

POLITECNICO DI MILANO

School of Industrial and Information Engineering
Master of Science in Computer Science and Engineering



DESIGN, IMPLEMENTATION, AND PILOT TESTING
OF AN AUTOMATED METHOD TO CHARACTERIZE
MOBILE HEALTH APPS' TOPICAL AREAS BY
EXTRACTING INFORMATION FROM THE WEB.

Supervisor:

Prof. Enrico Gianluca Caiani

Co-Supervisor:

Dr. Alessia Paglialonga

Master Thesis by:

Massimo Schiavo

Matr. 858554

Academic Year 2017 – 2018

Ad Edoardo...

Contents

Sommario	I
Abstract.....	III
List of Figures.....	V
List of Tables	VII
Chapter 1 Introduction	1
1.1 General Overview	1
1.2 The app stores	3
1.3 Hyper Text Markup Language (HTML)	6
1.4 Regular Expressions (REs)	7
1.5 Unified Medical Language System (UMLS)	7
1.5.1 Metathesaurus.....	8
A) Concepts.....	9
B) Concepts and Concept Identifier	9
1.5.2 The Semantic Network	10
1.5.3 SPECIALIST Lexicon and Lexical Tools	11
1.6 MetaMap	14
1.6.1 The MetaMap algorithm	14
A) Noun phrase variants.....	16
B) Metathesaurus candidates' retrieval	17
C) Metathesaurus candidates' evaluation	18
D) The final mapping.....	22
1.6.2 MetaMap Options.....	22
A) Data Options	22
B) Filtering Forms.....	24
C) Processing Options	25
D) Output options.....	26

Chapter 2	Materials and Methods	27
2.1	Apps link retrieval	28
2.1.1	Detection of apps in the M and H&F categories	28
2.1.2	HTML builder	29
2.2	Information Extraction	32
2.2.1	Identification of information in the HTML pages	33
2.2.2	Automated extraction of apps' attributes	35
2.3	Information cleaning	36
2.3.1	Removal of HTML residuals	36
2.3.2	Removal of duplicates	36
2.4	Pre-processing	37
2.4.1	Removal of apps whose description are too short	37
2.4.2	Removal of apps other than in English	38
2.4.3	Removal of NON-ASCII characters	39
2.5	Text analytics	39
2.5.1	Extraction of UMLS concepts and CUIs	40
	A) Input	41
	B) Settings	42
	C) Output	43
	D) Editing Function	45
2.5.2	Mapping CUIs to topical areas	45
	A) Identification of strong Semantic Types	46
	B) MeSH Tree analysis	47
2.5.3	Classification	50
2.6	Training & Test sets	52
2.6.1	Test set: performance evaluation	55
2.6.2	Keyword search comparison	58

Chapter 3 Results	61
3.1 Database creation	62
3.2 MetaMap Analysis	67
3.2.1 Input	68
3.2.2 Output	68
3.2.3 Improvements.....	69
3.3 Classification	71
3.3.1 Training set	71
3.3.2 Test set.....	75
3.3.3 Comparison with keyword search	82
3.3.4 Database classification	83
 Chapter 4 Discussions and conclusions	 85
4.1 Database creation	86
4.2 MetaMap analysis	87
4.3 Classification	88
4.3.1 Classification based on text analytics.....	88
4.3.2 Comparison with classification based on keyword search	91
4.3.3 The whole database	93
4.4 Conclusions and future developments	94
 Bibliography	 99
 Webliography	 101

Sommario

Il Mobile Health (mHealth) costituisce un possibile nuovo modello di assistenza socio-sanitaria, realizzabile tramite l'utilizzo di dispositivi mobili come gli smartphone, i dispositivi di monitoraggio dei pazienti, i personal digital assistants, e le tecnologie indossabili. Il mercato delle app mHealth è molto ampio ed individuare la giusta app per un bisogno specifico può essere difficile, sia per un utente medico che per un paziente. Inoltre, è alquanto difficile identificare le caratteristiche rilevanti di un'app prima di effettuarne il download. In questo scenario, nasce il bisogno di sviluppare metodologie valide a classificare app potenzialmente utili per la salute, e ad identificare le loro caratteristiche. Questo lavoro si sviluppa in questa area, al fine di proporre una metodologia automatizzata basata sull'analisi testuale delle informazioni estratte dal web. In particolare, ci si è focalizzati sulle apps nelle categorie "Medical" (M) e "Health & Fitness" (H&F) presenti nell'US iTunes App Store. A tal fine, 42008 M e 79557 H&F pagine web sono state scaricate e, dopo la rimozione di duplicati e apps non in inglese, è stato creato un database contenente 80490 apps, successivamente classificato con il metodo proposto in questo studio, basato sull'identificazione di concetti medici e la loro appartenenza a specifiche aree di interesse. Tale metodo è stato sviluppato partendo da un training set composto da 400 apps e testato su un sottoinsieme di 400 apps estratte in modo casuale da questo database. I risultati ottenuti suggeriscono la fattibilità della caratterizzazione automatizzata delle apps e inoltre evidenziano una serie di possibili miglioramenti futuri del metodo stesso: il miglioramento della funzione di classificazione, l'analisi dei Semantic Types, l'estrazione di ulteriori caratteristiche (promotori, servizi, gli utenti) delle app per

avere una visuale più ampia delle applicazioni. La disponibilità di un metodo come quello descritto in questa Tesi potrebbe essere da supporto per i professionisti del settore sanitario per una selezione più informata e consapevole delle apps da prescrivere ai loro pazienti.

Abstract

Mobile Health (mHealth) is a possible new model of social health care achieved through the use of mobile devices such as smartphones, patient monitoring devices, personal digital assistants, and wearable technologies. The market of mHealth apps is very large and finding the right app for a specific need can be challenging, both for medical users and for patients. Furthermore, it may also be difficult to identify the relevant features of an app before downloading it. This situation arises the need of automated methods to characterize mHealth apps. In this study, a method based on text analytics to characterize the features of mobile health apps was developed. In particular, apps in the Medical (M) and Health & Fitness (H&F) categories on the US iTunes App Store were analyzed. As a result, 42008 M and 79557 H&F apps' webpages were automatically crawled. After duplicates and non-English apps removal, a database of 80490 unique apps was created and classified with the proposed method, based on the identification of biomedical concepts and their membership to specific topical areas. This automated method was developed on a training set of 400 apps and validated on a test set of 400 apps randomly selected from this database. These preliminary results suggested the viability of automated characterization of apps and highlighted directions for improvement in terms of: classification rules and vocabularies, analysis of Semantic Types, and extraction of additional features (promoters, services, and users). The availability of automated tools for app characterization could support healthcare professionals in informed, aware selection of health apps to recommend and prescribe to their patients.

List of Figures

Figure 1.1 – Outline of basic concepts used in this thesis.....	3
Figure 1.2 – Screenshot with categories on the iTunes App Store	5
Figure 1.3 – Screenshot with categories on the Google Play Store	5
Figure 1.4 - Search view of "cold"	11
Figure 1.5 - Example of entries in the SPECIALIST lexicon.....	13
Figure 1.6 - Outline of MetaMap algorithm.....	14
Figure 1.7 - Variants for the generator "ocular"	17
Figure 1.8 – Metathesaurus candidates for “ocular complications”.....	17
Figure 1.9 - The best Metathesaurus Mappings for "ocular complications"	22
Figure 1.10 - Default MetaMap data options	23
Figure 1.11 - Default MetaMap processing options.....	25
Figure 1.12 - Default MetaMap output options.....	26
Figure 2.1 - Outline of methodological workflow	27
Figure 2.2 - Example of complete URL	29
Figure 2.3 – List of apps’ URLs in the first page of apps with A as initial	30
Figure 2.4 – Links retrieving algorithm.....	31
Figure 2.5 – Workflow for MetaMap use.	41
Figure 2.6 – Example of MetaMap Input for batch upload	41
Figure 2.7 – Example of MetaMap output	44
Figure 2.8 – MetaMap output core information	44
Figure 2.9 – Edited output containing only core information.....	45
Figure 2.10 - Mapping workflow from CUIs to topical areas.....	46
Figure 2.11 – Classification function	51

Figure 2.12 – Classification function definition	53
Figure 2.13 - Example of number of concepts vs. relevance.....	54
Figure 3.1 – Outline of results workflow	61
Figure 3.2 – H&F Database distribution	65
Figure 3.3 – M Database distribution	65
Figure 3.4 - Merged database distribution	67
Figure 3.5 – Example of description without punctuation	69
Figure 3.6 – Example of description with a part without punctuation	70
Figure 3.7 - Distribution of topical areas over the training set	72
Figure 3.8 - Distribution of topical areas over the test set.....	76
Figure 3.9 - Distribution of topical areas over the whole database	83
Figure 4.1 - Outline of discussions workflow	85

List of Tables

Table 1.1 - Example of Concepts and CUIs	10
Table 1.2 - Variant Distances.....	18
Table 1.3 - Evaluation of "Eye" candidate shown in Figure 1.8.....	20
Table 1.4 - Evaluation of "Complications" candidate shown in Figure 1.8	21
Table 2.1 – Selection of attributes.....	32
Table 2.2 – Location of the information in the HTML page.....	33
Table 2.3 – Mapping from attributes to RE.....	35
Table 2.4 – Examples of apps with descriptions shorter than 14 characters.....	37
Table 2.5 – Examples of descriptions other than in English	38
Table 2.6 – MetaMap upload modalities.....	41
Table 2.7 – Semantic Types excluded	42
Table 2.8 – Examples of CUIs discarded.....	47
Table 2.9 – Examples of CUIs included	47
Table 2.10 – Relevant MeSH.....	48
Table 2.11 - Example of confusion matrix.....	56
Table 2.12 - Formulas to compute metrics.....	57
Table 2.13 - Keyword list for each topical area.....	58
Table 3.1 – Creation of the H&F Database	63
Table 3.2 – Creation of the M Database	64
Table 3.3 – Development of the merged app database.....	66
Table 3.4 – M and H&F distribution over the merged database	67
Table 3.5 – MetaMap recap	70
Table 3.6 - M and H&F apps distribution over the training set (N=400)	71
Table 3.7 – Training set confusion matrix.....	73

Table 3.8 - Accuracy, Precision, and Recall computed for each topical area on the training set.....	74
Table 3.9 - Mean, Median, and Best value for Accuracy, Precision, and Recall in the training set.....	74
Table 3.10 - Metrics computed on the training set	75
Table 3.11 - M and H&F apps distribution over the test set (N=400)	76
Table 3.12 - Test set confusion matrix	77
Table 3.13 - Metrics computed on the test set.....	78
Table 3.14 - Differences in metrics between the training set and the test set...	78
Table 3.15 - Differences in distribution between the training set and the test set	79
Table 3.16 - Binary metrics computed for each topical area on the training set	80
Table 3.17 - Binary metrics computed for each topical area on the test set	81
Table 3.18 - Metrics computed on the training set with keywords search	82
Table 3.19 - Metrics computed on the test set with keywords search	82
Table 3.20 - Metrics comparison on the training set	83
Table 3.21 - Metrics comparison on the test set.....	83
Table 3.22 - Distribution of topical areas over the whole database (percentages, N=80490).....	84
Table 4.1 - The number of apps in different stores and regions at June 2018 ...	86
Table 4.2 - Example of comparison with keyword-based classifier and CUIs-based classifier	92

Chapter 1

Introduction

1.1 General Overview

Mobile applications (apps) are changing the world, enriching people's lives, and enabling developers to innovate like ever before. There's nothing like finding a new app that transforms the way by which a user works or plays. However, finding the right app is sometimes not so simple: the user needs to browse an app market place (or app store) by inserting keywords, restricting the results into a category chosen among the ones available.

Internet search engines highlight how the interest in digital health has developed among the internet/telecommunication (outsiders) versus the healthcare industry (insiders). As the digital health market expanded and matured, fewer "digital health" internet searches by outsiders were observed, while interest among health professionals strengthened. As of June 2018, there are more than 318500 mHealth apps in the market. [W1].

In addition to the traditional keyword search-based method present on all the app stores, to assist potential users, several online resources have been launched to index, comment, and review health related apps. These resources include web-portals, expert- and user- communities, app repositories, or news sites. (e.g., [W2-W7]). These services offer the advantage of removing the difficulties associated with app discovery and quality verification on the stores. Indeed, these resources are not exempt from limitations. In general, there is potential for bias due to the intrinsically subjective review process. In addition, measurements and reviews can take significant time, effort, and resources to be effective. Moreover, there is an inherent delay from app release to assessment, during which the app may have undergone numerous updates and substantial revisions, making the information outdated. [Paglialonga *et al*, 2018a]

Much more complicated and fundamental is to understand when a medical app can be considered as a medical device. Apps can help people manage their own health and wellness, promote healthy living, and gain access to useful information when and where they need it. The Food and Drug Administration (FDA) encourages the development of mobile medical apps that improve health care and provide consumers and health care professionals with valuable health information. The FDA also has a public health responsibility to oversee the safety and effectiveness of medical devices including mobile medical apps. Try to cope with apps certification, the FDA issued the Mobile Medical Applications Guidance for Industry and Food and Drug Administration Staff on September 25, 2013, that explains the agency's oversight about mobile medical apps as devices. The focus has been posed on the apps that present a greater risk to patients if they don't work as intended, and on apps that cause smartphones or other mobile platforms to impact the functionality or performance of traditional connected medical devices.

FDA's mobile medical apps policy does not regulate the sale or general consumer use of smartphones or tablets, does not consider entities that exclusively distribute mobile apps, such as the owners and operators of the "iTunes App store" or the "Google Play store," to be medical device manufacturers, and does not consider mobile platform manufacturers to be medical device manufacturers just because their mobile platform could be used to run a mobile medical app regulated by FDA. [W8]

A Decision by the European Court of Justice in December 2017 took a broad interpretation of when software shall fall within the notion of medical devices. The Court stated that a software is in itself a medical device when it is specifically intended by the manufacturer to be used for one, or several medical purposes outlined in the definition of medical device - even without use on humans as required for medical devices. [W9]

As a result, the scenario is likely to change in the near future having a major impact on the app market because the apps that support a medical diagnosis and have a medical use shall be CE marked as medical devices - but the

1.2 The app stores

outcomes of mandatory certification are difficult to predict. [Paglialonga *et al*, 2018a]

In this context, this research is within the scope to test the feasibility of automated methods to characterize the features of mobile health apps (mHealth) directly from the app store, so to potentially provide a different filtering modality for interested users to identify the app that best fits their needs. To this aim, we developed and tested the basic modules of an automated method, based on text analytics, to characterize the apps' medical topics by extracting information from the Web, focusing on apps in the Medical (M) and Health & Fitness (H&F) categories on the US iTunes App Store.

Preliminary results of this thesis have been recently published in an international publication. [Paglialonga *et al*, 2018b]

This chapter describes the main concepts and tools useful to understand this work, as outlined in Figure 1.1.

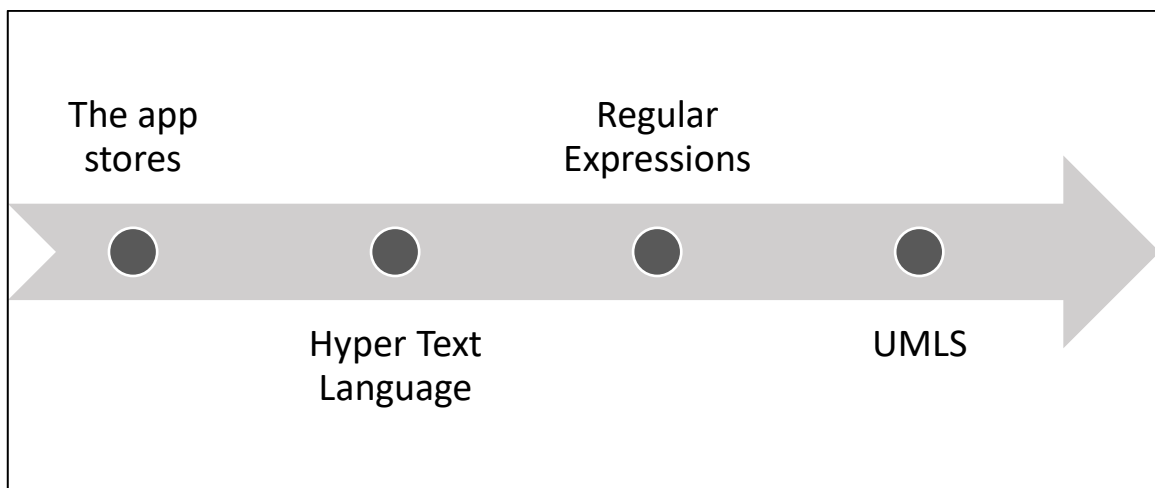


Figure 1.1 – Outline of basic concepts used in this thesis.

1.2 The app stores

The app store that can be accessed by a user depends on the type of device and operating system he/she has. If the user owns an Android phone, apps will be downloaded from Google Play, while for Apple devices the Apple's iTunes App Store will be accessed. For a BlackBerry, the BlackBerry App World

will be the reference, and in case of Windows Phone, the Windows Phone App Marketplace will be used.

Apps only work on the phones and tablets they are intended for, so for example an Android app won't work on an iPhone. Android and Apple are the biggest two competitors in the world of app stores, with 3.3 million and 2.2 million apps, respectively.

In addition, Android users have a larger choice of where to get apps: in addition to the official Google Play store, there are other marketplaces that could be accessed, such as the Amazon Appstore, which is more regulated. However, this wealth of choice comes with a few caveats.

Android is open to anyone to make apps and when published, there is no testing performed beyond the ones made by the developer. As a result, anyone is able to publish apps on the store and therefore, users can download apps that don't work properly, or worse, that could contain something nasty.

A minor issue is that many devices are running different versions of Android. It's almost impossible for app developers to test their app for different versions of Android, and for all the different Android phones, each with their different screen sizes and computing power.

Conversely, there is a greater guarantee of quality in the iTunes App Store. In fact, Apple tests and approves every app that goes on sale in the iTunes App Store, to guarantee that the app does what it's supposed to, and it is safe (from a software point of view) to be downloaded and used. Unlike Android, Apple only sells just few devices based on app utilization: the iPhone, iPod, Smartwatch, Tv and iPad. They all use the same iOS software, and the specifications are consistent across them.

On both stores, apps are organized into broad categories (e.g. games, medical, business) that the users can select to restrict their searches.

Figure 1.2 and Figure 1.3 show the list of categories available on the iTunes App Store and the Google Play Store, respectively.

1.2 The app stores



Figure 1.2 – Screenshot with categories on the iTunes App Store.

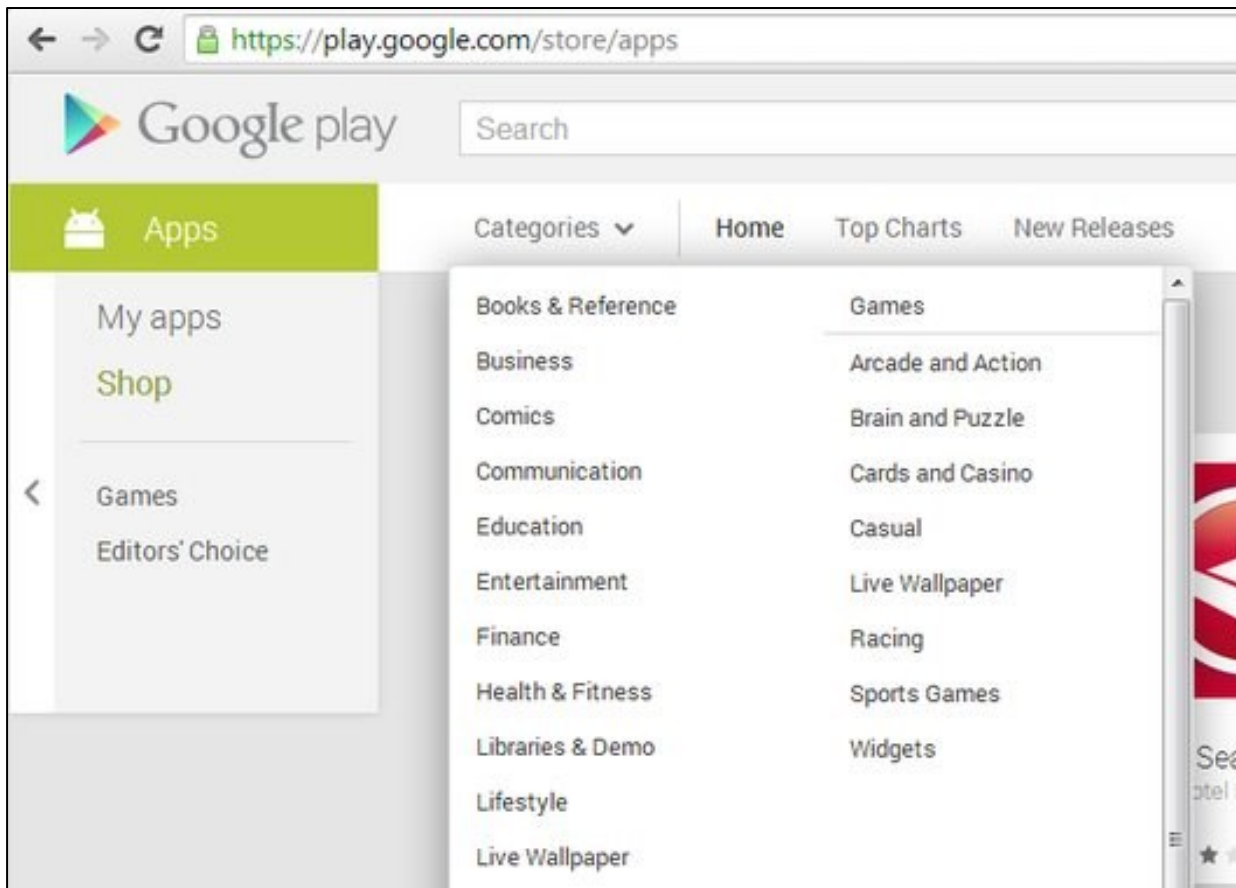


Figure 1.3 – Screenshot with categories on the Google Play Store.

Apps with possible relation to health are organized into two categories: "Health & Wellbeing" and "Medicine" in the Google Play Store and "Health & Fitness" and "Medical" in the iTunes App Store.

Since there's a better quality of the apps present in the iTunes App Store than in Google Play Store, this first study was focused on the iTunes App Store.

The iTunes App Store is a digital distribution platform, developed and maintained by Apple Inc., for mobile apps on its iOS operating system. The store allows users to browse and download apps developed with Apple's iOS software development kit. Apps can be downloaded on the iPhone smartphone, the iPod Touch handheld computer, the iPad tablet computer, and now to the Apple Watch smartwatch and 4th-generation or newer Apple TV as extensions of iPhone apps. The App Store was opened on July 10, 2008, with an initial 500 applications available. As of the first quarter of 2018, the iTunes App Store shows off more than a billion users with over 2.2 million apps. [W10].

1.3 Hyper Text Markup Language (HTML)

The Hyper Text Markup Language (HTML) is the language for describing the structure of Web pages. HTML gives authors the means to:

- Publish online documents with headings, text, tables, lists, photos, etc.
- Retrieve online information via hypertext links, at the click of a button.
- Design forms for conducting transactions with remote services, for use in searching for information, making reservations, ordering products, etc.
- Include spread-sheets, video clips, sound clips, and other applications directly in their documents.

With HTML, authors describe the structure of pages using markup. The elements of the language label pieces of content such as "paragraph," "list," "table," and so on. Cascading Style Sheet (CSS) is the language for describing the presentation of web pages, including colors, layout, and fonts. It allows the user to adapt the presentation to different types of devices, such as large screens, small screens, or printers. CSS is

1.4 Regular Expressions (REs)

independent of HTML and can be used with any eXtensible Markup Language (XML)-based markup language. The separation of HTML from CSS makes it easier to maintain sites, share style sheets across pages, and tailor pages to different environments. This is referred to as the separation of structure (or content) from presentation [W11]. The HTML of the app pages on the store was used in this work to extract the apps' attributes and create the respective database. As HTML is well structured and its structure and tags do not depend on the app, it can be parsed using regular expressions.

1.4 Regular Expressions (REs)

A regular expression (RE) is a specific kind of text pattern that can be used with many modern applications and programming languages. RE can be used to verify whether input fits into the text pattern, to find text that matches the pattern within a larger body of text, to replace text matching the pattern with other text or rearranged bits of the matched text, to split a block of text into a list of subtexts. [Goyvaerts & Levithan, 2012]

For example, "`<TAG\b[^>]*>(.*?)</TAG>`" matches the opening and closing pair of a specific HTML tag. Anything between the tags is captured into the first backreference (Backreferences match the same text as previously matched by a capturing group). The question mark in the RE makes the star lazy, to make sure it stops before the first closing tag rather than before the last, like a greedy star would do. This RE will not properly match tags nested inside themselves, like in "`<TAG>one<TAG>two</TAG>one</TAG>`".

1.5 Unified Medical Language System (UMLS)

The Unified Medical Language System (UMLS) facilitates the development of computer systems that behave as if they "understand" the language of biomedicine and health. To that end, the National Library of Medicine (NLM) produces and distributes the UMLS Knowledge Sources (databases) and associated software tools (programs). Developers use the Knowledge Sources and tools to build or enhance systems that create, process, retrieve, and

integrate biomedical and health data and information. The Knowledge Sources are multi-purpose and are used in systems that perform several functions involving information types such as patient records, scientific literature, guidelines, and public health data. The associated software tools assist developers in customizing or using the UMLS Knowledge Sources for particular purposes. The Lexical Tools work more effectively in combination with the UMLS Knowledge Sources but can also be used independently. There are three UMLS Knowledge Sources: the Metathesaurus, the Semantic Network, and the SPECIALIST Lexicon. [W12]

1.5.1 Metathesaurus

The Metathesaurus is a very large, multi-purpose, and multi-lingual vocabulary database that contains information about biomedical and health related concepts, their various names, and the relationships among them. Designed for use by system developers, the Metathesaurus is built from the electronic versions of various thesauri, classifications, code sets, and lists of controlled terms used in patient care, health services billing, public health statistics, indexing and cataloging biomedical literature, and/or basic, clinical, and health services research. These are referred to as the "source vocabularies" of the Metathesaurus. The term Metathesaurus draws on Webster's Dictionary third definition for the prefix "meta," i.e., "more comprehensive, transcending." In this sense, the Metathesaurus transcends the specific thesauri, vocabularies, and classifications it encompasses.

