Annotating Medical Forms Using UMLS

Victor Christen[1(✉)], Anika Groß[1], Julian Varghese[2], Martin Dugas[2], and Erhard Rahm[1]

[1] Department of Computer Science, Universität Leipzig, Leipzig, Germany
{christen,gross,rahm}@informatik.uni-leipzig.de
[2] Institute of Medical Informatics, Universität Münster, Münster, Germany
{Martin.Dugas,Julian.Varghese}@ukmuenster.de

Abstract. Medical forms are frequently used to document patient data or to collect relevant data for clinical trials. It is crucial to harmonize medical forms in order to improve interoperability and data integration between medical applications. Here we propose a (semi-) automatic annotation of medical forms with concepts of the Unified Medical Language System (UMLS). Our annotation workflow encompasses a novel semantic blocking, sophisticated match techniques and post-processing steps to select reasonable annotations. We evaluate our methods based on reference mappings between medical forms and UMLS, and further manually validate the recommended annotations.

Keywords: Semantic annotation · Medical forms · Clinical trials · UMLS

1 Introduction

Medical forms are frequently used to document patient data within electronic health records (EHRs) or to collect relevant data for clinical trials. For instance, case report forms (CRFs) ask for different eligibility criteria to include or exclude probands of a study or to document the medical history of patients. Currently, there are more than 180,000 studies registered on http://clinicaltrials.gov and every clinical trial requires numerous CRFs for data collection. Often these forms are created from scratch without considering existing CRFs from previous trials. Thus, there is a huge amount and diversity of existing medical forms until now, and this number will increase further. As a consequence, different forms can be highly heterogeneous impeding the interoperability and data exchange between different clinical trials and research applications.

To overcome such issues, it is important to annotate medical forms with concepts of standardized vocabularies such as ontologies [6]. In the biomedical domain, annotations are frequently used to semantically enrich real-world objects. For instance, the well-known Gene Ontology (GO) is used to describe molecular functions of genes and proteins [10], scientific publications in PubMed are annotated with concepts of the Medical Subject Headings (MeSH) [13], and concepts of SNOMED CT [5] are assigned to EHRs supporting clinical

N. Ashish and J.-L. Ambite (Eds.): DILS 2015, LNBI 9162, pp. 55–69, 2015.
DOI: 10.1007/978-3-319-21843-4_5

applications like diagnosis or treatment. These diverse use cases for annotations show that they can represent a variety of relationships between real-world objects improving semantic search and integration for comprehensive analysis tasks. In particular, ontology-based annotations of medical forms facilitate the identification of similar questions (items) and commonly used medical concepts. Well-annotated items can be re-used to design new forms avoiding an expensive re-definition in every clinical trial. Moreover, the integration of results from different trials will be improved due to better compatibility of annotated forms. Beside clinical trials, also other medical applications like routine documentation in hospitals can profit from form annotation. For instance, the fusion of two or more hospitals requires the integration of hospital data which will be less complex if data semantics are well-defined due to the use of ontology-based annotations.

The open-access platform *Medical Data Models* (MDM)[1] already aims at creating, analyzing, sharing and reusing medical forms in a central metadata repository [4]. Currently, MDM provides more than 9,000 medical form versions and over 300,000 items. Beside overcoming technical heterogeneities (e.g. different formats), MDM intends to semantically enrich the medical forms with concepts of the widely used Metathesaurus of the Unified Medical Language System (UMLS) [2], a huge integrated data source covering more than 100 different biomedical vocabularies. So far, medical experts could assign UMLS concepts to items of some medical forms in MDM, but many forms have no or only preliminary annotations. However, such a manual annotation process is a very time-consuming task considering the high number of available forms within and beyond MDM as well as the huge size of UMLS (>2.8 Mio. concepts). Thus, it is a crucial aim to develop automatic annotation methods supporting human annotators with recommendations.

The automatic annotation of medical forms is challenging since questions are written in free text, use different synonyms for the same semantics and can cover several different medical concepts. Moreover, the huge size of UMLS makes it difficult to identify correct medical concepts. So far, there has been some research on processing and annotation of different kinds of medical texts (e.g. [9,12,19]). However, (semi-) automatic annotation of medical forms has only rarely been studied (see Related Work in Sect. 5). We propose an initial solution to semi-automatically annotate medical forms with UMLS concepts and make the following contributions:

- We first discuss the challenges to be addressed for automatically annotating items in medical forms (Sect. 2).
- We propose an annotation workflow to automatically assign UMLS concepts to items of medical forms. The workflow encompasses three phases: a novel semantic blocking to reduce the search space, a matching phase and a post-processing phase employing a novel grouping method to finally select the correct annotations (Sect. 3).
- We evaluate our approaches based on reference mappings between MDM forms and UMLS. Results reveal that we are able to annotate medical forms in a

[1] www.medical-data-models.org/?locale=en.

largely automatic way. We further manually verify recommended annotations and present results for this semi-automatic annotation (Sect. 4).

Finally, we discuss related work in Sect. 5 and conclude in Sect. 6.

