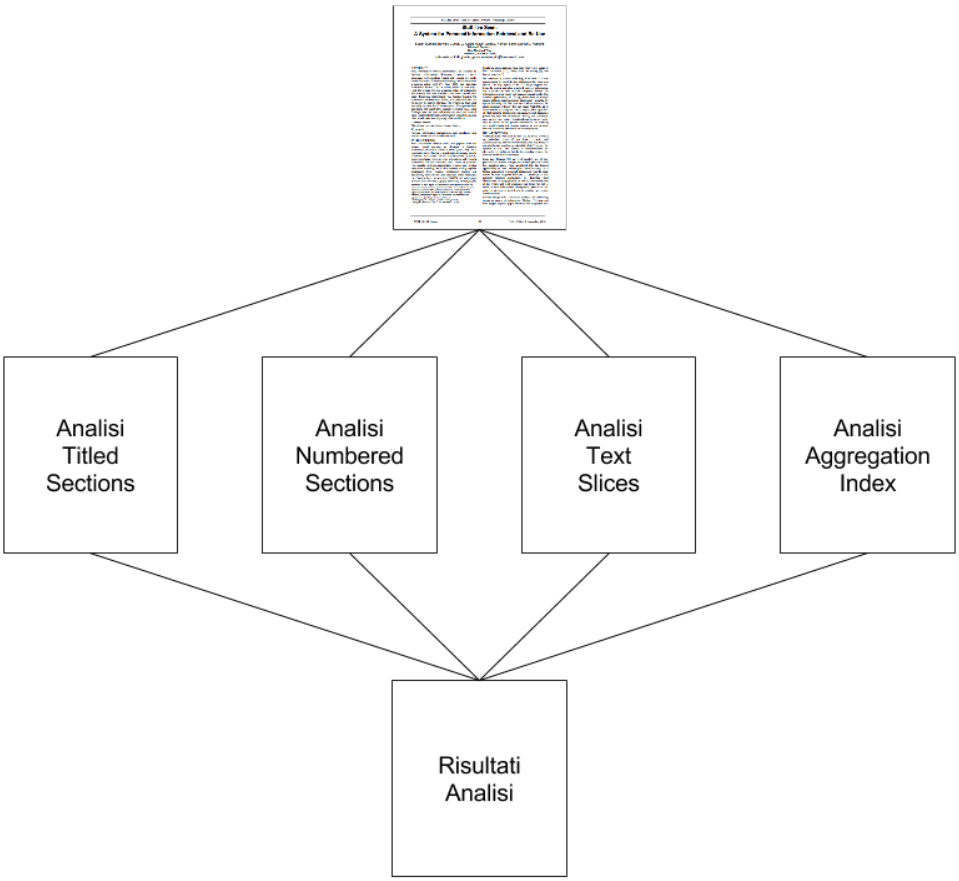


Figura 3.5: Schema delle analisi implementate in Diribia

3.3 Analisi Text Slices

L'analisi Text Slices è molto intuitiva, si pensi ad un articolo come ad un flusso di testo continuo senza considerarne la struttura (titolo, sezioni, paragrafi, riferimenti, etc.). A questo punto lo si divida in N parti con egual numero di caratteri e, immaginando di essere su un editor di testo, si evidenzi ognuna di queste con un colore diverso. Ogni parte di testo caratterizzata da uno specifico colore è l'idea di fetta che abbiamo utilizzato per la divisione logica. La figura 3.4, discussa precedentemente, mostra una parte di articolo scientifico divisa in 7 fette, dove ogni fetta è evidenziata da un diverso colore.
Questa tipologia di analisi si occupa di ricavare la distribuzione dei riferimenti su un numero N, arbitrario e definito a priori, di fette in cui l'articolo è stato logicamente diviso.

3.3.1 Text Slices su insiemi di articoli

Un domanda che nasce spontanea potrebbe essere: "se volessi analizzare un insieme di articoli anziché uno soltanto?".
L'analisi su un solo articolo non ci permette di fare assunzioni di nessun tipo su articoli riguardanti la stessa disciplina o articoli della stessa rivista o, ancora, articoli dello stesso autore.
Con le analisi applicate ad un insieme di articoli, invece, potremmo catturare trend di intere riviste o discipline così da poter fare, a posteriori, analisi più ampie; ad esempio:

- analizzare i risultati di insiemi di articoli riguardanti la stessa disciplina ma appartenenti a decenni differenti. Questo può mostrare come la letteratura per quella disciplina si sia evoluta;
- analizzare i risultati di insiemi di articoli appartenenti a diverse discipline; questo ci permette di catturare le differenze e le particolarità che le contraddistinguono;
- analizzare i risultati di insiemi di articoli riguardanti la stessa disciplina ma appartenenti a riviste differenti.

Analizzare un insieme di articoli vuol dire aggregare secondo un qualche criterio i risultati dei singoli articoli dell'insieme. Una soluzione è quella di definire un mapping tra le parti logiche dei diversi articoli in modo da aggregare i risultati relativi a queste parti logiche. La scelta del mapping è fondamentale e specifica per ogni analisi.
Le figure 3.6 e 3.7 mostrano due possibili tipologie di mapping tra le fette appartenenti a tre flussi di testo che simboleggiano tre articoli scientifici.
Il mapping descritto nella figura 3.6 non tiene conto della lunghezza di ogni articolo ma utilizza le fette definite per l'articolo più lungo come base per il mapping tra tutti gli articoli dell'insieme. A causa delle differenti lunghezze, questo mapping può far coincidere

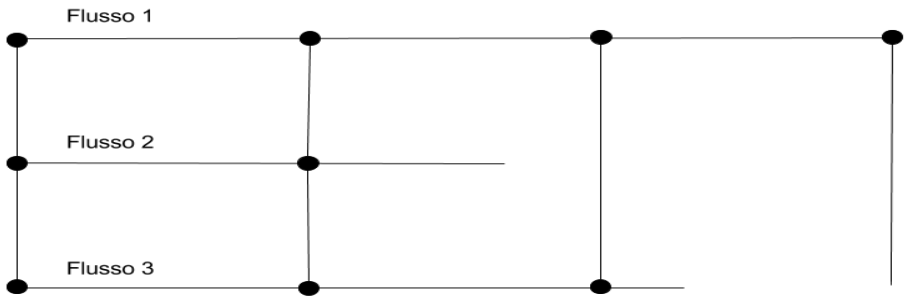


Figura 3.6: Possibile mapping tra le fette di tre flussi di testo

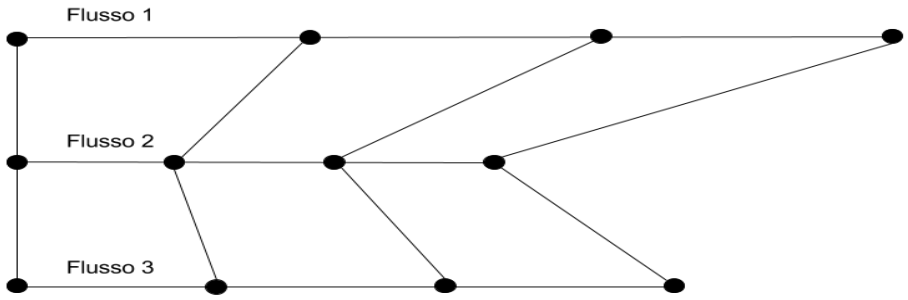


Figura 3.7: Mapping utilizzato da Diribia tra le fette di tre flussi di testo

risultati di fette totalmente identificate con risultati di fette parzialmente identificate con, ancora, risultati di fette inesistenti; in tal modo vengono resi inaffidabili i risultati aggregati.
Con il mapping descritto nella figura 3.7 il problema viene risolto. Ogni articolo, infatti, viene diviso nel medesimo numero di fette, dove ognuna avrà la propria lunghezza dipendentemente dalla lunghezza totale del relativo articolo. I risultati aggregati saranno ottenuti raggruppando i risultati di ogni articolo secondo il mapping definito, come mostrato in figura 3.7.
Gli articoli che presentano le fette con un numero di caratteri inferiore ad una costante, non verranno analizzati. Questo argomento sarà discusso nel capitolo 5.2.4.

3.4 Analisi Titled Sections

La divisione logica che caratterizza questa tipologia di analisi pone il focus sulla struttura dell'articolo e, principalmente, sui titoli delle sezioni che lo compongono. Sono oggetto di analisi, infatti, le distribuzioni dei riferimenti sulle sezioni caratterizzate da specifici titoli. Questi titoli possono essere scelti arbitrariamente creando un file json che verrà, poi, utilizzato dall'analizzatore durante l'esecuzione.
"Perché usare le sezioni e non un altro elemento presente nella struttura dell'articolo?". Le sezioni identificate da specifici titoli hanno un forte valore semantico, questo permette di comprendere meglio i risultati delle analisi. Ad esempio, se un numero alto di riferimenti si trova nella sezione "Related Work" mentre nessun riferimento viene rilevato nella sezione "Abstract", conoscendo i titoli di tali sezioni sarà facile capire le motivazioni di questa ipotetica distribuzione; se invece, non conoscessimo i titoli delle sezioni allora non potremmo fare alcuna osservazione di tipo semantico sulla distribuzione risultante.
Quando parliamo di sezione all'interno di questa tipologia di analisi, intendiamo una sezione ben precisa caratterizzata da un titolo.
Nella tabella 3.1 vengono mostrati i risultati dell'analisi Titled Sections applicata all'articolo raffigurato nella figura 3.3, assumendo di aver scelto come titoli delle sezioni da analizzare: "Abstract" e "Related Work".

Sezioni analizzate	**Distribuzione riferimenti bibliografici**
Abstract	0
Related Work	4
Other	6

Tabella 3.1: Distribuzione riferimenti bibliografici relativa alla figura 3.3

Come si può notare, la sezione con titolo "introduction" viene inglobata in una sezione fittizia chiamata "Other", questo poiché non è stata scelta come sezione da analizzare.
In generale, quindi, le distribuzioni saranno valutate sulle sezioni specificate dai titoli definiti a priori e, sulla sezione fittizia "Other". Quest'ultima rappresenta i risultati delle sezioni riconosciute come non rilevanti poiché aventi titoli non appartenenti a quelli scelti per l'analisi.

3.4.1 Titled Sections su insiemi di articoli

Per trattare il problema descritto nella sezione 3.3.1 per questa tipologia di analisi, viene presentato un esempio. Le tabelle 3.2 e 3.3 rappresentano le strutture di due ipotetici articoli scientifici con i relativi riferimenti bibliografici.

Sezioni articolo	**Riferimenti bibliografici**
Abstract	0
Introduction	6
Related Work	4

Tabella 3.2: Struttura del primo ipotetico articolo scientifico

Sezioni articolo	**Riferimenti bibliografici**
Abstract	1
Introduction	15
Review of the theory	4
Discussions and conclusion	2

Tabella 3.3: Struttura del secondo ipotetico articolo scientifico

A questo punto, proviamo ad effettuare un'analisi Titled Sections sui due ipotetici articoli appena descritti. Supponiamo che i titoli scelti per l'analisi siano: "Abstract", "Introduction" e "Discussions and Conclusion".
I risultati aggregati ottenuti sono riportati nella tabella 3.4.

Sezioni analizzate	**Distribuzione riferimenti bibliografici**
Abstract	0+1
Introduction	6+15
Discussions and conclusion	2
Other	4+4

Tabella 3.4: Analisi Titled Sections sugli articoli descritti dalle tabelle 3.2 e 3.3

Il mapping viene banalmente effettuato aggregando i risultati delle sezioni con il medesimo titolo, compresa la sezione fittizia "Other" che rappresenta le sezioni non trattate singolarmente.
Come si può notare, la sezione "Discussions and conclusion" è presente solo nel 50% degli articoli analizzati (ovvero in uno su due) ma questa informazione non è possibile

ricavarla dai risultati aggregati. Tale mancanza può essere colmata aggiungente ai risultati aggregati le informazioni relative alla percentuale di presenza delle sezioni all'interno dell'insieme analizzato.
Un problema potrebbe essere quello della definizione dei titoli relativi alle sezioni da analizzare in un contesto di centinaia o migliaia di articoli. Il rischio è di ottenere come risultati aggregati, nel caso pessimo, la sola sezione "Other". Come spiegato precedentemente, la sezione fittizia "Other" contiene i risultati di tutte le sezioni considerate non rilevanti, cioè le sezioni i cui titoli non sono tra quelli scelti per l'analisi.
Questo problema, quindi, è dato dalla scelta di titoli che hanno poca rilevanza all'interno dell'insieme di articoli da analizzare; per rilevanza, in questo contesto, si intende la percentuale di presenza di una determinata sezione (identificata dal titolo) all'interno di un insieme di articoli. Ad esempio, se una sezione ha una percentuale di presenza del 10% all'interno dell'insieme di articoli, analizzarla sarà inutile poiché non identifica una sezione che presenta risultati soddisfacenti e caratteristici per quell'insieme.
Per ovviare, almeno parzialmente, a questo problema, Diribia dispone di una componente per l'analisi e l'estrapolazione dei titoli più rilevanti che verrà descritto nella sezione 4.3.
Abbiamo usato il termine "parzialmente" poiché la componente appena introdotta non garantisce che vi sia uniformità delle sezioni scelte nell'intero insieme di articoli. Infatti, anche se i titoli vengono scelti accuratamente, le sezioni possono avere una percentuale di presenza molto alta ma non essere presenti in tutti gli articoli dell'insieme. Questo problema verrà discusso nel capitolo 4.2.2.