The Metathesaurus is organized by concepts or meanings. It links alternative names and views of the same concept and identifies useful relationships between different concepts.

The Metathesaurus is linked to the other UMLS Knowledge Sources – the Semantic Network and the SPECIALIST Lexicon. All concepts in the Metathesaurus are assigned to at least one Semantic Type from the Semantic Network. This provides consistent categorization of all concepts in the Metathesaurus at the relatively general level represented in the Semantic Network. Many of the words and multi-word terms that appear in concept names or strings in the Metathesaurus also appear in the SPECIALIST Lexicon.

1.5 Unified Medical Language System (UMLS)

The Lexical Tools are used to generate the word, normalized word, and normalized string indexes to the Metathesaurus.

A) Concepts

The Metathesaurus is organized by concepts. One of its primary purposes is to connect different names to the same concept from many different vocabularies. The Metathesaurus assigns several types of unique, permanent identifiers to the concepts and concept names it contains, in addition to retaining all identifiers that are present in the source vocabularies. The Metathesaurus concept structure includes concept names, their identifiers, and key characteristics of these concept names (e.g., language, vocabulary source, name type).

B) Concepts and Concept Identifier

A concept is a meaning. A meaning can have many different names. A key goal of Metathesaurus construction is to understand the intended meaning of each name in each source vocabulary and to link all the names from all of the source vocabularies that mean the same thing (the synonyms). However, this is not an exact science. The construction of the Metathesaurus assumes that specially trained subject experts can determine synonyms with a high degree of accuracy. Based on these inputs, Metathesaurus editors decide which synonyms to represent in the Metathesaurus concept structure.

Each concept or meaning in the Metathesaurus has a permanent concept unique identifier (CUI). The CUI has no intrinsic meaning. In other words, it is not possible to infer anything about a concept just by looking at its CUI. In principle, the identifier for a concept never changes, irrespective of changes over time in the names that are linked to it in the Metathesaurus or in the source vocabularies.

A CUI will be removed from the Metathesaurus when it is found that two CUIs are describing the same concept – in other words, when undiscovered synonyms come to light. In this case, one of the two CUIs will be retained, all relevant information in the Metathesaurus will be linked to it, and the other CUI will be eliminated.

Table 1.1 shows two examples of terms having different senses, and thus having different CUIs as they could represent different concepts.

Table 1.1 - Example of Concepts and CUIs.

Term	CUIs	Concept
Culture	C0010453	Anthropological Culture
	C0430400	Laboratory Culture
Cold	C0009264	Cold Temperature
	C0009433	Common Cold

1.5.2 The Semantic Network

The Semantic Network provides a consistent categorization of all concepts represented in the Metathesaurus and provides a set of useful relationships between these concepts. All information about specific concepts is found in the Metathesaurus; the Network provides information about the set of basic Semantic Types, or categories that may be assigned to these concepts, and it defines the set of relationships that may hold between the Semantic Types.

The Semantic Network contains 133 Semantic Types and 54 relationships. It serves as an authority for the Semantic Types that are assigned to concepts in the Metathesaurus, and defines these types, both with textual descriptions and by means of the information inherent to its hierarchies.

The Semantic Types are the nodes in the Network, and the Semantic Relations between them are the links. There are major groupings of Semantic Types for organisms, anatomical structures, biologic function, chemicals, events, physical objects, and concepts or ideas. The current scope of the UMLS Semantic Types is quite broad, allowing for the semantic categorization of a wide range of terminology in multiple domains.

For example, Figure 1.4 shows the search view of the “Cold Temperature” concept in which it is possible to note: its CUI (i.e. C0009264), its Semantic Type (i.e. Natural Phenomenon or Process), its definition with the set of all its synonyms, and finally its relation with other 542 concepts.

1.5 Unified Medical Language System (UMLS)

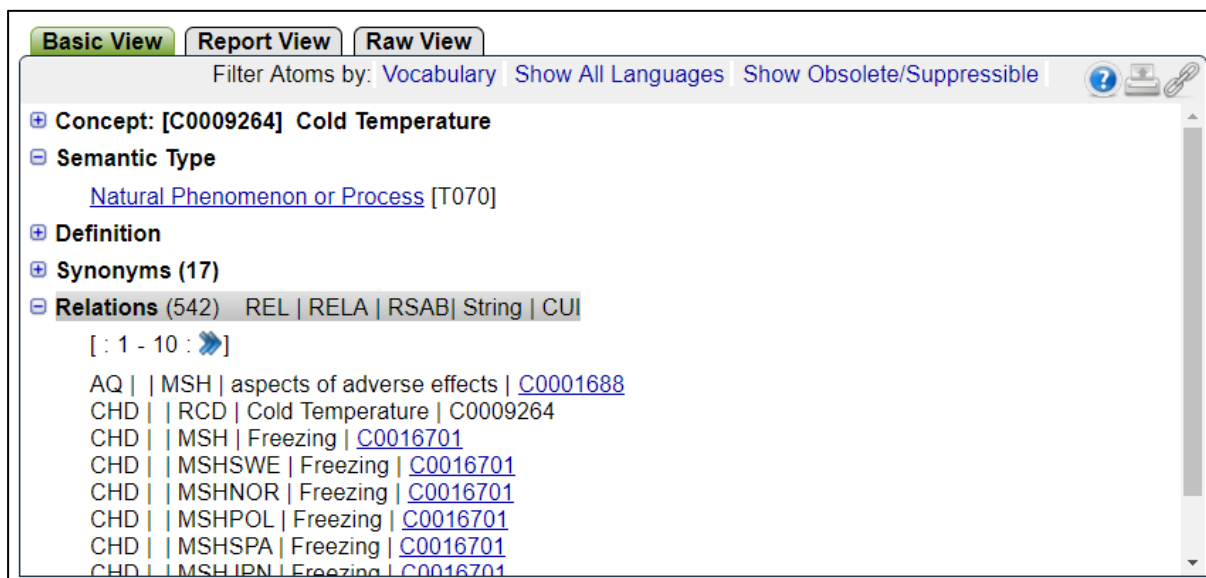


Figure 1.4 - Search view of "cold".

1.5.3 SPECIALIST Lexicon and Lexical Tools

The SPECIALIST Lexicon, as part of the UMLS Knowledge Sources, was developed by the US National Library of Medicine and it is intended to be a general English lexicon that includes many biomedical terms. Coverage includes both commonly occurring English words and biomedical vocabulary. The lexicon entry for each word or term records the syntactic, morphological, and orthographic information needed by the SPECIALIST Natural Language Processing System.

The Lexical Tools are designed to address the high degree of variability in natural language words and terms. Words often have several inflected forms that would properly be considered instances of the same word.

The verb "treat", for example, has three inflectional variants:

- treats — third person, singular present tense form
- treated — the past and past participle form
- treating — the present participle form

The Lexicon consists of a set of lexical entries with one entry for each spelling, or set of spelling variants in a particular part of speech. Lexical items may be "multi-word" terms made up of other words if the multi-word term is determined to be a lexical item by its presence, as a term in general English or medical dictionaries. Expansions of generally used acronyms and

abbreviations are also allowed as multi-word terms. Multi-word terms in the Metathesaurus and other controlled vocabularies may have word order variants in addition to their inflectional and alphabetic case variants.

The Lexical Tools allow the user to abstract away from several types of variation, including British English/American English spelling variation and character set variations. Lexical entries are not divided into meanings. Therefore, an entry represents a spelling-category pairing regardless of semantics.

The noun "act" has two senses that both show a capitalized and lower-case spelling: an act of a play and an Act of law. Since both senses share the same spellings and syntactic category, they are represented by a single lexical entry in the current lexicon. When different meanings have different syntactic behavior, codes for each behavior are recorded in a single entry. For example, "beer" has two meanings: the alcoholic beverage and the amount of a standard container of that beverage.

Words are selected for lexical coding from a variety of sources. Approximately 20000 words from the UMLS Test Collection of MEDLINE abstracts together with words appearing both in the UMLS Metathesaurus and Dorland's Illustrated Medical Dictionary form the core of the words entered. In addition, an effort has been made to include words from the general English vocabulary. The 10000 most frequent words listed in The American Heritage Word Frequency Book and the list of 2000 words used in definitions in Longman's Dictionary of Contemporary English have also been coded. Since the majority of the words selected for coding are nouns, an effort has been made to include verbs and adjectives by identifying verbs in current MEDLINE citation records, by using the Computer Usable Oxford Advanced Learner's Dictionary, and by identifying potential adjectives from Dorland's Illustrated Medical Dictionary using heuristics developed by McCray and Srinivasan (1990).

The unit lexical record is a frame structure consisting of slots and fillers. Each lexical record has a "base=" field whose filler indicates the base form, and optionally a set of "spelling_variants=" fields to indicate spelling variants. An "entry=" field records the unique identifier (EUI) of the record. EUI numbers

1.5 Unified Medical Language System (UMLS)

are seven-digit numbers preceded by an "E". Each record has a "cat=" field indicating part of speech. Nouns that are the nominalizations of verbs or adjectives (i.e. "treat" and "treatability") have a "nominalization_of=" field containing the base form, category and EUI of the verb or adjective of which they are the nominalizations. The "position=" slot is for adjective describing the syntactic positions in which they occur. The lexical record is delimited by braces "{...}" [Browne et al, 2000].

An example of entries in the SPECIALIST lexicon is described in Figure 1.5.

```
{base=treat
entry=E0061964
  cat=verb
  variants=reg
  intrans
  tran=np
  tran=pphr(with,np)
  tran=pphr(of,np)
  ditran=np,pphr(to,np)
  ditran=np,pphr(with,np)
  ditran=np,pphr(for,np)
  cplxtran=np,advbl
  nominalization=treatment|noun|E0061968
}
{base=treat
entry=E0061965
  cat=noun
  variants=reg
}
{base=treatability
entry=E0061966
  cat=noun
  variants=reg
  variants=uncount
  compl=pphr(of,np)
  nominalization_of=treatable|adj|E0061967
}
```

Figure 1.5 - Example of entries in the SPECIALIST lexicon.

1.6 MetaMap

MetaMap is a program developed at the NLM to map biomedical text to the Metathesaurus or to discover referred Metathesaurus concepts in a text. MetaMap uses a knowledge intensive approach based on symbolic, natural language processing (NLP) and computational linguistic techniques. Besides being applied for both Information Retrieval (IR) and data mining applications, MetaMap is one of the foundations of NLM's Indexing Initiative System that is applied to both semiautomatic and fully automatic indexing of the biomedical literature at the library.

MetaMap maps text into concepts from the UMLS Metathesaurus. Text is taken through a series of modules and broken down into the components that include sentences, phrases, lexical elements and tokens. Variants are generated from the resulting phrases, and candidate concepts from the UMLS Metathesaurus are retrieved and evaluated against their phrases. The resulting concepts are organized in such a way as to best cover the text, known as a final mapping.

1.6.1 The MetaMap algorithm

Figure 1.6 outlines the steps computed by the MetaMap algorithm to the final mapping.

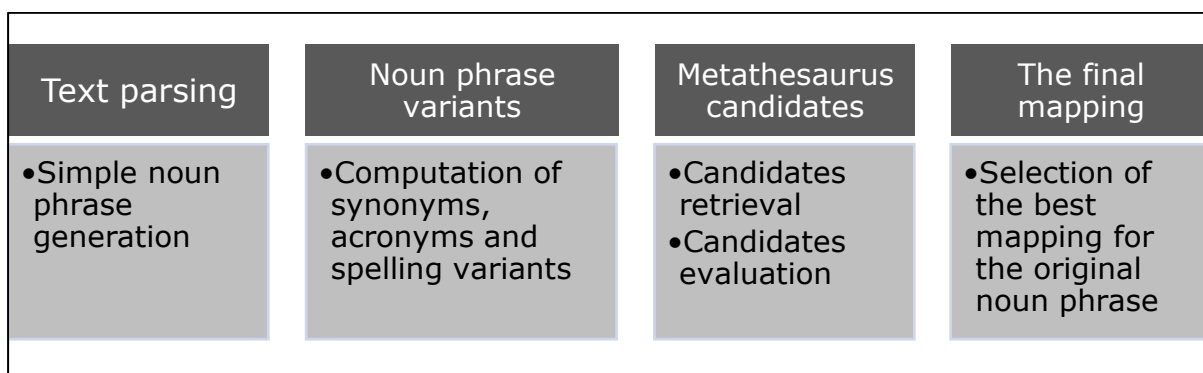


Figure 1.6 - Outline of MetaMap algorithm.

1.6 MetaMap

The algorithm follows these steps:

- 1) Text parsing: it parses arbitrary text into simple noun phrases; this limits the scope of further processing and thereby makes the mapping effort more tractable. Parsing is performed using the SPECIALIST minimal commitment parser [McCray AT *et al*, 1994] which produces a shallow syntactic analysis of the text. The parser uses the Xerox part-of-speech tagger [Cutting *et al*, 1992] which assigns syntactic tags (e.g., noun, verb) to words not having a unique tag in the SPECIALIST lexicon.

For example, consider the text fragment "*ocular complications of myasthenia gravis*". The parser detects two noun phrases: "*ocular complications*" and "*of myasthenia gravis*". A simplified syntactic analysis for "*ocular complications*" is [*mod(ocular), head(complications)*].

Note that the parser indicates that "*complications*" is the most central part, the *head*, of the phrase. Words with tags such as prepositions, conjunctions and determiners are normally ignored in subsequent processing;

- 2) Noun phrase variants: it generates the variants for the noun phrase where a variant essentially consists of one or more noun phrase words together with all of its spelling variants, abbreviations, acronyms, synonyms, inflectional and derivational variants, and meaningful combinations of these;
- 3) Metathesaurus candidates: for each candidate in the candidate set of all Metathesaurus strings containing one of the variants, MetaMap computes the mapping from the noun phrase and it calculates the strength of the mapping using an evaluation function. Afterwards, candidates are ordered by mapping strength;
- 4) The final mapping: it combines candidates involved with disjoint parts of the noun phrase, recomputes the match strength based on the combined candidates, and selects those having the highest score to form a set of best Metathesaurus mappings for the original noun phrase.

Descriptions of steps 2-4 of the mapping strategy are given in the next subsections, with related examples.

A) Noun phrase variants

The Metathesaurus mapping algorithm begins by computing a set of variant generators for each noun phrase discovered by the parser. A variant generator is any meaningful subsequence of words in the phrase, where a subsequence is meaningful if it is either a single word or occurs in the SPECIALIST lexicon.

For example, the variant generators for the noun phrase of "*liquid crystal thermography*" are "*liquid crystal thermography*", "*liquid crystal*", "*liquid*", "*crystal*" and "*thermography*" (prepositions, determiners, conjunctions, auxiliaries, modals, pronouns and punctuation are ignored). A simpler example which will be used throughout the sequel is based on the noun phrase "*ocular complications*". Its generators are simply "*ocular*" and "*complications*".

The approach taken in computing variants is a canonicalization approach. This simply means that a variant represents not only itself but all of its inflectional and spelling variants. Collapsing inflectional and spelling variants results in significant computational savings. Variants are computed for each of the variant. The computation for each generator proceeds as follows:

1. To compute all acronyms, abbreviations and synonyms of the generator;
2. To augment the elements of the three sets by computing their derivational variants and the synonyms of the derivational variants;
3. For each member of the Acronyms/Abbreviations set, to compute synonyms;
4. For each member of the Synonyms set, to compute acronyms/abbreviations.

Acronyms and abbreviations are not recursively generated since doing so almost always produces incorrect results. For example, the variants computed for the generator "*ocular*" are shown in Figure 1.7.

1.6 MetaMap

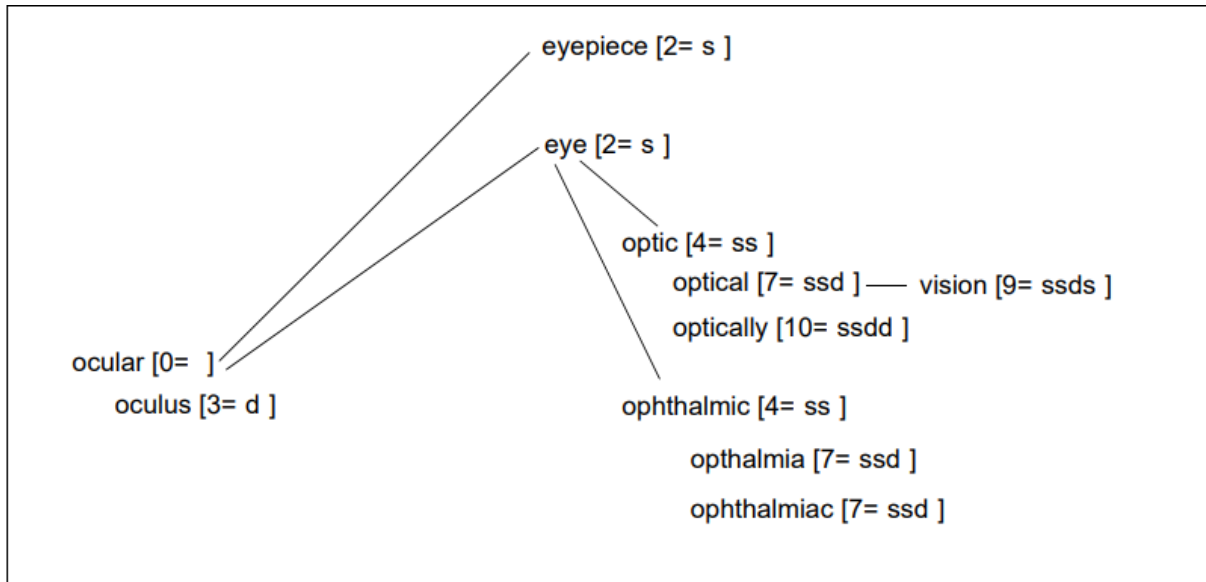


Figure 1.7 - Variants for the generator "ocular". Following each variant is its variant distance score, a rough measure of how much it varies from its generator and the history of how it was computed (How scores are computed is described later in section C). For example, "oculus", with variant Distance 3 and history d (3=d), is simply a derivational variant of the generator ocular; "optical", with variant distance 7 and history ssd (7=ssd), is a derivational variant of a synonym (optic) of a synonym (eye) of ocular; and vision, with variant distance 9 and history ssds (9=ssds), is a synonym of the derivational variant optical described above.

B) Metathesaurus candidates' retrieval

The Metathesaurus candidates for a noun phrase consist of the set of all Metathesaurus strings containing at least one of the variants computed for the phrase. The candidates are easily found by using a version of the Metathesaurus word index, an index from words to all Metathesaurus strings containing them. The Metathesaurus candidates for the noun phrase "ocular complications" are shown in Figure 1.8

861 Complications (Complication)
861 complications <1>
638 Eye
611 Optic (Optics)
588 Ophthalmia (Endophthalmitis)

Figure 1.8 - Metathesaurus candidates for "ocular complications".

The candidates are ordered according to the evaluation function described in the next section.

C) Metathesaurus candidates' evaluation

The evaluation function computes a measure of the quality of the match between a phrase and a Metathesaurus candidate. The evaluation function is based on four components: centrality, variation, coverage, and cohesiveness. A normalized value between 0 (the weakest match) and 1 (the strongest match) is computed for each of these components.

After this step, a weighted average is computed, in which the coverage and cohesiveness components receive twice the weight as the centrality and variation components. These weights were determined empirically by Dr. Alan (Lan) Aronson, the developer of Metamap. The result is then normalized to a value between 0 and 1000, where 0 indicates no match at all and 1000 indicates an identical match (except for spelling variation, capitalization, NOS suffixes and inversions such as "Cancer, Lung" vs. "Lung Cancer"). When MetaMap is set to ignore word order, the coverage component is replaced by an involvement component. Each of the evaluation function components is discussed below.

- The centrality value is simply 1 if the string involves the head of the phrase and 0 otherwise.
- The variation value estimates how much variants in the Metathesaurus string differ from the corresponding words in the phrase. It is computed by first determining the variation distance, as the sum of the distance values for each step taken during variant generation, for each variant in the Metathesaurus string. The values for each step are listed in Table 1.2.

Table 1.2 - Variant Distances.

Variant Type	Distance Value
spelling	0
inflectional	1
synonym or acronym/abbreviation	2
derivational	3

1.6 MetaMap

The variation distance determines the variation value for the given variant according to the formula $V=4/(D+4)$. As the total distance value, D , increases from its minimum value of 0, V decreases from a maximum value of 1 and is bounded below by 0. The final variation value for the candidate is the average of the values for each of the variants.

- The coverage value indicates how much of the Metathesaurus string and the phrase are involved in the match. To compute this value, the number of words participating in the match is computed for both the Metathesaurus string and the phrase. These numbers are called the Metathesaurus span and phrase span, respectively. The coverage value for the Metathesaurus string is the Metathesaurus span divided by the length of the string. Similarly, the coverage value for the phrase is the phrase span divided by the length of the phrase. The final coverage value is the weighted average of the values for the Metathesaurus string, and the phrase where the Metathesaurus string is given twice the weight as the phrase.
- The cohesiveness value is similar to the coverage value but emphasizes the importance of connected components. A connected component is a maximal sequence of contiguous words participating in the match. The connected components for both the Metathesaurus string and the phrase are computed. This information is abstracted by noting the size of each component. This produces a set of connected component sizes for both the Metathesaurus string and the phrase. The cohesiveness value for the Metathesaurus string is the sum of the squares of the connected Metathesaurus string component sizes divided by the square of the length of the string. A similar cohesiveness value is computed for the phrase. The final cohesiveness value is the weighted average of the Metathesaurus string and phrase values where the Metathesaurus string is again given twice the weight as the phrase.
- Also, a fifth component (involvement) exists, that is a replacement of the coverage value when word order is ignored.

Table 1.3 and Table 1.4 show the evaluation function computed for two of the candidates listed in Figure 1.8, in detail "Eye" and "Complications", respectively.

Table 1.3 - Evaluation of "Eye" candidate shown in Figure 1.8.

Metric	Value
Centrality	0 because it's not the head of the phrase.
Variation	$V = \frac{4}{4 + D}$ <p>Where D is the total distant value and it's equal to 2 since "eye" is a synonym of "ocular".</p> $V = \frac{4}{4 + 2} = \frac{2}{3}$
Coverage	$Coverage = \frac{(\frac{SPAN \text{ phrase}}{\text{length of the phrase}} + 2 * \frac{SPAN \text{ term}}{\text{length of the term}})}{3}$ <p>Where both the SPANs are equal to 1, the length of the phrase is 2 since it's composed by two terms and the length of the term is 1.</p> $Coverage = \frac{(\frac{1}{2} + 2 * \frac{1}{1})}{3} = \frac{5}{6}$
Cohesiveness	$Coverage = \frac{(\frac{SPAN \text{ phrase}^2}{\text{length of the phrase}^2} + 2 * \frac{SPAN \text{ term}^2}{\text{length of the term}^2})}{3}$ $Coverage = \frac{(\frac{1^2}{2^2} + 2 * \frac{1^2}{1^2})}{3} = \frac{(\frac{1}{4} + 2)}{3} = \frac{3}{4}$
Score	$Score = \frac{1000 * (\text{centrality} + \text{variation} + 2 * \text{coverage} + 2 * \text{cohesiveness})}{6}$ <p>That become:</p> $Score = \frac{1000 * (0 + 2/3 + 2 * 5/6 + 2 * 3/4)}{6} = 638$

1.6 MetaMap

Table 1.4 - Evaluation of "Complications" candidate shown in Figure 1.8.

Metric	Value
Centrality	1 because it's the head of the phrase.
Variation	$V = \frac{4}{4 + D}$ <p>Where D is the total distant value and it's equal to 0 since it's a spelling variant</p> $V = \frac{4}{4 + 0} = 1$
Coverage	$Coverage = \frac{\left(\frac{SPAN \text{ phrase}}{\text{length of the phrase}} + 2 * \frac{SPAN \text{ term}}{\text{length of the term}}\right)}{3}$ <p>Where both the SPANs are equal to 1, the length of the phrase is 2 since it's composed by two terms and the length of the term is 1.</p> $Coverage = \frac{\left(\frac{1}{2} + 2 * \frac{1}{1}\right)}{3} = \frac{5}{6}$
Cohesiveness	$Coverage = \frac{\left(\frac{SPAN \text{ phrase}^2}{\text{length of the phrase}^2} + 2 * \frac{SPAN \text{ term}^2}{\text{length of the term}^2}\right)}{3}$ $Coverage = \frac{\left(\frac{1^2}{2^2} + 2 * \frac{1^2}{1^2}\right)}{3} = \frac{\left(\frac{1}{4} + 2\right)}{3} = \frac{3}{4}$
Score	$Score = \frac{1000 * (\text{centrality} + \text{variation} + 2 * \text{coverage} + 2 * \text{cohesiveness})}{6}$ <p>That become:</p> $Score = \frac{1000 * (1 + 1 + 2 * 5/6 + 2 * 3/4)}{6} = 861$

D) *The final mapping*

The final step consists in examining combinations of Metathesaurus candidates that participate in matches with disjoint parts of the noun phrases. The evaluation function is applied to the combined candidates, and the best ones form the final mapping result. The best mappings for ocular complications are shown in Figure 1.9.

```
[mod([tokens([ocular]), metaconc([Eye])),  
head([tokens([complications]), metaconc([Complica-  
tions]))],  
confid(861)]  
and  
[mod([tokens([ocular]), metaconc([Eye]),  
head([tokens([complications]), metaconc([complica-
```

Figure 1.9 - The best Metathesaurus Mappings for "ocular complications".

The centrality, variation, coverage and cohesiveness values for the mapping in this example are 1, 2/3, 1 and 1, respectively. The final evaluation of the mapping is the weighted average $(1 + 2/3 + 2*1 + 2*1)/6$ which normalizes to 861 and is reported as a confidence value in the Figure 1.9 (i.e. `confid(861)`).

1.6.2 MetaMap Options

MetaMap is highly configurable, and its performance is controlled by option flags, each of which has a short name (e.g., `-I`) and a long name (e.g., `--show_cuis`).

