Overview of the ALTA 2012 Shared Task

Iman Amini* David Martinez[†] Diego Molla‡
***RMIT Dept of Computer Science and NICTA, Australia**
[†]NICTA and the University of Melbourne, CIS Department, Australia
‡Department of Computing, Macquarie University, Australia
iman.amini@rmit.edu.au
david.martinez@nicta.edu.au
diego.molla-aliod@mq.edu.au

Abstract

The ALTA shared task ran for the third time in 2012, with the aim of bringing research students together to work on the same task and data set, and compare their methods in a current research problem. The task was based on a recent study to build classifiers for automatically labeling sentences to a pre-defined set of categories, in the domain of Evidence Based Medicine (EBM). The partaking groups demonstrated strong skills this year, outperforming our proposed benchmark systems. In this overview paper we explain the process of building the benchmark classifiers and data set, and present the submitted systems and their performance.

1 Introduction

Medical research articles are one of the main sources for finding answers to clinical queries, and medical practitioners are advised to base their decisions on the available medical literature. Using the literature for the purpose of medical decision making is known as Evidence Based Medicine (EBM).

According to the EBM guidelines, users are suggested to formulate queries which follow structured settings, and one of the most used systems is known as PICO: Population (P) (i.e., participants in a study); Intervention (I); Comparison (C) (if appropriate); and Outcome (O) (of an Intervention). This system allows for a better classification of articles, and improved search. However curating this kind of information manually is unfeasible, due to the large amount of publications being created on daily basis.

The goal of the ALTA 2012 shared task was to build automatic sentence classifiers to map the content of biomedical abstracts into a set of pre-defined categories. The development of this kind of technology would speed up the curation process, and this has been explored in recent work (Chung, 2009; Kim et al., 2011). One of the aims of this task was to determine whether participants could develop systems that can improve over the state of the art.

2 Dataset

Different variations and extensions of the PICO classification have been proposed and the schema used for this competition is PIBOSO (Kim et al., 2011), which removes the *Comparison* tag, and adds three new tags: *Background* , *Study Design* and *Other*. Thus, the tag-set is defined as follows:

- *Population*: The group of individual persons, objects, or items comprising the study's sample, or from which the sample was taken for statistical measurement;

- *Intervention*: The act of interfering with a condition to modify it or with a process to change its course (includes prevention);

- *Background*: Material that informs and may place the current study in perspective, e.g. work that preceded the current; information about disease prevalence; etc;

- *Outcome*: The sentence(s) that best summarise(s) the consequences of an intervention;

- *Study Design*: The type of study that is described in the abstract;

Iman Amini, David Martinez and Diego Molla. 2012. Overview of the ALTA 2012 Shared Task. In *Proceedings of Australasian Language Technology Association Workshop*, pages 124−129.

	All	Struct.	Unstruct.
Total			
- Abstracts	1,000	38.9%	61.1%
- Sentences	11,616	56.2%	43.8%
- Labels	12,211	55.9%	44.1%
% per label			
- Population	7.0%	5.6%	7.9%
- Intervention	5.9%	4.9%	6.6%
- Background	22.0%	10.3%	34.2%
- Outcome	38.9%	34.0%	40.9%
- Study Design	2.0%	2.3%	1.4%
- Other	29.2%	42.9%	9.0%

Table 1: Statistics of the dataset. "% per label" refers to the percentage of sentences that contain the given label (the sum is higher than 100% because of multilabel sentences).

- *Other*: Any sentence not falling into one of the other categories and presumed to provide little help with clinical decision making, i.e. non-key or irrelevant sentences.

We rely on the data manually annotated at sentence level by (Kim et al., 2011), which consists of 1,000 abstracts from diverse topics. Topics of the abstracts refer to various queries relating to traumatic brain injury, spinal cord injury, and diagnosis of sleep apnoea. Over three hundred abstracts are originally structured, that is, they contain rhetorical roles or headings such as *Background*, *Method*, etc. For the competition, however, we do not separate abstracts based on their structuring, rather we leave them interspersed in the training and test data. Nonetheless, we provide participants with the headings extracted from the structured abstracts to be used as a set of structural features.

