Evaluating Machine Learning for Information Extraction

Neil Ireson N.IRESON@DCS.SHEF.AC.UK
Fabio Ciravegna F.CIRAVEGNA@DCS.SHEF.AC.UK
Department of Computer Science, University of Sheffield, Sheffield, UK

Mary Elaine Califf MECALIFF@ILSTU.EDU
Illinois State University, School of Information Technology, Campus Box 5150, Normal, IL 61790, USA

Dayne Freitag DBF@FAIRISAAC.COM
Fair Isaac Corporation, 901 Marquette Avenue, Suite 3200, Minneapolis, MN 55402 USA

Nicholas Kushmerick NICK@UCD.IE
School of Computer Science and Informatics, University College Dublin, Dublin 4, Ireland

Alberto Lavelli LAVELLI@ITC.IT
Centro per la Ricerca Scientifica e Tecnologica, Istituto Trentino di Cultura. I-38050 Povo TN, Italy

Abstract

Comparative evaluation of Machine Learning (ML) systems used for Information Extraction (IE) has suffered from various inconsistencies in experimental procedures. This paper reports on the results of the Pascal Challenge on Evaluating Machine Learning for Information Extraction, which provides a standardised corpus, set of tasks, and evaluation methodology. The challenge is described and the systems submitted by the ten participants are briefly introduced and their performance is analysed.

1. Introduction

As part of text understanding, Information Extraction (IE) is a long standing goal of Artificial Intelligence (AI). Recently, fuelled by the vast amount of electronic documents available via the WWW and a growing interest in annotating these documents to utilise them for the Semantic Web, automatic IE is receiving increasing attention from the Machine Learning (ML) community. In order to determine the appropriate ML techniques to use for IE from textual data it is necessary to perform comparative evaluations. However, to-date such evaluations have been hampered by under-defined experimental procedures. In addition the evaluations are carried out from an IE perspective so that the focus is not on the contribution of the ML algorithms to the IE task.

The MUC conferences were the first comprehensive effort to evaluate IE systems. They provided annotated corpora that are still used as standard testbeds (Hirschman, 1998), and also evaluation software (Douthat, 1998). More recently other corpora have been made available to the research community, such as the job postings collection (Califf, 1998), and the seminar announcements, corporate acquisition and university Web page collections (Freitag, 1998). However comparative evaluation using these corpora has suffered from various inconsistencies in experimentation which clouds the ability to assess the relative performance of systems. Much of this difficulty is shared with general evaluation of ML systems; where experiments use different data processing, sampling and feature selection it is difficult to determine the critical factors effecting performance. Other issues are more specific to IE, such as the partial scoring of inexact identification of slot fillers or how to count multiple fillers for a slot.

In IE there are very few comparative articles in the sense mentioned in (Daelemans and Hoste, 2002); most of the papers simply present the results of the new proposed approach and compare them with the results reported in previous articles. There is rarely any detailed analysis to ensure that the same methodology is used across different experiments (Lavelli et al., 2004).

Appearing in Proceedings of the 22nd International Conference on Machine Learning, Bonn, Germany, 2005. Copyright 2005 by the author(s)/owner(s).

The Pascal Challenge on the Evaluation of Machine Learning for Information Extraction aims to address the issues raised above by providing a methodology and actual experimental setup to guarantee a reliable comparison of the performance of multifarious ML algorithms. In order to simulate a wide variety of application situations the study examines the way these algorithms behave (their brittleness) when the test data is sampled from a different timeframe and thus diverges from the training data. In addition, the study assesses the algorithms' performance with progressively increasing availability of training data, tracing the learning curve, and also the ability to exploit unannotated material to aid learning. Thus the challenge's intension is threefold:

1. The definition of a methodology for fair comparison of ML algorithms for IE.
2. To actually perform the evaluation in a controlled situation, reporting on the relative benefits of ML techniques to IE and identify future challenges.
3. To make publicly available an extensible testbed to enable comprehensive, comparable research beyond the lifetime of the current challenge.