3.5 Analisi Numbered Sections

L'analisi Numbered Sections nasce dalla considerazione che possono esserci articoli i cui titoli sono molto specifici. In tal caso risulta difficile ottenere risultati rilevanti e, allo stesso tempo, generici al fine di poter fare confronti tra differenti articoli.

Sezioni articolo	**Riferimenti bibliografici**
Abstract	1
Structure of core scheme	3
Review of the theory	4
Discussions and conclusion	2

Tabella 3.5: Struttura del primo ipotetico articolo scientifico

Sezioni articolo	**Riferimenti bibliografici**
Introduction	8
Example use case	5
System components	0
Developer tools	2

Tabella 3.6: Struttura del secondo ipotetico articolo scientifico

Se osserviamo i due ipotetici articoli descritti dalle tabelle 3.5 e 3.6, ci accorgiamo, subito, che non vi sono sezioni comuni a entrambi. Ipotizziamo di procedere con un'analisi Titled Sections su entrambi gli articoli ipotetici per poi confrontare i risultati; questo risulterebbe impossibile poiché non sapremmo come metterli in relazione a causa delle differenze tra i titoli delle sezioni. Adesso, si pensi ad un'analisi che utilizzi la divisione in sezioni però tenendo conto solamente della posizione delle sezioni e non dei titoli che le caratterizza. Utilizzando questa possibile analisi, il problema descritto per l'esempio precedente verrebbe facilmente superato.
L'idea che sta alla base dell'analisi Numbered Sections, infatti, è di raggruppare le sezioni in insiemi e dividere l'articolo in N insiemi, con N arbitrario e definito a priori. E' importante sottolineare che la divisione logica che sta dietro questa analisi prende in considerazioni solo la posizione delle sezioni e non da alcuna rilevanza né al titolo né ad altri attributi. Le tabelle 3.7 e 3.8, di seguito, mostrano i risultati dell'analisi Numbered Sections con N pari a 2, applicata rispettivamente agli articoli definiti dalle tabelle 3.5 e 3.6.

	Sezioni accorpate	**Distribuzione riferimenti bibliografici**
Insieme di sezione 1	• Abstract • Structure of core scheme	1+3
Insieme di sezione 2	• Review of the theory • Discussions and conclusion	4+2

Tabella 3.7: Analisi Numbered Sections sull'articolo descritto dalla tabella 3.5

	Sezioni accorpate	**Distribuzione riferimenti bibliografici**
Insieme di sezione 1	• Introduction • Example use case	8+5
Insieme di sezione 2	• System components • Developer tools	0+2

Tabella 3.8: Analisi Numbered Sections sull'articolo descritto dalla tabella 3.6

3.5.1 Numbered Sections su insiemi di articoli

Per descrivere la soluzione al problema discusso nella sezione 3.3.1, ipotizziamo di prendere in input i due articoli descritti dalle tabelle 3.5 e 3.6. A questo punto procediamo con l'analisi di questi articoli singolarmente. I risultati sono già riportati nelle tabelle 3.7 e 3.8. Infine, applichiamo il mapping che consiste nel raggruppare i risultati degli insiemi di sezioni con il medesimo indice. La tabella 3.9 mostra i risultati aggregati ottenuti dopo che il mapping è stato effettuato.

	Distribuzione riferimenti bibliografici
Insieme di sezione 1	(1+3)+(8+5)
Insieme di sezione 2	(4+2)+(0+2)

Tabella 3.9: Risultati aggregati ottenuti dopo la fase di mapping tra le tabelle 3.7 e 3.8

Preso un insieme di articoli, l'unica variante tra gli insiemi di sezioni definiti per ogni articolo potrebbe essere la loro cardinalità poiché il numero di sezioni in un articolo può non essere multiplo del valore N fissato. Questo argomento verrà trattato nel capitolo 5.2.3. Gli articoli che presentano un numero di sezioni inferiore rispetto al numero degli insiemi in cui dovrebbero essere divisi non vengono analizzati.

3.6 Analisi Aggregation Index

Quest'ultima tipologia di analisi si differenzia dalle altre poiché non è caratterizzata da alcuna divisione logica. Prima della sua descrizione è necessario definire il concetto di aggregazione dei riferimenti. Le citazioni dei riferimenti bibliografici poste nel testo possono contenere singoli riferimenti o un insieme di riferimenti (citazione multipla). Nel caso vi siano citazioni singole consecutive, queste rappresentano, semanticamente, risorse riguardanti uno stesso argomento e verranno trattate come una citazione multipla. Il ruolo dell'indice di aggregazione, quindi, è di rappresentare il numero di riferimenti appartenenti ad ogni singola citazione, esso sarà uno se abbiamo una citazione singola o maggiore di uno nel caso di citazione multipla. Si rimanda alla figura 3.1 come ausilio alla definizione di indice di aggregazione.
Questa tipologia di analisi si occupa, quindi, di ricavare le distribuzioni con cui i riferimenti bibliografici sono aggregati, ovvero le distribuzioni degli indici di aggregazione. Grazie a tale analisi possiamo fare delle assunzioni che non riguardano la posizione dei riferimenti bibliografici bensì il modo in cui, nei diversi ambiti scientifici, si preferisce o si necessita aggregarli.
Di seguito sono riportate tre parti di testo presi da articoli scientifici che mostrano lo stesso indice di aggregazione, secondo la definizione riportata sopra. Nel primo caso abbiamo due riferimenti appartenenti alla stessa citazione e viene facile intuire che l'indice di aggregazione sia uguale a 2; nel secondo e nel terzo caso, invece, i due riferimenti appartengono a citazioni consecutive ma, presumibilmente, inerenti ad uno stesso argomento e quindi, elaborati come fossero un'unica citazione con indice di aggregazione uguale a 2.

```
B-splines were first introduced by [24,25] and [...]

The general approach is thoroughly explained in [21] and in [22] [...]

Their algebraic and isogeometric properties can be found in [26][37] [...]
```

3.6.1 Aggregation Index su insiemi di articoli

Come soluzione al problema descritto nella sezione 3.3.1 è stato identificato un mapping abbastanza semplice che si occupa di raggruppare i risultati relativi ai medesimi indici di aggregazione.

Capitolo 4

Architettura di Diribia

Il seguente capitolo prende in esame l'architettura di Diribia, definendo i compiti delle singole componenti e i relativi input e output.
La figura 4.1 mostra nel dettaglio l'architettura. Le componenti interne sono evidenziate dallo sfondo grigio, le frecce identificano gli output prodotti e, i segmenti rappresentano gli input per le componenti.
Dalla figura possiamo notare i ruoli delle singole componenti. Il convertitore prende in input gli articoli, li converte e dà in output i medesimi articoli nel formato intermedio, chiamato PLUS. Questi articoli nel formato PLUS saranno l'input per il core dell'applicazione, ovvero l'analizzatore dei riferimenti. Il formato degli articoli supportati in input è solamente XML ma, sviluppi futuri potrebbero prevedere la conversione anche di articoli in altri formato, ad esempio PDF.
Esso al suo interno prevedere le diverse analisi, descritte nel capitolo precedente. A seconda delle richieste dell'utente verrà applicata l'analisi scelta.
Infine i risultati prodotti dall'analizzatore dei riferimenti saranno dei file in json. Tali file vengono resi ben leggibili dal generatore dei grafici che si occupa, appunto, della loro trasformazione in schermate con grafici in modo da rendere intuitiva la lettura delle distribuzioni.
Da notare che l'analizzatore dei riferimenti e affiancato dall'analizzatore dei titoli, il quale può essere utilizzato per produrre i file di configurazione per una particolare tipologia di analisi, la Titled Sections, che necessita dei titoli delle sezioni da analizzare.

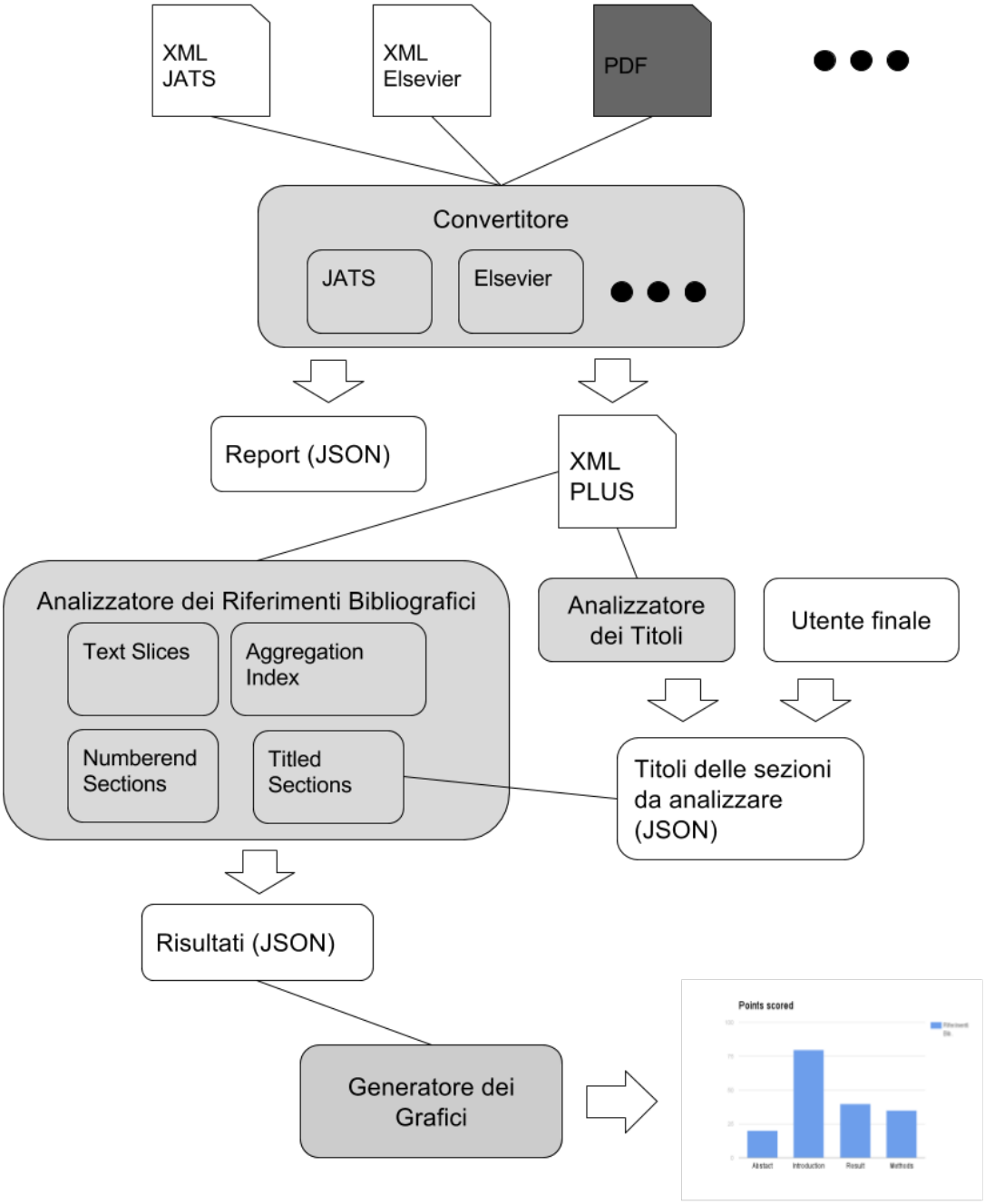


Figura 4.1: Architettura di Diribia

4.1 Convertitore

Il convertitore ha un ruolo fondamentale ai fini del funzionamento di Diribia.
Gli articoli che un utente potrebbe voler sottoporre ad analisi di un qualche tipo, possono essere caratterizzati da diversi formati, come si può notare nella figura 4.1.
Il formato supportato da Diribia è XML con DTD JATS o Elsevier. Gli articoli XML non utilizzano tutti gli stessi elementi e non hanno la medesima struttura. Infatti, cambiando il DTD(Document Type Definition), la cui funzione è quella di definire gli elementi di cui si compone un documento XML [5], possono presentarsi differenti caratteristiche strutturali. Gli analizzatori di Diribia non possono gestire contemporaneamente diversi DTD ma necessitano di un formato unico e ad hoc, sul quale poter reperire le informazioni senza conoscere la struttura originale di ogni singolo articolo.
Il convertitore si occupa, quindi, di colmare la distanza tra articoli scientifici e analizzatori. Esso prende in input gli articoli scientifici, nel formato XML, indipendentemente dal loro DTD e dalle loro particolarità e li converte in articoli XML con il medesimo contenuto ma senza alcun collegamento sintattico con i DTD originali.
Gli articoli convertiti presentano elementi, attributi ed una struttura standard; queste caratteristiche vanno a definire il formato ad hoc di Diribia che chiameremo PLUS.
Il formato PLUS è pensato per essere semplice e per rendere facilmente reperibili specifiche informazioni necessarie agli analizzatori.
Non vi è alcuna garanzia che tutti gli articoli in input vengano convertiti nel formato PLUS; la mancata conversione può essere causata dalla struttura non riconosciuta (XML con DTD non supportati) o dalla mancanza di riferimenti bibliografici che, per scelte implementative, è un requisito necessario per la conversione di ogni articolo. Le informazioni relative al numero di articoli convertiti e non convertiti sono reperibili nel report prodotto dal convertitore; la figura 4.2 ne fornisce un esempio.

