Using Text Mining and Natural Language Processing for Health Care Claims Processing

Fred Popowich

Axonwave Software
Suite 873, 595 Burrard
PO Box 49042
Vancouver, BC
CANADA V7X 1C4
popowich@axonwave.com

School of Computing Science
Simon Fraser University
8888 University Drive
Burnaby, BC
CANADA V5A 1S6
popowich@sfu.ca

ABSTRACT

A health care claims processing application is introduced which processes both structured and unstructured information associated with medical insurance claims. The application makes use of a natural language processing (NLP) engine, together with application-specific knowledge, written in a concept specification language. Using NLP techniques, the entities and relationships that act as indicators of recoverable claims are mined from management notes, call centre logs and patient records to identify medical claims that require further investigation. Text mining techniques can then be applied to find dependencies between different entities, and to combine indicators to provide scores to individual claims. Claims are scored to determine whether they involve potential fraud or abuse, or to determine whether claims should be paid by or in conjunction with other insurers or organizations. Dependencies between claims and other records can then be combined to create cases. Issues related to the design of the application are discussed, specifically the use of rule-based techniques which provide a capability for deeper analysis than traditionally found in statistical techniques.

Keywords

Information extraction, information retrieval, categorization, pattern matching, other party liability indicators.

1. INTRODUCTION

Text mining is concerned with the detection of patterns in natural language texts, just as data mining is concerned with the detection of patterns in databases. Information processing applications can benefit from having access to both structured information, as found in databases, along with unstructured information, traditionally found in documents or unstructured text fields within databases. When accessing this textual information, applications can also benefit from a more detailed linguistic analysis of the text, as opposed to a shallower "word based" analysis. There are a wide range of techniques that can be applied to analyzing these natural language texts, as reflected in the considerable amount of research in the field of natural language processing [5].

As noted in [9], document categorization is one of the most popular applications of text mining. In this paper, we consider the analysis of textual information and categorization in the context of an application for processing health care claims. In this context, the textual information is dominated by descriptions entered by call centre operators, and by comments associated with individual claims and/or cases. The texts that are encountered are highly constrained with respect to their semantics. These texts reference entities and relationships contained in standard treatment and diagnosis taxonomies.[1] The texts themselves may be highly fragmented and may make use of numerous abbreviations and acronyms. As a result of the constrained nature of the textual information, we are able to leverage the information contained in standard treatment and diagnosis taxonomies, together with concept taxonomies specific to other-party-liability and to fraud-and-abuse, to provide indicators that can be combined with structured information associated with insurance claims to obtain more effective identification of claims involving third party liability, subrogation, or fraud and abuse.

The process of automated medical claims auditing is outlined in Figure 1. It illustrates how the output of a natural language processing system, which performs detailed linguistic analysis using domain specific information in the form of Concept Taxonomies, is then used by a mining system to produce output which is then subjected to human analysis.

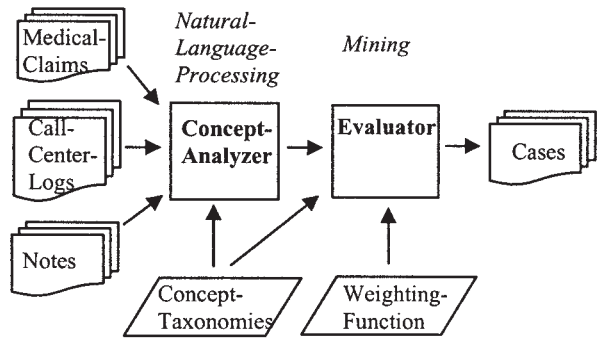

Figure 1. Automated Medical Claims Auditor.

[1] The disease and treatment taxonomies are discussed in [10].

We start, in section 2, by providing an introduction to NLP and outline the NLP techniques that we will be using. In section 3, we then introduce a specific application in the area of health care claims processing, and we see how the NLP techniques can be used to identify indicators of claims that may require detailed human investigation. In section 4, we look at how the concept taxonomies required for the claims processing system can be developed. Then in section 5 we examine how these indicators can be used by text and data mining techniques to detect patterns in claims, and score individual claims.