	Items		Associated UMLS concepts		
(a)	Patients with established **CRF (1)** as an indication for the **treatment (2)** of **anemia (3)**	○ **yes** ○ **no**	**1**	C0022661	Kidney Failure, Chronic
			2	C0039798	therapeutic aspects
			3	C0002871	Anemia
(b)	Patients who have had prior **recombinant erythropoietin (1)** treatment whose **anemia (2)** had **never responded (3)**	○ **yes** ○ **no**	**1**	C0376541	Recombinant Erythropoietin
			2	C0002871	Anemia
			3	C0438286	Absent response to treatment
(c)	**Ulcerating plaque (1)**	☐ **yes**	**1**	C0751634	Carotid Ulcer

Fig. 1. Example medical form items and associated annotations to UMLS concepts. (CRF = '*Chronic Renal Failure*' = '*Chronic Kidney Failure*').

2 Challenges

The automatic annotation of medical forms requires first of all the correct identification of medical concepts in form items. Figure 1 illustrates three annotated items: (a) and (b) ask for eligibility criteria for a study w.r.t. anemia, and item (c) asks for the abnormality 'ulcerating plaque' in the context of a quality assurance form. An item consists of the actual question and a response field or list of answer options. In our example, question (c) has one annotation, whereas (a) and (b) are annotated with three UMLS concepts. Thus, one form item can address several different aspects like diseases (e.g. CRF, anemia), treatments or a patient's response to a treatment. In the following we discuss general challenges that need to be addressed during the annotation process.

Natural Language Items: Typically, a form consists of a set of items. Questions can be short phrases like in item (c) or longer sentences written in free text (Fig. 1(a), (b)). It is a difficult task to correctly identify medical concepts in these natural language sentences. Moreover, the use of different synonyms complicate a correct annotation, e.g. in Fig. 1(a) '*CRF*' (= *Chronic Renal Failure*) needs to be assigned to C0022661 ('*Kidney Failure, Chronic*'). Simple string matching methods are not sufficient to generate annotations of high quality for medical form items. We will thus apply NLP (natural language processing) techniques such as named entity recognition and document-based similarity measures like TF/IDF to identify meaningful medical concepts that can be mapped to UMLS.

Complex Mappings: Every question can contain several medical concepts and one UMLS concept might be mapped to more than one question. In our example in Fig. 1 three UMLS concepts need to be assigned to questions (a) and (b) and the concept '*anemia*' occurs in both questions. By contrast, question (c) is only annotated with one concept. Thus, we might need to identify complex N:M mappings and do not know a priori how many medical concepts need to be tagged to one item. Conventional match techniques often focus on the identification of

1:1 mappings, but solely assigning one source concept to one target concept is a much simpler task. We thus need to develop sophisticated match techniques to correctly annotate items with several UMLS concepts.

Number and Size of Data Sources: There is high number of forms (e.g. 9000 only in MDM) that need be to annotated and every form can contain tens to hundreds of items. Moreover, UMLS Metathesaurus is a very large biomedical data source covering more than 2.8 million concepts. Matching 100 forms each comprising only 10 items to the whole UMLS would already require 2.8 billion comparisons. On the one hand this leads to serious issues w.r.t. memory consumption and execution time. On the other hand it is extremely hard to identify correct annotations in such a huge search space. It is thus essential to apply suitable blocking schemes to reduce the search space and restrict automatic annotation to the most relevant subset of UMLS.

Instances: Form items are not only characterized by medical concepts in the actual question but also by its possible instances or response options. Item answers have a data type (e.g. Boolean 'yes/no' in Fig. 1) and might be associated with value scales (e.g. between 1 and 5) or specific units (e.g. mg, ml). Often possible answers are restricted to a list of values (e.g. a list of symptoms). To improve the comparability of different forms, such instance information should be semantically annotated with concepts of standardized terminologies. In this paper, we focus on the annotation of item questions but see a correct annotation of answer options as an important future challenge.

In summary, the automatic identification of high-quality annotations for medical forms is a difficult task. However, studying automatic annotation is very useful to support human experts with recommendations. For a semi-automatic annotation process it is especially important to identify a high number of correct annotations without generating too many false positives. Thus, achieving high recall values is a major goal while precision should not be too low, since the number of presented recommendations should be manageable for human experts. Moreover, a fast computation of annotation candidates is desirable to support an interactive annotation process. To address these challenges, we present a workflow for semi-automatic annotation of medical forms in the following.

