Developing Corpora for Statistical Graphical Language Models

Andrew O'Sullivan[1], Laura Keyes[1] and Adam Winstanley[2]
1 School of Informatics and Engineering, Institute of Technology, Blanchardstown, Dublin 15, Ireland.
2 Department of Computer Science, NUI Maynooth, Co. Kildare Ireland

Abstract: In this work Statistical Graphical Language Models (SGLMs), a technique adapted from Statistical Language Models (SLMs), are applied to the task of graphical object recognition. SLMs are used in Natural Language Processing for tasks such as Speech Recognition and Information Retrieval. SGLMs view graphical objects as belonging to graphical languages and use this view to compute probabilistic distributions of graphical objects within graphical documents. SGLMs such as N-grams require large corpora of training data, which consist of graphical objects in contextual use (real world graphical documents). Constructing corpora is an important stage in developing the models and many issues need to be addressed. This paper discusses the development of graphical corpora and presents approaches to some of the problems encountered.

Keywords: Graphics Recognition, Statistical Language Modelling, Corpora.

1. Introduction

This paper is concerned with developing graphical language corpora for Statistical Graphical Language Models (SGLMs). SGLMs' technique is adapted from Statistical Language Models (SLMs) for the task of graphical object recognition [1]. This work is used in a graphics recognition system for the automatic recognition, indexing and retrieval of graphical data. Graphics recognition is a sub-field of pattern recognition and includes the classification and recognition of many types of graphical data (e.g. maps, engineering drawings, architectural plans). Statistical Graphical Language Models (SGLMs) are used to improve the performance of the graphics recognition system.

Statistical Language Models are used in Natural Language Processing for tasks such as Speech Recognition [2], Information Retrieval [3], Machine Translation and many more [4]. In SGLMs the graphical notation is treated as analogous to textual language. That is, while SLMs are designed for use with natural language phenomena such as words, sentences and whole documents, SGLMs are designed for use with graphical phenomena such as graphical symbols and graphical documents.

The *N*-gram model [5], which is frequently used in Natural Language Processing and has been adapted in this work for graphics recognition [1], requires a large corpus of training data. With text data, the corpora consist of words in their contextual use from real-world sources such as books, newspapers or telephone conversations. Likewise, with graphical data, the corpora needed must consist of examples of graphical objects taken from real-world sources such as architectural drawings, plumbing or electrical schematics and cartographic maps.

This paper describes a set of possible means of developing such corpora. Section 2 describes SGLMs in further detail, with reference to the *N*-gram model. Section 3 discusses the development of graphical corpora for SGLMs, focusing on object adjacencies and introducing the notion of direction into the development process. Section 4 discusses the representation of graphical objects and their relationships through graphs, corpora size and Part-of–Speech Tagging. Finally, Section 5 concludes the paper and discusses future work and directions.

2. Statistical Graphical Language Models (SGLMs)

Within this work on Statistical Graphical Language Models, graphical objects and symbols found on graphical documents, such as maps, architectural plans and electrical circuits form the graphical language used. Similarities exist between such graphical languages and natural language [1]. Based on these similarities statistical language models, normally used with textual data, are adapted and applied to graphical language. There are many statistical language models that can be used. These include, *Decision Tree* models [6], which assign probabilities to each of a number of choices based on the context of decisions. Some SLM techniques are derived from grammars commonly used by linguists. For example Sjilman et al. [7] use a declarative grammar to generate a language model in order to recognise hand-sketched digital ink. Other methods include *Exponential* models and *Adaptive* models. Rosenfeld [8] suggests that some other SLM techniques such as *Dependency* models, *Dimensionality* reduction and *Whole Sentence* models show significant promise. However this research focuses on the most powerful of these models, *N-grams*.

2.1 *N*-grams for graphical object recognition

The *N*-gram model is used to predict unknown objects based upon their neighbouring objects. It makes use of the fact that objects within a diagram may not have been placed randomly but have instead been placed with a purpose. This leads to the possibility that the objects within the diagram may have relationships with one another i.e. in some ways the objects' purposes are interlinked or inter-dependant. *N*-grams model these relationships and use them for recognition purposes. By knowing how objects relate to each other, an unknown object can be predicted based upon its relationships with its neighbours.

Typically either a Bi-gram ($N = 2$) or a Tri-gram ($N = 3$) is used. Bi-gram models use one neighbouring object at a time and Tri-gram models use two neighbouring objects. For a Bi-gram model the probability of an object is:

$$P(Object_i) = P(Object_i | Object_{i-1}) \quad (1)$$

And for a Tri-gram model the probability of an object is:

$$P(Object_i) = P(Object_i | Object_{i-2}\ Object_{i-1}) \quad (2)$$

Figure 1 shows a sample electrical circuit where one of the objects, marked X, is unknown. Object X's neighbouring objects are used to predict its identity. Using Formulae 1 and 2, the bi-gram probability of Object X being a particular object is:

$$P(Object_i) = P(Object_i | Capacitor) \quad (3)$$

where Objects i, i+1, ….k is the list of possible objects.

The tri-gram probability is:

$$P(Object_i) = P(Object_i | Resistor\ Capacitor) \quad (4)$$

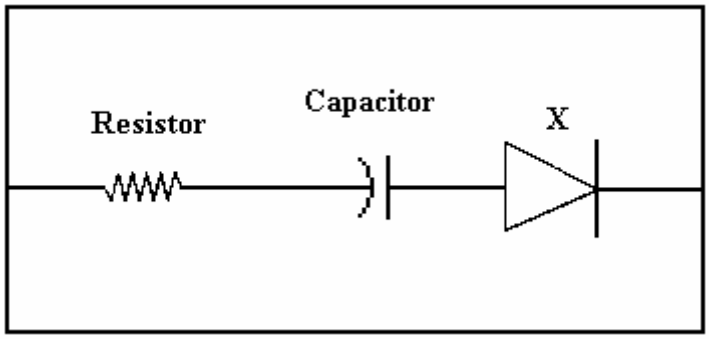

Figure 1. Sample electrical circuit with unknown object X.