The challenge methodology aims, as far as possible, to limit the variation in the dependent variables outside the ML technique used. This involves a standardised approach in three areas: the provision of the data, task specification and performance evaluation. The following three sections describe the approach taken in each of these areas[1]. Section 5 briefly describes systems submitted by the ten participants in the challenge and in Section 6 the comparative performance of the systems is analysed. The remaining section discusses conclusions from the challenge and further work.

2. Data

The domain selected for the challenge was the extraction of pertinent information from Workshop Call for Papers (CFP). The main reasons for the choice of domain was due to familiarity to the organisers (thus it is possible to interpret the documents without costly "experts"), there are good sources of data and it offers a range of difficulty. A further desirable feature is that, in common with many collections of documents, the text has a degree of semi-structured formatting; the learning algorithms should exploit such regularity.

A corpus of 1,100 documents was collected from various sources; it comprises of 850 Workshop CFP and 250 Conference CFP. The majority of the documents come from the field of Computer Science, due to the readily available archives, although other fields, such as biomedicine and linguistics, are also represented. Care was taken to ensure each document relates to a unique call. The documents are divided into three corpora:

- Training Corpus (400 Workshop CFP): The documents in the training corpus are randomly divided into 4 sets of 100 documents. Each of these sets is further randomly divided into 10 subsets of 10 documents. Each document relates to a workshop held between 1993 and 2000.

- Test Corpus (200 Workshop CFP): The documents in the training corpus relate to workshops held between 2000 and 2005.

- Enrich Corpus (250 Workshop CFP & 250 Conference CFP): The documents in the enrich corpus relate to workshops held between 2000 and 2005 & conferences held between 1997 and 2005.

Note that the training and test data is largely temporally distinct. Thus there will be less differentiation between the 4-fold cross-validation training and test data, as these are randomly sampled from the same timeframe. The Test Corpus may exhibit differences introduced by the temporal disparity providing a more rigorous test of a learning systems ability to generalise. As the Enrich Corpus offers documents taken from the same timeframe as the Test Corpus, it can potentially be exploited to uncover the temporal differences.

2.1 Preprocessor

The data was preprocessed using the GATE (http://www.gate.ac.uk/) system, which provides tokenisation, orthography, Part-Of-Speech tagging and named-entity recognition (Location, Person, Date, etc.) text features. The features selected are a fairly basic set in terms of linguistic processing. Future corpora could include other features such as lemmatisation or those derived from parsing which may be deemed necessary for IE tasks such as explicit relation extraction.

2.2 Annotation

The annotation exercise took place over roughly three months and involved a series of consultations between the challenge organisers and the annotators to determine the final annotations. The general methodology adopted was one of maximising the information provided by the annotations whilst minimising ambiguity during annotating. This meant that whilst it would have been desirable to extract the list of people on the organising committee, in the initial studies the annotators found very difficult to determine whether a name should or should not be included, and thus this annotation was removed from consideration.

For the final annotation exercise 10 people annotated an overlapping set of documents, with each document being annotated by two people. Conflicts were resolved by the

[1] The data, a more detailed description of the tasks and the evaluation server are available from http://tyne.shef.ac.uk/Pascal

overseeing annotator. An annotation tool (Melita[2]) was used to aid the process, although all automatic pattern-matching was switched off, except for exact-string matching, in order that the data was not bias towards the matching algorithm.

Each document can have 11 annotation types; 8 relating to the workshop itself (name, acronym, homepage, location, date, paper submission date, notification date and camera-ready copy date) and 3 relating to the associated conference (name, acronym and homepage).