3 Annotation Workflow

Our annotation workflow semantically enriches a set of medical forms by assigning UMLS concepts to form questions. An annotation is an association between a question and an UMLS concept. UMLS concepts are identified by their *Concept Unique Identifiers* (CUI) and are further described by attributes like a preferred name or synonyms. To identify annotations for a given medical form F, we determine a mapping $\mathcal{M}$ between the set of form questions $F = \{q_1, q_2, ..., q_k\}$ and the set of UMLS concepts $UMLS = \{cui_1, cui_2, ..., cui_m\}$. The mapping covers a set of annotations and is defined as:

$$\mathcal{M}_{F,UMLS} = \{(q, cui, sim) | q \in F, cui \in UMLS, sim \in [0, 1]\}.$$

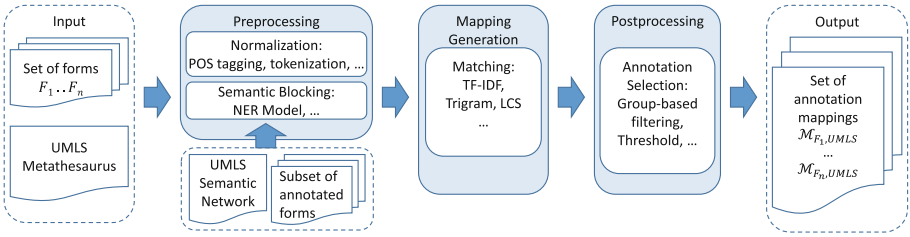

Fig. 2. Overview of the annotation workflow.

A question q in a form F is annotated with a concept *cui* from $UMLS$. Our automatic annotation method computes a similarity value *sim* indicating the strength of a connection. Greater *sim* values denote a higher similarity between the question and the annotated concept. Our annotation workflow (see Fig. 2) consists of three main phases that address the challenges discussed in Sect. 2. The input is a set of medical forms $F_1, \ldots, F_n$ each comprising a set of item questions as well as the UMLS Metathesaurus. During preprocessing we further use the UMLS Semantic Network and a subset of annotated forms. The output is a set of annotation mappings $\mathcal{M}_{F_1,UMLS}, \ldots, \mathcal{M}_{F_n,UMLS}$.

- In the *Preprocessing* phase we normalize input questions and UMLS concepts. Since a medical form is usually only associated to some domains covered by UMLS, we develop a novel semantic blocking technique to identify relevant concepts for the annotation generation. The approach is training-based and involves semantic types of UMLS concepts.
- In the *Mapping Generation* phase we identify annotations by matching the questions to names and synonyms of relevant UMLS concepts. We use a combination of a document retrieval method (*TF/IDF*) and classic match techniques (*Trigram, LCS (Longest Common Substring)*). By doing so we are able to identify complex annotation mappings for long natural language sentences as well as annotations to single concepts for shorter questions.
- During *Postprocessing* we remove probably wrong annotations to obtain a manageable set of relevant annotations for expert validation. Beside threshold selection we apply a novel group-based filtering to address the fact that questions might cover several medical concepts. For each question, we cluster similar concepts and keep only the best matching one per group.

Our workflow generates annotation recommendations which should be verified by domain experts since automatic approaches can not guarantee a correct annotation for all items. In the following, we discuss the methods in more detail.

3.1 Preprocessing

During preprocessing, we normalize the questions of a medical form as well as names and synonyms of UMLS concepts. In particular, we transform all string

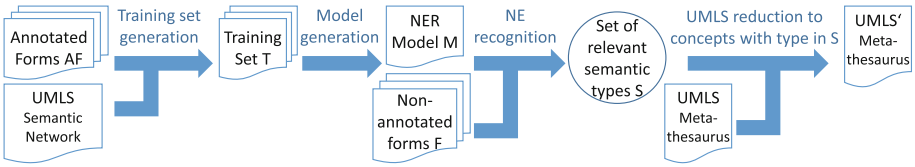

Fig. 3. Semantic blocking workflow. NER = Named Entity Recognition.

values to lower case and remove delimiters. We then remove potentially irrelevant parts of item questions. For instance, prepositions or verbs are typically part of natural language sentences, however they rarely cover information on medical concepts. We therefore apply a part-of-speech (POS) tagger[2] and keep only nouns, adjectives, adverbs and numbers/cardinals. We tokenize all strings into trigrams and word-tokens for the later annotation generation.

We further apply a semantic blocking to reduce the size of UMLS. UMLS Metathesaurus is a huge data source covering a lot of different subdomains. However, medical forms are usually only associated to a part of UMLS such that a comparison to the whole Metathesaurus should be avoided. We therefore aim at reducing UMLS by removing concepts that are probably not relevant for the annotation process. Our semantic blocking technique involves the UMLS Semantic Network. It covers 133 different semantic types and every UMLS concept is associated to at least one of the types. Our blocking technique follows a training-based approach and uses Named Entity Recognition (NER) to identify relevant semantic types for item questions. The general procedure is depicted in Fig. 3.

First, we build a training set T based on a subset of manually annotated forms AF. For each question in AF, we identify annotated named entities. Therefore, we compute the longest common part between a question and the names/synonyms of its annotated UMLS concepts. We then tag the identified question parts with the semantic types of the corresponding UMLS concept. Figure 4 illustrates an example for the training set generation. The given question is annotated with two UMLS concepts. The longest common part of the question and the concept *C0020517* is *Hypersensitivity*, while *C0015506* corresponds to the question part *Factor VIII*. Thus, *Hypersensitivity* is tagged with the semantic type of *C0020517* (*'Pathologic Function'*) and *Factor VIII* is labeled with *'Amino Acid, Peptide, or Protein'*. Based on the tagged training set T of forms AF we learn a NER-model M using the Open-NLP framework[3]. Our semantic blocking (see Fig. 3) then performs a named entity recognition using the model M to a non-annotated set of forms F. By doing so, we can recognize named entities for the questions in F and identify a set of relevant semantic types S. Finally, we reduce the UMLS Metathesaurus to those concepts that are associated to a semantic type in S and obtain the filtered $UMLS'$.

