Information Extraction from Pathology Reports in a Hospital Setting

David Martinez
NICTA & University of Melbourne
Dept. of Computer Science and Software Engineering, University of Melbourne
3010, Australia
david.martinez@nicta.com.au

Yue Li
NICTA & University of Melbourne
Dept. of Computer Science and Software Engineering, University of Melbourne
Victoria 3010, Australia
yolandali.nz@gmail.com

ABSTRACT

As more health data becomes available, information extraction aims to make an impact on the workflows of hospitals and care centers. One of the targeted areas is the management of pathology reports, which are employed for cancer diagnosis and staging. In this work we integrate text mining tools in the workflow of the Royal Melbourne Hospital, to extract information from pathology reports with minimal expert intervention. Our framework relies on coarse-grained annotation (at document level), making it highly portable. Our evaluation shows that the kind of language used in these reports makes it feasible to extract information with high precision and recall, by means of state-of-the-art classification methods, and feature engineering.

Categories and Subject Descriptors

J.3 [**Computer Applications**]: Life and Medical Sciences

General Terms

Algorithms

1. INTRODUCTION

As new technologies for health care are implemented, hospitals have to efficiently manage large amounts of data for each patient in their care. Pathology reports are one of those sensitive pieces of data, used to record information about cells and tissues of a patient. These reports are employed for cancer diagnosis and staging (describing the extent of cancer within the body), and to help to determine the treatment options. Having extensive and precise pathology reports would allow us to perform optimal clinical planning, offer treatments based on the best evidence, and identify the major prognostic factors across different populations. However, the current situation makes the implementation of this kind of analysis unfeasible, because of the lack of a structured and connected network of pathology reports.

One of the major stumbling blocks for the better management of this information is that pathology reports are created in natural language, and this makes it difficult to obtain the structured knowledge required for high-level analysis. Only in some cases the information is converted into structured format, and even then, we face the problems of different standards, data entry errors, and incomplete entries [7]. For this reason, the Australian National Cancer Data Strategy has identified the collection of population-based structured data as a fundamental strategical problem[1].

This scenario is promising for text mining research, because tools that can perform well in this space are likely to make an impact in the way health information is stored and used. The main challenges for developing these kinds of tools are the need of expert domain knowledge (by means of hand-coded rules or manual annotation), and the portability to different subdomains. However a recent systematic literature review has shown limitations of existing text mining tools for the biomedical domain, which tend to be context-dependent and not readily portable [10].

In this paper we try to overcome the portability limitations of text mining tools by automatically learning from the data generated in the workflow of the Royal Melbourne Hospital, with minimal expert intervention. Our framework relies on coarse-grained annotation, Machine Learning (ML), feature engineering, and error analysis in order to extract information for a set of categories in cancer pathology reports. Our feature set includes lemmatisation, different lexical databases (*SNOMED* and *UMLS*), and negation detection. We evaluate our performance using a reference corpus, and compare with other state-of-the-art results when possible. Differently to previous work in this area, we do not rely on domain experts for fine-grained annotation of examples or rules.

Our empirical evaluation will give us insight to build a classification framework for cancer pathology, and for the best methods to implement a practical system for the hospital. Our aim is to integrate a recommendation system in the hospital workflow, where the main features in the text will be highlighted, and where practitioners will accept or correct the system's suggestions. This will make the management of pathology reports easier, and it will help to avoid data input errors, which can have serious consequences in this domain. Another advantage of this approach is that the interactions of the users will be saved, and used to improve the underlying models. In order for this system to be of practical use, we first need to evaluate the quality of predictions over a sample of real data, and this is our goal in this paper.

Text mining from pathology reports has unique characteristics,

CIKM'11, October 24–28, 2011, Glasgow, Scotland, UK.
Copyright 2011 ACM 978-1-4503-0717-8/11/10 ...$10.00.

[1] http://www.canceraustralia.gov.au/cancer-data/cancer-data-improve-cancer-survival

due to the very specific terminology, and the short and semantically-dense content. These can be advantages for text mining, but there are few references in the literature to this domain, because of the lack of pathology reports freely available for research. Pathology reports contain sensitive material, and even after de-identification it is not easy to make them widely available. We obtained a test collection through our collaboration with the Royal Melbourne Hospital, and one of our goals is to provide insight into the type of text that can be found in this domain, and the type of features that contribute to automatic approaches. We expect our contribution to be useful for the implementation of tools for hospitals, and also to new datasets that are being made public, such as the repository of de-identified clinical reports available for research from the University of Pittsburgh NLP group (BLULab)[2]. These clinical records include 2,877 surgical pathology reports, which will be very valuable for researcher in the area. The whole collection of 101,711 clinical reports will be used in the upcoming TREC-MED 2011 challenge [3], which shows the growing interest on this area.