Table 1: Frequency distribution of annotations

ANNOTATION TYPE	CORPUS FREQUENCY			
	TRAIN	%	TEST	%
workname	543	11.8	245	10.8
workacro	566	12.3	243	10.7
workhome	367	8.0	215	9.5
workloca	457	10.0	224	9.9
workdate	586	12.8	326	14.3
workpape	590	12.9	316	13.9
worknoti	391	8.5	190	8.4
workcame	355	7.7	163	7.2
confname	204	4.5	90	4.0
confacro	420	9.2	187	8.2
confhome	104	2.3	75	3.3
TOTAL	**4583**	**100**	**2274**	**100**

Table 1 above shows the frequency distribution of the annotations for both the training and test corpora; as can be seen the two distributions are broadly similar. Note that, as not all workshops have an associated conference, the lowest proportions are observed for conference related annotations.

There are certain distinct differences in the types of annotation. For example, the CFP “important dates” (paper submission date, notification date and camera-ready copy date), are generally well prescribed by the surrounding text. Whilst the workshop and conference names are more defined by their position in the document and have a greater variation in length. Such differences will obviously influence the ability of the learning algorithms to identify given annotation types.

Each annotation (or slot) is defined by a start and end tag which are placed in a boundary between two tokens. The training data has approximately 430,000 tokens, which equates to the same amount of instances, 9,166 of these are positive instances of tag placement.

3. Tasks

For each task participants can only use the features provided by the preprocessor. They were encouraged to submit results not only for testing on the Test Corpus but also for the four-fold cross-validation experiment on the training corpus; with a 300 training, 100 testing document split using the partitions provided.

- Task1: Given all the available training documents (i.e. 300 documents for the 4-fold cross-validation and 400 documents for the Test Corpus experiment), learn the textual patterns necessary to extract the annotated information.
- Task2a (Learning Curve): Examine the effect of limited training resources on the learning process by incrementally adding the provided subsets to the training data. Thus there are 9 experiments; for the four-fold cross-validation experiment the training data has 30, 60, 90, 120, 150, 180, 210, 240 and 270 documents, and for the Test Corpus experiment the training data has 40, 80, 120, 160, 200, 240, 280, 320 and 360 documents.
- Task2b (Active Learning): Examine the effect of selecting which documents to add to the training data. Given each of the training data subsets used in Task2a, select the next subset to add from the remaining training documents. Thus a comparison of the Task2b and Task2a performance will show the advantage of the active learning strategy.
- Task3a (Enrich Data): Perform the above tasks exploiting the additional 500 unannotated documents. In practice only one participant attempted this task and only to enhance Task1 on the Test Corpus.
- Task3b (Enrich WWW Data): Perform either of the above tasks but using any other (unannotated) documents found on the WWW. In practice only one participant attempted this task and only to enhance Task1 on the Test Corpus.

4. Performance Evaluation

The performance of the systems was evaluated using the MUC scorer (Douthat, 1998). Each system was evaluated on its ability to identify every occurrence of an annotation and only exact matches were scored. Performance is reported using standard Information Extraction (IE) measures of Precision, Recall and F-Measure. The systems overall performance is calculate by micro-averaging the performance on each of the eleven slots.

[2] Available at http://nlp.shef.ac.uk/melita

Table 3: Summary of features of participants' systems

PARTICIPANT	NO. OF SYSTENS	ML ALGORITHM	TOKEN WINDOW	TAG CONTEXT	FEATURE SELECTION	INSTANCE SELECTION
Amilcare	2	LP^2	5	Adjacent tags		
Bechet	2	HMM	10			
Canisius	1	SVM, IBL	5		Freq & Info-gain	
Finn	1	SVM	4	Start/End	Info-gain	Yes
Hachey	1	HMM	5			
ITC-IRST	3	SVM	10			Yes
Kerloch	2	HMM	15			
Stanford	1	CRF	4			
T-Rex	2	SVM	6		Info-gain	
Yaoyong	3	SVM	10			

5. Systems

The challenge attracted ten participants[3]; each participant was entitled to submit up to 3 systems and, in total, 18 systems entered in the challenge. Table 2 shows the number of systems entered for each of the tasks, for both the 4-fold cross-validation and Test Corpus experiments. Whilst Task1 and Task2 attracted sufficient systems to provide comparative evaluation, Task3 did not. It is hoped that this task will receive greater attention in the future.

