Classification of Cancer Stage from Free-text Histology Reports

Iain McCowan, Darren Moore, Mary-Jane Fry

Abstract— **This article investigates the classification of a patient's lung cancer stage based on analysis of their free-text medical reports. The system uses natural language processing to transform the report text, including identification of UMLS terms and detection of negated findings. The transformed report is then classified using statistical machine learning techniques. A support vector machine is trained for each stage category based on word occurrences in a corpus of histology reports for pathologically staged patients. New reports can be classified according to the most likely stage, allowing the collection of population stage data for analysis of outcomes. While the system could in principle be applied to stage different cancer types, the current work focuses on lung cancer due to data availability. The article presents initial experiments quantifying system performance for T and N staging on a corpus of histology reports from more than 700 lung cancer patients.**

I. Introduction

The stage of a cancer categorises its progression, in terms of the size and location of the primary tumour, as well as any spreading to lymph nodes or formation of distant metastases. The stage is useful both to determine treatment for individual patients based on guidelines, and to stratify outcomes as a basis for population-level analysis of health programmes. These benefits have motivated the definition of international standard protocols, including the TNM (Tumour Nodes Metastases) standard of the AJCC (American Joint Committee on Cancer) and UICC (International Union Against Cancer), summarised in Table I [1]. Staging of patients according to this system is recommended as a standard of care by national cancer bodies, e.g. [2], and provides the basis for international benchmarking of outcomes.

For a variety of reasons, however, formal staging data is not routinely collected for all cancer patients; for instance, according to [3], in 2004 there was no on-going population-based collection of staging information in any Australian state or territory. The preferred method for collection of stage data is through multi-disciplinary team conferences, however due to their time- and resource-intensive nature, it will be difficult to ever meet the total demand in this way. Individual clinicians stage patients, however the consistency of this may vary and it is rarely documented in a formal manner. Technological support for the cancer stage decision has been limited to date. While some software products exist to assign a TNM stage (e.g [4], [5]), these generally rely on highly structured input, and therefore do not reduce the need for expert reading and interpretation of reports.

This article describes a system to collect stage data for cancer patients based on free-text medical reports. For a given patient, the input to the system consists of a variable number of textual reports describing the results of histology tests. The objective of the system is to determine TNM stage values for the patient by applying machine learning *text categorisation* techniques [6]. There are two main types of staging: clinical and pathologic [1]. Clinical staging uses all evidence prior to the first definitive treatment, and is often reliant on interpretation of radiological images. Pathologic staging makes use of more definitive evidence taken from surgery, such as histological examinations. In these initial experiments, the system focusses on predicting the pathologic stage from histology reports. As surgery and histologic testing is not commonly performed on metastatic cancer, the current system is constrained to T and N staging. While the system could in principle be applied to stage other cancer types, the present article focusses on staging lung cancer for reasons of data availability.

The remainder of this article is organised as follows. Section II contains a review of related literature and systems. Section III describes the proposed cancer stage classification system. An experimental evaluation of the system is presented in Section IV, followed by ongoing work and concluding remarks in Section V.

II. Related Work

The system in this article assigns a cancer stage by categorising a set of free-text medical reports. The following subsections briefly describe the context of related work - firstly in general text categorisation, and then in systems specific to the medical domain and cancer staging in particular.

A. Text Categorisation

Text categorisation (see [6], [7] for recent reviews) is the task of deciding if a document belongs to each of a set of predefined categories. Early work in this field focussed on knowledge-based approaches, mainly consisting of manual definition of sets of rules that attempt to encode the expert knowledge required to categorise documents. The major disadvantage with these approaches is the need for human experts to define and maintain the comprehensive rule set required for high accuracy. For this reason most text categorisation research in recent years has concentrated on machine learning approaches which automatically build text classifiers by learning the characteristics of each category from a set of preclassified documents (the training corpus). Such a machine learning approach is taken in the present system. The most common state-of-the-art text categorisation

I. McCowan and D. Moore are with CSIRO eHealth Research Centre, Brisbane, Australia {iain.mccowan, darren.moore}@csiro.au

M-J. Fry is with the Queensland Cancer Control Analysis Team, Brisbane, Australia mary-jane_fry@health.qld.gov.au

T: Primary Tumour	X	Primary tumor cannot be assessed.
	0	No evidence of primary tumor.
	is	Carcinoma in situ.
	1,2,3,4	Increasing size and/or local extent of the primary tumour.
N: Regional Lymph Nodes	X	Regional lymph nodes cannot be assessed.
	0	No regional lymph node metastasis.
	1, 2, 3	Increasing involvement of regional lymph nodes.
M: Distant Metastasis	X	Distant metastasis cannot be assessed.
	0	No distant metastasis.
	1	Distant metastasis.