[2] http://nlp.stanford.edu/software/tagger.shtml.
[3] https://opennlp.apache.org/.

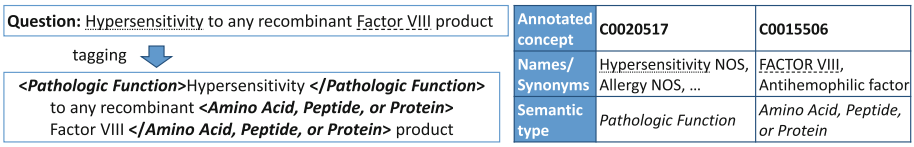

Fig. 4. Training set generation: example for tagging a question with semantic types.

3.2 Matching Phase

We generate annotation mappings between a set of medical forms $F_1, \ldots, F_n$ and the reduced $UMLS'$ using a combination of a document retrieval method (TF/IDF) and classic match techniques (*ExactMatch, Trigram, LCS*). These methods can complement each other such that we are able to identify complex annotation mappings for long natural language sentences as well as shorter questions covering only one concept. To generate annotations for each considered form, we compute similarities between all questions of a form and every concept in $UMLS'$. Note that, we tokenized strings during preprocessing. To enable an efficient matching, we encode every token (word or trigram), and compare integer instead of string values. Furthermore, we separate UMLS into smaller chunks and distribute match computations among several threads.

We apply for each question the three match methods. *Trigram* compares a question with concept names and synonyms, identifies overlapping trigram tokens, and computes similarities based on the Dice Metric. This is useful for shorter questions that slightly differ from the concept to be assigned. In our example in Fig. 1 the annotation for item (c) '*Ulcerating plaque*' needs to be assigned to the concept C0751634 ('*Carotid Ulcer*'). This correspondence can be identified by the synonym '*Carotid Artery Ulcerating Plaque*' of C0751634. Since there is only a partial overlap, it is feasible to identify the longest sequence of successive common word-tokens (LCS) between a question and a concept. LCS is also useful for complex matches when a question contains several medical concepts, e.g., 'recombinant erythropoietin' and 'anemia' in item (b) (Fig. 1).

Moreover, we use TF/IDF to especially reward common, but infrequent tokens between questions and UMLS concepts. For instance, in medical forms the token 'patient' occurs essentially more often then 'erythropoietin'. Thus, the computed similarity value should be higher for matches of rarely occurring, meaningful tokens compared to frequent tokens that appear in many questions and concepts. We compute tf-idf values for each token w.r.t. a question and an UMLS concept. The term frequency (tf) denotes the frequency of a token within the considered question or concept while the inverse document frequency (idf) characterizes the general meaning of a token compared to the total set of tokens. The tf-idf values are then used to compute the similarity between a token vector of the question and a token vector for names and synonyms of an UMLS concept. We choose a hamming-distance based measure to compare two token vectors. We compute distances between tf-idf values of two token vectors and normalize it based on the vector length. The normalized distance is converted

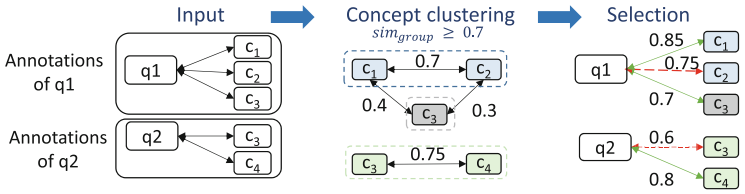

Fig. 5. Group-based filtering for two questions q_1 and q_2 and their annotations to concepts c_{1-4}. Uniformly colored concepts represent a group of similar concepts.

into a similarity value. We assign a smaller weight to the length of the longer vector to address cases, when one string consists of considerably more tokens than the other one, as this occurs for annotating long sentences. Thus, the measure does not penalize differences that are triggered by a differing vector length. High similarities between a shorter and a longer token vector can be achieved when a considerable number of meaningful tokens are contained in both vectors.

The generated annotation mappings are finally unified and similarities are aggregated by selecting the maximum *sim* value of a correspondence identified of several match methods to maximize the recall. Note that, we optimize the precision by performing the postprocessing phase. The match methods can identify overlapping results, but complement each other since they address different aspects of document and string similarity. We choose to adopt the three match methods in order to achieve a good recall by finding simple 1:1 as well as complex mappings for longer questions.