2. NATURAL LANGUAGE PROCESSING

2.1 Overview

Natural language processing (NLP) deals with the automatic processing and analysis of unstructured textual information. One direction of NLP research relies on statistical techniques, typically involving the processing of words found in texts [7]. Another approach makes use of rule based techniques, leveraging knowledge resources such as ontologies, taxonomies, and linguistic rule bases. Statistical human language processing systems require collections of training material which exemplify the desirable (and/or undesirable) relationships and dependencies. Subsequent modification of the system then requires some degree of retraining of the system. Instead of requiring training material, rule based techniques require knowledge in the form of on-line dictionaries, established linguistic theories, and they are able to leverage existing classification systems or taxonomic frameworks. NLP applications may make use of either or both of these techniques, and the decision of which technique to use is often dependent on the availability of training materials, external resources, and the actual text analysis tasks required in the resulting application.

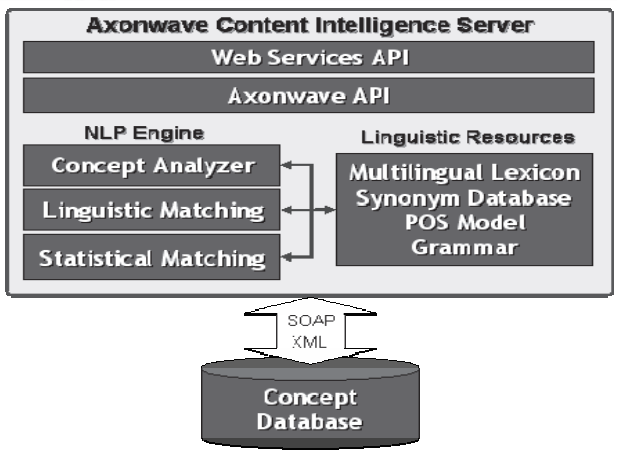

Figure 2. CIS Architecture.

2.2 Content Intelligence System

The Axonwave Content Intelligence System (CIS) contains core natural language processing systems that perform both rule-based and statistic-based NLP. The CIS is able to leverage existing knowledge sources, plus provide the capability for ordinary users to tailor or customize the knowledge base with concepts that are of interest to them. The general architecture of the system is shown in Figure 2.

The system makes use of a statistical tagger, rule-based partial parser, together with external resources including Wordnet [8].[2] The tagger and partial parser are robust, and able to deal with the often ungrammatical text found in call logs, which contains numerous instances of abbreviations and acronyms[3], as well as the more polished text found in medical services plan documents. The partial parser provides more information than just tagged words. It provides proper name identification, plus it determines the arguments and modifiers of relationships and entities found in a document (as appropriate).[4]

2.3 Concept Specification Language

The core technology concerns the matching of "Concepts" which are represented in a Concept Specification Language (CSL). CSL is used to specify rich linguistic patterns that incorporate as fundamental the notion of recursion (embedding) of patterns and various linguistic predicates.

CSL and concept matching are embodied in the CIS, which analyzes the structure of words, phrases and sentences (making use of general purpose linguistic rules and dictionaries). The first stage of analysis consists of abbreviation expansion and spelling correction, which is then followed by tagging and then partial parsing [1]. Specific information can then be extracted according to rules and concepts formulated with CSL which is organized within various taxonomies. CSL allows the definition of key concepts or terms; and the specification of the interrelationship among concepts in the form of multiple operators, such as OR, NOT, Precedes, Immediately Precedes, Is Related, or Causes; and also the formulation of advanced categories for concepts, such as whether a concept is a word, has synonyms, is a general or a specific term, etc.

To illustrate CSL, let us consider the definition of a concept which we will call AccidentsAndTrauma, which is intended to match a wide range of descriptions of different kinds of accidents or trauma that might be encountered in documents supporting

---

[2] One of the key challenges when using a resource like Wordnet, is to prevent overgeneration associated with inappropriate word senses and their associated synonyms. This issue is addressed in detail in [12], where there is a discussion on how Wordnet can be pruned for specific domains.

[3] The text analysis engine was originally designed to deal with poorly structured English containing numerous acronyms and abbreviations, like the text found in aviation safety reports [2]. Our approach has been to avoid "cleaning" the data, but instead providing the modules with enough knowledge so that they can deal with "dirty" data. For example, given a data collection, we perform a statistical analysis of the different abbreviations and acronyms encountered in a collection, and provide an appropriate semantics for these tokens. Some issues concerning acronyms are discussed in [13].