TABLE I

SUMMARY OF THE TNM STAGING PROTOCOL [1].

systems derive a Vector Space Model (VSM) document representation and then classify this using Support Vector Machines (SVM's).

In the VSM representation a document is represented as a vector of weights, with one vector element for each word (or term) that occurs in the entire training corpus. The weight assigned to a word can be either a binary value (to indicate the simple presence or absence of a word in the document), or a non-binary value based on its frequency of occurrence - see [7] for an overview of different weighting schemes.

Support vector machines (SVM's) were introduced in [8] and were first applied to the problem of text categorisation in [9]. A SVM is the hyperplane that maximises the separation between the closest positive and negative training examples (the support vectors). There are several properties that make SVM's suitable for application to text categorisation [9]: the ability to cope with very high dimensional input feature spaces (where most dimensions are relevant) and sparse document vectors, and the fact that text categories are often linearly separable.

B. Medical Text Analysis

Most medical-related automated text analysis work in the literature has dealt with the problem of converting free-text reports into standard codes or structured formats that are more suitable for further analysis, e.g. [10], [11]. Beyond such automatic coding systems, there have been a number of systems that have attempted classification of medical reports, for instance according to specific medical diseases or conditions. This has included: classification of radiology reports according to 6 conditions [12], classification of high quality MEDLINE articles [13], classification of emergency department reports into eight syndromic categories [14], detecting fever in emergency department patients [15], detection of radiology reports that support a finding of inhalational anthrax [16], detection of acute gastrointestinal syndrome from emergency department reports [17], and determining whether a finding or disease in a report is absent [18].

C. Software Support for Cancer Staging

Literature and market reviews have only uncovered a few instances of systems specifically designed to assist in the cancer stage decision. In the research literature, there is no reported work in staging cancer from unstructured free-text input. The stage of cervical cancer was determined using a neural network classifier in [19], using a 15-element vector encoding MRI and PET results as input. A soft-computing approach was used in [20] to classify cervical cancer cases into one of 4 FIGO stages based on a vector encoding the presence or absence of each major symptom. The mTuitive [4] xPert product line includes a module for cancer staging according to the AJCC TNM guidelines. From the information available on their website, their product is based on structured data entry of pathologic results. The Collaborative Staging Task Force [5] has produced a set of common software tools to determine the cancer stage according to multiple systems, based on a structured set of cancer-dependent data items.

The system proposed in the current article can be differentiated from the above systems in two main ways: firstly in its use of free-text reports rather than structured input data, and secondly as it uses probabilistic rather than deterministic algorithms. These distinctions may be important when when access to expert knowledge of staging is limited and when only partial and uncertain information is available.

III. DESCRIPTION OF SYSTEM

The proposed system consists of two phases: text pre-processing and assignment of the cancer stage.

A. Text Pre-processing

Step 1: Normalisation: Normalisation aims at reducing basic variations between different reports by enforcing consistent expression of common terms. Specifically, the formats of acronyms, numbers and dimensions are standardised, relevant abbreviations are expanded, spelling variants are mapped to a common form, and any non-informative character sequences are removed. These normalisation rules are encoded with regular expressions and implemented using search and replace operations.

Step 2: Parsing into UMLS Terms: In this step, the document is parsed into a sequence of terms from the Unified Medical Language System (UMLS) SPECIALIST Lexicon [21]. Each word from the input is first converted to its UMLS base form, then a parser converts these to a sequence of more general (potentially multi-word) UMLS code terms. The parser is implemented as a state machine using the Viterbi dynamic programming algorithm [22] to find the optimal decomposition of each sentence into UMLS terms. This dynamic programming approach is necessary as there may be several possible decompositions for a given sentence into multi-word terms. The optimal term sequence is defined as one having the minimum number of terms.

