TableTidier: An UMLS powered data extraction tool

Jesus A. Rodriguez Perez, PhD[1], Elaine Butterly, PhD[1], Lili Wei[1], Avirup Chowdhury, MSc[2], Peter Hanlon, MSc[1], David McAllister, MD,[1]
[1]The University of Glasgow, UK
[2]Public Health Suffolk, Suffolk County Council, UK

Abstract

TableTidier is a software designed to assist in the clinical systematic review process. Systematic reviews is an essential exploratory procedure which focusses on the extraction of data from a sample of clinical documentation. Although many tools exist to assist in different areas of the systematic review process, no solution addresses the extraction of table data into a machine readable format effectively. Consequently, researchers are mostly left to extract data manually which poses obvious limitations in terms of the depth and breadth of their efforts. Our tool combines human-interaction, supervised classification methodologies, and web technologies to provide an end-to-end tool for data extraction in the context of clinical systematic reviews. In this paper, we study the effects of including UMLS-based features into our classification models, and its impact on the overall architecture of TableTidier. Our evaluation results demonstrate that including UMLS-based features consistently improves the performance of our classifiers, thus optimising the behaviour of TableTidier, which in turn reduces the labour needed by humans when undertaking a systematic review.

Introduction

Systematic reviews are highly influential in clinical decision making[1]. Following a pre-specified protocol, systematic reviews seek to obtain and extract all available relevant information on a specific clinical question, with the largest and most influential reviews involving many hours of work for highly-trained researchers[2].

Consequently, a number of software tools[3] have been developed to assist researchers with the numerous labour-intensive tasks involved in systematic reviewing such as screening published papers for relevance, assessing studies for their risk of bias and extracting data from published results[4–6]. There are also a number of more generic software tools which may help with some aspects of systematic reviewing. For example, there are extensions available for programming languages commonly used in data analysis that provide functions for processing tabular data (E.g. Unpivotr[7] and Databaker[8]).

However, none of these tools are designed to assist in the (semi)-automatic extraction and standardisation of tabular results from published papers. Standardising such tables is not a trivial task; in the medical literature table design is highly idiosyncratic. Even where there are established reporting guidelines[9] there are no standards for table design, and aesthetic or branding considerations appear to be at least as important as consistency and accessibility. Indeed, features such as multi-level headers are common, and descriptions (labelling) of data-containing cells must often be inferred from ambiguous features such as formatting and the relative position of labels (Table 1). Consequently, it requires both time and expertise to extract results from tables in the published literature and such data extraction is potentially error-prone.

In traditional systematic reviews of clinical trials, analyses might be based on only a few numbers for each trial. However, advances in the field of clinical trial meta-analysis mean that richer data are now needed[10] while systematic reviews of epidemiological studies often involve the extraction of large quantities of data covering a range of exposures and outcomes[11]. Consequently, it has become more important to find ways to improve the efficiency of tabular data extraction.

To address this issue, we developed TableTidier, a software tool which assists the conversion of tabular data to standard formats where each value is unambiguously labelled. Developed using medical journals articles reporting clinical trial findings, TableTidier uses a machine learning algorithm (a support vector machine) to estimate the table structure and content, based on the position, formatting and actual text of the table content.

We hypothesised that the use of the Unified Medical Language System (UMLS), which has been shown to improve such diverse tasks as information retrieval, as[12] and[13], would improve the precision, accuracy and recall of this classification task.

Research Problem

Extracting data from tables is very useful albeit a very challenging task. The main hurdle in automated table data extraction is need to preserve the relationship of data cells with respect to the headings that describe them.

Participants		
	Placebo	Aspirin
Gender	200	300
Male	90	160
Female	110	140

Table 1. Example Table

Consider Table 1 as an example. We can easily see that the data cell 200 is related to the Gender and Placebo headings. Similarly, 140 is related to the Female and Gender as well as Aspirin and Participants concepts. We can tell because the cells containing numbers are aligned with the headings, but also because of the indented formatting which aligns related concepts such as Gender to Male and Female. An example of the desired machine readable form is shown in Table 2.