3.3 Postprocessing

Beside a simple threshold filtering, we apply a more sophisticated postprocessing step to filter the generated annotation mapping. Our aim is to identify all annotations to a question that are likely to be correct, i.e. to obtain high recall values. However, the result should not contain too many false positives in order to obtain a manageable set of recommendations to be presented to human experts. This is a complicated task when questions cover more than one medical concept, i.e. when we need to identify complex mappings. A simple approach would be to select the top k similar concepts for each question. However, it is possible that several annotations for the same medical concept in a question are among the top k. A top k selection could eliminate all annotations of medical concepts with lower *sim* values. We therefore apply a novel group-based filtering.

The group-based filtering first clusters concepts that are likely to belong to the same medical concept and then selects the most similar concept within a group. Figure 5 exemplarily describes the overall procedure for two questions q_1 and q_2 and their annotations to several concepts. Given a set of annotations for a question, we compute similarities between all UMLS concepts that are annotated to a question using trigram matching on concept names and synonyms. We than cluster concepts in one group if their similarity exceeds the required

sim_{group} threshold. In our example, we compare c_1, c_2 and c_3 for $q1$, and identify two groups $(\{c_1, c_2\}, \{c_3\})$. c_1 and c_2 are very similar ($sim_{group} \geq 0.7$), while c_3 builds an own group. Finally, the best annotation per group is selected to be included in the final mapping based on the annotation similarities from the previous phase. For instance, we remove (q_1, c_2) due to the lower annotation similarity within its group. Applying a simple top 2 selection would have preserved (q_1, c_2) but removed (q_1, c_3), although (q_1, c_3) is likely to be the best match for a different medical concept covered by question q_1. Using the group-based filtering, we are able to keep one annotation for each medical concept in a question and thus allow for complex annotation mappings. In the following, we evaluate the proposed annotation methods for real-world medical forms.

4 Evaluation

To evaluate the proposed annotation workflow we consider three datasets covering medical forms from the MDM portal [4]. Figure 6 gives an overview on the number of considered forms, the average number of items per form, the average number of tokens per item question and the average number of annotations per item. The first set of medical forms considers *eligibility criteria* (EC) that are used for patient recruitment in clinical trials w.r.t. diseases like Diabetes Mellitus or Epilepsy. The dataset covers 25 medical forms each comprising about 20 items on average. To recruit trial participants, a precise definition of inclusion and exclusion criteria is required, such that most questions are long natural language sentences (∼8 tokens on average) possibly covering several medical concepts. A correct identification of all annotations is very challenging for this dataset. Moreover, we consider medical forms for standardized *quality assurance* (*QA*) w.r.t. cardiovascular procedures. Since 2000 all German health service providers are obliged by law to apply these QA forms to prove the quality of their services [3]. The 23 QA forms contain about 49 items on average, but questions are shorter (∼3 tokens on average). We further consider a set of top items (TI) from the MDM portal. In [20], these items have been manually reduced to the relevant semantic question parts resulting in a low token number per question. We handle the 101 top items as one medical form. For UMLS, we only consider concepts that possess a preferred name or term, which is the case for ∼1 Mio. UMLS concepts. We involve names and synonyms of these UMLS concepts.

Dataset	Eligibility criteria (EC)	Quality assur. (QA)	Top Items (TI)
#forms	25	23	1
avg (#items)	20.5	48.8	101
avg(#tokens)	8.3	3.3	2.4

Fig. 6. Overview of the used datasets.

To evaluate the quality of automatically generated annotation mappings we use reference mappings between all considered MDM forms and UMLS. Our team consists of computer scientists as well as medical experts (two physicians), such that we could manually create the reference mappings based on expert knowledge. We compute precision, recall and F-measure for the annotation mappings of every medical form and show average values for the respective dataset (EC, QA or TI).

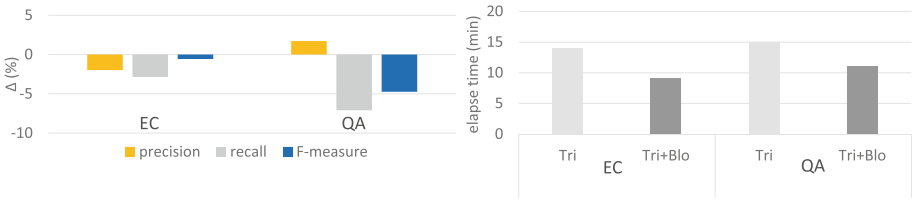

Fig. 7. Semantic blocking: quality differences (left) and execution time (right) for QA and EC, comparison of trigram without (*Tri*) and with semantic blocking (*Tri+Blo*).

Note, that the average of F-measures is not equal to a harmonic mean of average precision and average recall. Since a manual annotation is a difficult and time-consuming task, the initial reference mappings might not be complete. We therefore follow a semi-automatic annotation approach and manually validate the automatically generated annotations for the QA dataset to find further correct annotations (see Sect. 4.4). We first show evaluation results for EC and QA w.r.t. the methods of our annotation workflow (Sects. 4.1 and 4.2) and then give an overview on results for all datasets (Sect. 4.3).