[4] As expected, the performance of the tagger can be improved by providing training data specific to the targeted domain and style of text.

medical insurance claims. In Figure 3 below, we define this concept as a disjunction of subconcepts.

```
concept AccidentsAndTrauma (
          %Trauma
        | %AccidentalFall
        | %Accident-Sports
        | %Accident-Involving-Children
        | %Accident-Auto
    )
```

Figure 3. A High Level Concept.

Each of the subconcepts will have its own definition, resulting in a rich hierarchical taxonomy of concepts (Figure 4). Specifically, a concept like AccidentalFall includes a subconcept SlippedOrFell, which itself has a subconcept FallFromDifferentLevel. The categories are not necessarily mutually exclusive. So for an accident taxonomy, an excerpt of which is provided in Figure 4, a given incident could be an accidental fall, and an accident involving children.

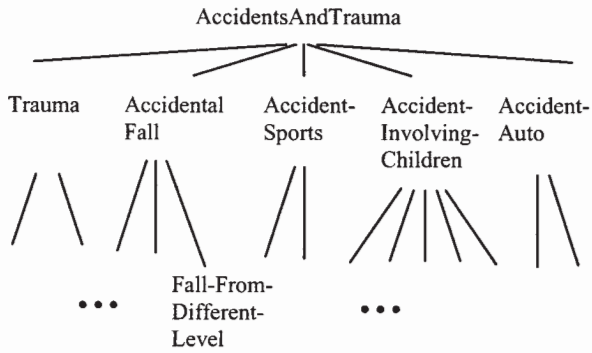

Figure 4. A Taxonomy.

The CSL for this final concept is given in Figure 5. Note that this concept contains individual words, (specifically "off", ""from", "to" and "feet"), which will match text that is linguistically linked (in this case, "Related") to a word or phrase that matches the SlippedOrFell concept. Alternatively, it will match a phrase in which a SlippedOrFell phrase is followed by the word "down" which is then followed somewhere later by a NOUN.

```
concept FallFromDifferentLevel(
   Related(
       %SlippedOrFell,
       (off | from | to | feet))
| (%SlippedOrFell &
   down &
   /NOUN )
```

Figure 5. A Low Level Concept.

This concept will match phrases such as "fell 15 ft" or "fell down a flight of stairs", and will annotate the text with the tag FallFromDifferentLevel. Note that there is no need to specify all the abbreviations for a word in the CSL, nor is it necessary to specify all of the synonyms. The CIS engine can automatically handle the different variations of a word. Once the text is tagged, the tags like these can then be used as indicators for the subsequent mining phase.

CSL can be viewed in some respects as a linguistic programming language, and from this perspective, it is similar to the declarative information analysis language (DIAL) described in [11] and used by Clearforest in their text mining applications. DIAL is presented as a rule-based information extraction language where "the pattern matching elements are either explicit strings found in the text (such as the word expression), a word class (a specific set of lexical terms), or another rule" [11, p. 848]. However, CSL provides a richer selection of matching elements, taking into account syntactic and semantic primitives. Furthermore, as will be seen in section 4, we can also use natural language processing techniques to assist in the creation of CSL.

3. INDICATORS IN DOCUMENTS

When a medical insurer is presented with a claim for a treatment in response to diagnosis, there is a large amount of information that might be relevant to the claim, beyond the codes and descriptions contained in the claim itself. Consider the situation where a patient is treated in the emergency room of hospital for a broken arm, which requires an initial examination, an x-ray, and the application of a cast. There will be charges associated with each of these aspects of the claim (also known as a line item), and there might very well be textual comments associated with line items, and in supporting documents. For example, there might be a note saying "patnt fell off desk while chnging light bulb at work", which could contain abbreviations, acronyms, and might even be ungrammatical. This information could provide evidence that the claim should be subject to workman's compensation rules, rather than being treated as a claim to be covered only by the insurance plan.

So, what kind of textual indicators are important when determining whether other parties should be partially responsible for covering the costs of claims? They are indicators which suggest that a claim falls into one of the following categories.