A) *Data Options*

MetaMap's data options determine the Knowledge Source (e.g., the version of the UMLS Metathesaurus to use), the Data Version, and the Data Model (e.g., (strict or relaxed)) used for processing. Because MetaMap is used both for highly focused semantic processing as well as browsing, three data models differing in the degree of filtering are created.

- **Strict Model:** all forms of filtering are applied. This view is most appropriate for semantic processing where the highest level of accuracy is needed.

1.6 MetaMap

- Moderate Model: manual, lexical and type-based filtering, but not syntactic filtering, are used. This view is appropriate for term processing where input text should not be divided into simple phrases but considered as a whole.
- Relaxed Model: only manual and lexical filtering are performed. This provides access to virtually all Metathesaurus strings and is appropriate for browsing.

If no model is specified, then the strict model is used. The default data version instead is the USAbase. The USAbase data version includes those source vocabularies with no associated restrictions beyond a UMLS license, and free for use for US-based projects; this version includes the Base vocabularies (those with Restriction Category 0), plus the five Category-4 sources and the four Category-9 sources (including, most notably, SNOMEDCT). Other data versions available are "Base" and "NML".

+ Expand All

▲ Data Options (pick one of each)

Knowledge Source (-Z): 2017AB ▼

Data Version (-V): USAbase ▼

Data Model: Strict Model (-A) ▼

Restrict to Sources uses only the specified UMLS Source Vocabularies while mapping concepts. E.g., -R ICD10CM,MSH. To see what sources are available for your selected Knowledge Source, Data Version, and Data Model, select the checkbox below and a separate window will popup with the list of available sources. **Please Note:** The text field is not editable, you must make your selections via the popup window which is accessible through the checkbox or the Edit button.

Restrict to Sources (-R)

Edit

Figure 1.10 - Default MetaMap data options.

B) *Filtering Forms*

The files requiring the most effort to be created are the word index files [Aronson, 2006]. The Metathesaurus files are filtered in four ways:

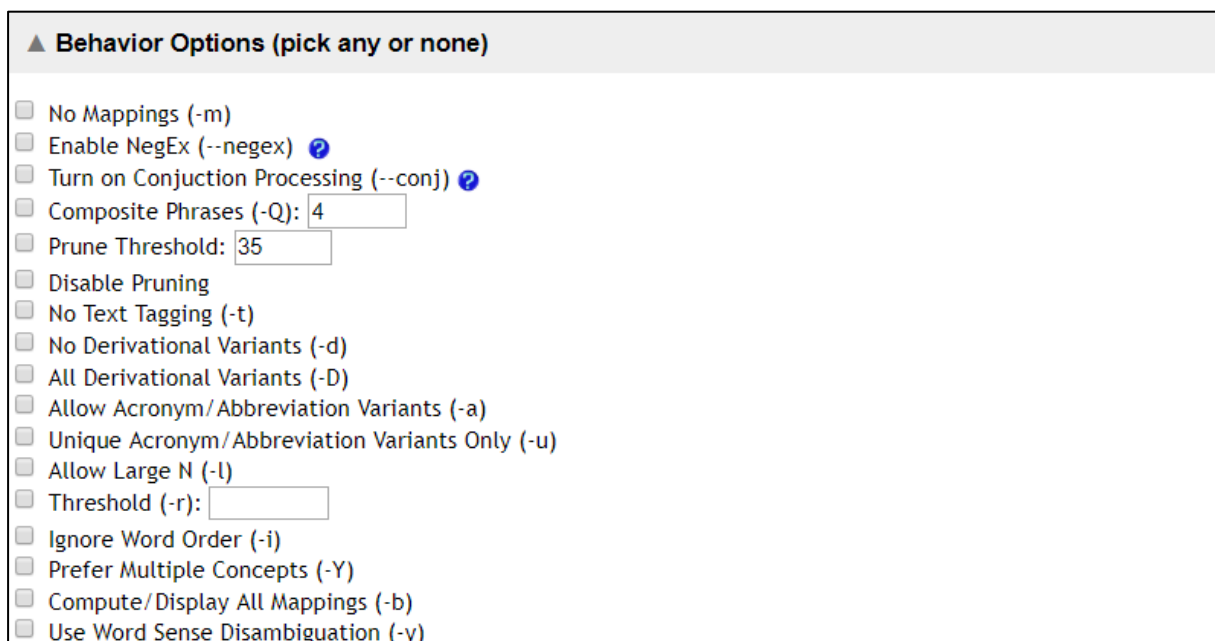
- 1) Manual filtering: a small number of Metathesaurus strings are problematic and have been manually suppressed before performing other forms of filtering. These include numbers, single alphabetic characters, special cases such as '*Periods*' for '*Menstruation*', and ambiguities. The most numerous problems here are the ambiguities. The creators of the Metathesaurus have instituted the notion of suppressible synonyms, strings that do not express themselves completely or that are abbreviatory or informal. Strings marked as suppressible account for most of the problematic ambiguity in the Metathesaurus.
- 2) Lexical filtering: it consists of removing strings for a concept that are effectively the same as another string for the same concept. This is accomplished by normalizing all strings for a given concept according to the above criteria and removing all but one string for each set of strings that normalize to the same thing.
- 3) Filtering by type: in addition to filtering out suppressible synonyms, terms are excluded based on their Term Type (TTY). The excluded types are generally abbreviatory, obsolete or have some kind of internal structure such as laboratory test descriptions in Logical Observation Identifiers Names and Codes (LOINC), one of the constituent Metathesaurus vocabularies.
- 4) Syntactic filtering: The final kind of filtering is based on applying the parser to the Metathesaurus strings themselves. Since normal MetaMap processing involves mapping the simple noun phrases found in text, it is highly unlikely that a complex Metathesaurus string will be part of a good mapping. Thus, strings consisting of more than one simple phrase are filtered out. Because of their tractability, composite phrases (the ones containing well-behaved prepositional phrases) are exempted from this filtering.

The filtering form has to be specified in the data options like in Figure 1.10.

1.6 MetaMap

C) Processing Options

Processing options control MetaMap's search algorithms and therefore affect the choice of UMLS concepts identified, as well as internal behavior such as how aggressive to be in generating word variants, whether or not to ignore Metathesaurus strings containing very common words, and whether to respect or to ignore word order. Options exist that allow specifying the maximum number of candidates to be used for constructing mappings, forcing MetaMap to generate variants dynamically rather than by looking up variants in a table (this option is normally used only for debugging purposes), allowing the use of any acronym/abbreviation variants, preventing the use of any derivational variation in the computation of word variants or forcing the use of all the derivational variation instead of only those between adjectives and nouns. Other options affect the phrase parsing allowing MetaMap to ignore the order of the words in the phrases it processes, preventing MetaMap from aborting its processing for commonly occurring phrases that are known to produce no mappings, and finally forcing MetaMap to process term rather than full text. In addition, options that affect the Metathesaurus candidates' retrieval enabling the retrieval for two-character words and one-character word are available.



The image shows a dialog box titled "▲ Behavior Options (pick any or none)". It contains a list of 18 options, each with a checkbox and a label. The options are:

- No Mappings (-m)
- Enable NegEx (--negex) ?
- Turn on Conjunction Processing (--conj) ?
- Composite Phrases (-Q): 4
- Prune Threshold: 35
- Disable Pruning
- No Text Tagging (-t)
- No Derivational Variants (-d)
- All Derivational Variants (-D)
- Allow Acronym/Abbreviation Variants (-a)
- Unique Acronym/Abbreviation Variants Only (-u)
- Allow Large N (-l)
- Threshold (-r):
- Ignore Word Order (-i)
- Prefer Multiple Concepts (-Y)
- Compute/Display All Mappings (-b)
- Use Word Sense Disambiguation (-y)

Figure 1.11 - Default MetaMap processing options.

D) Output options

Output options control how MetaMap displays results. It's possible to choose among different output both in machine and human (e.g. Prolog) processable formats.

Options to display all the mappings rather than displaying only the top scoring ones, and to show the CUI for each candidate and to number them are available. Thresholds can be defined to visualize only those candidates whose score equals or exceeds the specified threshold.

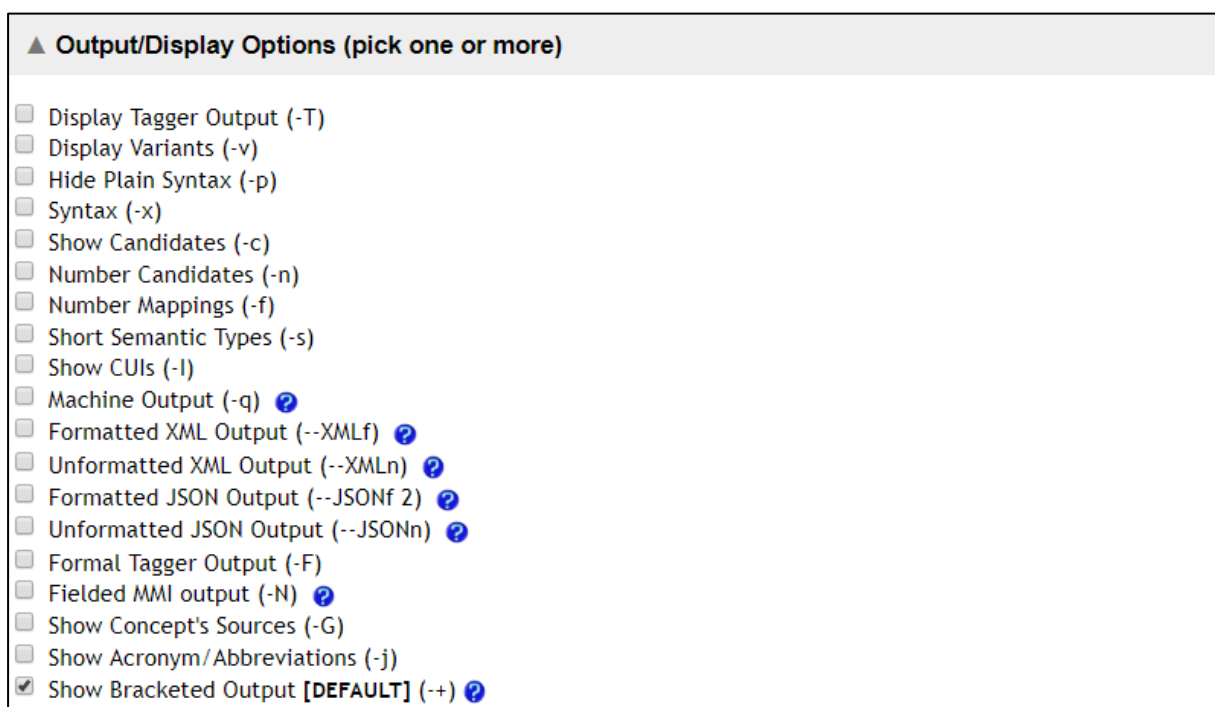


Figure 1.12 - Default MetaMap output options.

Chapter 2

Materials and Methods

Figure 2.1 shows the process flow for the proposed automated method for app classification subdivided into two main modules. The first module (panel A) describes the process for the development of the app database, while the second module (panel B) lists the main processes for the apps classification.

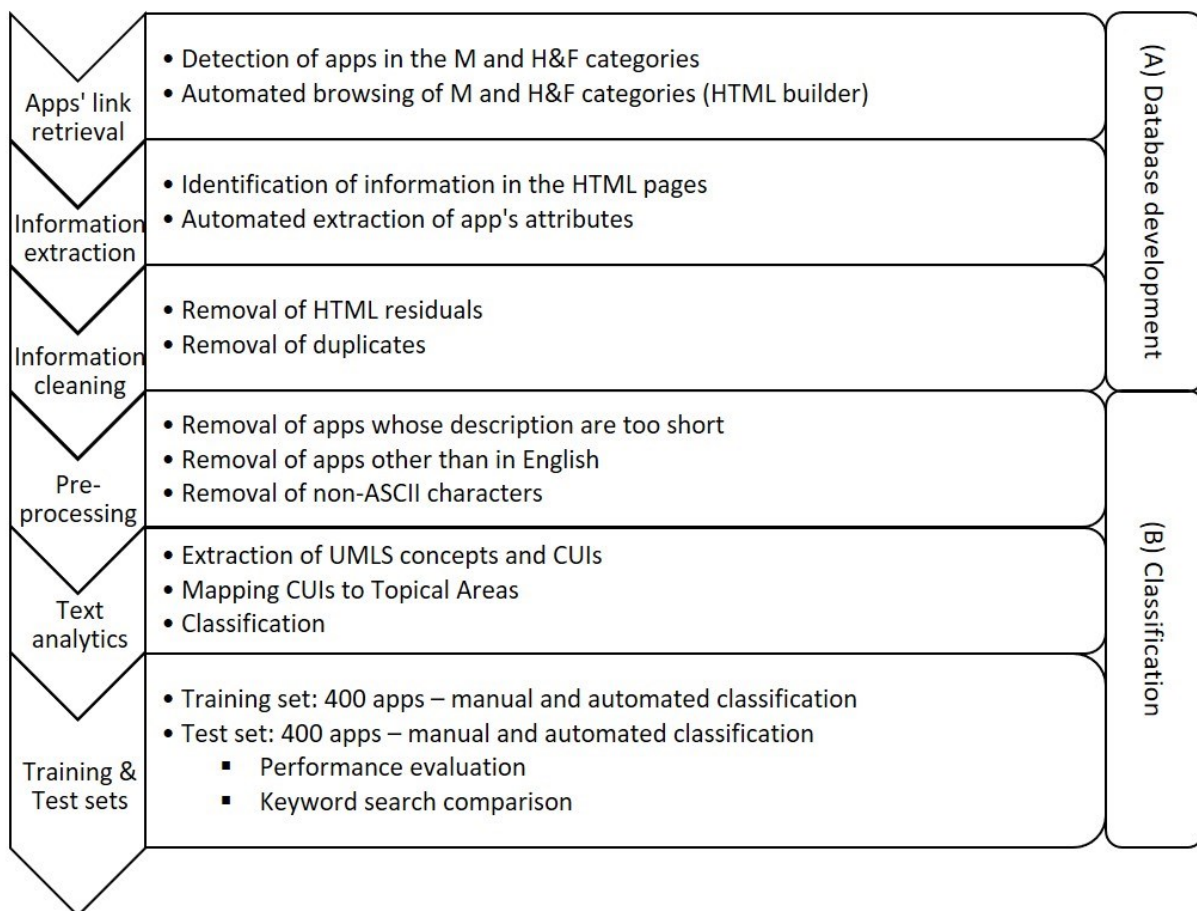


Figure 2.1 - Outline of methodological workflow. Panel A: App database creation. The function HTML builder crawls the iTunes App Store in the Medical and Health&Fitness categories by dynamically building the URLs of the app store webpages where the apps' names and hyperlinks are listed alphabetically so to access each app's webpage. For each webpage the HTML source code is parsed to extract apps' attributes to build the app database. Panel B: Data are pre-processed by removing, based on the app description, apps that are not in English and by converting HTML into plain text by using ASCII (7 bits) characters. Then, UMLS terms and, based on the MeSH hierarchical structure, are extracted with the use of a text analytic tool (Metamap) to characterize apps' features (in this study, the topical areas). Finally, the method presented in this study is evaluated in terms of performance and compared with another method based on keyword search.

2.1 Apps link retrieval

To analyze the apps of medical interest, it was necessary to firstly identify the categories inside the iTunes App Store to retrieve all the links pointing to them. Afterwards, the relevant web pages were downloaded to navigate their source code. Subsection 2.2 describes the methods followed to download all the web pages associated to the apps of interest.

2.1.1 Detection of apps in the M and H&F categories

To automatize the process of apps link retrieval, an analysis of the Apple App Store was performed. M and H&F categories are both identified on the Store by unique Uniform Resource Locators (URLs, i.e. the web addresses):

- “<https://itunes.apple.com/us/genre/ios-medical/id6020?mt=8>” for the M category
- “<https://itunes.apple.com/us/genre/ios-health-fitness/id6013?mt=8>” for the H&F category

Both these URLs point on a page containing a list of apps links alphabetically sorted and grouped into other pages. To visualize pages referring to a specific letter it’s possible to add a parameter in the query string of the URL (in bold).

- “[https://itunes.apple.com/us/genre/ios medical/id6020?mt=8&letter=?](https://itunes.apple.com/us/genre/ios%20medical/id6020?mt=8&letter=?)”
- “<https://itunes.apple.com/us/genre/ios-health-fitness/id6013?mt=8&letter=?>”

The question mark in both URLs can be replaced with the initial letter of the app’s name to visualize the respective list. As some apps’ name starts with non-alphabetical characters like numbers or symbols, to visualize them the question mark can be replaced with the “*” character, representing all the apps whose names don’t start with an alphabetical character. In the pages located by these URLs, a list of pages containing the lists of apps’ links is present. The number of these pages is not a-priori known since the apps number under certain initial letter frequently changes. To visualize a specific

2.1 Apps link retrieval

page, another parameter can be added to the query string, thus obtaining these final URLs:

- "https://itunes.apple.com/us/genre/ios medical/id6020?mt=8&letter=?&page=?#page"
- "https://itunes.apple.com/us/genre/ios-health-fitness/id6013?mt=8&letter=?&page=?#page"

As for letters, also for pages the question mark needs to be replaced with the number of the page to be visualized.

Figure 2.2 shows an example of a complete URL pointing to the first page of apps' list whose name start with the letter "A" in the H&F category.

<https://itunes.apple.com/us/genre/ios-health-fitness/id6013?mt=8&letter=A&page=1#page>

The red highlighted text represents the category
The green highlighted text represents the initial letter of the App's name
The blue highlighted text represents the page numer

Figure 2.2 - Example of complete URL.

2.1.2 HTML builder

To automatically browse all the pages, using a loop the first and second question marks in the sample URL were replaced by substituting them with the entire alphabet and with progressive numbers from one to the number of the last available page, respectively.

All the links built as in *Figure 2.2* point to a page like *Figure 2.3* that contains the links to the apps.

```
<li><a href="https://itunes.apple.com/us/app/ab-workouts-pro/id438441351?mt=8">Ab Workouts Pro</a> </li>
<li><a href="https://itunes.apple.com/us/app/ab-core-back-workout-fitify/id1225874419?mt=8">Ab, core & back Workout Fitify</a> </li>
<li><a href="https://itunes.apple.com/us/app/aba-advantage/id1082700780?mt=8">ABA Advantage</a> </li>
<li><a href="https://itunes.apple.com/us/app/aba-alliance/id1177660756?mt=8">ABA Alliance</a> </li>
<li><a href="https://itunes.apple.com/us/app/aba-data-notebook/id1008564477?mt=8">ABA Data Notebook</a> </li>
<li><a href="https://itunes.apple.com/us/app/aba-data-notepad-behaviors/id1248782710?mt=8">ABA Data NotePad - Behaviors</a> </li>
<li><a href="https://itunes.apple.com/us/app/aba-data-notepad-skills/id1248744125?mt=8">ABA Data NotePad - Skills</a> </li>
<li><a href="https://itunes.apple.com/us/app/aba-drug-card/id1029018464?mt=8">ABA Drug Card</a> </li>
<li><a href="https://itunes.apple.com/us/app/aba-mobile/id1331088366?mt=8">ABA Mobile</a> </li>
<li><a href="https://itunes.apple.com/us/app/aba-benefits/id665510828?mt=8">ABA-Benefits</a> </li>
<li><a href="https://itunes.apple.com/us/app/abaka-health/id1281266807?mt=8">Abaka Health</a> </li>
<li><a href="https://itunes.apple.com/us/app/abano-montegrotto-si/id571232086?mt=8">Abano Montegrotto Si</a> </li>
<li><a href="https://itunes.apple.com/us/app/abaplanet-pro/id989142096?mt=8">AbaPlanet PRO</a> </li>
<li><a href="https://itunes.apple.com/us/app/abate-panic-attacks/id445704203?mt=8">Abate Panic Attacks</a> </li>
<li><a href="https://itunes.apple.com/us/app/abbott-fish-chromosome-search/id422488613?mt=8">Abbott FISH Chromosome Search</a> </li>
```

Figure 2.3 – List of apps' URLs in the first page of apps with A as initial.

There was no static way to understand how many pages for each letter were available on the web, since this information was not present in the Apple App Store. Accordingly, the following pseudo-code, to understand when all the pages were effectively downloaded, was developed:

- 1- letter = k (*Selection of the letter*)
- 2- links = [] (*The list containing all the links retrieved for that is initialized*)
- 3- page = 1 (*The number of the page to browse is set to 1*)
- 4- old = 0 (*Initialization of old*)
- 5- While current page == 1 OR (len(links)-old) > 2
 - a. old = len(links) (*keep track about number of links retrieved before the current page*)
 - b. link = URL_builder(letter, page) (*build the new link to browse*)
 - c. link = re.findall('', requests.get(link).text) (*retrieve links in the current page*)
 - d. links.append(link) (*add the retrieved links to the list of links for selected letter*)
 - e. page += 1 (*set the page to be browsed in the next loop*)
 - f. jump to 4

2.1 Apps link retrieval

Figure 2.4 shows the block diagram of the pseudo-code explained above.

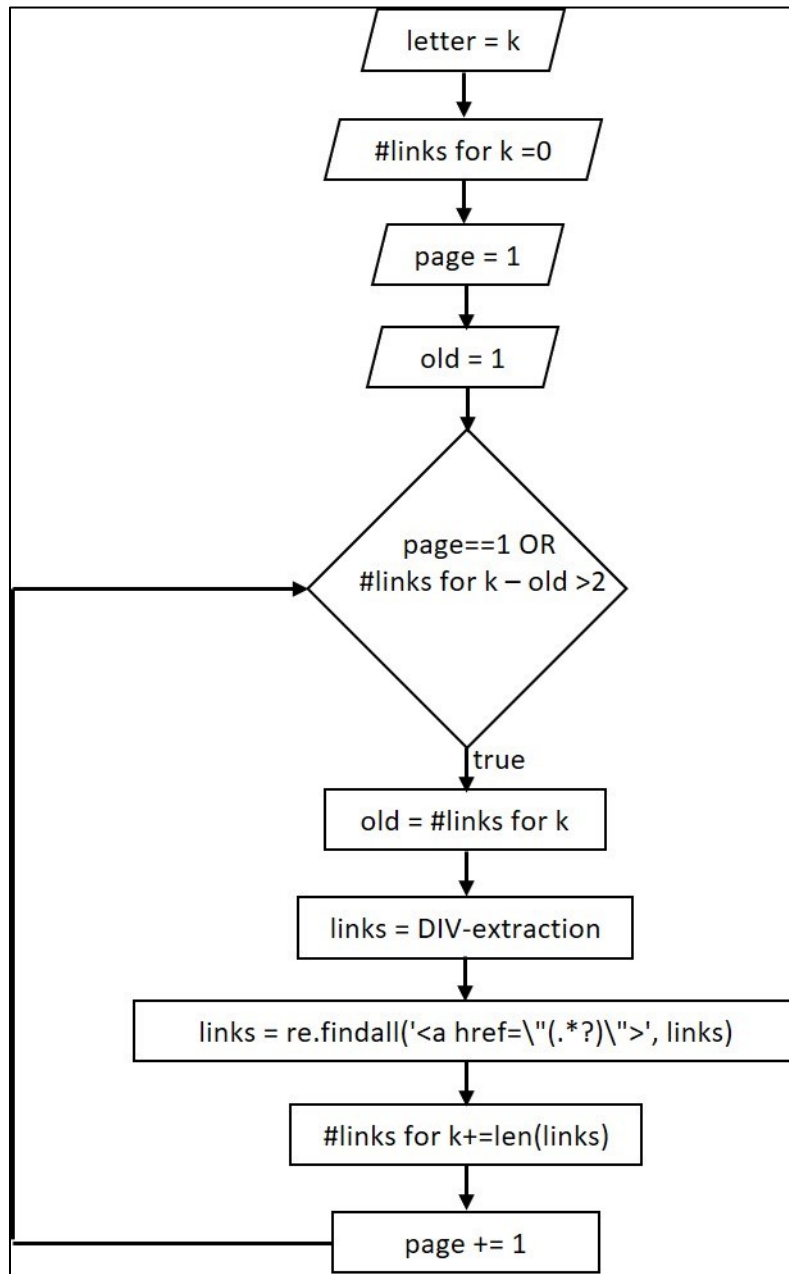


Figure 2.4 - Links retrieving algorithm.

If attempting to download a page that doesn't contain any link, the Apple App Store does not return any kind of error, but it simply shows a page containing a link to one of the last apps with the same letter: for this reason, the specified condition in the while loop has been set as greater than two.

With the algorithm described in Figure 2.4, it was possible to extract all the links pointing to all the apps on the store. Once all the links were extracted, all the source codes of the apps' web pages were downloaded.

2.2 Information Extraction

For a comprehensive description of apps to be used in the further steps, attributes in Table 2.1 were extracted from each app's webpage and saved in a Comma Separated Values (CSV) file. The first column shows the name of the attribute, the second shows the data type of the attribute and the third describes the necessary process to clean and transform the data according to the attribute's type in the second column.

For example, for Average rating stars, "4 and half stars" is translated into "4.5", while for Release date "2016-03-01" is translated into "01/03/2016".

Table 2.1 – Selection of attributes.

Name of the attribute	Type of the attribute	Operations to be done
Id	Long Integer	The Id attribute is the primary key which identifies each app on the store
Name	String	HTML residuals removal
Developer	String	HTML residuals removal
Version	String	-
Language	String	This information on the Web is a list of all the languages for which the app translation is available. The first one has been selected as the language and reported in the database.
SW compatibility	String	-
HW compatibility	String	-
Category	String	-
Keyword	String	HTML residuals removal
Number of ratings (current)	Integer	-
Number of ratings (all)	Integer	-
Average rating stars (current)	Decimal, Single	"and a half stars" needs to be translated into ",5"
Average rating stars (all)	Decimal, Single	-
Reviews	String	HTML residuals removal
Price	Double	"Free" has to be translated into 0 since price is a numerical attribute.
Currency	Char	-
Size	Double	-
Unit of measure	String	-

2.2 Information Extraction

Last update date	Date	m gg, YYYY into gg/mm/YYYY
Release date	Date	YYYY-mm-gg into gg/mm/YYYY
Age rating	Integer	-
Description	String	HTML residuals removal
Contacts	String	-
Url	String	-
Date retrieved	Date	Set by software
App store	String	-
App market	String	-

2.2.1 Identification of information in the HTML pages

To find out the location of the attributes described in Table 2.1 from the HTML source page, a subset of ten HTML pages were randomly selected and manually analyzed. Results are shown in Table 2.2 (the information of interest is highlighted in red).