Participants	Placebo	Gender		200
Participants	Placebo	Gender	Male	90
Participants	Placebo	Gender	Female	110
Participants	Aspirin	Gender		300
Participants	Aspirin	Gender	Male	160
Participants	Aspirin	Gender	Female	140

Table 2. Example Machine Readable Data

These are visual cues that are easily interpreted by our brains, no matter how complex the tables are. Unfortunately, this behaviour is not so easily translated into algorithmic solutions. Additionally, we associate terms semantically, independently of their position in the table. Finally, although there are patterns, the data can be organised in any arbitrary way resulting in potentially infinite possible structures.

However, we believe there are basic elements that can be effectively exploited to enable semi-automated data extraction. These features can be derived from the position of concepts within the table and their textual formatting. Consequently we believe that by describing the structure of a table in terms of these features, we can use this description as a proxy to extracting the data from cells whilst keeping the relationships to the headings.

Furthermore, we believe that the inclusion of semantic features capable of linking concepts through its associated metadata can improve the fidelity of this semi-automated process. Since UMLS contains the vocabulary utilised in our domain, and is already organised as an ontology, it is a promising resource for our purposes. Thus we pose the following research question to drive our investigation:

Can UMLS features enhance the performance and behaviour of the automated annotations provided by TableTidier?

	Stent plus placebo	Stent plus abciximab	p*	Balloon angioplasty plus abciximab	Pt
Diabetic patients					
Total	173	162		154	
Death	7 (4-1%)	2 (1-2%)	0-11	4 (2-6%)	0-440
Death, large MI	24 (14-0%)	8 (4-9%)	0-005	16 (10-3%)	0-290
Large MI	19 (11-1%)	7 (4-3%)	0-022	13 (8-4%)	0-389
TVR	39 (22-4%)	22 (13-7%)	0-035	39 (25-3%)	0-546
Non-diabetic patients					
Total	635	632		640	
Death	12 (1-9%)	6 (1-0%)	0-156	13 (2-0%)	0-852
Death, large MI	65 (10-3%)	34 (5-4%)	0-001	48 (7-5%)	0-084
Large MI	55 (8-7%)	28 (4-4%)	0-002	38 (6-0%)	0-062
TVR	87 (13-7%)	99 (15-6%)	0-369	120 (18-7%)	0-014

Figure 1. TableTidier: Table to be annotated

System Architecture

The TableTidier system is a web application composed by a number of modules organised within a classic Server / User Interface (UI) infrastructure.

Server written in Node.js combines all sub-modules that provide the data to the User Interface (UI):

- *Automatic Annotation*: Predicts the structure of a table from the position and formatting of the different concepts.
- *Data Extraction*: Module written in R which generates the machine readable data, from the table annotations.
- *Database*: Provided by PostgreSQL, and holding data for results, user interactions, look-up tables, etc.
- *MetaMap Interface*: Communicates with a "dockerised"[1] version of MetaMap in order to assign concepts to the strings extracted from cells.

User Interface (UI) coded in React.js, and comprises the following modules:

- *Table Annotator*: Allows the users to manually and/or automatically annotate the structure of a table, for later data extraction.