1. Commercial Coordination of Benefits
2. Medicare Coordination of Benefits
3. No-fault Recovery
4. Subrogation Recovery
5. Workers Compensation

Based on the rules that are used by claims examiners, we were able to construct a taxonomy of indicators that play a role in determining likelihood of one of these categories. The medical claims taxonomy contains approximately 3000 nodes and averages five levels deep. Associated with each of these indicators is a CSL specification that makes use of domain independent entities and relationships, combined with domain specific terminology.

The image in Figure 6 shows a collection of indicators extracted from medical call center notes.[5] The first note in Figure 6 shows two matches: one for the Trauma indicator, which matches "injury", and one for the Accident-Auto indicator, which matches

[5] Information in all images has been altered to protect the privacy of the individuals involved with the claims.

"fell off his motorcycle". The second and third note both contain evidence of a WorkersCompensation indicator, while the last note contains several indicators, one of which is SlipAndFall, which matches the text "slipped on the ice."

By using CSL to describe text indicative of given concept, we are able to take into account the high degree of variation encountered in the English text used to describe different circumstances and events. Specifically, one can specify entire classes of words, based on part-of-speech or based on meaning, rather than simple lists of words or strings. Additionally, one can specify constraints based on linguistic syntactic relationship (modifies) or even semantic relationships (cause/effect), rather than a simple proximity measure.

Since we are working in a highly constrained domain, it is possible to achieve very high levels of accuracy. Precision and recall measures are calculated on a regular basis for a selection of the thousands of indicators that are extracted from documents. Precision is calculated on a random selection of 100 matches taken from a corpus of customer service logs, and text fields from medical claims and management notes. For the Commercial and Medical Coordination of Benefits indicators mentioned earlier in this section, the average precision is 99%. Not all indicators are so accurate, though. The precision of the indicator for determining that a child is covered under more than one plan (Multiple Plan Child Coverage) is only 84%. The recall for an indicator is determined through a test procedure where a human evaluates documents containing regions of text that should match an indicator. They run the system on these documents to determine what percentage of these matches are found by the system. For Coordination of Benefits, the recall averages 85%, with the Multiple Plan Child Coverage indicator obtaining a recall of 81%.

The next step of the process is to use these indicators to determine which claims require further human investigation, and whether some claims can be combined together with supplemental information to form an actual case to be assigned to a human analyst. We achieve this by applying text-mining techniques to claims and documents annotated with indicators, rather than applying the techniques to just the original documents. At this time, we also leverage the structured information contained within the claims, such as the dollar value of the claim, claimant, zip-code, date, and so on. So, we are effectively performing traditional data mining on the structured information, which is augmented with the indicators extracted from unstructured text. We can perform clustering, clique analysis, outlier analysis, and many other techniques. However, given that the focus of the current paper is on natural language processing and unstructured text, in section 5 we will focus on the techniques that concentrate on the use of the indicators identified by the CIS engine. First, we will look in detail at how CSL can be created using natural language processing techniques.

Notes

Filter: ID = Find Clear

S	ID	Type	Indicator	Context
☐	103657-Sep 7 2004 2:40PM-MEM	CustServ	Accident Auto Trauma	Injury was swelling / bruising no breaks. Member fell off his motorcycle and had an elbow injury.
☐	107465-Oct 19 2004 1:12PM-Com	CustServ	WorkersComp	Comment states that it could be work related.
	88741-Mar 25 2004 10:02AM-400	CustServ	Trauma WorkersComp WorkersComp	PARTY REPSONSIBLE FOR THESE CLAIM. SHE SAID SHE HURT HERSELF AT WORK. THEN SHE TRIED TO FILE FOR WORKMAN'S COMP., BUT SHE WAS DENIED. THE MEMBER WANTS A CALL BACK.
☐	88168-Mar 18 2004 11:04AM s	CustServ	Accident InjuryThirdPartyProperty SlipAndFall	I spoke with member who informed me that in Feb 2004, he was at the GSB FC picking up his x-rays when slipped on the ice in the parking lot

Figure 6. Indicators Found in Customer Service Notes.