4.1 Semantic Blocking

To evaluate our semantic blocking approach we measure the quality of the generated annotation mappings as well as matching execution times. We run experiments on an Intel i7–4770 3.4 GHz machine with 4 cores. Our aim is to reduce execution times without affecting the recall. The generation of training data is an important step for the semantic blocking. So far, we generated training data by randomly selecting half of the manually annotated datasets. Note, that the training sets have some bias since we consider a special type of medical forms, namely eligibility criteria and quality assurance forms. However, it is feasible to choose relevant semantic types in UMLS based on form annotations in the considered domain. It is an interesting point for future work to study the training set generation for the semantic blocking in more detail. We evaluate the impact of the semantic blocking using a basic trigram matching (*Tri*) without group-based filtering (threshold $t = 0.8$). Figure 7 shows quality differences and execution time results for QA and EC. The overall number of tokens was to small to apply the named entity recognition for TI. Applying the semantic blocking (*Blo*), UMLS could be reduced to ∼600.000 concepts. This results in good execution time reductions of 26–36 % for both datasets. However, we observe for each dataset a reduction of the quality of −0.5 % for EC and −4.73 % for QA. In both cases, the semantic blocking might be too restrictive by filtering some relevant UMLS concepts. A reason might be that the selection of our training set is not representative for the unannotated set of forms. We plan to further study the NER model generation to improve the blocking of UMLS concepts. Overall, our semantic blocking leads to good execution time reductions by fairly preserving recall values.

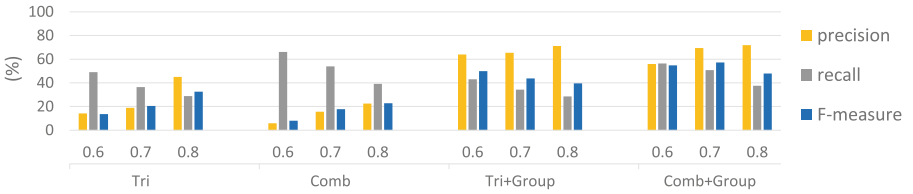

Fig. 8. Quality evaluation: comparison of trigram (*Tri*), combined matching (*Comb*) and group-based filtering (*Tri+Group* and *Comb+Group*) for QA forms.

4.2 Matching and Group-Based Filtering

We now generate annotation mappings by using a simple trigram matching (*Tri*), compare it to our combined match strategy based on TF/IDF, Trigram and LCS (*Comb*), and evaluate the impact of the group-based filtering (*Group*) for the QA dataset (see Fig. 8). We disable the blocking for this experiment and consider different threshold settings to evaluate the annotation quality. The combined match approach leads to higher recall values for all thresholds compared to trigram, since *Comb* detects a higher number of correct annotations compared to the single matcher. In particular, the combined matching achieves the best recall of ∼66 % ($t = 0.6$) which is 17 % more than for trigram. Trigram is more restrictive and results in higher precision values, such that the overall F-measure is better for low thresholds. In general, increasing the threshold improves the overall annotation quality due to a higher precision, e.g. for $t = 0.8$ the F-measure is 15 % higher than for $t = 0.6$ (*Comb*). However, we want to find a high number of correct annotations (high recall) during the annotation generation phase. Therefore, we then filter wrong correspondences using our group-based selection strategy (Fig. 8 right). This leads to significantly improved precision values and preserves the high recall. Since the combined match strategy results in higher recall values than the trigram matching, the F-measure values of the combined match strategy with the group-based selection (*Comb+Group*) are better than the trigram matching with the group-based selection (*Tri+Group*). For $t = 0.7$, we achieve the best average F-measure of 57 % for the QA dataset. Thus, the group-based filtering is a valuable selection strategy to remove wrong but keep correct annotations.

4.3 Result Summary

To give a result overview w.r.t. the annotation quality, we show average F-measure values for all datasets (EC, QA, TI) in Fig. 9. Since the semantic blocking decrease the quality, we compare the trigram matching (*Tri*), trigram matching with group-based filtering (*Tri+Group*) and combined matching with group-based filtering (*Comb+Group*) Due to a different amount of free text within the datasets, a uniform threshold not results in the best quality for each dataset, e.g., the TI dataset consists of mostly two words per item compared to the QA and EC dataset which have mostly more than three words per item. Therefore, we calculate the average for the thresholds 0.6, 0.7 and 0.8. The vertical lines indicate the

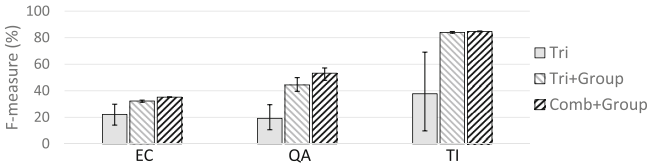

Fig. 9. Comparison of effectiveness of the combined matching strategy and group-based filtering approach for each dataset.

minimum and the maximum F-measure values for the underlying thresholds. We observe for each dataset an increasing of F-measure by applying group-based filtering compared to trigram matching. The precision increases heavily while most correct annotations are preserved. Since the combined matching strategy results in higher recall values than the trigram matching, the combination with group filtering leads to better F-measure values such that the difference of best F-measure values is ∼3 %(EC), ∼7 %(QA) and ∼0.5 %(TI). We achieve the best F-measure of ∼85 % for TI followed by ∼57 % for QA and ∼35 % for EC.