Table 2.2 – Location of the information in the HTML page.

Attribute	Location of the information
Id	<link rel="canonical" href=https://itunes.apple.com/us/app/heart-rate-monitor-measure-and-track-your-pulse-rate/id795738018?mt=8>
Name	<h1 itemprop="name">iCare Health Monitor-can measure blood pressure</h1>
Developer	<h2>By Beujung Jiajia Kangkang C.o. Ltd</h2>
Version	3.2.1
Language	<li class="language">Languages: English, Arabic, Czech, Dutch, French
SW compatibility	Compatibility:Requires iOS 7.0 or later. Compatible with iPhone, iPad, and iPod touch
HW compatibility	Compatibility:Requires iOS 7.0 or later. Compatible with iPhone, iPad, and iPod touch
Category	Medical
Keyword	<meta name="keywords" content="iCare Health Monitor-can measure blood pressure, Medical, Health, Fitness, iOS, apps, app, Appstore"/>

Number of ratings (current)	<div class="rating" role="img" tabindex="-1" aria-label="4 and a half stars, 45 Ratings" itemprop
Number of ratings (all)	<div class="rating" role="img" tabindex="-1" aria-label="4 and a half stars, 368 Ratings">
Average rating stars (current)	<div class="rating" role="img" tabindex="-1" aria-label="4 and a half stars, 45 Ratings" itemprop
Average rating stars (all)	<div class="rating" role="img" tabindex="-1" aria-label="4 and a half stars, 368 Ratings">
Reviews	You can use with iPhone 6 Plus with two hands
Price	<div itemprop="price" content="0" class="price">Free</div>
Currency	\$ → Fixed
Size	Size:57.0 MB
Unit of measure	Size:57.0 MB
Last Update date	Oct 12, 2016
Release date	Oct 12, 2016
Age rating	>Rated 12+ for the following:
Description	<p itemprop="description" class="truncate" style="height: 54px;"> iCare Health Monitor-Mobile measures...</p>
Contacts	<div class="app-links">
URL	<link rel="canonical" href="http://itunes.apple.com/us/app/icare-health-monitor-can-measure/id1062204827?mt=8">
Date retrieved	Retrieved from the system and not via web
App store	<html prefix="og:http://ogp.me/ns#" xmlns="http://www.apple.com/itunes/" lang="en">
App market	<link rel="canonical" href="http://itunes.apple.com/us/app/icare-health-monitor-can-measure/id1062204827?mt=8">

2.2 Information Extraction

2.2.2 Automated extraction of apps' attributes

Since HTML is a well-structured text, it was possible to use REs to extract the information. Table 2.3 shows for each attribute the relative REs used. If a field was not retrieved from the html page, the corresponding value in the table was set to NULL according to its type defined in this section.

Table 2.3 – Mapping from attributes to RE.

Attribute	Regular expression
Id	This attribute is not extracted from the web
Name	<h1 itemprop="name">(.*?)</h1>
Developer	</h1>\s*<h2>By(.*?)</h2>
Version	(.*?)
Language	<li class="language">Language.?:(.*?)
SW compatibility	<p>Compatibility:Requires iOS(.*?)<or later]>*. Compatible
HW compatibility	.*? Compatible with (.*?)</p>
Category	(.*?)
Keyword	<meta name="keywords" content="(.*?)".*?>
Number of ratings (current)	stars, (\d*) Ratings.*? itemprop=.*?aggregateRating.
Number of ratings (all)	<div>All Versions:</div>\n\s*<div class=.*?rating.*? role=.*?img.*?tabindex=.*?-1.*? aria-label=.*?stars, (\d*) Ratings
Average rating stars (current)	<div>Current Version:</div>\n.*?<div class=.*?rating.*? role=.*?img.*?tabindex=.*?-1.*? aria-label=(.*?), \d* Ratings.*? itemprop=.*?aggregateRating.
Average rating stars (all)	<div>All Versions:</div>\n.*?<div class=.*?rating.*? role=.*?img.*?tabindex=.*?-1.*? aria-label=(.*?), \d* Ratings.*?>
Reviews	Initial tag: <h4>Customer Reviews</h4> Final tag: <h2>Customers Also Bought</h2>
Price	<div itemprop="price" content="(.*?)" class="price">
Currency	This attribute is not extracted from the web
Size	Size: (.*?)\w*
Unit of measure	Size: .*?(\w*)
Last update date	itemprop="datePublished" content="(.*?)Etc/GMT">(.*?)<
Release date	itemprop="datePublished" content="(.*?) \d
Age rating	<div class=.*?app-rating.*>.*?Rated (.*?)\+[for the following]*
Description	<p itemprop="description".*?>(.*?)</p>\s*</div>

Contacts	<div class="app-links"><a rel="nofollow" target="_blank" href="(.*?)" class=".*?
Url	<link rel="canonical" href="(.*?)" .*>
Date retrieved	It's a system function
App store	<html prefix="og: (.*?)">
App market	<link rel="canonical" href="https://itunes.apple.com/(.*?)/.*>

2.3 Information cleaning

Up to this point, M and H&F databases were created. Further details about the resulting composition of these databases are available in the Results chapter.

2.3.1 Removal of HTML residuals

By the use of REs as in Table 2.3, all the information were extracted and also cleaned from HTML residual like " ". HTML residuals are tags or special characters mis-written and thus not correctly interpreted by the browser. This cleaning procedure automatically removes the HTML residuals from a string by performing a simple substitution with a blank character. Special characters are always enclosed between "&" and ";", while tags are enclosed between "<" and ">", thus they are easily recognizable by REs.

2.3.2 Removal of duplicates

Once uploading an app to the iTunes App Store, a developer can assign two categories to the app, a primary and a secondary category. The primary category is particularly important for app's retrieval on the App Store. This will be the category in which the app appears when the user browses the App Store or filters search results, and it determines placement on the Medical tab on the App Store [W13]. As the same app could be, in principle, under two categories, database union was performed to search and remove duplicates. If a developer has chosen both "Medical" and "Health & Fitness" categories for the app, the same app resulted extracted twice by the process described in section 2.2. During this process three binary flags were introduced to keep track of the origin of the tuples in the unified database: "net HF", "net Med" and "net Both". For the first two flags, the value is equal to 1 if the tuple in the database belongs to M or H&F database, respectively.

2.4 Pre-processing

In case the tuple belongs to both, the value of the third flag will be 1 as the result of the logical operation AND between the first two. The removal process was based on the apps' id since it uniquely identifies apps on the Store.

2.4 Pre-processing

The analyses to be performed later (i.e., text analytics) required some pre-processing operations to prepare the data in the correct format because the utilized tool (MetaMap) is able to parse and analyze only English plain text (ACII 7 bits).

2.4.1 Removal of apps whose description are too short

Apps whose description length was shorter than 14 characters were removed. This delimiter was decided according to the forced attribute "Not available" whose length is of 13 characters. These descriptions are too short to contain useful information for the analyses and can be removed without further loss. Some examples are provided in *Table 2.4*.

Table 2.4 – Examples of apps with descriptions shorter than 14 characters.

Name of the app	Description of the app
Alana K Macalik	Consent form
ATP Training	ATP Trainin
FIT STUDIO App	FIT Studio Ap
Human Pyramid	MAKE IT !!

2.4.2 Removal of apps other than in English

Removal of apps in languages other than English was necessary as the developed method uses tools for English text analytics. In fact, even if the US store was crawled, for several apps the description was not provided in English. Some examples are shown in Table 2.5.

Table 2.5 – Examples of descriptions other than in English.

Name of the app	Description of the app
A×O_a-by-o	千葉県若葉区みつわ台にある美容室A×Osince1988【A-BY-O】の公式アプリです。 ・アプリからお得な情報を配信しま
ADPM Falcão Azul	Você esta pronto para melhorar sua Saúde e ter uma qualidade de vida muito melhor? Uma boa academia esta...
Autogenes Training – gesund und stressfrei durch Entspannung	Endlich entspannt und stressfrei mit der bekannten Entspannungsmethode Autogenes Training.Mithilfe des Autogenen Trainings, einer sanften Sie versenken sich bei dieser...

Language detection was performed by using the Language Detection (langdetect) library ported from Google's language-detection. This library is a direct port of Google's library from Java to Python. It supports 58 languages out of the ISO 639-1 codes. ISO 639 [W14] is a standardized nomenclature used to classify languages. Each language is assigned a two-letter (639-1) and three-letter (639-2 and 639-3), lowercase abbreviation, amended in later versions of the nomenclature. The system is highly useful for linguists and ethnographers to categorize the languages spoken on a regional basis, and to compute analysis in the field of lexicostatistics. The supported languages are: Afrikaans, Arabic,

2.5 Text analytics

Bulgarian, Bengali, Czech, Danish, German, Greek (modern), English, Spanish, Castilian, Estonian, Persian, Finnish, French, Gujarati, Hebrew (modern), Hindi, Croatian, Hungarian, Indonesian, Italian, Japanese, Kannada, Korean, Lithuanian, Latvian, Macedonian, Malayalam, Marathi, Nepali, Dutch, Flemish, Norwegian, Panjabi, Punjabi, Polish, Portuguese, Romanian, Moldavian, Moldovan, Russian, Slovak, Slovenian, Somali, Albanian, Swedish, Swahili, Tamil, Telugu, Thai, Tagalog, Turkish, Ukrainian, Urdu, Vietnamese, Chinese, Twi.

The algorithm uses Bayesian filter and returns the language with the highest probability.

2.4.3 Removal of NON-ASCII characters

As MetaMap can extract information related to medical concepts only from plain non-ASCII text (7 bits), all the non-ASCII characters were removed. ASCII characters are the first 128 ones, so by getting the number of each character and strip them if out of range, it was possible to obtain as output the input string without non-ASCII characters.

This step was performed as the last step of the pre-processing operations since removal of non-ASCII characters before the language detection step would have altered the language detection of some descriptions.

2.5 Text analytics

To understand the topical areas relevant to each app, it was necessary to define the areas among which the search needed to be made.

For a comprehensive description of the possible topical areas, we considered the following:

- Fitness & Wellness (that does not necessarily coincide with the 'Health & Fitness' category on the App store;
- A comprehensive list of medical specialties derived from the Union Européenne des Médecins Spécialistes (UEMS) [W15], i.e.: Cardiology and Cardiovascular Medicine; Dermatology; Emergency Medicine; Gastroenterology, Hepatology, and Nephrology; Gynecology & Obstetrics and Neonatal care; Immunology & Endocrinology; Mental Health,

Neurology, Psychiatry; Oncology; Physiatry and Orthopedics; Pneumology (including Sleep and Respiratory care); Sensory Systems Healthcare (including hearing healthcare, Ear-Nose-Throat, vision healthcare, vestibular medicine, and speech and language therapy); Surgery;

- Nutrition;
- Dentistry.

Nutrition and Dentistry are not formally recognized as medical specialties by the UEMS but represent relevant medical areas and, as such, were included in the analysis.

Whenever an app was related to general medicine, medical education, nursing, or healthcare rather than to one or more topical areas among those listed above, it was classified as 'across specialties'. Whenever an app was not related to health or medicine (e.g., entertainment, games, business apps) or whenever its description was not informative about its content (e.g. "use this app to schedule your classes") it was classified as "NC" (i.e., No Content related to medicine or health).

Once this set of topical areas has been defined, it was necessary to decide what to be analyzed to retrieve as much information as possible about what an app is claiming to do. For this reason, among the attributes listed in Table 2.1, the description field was selected as the best candidate for this step. The app's description is an unstructured text in which the developer states the most important features of the app. To process this unstructured text, the MetaMap tool was used, as described in the following subparagraphs.

2.5.1 Extraction of UMLS concepts and CUIs

Figure 2.5 schematizes the followed steps to obtain the output useful for the extraction of UMLS concepts. MetaMap requires text as input, together with several optional settings, to produce an output that contains mappings of all the phrases in the input text.

The output of MetaMap is not all useful for our analysis, so those unnecessary portions were removed, thus limiting it to core information only.

2.5 Text analytics

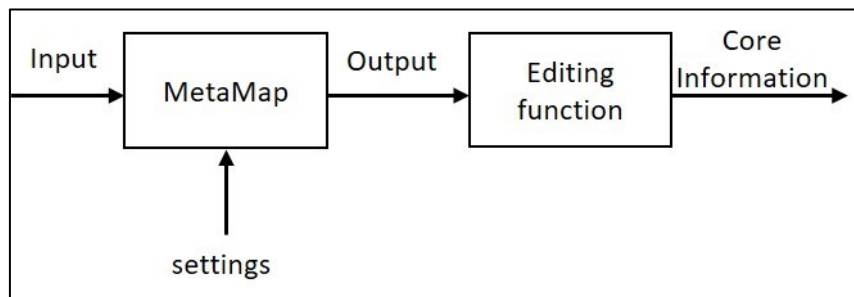


Figure 2.5 – Workflow for MetaMap use.

A) Input

MetaMap can be utilized in different ways and according to its usage the required input modality changes, as shown in *Table 2.6*.

Table 2.6 – MetaMap upload modalities.

MetaMap usage method	Input modality
Via web interactively	English plain text directly typed via web
Via batch upload	English plain text uploaded as a txt file
Via API	Method called on a variable that contains English plain text

The batch upload method was chosen in this study since once the input has been validated and accepted on the server’s side, the execution took place entirely on the server, thus not using computational resources on the user’s side. In this form of execution, the input included a series of app descriptions separated by “\n” characters.

Moreover, for easy retrieval of MetaMap output from the total output file, the input was formatted as in *Figure 2.6*, i.e. the apps’ ID were incorporated in the input text file.

```
341232718|Whether you want to lose weight, tone up, get healthy...  
462638897|Live a healthier, more active life with Fitbit...  
321367289| Medscape provides fast and accurate clinical answers...
```

Figure 2.6 – Example of MetaMap Input for batch upload where the number at the beginning of the sentence is the app’s ID (i.e. ‘341232718’ is the app ID).

B) Settings

Data options determine the underlying vocabularies and data model used by MetaMap. The default data version (USAbase) was selected.

Furthermore, options regarding restriction of Semantic Types were considered: 49 out of the 129 Semantic Types available were selected as relevant, whereas 80 were excluded as not relevant or misleading, as described in *Table 2.7*. This selection reduced the computation time, as MetaMap run only on the relevant Semantic Types.

Table 2.7 – Semantic Types excluded.

Semantic Type	Reason of discard
Carbohydrate Sequence, Cell Component, Chemical, Chemical Viewed Functionally, Chemical Viewed Structurally, Amino Acid Sequence, Antibiotic, Amino Acid Peptide or Protein, Gene or Gene Product, Gene or Genome, Genetic Function, Molecular Biology Research Technique, Molecular Function, Molecular Sequence, Nucleic Acid Nucleoside or Nucleotide, Element Ion or Isotope, Indicator Reagent or Diagnostic Acid, Cell Function, Nucleotide Sequence, Organic Chemical, Pharmacologic Substance, Receptor	They refer to term that are too specific to be found in an app's description and are poorly informative about the topical area.
Amphibian, Animal, Plant, Bird, Fish, Mammal, Reptile, Vertebrate	They refer to living beings that are not considered in the thesis.
Qualitative Concept, Quantitative Concept, Regulation or Law, Social Behavior, Manufactured Object, Object, Phenomenon or Process, Physical Object	They refer to a set of terms that are descriptive and not related to topical areas.
Geographic Area, Governmental or Regulatory Activity, Group, Group Attribute, Inorganic Chemical, Intellectual Product, Idea or Concept, Individual Behavior, Machine Activity, Spatial Concept, Temporal Concept, Human-caused Phenomenon or Process	They refer to a set of terms that are un-relevant for the analysis and not related to topical areas.
Archaeon, Cell or Molecular Dysfunction, Drug Delivery Device, Environmental Effect of Humans, Enzyme, Eukaryote, Experimental Model of Disease, Fully Formed Anatomical Structure, Fungus, Hazardous or Poisonous Substance, Hormone, Immunologic Factor, Professional Society, Research Device, Self-help or Relief Organization, Substance, Vitamin	They were initially retained and tested on the training set. As no results were found across the 400 apps in the training set, they were then discarded.

2.5 Text analytics

Injury or Poisoning, Biologically Active Substance, Event, Functional Concept, Conceptual Entity, Research Activity, Language, Population Group, Educational Activity, Human, Entity, Classification, Activity	They refer to a set of terms that are too generic to be correlated to a topical area.
--	---

As a result, the list of the 49 relevant Semantic Types included in the analysis was the following: "Acquired Abnormality", "Age group", "Anatomical Abnormality", "Anatomical Structure", "Bacterium", "Behavior", "Biologic Function", "Biomedical Occupation or Discipline", "Biomedical or Dental Material", "Body Location or Region", "Body Part, Organ, or Organ Component", "Body Space or Junction", "Body Substance", "Body System", "Cell", "Clinical Attribute", "Clinical Drug", "Congenital Abnormality", "Daily or Recreational Activity", "Diagnostic Procedure", "Disease or Syndrome", "Embryonic Structure", "Family Group", "Finding", "Food", "Health Care Activity", "Health Care Related Organization", "Laboratory Procedure", "Laboratory or Test Result", "Medical Device", "Mental Process", "Mental or Behavioral Dysfunction", "Natural Phenomenon or Process", "Neoplastic Process", "Occupation or Discipline", "Occupational Activity", "Organ or Tissue Function", "Organism", "Organism Attribute", "Organism Function", "Organization", "Pathologic Function", "Patient or Disabled Group", "Physiologic Function", "Professional or Occupational Group", "Sign or Symptom", "Therapeutic or Preventive Procedure", "Tissue", "Virus".

C) Output

The output of the result of MetaMap execution in batch upload mode was one unique output txt-file. This txt-file was split into multiples files, one for each description given in input. Figure 2.7 and Figure 2.8 show two examples of MetaMap output starting from a simple phrase. Each phrase might be mapped multiple times by combining different variants in its terms. MetaMap assigns to each mapping a score in the range 0-1000, reported on the left. In this study, no selection has been done on the mapping even if the higher mapping scores represent better than the lower the reality. However, it is known that medical terms could have different interpretations, and this fact needed to be taken into consideration.

```
>>>> Mappings
Meta Mapping (732):
  793  C3687603:Involved in training [Finding]
  744  C0243107:development (development aspects) [Physiologic Function]
Meta Mapping (732):
  793  C3687603:Involved in training [Finding]
  744  C0678723:Development (Biologic Development) [Organism Function]
Meta Mapping (793):
  793  C3687603:Involved in training [Finding]
<<<<< Mappings
```

Figure 2.7 – Example of MetaMap output.

Figure 2.8 shows an example of MetaMap output for an input sentence containing medical terms. To each mapping an overall score is assigned (e.g., 706 in Figure 2.8).

In addition, for each identified concept, MetaMap provides the following information (Figure 2.8):

- The concept's CUI (e.g. C0179432, C0819141),
- the concept's score (e.g. 748, 612),
- the UMLS string matched (e.g. Bronchoscope, Bronchial Tree),
- the concept's Preferred Name (e.g. Bronchoscopes, Bronchial tree),
- the concept's Semantic Type(s) (e.g. [Medical Device], [Body Part, Organ or Organ Component]).

```
Phrase: a HDTV bronchoscope through the bronchial tree
>>>> Phrase
a hdtv bronchoscope through the bronchial tree
<<<<< Phrase
>>>> Mappings
Meta Mapping (706):
  748  C0179432:Bronchoscope (Bronchoscopes) [Medical Device]
  612  C0819141:Bronchial Tree (Bronchial tree) [Body Part, Organ, or Organ Component]
<<<<< Mappings
```

Figure 2.8 – MetaMap output core information.

This core information is important for further analysis whereas the remaining can be discarded; to this aim, the output text was automatically edited and converted into a more compact format, as described in the following subparagraph.

2.5 Text analytics

D) Editing Function

In Figure 2.8 the information useful for further analysis is highlighted in orange. All the remaining text can be discarded. To extract useful information (scores, CUIs, UMLS strings matched, Concepts Preferred Name, Concepts Semantic Types) the following RE has been used: “\n(\s+\d+.*?)”.

After this editing process, the final output containing core information looked like the example shown in Figure 2.9. These portions of information were used to classify app’s descriptions into topical areas as described in section 2.5.3.

```
748 C0179432:Bronchoscope (Bronchoscopes) [Medical Device]
612 C0819141:Bronchial Tree (Bronchial tree) [Body Part, Organ, or Organ Component]
753 C0225594:CARINA (Structure of Carina) [Body Part, Organ, or Organ Component]
753 C1278906:Carina (Entire Carina) [Body Part, Organ, or Organ Component]
1000 C0037744:Spatial Orientation (Space Perception) [Mental Process]
770 C0026649:Movement [Organism Function]
604 C0179432:Bronchoscope (Bronchoscopes) [Medical Device]
966 C0237607:experience (Practice Experience) [Mental Process]
966 C0596545:Experience [Mental Process]
694 C0199168:Medical (Medical service) [Health Care Activity]
793 C3687603:Involved in training [Finding]
744 C0243107:development (development aspects) [Physiologic Function]
793 C3687603:Involved in training [Finding]
744 C0678723:Development (Biologic Development) [Organism Function]
793 C3687603:Involved in training [Finding]
```

Figure 2.9 – Edited output containing only core information.

2.5.2 Mapping CUIs to topical areas

Concepts, as fundamental unit in Metathesaurus, represent a single meaning and contain all atoms from every source that expresses that meaning in every way, whether formal or casual, verbose or abbreviated. All atoms within a concept are synonymous, and each concept is assigned to one or more Semantic Types. To every concept a CUI is assigned (e.g. C0179432, C0819141 as in Figure 2.9), to uniquely identify that single meaning.

In order to map each concept to the relevant topical areas, it was necessary to build a relation between the topical areas and the CUIs found with the MetaMap analysis.

Figure 2.10 shows the steps followed to obtain this relation, as described in the following subparagraphs.

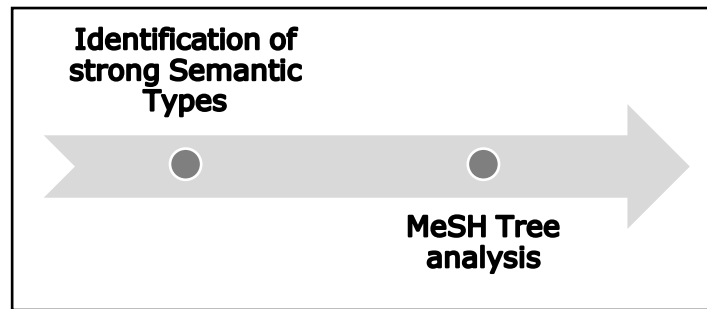


Figure 2.10 - Mapping workflow from CUIs to topical areas.

A) Identification of strong Semantic Types

To build an association between CUIs and topical areas, an analysis of the MetaMap filtered output was performed to search for strong Semantic Types. Strong Semantic Types are those that might be directly linked to a topical area. This analysis allowed discovering two strong semantic types:

1. [Food], that can be directly linked to "Nutrition",
2. [Daily or Recreational Activity], that can be directly linked to "Fitness & Wellness".

However, some of the CUIs belonging to these Semantic Types were misleading and were discarded.

Table 2.8 shows some example of CUIs belonging to these Semantic Types but not directly linkable respectively to "Nutrition" and "Fitness & Wellness". These type of CUIs were discarded as ambiguous and not informative.

2.5 Text analytics

Table 2.8 – Examples of CUIs discarded.

CUI	Associated Concept	Reason for discarding
C0359589	provides (Provide (product))	The act of providing food is no strictly related to "Nutrition"
C0475653	APPLE	Since the Apple App Store has been considered in this study, Apple can be an ambiguous term
C1875856	VITAL (Vital High Nitrogen Enteral Nutrition)	This word is often used in the health domain and rarely used referring to Nutrition because it's too specific
C0034754	Reading (Reading (activity))	This is a daily activity but not related with Fitness
C2136029	Listening to music	This is a recreational activity but not related with Fitness

Table 2.9 shows instead some examples of CUIs that are directly linkable to "Nutrition" and/or to "Fitness & Wellness" topical areas. These type of CUIs were highly informative and contributed to the identification of topical areas.

Table 2.9 – Examples of CUIs included.

CUI	Associated Concept
C0028707	Nutrition (Science of nutrition)
C0012159	Diet (Diet therapy)
C1262477	WEIGHTLOSS (Weight decreased)
C0035953	Running (Running (physical activity))
C1456706	Fitness
C0038039	Sport (Sports)
C0238703	Athletes

B) MeSH Tree analysis

To find a set of CUIs for all the other topical areas, the MeSH tree was analyzed to find out the medical heading of interest so to build rules for matching CUIs to topical areas. Table 2.10 shows, both in code and descriptive form, for each topical area, the upper node containing the medical headings of interest.

Afterwards, the list of medical headings was analyzed with MetaMap, to retrieve the CUIs codes for each topical area.

Chapter 2

Materials and Methods

Table 2.10 – Relevant MeSH.