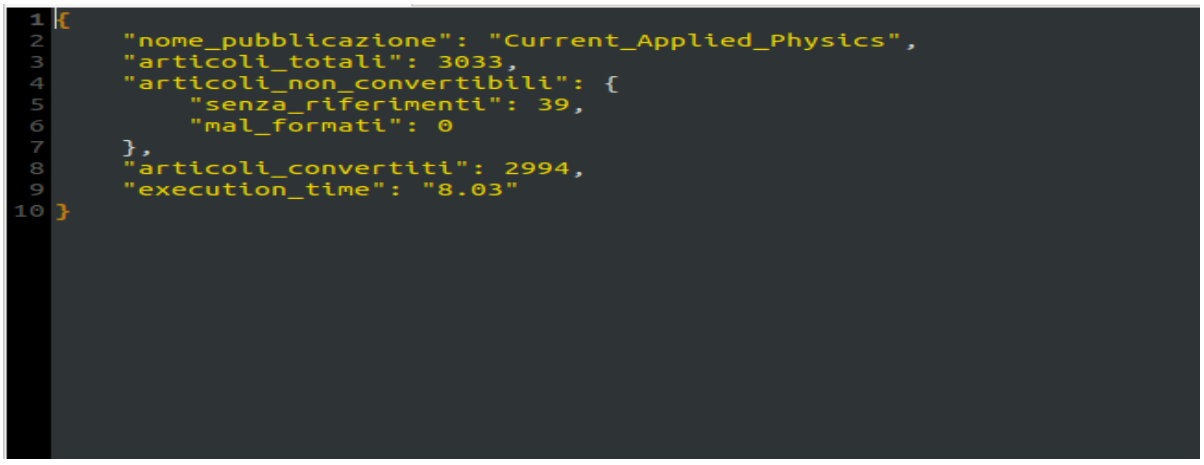


Figura 4.2: Esempio di output prodotto dal convertitore

Il convertitore è il punto di forza di questa applicazione perché la rende flessibile, in grado di essere utilizzata su diversi articoli, senza preoccuparsi della loro struttura. E' una componente modulare e questo rende semplice migliorare il supporto ai DTD già implementati o aggiungere il supporto a nuovi DTD oltre JATS ed Elsevier. Per la discussione riguardante la sua implementazione si rimanda alla sezione 5.
In precedenza abbiamo discusso i problemi dovuti alle differenze strutturali dei diversi DTD; la tabella 4.1 riassume tutte le differenze, rilevanti ai fini dello studio, tra i due formati supportati e le soluzioni adottate dal formato PLUS.
Questa tabella è il risultato di consultazioni alle documentazioni ufficiali di JATS [12] ed Elsevier [6, 7] e, anche, di analisi fatte su vari articoli al fine di estrapolare le caratteristiche che non sono descritte nelle documentazioni.

	JATS	**Elsevier**	**PLUS**
Tag principale	article	article, converted-article, simple-article	root
Tag parte iniziale contenente titolo, abstract, etc.	front	head, simple-head	
Tag abstract	abstract	abstract	abstract
Tag parte centrale contenente le sezioni	body	body	
Tag sezioni	sec	section	section
Tag riferimenti	xref	cross-ref, cross-refs	references
Attributo che caratterizza i rif. bibliografici	ref-type	refid	

Tabella 4.1: Differenze sintattiche tra i DTD JATS, Elsevier ed il formato PLUS

Il formato PLUS, come si può notare dalla tabella 4.1, non presenta elementi corrispettivi a tutti quelli elencati; la sua struttura è semplice e consiste nel solo elemento root, avente come nodi figli tutti gli elementi di nostro interesse ovvero le sezioni e l'abstract.

La tabella 4.2, invece, ci mostra la struttura adottata dal formato PLUS confrontata con le strutture dei DTD JATS ed Elsevier.

JATS	**Elsevier**	**PLUS**
article front abstract body sec_1 sec_2 sec_n	[simple-][converted-]article [simple-]head abstract body sections section_1 section_2 section_n	root abstract section_1 section_2 section_n

Tabella 4.2: Differenze tra le strutture dei DTD JATS, Elsevier ed il formato PLUS

Per completare la discussione su questa componente verrà mostrato un esempio per capire come i riferimenti bibliografici vengono convertiti. Di seguito sono riportate tre sequenze XML; le prime due, mostrano possibili sintassi per i riferimenti bibliografici, rispettivamente, di Elsevier e JATS; la terza sequenza, invece, mostra come i riferimenti bibliografici visti nei primi due XML, sono sintatticamente rappresentati nel formato PLUS. Non ci soffermiamo sul particolare testo "???4]]]" che compare nella terza sequenza XML poiché sarà oggetto di discussione nel capitolo 5.2.4.

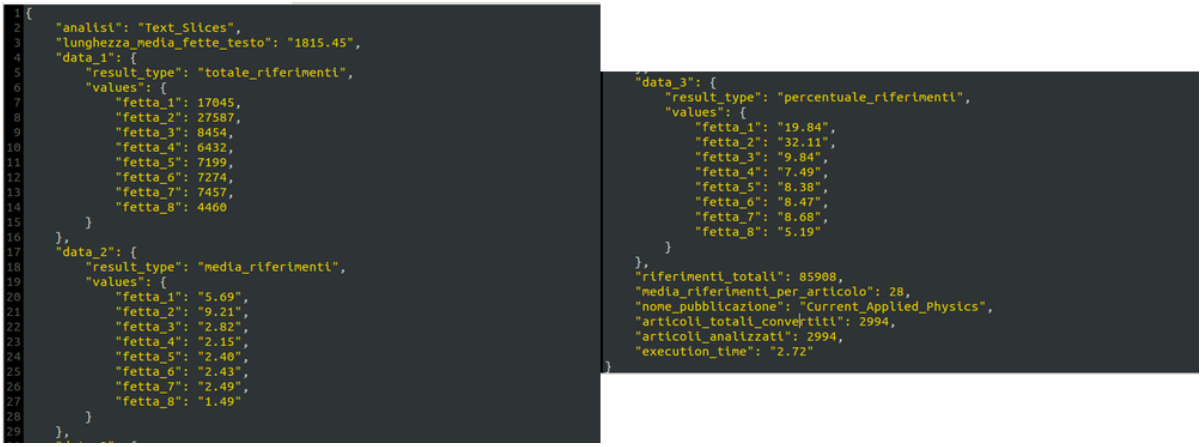
```
<ce:cross-refs refid=``bib1 bib2 bib3 bib4">
        [1-4]
</ce:cross-refs>
```

```
<xref rid=``B1" ref-type=``bibr">      1 </xref>
-
<xref rid=``B3" ref-type=``bibr">      3 </xref>
,
<xref rid=``B4" ref-type=``bibr"> 4 </xref>
```

```
<plus:references refid=``bib1 bib2 bib3 bib4" ref_num=``4">
        ???4]]]
</plus:references>
```

4.2 Analizzatore dei Riferimenti Bibliografici

I punti di discussione nel seguente capitolo sono: i risultati che l'analizzatore permette di ricavare, le eccezioni riguardanti le singole analisi, infine gli output che vengono prodotti. Questi ultimi vengono creati in formato json è sono necessari per la fase successiva ovvero l'esportazione grafica. La figura 4.3 mostra un esempio di output per l'analisi Text Slices su un insieme di articoli.



```
{
    "analisi": "Text_Slices",
    "lunghezza_media_fette_testo": "1815.45",
    "data_1": {
        "result_type": "totale_riferimenti",
        "values": {
            "fetta_1": 17045,
            "fetta_2": 27587,
            "fetta_3": 8454,
            "fetta_4": 6432,
            "fetta_5": 7199,
            "fetta_6": 7274,
            "fetta_7": 7457,
            "fetta_8": 4460
        }
    },
    "data_2": {
        "result_type": "media_riferimenti",
        "values": {
            "fetta_1": "5.69",
            "fetta_2": "9.21",
            "fetta_3": "2.82",
            "fetta_4": "2.15",
            "fetta_5": "2.40",
            "fetta_6": "2.43",
            "fetta_7": "2.49",
            "fetta_8": "1.49"
        }
    },
    "data_3": {
        "result_type": "percentuale_riferimenti",
        "values": {
            "fetta_1": "19.84",
            "fetta_2": "32.11",
            "fetta_3": "9.84",
            "fetta_4": "7.49",
            "fetta_5": "8.38",
            "fetta_6": "8.47",
            "fetta_7": "8.68",
            "fetta_8": "5.19"
        }
    },
    "riferimenti_totali": 85908,
    "media_riferimenti_per_articolo": 28,
    "nome_pubblicazione": "Current_Applied_Physics",
    "articoli_totali_convertiti": 2994,
    "articoli_analizzati": 2994,
    "execution_time": "2.72"
}
```


Figura 4.3: Esempio di output per l'analisi Text Slices su 8 fette di testo

L'output si può logicamente dividere in due parti, una standard per tutte le tipologie di analisi, l'altra dipendente da esse.
la parte standard, invece, è così composta:

- tipologia di analisi effettuata;
- numero dei riferimenti totali;
- media dei riferimenti per articolo;
- nome della pubblicazione analizzata;
- numero degli articoli analizzabili, cioè tutti gli articoli convertiti;
- numero degli articoli effettivamente analizzati;
- tempo di esecuzione.

La parte dipendente dall'analisi, invece, è descritta dettagliatamente nelle sottosezioni che seguiranno.
L'output può essere arricchito mediante l'attivazione della modalità "verbose" che permette di ricavare non solo i risultati aggregati ma anche i risultati relativi ad ogni singolo

articolo. La modalità "verbose" è disponibile per ogni tipologia di analisi.
L'analizzatore dei riferimenti bibliografici prende in input articoli nel formato PLUS.

4.2.1 Text Slices

Mediante l'analisi Text Slices sarà possibile ricavare:

- la distribuzione dei riferimenti bibliografici sulle fette di testo;
- la lunghezza media delle fette di testo.

Le distribuzioni sono valutate sui riferimento totali, sulle medie e sulle percentuali.
Ricavando la lunghezza media delle fette di testo forniamo un'informazione rilevanti all'utente. Infatti la lunghezza delle fette può cambiare ad ogni articolo a secondo della lunghezza totale ($lungh\_totale\_articolo_i$) e del numero di fette in cui si vuole dividere ogni articolo (N).
La formula 4.1 mostra le relazioni tra le variabili appena descritte dove L rappresenta la lunghezza delle fette per l'articolo i-esimo. Valutando questa formula per ogni articolo dell'insieme analizzato e calcolando la media dei risultati ottenuti si ricava la lunghezza media delle fette.

$$L = \frac{lungh\_totale\_articolo_i}{N} \tag{4.1}$$

Il numero di fette in cui dividere un articolo non presenta alcun limite superiore però, se il valore scelto determina una lunghezza delle fette inferiore ad una costante, descritta nel capitolo 5, quell'articolo verrà scartato. Le informazioni sugli articoli scartati sono presenti nell'output prodotto dall'analizzatore come si può notare dalla figura 4.3, mostrata precedentemente.

4.2.2 Titled Sections

Attraverso l'analisi Titled Sections sarà possibile ricavare:

- la distribuzione dei riferimenti bibliografici sulle sezioni scelte dall'utente o dall'analizzatore dei titoli(descritto nella sezione 4.3);
- informazioni sulla percentuale di presenze delle sezioni scelte tramite il titolo, all'interno dell'insieme analizzato.

Le distribuzioni sono valutate sui riferimento totali, sulle medie ponderate e sulle percentuali.
In questa tipologia di analisi, la distribuzione delle medie dei riferimenti bibliografici fa emergere un problema.
Le sezioni da analizzare spesso, anche se scelte attentamente così da superare una certa

percentuale di rilevanza, non sono presenti in tutti gli articoli dell'insieme e quindi, la media calcolate dividendo tutti i riferimenti di una determinata sezione per il numero totale degli articoli darebbe un risultato non corretto poiché quella sezione non è presente in tutti gli articoli.
Per ovviare a questo problema la distribuzione delle medie è sostituita dalla distribuzione delle medie ponderate. Con quest'ultima, infatti, tutti i riferimenti relativi ad una determinata sezione saranno divisi per i soli articoli che comprendono quella sezione al proprio interno.

4.2.3 Numbered Sections

Con l'analisi Numbered Sections sarà possibile ricavare:

- la distribuzione dei riferimenti bibliografici sugli insiemi di sezioni;
- la cardinalità media degli insiemi di sezioni.