Table 2: Number of system submitted for each task

TEST DATA	TASKS				
	1	2a	2b	3a	3b
4-fold	14	8	4	0	0
Test Corpus	17	10	5	1	1

Table 3 below gives a summary of each of the systems[4], the main features of the system shown are:

- ML Algorithm: Five participants employ Support Vector Machines (SVM) as the learning algorithm, whilst four participants use Hidden Markov Models (HMM) (Condition Random Field (CRF) is a form of Markov Model) which are widely used linguistic modelling, and one participant uses a rule induction technique (LP^2).
- Token Window: The size of the token window ranges from 4 to 15 tokens. Note that a token window of 4 means a left/right context of 8 tokens.
- Tag Context: Two participant systems consider other tags in tag placement; the Finn system considers annotation start tags in the context of end tags and vice-versa, whilst the Amilcare system considers tags in the context of all observed adjacent tags. In general most of the systems learn each tag separately. All of the systems use some form of conflict resolution in the final selection of the tags to use for annotation. Currently there is no research into the importance of this tag conflict resolution in IE.
- Feature Selection: Three participants employ feature selection methods removing low frequency or low information tokens.
- Instance Selection: Two participant systems use instance selection methods to reduce the number of negative instances, removing up to 50% of the instances to alleviate the class imbalance and to speed up processing.

[3] Another participant entered 3 systems which used a different pre-processor; these results have been excluded from this analysis.

[4] For a more comprehensive description of the systems see the Challenge web page at http://tyne.shef.ac.uk/Pascal/

Table 4: Performance on Task1 (Test Corpus) on each slot for systems providing at least one highest F-Measure

PARTICIPANT	MEASURE	WORKSHOP								CONFERENCE		
		NAME	ACRO	DATE	HOME	LOCA	PAPE	NOTI	CAME	NAME	ACRO	HOME
AMILCARE	PRE	0.656	0.887	0.769	0.864	0.621	0.876	0.889	0.876	0.792	0.922	0.656
	REC	0.241	0.844	0.632	0.619	0.402	0.851	0.889	0.865	0.422	0.888	0.28
	FME	0.352	**0.865**	0.694	0.721	0.488	**0.864**	**0.889**	**0.87**	**0.551**	**0.905**	**0.393**
YAOYONG1	PRE	0.629	0.738	0.81	0.656	0.611	0.719	0.867	0.764	0.649	0.619	0.368
	REC	0.539	0.523	0.666	0.87	0.674	0.763	0.821	0.736	0.411	0.348	0.093
	FME	0.58	0.612	0.731	**0.748**	0.641	0.74	0.843	0.75	0.503	0.445	0.149
STANFORD	PRE	0.618	0.806	0.822	0.678	0.737	0.747	0.87	0.777	0.643	0.576	0.389
	REC	0.576	0.358	0.693	0.665	0.576	0.68	0.774	0.791	0.4	0.428	0.093
	FME	0.596	0.496	**0.752**	0.671	0.647	0.712	0.819	0.784	0.493	0.491	0.151
YAOYONG2	PRE	0.713	0.796	0.838	0.734	0.717	0.767	0.943	0.845	0.775	0.634	0.455
	REC	0.437	0.481	0.586	0.679	0.612	0.636	0.784	0.669	0.344	0.278	0.067
	FME	0.542	0.6	0.69	0.705	**0.66**	0.696	0.856	0.747	0.477	0.387	0.116
ITC-IRST	PRE	0.852	0.733	0.85	0.672	0.812	0.841	0.921	0.911	0.795	0.667	0.556
	REC	0.539	0.259	0.451	0.419	0.406	0.617	0.795	0.687	0.344	0.235	0.067
	FME	**0.66**	0.383	0.589	0.516	0.542	0.712	0.853	0.783	0.481	0.348	0.119

6. Experimental Results

The following section summarises the results from the Pascal Challenge[5]. In general the results are only provided for experiments on the unseen Test Corpus; as the use of this data was mandatory for participants.