Topical area	MeSH nodes	Description
Cardiology and Cardiovascular Medicine	G09	Circulatory and Respiratory Physiological Phenomena
	A07	Cardiovascular System
	E01.370.600.875.249	Blood Pressure
	E01.370.600.875.500	Heart rate
	H02.403.429.163	Cardiology
	C14	Cardiovascular Diseases
	E01.370.370	Diagnostic Techniques, Cardiovascular
	E01.370.405.240	Electrocardiography
	G11.427.494.570	Myocardial contraction
	C23.550.073	Arrhythmias, Cardiac
	N02.360.810.128	Cardiologists
Dentistry	A14.549.167	Dentition
	C07.793	Tooth Disease
	M01.526.485.330	Dentists
	F01.829.401.650.410	Dentist-patient relationship
Dermatology	A17.815	Skin
	H02.403.225	Dermatology
	M01.526.485.810.215	Dermatologists
	C17	Skin and Connective Tissue Diseases
Emergency medicine	H02.403.250	Emergency Medicine
	N02.421.297	Emergency Medical Services
Immunology & Endocrinology	H01.158.782.323	Endocrinology
	H02.403.429.323	Endocrinology
	M01.526.485.810.303	Endocrinologists
	C19	Endocrine System Diseases
	A06	Endocrine System
	C18.452.394	Glucose Metabolism Disorders
	C19.246	Diabetes Mellitus
	C20.111.327	Diabetes Mellitus, Type 1
Gastroenterology, Hepatology, and Nephrology	H02.403.429.405	Gastroenterology
	C06.405	Gastrointestinal Diseases
	E04.210	Digestive System Surgical Procedures
	M01.526.485.810.438	Gastroenterologists
Gynaecology and Obstetrics and Neonatal care	G08.686	Reproductive Physiological Phenomena
	E07.325.569	Incubators, Infant
	H02.403.763.750	Gynecology

2.5 Text analytics

	H02.403.810.450	Obstetrics
	C13	Female Urogenital Diseases and Pregnancy Complications
Mental Health, Neurology, Psychiatry	A08	Nervous system
	C10	Nervous system diseases
	N02.360.810.652	Neurologists
	I03.450.769	Relaxation
	I03.450.769.647	Rest
	F03	Mental disorders
Nutrition	G07.203	Diet, Food and Nutrition
	N01.224.425.525	Nutritional Status
	M01.526.485.695	Nutritionists
	N02.360.695	Nutritionists
Oncology	C04	Neoplasms
	E02.319.170	Chemotherapy, Adjuvant
	C23.550.727	Neoplastic Processes
	M01.526.485.810.699	Oncologists
Pneumology	F02.830.855	Sleep
	G11.561.803	Sleep
Sensory system healthcare	F02.830.816.263	Hearing
	F02.830.816.643	Smell
	F02.830.816.724	Taste
	F02.830.816.781	Thermosensing
	F02.830.816.850	Touch
	F02.830.816.964	Vision, Ocular
	G04.835	Signal trasduccion
	G11.561.790.263	Hearing
	G11.561.790.643	Smell
	G11.561.790.724	Taste
	G11.561.790.781	Thermosensing
	G11.561.790.850	Touch
	G11.561.790.964	Vision, Ocular
	G14	Ocular Physiological Phenomena
	G11.427.690	Postural Balance
	E01.370.382.375	Hearing Tests
	E01.370.382.637	Otoscopy
	E01.370.380.850	Vision Tests
	E01.370.382.900	Vestibular Function Tests
	F02.463.593	Perception
	C11	Eye Disease
	C09.218	Ear Disease
	A04.531	Nose
	A01.456.505.420	Eye

	A09.371	Eye
	A09.246	Ear
	A01.456.313	Ear
Surgery	M01.526.485.810.910	Surgeons
	E07.858	Surgical Equipment
	E04	Surgical Procedures, Operative

2.5.3 Classification

Classification consists in predicting a certain outcome based on a given input. To predict the outcome, the algorithm processes a training set containing a set of attributes and the respective outcome, usually called goal or prediction attribute. The algorithm tries to discover relationships between the attributes that would make it possible to predict the outcome. Then, the algorithm is given a data set different than the training set, called prediction set, containing the same set of attributes, except for the prediction attribute – not yet known. The algorithm analyses the input and produces a prediction, where the prediction accuracy defines how “good” the algorithm is.

For classification of apps into topical areas, both the mapping between CUIs and topical areas, as well as the score that MetaMap assigns to each identified concept, were considered. The classification does not need to be forced into only one topical area, as an app might be classified into more than one. For example, an app that lets the user keep track of his/her own vital signs during fitness is also related to concepts relevant to cardiology, as for example heart rate and heart rate variability can be measured during fitness. Moreover, an app that reminds the user to drink at a fixed interval can be classified into both Nutrition and Wellness, as dehydration is a medical pathology but drinking water can also be considered more generally as a wellness practice not related to dehydration. Another example is provided by an app that helps the user to control his breath. This can reduce the level of stress in painful or difficult moment, but it also can help respiratory activity in general, as we do not know the final use of the app, so such apps might be classified both in “Fitness & Wellness” and in “Sleep and Respiratory Care” topical areas.

2.5 Text analytics

For the sake of classification, two scores were introduced, namely the Global Score (GS) and the Average Score (AS). GS is the sum of the scores whose related concepts belongs to a topical area, while AS is the average of the scores whose related concepts belongs to a topical area. Figure 2.11 shows the steps followed to compute these scores.

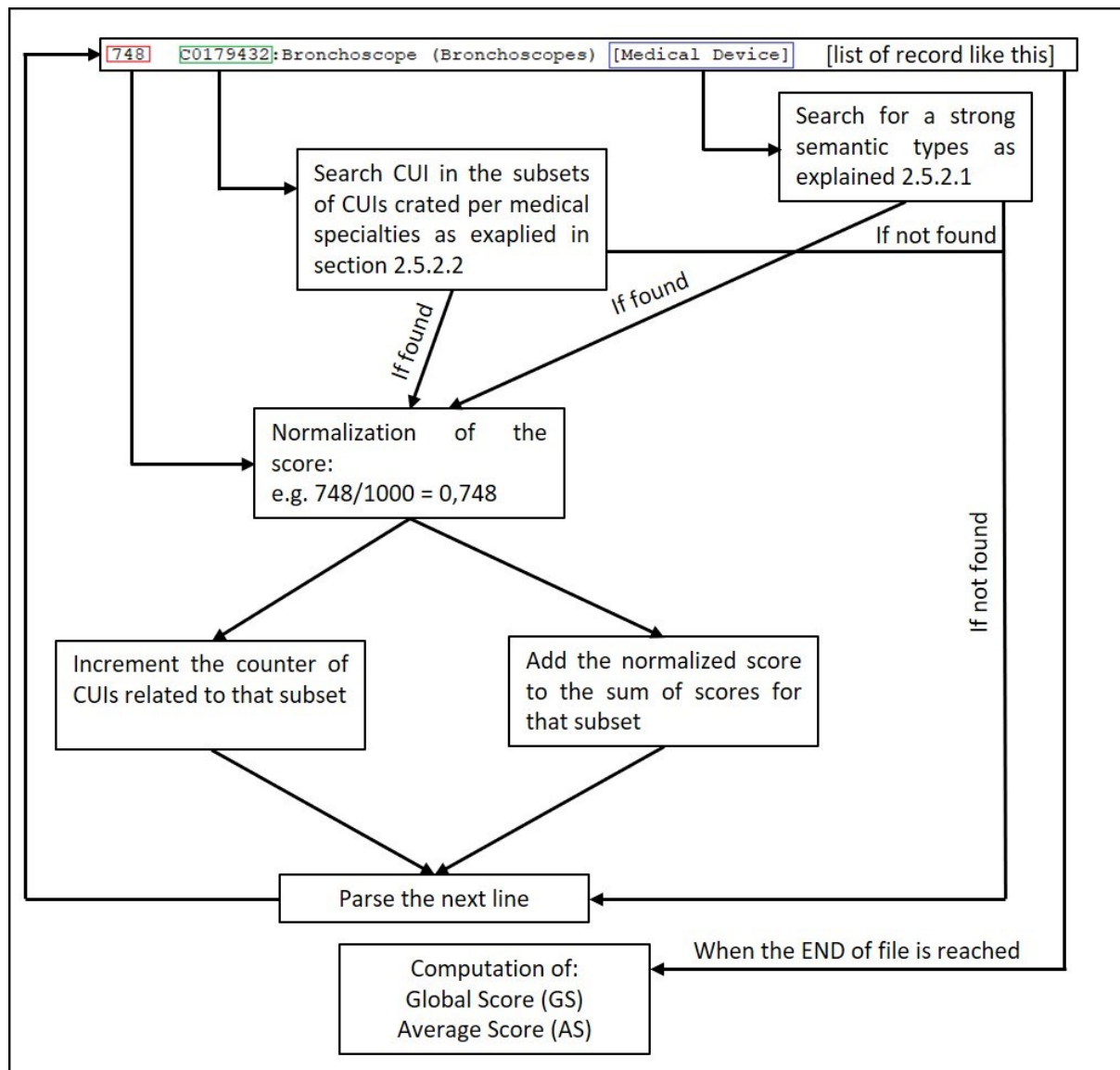


Figure 2.11 – Classification function.

The GSs were taken into consideration if their values were higher than 4: this guarantees a minimum of 4 concepts identified in the app’s description. The ASs instead were taken into consideration if their values were computed from at least one concept with a score value greater or equal than 0.8: this guarantees, in case of unique concept identified, a higher rate of similarity with the reality, as the score

MetaMap assigns is a probability for a concept to be correctly interpreted among a phrase.

Also, for ASs as for GSs, values higher than or equal than the 90% of the highest score found were considered: this allows multiple classification for topical area with similar scores. More details are reported in section 2.6.

2.6 Training & Test sets

A set of 800 apps were randomly extracted from the app database. This set was randomly subdivided into two subsets of 400 apps to build a training set and a test set. The training set was used to build and optimize the classification function whereas the test set was used to verify the performance for the classification approach.

The 800 apps were manually coded and classified into one or more topical areas based on the app description. Additional information, e.g. app snapshots, customer reviews, or linked websites were not considered as the developed automated method works on the app description only.

Figure 2.12 shows the steps followed to define the classification function explained in section 2.5.3.

2.6 Training & Test sets

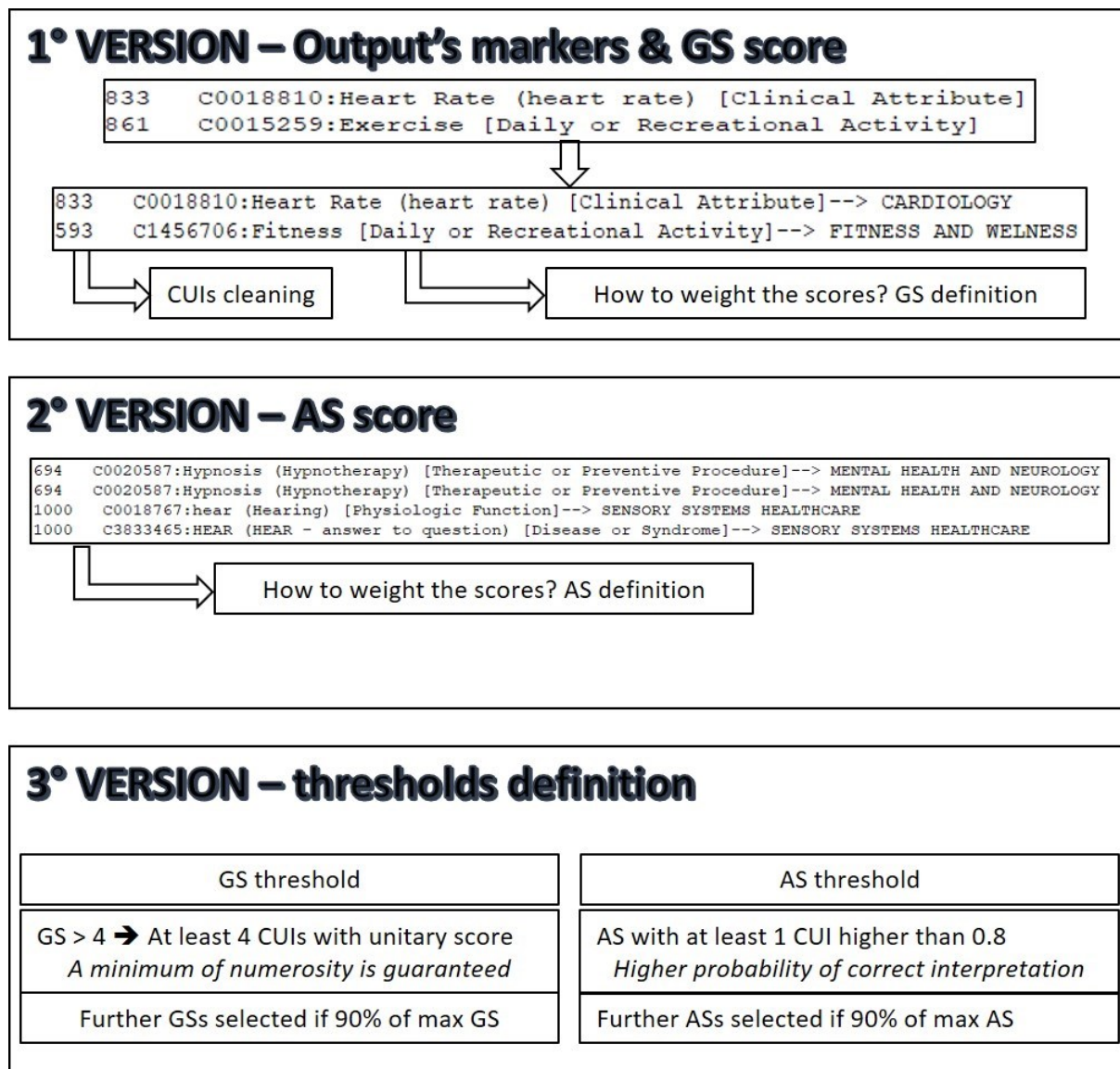


Figure 2.12 – Classification function definition.

In particular, three different strategies were explored:

- 1) MetaMap output containing core information (Figure 2.5) was parsed to produce a new output containing the mappings between the output itself and the CUIs subsets. The new output was manually analyzed to find out CUIs mis-assigned to a topical area. This was needed as once MeSH trees were analyzed, all CUIs coming from MetaMap output were retrieved and put in the relevant subset but, for example, "Tachycardia, Ectopic Junctional" is a medical heading referring to Cardiology but "Ectopic" identifies another concept which means the abnormal position of a part or organ, thus related to another topical area (Physiatry and Orthopedics). However, since there are over 28.000 descriptors in MeSH

Chapter 2 Materials and Methods

with over 90.000 entry terms, it was unfeasible to analyze all the MeSH MetaMap output to remove the undesired CUIs.

- 2) The GSs scores were computed. Both the number of concepts and their scores are important. A concept may be considered many times in different mappings of the same phrase, or it has more than one occurrence in the description, thus it has high number of occurrences in the output, but on the other side it could be possible that a concept with less occurrences could have higher score. If two concepts belong to the same topical area, it's clearly not to be a problem but if they belong to different areas, a method to evaluate them is needed.

Figure 2.13 shows an example in which the number of concepts and their relevance are in conflict. The identified "Nutrition"-related concepts are greater than those related to "Diabetes and Care" even if they have a lower score.

694	C0241863:DIABETIC (diabetic) [Finding]--> DIABETES CARE
623	C0241863:DIABETIC (diabetic) [Finding]--> DIABETES CARE
1000	C0011847:Diabetes [Disease or Syndrome]--> DIABETES CARE
1000	C0011849:Diabetes (Diabetes Mellitus) [Disease or Syndrome]--> DIABETES CARE
1000	C0011847:Diabetes [Disease or Syndrome]--> DIABETES CARE
1000	C0011849:Diabetes (Diabetes Mellitus) [Disease or Syndrome]--> DIABETES CARE
694	C0012155:Diet [Food]--> NUTRITION
694	C0012155:Diet [Food]--> NUTRITION
694	C0012159:Diet (Diet therapy) [Health Care Activity]--> NUTRITION
694	C0012159:Diet (Diet therapy) [Health Care Activity]--> NUTRITION
694	C0012159:Diet (Diet therapy) [Health Care Activity]--> NUTRITION
694	C1519433:DIET (Special Diet Therapy) [Therapeutic or Preventive Procedure]--> NUTRITION
694	C1519433:DIET (Special Diet Therapy) [Therapeutic or Preventive Procedure]--> NUTRITION
694	C1519433:DIET (Special Diet Therapy) [Therapeutic or Preventive Procedure]--> NUTRITION
694	C2983588:DIET (Basal Diet) [Food]--> NUTRITION
694	C2983588:DIET (Basal Diet) [Food]--> NUTRITION
694	C2983588:DIET (Basal Diet) [Food]--> NUTRITION
694	C3668949:Diet (Diet (animal life circumstance)) [Food]--> NUTRITION
694	C3668949:Diet (Diet (animal life circumstance)) [Food]--> NUTRITION
694	C3668949:Diet (Diet (animal life circumstance)) [Food]--> NUTRITION
861	C0012155:Diet [Food]--> NUTRITION
861	C0012159:Diet (Diet therapy) [Health Care Activity]--> NUTRITION
861	C1519433:DIET (Special Diet Therapy) [Therapeutic or Preventive Procedure]--> NUTRITION

Figure 2.13 - Example of number of concepts vs. relevance.

2.6 Training & Test sets

To overcome this problem, a weighted score – AS score – was introduced: AS is a score that averages the GS on the counter of CUIs per topical area.

- 3) Thresholds, as final step in Figure 2.12, were defined. These thresholds were selected after that the second version of the algorithm was launched on the training set.

This selection allowed including those topical areas that resulted to be closer to the ones that had the highest score.

2.6.1 Test set: performance evaluation

Evaluation metrics for multi-label classification performance are quite different from those used in multi-class and binary classification, due to the differences in the classification problem. In extending a binary metric to multi-label problems, the data is treated as a collection of binary problems, one for each label. There are several ways to average binary metric calculations across the set of classes, each of them may be useful in some scenario [Read *et al*, 2011].

Metrics used to evaluate performance are the followings [Godbole & Sarawagi, 2004]:

- General accuracy: the accuracy computed by averaging the single label accuracy.
- Exact match: the set of labels predicted for a sample must exactly match the corresponding set of true labels.
- Hamming loss: it's the fraction of labels that are incorrectly predicted.
- Recall: it's the ability of the classifier to find all positive samples
- Precision: it's the ability of the classifier not to label as positive a sample that is negative
- F1-score: it can be interpreted as a weighted average of the precision and recall.

Recall, Precision and F1-score can be computed in three different ways:

- Micro: metrics are globally computed by counting the total true positives, false negatives and false positives.

- Macro: metrics are computed for each label, and then their unweighted mean is found. This does not take label imbalance into account.
- Samples: metrics are computed for each instance, and then their average is found (*this is meaningful only for multilabel classification where this differs from accuracy score*).

To compute these metrics, it was necessary to define a confusion matrix for each topical area.

Table 2.11 shows an example of a confusion matrix [Stehman, 1997] that results in:

- “true positives” (TP) for correctly predicted topical area values
- “false positives” (FP) for incorrectly predicted topical area values
- “true negatives” (TN) for correctly predicted no-topical area values, and
- “false negatives” (FN) for incorrectly predicted topical area values.

Table 2.11 - Example of confusion matrix.

		Actual class	
		Topical area	Non-Topical area
Predicted class	Topical area	True Positives (TP)	False Positives (FP)
	Non-topical area	False Negatives (FN)	True Negatives (TN)

2.6 Training & Test sets

Once these measures were defined, it was possible to compute metrics with formulas as in Table 2.12.

Table 2.12 - Formulas to compute metrics.

Metric	Formula
General Accuracy	$\frac{\sum_{i=0}^N (TP_i + TN_i)}{\text{Total population}}$ <p>Where N is the number of distinct labels.</p>
Exact Match	$\frac{1}{N} \sum_{i=1}^N \frac{ \hat{y}_i \wedge y_i }{ \hat{y}_i \vee y_i }$ <p>Where "\wedge" and "\vee" are the logical "OR" and "AND" operations, and "\hat{y}_i" and "y_i" are the i-th predicted labels set and i-th the true labels set respectively.</p>
Hamming Loss	$\frac{1}{D} \sum_{i=1}^D \frac{\text{xor}(\hat{y}_i, y_i)}{ L }$ <p>Where: D: the number of samples, L: the number of labels, \hat{y}: the prediction, y: the ground truth.</p>
Micro-Recall	$\frac{\sum_{i=1}^N TP_i}{\sum_{i=1}^N (TP_i + FN_i)}$ <p>Where N is the number of distinct labels.</p>
Macro-Recall	$\sum_{i=1}^N \frac{TP_i}{TP_i + FN_i}$ <p>Where N is the number of distinct labels.</p>
Samples-Recall	$\sum_{i=1}^D \frac{TP_i}{TP_i + FN_i}$ <p>Where D is the number of samples.</p>
Micro-Precision	$\frac{\sum_{i=1}^N TP_i}{\sum_{i=1}^N (TP_i + FP_i)}$ <p>Where N is the number of distinct labels.</p>

Macro-Precision	$\sum_{i=1}^N \frac{TP_i}{TP_i + FP_i}$ <p>Where N is the number of distinct labels.</p>
Samples-Precision	$\sum_{i=1}^D \frac{TP_i}{TP_i + FN_i}$ <p>Where D is the number of samples.</p>
Micro-F1 Score	$2 * \frac{Precision_{micro} * Recall_{micro}}{Precision_{micro} + Recall_{micro}}$
Macro-F1 Score	$2 * \frac{Precision_{macro} * Recall_{macro}}{Precision_{macro} + Recall_{macro}}$
Samples-F1 Score	$2 * \frac{Precision_{samples} * Recall_{samples}}{Precision_{samples} + Recall_{samples}}$

2.6.2 Keyword search comparison

A classification function based on basic keyword search within the description was also built and verified on the test set. To define this function, a keywords list for each topical area was firstly defined as in Table 2.13.

Table 2.13 - Keyword list for each topical area.

Topical Area	Keyword list
Across Specialties	pharmacy, pharmacist, pharmacology, drug, medical, pharmaceuticals, prescription, telemedicine, hospital, child, baby, babies, kid, development.
Cardiology and Cardiovascular Medicine	electrocardiogr*, myocard*, heart, arrhythmia, tachycardia, cardiovascular, blood, cardiac, coronary, vessels, ecg, atri, ventricle, atrium, stroke, infarct, ictus, thrombosis, venous.
Dentistry	tooth, abscess, teeth, bridge, caries, dental, gingiva, molar, mouth, plaque, brace, dentin
Dermatology	skin, acne, dermatitis, dermatology, nail, dermatologist, sunburn, burns.
Emergency Medicine	emergency, 911, ambulance, first aid, resuscitation, rescue, defibrillator, dae, heart massage.

2.6 Training & Test sets

Fitness and Wellness	aerobics, coach, cycling, dancing, endurance, energy, exercise, fit, muscles, practice, relax, run, meditation, sport, stretching, team, train*, workout, yoga.
Gastroenterology, Hepatology, and Nephrology	diverticul, appendicitis, appendix, celiac, cirrhosis, colitis, colon, constipation, crohn's disease, diarrhea, digestive, endoscopy, endosonography, esophagus, fecal diversion, fistula, gas, intestine, nausea, pancreas, rectal, stomach, stoma, vomiting, stipsis, cholecystitis, colonoscopy.
Gynaecology and Obstetrics and Neonatal care	gynecology, amniocentesis, amniotic, contractions, embryo, epidural, fetal, fetus, gestation, nausea, pelvic floor, placenta, uterus, vagina, waters breaking, birth, labor, pregnancy, breastfeed, delivery, obstetric, obGyn, newborn.
Immunology & Endocrinology	glands, andropause, hormone, testosterone, cholesterol, endocrine, endocrinologist, estrogen, ovaries, pancreas, thyroid, hormonal, diabete, sugar, insulin, glucose, type 1, type 2, glucose tolerance test, ogtt, hba1c.
Mental Health, Neurology, Psychiatry	nerve, neurons, alzheimer, aphasia, cortex, brain, nervous, cerebral, dementia, electromyography, encephalitis, psychologist, neurologist, neurology, mental, electroencephalogram, psychology, sciatic, sciatica, psychiatrist, mental.
Nutrition	eat, calorie, carbohydrates, fat, nutrient, nutrition, food, protein, water, vitamins, cholesterol, sugar, salt, appetite, diet, bulimia, obesity, anorexia, calcium, dietary, dieting, fiber, omega 3, supplement.
Oncology	biopsy, cancer, carcinoma, carcinogen, chemotherapy, histology, mammogram, mastectomy, metastasis, oncologist, tumor.
Physiatry and Orthopedics	physiatrist, ankle, junction, cartilage, fracture, ligament, joint, articulation, femoral, bone, osteoporosis, skeletal, spinous, arthritis, vertebra, neck, back.

Chapter 2

Materials and Methods

Sensory Systems Healthcare	touch, hearing, view, sight, vision, eyesight, smell, olfaction, sensory, flavor, palate, taste buds, listen, hear, sound, sounds, optometry, colorblind, astigmatic, astigmatism, shortsighted, myopic, myope, nearsighted, eye, ear, tongue, deaf, hearing loss, hearing aid, speech, dyslexia, dyslexic, communication
Pneumology	sleep, respiratory, aerosol, airflow, breath, obstructive, polysomnography, pulmonary, spirometry, tracheostomy, somnolence, apnea, rem, ventilation
Surgery	plasty, tomy, incisions, surgery, operation, anesthesia, needle, surgeon,

Keywords are all lower case and they can be either word or prefixes and suffixes. The list shows also plurals variant for those keywords whose plural forms are not obtained by adding -s.

When the user searches for an app, a subset of these keywords is inserted in the search. For this comparative study, the entire set of keywords is inserted in the search. As a direct consequence, this search method is more powerful than the one a user can perform.

This algorithm was verified on the same test set used for the first method. To evaluate performance comparison with respect to the search based on CUIs, metrics explained in sub-section 2.6.1 were calculated.

Chapter 3

Results

Figure 3.1 shows an outline of results as they are presented in this chapter, following the same structure as in the previous chapter. First, the creation of the database is presented (Section 3.1), as obtained through Apps' links retrieval, Information Extraction, Information Cleaning, and Pre-processing. Then, the results of text analytics are shown (Section 3.2) in terms of MetaMap input, output, and methodological improvements. Finally, the results of classification on the training set, test set, and the whole database are presented (Section 3.3).