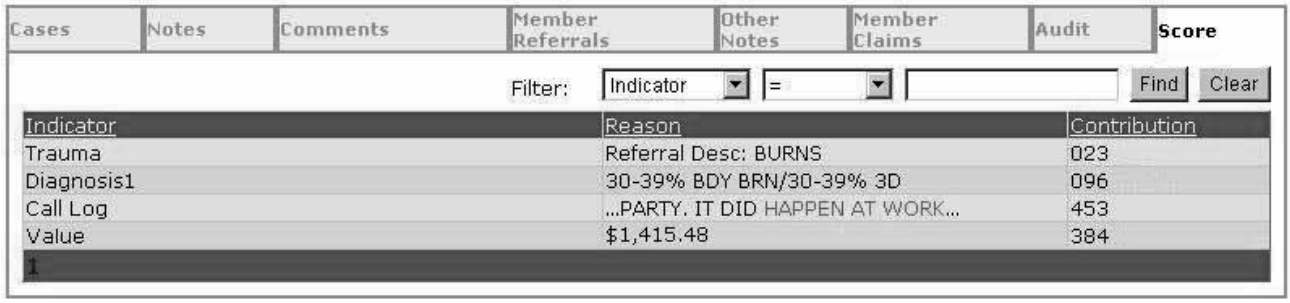

Figure 7. Indicators in Scoring.

4. CREATING CONCEPTS

While it is possible to create very complex and accurate specifications using CSL, this can be a very time consuming task, Furthermore, it may require both linguistic expertise, and domain expertise. To facilitate this task, we can leverage the linguistic and domain expertise contained within the linguistic rules and knowledge base of a natural language processing system to assist in the creation of new CSL. So, we can boot-strap from an existing system to create a new system that has a richer knowledge base using what we will call text-based concept creation, allowing a user to create CSL without any knowledge of CSL.

The text-based concept creation algorithm consists of the following eight steps. An example that illustrates each of these steps is then provided in Figure 8.

1. Input of text fragments. The user is prompted for one or more text fragments. These fragments are input to the next step.
2. Fragments split into words. The fragments are split into individual words using the Concept Analyzer from Figure 2.
3. Selection of relevant words. The user selects relevant words in the text fragments. (Default selection is available.)
4. Optional operations on relevant words. For any selected relevant word, the user can select any synonyms, hypernyms, and hyponyms available in Wordnet (or can automatically include them).
5. Concept matching. A predefined set of Concepts from the user are run over the fragments and all matches are returned. The predefined set of Concepts is for (domain-independent) grammatical constructions such as Subj_Verb_Obj. The resulting matches are known as a "Concept matches".
6. Removal of Concept matches. Certain Concept matches are removed, depending on (1) what words have been marked as "relevant" and (2) the interpretation placed on "relevant" by the user (the algorithm may optionally do one or both steps automatically).
7. Building of Concept chains (tiling). A list of "chains" is built from the Concept matches kept from the previous step, where a "chain" (also known as "tiles" and "generalizations") is a sequence of Concept matches such that:
    a. No two matches in the chain overlap, and
    b. No match can be added to a particular chain without violating (a) (i.e., the chains are of maximum length).
8. Chains written as CSL Concept. Every chain that passed through the previous step is written out as CSL. The matches within a chain are written into CSL as a conjunction with an "^" (AND) Operator. If there is more than one chain, then all chains are written into CSL as disjunctions (alternatives) with an "|" (OR) Operator. Chains are written out as follows:
    a. Take the first chain.
    b. Take the first match.
    c. Look up the match in the Rule Base (described below) to get Concept.
    d. Write out Concept.
    e. If there is another match in the chain, write out a "^" (AND) Operator and go to step c. with the next match.
    f. (No more matches.) If there is another chain, then write out a "|" (OR) Operator and go to step b. with the next chain. Else, exit (the defined Concept covers the text fragments).

The Rule Base contains domain independent concept definitions, along with rules that transform general Concepts that matched the text fragments into Concepts of the resulting Concept. As an example of a rule, consider "Subj_Passive_Verb_Obj => Subj_Verb_Obj". This rule states that if a text fragment contains a construct that matches the Subj_Passive_Verb_Obj Concept, then the resulting Concept should contain a slightly more general Concept Call Subj_Verb_Obj.