2. BACKGROUND

Our domain of application is colorectal cancer, and we study pathology reports from the Royal Melbourne Hospital. In the current hospital workflow, these reports are written in natural language, and then manually converted into standard forms to represent the structured knowledge. This is a complex task, involving heterogeneous classes with a large set of possible values, both numeric and nominal. It requires clinicians to carefully read each report searching for the relevant information. We will analyse the contribution of text mining to this process, requiring minimal expert intervention to build our classifiers.

The core information contained in pathology reports refers to the staging and location of the cancer, and we will build classifiers to predict 6 categories that represent these factors, which are shown in Table 1. These categories can be seen as 6 multiclass problems, where the goal is to identify a unique goldstandard label.

Staging information has been the target of previous work in the literature, and this will allow us to compare our performance to other approaches. Early work on lung cancer staging was conducted by [8] using Support Vector Machines. Their initial experiments showed the difficulty of the primary tumour stage detection (T), with a top accuracy of 64%. In a follow-up paper they explored richer annotation, and a combination of ML and rule-based post-processing [7]. They performed fine-grained annotation of stage details for each sentence in order to build their system, and they observed improvements over a coarse-grained (document-level) multiclass classifier. However, the authors explain that the annotation cost is high, and the performance for “Stage T” was still low (65% accuracy). In their latest work [9] rely heavily on the *SNOMED* (Systematized Nomenclature of Medicine - Clinical Terms)[4] concepts and relationships to identify the relevant entities. They argue that this approach is more portable than fine-grained annotation, although there is a loss in accuracy with respect to their best ML approach, and it requires involvement from the experts. They report accuracies of 78% for ML, and 73% for rules.

Another relevant work in this area was conducted by [3], where the authors defined an extensive knowledge model for pathology reports. Their model was linked to hand-built inference rules to process unseen data. They reported high performance over 9 target classes for a hand-annotated 300-report dataset. This system seeks to build a representation of the domain by relying on human experts, and its portability to a different dataset or class-set could be problematic. The classes they evaluate on are not present in our dataset.

Category	Description	Type
Tumour site	Location of Tumour	Nominal
Nodes examined	Number of nodes analysed	Numeric
Nodes positive	Number of nodes found positive	Numeric
Staging T	Primary tumour staging	Nominal
Staging N	Regional lymph nodes staging	Nominal
Staging ACPS	Australian clinico-pathological stage	Nominal

Table 1: Categories and description.

3. EXPERIMENTAL SETTING

3.1 Dataset

For our analysis we rely on a corpus of 217 de-identified clinical records from the Royal Melbourne Hospital. These records were first written in natural language, and then structured information concerning 36 fields of interest was introduced to the Colorectal Cancer Database of the hospital, by means of a standarised form. The written records tend to be brief, usually covering a single page, and semantically dense. Each report contains three sections describing different parts of the intervention: macroscopic description, microscopic description, and diagnosis. All sections contain relevant information for the database.

The annotation we have is coarse-grained (at document level), and we do not know which parts of the document contain the relevant information for each class of interest. However, in some cases we can parse the text and identify the target labels to find the exact surface representation. For instance, if the target class is “Nodes examined”, and the goldstandard value is “9”; then we can search for the strings “9” and “nine” and create a fine-grained training instance. This approach will be noisy, because there may be other uses of the number in the document that do not refer to the target class, and therefore we manually correct the outcomes. We used this method to build sentence-annotated data for numeric classes and non-zero values only, while for nominal classes we rely on the document-level annotation.

3.2 Target classes

The selection of 6 target classes was made according to their importance for determining the staging and location of the cancer. The first category we analyse is “Tumour site”, which refers to the location of the tumour. Our dataset contains 11 different locations, such as “Rectosigmoid”, “Caecum”, etc. There is a balanced distribution of records for this class, with the most frequent location having 40 instances, and the less frequent having 6 records[5].

The next target categories are “Nodes examined” and “Nodes positive”, which are numeric categories that contain information about the number of lymph nodes studied and found positive respectively. These classes, and especially “Nodes positive” represent a crucial part of the knowledge contained in the report.