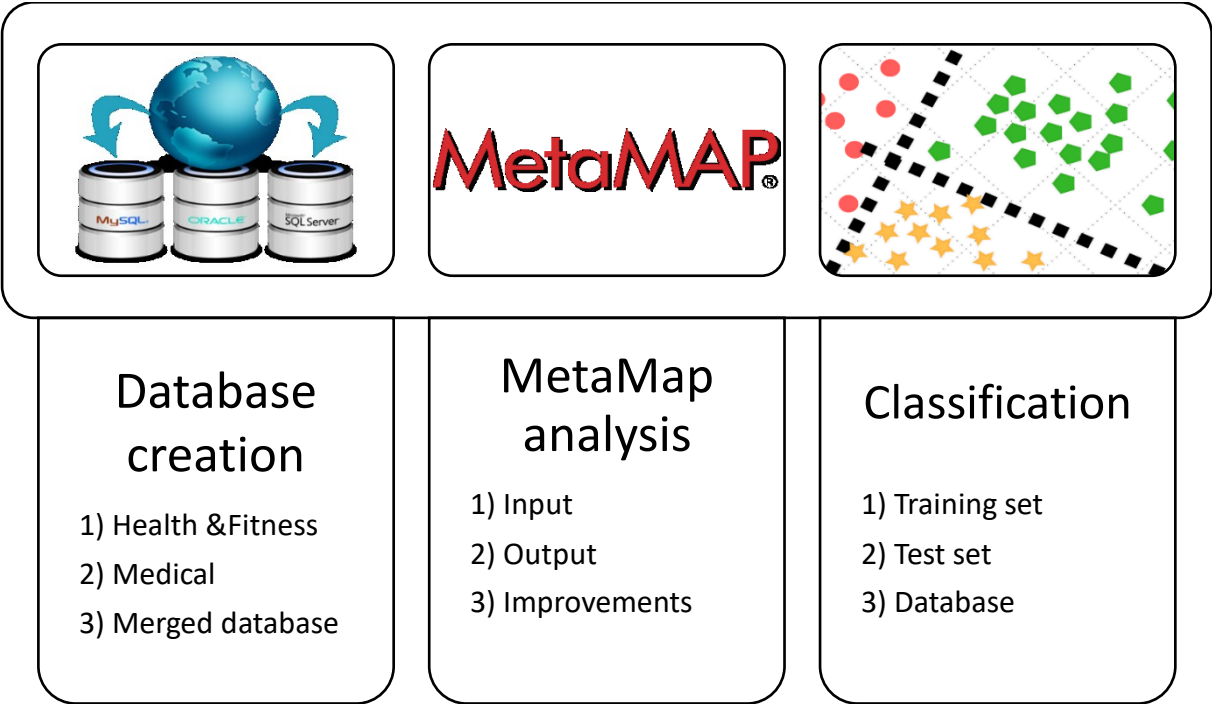


Figure 3.1 – Outline of results workflow

3.1 Database creation

A total of 79557 H&F and 42008 M apps' webpages were crawled on the US iTunes App Store. Some apps (i.e., 68 H&F and 37 M) had empty or very short description (below 14 characters); moreover, a not negligible number of apps were described in languages other than English (i.e., 18382 H&F and 11397 M). Details are reported in Table 3.1 and Table 3.2 for apps in the H&F and M category, respectively. Specifically, the two tables show, for each initial letter and symbol:

- The number of apps found on the Apple App Store
- The number of apps correctly retrieved from the Apple App Store
- The number of apps with description in English
- The number of apps with description in languages other than English
- The number of apps that were excluded because their description was shorter than 14 characters.

3.1 Database creation

H&F DATABASE

Table 3.1 – Creation of the H&F Database.

Letter	Number of apps retrieved from the App Store	Number of apps in English	Number of apps other than in English	Number of apps with description shorter than 14 characters
A	4602	3384	1213	5
B	5826	4743	1081	2
C	5456	4159	1292	5
D	3509	2566	941	2
E	2687	1958	729	0
F	4482	3161	1319	2
G	2808	2129	679	0
H	4050	3355	692	3
I	1743	1346	396	1
J	876	706	169	1
K	1461	1033	424	4
L	2590	1943	644	3
M	6742	5183	1550	9
N	2135	1623	510	2
O	1440	1054	386	0
P	5544	4045	1492	7
Q	439	364	75	0
R	2998	2445	553	0
S	7706	6020	1680	6
T	3394	2732	661	1
U	865	671	194	0
V	1714	1249	464	1
W	2422	2089	329	4
X	193	144	46	3
Y	1236	1071	164	1
Z	554	326	226	2
#	2085	1608	473	4
TOT.	79557	61107	18382	68

MEDICAL DATABASE

Table 3.2 – Creation of the M Database.

Letter	Number of apps retrieved from the App Store	Number of apps in English	Number of apps other than in English	Number of apps with description shorter than 14 characters
A	3338	2396	939	3
B	2210	1766	442	2
C	3621	2658	961	2
D	2536	1673	861	2
E	1800	1376	424	0
F	1451	916	535	0
G	1240	816	424	0
H	2079	1676	403	0
I	1245	937	308	0
J	369	290	79	0
K	642	375	267	0
L	1145	801	344	0
M	4048	3156	885	7
N	1454	1162	292	0
O	1152	853	299	0
P	3507	2443	1060	4
Q	233	178	55	0
R	1423	1072	350	1
S	3622	2611	1008	3
T	1537	1049	484	4
U	554	408	145	1
V	1185	860	323	2
W	696	587	108	1
X	134	78	56	0
Y	210	178	30	2
Z	203	69	134	0
#	374	190	181	3
TOT.	42008	30574	11397	37

For both M and H&F categories, the total number of apps retrieved from the store corresponds to the total number of apps available on the store.

As shown in Table 3.1 and Table 3.2, a not negligible number of apps on the US iTunes App Store were described in languages other than English, i.e. 18382 H&F (23.11%) and 11397 M (27.13%). Only a minor number of apps had a description shorter than 14 characters, i.e. 68 H&F (0.09%) and 37 M (0.09%). Results are summarized in Figure 3.2 and Figure 3.3.

3.1 Database creation

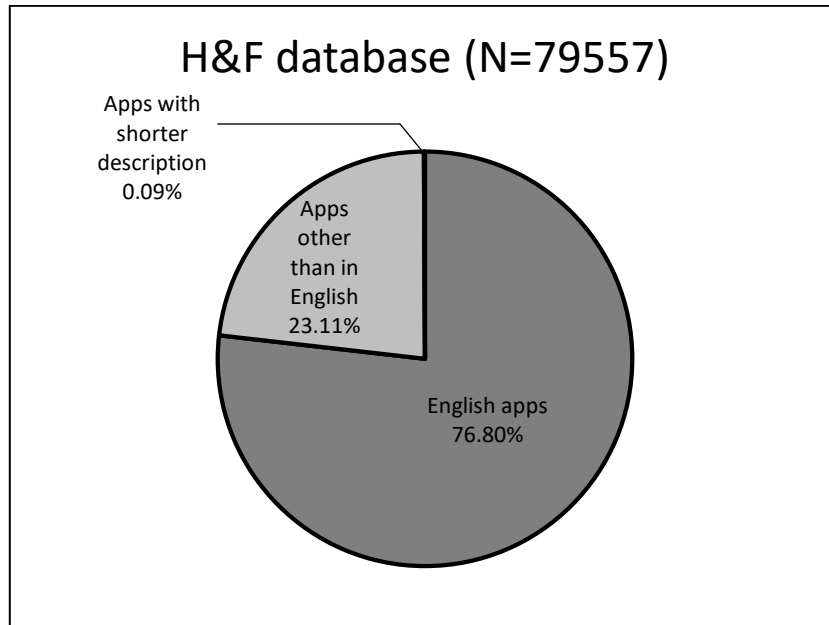


Figure 3.2 – H&F Database distribution.

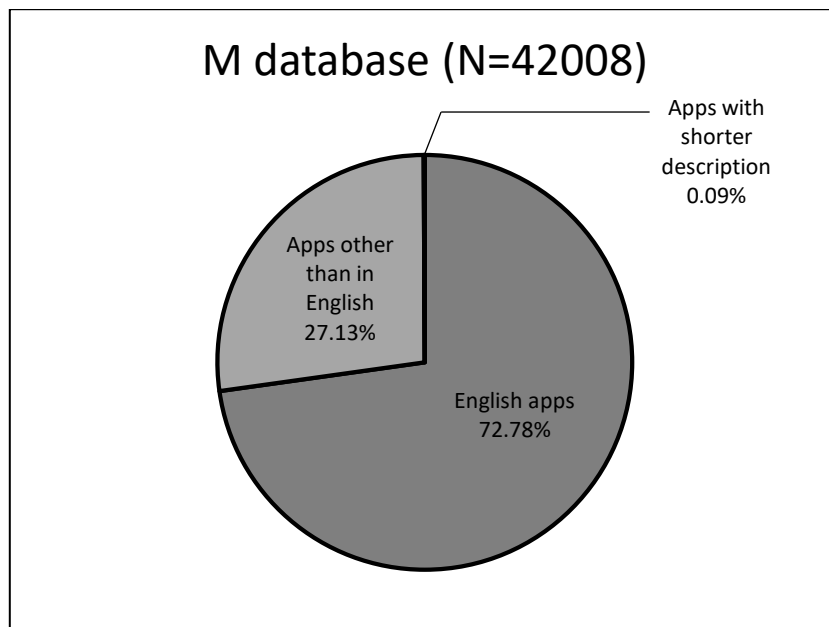


Figure 3.3 – M Database distribution.

After removing the apps described in languages other than English and those with description shorter than 14 characters, possible duplicates between the H&F and M databases were identified and removed to develop a merged database. As a result, a database of 80490 unique apps was obtained: 49925 (62.03%) that belonged to the H&F category, 19374 (24.07%) that belonged to the M category, and 11191 (13.90%) that belonged to both categories. Results are summarized in Table 3.3, Table 3.4 and Figure 3.4.

MERGED DATABASE

Table 3.3 – Development of the merged app database.

Letter	Number of apps retrieved from both M and H&F categories.	Number of apps in English	Number of distinct apps that are MetaMap analyzable	Number of duplicated apps
A	7940	5780	5067	713
B	8036	6509	5718	791
C	9077	6817	5939	878
D	6045	4239	3570	669
E	4487	3334	2913	421
F	5933	4077	3706	371
G	4048	2945	2613	332
H	6129	5031	4303	728
I	2988	2283	2022	261
J	1245	996	925	71
K	2103	1408	1259	149
L	3735	2744	2455	289
M	10790	8339	7037	1302
N	3589	2785	2452	333
O	2592	1907	1627	280
P	9051	6488	5595	893
Q	672	542	468	74
R	4421	3517	3171	346
S	11328	8631	7643	988
T	4931	3781	3400	381
U	1419	1079	964	115
V	2899	2109	1774	335
W	3118	2676	2417	259
X	327	222	209	13
Y	1446	1249	1139	110
Z	757	395	368	27
#	2459	1798	1736	62
TOT	121565	91680	80490	11190

3.2 MetaMap Analysis

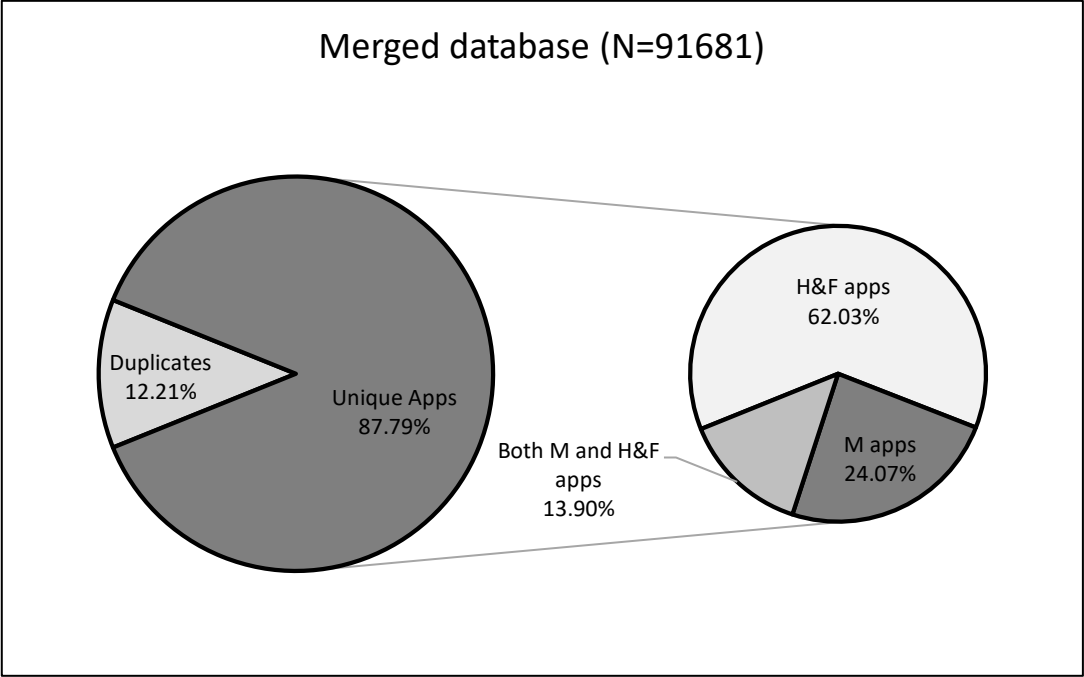


Figure 3.4 - Merged database distribution.

Table 3.4 – M and H&F distribution over the merged database.

M only	Percentage
19374	24.07%
H&F only	Percentage
49926	62.03%
Both M and H&F	Percentage
11190	13.90%

3.2 MetaMap Analysis

Once the MetaMap via batch upload was selected as MetaMap usage type, the app’s descriptions were extracted and formatted in a txt file to be successively analyzed, as described in Section 3.2.1. The following sub sections 3.2.1, 3.2.2 and 3.2.3 illustrate the process step by step.

3.2.1 Input

The txt input file for batch upload contained all the 80490 descriptions of apps extracted from the merged database. To be able to link each app's description with the output file resulting from MetaMap analysis, the txt file was formatted as follows:

id app #1 | app description #1

id app #2 | app description #2

....

id app #80490 | app description #80490.

3.2.2 Output

After 5835 seconds (i.e. 1 hour, 7 minutes and 15 seconds), MetaMap returned the results of the completed analysis. 80208 out of the 80490 apps descriptions were correctly analyzed whereas the remaining 282 produced processing errors. For each of these sets, a txt file has been produced.

The txt file from the 80208 apps was split to separate the output from each app in a way that each file contained the original id of the app and the resulting MetaMap analysis performed on its description.

The 282 apps that produced processing errors were rearranged in a new txt file to be uploaded again on MetaMap. For the analysis of this subset of apps, the timeout was incremented to 15 minutes per app to try to overcome problems related to longer analysis time. After this second cycle of MetaMap analysis, 215 out of the 282 apps were correctly analyzed. As for the first set of 80208 apps, the file obtained from these 215 apps was split to separate the output from each app.

As a result, 80423 out of 80490 (99.70%) apps were correctly analyzed by MetaMap.

The remaining 67 apps required a more detailed assessment to understand why MetaMap was not able to process them. This process is explained in subsection 3.2.3.

3.2 MetaMap Analysis

3.2.3 Improvements

By analyzing the log error produced by MetaMap after its analysis, it was found that the analysis of the 67 apps had reached the timeout. To understand the reasons why for these apps the timeout has been reached, a manual analysis of their descriptions was performed.

- After a first manual revision, sequences of ASCII symbols (e.g. "****") with no particular sense emerged.
- After the removal of these characters, another MetaMap analysis has been launched but produced the same errors.
- As a second revision, each app was manually analyzed word by word. MetaMap divides text into phrases thanks to punctuation: if punctuation is rare or it is not present at all, MetaMap considers as a phrase the entire portion of text without punctuation.

Figure 3.5 shows an example of app description with almost total absence of punctuation. This implies a bigger amount of time for the phrase to be interpreted, to find all the variants and to compute all the possible mappings. As a result, the time needed is much longer than the timeout set before.

```
Remote User: massimoschiavo interfile:
/usr/local/apache/htdocs/II/Scheduler/foo/inter_11042017_10:45:20_13199_massimoschiavo_633927611 text:
Perform essential day to day cardiology calculations more accurately and efficiently with the Cardiology Tool
application. Access over 25 calculators and tools using a variety of features: Quickly locate calculators using the
Search Save frequently used calculators with "Favorite" feature Reference last calculators accessed with "Recently
Viewed" featureCalculations Tools include: BP Percentiles for Children CHADS2 Score for Afib Stroke Risk CO2
Production Cardiac Output Cardiac Output MultiCalc Friedewald Equation for LDL Gorlin Formula for Valve Area
International Normalized Ratio (INR) Mean Arterial Pressure (MAP) Metabolic Syndrome Criteria (AHANHLBI
2005) Metabolic Syndrome Criteria (ATP III) O2 Consumption O2 Content of Arterial Blood O2 Content of Venous
Blood Oxygenation Index (OI) Pulmonary Vascular Resistance QT Interval Correction QsQt Right to Left Shunt
Fraction Respiratory Quotient Systemic Vascular Resistance Systolic Pressure Variation TIMI Score for NSTEMI
and UA TIMI Score for STEMI Very Low Density Lipoprotein (VLDL)Serve your patients better at the point of care
by referencing the Cardiology Tool for all of your cardiology related calculations. Please note that an Epocrates
account is required to use this free application. Registration is free and takes just a minute to complete.More than 1
million active members, including 50% of U.S. physicians, rely on Epocrates to enable better patient care by
delivering the right information, right when its needed.If you have questions regarding this application, please contact
us at supportepocrates.co
```

Figure 3.5 – Example of description without punctuation.

Figure 3.6 shows another example in which the text highlighted in blue does not contain any forms of punctuation. Even if the first part could be correctly analyzed without spending a lot of time, on the contrary the second part instead requires more than 15 minutes to be analyzed.

add on pack purchased. Individual sections to suit your needslikesrequirements including Baby, Household, Natures Animals, Natures Water, Weather and Chill Out.Free Version10 Absolutely free sounds and images selected from our full version categories to give you a feel for them all, to see which suits your own specific relaxing sounds before purchase that helps you drift a sleep . We hate ads and if you purchase the full version all are removed.Includes for free the relaxing colour visuals, specifically designed by us to provide a mood lighting sequence to help relax the brain and gently drift you off to sleep, however limited by number of colour visuals in the free versionTaster of the sounds availableOcean Waves Ocean Curls Rainfall Tropical Rainfall Thunderstorm Bamboo Water Zen Garden Monkey Business Noisy Crickets Insects Cat Purring Whales Birds Frogs Night Owl Light Breeze Strong Wind Hurricane Thunderstorm Tropical Rainfall Fire Wind Chime Bamboo Wind Chime Ceramic Ice Lawn Mower Fan Ceiling Fan Hoover Washing Machine Grandfather Clock Hair Dryer Boiling Water Radio Static Nursery Rhyme Hoover Classical Music Heartbeat Stirring Coffee Birds Singing Elastic Bands Ticking Clock Electric Guitar The Sax Violin Mouth Organ Smooth Drums Glass Bottles Whistles Under Water

Figure 3.6 – Example of description with a part without punctuation.

Finally, this set of apps was prepared for another MetaMap analysis by increasing the timeout at the maximum allowed, i.e. 45 minutes per phrase. As a result, 42 out of 67 apps were correctly analyzed.

Table 3.5 summarizes the number of apps that were correctly analyzed at each iteration with MetaMap and those which turned out errors.

Table 3.5 – MetaMap recap.

Iteration	Descriptions correctly analyzed	Descriptions that returned processing errors
1	80208	282
2	215	67
3	42	25
TOT.	3	80465

For the remaining 25 apps, it was found that MetaMap was not able to analyze them because the descriptions were too long and, as a result, blank txt files were produced.

3.3 Classification

3.3 Classification

This section will illustrate all the results obtained after the automated classification performed on 400 apps in the training set (Section 3.3.1), the 400 apps in the test set (Section 3.3.2) and the 80490 apps in the whole database (Section 3.3.4).

3.3.1 Training set

Table 3.6 shows the distribution of M and H&F apps over the training set: it is possible to note that these percentages parallel those shown in Table 3.4 for the merged database.

Table 3.6 - M and H&F apps distribution over the training set (N=400).

M only	Percentage
99	24.75%
H&F only	Percentage
246	61.50%
Both M and H&F	Percentage
55	13.75%

Figure 3.7 shows the distribution of topical areas over the training set.

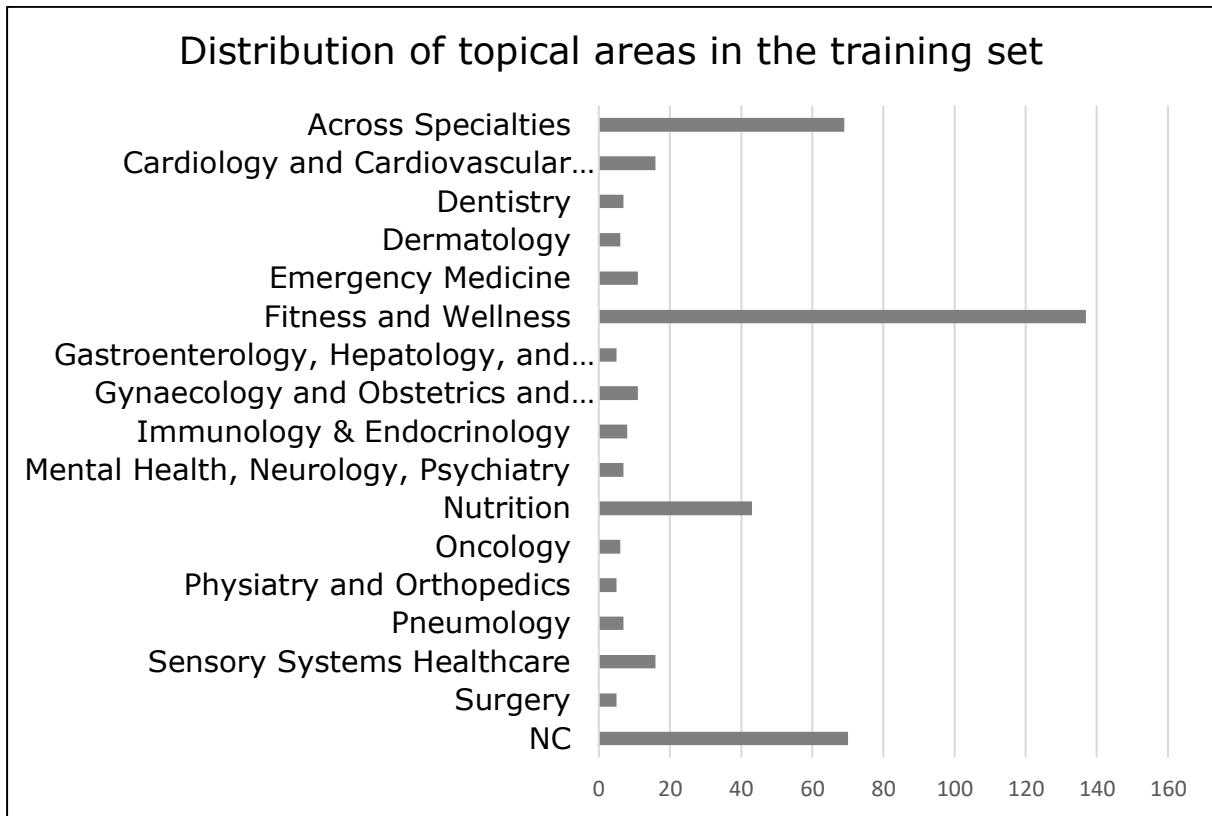


Figure 3.7 - Distribution of topical areas over the training set.

To compute the performance evaluation metrics it was necessary, as a first step, to build the confusion matrix. *Table 3.7* shows the confusion matrix for each topical area obtained following automated classification of apps in the training set, as compared to manual classification.

3.3 Classification

Table 3.7 – Training set confusion matrix.

Topical Area	True Positives (TP)	False Positives (FP)	False Negatives (FN)	True Negatives (TN)
Across Specialties	43	21	26	310
Cardiology and Cardiovascular Medicine	13	34	3	350
Dentistry	7	9	384	0
Dermatology	5	3	1	391
Emergency Medicine	9	8	2	381
Fitness and Wellness	116	37	21	226
Gastroenterology, Hepatology, and Nephrology	3	4	2	391
Gynaecology and Obstetrics and Neonatal care	10	13	1	376
Immunology & Endocrinology	8	3	0	389
Mental Health, Neurology, Psychiatry	7	42	0	351
Nutrition	41	40	2	317
Oncology	6	4	0	390
Physiatry and Orthopedics	5	3	0	392
Pneumology	7	16	0	377
Sensory Systems Healthcare	15	19	1	365
Surgery	5	19	0	376
NC	46	24	24	306

By considering the original multi-label classification problem as a binary classification problem, for each label in each topical area, it was possible to compute Accuracy, Precision and Recall for each of the topical areas.

Since Precision and Recall represent the ability of the classifier to find all the positive samples and not to label as positive a sample that is negative, the computation of these indices for each topical area is useful to better direct future improvements.

Table 3.8 shows Accuracy, Precision and Recall computed on the training set for each of the topical areas.

Table 3.8 - Accuracy, Precision, and Recall computed for each topical area on the training set.

Topical Area	Accuracy	Precision	Recall
Across Specialties	88.25%	67.19%	62.32%
Cardiology	90.75%	27.66%	81.25%
Dentistry	97.75%	43.75%	100.00%
Dermatology	99.00%	62.50%	83.33%
Emergency Medicine	97.50%	52.94%	81.82%
Fitness and Wellness	85.50%	75.82%	84.67%
Gastroenterology	98.50%	42.86%	60.00%
Gynaecology and Obstetrics and Neonatal care	96.50%	43.48%	90.91%
Immunology & Endocrinology	99.25%	72.73%	100.00%
Mental Health, Neurology, Psychiatry	89.50%	14.29%	100.00%
Nutrition	89.50%	50.62%	95.35%
Oncology	99.00%	60.00%	100.00%
Physiatry and Orthopedics	99.25%	62.50%	100.00%
Pneumology	96.00%	30.43%	100.00%
Sensory Systems Healthcare	95.00%	44.12%	93.75%
Surgery	95.25%	20.83%	100.00%
NC	88.00%	65.71%	65.71%

Table 3.9 shows the mean value, the median and the best value for all these three metrics, along all topical areas.

Table 3.9 - Mean, Median, and Best value for Accuracy, Precision, and Recall in the training set.

Metric	Mean value	Median	Best value
Accuracy	94.38%	96.00%	99.25%
Precision	49.26%	44.12%	75.82%
Recall	88.18%	93.75%	100.00%

3.3 Classification

For a comprehensive performance evaluation of the method, the metrics reported in Table 2.12 were computed, and corresponding results shown in Table 3.10.

Table 3.10 – Metrics computed on the training set.