Le distribuzioni sono valutate sui riferimento totali, sulle medie e sulle percentuali.
La necessità di ricavare la cardinalità media è dovuta al fatto che essa può cambiare ad ogni articoli, a secondo del numero totale di sezioni che lo compongono ($sezioni\_articolo_i$) e del numero di insiemi in cui si vogliono dividere gli articoli (N).
La formula 4.2 mostra le relazioni tra le variabili appena descritte dove C rappresenta la cardinalità degli insiemi per l'articolo i-esimo. Valutando questa formula per ogni articolo dell'insieme analizzato e calcolando la media dei risultati ottenuti si ricava la cardinalità media degli insiemi di sezioni.

$$C = \frac{sezioni\_articolo_i}{N} \tag{4.2}$$

Se il numero di insiemi di sezioni, in cui dividere un articolo, supera il numero di sezioni presenti al suo interno, quest'ultimo verrà scartato. Le informazioni sugli articoli scartati sono presenti tra i risultati prodotti dall'analizzatore.

4.2.4 Aggregation Index

Mediante l'analisi Aggregation Index è possibile ricavare:

- la distribuzione degli indici di aggregazione, descritti nella sezione 3.6;
- l'indice di aggregazione medio.

L'indice di aggregazionemedio (*indice_medio*) è ricavato mettendo in relazione il numero totale di riferimenti bibliografici ($riferimenti\_totali$) con il numero di citazioni dei riferimenti bibliografici ($riferimenti\_aggregati$) che rappresentano il numero dei riferimenti bibliografici aggregati aventi cardinalità maggiore o uguale ad uno.
La formula 4.3 definisce la relazione tra le variabili appena definite:

$$indice\_medio = \frac{riferimenti\_totali}{riferimenti\_aggregati} \tag{4.3}$$

In una prima fase, abbiamo pensato a questa analisi per la sola valutazione dell'indice medio di aggregazione ma, durante i test, questa informazione si è rivelata insufficiente per avere una visione corretta del modo in cui i riferimenti bibliografici sono aggregati.
Analizzando le due situazioni proposte di seguito, si può intuire il problema che sta dietro l'incompletezza dell'informazione derivata dal solo indice medio.
La prima situazione ci mostra tre citazioni di riferimenti bibliografici con indici di aggregazione differenti dove la media tra questi è 3.

```
<cross-refs refid=``b5 b6 b7 b8 b9 b10 b11">[5-11]</cross-refs>
<cross-refs refid=``b3">[3]</cross-refs>
<cross-refs refid=``b1">[1]</cross-refs>
```

Nella seconda situazione troviamo altre tre citazioni di riferimenti bibliografici ma questa volta, tutti, con gli stessi indici di aggregazione; la media è ancora una volta 3.

```
<cross-refs refid=``b5 b6 b7">[5-7]</cross-refs>
<cross-refs refid=``b15 b16 b17">[15-17]</cross-refs>
<cross-refs refid=``b25 b26 b27">[25-27]</cross-refs>
```

E' evidente come questi due casi producano lo stesso risultato essendo situazioni del tutto differenti. Questo è dovuto al fatto che le loro differenze sono mascherato da uno stesso indice medio di aggregazione. Per ovviare alla perdita di informazioni abbiamo deciso di aggiungere la distribuzione degli indici di aggregazione ai dati ricavabili mediante questa analisi.

4.3 Analizzatore dei Titoli

Definiamo innanzitutto gli obiettivi che hanno portato all'implementazione di questa componente:

- permettere analisi Titled Sections più accurate;
- offrire all'utente una funzionalità generica e completamente indipendente dal contesto dei riferimenti bibliografici.

L'analizzatore permette di estrarre, dato un insieme di articoli nel formato PLUS, i titoli delle sezioni più ricorrenti all'interno dell'insieme. E' possibile adattare l'analisi dei titoli alle specifiche necessità in quanto l'utente può settare la percentuale minima di presenza che, i titoli in output, devono rispettare relativamente all'insieme di articoli.
In output, l'analizzatore dei titoli darà due file json: uno contenente tutti i titoli che hanno superato una percentuale minima, costante, del 10%; l'altro contiene i titoli che hanno superato la percentuale settata dall'utente. Entrambi i file contengono anche le informazioni relative al numero di presenze del titolo all'interno dell'insieme analizzato. Le figure 4.4 e 4.5 mostrano un esempio dei due file json prodotti dall'analizzatore.

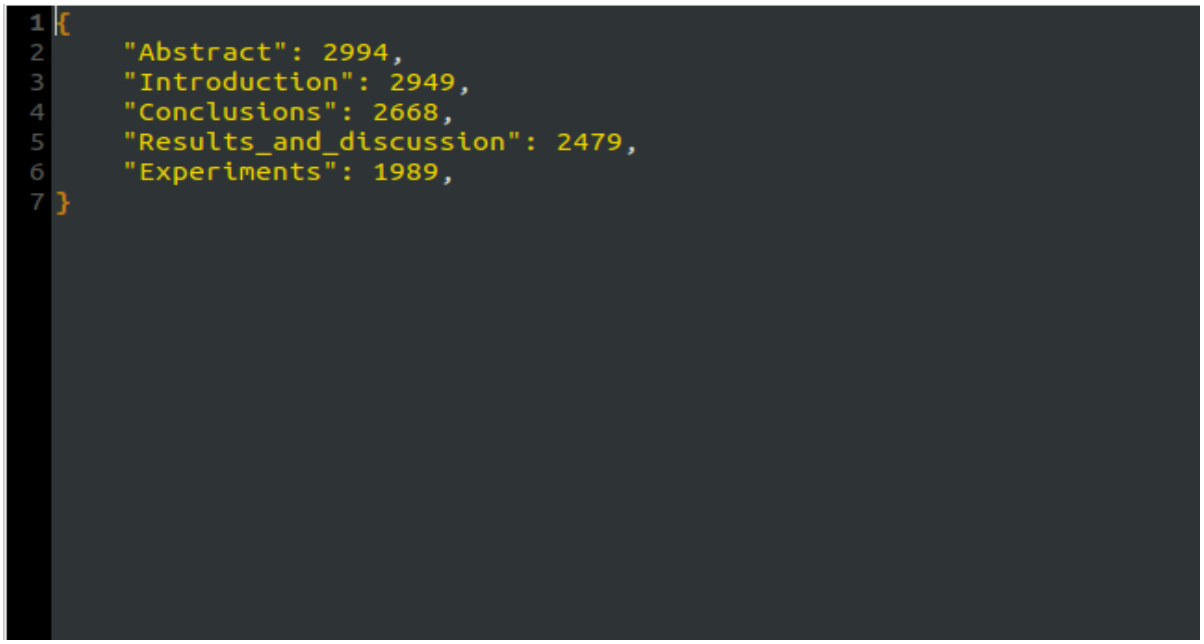


Figura 4.4: Json contenente i titoli con percentuale di presenza superiore al 10%

Figura 4.5: Json contenente i titoli che soddisfano la percentuale di presenza settata (70% nell'esempio)

4.4 Generatore dei Grafici

Questa componente si occupa dell'esportazione grafica dei risultati ottenuti dall'analizzatore dei riferimenti bibliografici. Il generatore dei grafici, prende in input i file json ottenuti dall'analizzatore dei riferimenti ed è in grado di generare schermate contenenti grafici relativi alle distribuzioni dei riferimenti ed informazioni esterne ai grafici e specifiche per ogni tipologia di analisi, che chiameremo "di contorno".
Ad esempio, per l'analisi Aggregation Index, il generatore mostrerà il grafico con la distribuzione degli indici e, come informazione di contorno, l'indice medio di aggregazione. Per l'analisi Titled Sections, invece, mostrerà il grafico con la distribuzione dei riferimenti sulle sezioni scelte e, come informazione di contorno, i dati sulla percentuale di presenza delle sezioni scelte, all'interno dell'insieme di articoli analizzato. Questo esempio, mostra che possiamo ottenere risultati differenti, da parte del generatore, a seconda delle diverse tipologie di analisi.
Grazie all'esportazione grafica riusciamo ad avere una rappresentazione intuitiva dei risultati prodotti dall'analizzatore, cosa impossibile mediante la consultazione dei risultati in formato json.
La figure 4.6 e 4.7 mostrano due esempi di output generati a partire, rispettivamente, dall'analisi Text Slices e Titled Sections. Possiamo notare come essi si differenziano al fine di mostrare tutte le informazioni rilevanti per le specifiche analisi.

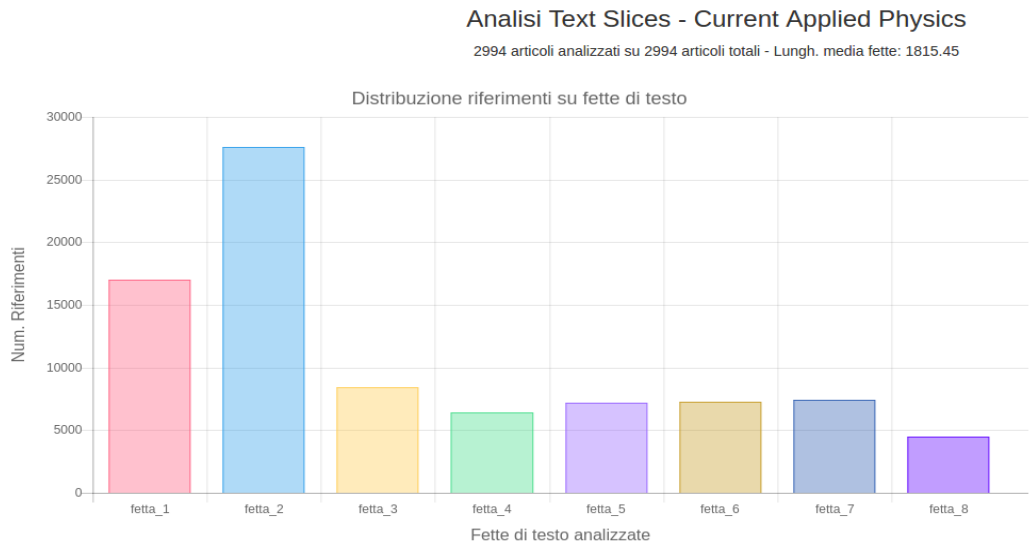


Figura 4.6: Output prodotto dal generatore dei grafici a partire dai risultati dell'analisi Text Slices, mostrati nella figura 4.3

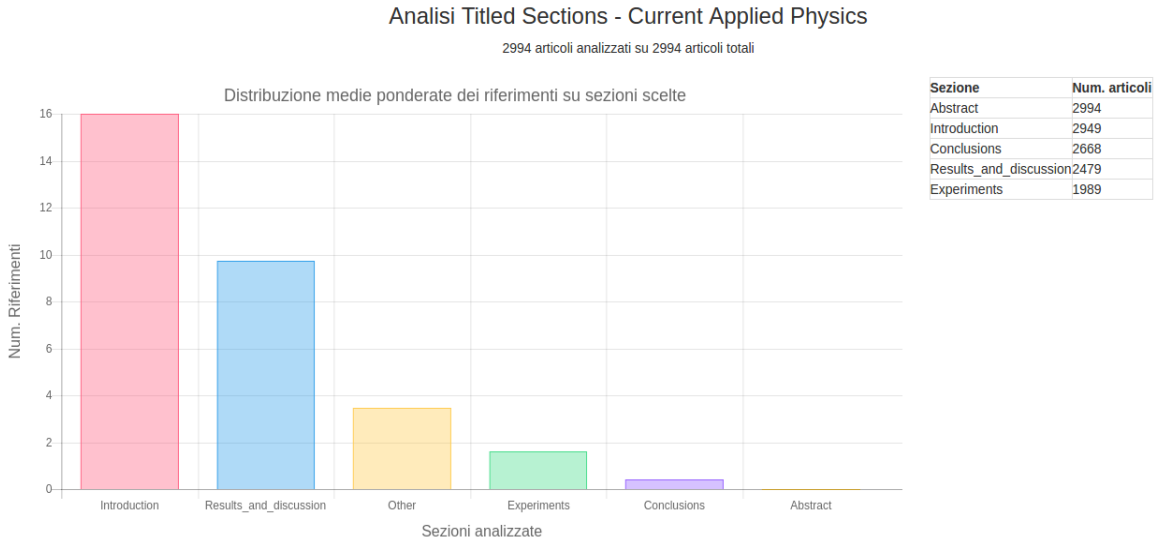


Figura 4.7: Esempio di output prodotto dal generatore dei grafici per l'analisi Titled Sections

Capitolo 5

Implementazione

Questo capitolo si occupa di fornire informazioni generali su tutto il sistema per poi analizzare nel dettaglio le scelte implementative più rilevanti.