Step 2: Detection of Negated Terms: An algorithm (NegEx) for determining the presence or absence of a finding in a free-text report was proposed in [18]. The algorithm detects a number of common medical negation phrases (e.g. "no evidence of"), and then associates these with neighbouring disease or condition terms. In the current system, only a small subset of approximately 30 UMLS terms is considered for negation, comprising words that are highly relevant to the TNM lung cancer staging protocol (e.g. tumour, mediastinum, pleura). These UMLS terms are replaced with a new term code indicating a negated form.

B. *Assignment of Cancer Stage*

Step 1: Feature Extraction: A vector space model is used to represent each text report in a data corpus as a vector of term weights. The term weights are calculated according to the LTC-weighting scheme [23], [7]. The LTC weights are commonly used in state-of-the-art text categorisation systems, as they effectively de-emphasise common terms (occurring often in many documents), produce normalised weights across different length documents, and reduce the impact of large differences in frequency.

Step 2: SVM Classification: In this step, standard Support Vector Machines (SVM's) are used to classify the cancer stage of each medical report. For each cancer stage category, a binary SVM classifier is trained based on whether each document in the training corpus is relevant to that particular category or not. Note that this means a given medical report will not necessarily be assigned a single stage within the T and N groupings, but rather zero or more. The SVM's are implemented using the *SVM*^{light} [24] toolkit. The parameters of the SVM are first trained from a corpus of text reports with stage categories. During testing, the SVM outputs a score that can be thresholded to decide if a new document belongs to that particular class.

IV. Experimental Evaluation

A. *Method*

Data Corpus: To train and validate the system, a corpus of de-identified medical reports with corresponding stage data was obtained for 718 lung cancer patients following research ethics approval. The corpus was compiled from two separate sources: a database of pathologic staging decisions for lung cancer patients (Queensland Integrated Lung Cancer Outcomes Project data [25]) for use as ground-truth for the classifier training and testing, and a set of histology reports for lung cancer patients extracted from the state pathology information system (AUSLAB).

SVM Training and Testing: A binary classifier was trained for each stage category (e.g. T1 vs not-T1). In order to maximise the amount of SVM training data while still reporting significant results on this dataset, an N-fold scheme was applied. First the data was randomly divided into 100 subsets, then in each fold system output was generated for one subset from an SVM trained on the remaining 99 subsets. This meant that over the 100 folds, results could be reported on the full patient list while ensuring each result was produced by an unbiased system (where test data was not used during system training).

Stage	Cases	Sensitivity	Specificity	PPV	F1
T1	204	0.52	0.82	0.54	0.53
T2	405	0.67	0.59	0.68	0.68
T3	52	0.67	0.97	0.67	0.67
T4	49	0.41	0.95	0.38	0.39
macro-average	718	0.57	0.83	0.57	0.57
micro-average	718	0.61	0.88	0.62	0.61

Stage	Cases	Sensitivity	Specificity	PPV	F1
N0	437	0.87	0.81	0.88	0.87
N1	150	0.59	0.89	0.58	0.59
N2	72	0.67	0.96	0.63	0.65
macro-average	718	0.71	0.89	0.70	0.70
micro-average	718	0.78	0.90	0.78	0.78

TABLE II

Classifier results with macro- and micro-averages: Positive cases, Sensitivity (Recall), Specificity, Positive Predictive Value (PPV, Precision), F1 Measure.

Performance Measures: Results are reported for each classifier in terms of standard binary test measures: *sensitivity*, *specificity* and *positive predictive value* (*PPV*). Given the True Positives (TP), True Negatives (TN), False Positives (FP) and False Negatives (FN), then Sensitivity = TP / (TP + FN), Specificity = TN / (TN + FP), and PPV = TP / (TP + FP). In the text classification literature, sensitivity and PPV are referred to as *recall* and *precision*, respectively. If a single performance measure is required, the *F1-measure* is commonly used in the text classification literature; this is the harmonic mean of precision and recall. Each measure can be calculated on a per-category basis and then averaged across categories to give macro-averaged results, or across all patients to give micro-averaged results. Depending on the application, a trade-off exists between complementary measures, such as sensitivity and specificity, and this can be controlled by varying one or more classifier hyper-parameters. It is common to report performance at a task-relevant operating point, such as the break-even point between two complementary measures.