In order to build classifiers, 800 annotated training abstracts were provided, and the goal was to automatically annotate 200 test abstracts with the relevant labels. Table 1 shows the exact number of sentences and the percentages of the frequency of labels across the data set. We relied on "Kaggle in Class" to manage the submissions and rankings[1], and randomly divided the test data into "public" and "private" evaluation; the former was used to provide preliminary evaluations during the competition, and the latter to define the final classification of systems.

We provided two benchmark systems at the beginning of the competition. The first system is a simple frequency-based approach, and the second system is a variant of the state-of-the-art system presented by (Kim et al., 2011), using a machine learning algorithm for predictions.

2.1 Naive Baseline

For the naive baseline we merely rely on the most frequent label occurring in the training data, given the position of a sentence. For instance, for the first four sentences in the abstract the most frequent label is *Background*, for the fifth it is *Other*, etc.

2.2 Conditional Random Field (CRF) Benchmark

CRFs (Lafferty et al., 2001) were designed to label sequential data, and we chose this approach because it has shown success in sentence-level classification (Hirohata et al., 2008; Chung, 2009; Kim et al., 2011). Thus we tried to replicate the classifier used by (Kim et al., 2011). However our systems differ in the selection of features used for training. We use lexical and structural features:

1. **Lexical features:** bag of words and Part Of Speech (POS) tags for the lexical features; and

2. **Structural features:** position of the sentences and the rhetorical headings from the structured abstracts. If a heading *h1* covered three lines in the abstract, all the three lines will be labeled as *h1*.

We used NLTK (Bird et al., 2009) to produce a list of POS tags and for the CRF classifier we utilized the Mallet (McCallum, 2002) open source software.

Upon completion of the challenge we learned that our input to the CRF Benchmark did not have a separation between abstracts, causing Mallet to underperform. We rectified the training representation and obtained the accurate score which we refer to as CRF_corrected.

3 Evaluation

Previous work has relied on F-score for evaluating this task, but we decided to choose the *receiver operating characteristic* (ROC) curves and corresponding *area under curve* (AUC) value as

[1] http://www.kaggle.com/

Student Category	Open Category
Marco Lui	Macquarie Test
A_MQ	DPMCNA
System_Ict	Dalibor
	Starling
	Mix

Table 2: Team names and categories.

the main metric. ROC curves plot the fraction of true positives out of the positives (TPR = true positive rate) vs. the fraction of false positives out of the negatives (FPR = false positive rate), at various threshold settings. The AUC score is the area under this plot, and the main benefit of this metric is that it allows us to compare classification outputs that assign probability distributions to labels, instead of a binary decision. We also provide F-scores for a better comparison with the existing literature.

Table 2 shows the team names and the categories. There were two categories: "student" and "open". Members of the "student" category were exclusively students at any level: undergraduate or postgraduate. None of the members of the "student" category can hold a PhD in a relevant area. Members of the "open" category included those who could not participate in the "student" category. The winner of the student category and winner overall was Marco Lui from NICTA and the University of Melbourne, followed by Team A_MQ (Abeed Sarker) from Macquarie University and Team System_Ict (Spandana Gella and Duong Thanh Long) from the University of Melbourne. The top participants of the open category were Team Macquarie_Test (Diego Mollá, one of the task organisers) from Macquarie University, and Team DPMCNA (Daniel McNamara) from Australia National University and Kaggle. The description of the systems is provided in Section 4.

Table 3 shows the final scores obtained by the 8 participants and the baseline systems. The scores for private and public test data are very similar. We can see that the top system improved over our state-of-the-art baseline, and all the top-3 were close to its performance.