[1] https://metamap.nlm.nih.gov/DockerImage.shtml

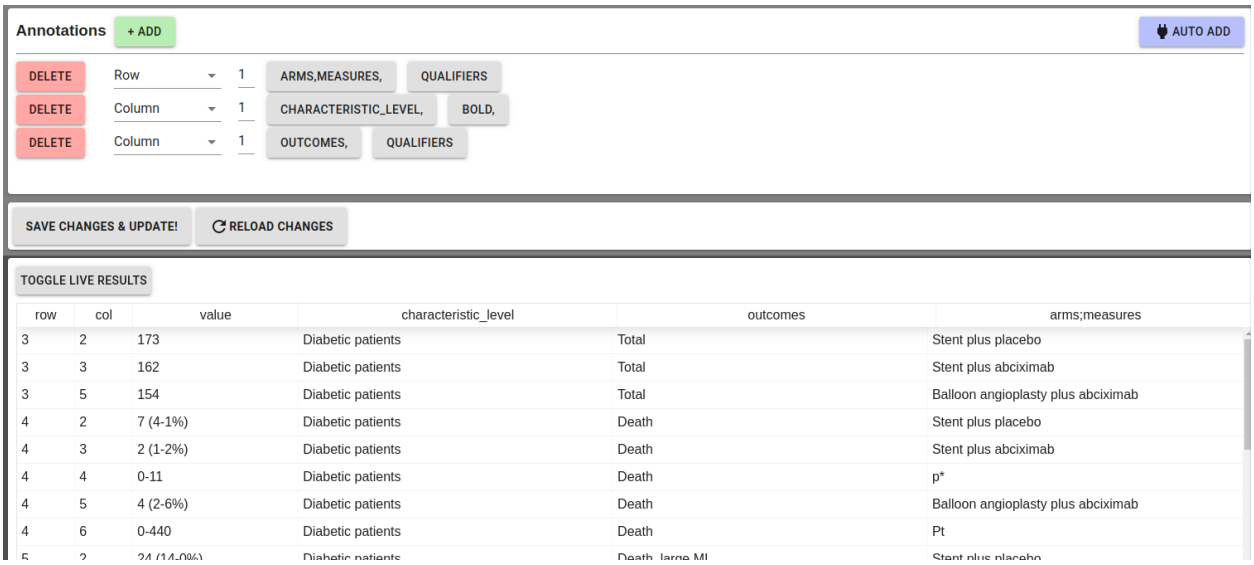

Figure 2. TableTidier: Annotation and Live preview example

- *Live Table Editor*: Allows the user to edit tables on the fly, mostly used to fix OCR or transcription issues.
- *Live Data Previewer*: Shows the data extracted on the fly, as users make changes to the table annotations.
- *UMLS Concept Reviewer*: Allows to manually review, and edit the UMLS concepts automatically assigned to each of the strings in the tables.

Data Extraction By Annotation

Our approach to table data extraction involves a number of steps to be carried out by a user through TableTidier's UI:

1. For every table to be annotated (Figure 1), the user will be presented with two options (Figure 2):
   A. Provide a manual "annotation" which describes the structure of the table in terms of its row/column content and format. Annotations are described by assigning a label to each relevant row and column which relates to its content, and by tagging any relevant formatting options as shown in Figure 2.
   B. Utilise the "Automatic Annotator Module" which attempts to simulate a human annotation (The focus of this study), and edit the annotation if needed.
2. In both cases, the "Live Data Previewer" (Figure 2) attempts to extract the data given the previous annotation, and display a preview of the machine readable format.
3. Once all tables are annotated, the user can obtain all extracted data in a machine readable format directly from the interface.

Optional 1 If any problems are encountered, the user can manually edit the table with the Live Table Editor.

Optional 2 UMLS concepts will be automatically assigned to each of the table concepts, and the user may decide to review and/or later them in the UMLS Concept Reviewer

Data sources

Tables Dataset : The 1900 tables in our dataset originate from the "denominator trials" shown in Figure 1 of our recent publication[14]. Briefly, eligible trials were registered via the US Clinical Trials Register (clinicaltriasl.gov), started on or after 1st January 1990, were phase 3 or 4, recruited at least 300 participants, had an upper age of at elast 60 years and evaluated drugs for a selected set of chronic conditions.

UMLS : The UMLS[15], or Unified Medical Language System, is a set of medical ontologies and interfacing software, which allows the interaction between many biomedical vocabularies, thus enabling the interoperability of computer systems. Its Metathesaurus includes popular vocabularies such as ICD-10, MeSH and SNOMED, linked together by means of a semantic ontology which associates concepts through semantic relationships.