5.1 Vista complessiva del sistema

Diribia è un'applicazione composta da diverse componenti come mostrato nella figura 4.1. Ogni componente è implementata da uno o più moduli e il tutto è completamente scritto con il linguaggio PHP.
Solamente il modulo per la generazione dei grafici è scritto oltre che in PHP, anche con HTML, CSS e JavaScript; questa caratteristica, che lo contraddistingue dagli altri moduli, è dovuto alla gestione della parte grafica. Tale modulo, infatti, produrrà in output un file HTML completo di CSS e JavaScript che aperto tramite un apposito lettore, mostrerà la schermata grafica generata. La parte dei grafici è gestita in JavaScript mediante la libreria Chart.js [1] che si occupa della loro formattazione.
Una componente molto rilevante in Diribia è il convertitore. Essa si occupa della conversione degli articoli XML in input, in articoli XML con formato PLUS. Questo componente è implementata da una classe, che raccoglie le caratteristiche comuni a tutti i DTD supportanti, estesa da più sottoclassi, una per ogni DTD supportato, che implementano la gestione della conversione da uno specifici DTD al formato PLUS.
Questo struttura rende semplice l'aggiunta del supporto a nuovi DTD poiché equivale a creare una nuova sottoclasse senza mettere mano, quindi, a classi già esistenti.
Una scelta progettuale importante è la separazione tra l'analizzatore dei titoli è l'analizzatore dei riferimenti bibliografici. Questa separazione garantisce una corretta divisione logica dei lavori ed, anche, l'indipendenza e il riutilizzo, in altri contesti, di entrambi gli analizzatori.Diribia fornisce un'interfaccia a riga di comando con la quale è possibile utilizzare tutte le componenti, definendo directory di input e output ed eventuali parametri per personalizzare i risultati.

5.2 Problematiche principali

Di seguito verranno trattate le problematiche più rilevanti e le soluzioni adottate per risolverle.

5.2.1 Tempi risposta dell'analisi dei titoli

L'analizzatore dei titoli prende in esame, per ogni articolo in input, tutte le sue sezioni estrapolando i relativi titoli. Questi ultimi verranno inseriti nell'array associativo "relevant_titles" dove le chiavi sono i titoli analizzati, mentre i valori sono i numeri di presenze all'interno dell'insieme di articoli. I titoli vengono inseriti in questo array secondo questi criteri:

- se il titolo supera una certa percentuale di similitudine, calcolata grazie alla funzione "similar_text", con un elemento dell'array allora il valore di quell'elemento viene incrementato di uno;
- se nessun elemento dell'array ha un certa percentuale di similitudine con il titolo analizzato, quest'ultimo viene inserito nell'array con valore uno.

Fondamentale per questa analisi è la seguente regola: ogni chiave dell'array "relevant_titles" può essere incrementata al più di uno dall'analisi dei titoli di un articolo.
L'array "tmp" si occupa di tenere memoria dei titoli che sono già stati trovati nell'articolo presente e che, quindi, sono già stati incrementati o aggiunti nell'array "relevant_titles", in questo modo mi assicuro di rispettare la regola appena descritto.

```
/*Ciclo per ogni articoli con formato PLUS in input*/
$tmp=[];

$all_titles = $xpath->query('/plus:root/plus:section/plus:section-title');
$abstract = $xpath->query('/plus:root/plus:abstract');
if(!empty($abstract)){
        $relevant_titles["Abstract"]++;
        $tmp["Abstract"] = 1;
}

foreach($all_titles as $title){
        $title_matching = false;
        $titleSec = $title->nodeValue;
        $maxPerc = 0;
                foreach($relevant_titles as $key=>$value){
                        similar_text($titleSec,$key,$perc);
                        if ($perc >= 51){
                                if ($perc > $maxPerc){
                                        $maxPerc = $perc;
                                        $actualKey = $key;
```

```
                                }
                                $title_matching = true;
                        }
                }
                if ($title_matching){
                        if(!isset($tmp[$actualKey])){
                                $tmp[$actualKey] = 1;
                                $relevant_titles[$actualKey]++;
                        }
                }
                else{
                        $tmp[$titleSec] = 1;
                        $relevant_titles[$titleSec]++;
                }
}
```

Il problema che nasce, osservando il codice appena proposto, è che l'array "relevant_titles" diventa sempre più grande e, inoltre, per tutti i titoli delle sezioni di ogni articolo, l'algoritmo dovrà ciclare su tutto l'array.
La soluzione trovata è la "potatura" o riduzione dell'array così da riuscire a mantenere dimensioni adeguate per ridurre i tempi di risposta senza, però, perdere accuratezza nei risultati. Segue, il codice che si occupa della definizione delle fasi di riduzione. Nel codice si potrà notare l'utilizzo di due costanti, entrambe definite dopo vari test che miravano a trovare il giusto rapporto tra accuratezza dei risultati e tempi di risposta.

```
$value_for_stepDefinition = 1000;
$value_for_stepDefinition2 = 200;
$step_remove = [];
$num_step_removal = intval($num_article*$value_for_stepDefinition2)/
    $value_for_stepDefinition;
$soglia;

if($num_step_removal>1){
    $articleInterval_for_step = intval($num_article_input/
        $num_step_removal);
    $soglia = $articleInterval_for_step/4;
    for ($i=1;$i<$num_step_removal;$i++){
        $step_remove[] = $articleInterval_for_step*$i;
    }
}
```

Adesso introduciamo il codice che si occupa della vera e propria riduzione dell'array; il numero di riduzioni e i passi in cui eseguirle sono definiti nel codice sopra proposto.
Il codice seguente viene inserito all'interno del ciclo dell'array "relevant_titles" che viene mostrato nella prima parte di codice proposta in questa sezione. Tale scelta ci permette di eseguire le riduzioni dentro il normale svolgimento dell'analisi e non in un ciclo dedicato però, ci obbliga a tenere una variabile ("reduceArray_ok") poiché la riduzione deve essere

fatta nel ciclo del primo titolo dell'i-esimo articolo analizzato, se e solo se quest'ultimo è identificato come passo di riduzione.

```
/*ciclo dell'array relevant_titles*/
    if($num_step_removal>1 && in_array($articoli_analizzati,$step_remove)
       && !$reduceArray_ok){
        $tmp_boolean = true;
        if ($value < $soglia){
            unset($relevant_titles[$key]);
            continue;
        }
```

5.2.2 Riconoscimento delle varie sintassi XML per i riferimenti bibliografici

Il convertitore di Diribia si occupa di prendere articoli con qualsiasi DTD e trasformarli nel formato PLUS. Per ottenere questo risultato deve scansionare la struttura degli articoli in input per rilevare tutte gli elementi e gli attributi di cui necessita il formato PLUS, gli elementi più complessi da rilevare sono quelli che identificano i riferimenti bibliografici.
I riferimenti possono essere bibliografici oppure riferimenti che richiamano elementi dello stesso articolo, questa distinzione non viene fatta dai tag bensì da un attributo che è standard nel DTD JATS ma non lo è nel DTD Elsevier.
Adesso analizziamo la gestione di entrambi i DTD per vederne le particolarità e la loro gestione da parte del convertitore.

DTD Elsevier: Riconoscimento dei riferimenti bibliografici

Il DTD Elsevier è più semplice da gestire, esso consente solo due modi sintattici per esprimere i riferimenti: "cross-ref" per i riferimenti singoli e "cross-refs" per i riferimenti multipli dove il numero dei riferimenti aggregati è reperibile tramite l'attributo "refid".
Il problema è la difficoltà nel riconoscere se si tratta di un riferimento bibliografico o meno poiché vi è solo l'attributo "refid" che non ha un valore univoco ma è necessario per identificare i riferimenti bibliografici. Per ovviare a tale problema, si è condotta un'analisi su un certo numero di articoli per ricercare i valori possibili dell'attributo "refid". I risultati di questa analisi sono riportati nelle righe seguenti.

```
<ce:cross-ref refid=``BIB1">[1]</ce:cross-ref>
<ce:cross-ref refid=``B1">[1]</ce:cross-ref>
<ce:cross-ref refid=``bib1">[1]</ce:cross-ref>
<ce:cross-ref refid=``b1">[1]</ce:cross-ref>
<ce:cross-ref refid=``br1">[1]</ce:cross-ref>
```

Alla luce di questi risultati il codice per rilevare tutti i riferimenti bibliografici negli articoli con DTD Elsevier è il seguente.

```
$list_multi_refs = $this->xpath->query('.//ce:cross-refs[starts-with(
    @refid,\'b\') or starts-with(@refid,\'B\')]',$this->article);
foreach($list_multi_refs as $ref){
    /*istruzioni per creare la sintassi definita da PLUS*/
}

$list_multi_ref = $this->xpath->query('.//ce:cross-ref[starts-with(@refid
    ,\'b\') or starts-with(@refid,\'B\')]',$this->article);
foreach($list_multi_ref as $ref){
    /*istruzioni per creare la sintassi definita da PLUS*/
}
```

All'interno dei cicli vengono fatte delle operazione per convertire la sintassi e la struttura del DTD Elsevier nel formato PLUS come descritto nel capitolo 4.1.

DTD JATS: Riconoscimento dei riferimenti bibliografici

Il riconoscimento dei riferimenti bibliografici nel DTD JATS è piena di casistiche che sono state raccolte dopo varie analisi e fasi di test. In questo DTD non vi è il problema di distinguere riferimenti bibliografici e non poiché esiste un attributo con un valore standard, definito nella documentazione di JATS [12], per identificare se si tratti di un riferimenti bibliografico(ref-type = "bibr").
Di seguito verrà elencata una lista non esaustiva delle possibili sintassi che sono stati trovate e gestite.

```
1) [<xref rid=``5" ref-type=``bibr"> 5 </xref> - <xref rid=``8" ref-type
   =``bibr"> 8 </xref>]

2) [<xref rid=``8" ref-type=``bibr"> 8 </xref> - <xref rid=``5" ref-type
   =``bibr"> 5 </xref>]

3) [<xref rid=``5" ref-type=``bibr"> 5 </xref> - <xref rid=``7" ref-type
   =``bibr"> 7 </xref>, <xref rid=``8" ref-type=``bibr"> 8 </xref>]

4) [<xref rid=``5" ref-type=``bibr"> 5 </xref>, <xref rid=``6" ref-type=``
   bibr"> 6 </xref>, <xref rid=``7" ref-type=``bibr"> 7 </xref>]

5) (<xref ref-type=``bibr" rid=``B36">36</xref>; <xref ref-type=``bibr"
   rid=``B34">34</xref>; <xref ref-type=``bibr" rid=``B4">4</xref>)

6) <xref rid=``bib5 bib6" ref-type=``bibr"> [5,6] </xref> (sintassi simile
    a quella del DTD Elsevier)

7) <xref rid=``b0005 b0006 b0007 b0008" ref-type=``bibr"> [5-8] </xref> (
   sintassi simile a quella del DTD Elsevier)

8) <xref rid=``8" ref-type=``bibr"> 8 </xref>, [...] <xref rid=``25" ref-
   type=``bibr"> 25 </xref>

9) <xref rid=``8" ref-type=``bibr"> 8 </xref>, and <xref rid=``9" ref-type
   =``bibr"> 9 </xref>
```

Ognuno dei casi, sopra descritti, tranne la casistica numero (8) verrà convertito in un unico elemento "references" rappresentante una citazioni con più riferimenti. Questo perché le casistiche rappresentano citazioni multiple o citazioni singole consecutive che sono trattate come una citazione multipla secondo l'idea di indice di aggregazione descritta nella sezione 3.6.
Adesso, come per la trattazione di Elsevier, verrà proposto il codice che si occupa di