System	Private Test	Public Test	F-score
Marco Lui	**0.96**	**0.97**	**0.82**
A_MQ	0.95	0.96	0.80
Macquarie Test	0.94	0.94	0.78
DPMCNA	0.92	0.93	0.71
System_Ict	0.92	0.93	0.73
Dalibor	0.86	0.92	0.73
Starling	0.86	0.87	0.78
Mix	0.83	0.84	0.74
Benchmarks			
- CRF_corrected	0.86	0.88	0.80
- CRF_official	0.80	0.83	0.70
- Naive	0.70	0.70	0.55

Table 3: AUC and F-scores for public and private tests. The best results per column are given in bold.

We relied on a non-parametric statistical significance test known as random shuffling (Yeh, 2000) to better compare the F-scores of the participating systems and benchmarks. We present in Table 5 the ranking of systems according to their F-scores, and the p-value when comparing each system with the one immediately below it in the table[2]. The p-values illustrate different clusters of performance, and they show that team "Marco Lui" significantly improves the CRF_corrected state-of-the-art benchmark, and that team "A_MQ" and CRF_corrected perform at the same level.

Table 4 shows the F-scores separately for each class; the best scoring system is superior for most of the 6 classes. We observed that the ranking of the participants as measured by the official AUC score was the same for the top participants, but the ranking at the bottom of the list of participants differed. The *Outcome* and *Intervention* labels have the highest and lowest scores, respectively, which mostly correlates to the amount of available training instances for each.

4 Description of Systems

The top participants in the task kindly provided a short description of their architectures, which is given in the Appendix. All these submissions relied on Machine Learning (ML) methods, namely Support Vector Machines (SVM), Stacked Logistic Regression, Maximum Entropy, Random Forests, and CRF. Only one of the top participants

[2] The p-value gives the probability of obtaining such an F-score difference between the compared systems assuming that the null hypothesis (that the systems are not significantly different from each other) holds.

System	Population	Intervention	Background	Outcome	Study Design	Other
Marco Lui	**0.58**	0.34	**0.80**	**0.89**	0.59	**0.85**
A_MQ	0.51	**0.35**	0.78	0.86	0.58	0.84
Macquarie Test	0.56	0.34	0.75	0.84	0.52	0.80
Starling	0.32	0.20	**0.80**	0.87	0.00	0.82
DPMCNA	0.28	0.12	0.70	0.78	0.48	0.73
Mix	0.45	0.19	0.68	0.82	0.40	0.81
System_Ict	0.30	0.15	0.68	0.84	0.35	0.83
Dalibor	0.30	0.15	0.68	0.84	0.40	0.83
Naive	0.00	0.00	0.59	0.68	0.00	0.15
CRF_official	0.33	0.22	0.55	0.78	0.67	0.81
CRF_corrected	**0.58**	0.18	**0.80**	0.86	**0.68**	0.83
Aggregate	0.38	0.21	0.71	0.83	0.42	0.76

Table 4: F-scores across each individual label class and the aggregate. The best results per column are given in bold.

System	F-score	p-value
Marco Lui	0.82	0.0012
CRF_corrected	0.80	0.482
A_MQ	0.80	0.03
Starling	0.78	0.3615
Macquarie Test	0.78	0.0001
Mix	0.74	0.1646
System_Ict	0.73	0.5028
Dalibor	0.73	0.0041
DPMCNA	0.71	0
Naive	0.55	-

Table 5: Ranking of systems according to F-score, and pairwise statistical significance test between the target row and the one immediately below. The horizontal lines cluster systems according to statistically significant differences.

relied on sequential classifiers (team "System_Ict" applied CRFs).

Two of the top systems (teams "Marco Lui" and "Macquarie Test") used a two-layered architecture, where features are learned through a first pass (supervised for "Marco Lui", unsupervised for "Macquarie Test"). Team "A_MQ" performed parameter optimisation separately for each of the PIBOSO categories, and it was the only team to use Metamap as a source of features. Feature selection was used by teams "Daniel McNamara" and "System_Ict", which also achieved high performances.

5 Conclusions

The third shared task aimed at fostering research on classifying medical sentences into the predefined PIBOSO category to aid the practice of EBM. Participants from Australia and world-wide competed on this task and the winning team obtained better results than state of the art where the difference was shown to be statistically significant. The best performing technique was attributed to the usage of the meta-learner feature stacking approach using three different sets of features.