The last three categories refer to the staging of the tumour with regards to the TNM cancer staging system. This protocol describes the extent of cancer in a patient’s body, and it has been studied previously in the text mining literature (cf. Section 2). The first staging category is staging T, which refers to the primary tumour staging, and we can see its values and document frequencies at the

[2] http://nlp.dbmi.pitt.edu/report_repository.html

[3] http://groups.google.com/group/trec-med

[4] http://www.nlm.nih.gov/research/umls/Snomed/snomed_main.html

[5] Here we calculate the distribution for the 203 instances we use in cross-validation, and not for the held-out data (cf. Section 3.5).

Staging T							
Code	0	IS	1	2	3	4	X
Frequency	8	0	15	22	127	20	8
Staging N							
Code	0	1	2	X			
Frequency	113	46	27	14			
Staging ACPS							
Code	A	B	C	O	X	Y	
Frequency	30	73	38	0	0	6	

Table 2: Values and frequencies for staging categories. See http://en.wikipedia.org/wiki/TNM_staging_system and [4] for a full description of the codes.

top of Table 2. This category has a strong bias for code "3", and this will affect classifier performance.

The next main staging category, "Staging N", describes regional lymph nodes that are involved, and its codes and frequencies are given in Table 2, with a strong bias towards the "0" value. The last piece of information of the TNM system is "Staging M", which refers to distant metastases. This category is not well represented in our dataset, with a strong bias towards the negative label, and few textual clues to identify it. We decided to leave this category from the analysis for this work, and instead we work on "Staging ACPS", which is the Australian Clinico-Pathological Staging system [4]; the possible values for this class (and their frequencies) are shown at the bottom of Table 2. We can see that the main 4 labels are well represented in the data.

3.3 Knowledge sources and feature representation

In order to better represent the document for information extraction, we rely on diverse tools and knowledge sources that will allow us to test different aspects of the text. We will introduce each feature type in turn, and as an example we show in Table 3 the feature representation for the sentence *"No evidence of metastatic tumour is seen in any of 11 lymph nodes"* from our corpus.

The most basic representation we apply is the widely-used bag-of-words (*BOW*), which treats each token in the document as a separate feature, without including information about the relationships and ordering of the tokens. We extract this feature by tokenising the text with an in-house tokeniser based on regular expressions to separate words, numbers, and punctuation. We also use regular expressions to convert the textual mentions of numbers into their numeric representation. Finally, we include the binary feature "NUMBER" to indicate whether there is a numeric reference in the text. We do not rely on stopwords.

We also analyse the use of lemmas instead of *BOW*, and for that we rely on the GENIA tagger [11], which provides information about lemmatisation, part-of-speech, entities, and phrase chunking. For our experiments we only rely on lemma information, since the entities belong to a different domain.

For semantic features we employ the Metathesaurus from the *Unified Medical Language System* (*UMLS*) [6], which provides a set of ontologies for the biomedical domain with semantic relationships between terms (e.g. synonyms and hypernyms). We use this resource by parsing each sentence in the document with the MetaMap analyser [1] (version 2009), and default parameters. As a result we obtain concept identifiers, which map the text into the ontological concepts. This allows us to identify connections between different word forms of the same concept (e.g. "disease" and "disorder").

The concepts in the Metathesaurus originate from terminologies used in different areas[6], and due to the diversity of this database, we also explored using a subset that comprises a medical terminology: *SNOMED* [7]. In this case we also rely on MetaMap to extract the concepts, but we filter out those that are not present in *SNOMED*.

Feature type	Features
BOW	*No*, *evidence*, *of*, *metastatic*, *tumour*, *is*, *seen* *in*, *any*, *of*, *11*, *lymph*, *nodes*, *NUMBER*
Lemma	*No*, *evidence*, *of*, *metastatic*, *tumour*, *be*, *see* *in*, *any*, *of*, *11*, *lymph*, *node*, *NUMBER*
UMLS	C0332120:Evidence, C0027627:Tumor (metastatic) C0205397:Seen, C0024204:lymph nodes
SNOMED	Tumor (metastatic)
ConcNeg	Tumor (metastatic)

Table 3: Feature representation for *"No evidence of metastatic tumour is seen in any of 11 lymph nodes"*; features separated by ",".

Finally, we also identify phrases that are negated in the text, and for that we rely on the tool Negex [2], which indicates whether a clinical condition has been negated by relying on regular expressions in context. We use this tool both over *UMLS* and *SNOMED* outputs, and the negated concepts are marked accordingly in the feature representations; we refer to this feature as "ConcNeg". For combining different feature sets we simply join their representations into a single vector.

3.4 Method

Our classification methods will be initially based on ML and feature selection, with a second phase for difficult categories, where we will perform error analysis and identify salient features manually. Another goal of the second step is to provide insight on the cost-benefit of performing fine-grained annotation and of building manual rules for classification.