rilevare le possibili sintassi, sopra elencate, e alcune loro varianti. Il codice è diviso in due, una parte è composta da "generalizzaRiferimenti" il cui obiettivo è identificare i riferimenti, fare un'analisi preliminare e, se necessario, richiamare la funzione "lookingForNearRef", descritta nella seconda parte. Quest'ultima si occupa di rilevare se il riferimento successivo a quello passato in input fa parte della stesso citazione o meno, così da incrementare il numero di riferimenti aggregati.
Nel codice che segue, la variabile "verification_Value" è una costante che serve per scartare sintassi come quella vista al punto 8 e riconoscere sintassi simili al punto 9. Queste due sintassi sono molto simili ma semanticamente danno due informazioni completamente differenti che l'algoritmo deve riuscire a cogliere.
La variabile "ref_count" è definita nella superclasse e identifica il numero di riferimenti aggregati che si stanno analizzato ad ogni passo dell'esecuzione. Essa viene settata a zero ogni qual volta che l'algoritmo termina l'analisi di un gruppo di tag "xref", riconosciuti come appartenenti ad un'unica citazione. Ognuno di questi gruppi di tag "xref" è quindi convertiti in un solo elemento "references" con più riferimenti al suo interno che saranno specificati dalla variabile "ref_count".

```
protected function lookingForNearRef ($ref){
        $textBetween = $ref->nextSibling;
  if (is_null($textBetween)){
      $this->ref_count++;
      return $this->ref_count;
  }
  else{
    if (preg_match ('/^\s*-/', $textBetween->nodeValue)){
        $lengh = strlen($textBetween->nodeValue);
        $ref2 = $textBetween->nextSibling;
        if ($lengh < $this->verification_Value && !is_null($ref2) &&
           strcasecmp($ref2->nodeName, ``xref")==0){
            $ref_id = intval($ref->nodeValue);
            $ref2_id = intval($ref2->nodeValue);
            if($ref2_id < $ref_id)
                $this->ref_count += ($ref_id - $ref2_id);
            else
                $this->ref_count += ($ref2_id - $ref_id);
            $ref->parentNode->removeChild($ref);
            return (-2);
        }
        else{
            $this->ref_count++;
            return $this->ref_count;
        }
    }
    else if(preg_match('/^\s*--',$textBetween->nodeValue)){
        $lengh = strlen($textBetween->nodeValue);
        $ref2 = $textBetween->nextSibling;
```

```
        if ($lengh < $this->verification_Value && !is_null($ref2) &&
    strcasecmp($ref2->nodeName, ``xref")==0){
                $ref_id = intval($ref->nodeValue);
                $ref2_id = intval($ref2->nodeValue);
                if($ref2_id < $ref_id)
                    $this->ref_count += ($ref_id - $ref2_id);
                else
                    $this->ref_count += ($ref2_id - $ref_id);
                $ref->parentNode->removeChild($ref);
                return (-2);
            }
            else{
                $this->ref_count++;
                return $this->ref_count;
            }
        }
        else if (preg_match ('/^\s*,/', $textBetween->nodeValue)){
            $lengh = strlen($textBetween->nodeValue);
            $ref2 = $textBetween->nextSibling;
            if($lengh < $this->verification_Value && !is_null($ref2) &&
               strcasecmp($ref2->nodeName, ``xref")==0){
            $ref->parentNode->removeChild($ref);
            $this->ref_count++;
            return (-1);
        }
        else{
            $this->ref_count++;
            return $this->ref_count;
        }
    }
    else if (preg_match ('/^\s*;/', $textBetween->nodeValue)){
        $lengh = strlen($textBetween->nodeValue);
        $ref2 = $textBetween->nextSibling;
        if($lengh < $this->verification_Value && !is_null($ref2) &&
    strcasecmp($ref2->nodeName, ``xref")==0){
                $ref->parentNode->removeChild($ref);
                $this->ref_count++;
                return (-1);
            }
            else{
                $this->ref_count++;
                return $this->ref_count;
            }
        }
        else{
            $this->ref_count++;
            return $this->ref_count;
        }
    }
```

```
}
```

```
protected function generalizzaRiferimenti(){
  $list_multi_Xref = $this->xpath->query('./body//xref[@ref-type=\'bibr\']
      | ./front//xref[@ref-type=\'bibr\']',$this->article);
  foreach($list_multi_Xref as $ref){
    /*  se nell'attibuto rid ho il numero di riferimenti li conto.
     *  Esempio: rid=``b0001 b0010"  */
    $lastRef = $ref;
    $id_ref = $ref->getAttribute(``rid");
    $num_id_ref = split(`` ",$id_ref);
    if (count($num_id_ref)>1){
        /*istruzioni per creare la sintassi definita da PLUS*/
    }
    /* senno' devo fare l'analisi con l'ausilio della funzione
       lookingForNearRef poiche' l'attributo rid non mi da informazioni*/
    else{
        $data = $this->lookingForNearRef($ref);
                                /*se data != -1 vuol dire data contiene il
                                    numero di riferimenti aggregati e
                                   posso procedere a memorizzare queste
                                   informazioni nel riferimento corrente*/
        if($data != -1){
            /*istruzioni per creare la sintassi definita da PLUS*/
        }
        /*se invece data == -1 o == -2  posso continuare a ciclare poiche'
            potremmo avere altri rif. aggregati da aggiungere*/
    }
  }
}
```

5.2.3 Scelte implementative per l'analisi Numbered Sections

L'analisi Numbered Sections si basa sulla divisione dell'articolo in insiemi di sezione, il numero di insiemi è definito a priori dall'utente e non ha limiti superiori.
Come definito nella sezione 4.2.3, se il numero di sezioni di un articolo è inferiore al numero di insiemi in cui dovrebbe essere diviso, non avremmo ogni insieme con almeno una sezione e, quindi, quell'articolo verrà scartato.
Un altro caso da gestire nasce quando il numero di sezioni è superiore ma non è multiplo del numero di insiemi in cui si vuole dividere l'articolo. Se, ad esempio, abbiamo da analizzare un articolo con otto sezioni(l'abstract è elaborato come se fosse una sezione) con un'analisi Numbered Sections su tre insiemi di sezioni, dividendo il numero di sezioni per il numero di insiemi da creare otterremo che ogni insieme conterrà due sezioni e ne rimarranno due fuori. "Come gestiamo queste rimanenti due sezioni?".
Le opzioni per gestire le sezioni che rimango fuori dagli insiemi possono essere due:

inserire più sezioni in alcuni insiemi, secondo un certo criterio, in modo che nessuna sezioni rimanga fuori, oppure, inserire tutte le sezioni rimanenti in un unico insieme, in quest'ultima opzione deve essere deciso l'insieme adibito a contenere le sezioni restanti. L'opzione implementata in Diribia è la seconda, questa scelta è emersa dopo aver analizzato diversi insiemi di articoli, appartenenti anche a discipline differenti. Ci si è accorti che, in generale, il numero di riferimenti tende a diminuire drasticamente nella parte finale degli articoli e quindi nelle sezioni finali.
Questa osservazione ha condotto alla scelta dell'ultimo insieme come candidato a contenere tutte le sezioni rimaste fuori dalla divisione iniziale. L'insieme scelto conterrà, quindi, le sezioni definite dalla divisione iniziale più le sezioni rimanente. Tale scelta può portarlo ad avere una cardinalità superiore agli altri insiemi ma i risultati finali non mostreranno incoerenza tra gli insiemi di sezioni poiché le sezioni finali di un articolo hanno rilevanza minima ai fini delle distribuzioni dei riferimenti bibliografici.

5.2.4 Divisione logica in fette di testo e rilevamento dei riferimenti senza struttura XML

Come introdotto alla fine della sezione 4, il convertitore crea particolari tag "references", caratteristici del formato PLUS. Questi tag presentano come testo delle stringhe del tipo "???XXX]]]" dove XXX identificano il numero dei rifermenti aggregati. Di seguito viene riproposto un esempio visto precedentemente.

```
<plus:references refid=``bib1 bib2 bib3 bib4" ref_num=``4">
        ???4]]]
</plus:references>
```

Questo particolare testo è necessario nel momento in cui rimuoviamo tutta la struttura XML per ottenere solo il flusso di testo al fine di definire le fette, caratteristica dell'analisi Text Slices. Dopo aver fatto ciò, l'algoritmo deve ricercare tutti i riferimenti esistenti in ogni fetta di testo per estrapolarne la distribuzione. Quest'ultimo passaggio è possibile grazie alle informazioni testuali relative ai riferimenti bibliografici che sostituiscono l'elemento "references" utilizzato dalle altra analisi per rilevare i riferimenti.
L'algoritmo, per ricercare i riferimenti bibliografici, dovrà ricercare la stringa "???" nel testo, una volta trovata fare gli opportuni controlli per verificare se essa è seguita da un numero e dalla stringa "]]]". Il codice seguente, mostra l'implementazione del meccanismo appena descritto e la gestione dei casi in cui le fette di testo vengono spezzate su un riferimento, ovvero su un carattere della stringa "???XXX]]]". La funzione "text_slices_analysis" crea il flusso di testo, divide in fette(numero fette è il parametro in entrata) e crea un elemento per ogni fetta. La funzione "referencesCount_text_slices" richiama la funzione "strpos_all" per ottenere tutti i riferimenti di ogni fette. Dopo che i riferimenti gli sono ritornati, provvederà a rimuovere le stringhe "???" e "]]]" risalendo, così, al numero di riferimenti aggregati. Infine ritornerà l'array contenente i valori della

distribuzione dei riferimenti bibliografici sulle fette di testo.

```
private function strpos_all($haystack, $needle) {
  $offset = 0;
  $allpos = array();
  while (($pos = strpos($haystack, $needle, $offset)) !== FALSE) {
      $offset = $pos + 1;
      $test_string = substr($haystack,$pos,4);
      // se sono nel caso ``????XX" allora aumento l'offset di 1 e basta,
          al pross giro avro' ``???XX"
      if(strcasecmp($test_string,``????")==0)
          continue;
      $allpos[] = $pos;
  }
  return $allpos;
}
```

```
private function referencesCount_text_slices($dom){
  $array_Rif = array();
  $root = $dom->firstChild;
  foreach($root->childNodes as $child){
    $count = 0;
    $array_pos= $this->strpos_all($child->nodeValue, ``???");
    foreach ($array_pos as $value){
        /* Uso 7 come lunghezza partendo dal valore ``value" poiche' cosi
         * sono sicuro che avro' ???X]]] OPPURE ???XX]] OPPURE ???XXX] e
             quindi dopo aver usato
         * replace mi assicuro che mi rimarra solo X o XX o XXX cioe' il
             num. di rif. annidati.
         */
        $tmp = substr($child->nodeValue,$value,7);
        $tmp = str_replace(``?",``",$tmp);
        $tmp = str_replace(``]",``",$tmp);
        $count += intval($tmp);
    }
    $array_Rif[$child->nodeName] = $count;
  }
return $array_Rif;
}
```

```
public function text_slices_analysis($slices_num){
  $allText = utf8_encode($currentDoc->firstChild->nodeValue);

  if (intval(strlen($allText)/$slices_num)<9)
      return false;

  $doc_to_analyze = new DOMDocument('1.0', 'utf-8');
  $root = $doc_to_analyze->createElement(``root");
  $doc_to_analyze->appendChild($root);
```

```
$slice_length = intval(strlen($allText)/$slices_num);
$this->charNumber_for_eachSlices = $slice_length;
$remaining_char = strlen($allText)
$slices_array = [];

$skipped_char = 0;
/* skipped_char mi definisce(per ogni fetta) i caratteri che devo
   recuperare per l'eventuale divisione
 * precoce dovuto alla presenza di un punto interr. o un numero nella
    fette precedente */
for($i=0; $i<$slices_num; $i++){
  $pos_primo_carattere = ($slice_length*$i)+$skipped_char;
  $pos_last_char = ($slice_length*($i+1));
  $slice_length_current = $slice_length-$skipped_char;

  $skipped_char = 0;
  if($i+1 == $slices_num){ //se sto definendo l'ultima fetta aggiungo i
     caratteri derivati dal resto della divisione iniziale.
      $slice_length_current += $remaining_char;
      $pos_last_char += $remaining_char;
  }
  else{
      if(strcasecmp($allText{$pos_last_char},``?") == 0){
          $to_verify = substr($allText, $pos_last_char-2, 5);
          $string_start=strpos($to_verify,``???");
          if ($string_start !== false){
              $skipped_char += $string_start-3;
              $slice_length_current += $skipped_char;
          }
      }
      else if (is_numeric($allText{$pos_last_char})){

          $to_verify = substr($allText, $pos_last_char-5, 5);
          $string_start=strpos($to_verify,``???");
          if ($string_start !== false){
              $skipped_char += $string_start-6;
              $slice_length_current += $skipped_char;
          }
      }
  }

  //testo che identifica una fetta
  $text = substr($allText,$pos_primo_carattere,$slice_length_current);

  //creo un nuovo elemento con dentro il testo(fetta) dell'iesima
     iterata
  $id_fetta = $i+1;
```

```
    $slices_array[$i] = $doc_to_analyze->createElement(``fetta_".$id_fetta
        );
    $slices_array[$i]->appendChild($doc_to_analyze->createTextNode($text))
        ;

    $root->appendChild($slices_array[$i]);
  }
  return $this->referencesCount_text_slices($doc_to_analyze);
}
```

Rilevante è il controllo fatto all'inizio della funzione "text_slices_analysis" per verificare se la dimensione delle fette è inferiore ai nove caratteri, se così fosse, l'elaborazione verrebbe fermata e l'articolo scartato.
Questo limite è necessario per il contesto in cui ci poniamo, ovvero basato su riferimenti riconoscibili testualmente. Se ogni fetta avesse dimensione inferiore ai nove caratteri perderemmo le informazioni relative ai riferimenti poiché essi sono rilevabili, solamente, mediante una stringa di lunghezza pari a nove caratteri.

Capitolo 6

Risultati

Nel capitolo seguente saranno descritte le riviste utilizzate per questo studio e, in seguito, saranno riportati e discussi i risultati prodotto da Diribia solo per alcune riviste tra quelle a nostra disposizione.
La tabella 6.1 descrive tutte le riviste utilizzate, evidenziando il numero di articoli che le compongono, l'anno della pubblicazione e i riferimenti totali al loro interno.

Rivista	**Anni pubblicazione**	**Numero articoli**	**Numero riferimenti bibliografici**
7 Riviste di biomedicina e biologia (PMC)	2000 - 2016	52797	2610334
29 Riviste di Psichiatria (PMC)	2001 - 2016	5283	365845
Journal of Computational Science	2010 - 2013	269	10447
Current Applied Physics	2002 - 2013	3033	85908
Web Semantics: Science, Services and Agents on the World Wide Web	2003 - 2014	367	17165

Tabella 6.1: Caratteristiche delle riviste analizzate

Seguiranno, una sezione che mostrerà i risultati ottenuti per la rivista "Current Applied Physics" e un'altra sezione che riporterà i risultati di altre riviste per poi fare dei confronti ed evidenziare le differenze tra le diverse discipline.