We will endeavor to identify such important research problems and provide a forum for research students to provide their effective solutions in the forthcoming shared tasks.

6 Acknowledgements

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 all the sponsors, management of Kaggle in Class and participants. We would also like to congratulate the winners for their fine work.

References

Steven Bird, Ewan Klein, and Edward Loper. 2009. *Natural Language Processing with Python*. O'Reilly Media.

Grace Chung. 2009. Sentence retrieval for abstracts of randomized controlled trials. *BMC Med Inform Decis Mak*, 9:10.

Spandana Gella and Duong Thanh Long. 2012. Automatic sentence classifier for event based medicine: Shared task system description. In *Australasian*

Language Technology Workshop 2012 : ALTA Shared Task.

Kenji Hirohata, Naoaki Okazaki, Sophia Ananiadou, and Mitsuru Ishizuka. 2008. Identifying sections in scientific abstracts using conditional random fields. In *Proc. of 3rd International Joint Conference on Natural Language Processing*, pages 381–388.

Su Nam Kim, David Martinez, Lawrence Cavedon, and Lars Yencken. 2011. Automatic classification of sentences to support evidence based medicine. *BMC bioinformatics*, 12:S5.

John Lafferty, Andrew Kachites McCallum, and Fernando Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, pages 282–289. Morgan Kaufmann Publishers Inc.

Marco Lui. 2012. Feature stacking for sentence classification in evidence-based medicine. In *Australasian Language Technology Workshop 2012 : ALTA Shared Task*.

Andrew Kachites McCallum. 2002. Mallet: A machine learning for language toolkit. http://mallet.cs.umass.edu.

Diego Molla. 2012. Experiments with clustering-based features for sentence classification in medical publications: Macquarie test's participation in the alta 2012 shared task. In *Australasian Language Technology Workshop 2012 : ALTA Shared Task*.

Alexander Yeh. 2000. More accurate tests for the statistical significance of result differences. In *Proceedings of the 18th Conference on Computational Linguistics (COLING)*, pages 947–953, Saarbrücken, Germany.

Appendix: Description of the top systems

The following text is by the team competitors who kindly agreed to send us their system descriptions.

Team Marco (Marco Lui)

A full description of this system is given in (Lui, 2012). We used a stacked logistic regression classifier with a variety of feature sets to attain the highest result. The stacking was carried out using a 10-fold cross-validation on the training data, generating a pseudo-distribution over class labels for each training instance for each feature set. These distribution vectors were concatenated to generate the full feature vector for each instance, which was used to train another logistic regression classifier. The test data was projected into the stacked vector space by logistic regression classifiers trained on each feature set over the entire training collection. No sequential learning algorithms were used; the sequential information is captured entirely in the features. The feature sets we used are an elaboration of the lexical, semantic, structural and sequential features described by Kim et al (Kim et al., 2011). The key differences are: (1) we used part-of-speech (POS) features differently. Instead of POS-tagging individual terms, we represented a document as a sequence of POS-tags (as opposed to a sequence of words), and generated features based on POS-tag n-grams, (2) we added features to describe sentence length, both in absolute (number of bytes) and relative (bytes in sentence / bytes in abstract) terms, (3) we expanded the range of dependency features to cover bag-of-words (BOW) of not just preceding but also subsequent sentences, (4) we considered the distribution of preceding and subsequent POS-tag n-grams, (5) we considered the distribution of preceding and subsequent headings. We also did not investigate some of the techniques of Kim et al, including: (1) we did not use any external resources (e.g. MetaMap) to introduce additional semantic information, (2) we did not use rhetorical roles of headings for structural information, (3) we did not use any direct dependency features.

