

The Scamseek Project – Text mining for Financial Scams on the Internet

Jon Patrick

Sydney Language Technology Research Group
School of Information Technologies
University of Sydney
and
Capital Markets Co-operative Research Centre
jonpat@it.usyd.edu

Abstract

The Scamseek project, as commissioned by ASIC has the principal objective of building an industrially viable system that retrieves potential scam candidate documents from the Internet and classifies them as to their potential risk of containing an illegal investment proposal or advice. The project produced multiple classifiers for different types of data, and achieved higher than expected performance statistics on classifications. The development of the system required the solution of two major problems in document classification, namely accurate identification of classes with very small footprints, <.1%, and classification using meaning intention rather than word strings. The approach taken used Systemic Functional Grammar to model the semantics of the scam classes and used unigrams with significant language pre-processing to assist in separating irrelevant documents. Litigations have been initiated by ASIC from classifications made by the system¹. ASIC operates the system on a 24/7 basis. The estimate of savings in human effort in its monitoring role is the order of 100-fold. The estimate in savings to the community cannot be estimated readily but is likely to be of the order of tens of millions of dollars.

1 Introduction

Text Classification has a tradition of treating documents to be processed as a “bag-of-words” or n-grams, that is, the words or word groups within a text are treated as independent and uncorrelated with each other. Such a model of language is exceedingly simple but has been proven to satisfy many researchers.

The Scamseek project has sought to separate itself from the bag-of-words tradition of text classification. In particular the model of language used in the project was Systemic Functional Grammar (SFG) [1]. This model takes the position that language usage is a matter of choice set in a configuration of hierarchically layered strata of graphetics, graphology, lexicogrammar, semantics and context. Systemic grammar is a network of “systems” that interact with each other rather than a set of rules as with generative grammar.

In computing classifications, texts of a given class, apart from a small number of very common topic words, are more closely related by the minute intricacies of a weak network or chains of correlations that persist at low levels across small sub-sets of the class and the persistent meaning they represent, rather than by large persistent clusters of resoundingly dominant word sets that trumpet the presence of their class. In this case, the use of SFG states that the social context of the text’s composition dictates choices of meaning intentions which in turn influences the form of the text. The linguist’s task is to make sense of the decision making process and render it in a manner that might be suitable for computation. The computational linguist then has to convert the linguist’s model into a computable representation in the context of his target analytical methods which in this case is the procedures of machine learning.

¹ See ASIC Media Release 04-178: Grammax Investment Club operating unlicensed investment club is believed to have moved over \$10M overseas in the prior year.

2 Scamseek Project Specifications

The Scamseek project was devised in two stages. The first stage had the aim of producing a production system for retrieving and classifying web pages. The client provided a manually classified corpus of about 8000 documents. The delivery time was 6 months from project commencement. The project team consisted of 1 linguist, 1 computational linguist and 3 software engineers.

The second phase ran for 9 months to 30 June 2004 and had the objectives of improving the accuracy of the web page classifier and the development of new classifiers for a number of other Internet data types. In Phase 2 the contract had more data sources to be scrutinized, entity recognition and performance requirements, plus in each case retrieval mechanisms had to be developed and for one source the corpus had to be compiled. The team was expanded with another linguist and computational linguist. Other part-time staff and consultants also made contributions.

3 Project Operations

The Scamseek team was set up with a clear operational model that was effective throughout the life of the project, but adapted as work patterns developed to maturity. The operational model represented the task as consisting of 4 groups with different job functions; the client, linguists, computational linguists, and software engineers. The client was in contact with the linguists to deal with the classification of data. The linguists had the task of preparing the linguistic models of the data and passing that to the computational linguists who in turn had to prototype computational methods to compute the language models and devise machine learning experiments to optimize the classifiers. The computational linguists would pass their prototype code to the software engineers for efficient industrial quality implementation. This configuration operated effectively throughout the project development phases.

4 Computational Linguistic Research Topics

4.1 Linguistic vs. administrative classes

One of the early problems to emerge with the project requirements was the difference between the classification scheme of the client designed to conform to an administrative perception, that is, there are three types of scam under the law (unlicensed advisors, unregistered fundraising, and share ramping), and the linguistic manifestation of those three types. After a significant amount of linguistic analysis a set of registers (scam document sub-types based on their linguistic characteristics) were created representing subdivisions of the 3 scam types. This configuration was changed a number of times and expanded in phase 2 when the client opted to create different subdivisions in the data. The 3 scam types were treated as 1 document class with sub-classes or registers and the remaining part of the corpus was classified by the client into three more classes, Other-Agency-Scams, Scam-like and Irrelevant. These classes were also divided into registers to capture the linguistic variation within the classes. In all, over 50 registers were created with more than 20 in the scam class.

4.2 Linguists' compilation procedures

The linguists conducted their work by a two part strategy. Firstly they read the documents and collated them into registers and at the same time created register descriptions. In the latter stages of the work the linguists were able to scrutinize documents that were incorrectly classified and attempt to adjust their ontologies for both the register of the misclassified document and the register it was computed to belong to.