6.1 Current Applied Physics - Risultati ottenuti

La seguente sezioni vuole mostrare al lettore le singole analisi applicate alla rivista "Current Applied Physics" con delle brevi discussioni sulle informazioni che si possono estrapolare dalle distribuzioni risultati. La figura 6.1 mostra la distribuzione dei riferimenti bibliografici nelle dieci fette in cui è stato logicamente diviso. Questa analisi mostra come i riferimenti siano principalmente posizionati nella parte iniziale dell'articolo, per poi diminuire drasticamente nella parte centrale e raggiungendo percentuali minime nella parte finale dell'articolo.

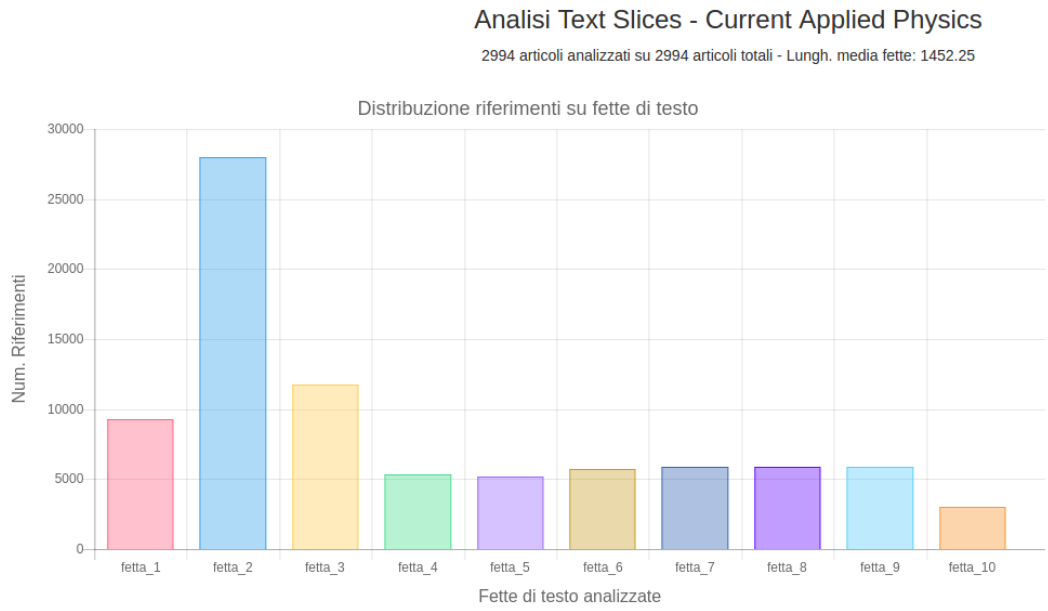


Figura 6.1: Analisi Text Slices - Current Applied Physics

L'analisi Titled Sections, mostrata nella figura 6.2, evidenzia i titoli scelti dall'utente o selezionati dall'analizzatore dei titoli, con i relativi valori di presenza, nella parte destra della schermata. Il grafico mostra che la maggior parte dei riferimenti bibliografici sono posizionati nelle sezioni "Introduction" e "Result and Discussion". La sezione fittizia "Other" rappresenta tutte le sezioni rimaste fuori dall'analisi perchè non scelte dall'utente o non rilevanti all'interno dell'insieme analizzato. Per ulteriori dettagli sulla rilevanza dei titoli e su "Other", si rimanda alla sezione 3.4.1.

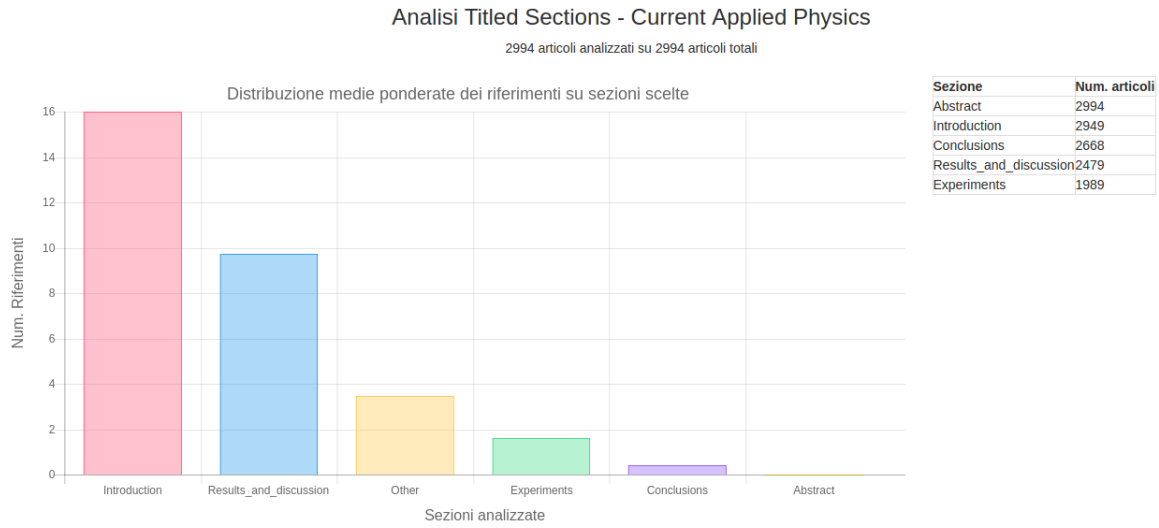


Figura 6.2: Analisi Titled Sections - Current Applied Physics

La figura 6.3, proposta di seguito, fornisce dei risultati simili all'analisi Text Slices ma presentano divisione logica completamente differente. L'analisi Numbered Sections divide l'articolo in quattro insiemi di sezioni e la cardinalità media di tali insiemi è riportata sotto il titolo dell'analisi. La distribuzione mostra la maggiore concentrazione di riferimenti bibliografici nelle sezioni iniziali, invece, la minore concentrazione delle sezioni finali. Questo dato è molto rilevante perché sarà una costante per molte riviste, anche riguardanti differenti discipline.

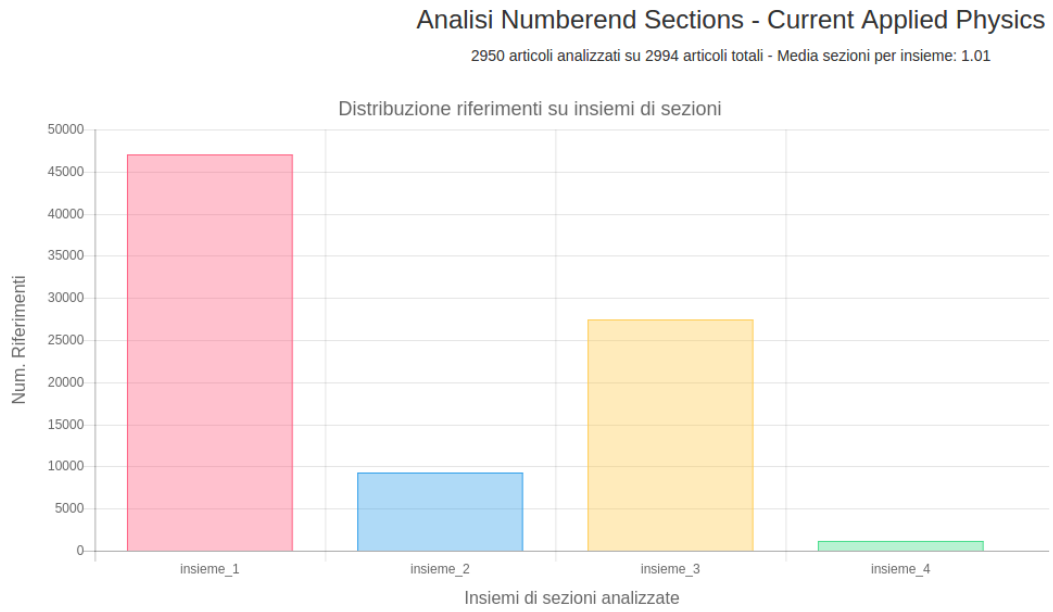


Figura 6.3: Analisi Numbered Sections - Current Applied Physics

Infine, nella figura 6.4 viene mostrata l'analisi Aggregation Index che tratta l'aggregazione dei riferimenti. La schermata mostra subito l'indice medio di aggregazione seguito dalla distribuzione. Questa rivista, rispetto molte altre, presenta un indice medio di aggregazione alto infatti, dalla distribuzione possiamo notare che la percentuale rimane abbastanza elevata fino al valore quattro.

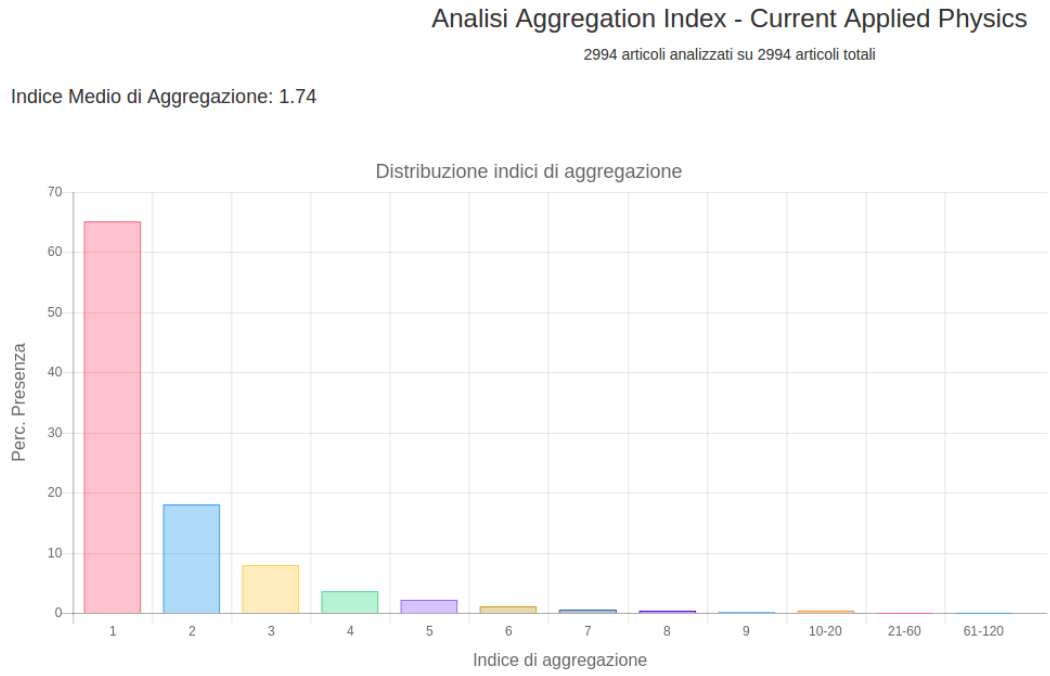


Figura 6.4: Analisi Aggregation Index - Current Applied Physics

6.2 Risultati e confronti tra differenti discipline

Adesso verranno mostrati i risultati più rilevanti per le riviste di psichiatria e per la rivista "Journal of Computational Science". Le distribuzioni risultati saranno descritte e confrontate tra di loro e, anche, con le distribuzioni ottenute per la rivista "Current Applied Physics".
Le figure 6.5 e 6.6 mostrano i risultati dell'analisi Text Slices. Questa analisi mette in evidenza la propensione all'utilizzo dei riferimenti bibliografici nella parte iniziale degli articoli. Tale caratteristica è abbastanza evidente nelle riviste "Journal of Computational Science" e "Current Applied Physics", è meno evidente nelle riviste psichiatriche analizzate. In queste ultime i riferimenti bibliografici sono concentrati principalmente nella parte iniziale ma, troviamo anche un'alta concentrazione nella parte finale.