Metric	Percentage
General Accuracy	94.38%
Exact match	49.00%
Hamming loss	5.62%
Micro-Recall	80.65%
Macro-Recall	88.18%
Samples-Recall	81.25%
Micro-Precision	53.64%
Macro-Precision	49.26%
Samples-Precision	64.74%
Micro-F1 Score	64.43%
Macro-F1 Score	60.32%
Samples-F1 Score	69.24%

The classifier was satisfactorily able to identify the topical area(s) to which an app belongs (General Accuracy = 94.38%, Hamming loss = 5.62% = 100 - 94.38%). The Exact Match metric is lower (49.00%) as it relates the classifier's ability to classify the whole set of predicted labels for a sample with exactly the corresponding set of true labels, which is more difficult to achieve over the whole set of labels. The ability to not label as positive a negative sample is fairly good (Macro-Precision = 49.26%). The classifier was able to find all the positive samples in each topical area well enough (Macro-Recall = 88.18%). F1 score can be interpreted as a weighted average of the precision and recall, where an F1 score reaches its best value at 100% and worst score at 0%. The relative contribution of precision and recall to the F1 score are equal (i.e. 60.32%).

3.3.2 Test set

Table 3.11 shows the distribution of M and H&F apps over the test set: it's noticeable as these percentages parallel those shown in Table 3.4 for the merged database.

Table 3.11 - M and H&F apps distribution over the test set (N=400).

M only	Percentage
100	25.00%
H&F only	Percentage
246	61.50%
Both M and H&F	Percentage
54	13.50%

Figure 3.8 shows the distribution of topical areas over the test set.

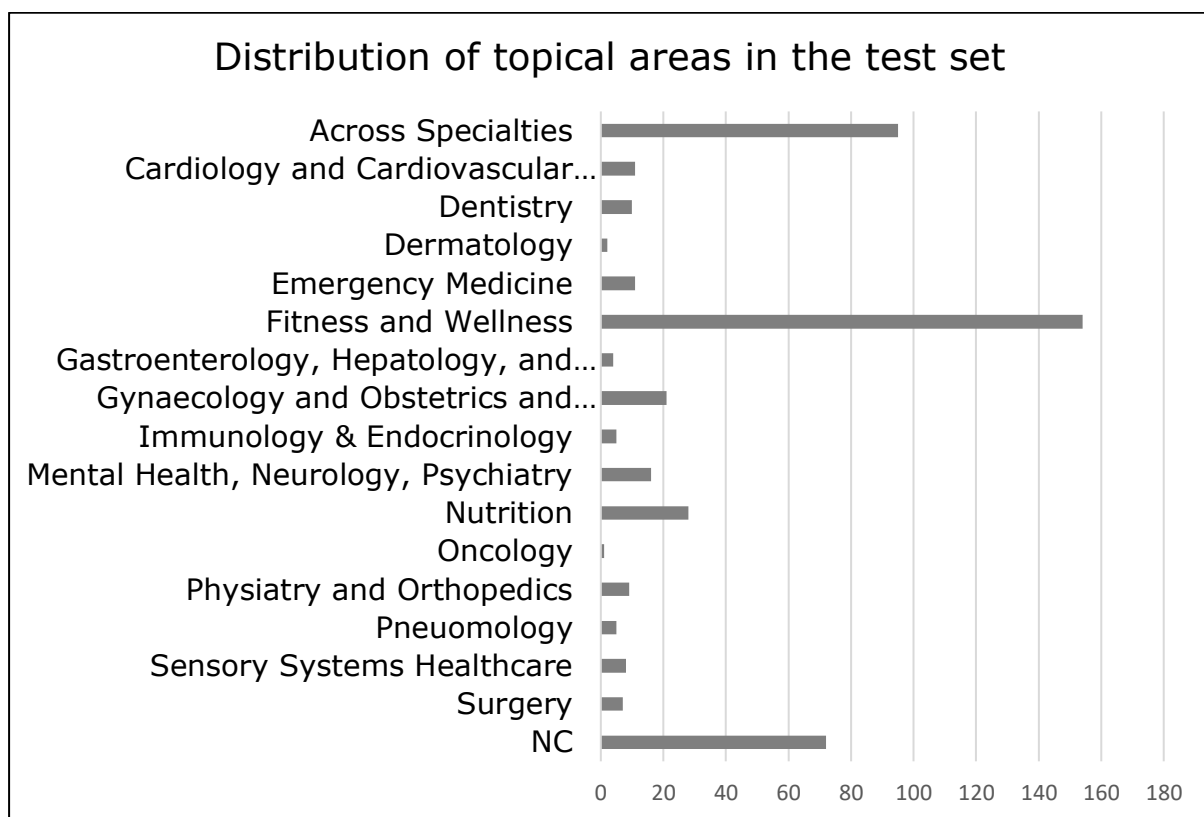


Figure 3.8 - Distribution of topical areas over the test set.

To compute indices of performance evaluation metrics it was necessary, as a first step, to build the confusion matrix. Table 3.12 shows the confusion matrix obtained following automated classification of apps in the test set, as compared to manual classification.

3.3 Classification

Table 3.12 - Test set confusion matrix.

Topical Area	True Positives (TP)	False Positives (FP)	False Negatives (FN)	True Negatives (TN)
Across Specialties	49	24	46	281
Cardiology and Cardiovascular Medicine	9	53	2	336
Dentistry	7	4	386	3
Dermatology	2	16	0	382
Emergency Medicine	10	7	1	382
Fitness and Wellness	104	26	50	220
Gastroenterology, Hepatology, and Nephrology	0	3	4	393
Gynaecology and Obstetrics and Neonatal care	14	15	7	364
Immunology & Endocrinology	3	5	2	390
Mental Health, Neurology, Psychiatry and Neurology	9	33	7	351
Nutrition	24	50	4	322
Oncology	1	7	0	392
Physiatry and Orthopedics	1	4	8	387
Pneumology	3	10	2	385
Sensory Systems Healthcare	4	23	4	369
Surgery	4	23	3	370
NC	43	50	29	278

For a comprehensive performance evaluation of the method, the metrics reported in Table 2.12 were computed. Results are shown in Table 3.13.

Table 3.13 – Metrics computed on the test set.

Metric	Percentage
General Accuracy	92.28%
Exact match	36.00%
Hamming loss	7.72%
Micro-Recall	62.53%
Macro-Recall	62.85%
Samples-Recall	63.50%
Micro-Precision	44.84%
Macro-Precision	33.31%
Samples-Precision	51.51%
Micro-F1 Score	52.23%
Macro-F1 Score	39.00%
Samples-F1 Score	54.50%

To assess the differences in performance between automated classification of the training set and of the test set, the differences for each metric and the difference in distribution of topical areas were computed. Results are shown in Table 3.14 and Table 3.15. Positives values indicate improvements in the test set compared to the training set whereas negative values indicate worsening.

Table 3.14 – Differences in metrics between the training set and the test set.

Metric	Percentage
General Accuracy	-2,10%
Exact match	-13,00%
Hamming loss	+2,10%
Micro-Recall	-18,12%
Macro-Recall	-25,33%
Samples-Recall	-17,75%
Micro-Precision	-8,80%
Macro-Precision	-15,95%
Samples-Precision	-13,23%
Micro-F1 Score	-12,20%
Macro-F1 Score	-21,32%
Samples-F1 Score	-14,74%

3.3 Classification

Table 3.15 - Differences in distribution between the training set and the test set.

Topical Area	Percentage
Across Specialties	+4.61%
Cardiology and Cardiovascular Medicine	-1.33%
Dentistry	+0.55%
Dermatology	-0.96%
Emergency Medicine	-0.17%
Fitness and Wellness	+1.62%
Gastroenterology, Hepatology, and Nephrology	-0.29%
Gynaecology and Obstetrics and Neonatal care	+2.01%
Immunology & Endocrinology	-0.78%
Mental Health, Neurology, Psychiatry	+1.85%
Nutrition	-3.92%
Oncology	-1.18%
Physiatry and Orthopedics	+0.80%
Sensory Systems Healthcare	-1.99%
Pneumology	-0.54%
Surgery	+0.36%
NC	-0.63%

As expected, all the values in Table 3.14 showed a decrease in performance. However, even if indices decreased, the General Accuracy remained high (92.28%). From the differences in distribution of topical areas over the training and the test set (Table 3.15) it can be appreciated that the sets could be considered with more or less the same distribution of topical areas.

Table 3.16 and Table 3.17 show the binary metrics computed considering each topical area as a single binary classification problem. In detail, for each topical area, indices computed were: Accuracy (ACC), Precision (PPV), False Discovery Rate (FDR), False Omission Rate (FOR), Negative Predictive Value (NPV), Recall (TPR), False Positive Rate (FPR), False Negative Rate (FNR), True Negative Rate (TNR), Positive Likelihood Ratio (LR+), Negative Likelihood Ratio (LR-), Diagnostic Odds Ratio (DOR), F1 score.

Table 3.16 – Binary metrics computed for each topical area on the training set.

Subject Area	ACC	PPV	FDR	FOR	NPV	TNR	FPR	FNR	TNR	LR+	LR-	DOR	F1
Across Specialties	88.25%	67.19%	32.81%	7.74%	92.26%	62.32%	6.34%	37.68%	93.66%	9.82	0.40	24.41	64.66%
Cardiology	90.75%	27.66%	72.34%	0.85%	99.15%	81.25%	8.85%	18.75%	91.15%	9.18	0.21	44.61	41.27%
Dentistry	97.75%	43.75%	56.25%	0.00%	100.00%	100.00%	2.29%	0.00%	97.71%	43.67	0.00	N/A	60.87%
Dermatology	99.00%	62.50%	37.50%	0.26%	99.74%	83.33%	0.76%	16.67%	99.24%	109.44	0.17	651.67	71.43%
Emergency Medicine	97.50%	52.94%	47.06%	0.52%	99.48%	81.82%	2.06%	18.18%	97.94%	39.78	0.19	214.31	64.29%
Fitness and Wellness	85.50%	75.82%	24.18%	8.50%	91.50%	84.67%	14.07%	15.33%	85.93%	6.02	0.18	33.74	80.00%
Gastroenterology	98.50%	42.86%	57.14%	0.51%	99.49%	60.00%	1.01%	40.00%	98.99%	59.25	0.40	146.63	50.00%
Gynaecology and Obstetrics and Neonatal care	96.50%	43.48%	56.52%	0.27%	99.73%	90.91%	3.34%	9.09%	96.66%	27.20	0.09	289.23	58.82%
Immunology & Endocrinology	99.25%	72.73%	27.27%	0.00%	100.00%	100.00%	0.77%	0.00%	99.23%	130.67	0.00	N/A	84.21%
Mental Health, Neurology, Psychiatry	89.50%	14.29%	85.71%	0.00%	100.00%	100.00%	10.69%	0.00%	89.31%	9.36	0.00	N/A	25.00%
Nutrition	89.50%	50.62%	49.38%	0.63%	99.37%	95.35%	11.20%	4.65%	88.80%	8.51	0.05	162.46	66.13%
Oncology	99.00%	60.00%	40.00%	0.00%	100.00%	100.00%	1.02%	0.00%	98.98%	98.50	0.00	N/A	75.00%
Physiatry and Orthopedics	99.25%	62.50%	37.50%	0.00%	100.00%	100.00%	0.76%	0.00%	99.24%	131.67	0.00	N/A	76.92%
Sensory Systems Healthcare	95.00%	44.12%	55.88%	0.27%	99.73%	93.75%	4.95%	6.25%	95.05%	18.95	0.07	288.16	60.00%
Pneumology	96.00%	30.43%	69.57%	0.00%	100.00%	100.00%	4.07%	0.00%	95.93%	24.56	0.00	N/A	46.67%
Surgery	95.25%	20.83%	79.17%	0.00%	100.00%	100.00%	4.81%	0.00%	95.19%	20.79	0.00	N/A	34.48%
NC	88.00%	65.71%	34.29%	7.27%	92.73%	65.71%	7.27%	34.29%	92.73%	9.04	0.37	24.44	65.71%

3.3 Classification

Table 3.17 - Binary metrics computed for each topical area on the test set.

Subject Area	ACC	PPV	FDR	FOR	NPV	TPR	FPR	FNR	TNR	LR+	LR-	DOR	F1
Across Specialties	82.50%	67.12%	32.88%	14.07%	85.93%	51.58%	7.87%	48.42%	92.13%	6.55	0.53	12.47	58.33%
Cardiology and Cardiovascular Medicine	86.25%	14.52%	85.48%	0.59%	99.41%	81.82%	13.62%	18.18%	86.38%	6.01	0.21	28.53	24.66%
Dentistry	98.25%	63.64%	36.36%	0.77%	99.23%	70.00%	1.03%	30.00%	98.97%	68.25	0.30	225.17	66.67%
Dermatology	96.00%	11.11%	88.89%	0.00%	100.00%	100.00%	4.02%	0.00%	95.98%	24.88	0.00	N/A	20.00%
Emergency Medicine	98.00%	58.82%	41.18%	0.26%	99.74%	90.91%	1.80%	9.09%	98.20%	50.52	0.09	545.71	71.43%
Fitness and Wellness	81.00%	80.00%	20.00%	18.52%	81.48%	67.53%	10.57%	32.47%	89.43%	6.39	0.36	17.60	73.24%
Gastroenterology, Hepatology, and Nephrology	98.25%	0.00%	100.00%	1.01%	98.99%	0.00%	0.76%	100.00%	99.24%	0.00	1.01	0.00	0.00%
Gynaecology and Obstetrics and Neonatal care	94.50%	48.28%	51.72%	1.89%	98.11%	66.67%	3.96%	33.33%	96.04%	16.84	0.35	48.53	56.00%
Immunology & Endocrinology	98.25%	37.50%	62.50%	0.51%	99.49%	60.00%	1.27%	40.00%	98.73%	47.40	0.41	117.00	46.15%
Mental Health, Neurology, Psychiatry	90.00%	21.43%	78.57%	1.96%	98.04%	56.25%	8.59%	43.75%	91.41%	6.55	0.48	13.68	31.03%
Nutrition	86.50%	32.43%	67.57%	1.23%	98.77%	85.71%	13.44%	14.29%	86.56%	6.38	0.17	38.64	47.06%
Oncology	98.25%	12.50%	87.50%	0.00%	100.00%	100.00%	1.75%	0.00%	98.25%	57.00	0.00	N/A	22.22%
Physiatry and Orthopedics	97.00%	20.00%	80.00%	2.03%	97.97%	11.11%	1.02%	88.89%	98.98%	10.86	0.90	12.09	14.29%
Sensory Systems Healthcare	93.25%	14.81%	85.19%	1.07%	98.93%	50.00%	5.87%	50.00%	94.13%	8.52	0.53	16.04	22.86%
Pneumology	97.00%	23.08%	76.92%	0.52%	99.48%	60.00%	2.53%	40.00%	97.47%	23.70	0.41	57.75	33.33%
Surgery	93.50%	14.81%	85.19%	0.80%	99.20%	57.14%	5.85%	42.86%	94.15%	9.76	0.46	21.45	23.53%
NC	80.25%	46.24%	53.76%	9.45%	90.55%	59.72%	15.24%	40.28%	84.76%	3.92	0.48	8.24	52.12%

3.3.3 Comparison with keyword search

The same metrics as in Table 2.12 were computed on the training set and test set to assess classification performance of a basic keyword search classification, as described in section 2.6.4. Table 3.19 shows the metrics computed for keywords search classification on the test set.

Table 3.18 - Metrics computed on the training set with keywords search.

Metric	Percentage
General Accuracy	44.75%
Exact match	28.75%
Hamming loss	9.15%
Micro-Recall	47.09%
Macro-Recall	45.70%
Samples-Recall	47.00%
Micro-Precision	33.84%
Macro-Precision	41.85%
Samples-Precision	40.37%
Micro-F1 Score	39.37%
Macro-F1 Score	37.08%
Samples-F1 Score	41.33%

Table 3.19 - Metrics computed on the test set with keywords search.

Metric	Percentage
General Accuracy	39.25%
Exact match	27.75%
Hamming loss	9.03%
Micro-Recall	49.56%
Macro-Recall	49.38%
Samples-Recall	50.13%
Micro-Precision	36.89%
Macro-Precision	32.82%
Samples-Precision	44.54%
Micro-F1 Score	42.29%
Macro-F1 Score	35.09%
Samples-F1 Score	44.71%

To assess the differences in performance between classification based on CUIs and classification based on simply keyword search of the training set and of the test set, the differences for each metric were computed, and results shown in Table 3.14 and Table 3.15. Positive values indicate improvements in classification based on CUIs compared to classification based on keyword search whereas negative values indicate instead worsening.

3.3 Classification

Table 3.20 – Metrics comparison on the training set.

Metric	Percentage
General Accuracy	+49.63%
Exact match	+20.25%
Hamming loss	-3.53%
Micro-Recall	+33.56%
Macro-Recall	+42.48%
Samples-Recall	+34.25%
Micro-Precision	+19.80%
Macro-Precision	+7.41%
Samples-Precision	+24.37%
Micro-F1 Score	+25.06%
Macro-F1 Score	+23.24%
Samples-F1 Score	+27.91%

Table 3.21 – Metrics comparison on the test set.

Metric	Percentage
General Accuracy	+53.03%
Exact match	+8.25%
Hamming loss	-1.31%
Micro-Recall	+12.97%
Macro-Recall	+13.47%
Samples-Recall	+13.37%
Micro-Precision	+7.95%
Macro-Precision	+0.49%
Samples-Precision	+6.97%
Micro-F1 Score	+9.94%
Macro-F1 Score	+3.91%
Samples-F1 Score	+9.79%

It is clear how using CUIs resulted in high improvement in all metrics compared to using keywords, even the full set as used here, both for the training and test sets.

3.3.4 Database classification

As a final result, Figure 3.9 and Table 3.22 show the distribution over the whole app database for topical areas after the classification function was performed.

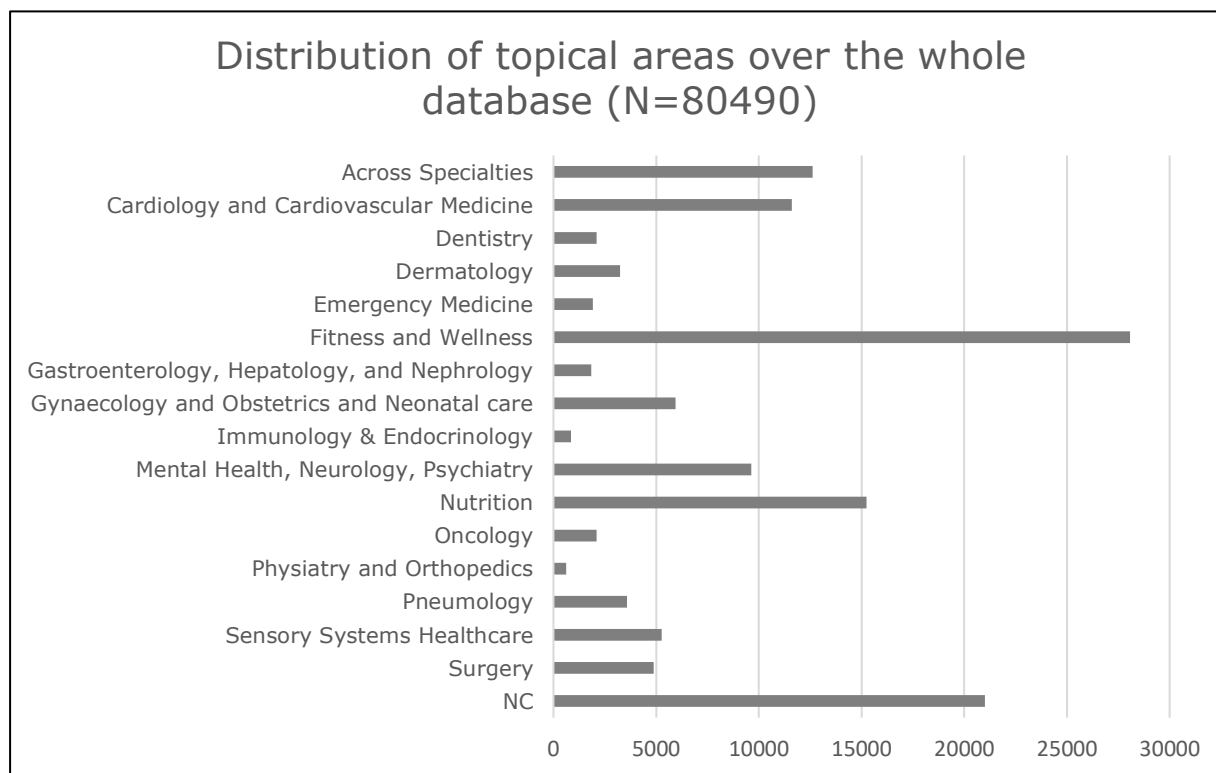


Figure 3.9 - Distribution of topical areas over the whole database.

Table 3.22 - Distribution of topical areas over the whole database (percentages, N=80490).

Topical Area	Percentage
Across Specialties	9.66%
Cardiology and Cardiovascular Medicine	8.90%
Dentistry	1.61%
Dermatology	2.49%
Emergency Medicine	1.47%
Fitness and Wellness	21.51%
Gastroenterology, Hepatology, and Nephrology	1.42%
Gynaecology and Obstetrics and Neonatal care	4.56%
Immunology & Endocrinology	0.65%
Mental Health, Neurology, Psychiatry	7.37%
Nutrition	11.67%
Oncology	1.61%
Physiatry and Orthopedics	0.47%
Sensory Systems Healthcare	4.04%
Pneumology	2.75%
Surgery	3.74%
NC	16.08%

Considering apps classified to a single area, a predominance of Fitness and Wellness apps is noticeable, followed by apps related to Nutrition, Cardiology and Cardiovascular Medicine, and Mental Health, Neurology and Psychiatry.

All other areas have % less than 5%, where some are practically absent (<1%, Immunology & Endocrinology, and Physiatry and Orthopedics).

This distribution highlights the fields of medicine in which the use of digital tools is more mature for exploitation, compared to those in which digital applications are not currently part of the resources available to patients and caregivers.

Chapter 4

Discussions and conclusions

In this study, the basic modules of an automated method, based on text analytics, were developed to classify mobile apps of possible medical relevance among specific subject areas by analyzing the information reported on the App Store webpages as explained in the Chapter 2.

Figure 4.1 shows an outline of discussions as they are presented in this chapter, following the same structure as in the previous chapter. First, information and numerical results retrieved during the development of the database are discussed. Then, results concerning text analytics will be considered, followed by discussion about classification results and future improvements.

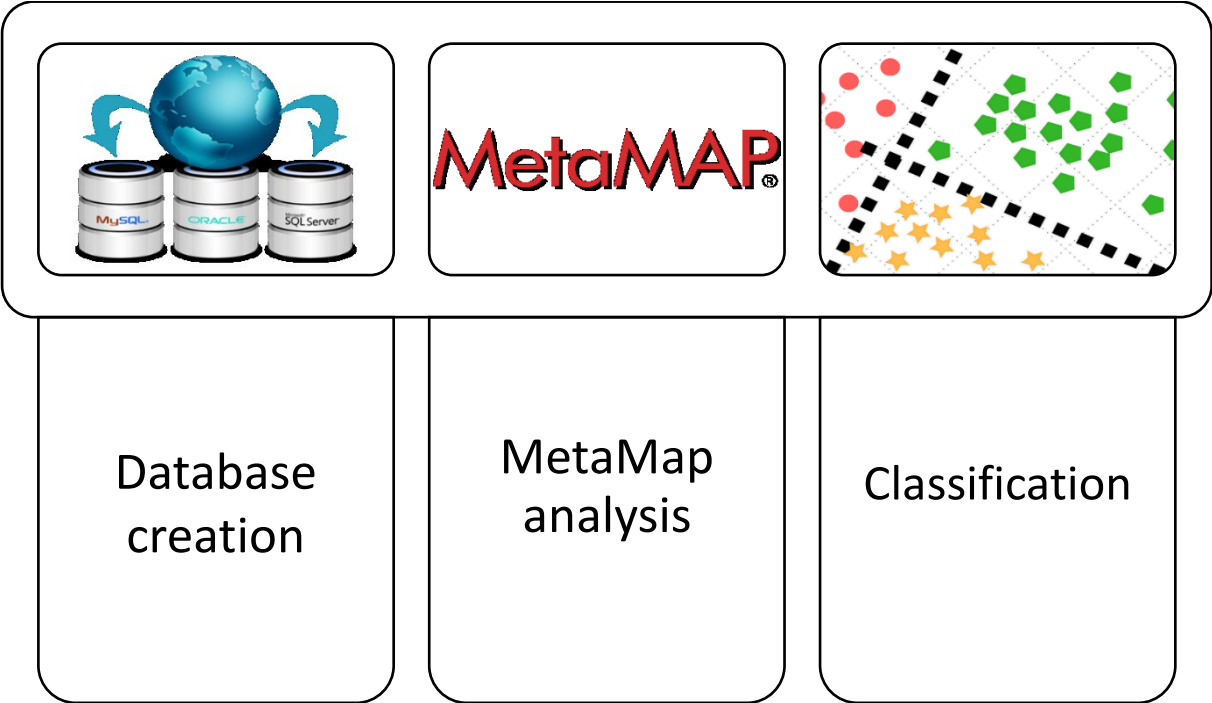


Figure 4.1 - Outline of discussions workflow.

4.1 Database creation

The market for mobile health (mHealth) has been growing over the last years and continues to do so. As of June 2018, there are more than 318500 mHealth apps in the market. Among all the stores, two of the major stores providing apps for users are the Apple App Store and the Google Play Store. This work is focused on the US iTunes App Store due to the quality of apps described in Chapter 1.

The choice of language is based instead on the target of reachable users, since English is the international language developers try to translate their apps in English more than into other languages.

In addition, there are slightly more apps available in the US iTunes App Store than in any of the other countries with the most established Internet markets (China, Japan, Brazil, Russia) [Xu & Liu, 2015], as it possible to note in Table 4.1 (information is highlighted in bold).

Google Play Store offers a little apps more than iTunes App Store but, for the reasons of quality explained above, this work was focused on the iTunes App Store.

Table 4.1 - The number of apps in different stores and regions at June 2018.