We develop two types of models, depending on the target category. For nominal classes we define a straightforward document classification problem, where the features are extracted from the full document for classification. On the other hand, for the numeric classes we apply a two-step process, and for that we rely on the sentence annotation that we build semi-automatically (as explained in Section 3.1). The reason for this is that numeric values are easier to manually-annotate than the nominal classes at sentence level, and the information is usually given at a single sentence in the document, without repetition.

We define the process for numeric classes as follows. The first step is to build sentence classifiers for each class, by using the sentence-level annotations. Note that only numbers different to zero are detected, and the zero label is assigned only in cases where the sentence classifiers fail to identify any number. After the model identifies the positive sentences, the numeric values are extracted, and the number closest to the median of the class (in training data) is assigned. In the cases where no positive sentences are identified the number zero is assigned.

Each of our models is tested with a suite of classifiers provided by the Weka toolkit [12]. We chose a set of classifiers that has been widely used in the text mining literature in order to compare their performances over our dataset: Naive Bayes (*Naive Bayes*), Support Vector Machines (*SVM*), and AdaBoost (*AdaBoost*). We use the default parameter settings of Weka (version 3-6-2) for each of the classifiers. As underlying classifier for *AdaBoost* we rely on simple Decision Stumps (one-level decision trees).

We also explore the contribution of feature selection to the classification performance. We apply a correlation-based feature subset selection method, which considers the individual predictive ability

[6] http://www.nlm.nih.gov/pubs/factsheets/umlsmeta.html

[7] http://www.nlm.nih.gov/research/umls/Snomed/snomed_main.html

Features	*MC*	*Naive Bayes*	*SVM*	*AdaBoost*
BOW	19.7	40.4	38.4	34.0
BOW (FS)	19.7	54.2	44.8	34.0
Lemma	19.7	39.9	37.4	34.0
Lemma (FS)	19.7	51.7	43.8	34.0
BOW and *UMLS*	19.7	40.9	36.5	34.0
BOW and *UMLS* (FS)	19.7	56.7	52.2	34.0
BOW, *UMLS* and ConcNeg	19.7	39.4	36.5	32.0
BOW, *UMLS* and ConcNeg (FS)	19.7	55.7	50.2	32.0
SNOMED and *UMLS*	19.7	40.9	36.0	34.0
SNOMED and *UMLS* (FS)	19.7	**58.1**	53.7	34.0
SNOMED, *UMLS* and ConcNeg	19.7	39.9	35.5	32.0
SNOMED, *UMLS* and ConcNeg (FS)	19.7	55.7	50.2	32.0

Table 4: Machine Learning performances for the "Tumour Site" category (F-score). The best result is given in bold.

of each feature and the redundancy of each subset [5]. We relied on Weka's implementation of this technique, and used Best-First search, with a cache-size of one element, and 5 levels of backtracking.

3.5 Evaluation

In order to evaluate the different models and classifiers we report F-score, which is the harmonic mean of precision and recall. We apply balanced F-score in all experiments, giving equal weight to precision and recall, and we micro-average the results over the different class values. 10-fold cross-validation is used in all our ML experiments. We kept 14 random records from the full collection of 217 as held-out data, in order to separately evaluate the performance of manually-built rules; therefore the cross-validation experiments are performed over the remaining 203 instances. As baseline we rely on the Majority Class (*MC*) classifier, which assigns the most frequent class from training data to all test instances. In case of ties the value is chosen randomly among those tied.

4. RESULTS

4.1 Tumour site

The first target category is "Tumour site", and the results of the different ML systems and feature sets are given in Table 4, with and without feature selection. We can see that the best performance only reaches 58.1% F-score, which seems insufficient for an application, even if it is well above the *MC* baseline. Regarding the different systems, the best results are obtained when using *Naive Bayes*, and when applying *BOW*, *UMLS*, and feature selection.

We analysed the causes of the low performance, and found that this class would be difficult even for non-expert human annotators, because the target values are rarely explicitly written in the text. We searched for the goldstandard label in the reports, and this only retrieved 45% of matches, and in particular, for the value "Hepatic flexure" there is only a single mention in the 10 reports that are annotated as such. We studied the top features chosen in the feature selection approach, by measuring their entropy and frequency. This showed that only a few of the target labels had features with low entropy and high frequency, and this makes it difficult to classifiers to generalise. All top features were of type *BOW*.

For this category, we have to conclude that we are not able to deploy a practical extraction system with the current data. However, we can provide aids to the manual process at the hospital, where top features are highlighted, and the top predictions are presented.