6.1 Task1

Nine participants submitted 17 systems for Task1; Figure 1 shows the performance of the best system for each participant.

As can be seen in Figure 1, there is a tendency to favour precision over recall in all but one system. This is probably due to IE systems, in general, being aimed at applications which place a higher cost on false positives. It is interesting to note that the top three systems (in terms of F-Measure), Amilcare, Yaoyong and Stanford, use very different ML techniques; rule induction, SVM and CRF, respectively. Also, although five of the systems use SVM, there is a considerable variation in their performance. Thus no clear advantage is observed simply by the adoption of a given ML technique.

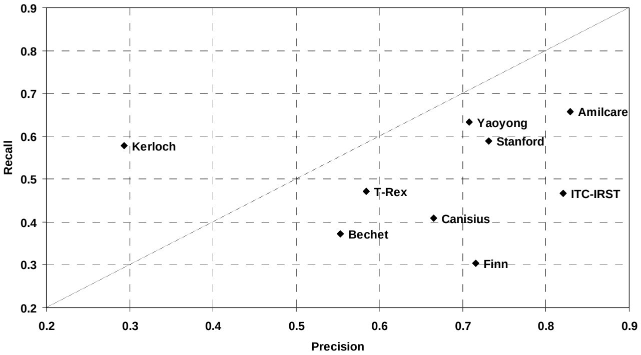

Figure 1: Precision & Recall on Task 1 (Test Corpus)

Figure 2 shows the change in performance for the seven participants who submitted systems for both the 4-fold cross-validation and the Test Corpus. Care should be taken when drawing conclusions from comparing these experiments as both the training and test sets differ. However it is interesting to note that the ITC-IRST, Finn and Canisius systems suffer a large fall in performance when using the Test Corpus; all three systems use SVM techniques. This fall in performance indicates that these

[5] The results are available in full at http://tyne.shef.ac.uk/Pascal/results.php

systems have a tendency to over-fit the training data and thus are more brittle to changes between the training and test data. The other system which uses SVM, Yaoyong, does not exhibit such a large fail in performance which is possible due to the adoption of a more robust type of SVM (i.e. one using uneven margins).

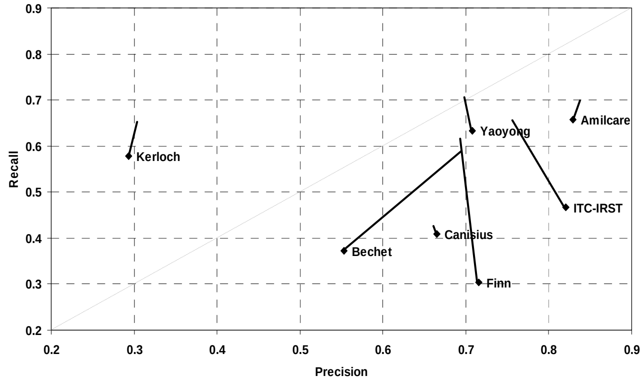

Figure 2: Performance change from 4-fold to test corpus

There is a considerable variation in the ability of all the systems to identify certain slots. Table 4 shows the five systems which provide the highest F-Measure on at least one slot. Amilcare achieves the highest on 7 out of the 11 slots and the other four systems achieve the highest for one slot each; Yaoyong system 1 (workshop homepage) and 2 (workshop location), ITC-IRST (workshop name) and Stanford (workshop date).