MetaMap: MetaMap[16] is software developed by Dr. Alan (Lan) Aronson at the National Library of Medicine (NLM) which allows the processing of text utilising NLP[2] techniques, and its subsequent linkage to the UMLS metathesaurus. We make extensive use of MetaMap in our evaluation to derive our UMLS-based features, in particular CUIs[3] and SemanticTypes[17].

Features

1. **pos_start**: Whether the concept appears in the first row of the table.
2. **pos_middle**: Whether the concept appears between the first and last rows of the table.
3. **pos_end**: Whether the concept appears in the last row of the table.
4. **inRow**: Whether the concept appears in a row containing headings
5. **inCol**: Whether the concept appears in a column of headings
6. **is_bold**: Whether the concept is formatted as bold
7. **is_italic**: Whether the concept is formatted as bold
8. **is_indent**: Whether the concept is formatted as indented
9. **is_empty_row**: Whether the concept is in the only populated cell of its row
10. **is_empty_row_p**: Whether the concept appears in a row, where the only other populated cell contains a "P value"
11. **semanticTypes**: The UMLS semantic groups assigned by MetaMap to the text on each cell. (*e.g. "inpo" semanticType code for "Injury or Poisoning"*)
12. **cuis**: The Concept Unique Identifiers (CUIs) assigned by MetaMap to each of the text strings in the table cells. (e.g. C0001779 which represents the concept "Age")

Note: Multiple CUIs and semanticTypes can be associated with a single string of text.

Evaluation

In this section we introduce the specifications of our evaluation dataset, evaluation metrics and the classifiers utilised in this study.

[2]Natural Language Processing
[3]Concept Unique Ids

Dataset

Our evaluation dataset comprises manual annotations on the 1900 tables mentioned above. We performed a feature extraction over all concepts belonging to either annotated rows or columns, and all features described in the previous Section. This was done on the basis of the original table label strings, prior to identification of any CUIs from metamap.

We then mapped each string to metamap, resulting in a matrix with 40729 rows and 6550 features, as each possible CUI and semanticType is assigned a column.

The features introduced in the Section above, were utilised to produce four different feature sets to thoroughly test our classifiers:

- **Basic**: Set without UMLS features (Features 1-10).
- **UMLS-SemTypes**: Same as Basic, but adding the **semanticTypes** feature from UMLS (Features 1-11).
- **UMLS-CUIs**: Same as Basic, but adding the **cuis** feature from UMLS (Features 1-10, 12).
- **UMLS-Full**: Includes all features (Features 1-12)

The dataset was split into training (70%) and testing (30%) stratifying on the target labels to be classified (eg arms, characteristic_level, characteristic_name, measures, other, outcomes, p-interaction, time/period), which ensured all labels are well represented[4].

Classifiers

To test whether any additional benefits from adding UMLS features to the text are dependent on a particular modelling approach, we repeated the modelling using a number of commonly used classifiers based on diverse methodologies:

- Neural Network: Multi-layer Perceptron (**MLP**)
- Naive-Bayes: **MultinomialNB**
- Logistic Regression: **LogReg**
- Tree: **RandomForest**
- Ensemble: **AdaBoost**

Evaluation Metrics

We used a range of evaluation metrics across for each of these comparisons, precision, recall, F1-Score, accuracy Macro_Avg and weighted_Avg:

- **Precision**: Also referred to as positive predictive value, the number of correctly assigned cells as a proportion of the total number of cells assigned to that label
- **Recall**: Also referred to as sensitivity, the number of correctly assigned cells as a proportion of the total number of cells which should have been assigned to that label
- **F1-Score**: A metric that combines precision and recall given a mixing parameter F. $F1$ in particular gives the same importance to Precision and Recall, thus $F = 1$

[4]https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.train_test_split.html