Store_Region_Category	The total number of apps in each specified combination of store, region, and category.
AppStore_BR_Health&Fitness	79513
AppStore_BR_Medical	39136
AppStore_CN_Health&Fitness	78620
AppStore_CN_Medical	38329
AppStore_JP_Health&Fitness	79360
AppStore_JP_Medical	38775
AppStore_RU_Health&Fitness	79161
AppStore_RU_Medical	38658
AppStore_US_Health&Fitness	81153
AppStore_US_Medical	40412
GooglePlay_US_Health&Fitness	98681
GooglePlay_US_Medical	43258

4.2 MetaMap analysis

As shown in Table 3.3, a total of 121565 apps' webpages related to mobile health were crawled on the US iTunes App Store. In this subset of apps, a not negligible number of apps that had to be removed because of a language other than in English were included. Also, duplication of apps present twice in the US iTunes Store both in H&F and M category was observed.

Finally, as reported in Table 3.5, 80490 was the effective number of apps medically relevant available on the US iTunes Store.

The availability of an automated method to crawl the US iTunes App Store is important because the analysis of information regarding mHealth apps can provide insights for future mHealth research developments. [Xu & Liu, 2015]

4.2 MetaMap analysis

Analyzing the information retrieved from the webpages with a natural language processing is necessary firstly to ensure that all the apps included in the database are health-related. [Xu & Liu, 2015]

Second, to understand the topical areas of interest is necessary to extract the Metathesaurus concepts referred to from the app's description.

For these reason, MetaMap was the selected tool in this work since it is able to map biomedical text to the Metathesaurus, by using a knowledge intensive approach based on symbolic, natural language processing (NLP) and computational linguistic techniques. [Aronson, 2001]

Accurate concept identification is crucial to biomedical natural language processing. However, ambiguity is common during the process of mapping terms to biomedical concepts (one term can be mapped to several concepts). A cost-effective approach to disambiguation relating to training is via semantic classification of the ambiguous terms, provided that the semantic classes of the concepts are available and are all different. Each concept in the UMLS contains a set of synonyms and is associated with Semantic Type(s), which are categorical semantic annotations assigned by human experts. However, MetaMap frequently provides more than one concept (sense) to each term it maps, resulting from the

fact that many terms have more than one meaning and thus are ambiguous. [Fan & Friedman, 2008]

This was a limitation when a concept had multiple interpretations, and thus assigned to different Semantic Types, and the multiple interpretations were related to different topical areas. For example, the term "Listen" could be mapped to the simple act of listening to the music and it could be mapped as the ability to hear something too.

Results concerning MetaMap analysis are excellent since 80465 out of the 80490 apps descriptions were correctly analyzed (99.97%). In addition, time spent in analysis with batch upload was really small: 5835 seconds to analyze 80490 apps (i.e. 0.072 sec/app's description).

Furthermore, the user-side resources were free during the analysis. Having chosen the batch upload method as MetaMap usage method, the whole analysis was remotely performed on the servers.

4.3 Classification

This section will explain all the results obtained concerning classification. In detail, discussions about classification based on text analytics (section 4.3.1), on the comparison performed between the two methods (i.e. Classifier based on text analytics and classifier based on keywords search)(section 4.3.2), and about the distribution of the subject areas in the whole database (section 4.3.3) will be discussed.

4.3.1 Classification based on text analytics

The key concept in the classification based on text analytics is to analyze the text to extract concepts related to the biomedical field in order to understand the origin of the app with respect to the topical areas. Once the biomedical concepts were identified in the app's description, it was necessary to understand those that were more relevant than the others, as described in the section 2.5 of the second chapter. Afterwards, to each app a set of labels identifying the topical areas was assigned.

4.3 Classification

To assess the performance of the developed method, it was necessary to build both the training set and test set and manually classify these sets among the topical areas. The distribution of the topical areas over these sets showed what is called label imbalance. For both the sets there are topical areas with much more samples than the others (i.e. "Across Specialties", "Fitness and Wellness", and "NC"). This has to be taken into consideration for future improvements, as to define an algorithm that has to be trained automatically onto set, label imbalance may cause some challenges and problems. [Murphey et al, 2004]

In this work the classification function does not come from an automatic appendment on the training set, as for example neural network does, and thus label imbalance is not a problem.

Both the training and test set were built to reflect the same distribution of M and H&F observed in the database, so they recreated the same small-scale environment. This was necessary to guarantee a complete visual of the apps to analyze. If the sets were built using apps belonging only from one of the categories, the classification function will surely fail for the other category. In the M category there are no apps related to fitness as well as in the H&F category there are no apps related to oncology, for example: to identify apps coming from all the defined topical areas, it was important to extract training and test sets that were representative samples of the whole database.

Results concerning classification based on text analytics are encouraging since the average accuracy was 94.38% for the training set and 92.28% for the test set. These values demonstrate that the classifier was able to identify the topical areas to whom an app belongs in a satisfactory way, where the loss of performance in the test set in respect to the training set was negligible (-2.10%).

To further improve the average accuracy, the topical areas with lower accuracy will need to be addressed: specifically, the "Fitness and Wellness" resulted in the lower accuracy (85.50% in the training set, 81.00% in the test set).

As regards the criteria chosen to quantify classification, in traditional classification, the standard evaluation criterion is the accuracy. In multilabel classification, a

direct extension of the accuracy is represented by the exact match ratio, which considers one instance as correct if and only if all associated labels are correctly predicted. [Fan & Friedman, 2008]

For the training set, the Exact Match value was 49.00% while for the test set it was reduced considerably to 36.00%, with a loss in performance.

However, this ratio may not be the most suitable performance measure as it does not count partial matches. For example, in text categorization, one news story may belong to both sports and politics. If one states that this story is related to only sports, the statement is partially correct.

Tague (1981) proposes two different criteria based on the F-measure: macro-average and micro-average F-measures where both consider partial matches. The F-measure is strictly related to both Recall and Precision. [Fan & Friedman, 2008]

On one side, if the classifier was able to identify well all the positive samples (Micro-Recall = 80.65%, Macro-Recall = 88.18%), on the other side there was a lack in term of Precision (Micro-Precision = 53.64, Macro-Precision = 49.26%). Therefore, according to the Precision indices values, the ability of the classifier not to label as positive a negative sample is low.

Combining the Precision and Recall indices, the measures of F1-score obtained resulted satisfactory and encouraging for this exploratory study.

The achieved average Accuracy and the Recall values are very good. For Precision instead, values have to be improved.

Precision is strictly correlated to False Positives Rate. This high rate arises from misinterpretation of common language terms for which better rules have to be defined. As described in section 4.2, this problem arose from a limitation of the used tool (MetaMap).

Trying to improve Precision leaving the Recall unchanged is challenging since they are related. As precision increases, recall decreases and vice-versa. A way to achieve better results will be to maximize the F1-score. [Lipton et al, 2014]

4.3 Classification

4.3.2 Comparison with classification based on keyword search

The key concept in the classification based on keyword search is to retrieve keywords from the apps descriptions.

The list of keywords used for the search was compiled using commonly words that the average user would enter in the search box. After their retrieval from the app's description, keywords were mapped to topical areas, and to each app was assigned a set of labels identifying the topical areas to which the app belonged to.

Results concerning classification based on keyword search are worse than ones obtained with the classifier based on text analytics. The average accuracy was 44.75% and 39.25% for the training set and the test set, respectively. These values show that the keyword-based classifier was not as good as the proposed classifier to identify the topical area(s) from which an app belongs to. Considering that a comprehensive list of keywords for each topical area was used, these estimates are on the safe side as it is reasonable that the average user will enter only a small subset of those keywords, and thus the real performance is likely to be lower. In terms of average accuracy, the method based on text analytics was +49.63% and +53.03% better than the one based on keyword search for the training set and the test set, respectively.

Regarding the Exact Match, the values were 28.75% and 27.75% for the training set and test set, respectively. Even if there was a lower loss (1.00%) in this index between the test set and the training set than in the first method (13.00%), these values are much lower than the ones previously obtained. Even if this ratio may not be the most suitable performance measure as it does not account for partial matches, the values suggest poor performance using just keywords. This may be related to the fact that apps descriptions, as provided by the developers, are general and not so focused on a topical area. On the other hand, the text analytics method here developed is context aware and able to classify apps based on concepts and their relationships.

Also, Precision and Recall results for keyword search were lower (Micro-Recall = 47.09%, Macro-Recall = 45.70%, Micro-Precision = 33.84%, Macro-Precision =

Discussions and conclusions

41.85%). Therefore, the ability to identify all the positive samples is low, as the ability to not label as positive a negative sample.

As a conclusion, the classifier based on text analytics is better than the one based on keywords search since Precision, Recall and F1-Score obtained with the first classifier are higher than the ones obtained with the second classifier.

These results confirmed the expected hypothesis that searching for keywords is less powerful than searching for concepts because searching keywords means simply retrieve them from a text without understanding how they are used and interpreted into apps' descriptions.

Table 4.2 - Example of comparison with keyword-based classifier and CUIs-based classifier.

App's name	App's description
Eye Workout	Minimum operations - just set notification in settings and follow for exercises that will take no longer than 5 minutes, 3 times per day and your eyesight will improve dramatically! And when your brain realizes that vision improves, these workouts will deliver the kind of pleasure for you!...

Table 4.2 shows an example that describes why searching concepts is more powerful than searching keywords. If a user types "workouts" in the search box for searching fitness apps, he will retrieve also this app. It is noticeable that "Eye Workout" it's related to other than "Fitness and Wellness".

Furthermore, the real performance may be even lower because when users search for app, the iTunes App Store returns a list of apps that are ranked based on matching for app's title, keywords, and primary category. [W16] As a direct consequence, for example, if the developer has designed an app for the first aid (i.e. "Emergency Medicine" as topical area) and "emergency responder" was not indicated as keyword, a search by the user with the term "Emergency First Responder" will not result in desired app.

4.3 Classification

The same app searched using the proposed text analytics method will result in the positive finding of the app in the search list.

Compared to conventional keyword-based search, the proposed method represents the basis for a filtering tool that is context-aware, as it is based on computational-linguistic techniques and also it includes algorithms to estimate the probability of the correct interpretation of terms and phrases (i.e., the MetaMap scores), as well as optimized rules that enable classification into multiple categories.

4.3.3 The whole database

By the application of the proposed method to the whole database, the obtained results deserve some attention. A relatively high number of apps (8397 out of the 20998 "NC") resulted not medically relevant even if they were declared to be of medical interest since they belonged to M category.

Also, 12061 out of the 20998 "NC" apps declared to be related to fitness and wellness were not.

As regard apps medically relevant, the topical areas with the highest distribution were "Fitness and Wellness", "Nutrition", "Across Specialties", "Cardiology and Cardiovascular Medicine".

Apps are present in everyday lives. They have become the primary tools for communicating, navigating, working, and entertaining. With the rise of fitness apps, smartphones are also helping individuals improve their health and wellness through a suite of tracking, coaching, and other lifestyle apps. The growth in this segment of the app stores has been explosive in recent years. According to the mobile analytics firm Flurry [W17], health and fitness app usage in the U.S. has seen a sharp uptick over the past few years, growing by 330% between 2014 and 2017. Most phones' default health apps (Apple Health, Google Fit, or S Health) automatically track steps and distance taken, which in itself can accommodate a walking or a step challenge at work. Workout apps will log the time spent in each session, heart rate (if paired with a capable activity tracker), and calories burned. A new wave of meditation apps will also record a measurement called "mindfulness

moments,” which logs how many minutes users spent meditating. This is a rich new way for companies to expand their wellness programs beyond exercise and nutrition. Research [W18] has also shown that the act of recording one’s food intake in a diary leads to better food choices and better weight control.

Not surprisingly, “Fitness and Wellness” and “Nutrition” were the two major topical areas of mHealth on the US iTunes Store.

“Across Specialties” followed the previous two. This topical area groups apps generally related to medicine, medical education, nursing, healthcare rather than to one or more specialties. As a direct consequence, all the apps related to medicine with a too general description are grouped in this topical area.

The last topical area with a relevant distribution of apps was “Cardiology and Cardiovascular Medicine”. In the last years, technology had a high impact in this field thanks to new and precise way to monitor vital signs like heart rate, pressure, blood oxygenation with the simple use of the smartphone’s camera. Lately, new implantable devices which transfer data via Bluetooth have been developed (like implantable cardioverter-defibrillators) and thus also the availability of mHealth apps in this field has grown a lot.

4.4 Conclusions and future developments

Assessing the distribution and pattern of features in the market or in specific application areas could be useful to highlight possible trends and gaps in the market, as well as challenges and opportunities for developments, and to monitor these trends along time.

This could be possible by identifying multiple times in a year both the distribution of mHealth apps on the store and their distribution among the topical areas. It could also be useful to assess the quality of the apps by considering other factors like, for example, the last update date, or the ratings users have assigned to the apps.

To perform this analysis, firstly the algorithm crawling webpages must parallel the changes in the structure of the iTunes Store website. In addition, word sense

4.4 Conclusions and future developments

disambiguation has to be improved to better recognize the topical areas of the apps.

Finally, to identify useful patterns among population, also other stores have to be taken into consideration (Google Play Store, Windows Store, etc.)

Further developments are necessary to improve classification performance, also including additional features and web sources, as well as assessing factors that influence classification accuracy, so to take full advantage of the potential offered by natural language processing for the analysis of large databases.

A closer look at the mismatches between manual and automated characterization showed that failures of the algorithm were frequently related to the presence of general terms in the description whereas MetaMap could hardly classify these apps as related to health or medicine. The observed trends suggest that the identification of apps that are Across Specialties (i.e. apps related to general medicine, medical education, nursing, or healthcare in general) poses major problems. This is likely due to the inherent tendency of MetaMap to characterize terms, including the most general ones, as related to specific topical areas [Aaronson, 2001]. For the same reason, the proposed method might tend to characterize as related to health or medicine some apps that have no medical content. Therefore, additional rules and ad-hoc vocabularies including common language terms to complement the highly specific UMLS terms used by MetaMap should be developed and included in future versions in order to better identify the medical content of apps and to characterize those that have general medical content (i.e., Across Specialties).

In addition, some categories are inherently close to each other in terms of related vocabularies. For example, Fitness & Wellness share several terms with Cardiology and with Physiatry & Orthopedics. Similarly, Pneumology has many terms in common with Mental Health, Neurology, Psychiatry and with Fitness & Wellness (especially related to relaxation and stress relief).

In general, it might be useful to investigate whether, in addition to the analysis of terms and scores considered in this first version of the tool, the analysis of

Semantic Types (as retrieved by the MetaMap algorithm) might assist in a more robust characterization of topical areas. For example, some Semantic Types such as, e.g., Biomedical Occupation or Discipline, Daily or Recreational Activity are likely to be revealing about the topical area, and thus could be thus taken into account in the analysis as additional weighting factors or elements for the classification rules.

Similarly, Semantic Types such as Animal and Functional Concept can assist in the identification of apps not related to health or medicine, i.e. recreational apps or games, respectively. In general, it will be important to further evaluate in future studies which Semantic Types to include and how to analyze them as well as the potential improvements in classification accuracy.

Nevertheless, the results of this study suggested that automated methods based on text analytics are helpful to extract meaningful information from the app stores' webpages regarding the topical areas of apps. It will be important to upgrade the method by including automated characterization of additional features like apps' promoters, offered services, and target users. In Medicine or more general in Health, a great importance is related to the understanding of the user type for whom the app has been designed.

It's important to understand if an app has been designed for patients or for caregivers. Furthermore, if an app has been designed for patient, it's fundamental to understand if it has been designed for chronic patients, general patients or for users in general. In addition, it's high fundamental to understand if the app is considered as a medical device.

Another important aspect is the offered services: it's important to know if an app is designed for primary prevention, specific prevention, or it has been designed to help users during rehabilitation, or it's related to patient's management.

To understand these important aspects, the information present in the apps' webpages are not enough since the apps' description reported on the app store can provide only a partial picture of the various features of the apps and the

4.4 Conclusions and future developments

information reported is usually fragmented and sometimes incomplete since it depends only on the developer's behavior.

Nevertheless, even if the method proposed in this thesis was tested and used on the US iTunes Store, in principle, it can be adapted for use in various platforms and can be used to extract meaningful information by combining several sources. To this a lot of time has to be spent to analyze both the structures of the different sources and the availability of the same information among them to guarantee homogeneous and consistent data. In addition, methods to identify duplicates among the different sources have to be developed.

Another strategic direction for improving the proposed method and further support clinicians in their practice would be to explore ways to include direct or indirect measures of quality. This is particularly challenging as it is difficult to identify the core components of quality as well as appropriate measures to assess them. How to combine automated characterization of descriptive features with assessment, possibly automated, of quality measures is an entirely open question. It will be important in future studies to investigate whether some methods, among those proposed in the literature, could be used or adapted to be included in an automated approach. This would be of great value to fully empower clinicians with tools to assess and compare the quality of the several apps available with the ultimate goal of providing greater benefit to patients.

Following improvement and optimization of the method, quantitative analysis of its performance and computational running time compared to conventional keyword search will be essential to assess the benefit of this approach and its viability for use in real-time characterization of apps.

The availability of automated tools for app filtering and characterization could be a valuable opportunity for potential app users (patients, healthcare professionals, as well as citizens) and support them in informed, aware selection and adoption of health apps.

Bibliography

- Aronson, A. R. (2001). Effective mapping of biomedical text to the UMLS Metathesaurus: the MetaMap program. In *Proceedings of the AMIA Symposium* (pp. 17-21). American Medical Informatics Association.
- Aronson, A. R. (2006). MetaMap: Mapping text to the UMLS Metathesaurus. *Bethesda, MD: NLM, NIH, DHHS*, pp. 1-26.
- Browne, A. C., McCray, A. T., & Srinivasan, S. (2000). The specialist lexicon. *National Library of Medicine Technical Reports*, pp. 18-21.
- Cutting, D., Kupiec, J., Pedersen, J., & Sibun, P. (1992, March). A practical part-of-speech tagger. In *Proceedings of the third conference on Applied natural language processing* (pp. 133-140). Association for Computational Linguistics.
- Fan, J. W., & Friedman, C. (2008). Word sense disambiguation via Semantic Type classification. In *AMIA Annual Symposium Proceedings* (Vol. 2008, p. 177). American Medical Informatics Association.
- Godbole, S., Sarawagi, S. (2004). Discriminative methods for managing variation in biomedical terminologies. In *Proceedings of the 8th Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pp. 22-30.
- Goyvaerts J., Levithan S. (2012). Introduction to Regular Expressions, "Regular Expressions Cookbook, 2nd ed." (pp. 1-27), O'Reilly Media.
- Lipton, Z. C., Elkan, C., & Naryanaswamy, B. (2014, September). Optimal thresholding of classifiers to maximize F1 measure. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases* (pp. 225-226). Springer, Berlin, Heidelberg.
- McCray, A. T., Srinivasan, S., & Browne, A. C. (1994). Lexical methods for managing variation in biomedical terminologies. In *Proceedings of the Annual Symposium on Computer Application in Medical Care* (pp. 235-239). American Medical Informatics Association.

- Murphey, Y. L., Guo, H., & Feldkamp, L. A. (2004). Neural learning from unbalanced data. *Applied Intelligence*, 21(2), pp. 117-128.
- Paglialonga, A., Lugo, A., & Santoro, E. (2018a). An overview on the emerging area of identification, characterization, and assessment of health apps. *Journal of biomedical informatics*, 83, pp. 97-102.
- Paglialonga, A., Schiavo, M., & Caiani, E. G. (2018b). Automated Characterization of Mobile Health Apps' Features by Extracting Information from the Web: An Exploratory Study. *American Journal of Audiology*. Accepted, in press.
- Read, J., Pfahringer, B., Holmes, G., & Frank, E. (2009). Classifier chains for multi-label classification. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases* (pp. 261-262). Springer, Berlin, Heidelberg.
- Stehman, S. V. (1997). Selecting and interpreting measures of thematic classification accuracy. *Remote sensing of Environment*, 62(1), pp. 77-89.
- Xu, W., & Liu, Y. (2015). mHealthApps: a repository and database of mobile health apps. *JMIR mHealth and uHealth*, 3(1), e28.
<http://doi.org/10.2196/mhealth.4026>

Webliography

[W1] The state of mobile health apps in 2018.

Retrieved from:

<https://ehealth.intersog.com/blog/the-state-of-mobile-health-apps-in-2018>

[W2] NHS Choices Health Apps Library.

Available at:

<https://apps.beta.nhs.uk>

[W3] RANKED Health.

Available at:

<http://www.rankedhealth.com>

[W4] iMedicalApps.

Available at:

<http://www.imedicalapps.com>

[W5] Organization for the Review of Care and Health Applications (ORCHA).

Available at:

<https://www.orchacoh.co.uk/about>

[W6] MobiHealthNews.

Available at:

<http://www.mobihealthnews.com>

[W7] MyHealthApps.

Available at:

<http://myhealthapps.net>

[W8] U.S. Food and Drug Administration.

Retrieved from:

<https://www.fda.gov/default.htm>

[W9] Judgment of the Court (fourth chamber), 2017.

Retrieved from:

[http:// curia.europa.eu/juris/document/document.jsf;jsessionid=9ea7d2dc30d62f2457a04be84233bc82c982c0eec480.e34KaxiLc3qMb40Rch0SaxyNa3z0?text=&docid=197527&pageIndex=0&doclang=EN&mode=req&dir=&occ=first&part=1&cid=1166739](http://curia.europa.eu/juris/document/document.jsf?jsessionid=9ea7d2dc30d62f2457a04be84233bc82c982c0eec480.e34KaxiLc3qMb40Rch0SaxyNa3z0?text=&docid=197527&pageIndex=0&doclang=EN&mode=req&dir=&occ=first&part=1&cid=1166739)

[W10] Number of apps available in the leading App Stores.

Retrieved from:

<https://www.statista.com/statistics/276623/number-of-apps-available-in-leading-app-stores/>

[W11] World Wide Web Consortium, 2018.

Retrieved from:

<https://www.w3.org/standards/webdesign/htmlcss>

[W12] UMLS Reference Manual, Bethesda (MD), 2009.

Retrieved from:

<https://www.ncbi.nlm.nih.gov/books/NBK9684/>

[W13] Apple Developer Guidelines, 2018.

Retrieved from:

<https://developer.apple.com/app-store/categories/>

[W14] International Organization for Standardization – Languages Code.

Retrieved from:

<https://www.iso.org/iso-639-language-codes.html>

[W15] Union Européenne des Médecins Spécialistes, the European Union of Medical Specialists.

Retrieved from:

<https://www.uems.eu/about-us/medical-specialties>

4.4 Conclusions and future developments

[W16] Optimizing for App Store Search, 2018.

Retrieved from:

<https://developer.apple.com/app-store/search/>

[W17] Fitness App Usage Grows by Leaps and Bounds, 2017

Retrieved from:

<https://www.emarketer.com/Article/Fitness-App-Usage-Grows-by-Leaps-Bounds/1016486>

[W18] Keeping A Food Diary Doubles Diet Weight Loss, Study Suggests, 2008.

Retrieved from:

<https://www.sciencedaily.com/releases/2008/07/080708080738.htm>

Ringraziamenti

Primi tra tutti ringrazio voi, mamma papà: senza di voi probabilmente non so se oggi sarei qui; o forse sì, ma certamente in maniera diversa. Vi ringrazio per avermi permesso di intraprendere questo cammino, di avermi sempre sostenuto e mai criticato, di aver creduto e di credere in me, e soprattutto di avermi sopportato: sicuramente ora ci sarà qualche foglio e quaderno in meno sparsi in casa, e qualche tavolo libero in più ;)

Un grazie fondamentale a te, Marco: non so cosa possa voler dire essere figlio unico...e non ci tengo nemmeno a saperlo! Grazie fratello per esserci stato sempre! E grazie Roberta, mia super cognata, per la vicinanza in questi lunghi anni...partita in un modo decisamente singolare e divertente!

Un grazie basilare lo rivolgo alla dr. Alessia Paglialonga e il prof. Enrico G. Caiani, oltre che per l'aiuto e la pazienza fornitemi durante tutto il periodo di stesura della tesi, anche per la vicinanza e comprensione avuta in un momento particolarmente difficile per me e la mia famiglia. A loro sono grato per il traguardo che oggi ho finalmente raggiunto.

Un grazie particolare al mio "amo", Gessichina: grazie per le infinite ore d'ascolto dei miei poli-sfasi, grazie per esserci sempre e fare il tifo per me. Si dice che gli amici rispetto ai parenti abbiano una grande differenza: i primi li scegli mentre i secondi te li ritrovi. Beh, io sono proprio contento di averti trovato già al mio fianco senza aver fatto fatica per cercarti :)

Un grazie di cuore al mio mon amour, Sara: grazie per le infinite ore passate a sentirmi ripetere lezioni in italiano e in inglese, ma soprattutto grazie per avermi scelto come amico e non avermi mai lasciato andare ed essere stata così vicino a me in questi anni!

Grazie a Ilaria, Annalisa, e Yari: le prime persone conosciute qui al Poli con cui ho condiviso tutti i giorni del mio primo anno. Grazie per essermi amici ancora oggi nella vita.

Grazie a tutto il gruppo "Michele non sarai admin" e Daniele, amici universitari con cui ho condiviso il mio percorso in triennale. Di questi un grazie importante va a te e alla tua famiglia, Elena.

Un grazie a Lucia e Patrizia, compagne nel mio biennio di magistrale e nuove amicizie di questi anni. Con voi non solo il Poli è stato meno pesante, ma è stato tutto più bello!

Grazie Chiara, grazie infinite del percorso condiviso: non solo sei stata una mia compagna universitaria, sei stata la persona e amica che in questi due anni di magistrale ho visto più di chiunque altro! Grazie per la tua meticolosa attenzione che ha fatto sì io mi distraessi di meno, grazie per avermi ascoltato in tutte le cose che non riguardavo l'università, grazie anche per tutte le cose che mi hai spiegato all'università, soprattutto nel panico preesame di tutti i nostri esami :-). Grazie perché so che parlo sempre e alla velocità della luce eppure tu sei sempre stata lì ad ascoltarmi!!! Grazie, per aver camminato a fianco a me in questi due anni!

Un ultimo grazie - ma non meno importante - va a tutti voi altri: sapete benissimo che se dovessi elencarvi tutti finirei per scrivere una tesi di ringraziamenti su ognuno di voi! Grazie a voi, amici del quotidiano, perché mi permettete di condividere un pezzetto della mia vita con voi!