4.2 Nodes

We evaluate in this section both the the number of nodes that have been examined, and the number of nodes found positive. We start with the examined nodes, and present the result of ML approaches in Table 5. We can see that the *MC* baseline is very low in this case (7.4%), and ML approaches produce F-scores around the 70% mark, which could be useful for the implementation of a prediction system at the hospital. In this case the best approach is *SVM*, with *BOW* and *UMLS*.

Features	*MC*	*Naive Bayes*	*SVM*	*AdaBoost*
BOW	7.4	68.4	68.4	67.6
BOW (FS)	7.4	33.6	22.2	25.9
Lemma	7.4	67.1	68.2	63.4
Lemma (FS)	7.4	34.9	20.4	24.2
BOW and *UMLS*	7.4	63.7	**70.1**	68.0
BOW and *UMLS* (FS)	7.4	33.1	21.7	25.9
BOW, *UMLS* and ConcNeg	7.4	64.3	68.6	57.0
BOW, *UMLS* and ConcNeg (FS)	7.4	23.6	13.2	11.3
SNOMED and *UMLS*	7.4	62.6	69.1	68.0
SNOMED and *UMLS* (FS)	7.4	33.2	25.3	27.3
SNOMED, *UMLS* and ConcNeg	7.4	64.7	68.6	59.1
SNOMED, *UMLS* and ConcNeg (FS)	7.4	20.7	13.1	11.4

Table 5: Machine Learning performances for the "Nodes Examined" category (F-score). The best result is given in bold.

Our next step was to perform error analysis over our best classifier to find the main causes of error. We observed that our binary sentence classifier was biased towards the negative class, and therefore the recall for identifying positive sentences was low. However, the major problem was that when the target sentence was correctly identified, in most cases both "nodes examined" and "nodes positive" were provided in the same sentence (e.g. *2 out of 9 nodes were found positive*), and our simple strategy of using the median was not good enough to identify the right number.

In order to implement a practical system for the hospital, we then manually analysed the top features selected by the correlation-based feature subset selection method, and identified that features "lymph", "tumour", and "metastatic" had a strong correlation with the positive class. We focused in the top feature, and our manual analysis of the sentences allowed us to observe a frequent pattern in the way "nodes examined" was given in the text. We devised two simple rules to apply this knowledge for improved performance:

- if the "lymph" word is found in the sentence, select highest number (maximum 50)
- if the "lymph" word is found, and "further <NUMBER>" or "other <NUMBER>" patterns are found, add # to current count

This simple approach achieved high performance (93.8% F-score), which would be very useful for a practical system. However, there is risk of overfitting the data, and we rely on the 14 held-out instances for validation of the rule-based approach. We achieve an F-score of 85.7%, which shows that this method could be useful for the implementation of a practical system. The rules could be applied directly to make predictions, and this approach could be used to collect more data from clinicians, who will validate or reject the recommendation.

The next target category is "nodes positive". The results of our ML systems are given in Table 6. In this case the *MC* baseline is high (64% F-score), because most records found zero positive nodes, but the ML approaches are able to improve this number to reach 80% F-score (using *SVM* with lemmas). We then perform error analysis to observe the ways this type of information is given in the reports. The problems are similar to those observed for "nodes examined", with low recall and difficulty extracting the right number from the target sentence. We find that in most cases the number of positive nodes is given together with the examined nodes, and we define the following rule to identify the number of positive nodes:

- if the "lymph" word is found in the sentence, and two numbers (smaller than 51) occur, then select the smallest number

Features	*MC*	*Naive Bayes*	*SVM*	*AdaBoost*
BOW	64.0	62.3	77.4	67.5
BOW (FS)	64.0	68.0	68.0	67.8
Lemma	64.0	57.8	**80.1**	74.1
Lemma (FS)	64.0	72.0	71.1	69.7
BOW and *UMLS*	64.0	53.7	76.1	67.5
BOW and *UMLS* (FS)	64.0	68.5	68.0	68.3
BOW, *UMLS* and ConcNeg	64.0	57.6	77.8	69.1
BOW, *UMLS* and ConcNeg (FS)	64.0	72.2	69.3	68.8
SNOMED and *UMLS*	64.0	55.3	77.1	67.5
SNOMED and *UMLS* (FS)	64.0	73.8	68.8	68.3
SNOMED, *UMLS* and ConcNeg	64.0	57.8	78.8	69.1
SNOMED, *UMLS* and ConcNeg (FS)	64.0	71.7	69.3	68.8

Table 6: Machine Learning performances for the "Nodes Positive" category (F-score). The best result is given in bold.