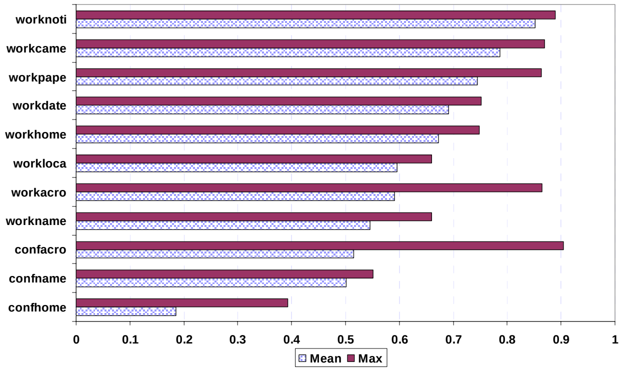

Figure 3: Slot F-Measure for the best systems

Figure 3 summarises the results in Table 4, showing the mean and maximum F-Measure for these five systems. When considering the mean F-Measures, the best performance is observed (as might be expected) on the four workshop dates; these have a well-defined format and are highly prescribed by the surrounding text. The worst performance is on the three conference annotations, which have a relatively low representation in the data. However, looking at the maximum F-Measure for the acronyms shows that Amilcare performs significantly better than all the other systems. This is possible due to Amilcare being the only system which considers the identification of annotation tags in the context of the other tags, potentially an important feature in identifying acronyms. A similar performance difference is observed for the conference homepage annotation. However the Amilcare system provided the worst F-Measure of the five systems on the workshop name and location, showing its techniques do not guarantee good performance on all slot types. Future improvements will have to concentrate on identifying the relative benefits of different techniques at identifying different annotation types. Unfortunately no statistics were kept for inter-annotator agreement. In retrospect this would have been very useful in comparing the relative performance of the machine learning algorithms against the annotators for each of the slot.

6.2 Task2a: Learning Curve

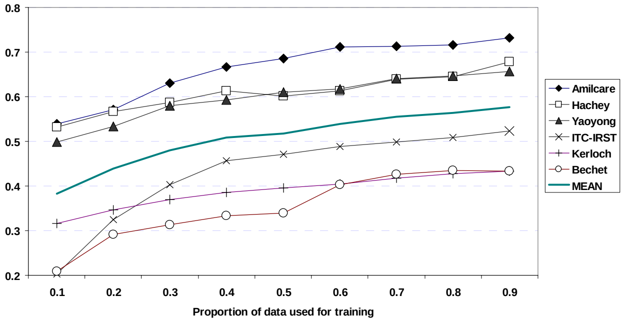

Figure 4: Learning Curve F-Measure

Figure 4 shows the F-Measure for the six systems which submitted results for the learning curve experiment on the test corpus. Each point on the graph represents an increase of one tenth of the training data. The systems can be divided into the three above the mean and the three below the mean F-Measure. The Amilcare system provides the highest F-Measure throughout the learning curve and shows a significant increase when using more than half of the training data. The Hachey system uses a similar technique to the Stanford system and, as for that technique in Task1, produces good results. It is interesting to note that the performance of the ITC-IRST system is most affected by low amounts of training data.

Figures 5 & 6 show the Precision and Recall, respectively, for the six systems. This shows that the systems pursue different strategies in terms of the performance measures. Both the Amilcare and ITC-IRST systems maintain a high Precision throughout the learning curve, indeed they mirror each other’s performance. The difference in performance of these two systems is entirely due to a difference in Recall. Indeed there is generally more variation in Recall than in Precision for all the systems, except the Kerloch system which greatly favours Recall over Precision. As was stated above this is partly due to IE systems being aimed at applications which place a higher cost on false positives. If the system’s object was