Team A_MQ (Abeed Sarker)

In our approach, we divide the multi-class classification problem to several binary classification problems, and apply SVMs as the machine learning algorithm. Overall, we use six classifiers, one for each of the six PIBOSO categories. Each sentence, therefore, is classified by each of the six classifiers to indicate whether it belongs to a specific category or not. An advantage of using binary classifiers is that we can customise the features to each classification task. This means that if there are features that are particularly useful for identifying a specific class, we can use those features for the classification task involving that class, and leave them out if they are not useful for other classes. We use RBF kernels for each of our SVM classifiers, and optimise the parameters using 10-fold cross validations over the training data for each class. We use the MetaMap tool box to identify medical concepts (CUIs) and semantic types for all the medical terms in each sentence. We use the MedPost/SKR parts of speech tagger to annotate each word, and further pre-process the text by lowercasing, stemming and removing stopwords. For features, we use n-grams, sentence positions (absolute and relative), sentence lengths, section headings (if available), CUIs and semantic types for each medical concept, and previous sentence n-grams. For the outcome classification task, we use a class-specific feature called ‘cue-word-count’. We use a set of key-words that have been shown to occur frequently with sentences representing outcomes, and, for each sentence, we use the number of occurrences of those key-words as a feature. Our experiments, on the training data, showed that such a class-specific feature can improve classifier performance for the associated class.

Team Macquarie Test (Diego Molla)

A full description of this system is given in (Molla, 2012). The system is the result of a series of experiments where we tested the impact of using cluster-based features for the task of sentence classification in medical texts. The rationale is that, presumably, different types of medical texts will have specific types of distributions of sentence types. But since we don’t know the document types, we cluster the documents according to their distribution of sentence types and use the resulting clusters as the document types. We first trained a classifier to obtain a first prediction of the sentence types. Then the documents were clustered based on the distribution of sentence types. The resulting cluster information, plus additional features, were used to train the final set of classifiers. Since a sentence may have multiple labels we used binary classifiers, one per sentence type. At the classification stage, the sentences were classified using the first set of classifiers. Then their documents were assigned the closest cluster, and this information was fed to the second set of classifiers. The submission with best results used Maxent classifiers, all classifiers used uni-gram features plus the normalised sentence position, and the second classifiers used, in addition, the cluster information. The number of clusters was 4.

Team DPMCNA (Daniel McNamara)

We got all of the rows in the training set with a 1 in the prediction column and treated each row as series of predictors and a class label corresponding to sentence type (’background’, ’population’, etc.) We performed pre-processing of the training and test sets using stemming, and removing case, punctuation and extra white space. We then calculated the training set mutual information of each 1-gram with respect to the class labels, recording the top 1000 features. For each sentence, We converted it into a feature vector where the entries were the frequencies of the top features, plus an entry for the sentence number. We then trained a Random Forest (using R’s randomForest package with the default settings) using these features and class labels. We used the Random Forest to predict class probabilities for each test response variable. Note that We ignored the multi-label nature of the problem considering most sentences only had a single label.

Team System_Ict (Spandana Gella, Duong Thanh Long)

A full description of this system is given in (Gella and Long, 2012). Our top 5 sentence classifiers use Support Vector Machine (SVM) and Conditional Random Fields (CRFs) for learning algorithm. For SVM we have used libsvm 1 package and for CRF we used CRF++ 2 package. We used 10-fold cross validation to tweak and test the best suitable hyper parameters for our methods. We have observed that our systems performed very well when we do cross validation on train data but suffered over fitting. To avoid this we used train plus labelled test data with one of the best performing systems as our new training data. We observed that this has improved our results by approximately 3%. We trained our classifiers with different set of features which include lexical, structural and sequential features. Lexical features include collocational information, lemmatized bag-of-words features, part-of-speech information (we have used MedPost part-of-speech tagger) and dependency relations. Structural features include position of the sentence in the abstract, normalised sentence position, reverse sentence position, number of content words in the sentence, abstract section headings with and without modification as mentioned in (Kim et al., 2011). Sequential features were implemented the same way as in (Kim et al., 2011) with the direct and indirect features. After having the pool of features from the above defined features, we perform feature selection to ensure that we always have the most informative features. We used the information gain algorithm from R system3 to do feature selection.