Features	*MC*	*Naive Bayes*	*SVM*	*AdaBoost*
BOW	62.6	62.1	67.5	63.1
BOW (FS)	62.6	80.8	71.9	63.1
Lemma	62.6	63.1	67.5	61.6
Lemma (FS)	62.6	77.8	75.4	61.6
BOW and *UMLS*	62.6	62.6	66.0	61.6
BOW and *UMLS* (FS)	62.6	79.8	75.9	61.6
BOW, *UMLS* and ConcNeg	62.6	63.1	64.5	60.6
BOW, *UMLS* and ConcNeg (FS)	62.6	**82.3**	73.9	60.6
SNOMED and *UMLS*	62.6	63.1	65.5	61.6
SNOMED and *UMLS* (FS)	62.6	81.3	75.9	61.6
SNOMED, *UMLS* and ConcNeg	62.6	62.6	64.5	60.6
SNOMED, *UMLS* and ConcNeg (FS)	62.6	80.8	72.9	60.6

Table 7: Machine Learning performances for the "Staging T" category (F-score). The best result is given in bold.

In this case there is another improvement over the ML system, with an F-score of 85.2%. We also test this rule over the held-out data, and again obtain an F-score of 85.7%. However, in this case rules may not be necessary for "Nodes Positive", since the ML system performs above 80% F-score, and there is less risk of overfitting.

4.3 Staging

We start this section by predicting the category "Staging T", which refers to the progression of the primary tumour. The results of the different ML approaches are given in Table 7. We can see that the results of *Naive Bayes* with feature selection reach a promising 80% F-score (with *BOW*, *UMLS*, and ConcNeg), which compares positively with the state of the art. Recent work on the detection of "Staging T" over 710 reports of lung cancer patients presents accuracies of 73% using a rule-based approach [8], and 78% using Machine Learning [9].

The next staging category is "Staging N", which refers to the staging of regional nodes. In this case we do not apply the document classifier directly, but we rely on the predictions from "Nodes positive" to infer the label of "Staging N" using deterministic rules (e.g. if "Nodes Positive" is between 1 and 3, then assign code 1; cf. Table 2). Our first step is to rely on the outputs of the ML classifiers to apply the rules, and the results are shown in Table 8. We can see that the best F-score is 73.2%, which is above the baseline, but with a high error rate. The best approach is *SVM* with lemmas.

Apart from the ML predictions, we can also rely on the rule-based system to obtain values of positive nodes, and then use the meta-rules on top of these. We show the results of this approach in Table 9, together with the results of applying rules on top of "Nodes Positive" goldstandard results. We do this in order to illustrate that the application of the rules does not give us 100% F-score, but 91.1% instead. The main reason for this is the presence of the code "X", which is indistinguishable from "0" in our approach. However, the error analysis also allowed us to discover two mismatches in the annotation. In one of the cases the text in the pathology report is incomplete, and the information that one positive node was detected cannot be found; the other mismatch occurs when the text clearly specifies that two nodes were found positive, but this information is not recorded in the form. These examples illustrate that text mining techniques can be helpful to identify inconsistencies in the way the information is recorded.

Features	*MC*	*Naive Bayes*	*SVM*	*AdaBoost*
BOW	57.1	56.4	72.0	64.0
BOW (FS)	57.1	60.2	61.7	61.6
Lemma	57.1	53.4	**73.2**	68.5
Lemma (FS)	57.1	63.3	63.0	63.0
BOW and *UMLS*	57.1	57.1	57.1	57.1
BOW and *UMLS* (FS)	57.1	60.7	61.7	62.1
BOW, *UMLS* and ConcNeg	57.1	57.1	57.1	57.1
BOW, *UMLS* and ConcNeg (FS)	57.1	66.2	63.4	62.6
SNOMED and *UMLS*	57.1	57.1	57.1	57.1
SNOMED and *UMLS* (FS)	57.1	66.0	62.2	62.1
SNOMED, *UMLS* and ConcNeg	57.1	57.1	57.1	57.1
SNOMED, *UMLS* and ConcNeg (FS)	57.1	65.7	63.4	62.6

Table 8: Machine Learning performances for the "Staging N" category (F-score). The best result is given in bold.

Base classifier ("nodes positive")	F-score
Rule-based	81.3
Goldstandard	**91.1**

Table 9: Performances for the "Staging N" category (F-score) using rules over "nodes positive". The best result is given in bold.