The automatic annotation of the EC dataset showed to be very difficult, since EC contains items with specifically long natural language sentences covering an unknown number of medical concepts. The annotation of QA forms leads to better results, but still needs improvement. For the annotation of the top items (TI) we achieve very good results. These items have been manually reduced to the relevant medical terms having a positive impact on the automatic assignment of UMLS concepts for this dataset. The semantic blocking was valuable to reduce executions times, and the combined match strategy together with the group-based filtering showed to produce very good results compared to a simple trigram matching. Overall, the automatic annotation of medical forms is a challenging task and requires future research, e.g. to further improve the recall.

4.4 Validation

We applied a semi-automatic annotation for the QA dataset by manually validating recommendations generated by our automatic annotation workflow. We computed mappings for all 23 QA forms using semantic blocking, combined matching and group-based filtering. For every form and question, we presented the expected correct annotations as well as our recommendations, and highlighted false negatives, false positives and true positives.

Medical experts could identify 213 new correct annotations out of the set of false positives. We further found 5 wrong annotations in the reference mappings based on our automatically generated recommendations. According to these findings we adapted the QA reference mappings leading to an average F-measure improvement of 9 % (for $t = 0.7$). Note, that we used these adapted QA reference mappings in the previous sections. Some of the recommendations were especially valuable. In particular, we found correct UMLS concepts for 38 so far not annotated questions, e.g.:

Question	Annotated concept
Heartbeat skipping (except for sleeping phases)	Dropped beats – heart (C0425591)
Ulcerating plaque	Carotid Ulcer (C0751634)
Malignant tumor (without curative treatment)	Malignant Neoplasms (C0006826)

The manual annotation of medical forms is difficult for curators. UMLS Metathesaurus is very huge, and even for medical experts it is hard to find a complete set of annotations. Sometimes it is difficult to decide for the correct concept, since UMLS contains similar concepts that might be suitable for the same medical concept in a question of a medical form [20]. Applying our automatic annotation workflow led to new correct annotations and could even indicate some false annotations. Our results point out the importance of semi-automatic annotation approaches. Combining manual and automatic annotation techniques (1) reduces the manual annotation effort and (2) leads to more complete and correct overall results. Semi-automatic annotation is especially relevant, since many medical forms are sparsely or not annotated. For instance, in MDM most items are only pre-annotated and need to be curated again. Part of the forms could not be annotated so far, and MDM is continuously extended by new non-annotated forms. Medical forms in MDM and can be semantically enriched by applying our annotation workflow in combination with expert validation.

5 Related Work

Our work on automatic annotation of medical forms is related to the areas of information retrieval [15] and ontology matching [8,17]. Both research fields have been studied intensively and provide useful methods to process free-text and match identified concepts to standardized vocabularies. Our system GOMMA [11] already allows for efficient and effective matching of especially large life science ontologies and can be a basis to align items with concepts of large ontologies. However, GOMMA does not provide methods to match free-text like form items.

In the medical domain, manual and automatic annotation methods have been studied to semantically enrich different kinds of documents. For instance, in [9] the authors clustered similar clinical trials by performing nearest neighbor search based on similarly annotated eligibility criteria. In [12] the application of a dictionary-based pre-annotation method could improve the speed of manual annotation for clinical trial announcements. The work in [19] focuses on the manual annotation process by presenting a semantic annotation schema and guidelines for clinical documents like radiology reports. The tool MetaMap [1] allows to retrieve UMLS concepts in medical texts based on information retrieval methods like tokenization and lexical lookup. In own initial tests by medical experts, MetaMap annotation results were not sufficient for our purposes. Moreover, there is evidence in the literature that MetaMap results are not fine-grained enough [14], contain too many spurious annotations [16] and do not cover mappings to longer medical terms [18]. In own previous work we already used manual annotations to compare and cluster different medical forms from the MDM

platform [7]. We further identified most frequent eligibility criteria in clinical trial forms and performed a manual annotation for these top terms [20].

Previous research showed the usefulness of semantic annotations for different kinds of clinical documents. However, the problem remains that annotations, in particular, for medical forms are only sparsely available. So far, there is no automatic annotation tool to support the semantic annotation of large medical form sets as provided by MDM. In contrast to previous work on document annotation in the medical domain, we here focus on the development of automatic annotation methods for medical forms. In particular, we use a novel blocking technique to reduce the complexity of UMLS as well as a combined match approach to cope with shorter as well as free-text questions. A novel group-based filtering allows to select the most likely set of question annotations to be presented for further manual validation.

6 Conclusions and Future Work

We proposed a workflow to (semi-)automatically annotate items in medical forms with concepts of UMLS. The automatic annotation is challenging since form questions are often formulated in long natural language sentences and can cover several medical concepts. The huge size of UMLS further complicates the annotation generation. We used a combined match strategy and presented a novel semantic blocking as well as a group-based filtering of annotations. We applied our methods to annotate real-world medical forms from the MDM portal and performed a manual validation of the generated annotations. Our methods showed to be effective and we could generate valuable recommendations. Medical experts can benefit from automatic form annotation since it reduces the manual effort and can prevent from missing or incorrect annotations.