Features	Metrics Averaged Over All Target Labels (Macro_avg)				
	Classifier	**Precision**	**Recall**	**F1-Score**	**Accuracy (12k)**
Basic	MLP	**0.46**	**0.23**	**0.19**	**0.59**
	MultinomialNB	0.22	0.22	0.17	0.57
	LogReg	0.21	0.22	0.18	**0.59**
	RandomForest	0.39	**0.23**	**0.19**	**0.59**
	AdaBoost	0.14	0.21	0.17	0.58
UMLS-SemTypes	MLP	0.52	0.42	0.45	0.64
	MultinomialNB	0.34	0.29	0.28	0.60
	LogReg	0.42	0.29	0.29	0.62
	RandomForest	**0.56**	**0.43**	**0.46**	**0.65**
	AdaBoost	0.33	0.23	0.22	0.57
UMLS-CUIs	MLP	0.61	**0.51**	**0.53**	**0.67**
	MultinomialNB	0.42	0.35	0.36	0.62
	LogReg	0.59	0.39	0.42	0.66
	RandomForest	**0.65**	**0.51**	0.52	**0.67**
	AdaBoost	0.23	0.24	0.20	0.59
UMLS-Full	MLP	**0.59**	0.52	**0.54**	**0.67**
	MultinomialNB	0.42	0.36	0.36	0.62
	LogReg	0.58	0.39	0.42	0.66
	RandomForest	**0.59**	0.50	0.53	**0.67**
	AdaBoost	0.23	0.26	0.21	0.59

Table 3. Summary statistics for all feature sets and classifiers

- **Accuracy**: Is the proportion of correctly assigned labels from all test samples
- **Macro_avg**: Standard average over results across all labels
- **Weighted_avg**: Average weighted over the number of existing samples for each of the labels to be predicted

— Seems[18] that ttest is appropriate for IR metrics given their experiments —

Discussion

Table 3 shows summary results for all classifiers with four different feature sets, *Basic*, *UMLS-SemTypes*, *UMLS-CUIs* and *UMLS-Full*. The results were averaged over the results obtained for each label to be classified. In other words, the evaluation metrics were computed as in Table 5 for all labels (e.g. arms, measures, outcomes...), and we utilised the average over results for all labels (e.g. macro_avg) to produce the summary results in Table 3. Accuracy is an exception as it does not relate to any labels and thus was computed over the whole test set.

There was substantial variation in the performance across the classifiers. Nonetheless, all five showed improved results for all evaluation metrics on the addition of UMLS features.

There was an improvement in all precision (and therefore by definition recall) with no loss of accuracy. UMLS semantic types[17] alone improved precision, recall and accuracy compared to basic feature set. However, there was a larger improvement with the addition of CUIs. There was no evidence of further improvement when both sematic types and CUIs were combined in the modelling (*UMLS-Full*).

Since the improvement was similar across all classifier types, we used the RandomForest classifier as an example to formally test whether the changes in accuracy with addition of UMLS features were statistically significant. Table 4, as expected, is consistent with the findings in Table 3. There was statistically significant

	Basic	UMLS-SemTypes	UMLS-CUIs	UMLS-Full
Basic		1.47^{-61}	9.91^{-77}	7.67^{-90}
UMLS-SemTypes	1.47^{-61}		6.25^{-6}	3.25^{-11}
UMLS-CUIs	9.91^{-77}	6.25^{-6}		0.01
UMLS-Full	7.67^{-90}	3.25^{-11}	0.01	

Table 4. Paired T-Test p-values comparing RandomForest classification runs

	Precision	Recall	F1-Score	Samples
arms	0.53	0.81	0.64	1217
characteristic_level	0.60	0.95	0.73	6466
characteristic_name	0.37	0.01	0.01	2636
measures	0.72	0.04	0.08	754
other	0.73	0.05	0.09	167
outcomes	0.12	0.00	0.00	895
p-interaction	0.00	0.00	0.00	10
time/period	0.00	0.00	0.00	74
macro_avg	0.39	0.23	0.19	12219.00
weighted_avg	0.52	0.59	0.46	
accuracy	0.58			

Table 5. RandomForest classifier results, basic features (No UMLS)

evidence (at the conventional $p < 0.05$) level of improved performance on adding UMLS semantic types or CUIs to the basic feature set, and better performance of CUIs compared to semantic types, but little evidence for using both CUI and semantic types.