Thus, as illustrated in Table 9, the performance of our rule-based classifier is 81.1% F-score. Regarding the state of the art, previous work on "Staging N" has reported accuracies of 87% for a rule-based approach [8], and 82% for Machine Learning [9]. Similarly, we obtain our best results when using rules, which could indicate that we need more data for the ML system to be practical. At an initial stage, the rule-based approach seems more promising for "Staging N" and the underlying "Nodes Positive".

Finally, we present the results for "Staging ACPS", which is the Australian Clinico-Pathological Staging system. The results for our ML systems are given in Table 10. We can see that the best results achieve 74.9% F-score, and are obtained with *Naive Bayes* and feature selection; using *UMLS* in two different configurations (lemmas and *SNOMED*). We performed error analysis, and observed that frequently the label codes are used explicitly in the text, although they can be confused with section headings. This analysis also allowed us to see that for 4 instances the annotations were not correct, and manual input errors were made[8]. We expect that the implementation of a recommendation system will help to avoid these kinds of errors in the future.

5. DISCUSSION

The six categories that we have studied behave differently, and we present a summary of the best results per category in Table 11, where we show the results of the *MC* baseline, ML methods, and rules. Note that we classify "Staging N" as numeric, because the underlying classifier that we use is based on the numeric prediction of "Nodes positive". The results show that all categories are able to clearly improve over the *MC* baseline, which indicates that they are able to learn useful clues from the text.

Looking first at the classification techniques, we can see that nominal classifiers benefit from the *Naive Bayes* model with feature selection and rich features, while the best approach for all numeric classes is *SVM* with basic features. The main reason for this is the short representation (sentence-level) used by numeric classifiers, which requires robust models such as *SVM*. Regarding *AdaBoost*,

[8] The figures in the result tables refer to the corrected goldstandard.

Features	*MC*	*Naive Bayes*	*SVM*	*AdaBoost*
BOW	36.0	43.8	48.3	49.3
BOW (FS)	36.0	71.9	59.1	49.3
Lemma	36.0	42.9	48.8	35.5
Lemma (FS)	36.0	66.5	59.6	35.5
BOW and *UMLS*	36.0	42.4	51.2	49.3
BOW and *UMLS* (FS)	36.0	**74.9**	62.1	49.3
BOW, *UMLS* and ConcNeg	36.0	50.7	56.2	48.3
BOW, *UMLS* and ConcNeg (FS)	36.0	69.5	65.0	48.3
SNOMED and *UMLS*	36.0	42.9	51.2	49.3
SNOMED and *UMLS* (FS)	36.0	**74.9**	61.1	49.3
SNOMED, *UMLS* and ConcNeg	36.0	50.2	56.2	48.3
SNOMED, *UMLS* and ConcNeg (FS)	36.0	69.0	66.0	48.3

Table 10: Machine Learning performances for the "Staging ACPS" category (F-score). The best result is given in bold.

the small dataset seems to be a strong limitation to build a robust meta-learner, and it performs poorly.

With regards to the feature sets and resources, we can see that all types of features contribute to one or other task. Nominal classes are clearly the most benefited from rich features, and *UMLS* is the most robust feature type. *SNOMED* is helpful for "Tumour site" and "Staging ACPS", while negation contributes to the best performance for "Staging T".

On the different categories, we observe that the main staging categories perform in the state of the art using ML, and that "Staging N" can boost its F-score by using rules to predict "Nodes Positive". It is promising that all 6 categories perform above the *MC* baseline; 4 out of 6 categories perform above 80% F-score; and "Staging ACPS" performs at 74.9%, which suggests that a practical recommendation system could be implemented using our framework.

Finally, our experiments have also allowed us to identify 6 manual errors in the manual collection of the data, which is a considerable amount for such a sensitive and small dataset. A recommendation tool could be useful to prevent these errors, by highlighting the relevant textual features used by the classifier; it could also be used to process already collected data, and double-check disagreements between the classifiers and the human input to identify errors.

6. CONCLUSIONS

We have implemented and evaluated a framework to identify relevant categories from pathology reports written in natural language. Our results indicate that we are able to predict the labels of 5 of the 6 multiclass problems with F-scores above 74.9%, and above 85% for two of the categories. We also observed that simple feature representations (*BOW* and lemmas) have clear limitations for some target categories, and richer semantic resources are needed for improved performance. Also for classifiers there were big differences in performance depending on the task; *Naive Bayes* with rich features and feature selection was more effective for nominal classes (document classification), while *SVM* with basic features performed better for numeric classes (sentence classification).