The Concept creation process ensures that only the Concepts that cover the selected relevant key words are considered. In cases where there is more than one Concept covering the input fragment, it uses the tiling algorithm (from step 7 of the earlier ten-step algorithm) to pick the most important Concepts. The ranking of the different possible Concept chains is determined by the order of the concept definitions contained in the Rule Base.

Consider the example shown in Figure 8. For the first 4 steps of the process, the user is required to provide inputs to guide the creation of the CSL. Note that the user is not required to have any knowledge of the syntax of CSL. The user needs only domain knowledge, plus basic knowledge of language. The algorithm determines in step 5 that several concepts match the input, and that both are potentially relevant (since they all contain the key words). Three of these matching concepts are shown in Figure 8, and assuming that the user does not remove any of these selections, the tiling algorithm finds one chain that spans the input. It then generates the Adoration concept, (where the name of the concept can be supplied by the user).

Algorithm Step Number	User Input	Example
1	Text fragments	*Mary was adored by John since high school*
2	Split into words	*Mary, be, adore, by, John, since, high, school*
3	Relevant words	*John, Mary, adore*
4	Synonyms (for *adore*)	*love intensely*
5		*Subj_Passive_Verb_Obj(john, adore, mary)* *Noun_Noun(john, mary)* *Noun_Verb(john,adore)* ...
6		*Subj_Passive_Verb_Obj(john, adore, mary)* *Noun_Noun(john, mary)*
7		*Subj_Passive_Verb_Obj(john, adore, mary)*

8	Adoration	Concept Adoration {
		Subj_Verb_Obj(john, @adore, mary)
		}

Figure 8. CSL from Text

5. EVALUATING INDICATORS

By themselves, the textual indicators that are identified or extracted from the documents based on CSL specifications do not have any real value. The value lies in the patterns or relationships between the indicators that are not only valid, but also interesting (with respect to some user-defined measure of what is interesting) [6]. Given that there are already well established human procedures to determine whether an insurance claim is interesting (with respect to reimbursement [4]), we can encode this human knowledge, and use it as a starting point for a scheme for scoring and ranking medical claims, based on a selection of indicators derived from structured information, and from unstructured information.

For each of the high level indicators, rules are defined with initial weights specified by human experts. As reflected in Figure 7, these initial weights contain references to not only structured information (like dollar value, and diagnosis code), but also unstructured information, including diagnosis and treatment indicators extracted from call logs and notes. These initial rules can also take into account conflicts between the structured and unstructured information. For example, structured data-field stating that the claim was not a work-related injury may conflict with a call log entry for the same claim which stipulates with a high certainty that it was a work place injury. Depending on the dollar value of the claim, or perhaps the claim history of a patient, a provider, and a health services organization, such a conflict may be sufficient to categorize a claim as one which requires human investigation.

Due to the hierarchical nature of the different indicators and subindicators, one can also establish relationships between closely related indicators in circumstances where there might be sufficient evidence from any one indicator. Consider the case where a patient has one claim for an injury resulting from different types of accidents all happening at malls. By generalizing over the different types of accidents, the data may call for further investigation into this individual, and could create a "case" resulting from the data accumulated in several claims, over even the creation of a case before a claim has been submitted into the system.

The result of this evaluation process is a prioritized list of medical claims, as shown in Figure 9, where the score (Scr) is the value calculated from the different indicators. The score is an integer between 1 and 1000. If the user wants to see the detail concerning how the score was calculated (s)he need only click on the score contained on the summary page shown below.

Finally, since a health care claims auditing system is a system which involves a human in the investigation of the resulting claims and cases, it is possible, over time, to build up a rich corpus of what constitutes an interesting (or uninteresting) claim, along with a wide range of associated indicators. With this data, it is possible to automatically change the weights associated with the different indicators, or even introduce new indicators into the equation.