Tables 5 and 6 show similar metrics as Table 3, solely for the *RandomForests* classifier, but broken down by the different label types. The performance improves for all features when utilising UMLS features as evidenced across all metrics, with the exception of the "other" category. Since the "other" label was poorly defined by the manual reviewers, it is difficult to interpret this finding. For some labels, "p-interaction" and "time/period" there were striking improvements in performance when UMLS features were included.

We can conclude that utilising UMLS features significantly improves performance of our classifier, which in turn should positively affect the automatic annotation module of TableTidier. This improvement translates directly to less manual labour for the users, thus reducing the time involved in data extraction in tasks such

	Precision	Recall	F1-Score	Samples
arms	0.82	0.85	0.83	1217
characteristic_level	0.68	0.86	0.76	6466
characteristic_name	0.46	0.22	0.30	2636
measures	0.72	0.62	0.66	754
other	0.60	0.32	0.42	167
outcomes	0.68	0.53	0.60	895
p-interaction	0.20	0.10	0.13	10
time/period	0.55	0.57	0.56	74
macro_avg	0.59	0.51	0.53	12219
weighted_avg	0.65	0.67	0.64	
accuracy	0.66			

Table 6. RandomForest classifier results with Full UMLS features enabled

as systematic reviews.

Conclusion

Systematic reviews are highly influential in clinical decision making, and increasingly require the extraction of large quantities of data from tables published in scholarly journals, which is a challenging and labour-intensive task. We found that adding UMLS features to software designed to aid this task markedly improved precision, recall and accuracy.

We built a system that exploits the structure of tables in order to convert them into a machine readable format, in a semi-automated manner, by means of a classifier. The basic software relied upon the position of concepts within the tables, as well as the existence of formatting cues. However, we also found that adding UMLS-based features into the classifier - CUI codes and SemTypes assigned by MetaMap which represent features with different levels of specificity - markedly improved prediction. Importantly, this finding was robust to the choice of classifier (eg random forest, neural network), making it unlikely that a better machine-learning algorithm, or more skill in fitting such a model would, without the rich contextual knowledge embedded in UMLS and MetaMap, have been unlikely to achieve similar levels of performance.

MetaMap is a complex software with many options and functions. We elected to use the out-of-the-box defaults, and alternative options may have led to better or worse performance. However, our decision to do so does mean that over-fitting is a highly unlikely explanation for the improved performance we observed, since we made a very small number of modelling choices (i.e. the basic model versus two different sources of information from Metamap using default settings.)

A strength of this study includes the fact that the allocation of labels by the manual reviewers was done blinded to (indeed prior to) the mapping of strings to UMLS concepts. However, there are number of weaknesses in our study. First, we only examined tables for one subject area, clinical trials, and we do not know if the use of UMLS will increase performance similarly in closely related fields such as pharmaco-epidemiology, or in moderately-related fields such as genetic, biomarker, clinical and social epidemiology. Nonetheless, the influential role of clinical trials within evidence-based medicine mean that our findings are of some importance despite our selection of a single field.

Mapping strings from published tables of clinical trial reports to UMLS using MetaMap resulted in dramatic improvements in our ability to automatically classify labels. This was consistent across the type of label (eg arm, time, outcome) and across the choice of classifier (eg random forest, neural network). Thus the integration of UMLS and MetaMap into our (and perhaps other's) tools have the potential to reduce the manual work involved in conducting systematic reviews.

Acknowledgements

We would like to thank Hebe Shedden and Yifan Shen for their help in compiling a substantial part of the training data used in this study. Furthermore, we would like to thank Wellcome Trust for funding this project.