Our work also illustrates that this domain is well suited for text mining, even with minimal expert input. The performance for all categories is well above the *MC* baseline, and this is an indication that our classifiers are able to learn useful information from the textual clues. The main reason for the good performance seems to be the type of language used in the texts. In contrast to other domains, such as medical literature, these reports are written economically, using highly repetitive and structured language, and our classifiers are able to build models that show good performance.

For future work our plan is to integrate our prediction system to make recommendations into the Royal Melbourne Hospital workflow. Apart from off-line performance, we will evaluate the impact that the usage of our tool produces in a real setting, both in terms of efficiency and efficacy to avoid human errors.

Type	Category	*MC*	Rules	ML	Class.	Features
Nom	Tumour site	19.7		58.1	NB	*SNOMED* and *UMLS* (FS)
	Staging T	62.6		82.3	NB	*BOW*, *UMLS* and ConcNeg (FS)
	Staging ACPS	36.0		74.9	NB	(*BOW* | *SNOMED*) & *UMLS* (FS)
Num	Nodes Exam.	7.4	93.8	70.1	*SVM*	*BOW* & *UMLS*
	Nodes Positive	64.0	85.2	80.1	*SVM*	Lemma
	Staging N	57.1	81.3	73.2	*SVM*	Lemma

Table 11: Summary of best F-scores per category and method, together with the feature combinations.

7. ACKNOWLEDGMENTS

NICTA is funded by the Australian Government as represented by the Department of Broadband, Communications and the Digital Economy and the Australian Research Council through the ICT Centre of Excellence program. We would like to thank Henry Gasko and his colleagues at the Royal Melbourne Hospital for providing pathology reports for our research.

8. REFERENCES

[1] A. R. Aronson. Effective mapping of biomedical text to the UMLS Metathesaurus: the MetaMap program. In *AMIA Annual Symposium Proceedings*, pages 17–21, Washington DC, 2001.

[2] W. W. Chapman, W. Bridewellb, P. Hanburya, G. F. Cooperb, and B. G. Buchananb. A simple algorithm for identifying negated findings and diseases in discharge summaries. *Journal of Biomedical Informatics*, 34(5):301–310, October 2001.

[3] A. Coden, G. Savova, I. Sominsky, M. Tanenblatt, J. Masanz, K. Schuler, J. Cooper, W. Guan, and P. C. de Groen. Automatically extracting cancer disease characteristics from pathology reports into a disease knowledge representation model. *Journal of Biomedical Informatics*, 42:937–949, 2009.

[4] N. C. Davis and R. C. Newland. Terminology and classification of colorectal adenocarcinoma: The australian clinico-pathological staging system. *Australian and New Zealand Journal of Surgery*, 53(3):211–221, 1983.

[5] M. Hall. *Correlation-based Feature Subset Selection for Machine Learning*. PhD thesis, Department of Computer Science, University of Waikato, New Zealand, 1999.

[6] D. A. Lindberg. The unified medical language system. *Method of Information in Medicine*, 32(4):281–291, 1993.

[7] I. A. McCowan, D. C. Moore, A. N. Nguyen, R. V. Bowman, B. E. Clarke, E. E. Duhig, and M.-J. Fry. Collection of cancer stage data by classifying free-text medical reports. *Journal of the American Medical Informatics Association (JAMIA)*, 14:736–745, 2007.

[8] A. Nguyen, D. Moore, I. McCowan, and M.-J. Courage. Multi-class classification of cancer stages from free-text histology reports using support vector machines. *Proceedings of the IEEE Engineering in Medicine and Biology Society Conference*, 2007:5140–5143, 2007.

[9] A. N. Nguyen, M. J. Lawley, D. P. Hansen, R. V. Bowman, B. E. Clarke, E. E. Duhig, and S. Colquist. Symbolic rule-based classification of lung cancer stages from free-text pathology reports. *Journal of the American Medical Informatics Association (JAMIA)*, 17:440–445, 2010.

[10] M. H. Stanfill, M. Williams, S. H. Fenton, R. A. Jenders, and W. R. Hersh. A systematic literature review of automated clinical coding and classification systems. *J Am Med Inform Assoc*, 17(6):646–651, Nov 2010.

[11] Y. Tsuruoka, Y. Tateishi, J.-D. Kim, T. Ohta, J. McNaught, S. Ananiadou, and J. Tsujii. Developing a robust part-of-speech tagger for biomedical text. In *Advances in Informatics – 10th Panhellenic Conference on Informatics*, pages 382–392, Volas, Greece, 2005.

[12] I. H. Witten and E. Frank. *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann, San Francisco, USA, 2005.