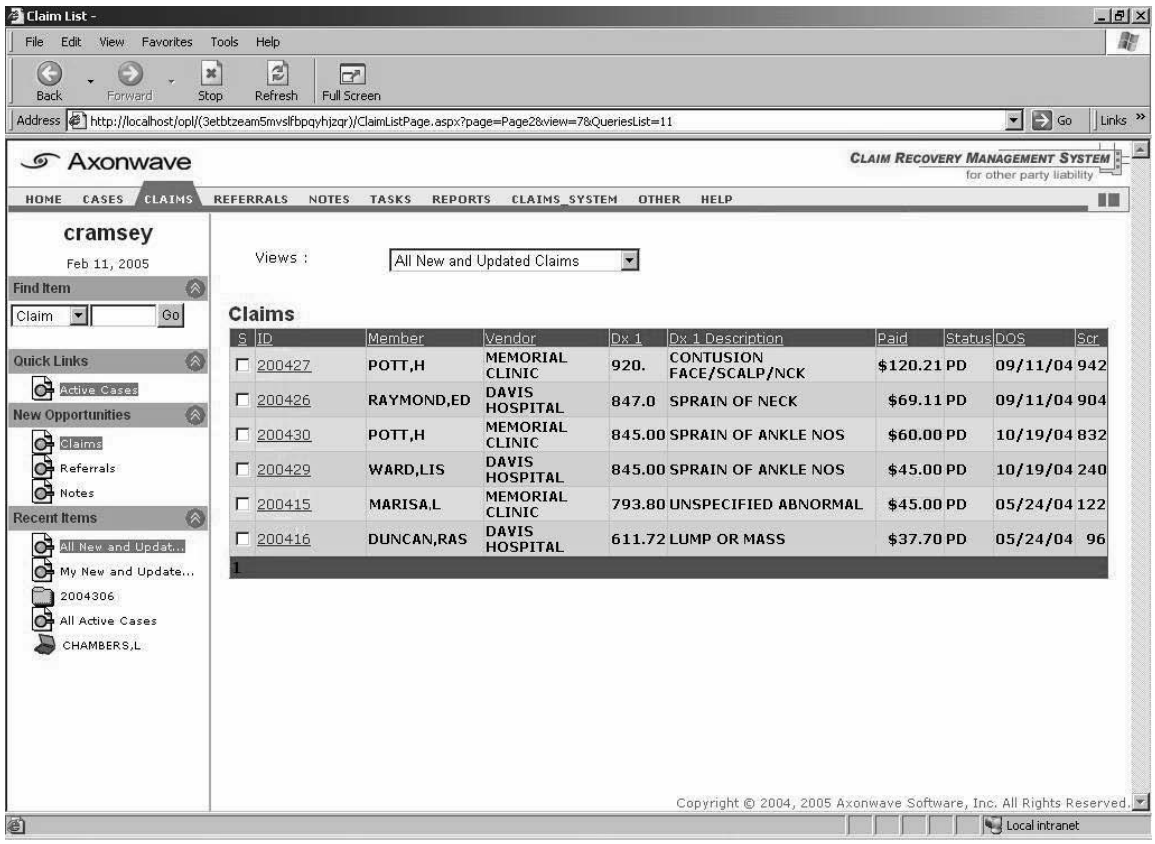

Figure 9. Scored Claims

6. CONCLUSIONS

In developing a health care claims auditor, we have created a system that combines both text mining and NLP, and we have illustrated one way to "bridge the gap" between NLP and text mining.

Using an NLP Concept Matcher, we obtain the capability to enrich text with semantic tags in a manner than can deal with spelling errors, abbreviations, acronyms, and the different variations in phasing that are used to express a concept. The Concept Matcher effectively provides the means to normalize the unstructured textual data into standard tags which can be extracted and feed into different data or text mining algorithms. It uses not only part of speech information, but also a syntactic parser, a rich lexicon, and a great deal of domain knowledge embodied on concept taxonomies.

The system is successful largely due to the constrained nature of the semantic domain. Because the system only deals with diseases, treatments, and medical insurance claim categorization, it is feasible to create a relatively complete knowledge model, leveraging existing taxonomies for diseases and treatments. We have seen techniques to semi-automatically create the CSL used in knowledge models. These techniques are applicable not only to health care claims auditing systems, but any system in which there is linguistic knowledge and domain specific semantic knowledge.

What we have seen is that it is possible to gain high value by using NLP techniques to map different sequences of natural language text to a relatively small number of high level indicators. The frequency, distribution and co-occurrence of these indicators form patterns and provide scores for claims, which can then be used to prioritize claims for human investigation, and create cases consisting of the claims and the relevant supporting information.