References

1. Murad MH, Asi N, Alsawas M, Alahdab F. New evidence pyramid. BMJ Evidence-Based Medicine. 2016;21(4):125–127. Available from: https://ebm.bmj.com/content/21/4/125.

2. Higgins JP, Green S. Cochrane handbook for systematic reviews of interventions. vol. 4. John Wiley & Sons; 2011.

3. Marshall C, Sutton A. Systematic Review Tools; 2020 (accessed February 9, 2020). http://systematicreviewtools.com/index.php.

4. Jonnalagadda SR, Goyal P, Huffman MD. Automating data extraction in systematic reviews: a systematic review. Systematic reviews. 2015;4(1):78.

5. Millard LA, Flach PA, Higgins JP. Machine learning to assist risk-of-bias assessments in systematic reviews. International Journal of Epidemiology. 2015 12;45(1):266–277. Available from: https://doi.org/10.1093/ije/dyv306.

6. Tsafnat G, Glasziou P, Choong MK, Dunn A, Galgani F, Coiera E. Systematic review automation technologies. Systematic Reviews. 2014 Jul;3(1):74. Available from: https://doi.org/10.1186/2046-4053-3-74.

7. Garmonsway D. Unpivotr: Unpivot Complex and Irregular Data Layouts; 2019 (accessed February 9, 2020). https://github.com/nacnudus/unpivotr.

8. Company TSC. Databaker; 2018 (accessed February 9, 2020). https://github.com/sensiblecodeio/databaker.

9. Bian Zx, Shang Hc. CONSORT 2010 statement: updated guidelines for reporting parallel group randomized trials. Annals of internal medicine. 2011;154(4):290–291.

10. Phillippo DM, Dias S, Elsada A, Ades AE, Welton NJ. Population Adjustment Methods for Indirect Comparisons: A Review of National Institute for Health and Care Excellence Technology Appraisals. International Journal of Technology Assessment in Health Care. 2019;35(3):221–228.

11. Shah AS, Lee KK, McAllister DA, Hunter A, Nair H, Whiteley W, et al. Short term exposure to air pollution and stroke: systematic review and meta-analysis. bmj. 2015;350:h1295.

12. Zhang Y, Srimani PK, Wang JZ. Combining MeSH Thesaurus with UMLS in pseudo relevance feedback to improve biomedical information retrieval. In: 2016 IEEE International Conference on Knowledge Engineering and Applications (ICKEA); 2016. p. 67–71.

13. Gurulingappa H, Toldo L, Schepers C, Bauer A, Megaro G. Semi-Supervised Information Retrieval System for Clinical Decision Support. In: TREC; 2016. .

14. Hanlon P, Hannigan L, Rodriguez-Perez J, Fischbacher C, Welton NJ, Dias S, et al. Representation of people with comorbidity and multimorbidity in clinical trials of novel drug therapies: an individual-level participant data analysis. BMC Medicine. 2019 Nov;17(1):201. Available from: https://doi.org/10.1186/s12916-019-1427-1.

15. Bodenreider O. The Unified Medical Language System (UMLS): integrating biomedical terminology. Nucleic Acids Research. 2004 01;32(suppl_1):D267–D270. Available from: https://doi.org/10.1093/nar/gkh061.

16. Aronson AR, Lang FM. An overview of MetaMap: historical perspective and recent advances. Journal of the American Medical Informatics Association. 2010 05;17(3):229–236. Available from: https://doi.org/10.1136/jamia.2009.002733.

17. NLM. NLM: Semantic Types; 2016 (accessed February 9, 2020). https://www.nlm.nih.gov/research/umls/META3_current_semantic_types.html.

18. Urbano J, Lima H, Hanjalic A. Statistical Significance Testing in Information Retrieval: An Empirical Analysis of Type I, Type II and Type III Errors. In: Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval; 2019. p. 505–514.