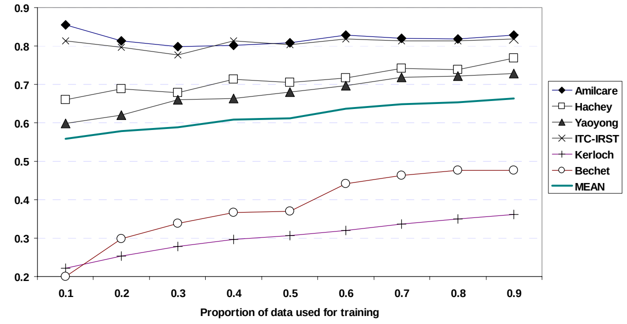

Figure 5: Learning Curve Precision

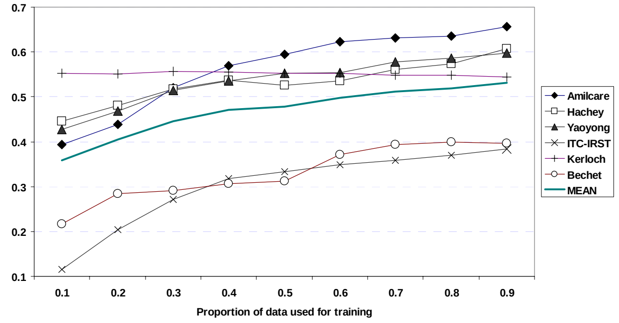

Figure 6: Learning Curve Recall

merely to maximise F-Measure, advantage could be gained from increasing Recall at the expensive of some Precision, especially when lesser amounts of training data are available, even more so for systems which exhibit a large Precision/Recall imbalance such as the ITC-IRST system.

6.3 Task2b: Active Learning

Three participants entered 5 systems for the Active Learning experiment; Amilcare, Hachey and the three Yaoyong systems. All three systems use some form of divergence to determine the appropriate documents to add to the training set. Hachey measures the divergence between the tags inserted in the potential training documents by two different classifiers. Amilcare applies a measure of the divergence between the expected and inserted tags in a document. Whilst Yaoyong measures divergence between an example subset sampled from the documents.

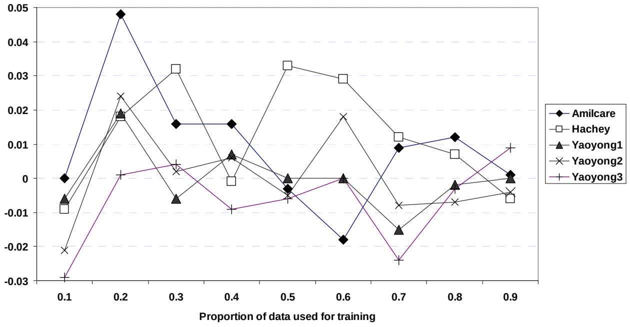

Figure 7: Active Learning Improvement in F-Measure

Figure 7 shows the change in F-Measure between the Learning Curve and Active Learning experiments. As can be seen no single system shows any clear advantage, although the Amilcare and Hachey systems provided the greatest improvements when using their active learning strategies.

The Hachey system provides the most consistent improvement in F-Measure, with the highest increase observed in the mid-range of training data (except for an unfortunate blip at 0.4 of the total training data). It would be expected that for low amounts of training data the learned classifiers do not have enough information to make reasonable judgements on the potential of the remaining documents, whilst when most of the data is used for training there are less potential documents from which to select thus less chance of improving over a random selection. Confusingly Amilcare's performance appears contrary to this intuition, with the improvements at low and high, rather than mid, amounts of training data. Given the inconsistent results and the small improvements (although the largest gain does represent 10% a fall in error rate) more work needs to be done on the active learning task.

6.4 Task3: Enrich Data

Unfortunately only two systems were submitted from the Task3 experiments. The Stanford system performed worse when using the Enrich data. The Amilcare system specifically aimed to improve its performance on the workshopname and indeed some improvement in Recall and F-Measure for this slot was observed, however overall performance was not changed.

7. Conclusion

This paper reports on the initial evaluation of the Pascal Challenge; whilst it is difficult to draw concrete conclusions without a more in-depth comparison of the systems' features, a number of interesting observations can be made from their performances.