4.3 Specification of linguistic model

From the outset a decision was made to use a strong linguistic model to govern the direction of the work. This position was taken because the problem of identifying specialist content very thinly distributed and written in a particular manner was not believed accessible automatically by any other strategy.

The development of the linguistic model of the registers went hand in hand with the creation of the registers. The linguists read the documents and developed small scale characterizations of them. As the

work developed documents of similar ilk were paired together until all scams were assigned to a register and described for their features of differentiation and “scaminess”.

It was decided to represent register descriptions in an ontology rendered by XML. The upper part of these ontologies conformed to the SFG grammar as generally published, and the lower part is an ever increasing delicate rendition of the detail of the relevant content in the documents of the register. An objective of the work that was never achieved was the capacity to view a document and render it with an overlay of a register ontology and allow the linguists to do their extraction directly from the document image on the screen rather than their laborious hand collation.

The register descriptions and allocations resulted in a final list of more than 20 scam registers and 40 other registers spread across the 4 classes. At the same time the linguists with increasing understanding of the nature of the corpus advocated that greater amounts of the most structural components of the SFG model needed to be introduced into the assessment. Hence, the SFG networks for specific grammatical concepts were introduced as separate ontologies.

4.4 Small footprints of target classes

The scam class as a whole represented less than 2% of the corpus in phase 1, however with the development of the register model of the data there became registers with sizes <.1%. This represented significant problems with underrepresented classes and led to an experimental program to alleviate its effects. In phase 2 the client changes doubled the size of the scam class, however it also triggered a need to redevelop the whole set of scam registers to disperse a heterogeneous register into a homogeneous set. This also caused more small registers to be created and thereby not particularly improve the overall problem of the small footprints of registers.

Ultimately the small footprint problem was resolved by the development of the SFG ontologies for each register. The amount of effort spent on each individual register was related to some degree to the difficulty of separating it from other registers and therefore de facto addressed this problem.

4.5 Hybrid Language Model

The linguistic model can be considered to be designed in two parts. The first part was the register descriptions of the most important subdivisions of the corpus either on client needs basis, the scams, or for processing efficiency, that is, the largest groups of non-scam documents. The second part was the collection of all the non-scam classes and the completely irrelevant material which was the largest class (about 60%). These parts were in turn grouped into the four classes of the client. The task required was to develop classifiers for the major classes as well as the scam registers. The solution chosen was to develop ontologies for separating registers and use an n-grams approach to support the separation of the larger classes. This led to multiple lines of experimentation, namely, developing language processing functions for the ontologies, exploring the optimum feature selection for the classes independently of the registers, and, finally bringing the two solutions together to construct a combined classifier.

The SFG ontologies consisted of words and phrases from the texts organised in an SFG hierarchy, the upper parts reflect the theory of SFG and the lower parts represent the greater delicacy of the documents under analysis. The leaves of the ontologies were initially strings chosen from the texts classified in the given register. Over time the ontologies were developed and they became rich representations of the total document collection in the respective registers.

4.6 Machine Learning – Classifier Development Programme

The program for optimising the classifiers in the first phase concentrated on the problem of developing a single optimal classifier for web pages. In phase 2 separate classifiers were required for each data source and so experiments followed multiple strategies for all sources. SVMs were quickly identified as the best classifier for the data set.

Investigations were made of the selection of features from the collection of registers vis-à-vis the set of classes. While there was a significant overlap in the features chosen by an Information Gain metric there

was still an appreciable improvement by using the feature set chosen from the registers in the small classes and between the classes in the large classes thus giving a blend of feature selection methods.

Selection of features from register ontologies required an extensive series of experiments. The register ontologies performed well independently of other feature sets once they were developed to a very mature stage. Later the four grammatical ontologies were added which made various levels of contributions in intriguing ways. For example the Modality grammatical ontology performed particularly poorly by itself on some occasions classifying no documents correctly, yet when it was added to other models it consistently improved their scores. This result indicated clearly that there is an interaction effect within the grammatical ontologies that exploits a weak correlation not recognizable within the individual systems themselves. It is their union with other systems that created their strength. This result is entirely predictable with the SFG model of language and further justifies its use for this task.

4.7 Mapping features to attributes.

We use the terminology of *feature* for the linguistic phenomena that is the target of interest, and *attribute* for its numerical instantiation, and *mapping* for the computational transformation of the frequency count of the feature into its attribute representation. This distinction is unimportant for n-gram methods as the difference between features and attributes is inconsequential since the mapping transformation is trivial. This position cannot be taken in our work as the mapping transformation is different depending on the theoretical origin of the feature.

Feature representation for the ontologies was created by accumulating scores up the ontology tree. SFG in principle argues that the language is choice and therefore the important aspect of understanding the difference between two texts is the choice made by the authors. Hence by this principle the relative proportions of the choice to use one part of the tree over another should be the best differentiating feature. This is the case for the grammar ontologies but however does not apply to the register ontologies. The reason is that the register ontologies represent the most common semantic phenomena of a given register type, rather than choices between competing ways of expression. Hence, the attributes of domain register features are mappings to accumulative scores which are unnormalised, and grammar register features are mapped to proportional scores, whereas the n-gram word tokens are frequency counts normalized by document length.