In the future, when more indicator-enhanced claim data becomes available, it will be possible to apply additional data-mining techniques [3] to detect previously unknown patterns. Of particular interest will be the use of association rules for fraud and abuse detection.

7. ACKNOWLEDGMENTS

Thanks to all the staff at Axonwave Software who have devoted years to the development of the algorithms and the infrastructure needed to support the work described in this paper. Special thanks go to Julia Birke and Lorna Fadden for their work on developing and integrating the concept taxonomies, and Dan Fass for his

formalization and description of the algorithm introduced in section 4.

8. REFERENCES

[1] Abney, S. Part-of-Speech Tagging and Partial Parsing. In Young, S. and Bloothooft, G. (eds), Corpus-Based Methods in Language and Speech Processing, Kluwer, Dordrecht, 1997, 118-136.

[2] Dilkina, K., and Popowich, F. An algorithm for anaphora resolution in aviation safety reports. Proceedings of the Sixteenth Conference of the Canadian Society for Computational Studies of Intelligence (AI 2004) (London, Canada, May 17-19 , 2004). Springer-Verlag, New York, NY, 2004, 524-539.

[3] Han, J., and Kamber, M. Data Mining: Concepts and Techniques. Morgan Kaufmann, San Francisco, CA, 2000.

[4] Jones, L.M. (ed). Reimbursement Methodologies for Healthcare Services, American Health Information Management Association, Chicago, IL, 2001.

[5] Jurafsky, D., and Martin, J. Speech and Language Processing. Prentice Hall, Upper Sale River, NJ 2000.

[6] Lavrac, N. and Grobelnik, M. Data Mining. In Mladenic, D., Lavrac, N., Bohanec, M. and Moyle, S. (eds), Data Mining and Decision Support Integration and Collaboration, Kluwer, Dordrecht, 2003.

[7] Manning, C. and Schutze, H. Foundations of Statistical Natural Language Processing, MIT Press, Cambridge, MA, 1999.

[8] Miller, G.A., Beckwith, R., Fellbaum, C., Gross, D., Miller, K. and Tengi, R. Five papers on WordNet, Princeton University, August 1993 ,[Online: ftp://ftp.cogsci.princeton.edu/pub/wordnet/5papers.pdf]

[9] Mladenic, D. and Grobelnik, M. Text and Web Mining. In Mladenic, D., Lavrac, N., Bohanec, M. and Moyle, S. (eds), Data Mining and Decision Support Integration and Collaboration, Kluwer, Dordrecht, 2003.

[10] Popowich, F. Use of Text Analytics and Taxonomies for Fraud and Abuse Detection in Medical Insurance Claims. Proceedings of Semantic Web Symposium of I2LOR-04 Towards the Educational Semantic Web (Université de Quebec à Montréal, Montréal, November 19, 2004), [Online: http://www.cscsi.org/home/CSCSI/Members/swig/swig04papers/popowich-swig.pdf]

[11] Shatkay, H. and Feldman, R. Mining the Biomedical Literature in the Genomic Era: An Overview, Journal of Computational Biology 10(6), 2003, 821-855.

[12] Turcato, D., Popowich, F. Toole, J. Fass, D. Nicholson, D. and Tisher, G. Adapting a Synonym Database to Specific Domains. In Proceedings of ACL'2000 Workshop on Information Retrieval and Natural Language Processing, Hong Kong, October, 2000.

[13] Zahariev, M. A linguistic approach to extracting acronym expansions from text. KAIS: Knowledge and Information Systems 6(3), 2004, 366-373.

About the author:

Dr. Fred Popowich is a Professor of Computing Science at Simon Fraser University, and an Associate Member of the Department of Linguistics. His non-academic roles include President and Chief Technology Officer of Axonwave Software, and chair of the Canadian Language Technology Roadmap Committee. He received his Ph.D. in Cognitive Science from the University of Edinburgh in 1989. Over the course of his twenty-year research career he has produced over fifty refereed publications. He and his colleagues jointly founded Axonwave Software in 1999, and have developed technologies for sophisticated classification, filtering, monitoring and retrieval of unstructured information, using natural language processing techniques, to produce claim recovery and cost containment software solutions for the health care industry.