The top three systems, in terms of F-Measure, submitted for Task1 use very different learning algorithms (rule induction, SVM and CRF), and systems which use similar techniques provide diverse performance. Thus the adoption of a ML algorithm, in itself, does not provide a guaranteed advantage.

The 4-fold cross-validation experiments produced more consistent performance between the systems. However, a number of systems exhibited a degree of brittleness when testing on the Test Corpus which indicates a tendency to over-fit the training data. Given that systems using similar techniques exhibited more robust behaviour this indicates that the fall is due to system parameterisation rather than an intrinsic feature of the ML algorithm used.

When examining the performance on individual slots a considerable amount of variation is observed. Whilst the Amilcare system provides the highest F-Measure for 7 out of the 11 slots it also performs poorly on two of the others. Future work will need to determine particular annotation types (i.e. those defined by internal syntax, surrounding text, document position, etc.) and the features of ML system which allow it to perform well on a given annotation type. For example, as Amilcare uses other tags to identify tag placement, its good performance is partially due to the relatively large number of slots in this experiment, especially if the annotation placement has a regular structure in each document.

Given the differences between the systems which perform well it is hoped that further examination will enable the development of a combined system to improve performance further by exploiting the synergistic features of the techniques used, which provide good performance on different slot types.

The (Task2a) Learning Curve experiment shows that to optimise performance (especially in terms of F-Measure) the Precision/Recall balance should be considered in terms of the amount of training data available. This is particularly evident for the systems which strongly favour Precision over Recall.

The (Task2b) Active Learning experiment did not provide clear results although both the Amilcare and Hachey approaches showed some advantage. Future work focusing on this task will implement the two strategies using a single system to better determine their effectiveness.

Unfortunately there was not sufficient interest in the (Task3) Enrich data experiments to draw any conclusions.

One of the abiding outcomes of the challenge is that document annotation is both time-consuming and tedious. Whilst such an exercise is very useful to evaluate systems for ML research, for “real-world” application the annotation exercise will have to show that the significant cost of annotation is outweighed by the benefits derived. Also Google currently indexes over 8 billion pages, manually annotating any reasonable proportion of these for the Semantic Web is an unfeasible task. Thus it is crucial that the ML community examines ways of using unannotated data, in semi-supervised learning methods, to provide effective IE.

It is hoped that the Pascal Challenge data and methodology presented in this paper will receive further attention and will be augmented with other data sets and evaluation tasks providing a comprehensive testbed for ML approaches to IE.

Acknowledgments

This work was partly supported by the Pascal Challenge Network. F. Ciravegna, C. Giuliano, N. Ireson, A. Lavelli and L. Romano are supported by the IST-Dot.Kom project (http://www.dot-kom.org), sponsored by the European Commission as part of the Framework V (grant IST-2001-34038).

References

Califf, M. E., (1998). *Relational Learning Techniques for Natural Language Information Extraction.* Doctoral dissertation, University of Texas at Austin.

Daelemans, W. & Hoste, V., (2002). *Evaluation of machine learning methods for natural language processing tasks.* In Proceedings of the Third International Conference on Language Resources and Evaluation (LREC 2002). Las Palmas, Spain.

Douthat, A., (1998). *The message understanding conference scoring software user's manual*. In Proceedings of the 7th Message Understanding Conference (MUC-7).

Freitag, D., 1998. *Machine Learning for Information Extraction in Informal Domains*. Doctoral dissertation, Carnegie Mellon University.

Hirschman, L., (1998). *The evolution of evaluation: Lessons from the Message Understanding Conferences.* Computer Speech and Language, 12, 281--305.

Lavelli, A. Califf, M.E., Ciravegna, F., Freitag, D., Giuliano, C., Kushmerick, N., Romano (2004) *IE Evaluation: Criticisms and recommendations.* In *Proceedings of the AAAI-04 workshop on Adaptive Text Extraction and Mining (ATEM 2004)*, San Jose, California