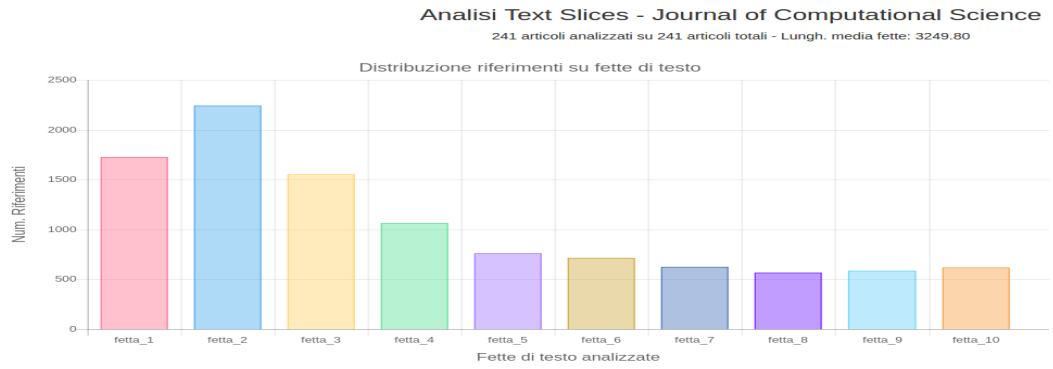


Figura 6.5: Analisi Text Slices - Journal of Computational Science

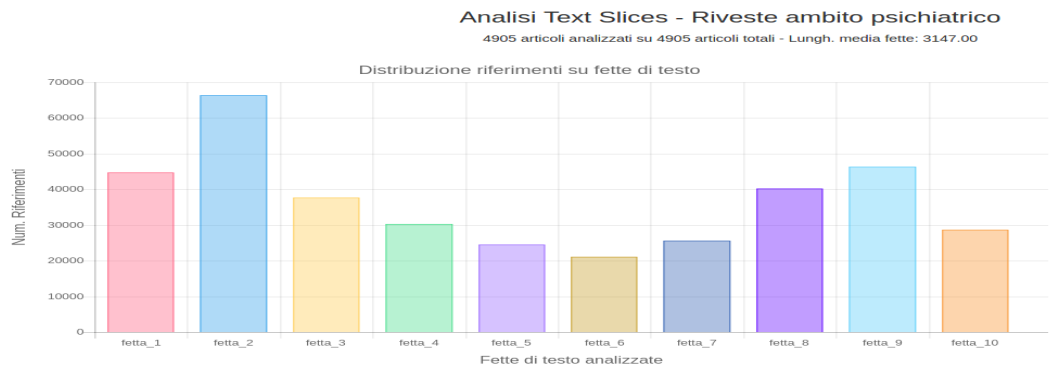


Figura 6.6: Analisi Text Slices - Riviste di psichiatria

Le figure 6.7 e 6.8 mostrano i risultati dell'analisi Aggregation Index. Possiamo notare come cambia l'indice medio di aggregazione tra le diverse riviste. In più, grazie alle distribuzioni, è evidente come la rivista "Current Applied Physics", mostrata in figura 6.4, presenta una percentuale maggiore di indici pari a due, tre e quattro rispetto le altre riviste analizzate, in modo coerente a quanto riportato nell'indice di aggregazione medio.

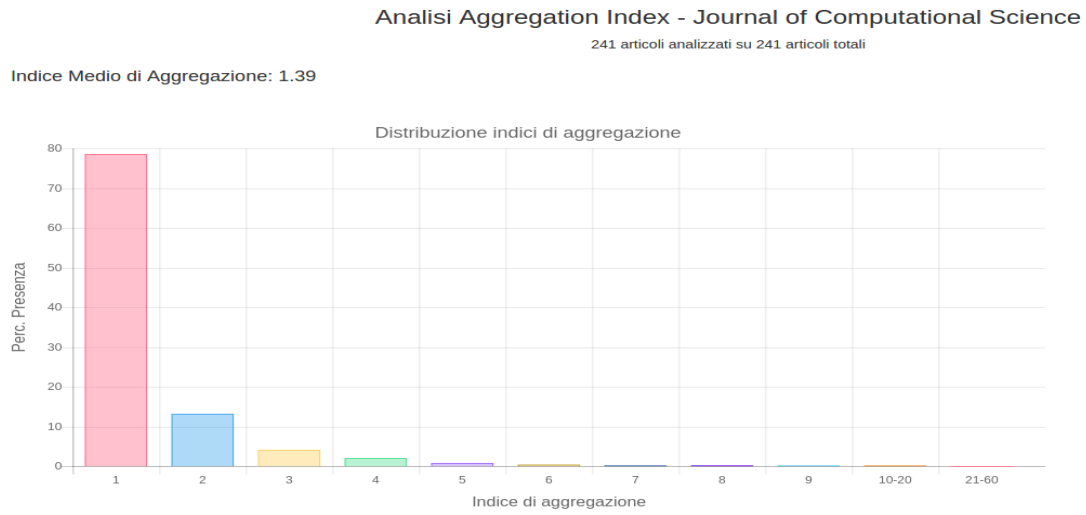


Figura 6.7: Analisi Aggregation Index - Journal of Computational Science

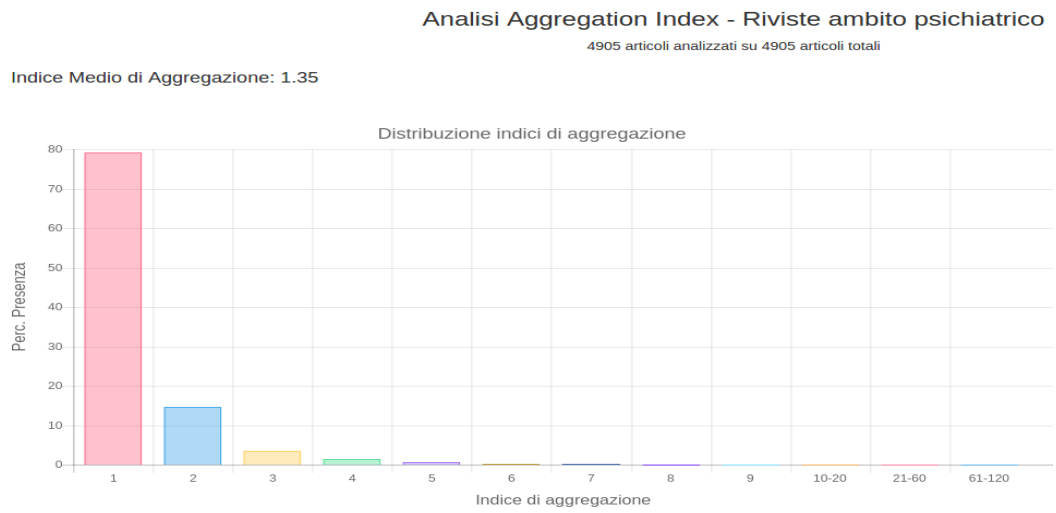


Figura 6.8: Analisi Aggregation Index - Riviste di psichiatria

Capitolo 7

Conclusioni e sviluppi futuri

Dalle analisi condotte sulle riviste descritte nel capitolo 6 sono emerse alcune caratteristiche comuni a tutte le riviste analizzate ed anche, caratteristiche specifiche per disciplina. Grazie a queste osservazioni siamo riusciti a raggiungere l'obiettivo posto nel capitolo 1, ovvero di trovare trend di riviste e discipline partendo dalle distribuzioni, prodotte dall'analizzatore dei riferimenti bibliografici.
Di seguito sono elencate le osservazioni più evidenti estrapolate dai risultati prodotti nella fase di analisi:

- la propensione ad inserire i riferimenti bibliografici nella parte iniziale degli articoli, questo è stato notato grazie alle analisi Text Slices e Numbered Sections;
- il valore dell'indice di aggregazione medio oscilla tra uno e due, durante la fase di analisi non si è mai notato un indice medio superiore a due;
- grazie all'analizzatore dei titoli sono emerse delle sezioni che accomunano riviste riguardanti discipline correlate. Le sezioni più frequenti nelle riviste biomediche e biologiche sono "abstract", "methods", "results" e "discussion". Nelle riviste che trattano argomenti di fisica ed informatica sono emerse, come più frequenti, le sezione "abstract" e "introduction";
- un'alta percentuale di riviste biomediche e biologiche presentano la maggior parte dei riferimenti bibliografici concentrati nella sezione "discussion", seguita dalla sezione "results" come concentrazione di riferimenti;
- l'abstract risulta presente in tutti gli articoli analizzati indipendentemente dalla disciplina e dalla rivista di appartenenza, eccetto qualche caso. Un'altra osservazione rilevante è la totale mancanza di riferimenti bibliografici collocati in questa sezione.

Altre caratteristiche sono state notate ma non in modo evidente come quelle descritte sopra. Questo fa emergere un problema, la carenza di riviste da poter analizzare. Le

riviste a nostra disposizione trattano solo alcune discipline scientifiche ma per un'analisi più accurata sono necessarie grandi quantità di riviste appartenenti a differenti discipline. In questo modo sarebbe possibile un'analisi più ampia poiché potremmo fare delle assunzioni su discipline completamente differenti ed anche su discipline affini.
Alla luce di questi risultati si può pensare di perfezionare Diribia così da renderla un'applicazione più flessibile e performante. Diribia può essere migliorata, infatti, dal punto di vista prestazionale e ampliata dal un punto di vista funzionale. Il secondo aspetto è più rilevante in quanto è un'applicazione piccola che tratta un argomento molto vasto e con molti punti di osservazione. Una prima estensione potrebbe essere la gestione di altri formati oltre XML. Ad esempio, poter prendere in input articoli PDF trovando dei meccanismi per convertirli in XML e modellarli per ottenere la struttura definita dal formato PLUS. Questo renderebbe l'applicazione molto più flessibile.
Si potrebbe pensare a possibili divisioni logiche come punto di partenza per altre tipologie di analisi sui riferimenti bibliografici. Quelle esistenti coprono certi punti di osservazione ma, a secondo delle necessità, possono essere necessarie divisioni logiche che fanno emergere altri risultati.
L'analisi Titled Sections attualmente lavora facendo un controllo della similitudine tra i titoli in analisi e quelli selezionati dall'utente. Un passo in avanti potrebbe essere l'abbandono dei controlli testuali (similitudine tra stringhe) e l'utilizzo di controlli semantici. Tali controlli permetterebbero di riconoscere come identici i titoli testualmente differenti ma che identificano la medesima sezione dal punto di vista semantico.
L'analizzatore dei titoli, come descritto nella sezione 5.2.1, si basa su alcune costanti per trovare un equilibrio tra prestazioni e accuratezza dei risultati. Queste costanti possono essere testate e perfezionate.
Infine, per la componente riguardante la generazione delle schermate grafiche, una possibile estensione potrebbe essere quella di creare grafici aggregati partendo da più distribuzioni. Attualmente la componente può prendere in input solo una distribuzione per volta e quindi fare analisi sulle differenze e analogie delle diverse riviste non è istantaneo.

Riferimenti

[1] *Chart.js*,
[http://www.chartjs.org/docs/].

[2] Councill, I. G., Giles, C. L., & Kan, M. Y. (2008). ParsCit: an Open-source CRF Reference String Parsing Package. In *LREC* (Vol. 8, pp. 661-667).

[3] Di Iorio, A., Nuzzolese, A. G., & Peroni, S. (2013, May). Towards the Automatic Identification of the Nature of Citations. In *SePublica* (pp. 63-74).

[4] Di Iorio, A., Nuzzolese, A. G., Peroni, S., Shotton, D., & Vitali, F. (2014). Describing bibliographic references in RDF. In *SePublica*.

[5] *Document type definition*,
[https://en.wikipedia.org/wiki/Document_type_definition].

[6] *Elsevier Journal Article Input DTD version 5.4.0*,
[https://www.elsevier.com/__data/assets/text_file/0003/58881/ja5_art540_dtd.txt].

[7] *Elsevier Journal Article Input DTD version 4.5.2*,
[https://www.elsevier.com/__data/assets/text_file/0011/58898/ja45_art452_dtd.txt].

[8] Garfield, E. (1963). *Science Citation Index*.
[http://garfield.library.upenn.edu/papers/80.pdf].

[9] Garfield, E. (2006). Citation indexes for science. A new dimension in documentation through association of ideas. *International journal of epidemiology*, 35(5), 1123-1127.

[10] Garfield, E. (1963). Citation indexes in sociological and historical research. *American documentation*, 14(4), 289-291.

[11] Hirsch, J. E. (2005). An index to quantify an individual's scientific research output. *Proceedings of the National academy of Sciences of the United States of America*, 16569-16572.

[12] *JATS: Journal Article Tag Suite, version 1.1*, [http://www.niso.org/workrooms/journalmarkup].

[13] Lawrence, S. et al. (2001). Persistence of web references in scientific research. *IEEE Computer*, 34(2), 26–31.

[14] McCown, F., Chan, S., Nelson, M. L., & Bollen, J. (2005). The availability and persistence of web references in D-Lib Magazine. arXiv preprint cs/0511077.

[15] Peng, F., & McCallum, A. (2006). Information extraction from research papers using conditional random fields. *Information processing & management*, 42(4), 963-979.

[16] Peroni, S., & Shotton, D. (2012). FaBiO and CiTO: ontologies for describing bibliographic resources and citations. *Web Semantics: Science, Services and Agents on the World Wide Web*, 17, 33-43.

[17] *PubMed Central*, [https://www.ncbi.nlm.nih.gov/pmc/about/intro/].

[18] *Semantic Publishing* , [https://semanticpublishing.wordpress.com/].

[19] Shotton, D. (2009). Semantic publishing: the coming revolution in scientific journal publishing. *Learned Publishing*, 22(2), 85-94.

[20] Wren, J. D. (2004). 404 not found: the stability and persistence of URLs published in MEDLINE. *Bioinformatics*, 20(5), 668-672.

[21] Yang, S., Han, R., Ding, J., & Song, Y. (2012). The distribution of Web citations. *Information Processing & Management*, 48(4), 779-790.

[22] Zhang, Q., Cao, Y. G., & Yu, H. (2011). Parsing citations in biomedical articles using conditional random fields. *Computers in biology and medicine*, 41(4), 190-194.

[23] Zou, J., Le, D., & Thoma, G. R. (2010). Locating and parsing bibliographic references in HTML medical articles. *International Journal on Document Analysis and Recognition*, 13(2), 107-119.