5 Software Engineering Issues

5.1 Regulating experimental practices.

In the background, the engineers created an architecture that was intended to automate as much as possible the roll-out of the production system. As the production system required the use of the specific language processing methods and parameters of the very best machine learning experiment, the experimental programme had to be fully integrated into the engineers' software production process. Hence all computational linguists were coerced by the engineers into producing their code within the CVS system. This ensured that all the computational linguists' code was designed, at least architecturally, to fit the current production system.

5.2 Automatic roll-out of production classifiers.

The integration of the computational linguists work into the CVS system ultimately enabled the complete automatic generation of the production system merely by supplying the number of the experiment which had produced the "best" classifier. With this number all the language models, all the language processing code and all the background system code (database schema, user interfaces, data retrieval, etc.) were automatically assembled into a single system for shipping to the client.

5.3 Use of Open Source Software

The project used open source software for all aspects of its operations. The underlying operating system was Linux. Programming was in Python and interfaces were constructed using GTK with GLADE and CVS was used for code management and Bugzilla used for software revision requests. Postgres was used

for database management, and all machine learning experiments used the Weka suite. The only purchased software was XMLSpy to manage the descriptions of the SFG ontologies.

6 Results –Phase 1

The results of the first phase of the web page classifier for the scam classes as applied to an audit corpus have performance values of: Precision=.75, Recall=.41, and F-value=.53 and are to be contrasted with the laboratory results of the completed system on the training corpus using 10-fold cross validation of: Precision = .74, Recall = .35, F=.48. A baseline of 1000 single words has F=.21. ASIC was entirely satisfied with these results and made a commitment to a larger project in Phase 2.

This corpus was unseen by the development team and made available by ASIC at the time of delivery of the system. The processing was conducted by the ASIC staff and the project team was given one week in which to request revisions to ASIC’s manual classifications. The Scamseek classifier in this instance identified 4 scams that had been manually misclassified by ASIC.

7 Results – Phase 2

The results of the second phase of the web page classifier for the scam classes applied to the corpus are presented in figure 1. ASIC was satisfied by the performance of the system in phase 1 not to require a second audit corpus assessment. Figure 1 provides results for 3 separate corpora, web pages as in the phase 1 experiments, and two other corpora developed for phase 2. The Web Pages result represents the system delivered to ASIC as of 30 June, 2004. The exact nature of the other corpora cannot be presented due to security obligations. The performance figures are determined by 10-fold cross-validation.

Figure 1. The performance results from the web pages classifier and 2 other classifiers for identifying scams on the Internet as delivered to ASIC.

	Web Pages	Corpus 2	Corpus 3
Precision	.744	.850	.852
Recall	.528	.834	.639
F-value	.618	.844	.730
Scam/non-scam texts	373/6391	686/1483	1395/13716

8 Conclusions

The Scamseek project is a success for ASIC in that it is operable 24 hours a day 7 days a week. In its first operational run it discovered an activity that has since been taken to the stage of litigation. The estimate of savings in human effort in its monitoring role is the order of 100-fold, as previously ASIC had to read 80 documents to find one of interest they now read 5 documents to find 4 of interest. The estimate in savings to the community by bringing speedier detection and intervention of scams cannot be estimated readily but is likely to be of the order of tens of millions of dollars. ASIC is not prepared to release all details about the technology but has released the following summary statement: “The Scamseek technology is deployed in

such a way that any scam proposal on any Internet channel that is generated in Australia or directed at Australians is highly likely to come under scrutiny”.

The research contribution has been significant in that it is the first project that has used Systemic Functional Grammar for automated text classification. Solutions to serious problems in practical text classification, namely unbalanced classes, and the integration of semantic and n-gram language models have also been developed.

The project has also made a significant contribution to the issues of software engineering in language technology in that it has shown that computational linguistics research can be performed in the context of reaching industrial objectives.

9 Acknowledgements

The following people worked on the Scamseek project and made contributions to the final solutions, Michele Wong, Kathryn Tuckwell, Stephen Anthony, Tim Yeates, Dr. James Farrow, Neil Balgi, Jian Hu, Carlos Aya, Will Radford, Mathew Honnibal, David Smoker, Naomi Carter. The following doctoral students contributed to the work Maria Couchman, Casey Whitelaw, David Bell. The following people acted as advisors: Prof Christian Matthiessen, Prof Jim Martin, and Prof Vance Gledhill. Participating organizations were Australian Securities & Investment Commission (ASIC), Capital Markets Co-operative Research Centre (CMCRC), University of Sydney, Macquarie University, and the Australian Centre for Advanced Computing and Communications (AC3).

References

1. Halliday, M. (1994). *Introduction to Functional Grammar*. 2nd Edition. London: Arnold.