B. *Discussion of Results*

Results are reported in Table II at the sensitivity/PPV break-even point. Results are not given for T0 or N3, due to lack of data (as pathologic staging is rarely conducted for these categories). Results show consistently high specificity across categories indicating strong reliability in a negative decision. In general, sensitivity and PPV are higher for N staging. The T4 stage category shows the lowest performance, which may be attributed to the use of only histology reports; very few T4 cases are surgically resected and these will often be incomplete, restricting the pathologist's ability to assess T4 status. A further factor in results is the benefit of using more, and better balanced, data for SVM training; stage categories with a significant number of cases have better classification performance. To show the trade-off that exists

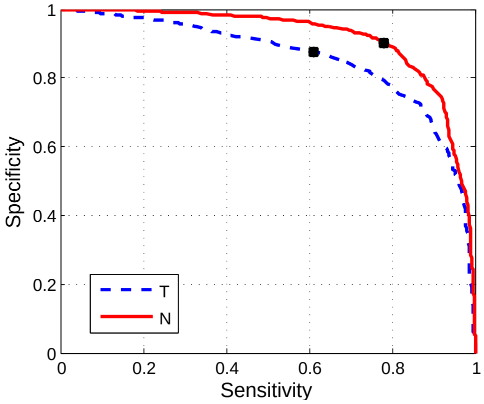

Fig. 1. Receiver Operator Characteristic (ROC) Curves for T and N micro-averaged Specificity vs Sensitivity (T area = 0.86, N area = 0.92). The point on each curve indicates the operating point for Table II results.

between performance on positive and negative cases, the sensitivity and specificity Receiver Operator Characteristic (ROC) curves are plotted for T and N staging in Figure 1. A further measure of overall system performance is provided by the area under the ROC curve: these plots show an area of 0.86 for T and 0.92 for N staging, which is a promising initial result.

V. Conclusions and Future Work

This article has presented initial work towards a system to automatically determine a patient's cancer stage from their medical reports. An SVM-based text classification system was implemented and evaluated on a corpus of histology reports and pathologic stage data for 718 lung cancer patients. At the sensitivity/PPV break-even point, the system achieves average sensitivity of 0.61 and specificity of 0.88 for T staging, and sensitivity of 0.78 and specificity of 0.90 for N staging. While these results are encouraging, there is potential to improve this by adding other data sources (e.g. radiology reports) and incorporating expert knowledge of the staging protocols in the system design. Ongoing work will investigate this and evaluate the system more thoroughly as a tool for retrospective collection of population stage data.

Acknowledgements

This research was done in partnership with the Queensland Cancer Control and Analysis Team (QCCAT) in Queensland Health. The authors would like to acknowledge: Hazel Harden, Shoni Colquist and Steven Armstrong from QCCAT for their help in system concept definition, and providing access to data and clinical experts; Jaccalyne Brady and Donna Fry from the Queensland Integrated Lung Cancer Outcomes Project for explaining current cancer staging processes; Dr Rayleen Bowman and Dr Belinda Clarke for their invaluable expert advice; and Wayne Watson from the AUSLAB Support Group for extracting data for the research corpus.

References

[1] F.L. Greene, D.L. Page, I.D. Fleming, A. Fritz, C.M. Balch, D.G. Haller, and M. Morrow, editors. *AJCC Cancer Staging Manual*. Springer, 6 edition, 2002.

[2] *Clinical practice guidelines for the prevention, diagnosis and management of lung cancer*. The Cancer Council Australia, 2004.

[3] T. Threlfall, J. Wittorff, P. Boutdara, L. Fritschi, J. Heyworth, P. Katris, and H. Sheiner. Collection of Population-based Cancer Staging Information in Western Australia - A Feasibility Study. Technical report, National Cancer Control Initiative, Melbourne, 2004. Available at www.ncci.org.au.

[4] mTuitive. http://www.mtuitive.com/.