We see several directions for future work. We will extend our annotation workflow to enable an adaptive matching which automatically determines the thresholds and select a set of appropriate match approaches by considering useful dataset characteristics. We further plan to annotate the instance information of items, e.g. their response options or data types. To test whether recommendations computed by different annotation methods can complement each other, we will integrate results of other tools like MetaMap. Furthermore, we plan to develop a reuse repository to facilitate the annotation of existing and creation of new medical forms based on well-annotated items.

Acknowledgment. This work is funded by the German Research Foundation (DFG) (grant RA 497/22-1, "ELISA - Evolution of Semantic Annotations").

References

1. Aronson, A.R., Lang, F.M.: An overview of MetaMap: historical perspective and recent advances. J. Am. Med. Inform. Assoc. **17**(3), 229–236 (2010)

2. Bodenreider, O.: The Unified Medical Language System (UMLS): integrating biomedical terminology. Nucleic Acids Res. **32**(suppl 1), D267–D270 (2004)
3. Bramesfeld, A., Willms, G.: Cross-Sectoral Quality Assurance. Â§137a Social Code Book V. Public Health Forum, pp. 14.e1–14.e3 (2014)
4. Breil, B., Kenneweg, J., Fritz, F., et al.: Multilingual medical data models in ODM format-a novel form-based approach to semantic interoperability between routine health-care and clinical research. Appl. Clin. Inf. **3**, 276–289 (2012)
5. Donnelly, K.: SNOMED-CT: The advanced terminology and coding system for eHealth. Stud. Health Technol. Inform. Med. Care Compunetics **3**(121), 279–290 (2006)
6. Dugas, M.: Missing semantic annotation in databases. The root cause for data integration and migration problems in information systems. Methods Inf. Med. **53**(6), 516–517 (2014)
7. Dugas, M., Fritz, F., Krumm, R., Breil, B.: Automated UMLS-based comparison of medical forms. PloS one **8**(7) (2013). doi:10.1371/journal.pone.0067883
8. Euzenat, J., Shvaiko, P.: Ontology Matching, vol. 18. Springer, Heidelberg (2007)
9. Hao, T., Rusanov, A., Boland, M.R., et al.: Clustering clinical trials with similar eligibility criteria features. J. Biomed. Inform. **52**, 112–120 (2014)
10. Huntley, R.P., Sawford, T., Mutowo-Meullenet, P., et al.: The GOA database: gene Ontology annotation updates for 2015. Nucleic Acids Res. **43**(D1), D1057–D1063 (2015)
11. Kirsten, T., Gross, A., Hartung, M., Rahm, E.: GOMMA: a component-based infrastructure for managing and analyzing life science ontologies and their evolution. J. Biomed. Semant. **2**(6), 1–24 (2011)
12. Lingren, T., Deleger, L., Molnar, K., et al.: Evaluating the impact of pre-annotation on annotation speed and potential bias: natural language processing gold standard development for clinical named entity recognition in clinical trial announcements. J. Am. Med. Inform. Assoc. **21**(3), 406–413 (2014)
13. Lowe, H.J., Barnett, G.O.: Understanding and using the medical subject headings (MeSH) vocabulary to perform literature searches. J. Am. Med. Assoc. (JAMA) **271**(14), 1103–1108 (1994)
14. Luo, Z., Duffy, R., Johnson, S., Weng, C.: Corpus-based approach to creating a semantic lexicon for clinical research eligibility criteria from umls. AMIA Summits Transl. Sci. Proc. **2010**, 26–30 (2010)
15. Manning, C.D., Raghavan, P., Schütze, H.: Introduction to Information Retrieval, vol. 1. Cambridge University Press, Cambridge (2008)
16. Ogren, P., Savova, G., Chute, C.: Constructing evaluation corpora for automated clinical named entity recognition. In: Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC), pp. 3143–3150 (2008)
17. Rahm, E.: Towards large-scale schema and ontology matching. In: Bellahsene, Z., Bonifati, A., Rahm, E. (eds.) Schema Matching and Mapping. Data-Centric Systems and Applications, pp. 3–27. Springer, Berlin (2011)
18. Ren, K., Lai, A.M., Mukhopadhyay, A., et al.: Effectively processing medical term queries on the UMLS Metathesaurus by layered dynamic programming. BMC Med. Genomics **7**(Suppl 1), 1–12 (2014)
19. Roberts, A., Gaizauskas, R., Hepple, M., et al.: Building a semantically annotated corpus of clinical texts. J. Biomed. Inform. **42**(5), 950–966 (2009)
20. Varghese, J., Dugas, M.: Frequency analysis of medical concepts in clinical trials and their coverage in MeSH and SNOMED-CT. Methods Inf. Med. **53**(6), 83–92 (2014)