[5] Collaborative Staging. http://www.cancerstaging.org/cstage/index.html.

[6] F. Sebastiani. Machine learning in automated text categorization. *ACM Computing Surveys*, 34(1):1–47, March 2002.

[7] K. Aas and L. Eikvil. Text categorisation: A survey. Technical report, Norwegian Computing Center, 1999.

[8] V.N. Vapnik. *The nature of statistical learning theory*. Springer, 1995.

[9] T. Joachims. Text categorization with support vector machines: Learning with many relevant features. In *Proceedings of the 10th European Conference on Machine Learning*, 1998.

[10] G. Cooper and R. Miller. An experiment comparing lexical and statistical methods for extracting MeSH terms from clinical free text. *Journal of American Medical Informatics Association*, 5:62–75, 1998.

[11] B. Hazlehurst, H.R. Frost, D.F. Sittig, and V.J. Stevens. MediClass: A system for detecting and classifying encounter-based clinical events in any electronic medical record. *Journal of the American Medical Informatics Association*, 12:517–529, 2005.

[12] A.B. Wilcox and G. Hripcsak. The role of domain knowledge in automating medical text report classification. *Journal of the American Medical Informatics Association*, 10:330–338, 2003.

[13] Y. Aphinyanaphongs, I. Tsamardinos, A. Statnikov, D. Hardin, and C.F. Aliferis. Text categorization models for high-quality article retrieval in internal medicine. *Journal of the American Medical Informatics Association*, 12(2):207–216, 2005.

[14] W.W. Chapman et al. Classifying free-text triage chief complaints into syndromic categories with natural language processing. *Artificial Intelligence in Medicine*, 33(1):31–40, 2004.

[15] W.W. Chapman, J.N. Dowling, and M.M. Wagner. Fever detection from free-text clinical records for biosurveillance. *Journal of Biomedical Informatics*, 37:120–127, 2004.

[16] W.W. Chapman, G.F. Cooper, P. Hanbury, B.E. Chapman, L.H. Harrison, and M.M. Wagner. Creating a text classifier to detect radiology reports describing mediastinal findings associated with inhalational anthrax and other disorders. *Journal of the American Medical Informatics Association*, 10:494–503, 2003.

[17] O. Ivanov, M.M. Wagner, W.W. Chapman, and R.T. Olszewski. Accuracy of three classifiers of acute gastrointestinal syndrome for syndromic surveillance. proc american medical informatics association symp. 2002:345-9. In *Proceedings of the American Medical Informatics Association Symposium*, 2002.

[18] W.W. Chapman, W. Bridewell, P. Hanbury, G.F. Cooper, and B.G. Buchanan. A simple algorithm for identifying negated findings and diseases in discharge summaries. *Journal of Biomedical Informatics*, 34:301–310, 2001.

[19] P. Phinjaroenphan and S. Bevinakoppa. Automated prognostic tool for cervical cancer patient database. In *Proceedings of International Conference on Intelligent Sensing and Information Processing*, 2004.

[20] P. Mitra, S. Mitra, and S.K. Pal. Staging of cervical cancer with soft computing. *IEEE Transactions on Biomedical Engineering*, 47(7):934–940, 2000.

[21] Unified Medical Language System (UMLS). http://www.nlm.nih.gov/research/umls/.

[22] A. Viterbi. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE Transactions on Information Theory*, 13(2):260267, April 1967.

[23] C. Buckley, G. Salton, J. Allan, and A. Singhal. Automatic query expansion using SMART: TREC 3. In *Proceedings of the 3rd Text Retrieval Conference*. NIST, 1994.

[24] T. Joachims. Making large-scale SVM learning practical. In B. Schölkopf, C. Burges, and A. Smola, editors, *Advances in Kernel Methods - Support Vector Learning*. MIT Press, 1999.

[25] K.M. Fong, R. Bowman, D. Fielding, R. Abraham, M. Windsor M, and G. Pratt. Queensland integrated lung cancer outcomes project (QILCOP): Initial accrual and preliminary data from the first 30 months. Abstract of Presentation at The Thoracic Society of Australia and New Zealand Annual Scientific Meeting, April 2003.