These probabilities are estimated by using the relative frequencies of objects and their co-occurrences within a corpus of training data, which consists of examples of the objects in their real-world use. All of the co-occurrences of objects within the corpus are counted and listed in terms of relative frequency. For Bi-gram models object phrases consisting of two objects are counted e.g. Resistor – Capacitor. For Tri-gram models object phrases consisting of three objects are counted e.g. Resistor – Capacitor – Transistor. Once all the possible phrases have been counted they are stored in order of frequency in *N-gram Tables*. The probabilities from equation 3 and 4 can now be estimated using the following:

$$P(\ Object_i \mid Capacitor) = C(\ Capacitor, Object_i\ )\ /\ C\ (Capacitor) \qquad (5)$$

and

$$P(\ Object_i \mid Resistor, Capacitor)$$
$$= C(\ Resistor, Capacitor, Object_i\ )\ /\ C\ (Resistor, Capacitor) \qquad (6)$$

where C is the frequency of the relevant objects or phrases of objects within the training corpus.

The primary problem in implementing the process is how to construct the corpus of training data and count the objects and object phrases efficiently and effectively. Figure 1 shows a small section of an electrical circuit diagram where relationships between objects on this portion of the circuit diagram can be identified without difficulty making it easy to form object phrases. Graphical data however exists in a huge variety of domains and complexity, which depending on the domain or schema in question could make the object phrase construction very difficult. Figure 2 below shows an example of a more complex diagram where formation of object phrases poses a more difficult task. The definition of how objects relate to each other and therefore form phrases is fundamental to this problem. Using the concept of object *adjacencies* is identified here as one way to tackle the problem.

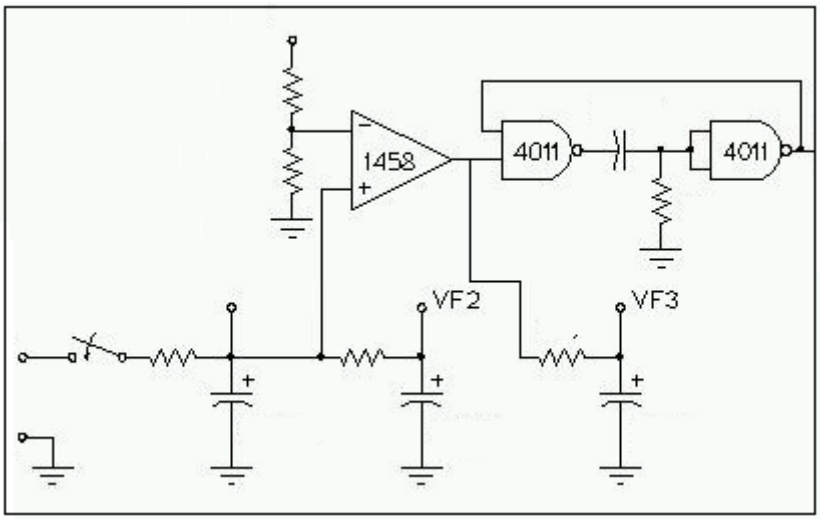

Figure 2. Example of a more complex electrical circuit

3. Developing corpora using object adjacencies

In SGLMs, neighbouring objects are used to form object phrases. How the term *neighbouring* is defined will govern how the object construction process works. Object *adjacencies* are used for this purpose, with the adjacencies defining how objects relate to each other. Once an adjacency is defined for a particular domain or diagram all the objects within that data that are adjacent to one another can be used to form object phrases,. These phrases are stored for processing in *N*-gram tables. In order to define object adjacencies from a document, information about the domain must be known.

SGLMs rely on the assumption that the domain in question has an underlying logical system, that is, the objects have not been placed randomly but have purposes and relationships with other objects. If the objects have been placed randomly the identity of an unknown object's neighbours will provide no useful information and the *N*-gram model will not be correct. So far this research has focused on electrical circuit schematics. Most electrical circuits have an inherent logical underpinning. For example, (see Figure 3), it is not unusual to have a resistor connected to a transistor, as the resistor protects the transistor from being damaged from too high a voltage/current. The resistor, like most

other objects within a circuit, has not been placed randomly. These relationships provide the semantics of the diagram and can be used in the recognition system.

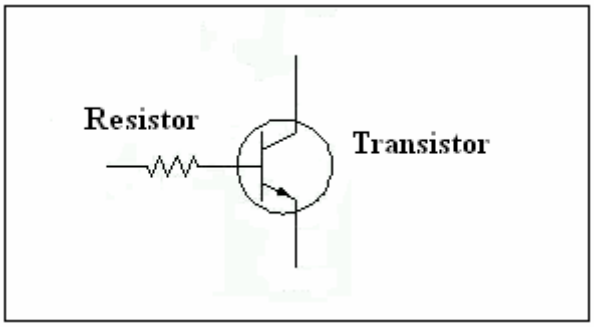

Figure 3. Example of objects relating to each other: a Resistor protecting a Transistor from high voltage

3.1 Object adjacencies for electrical circuits

The most important decision in designing SGLMs is how to define the object adjacency rules that will govern how the object phrases are constructed. The adjacencies must be defined so that the most meaningful relationships between objects are taken into account. As wires connect most objects within electrical circuits this research has defined objects to be *adjacent* to one another if they are connected in sequence by a wire. There are now several choices to be made before *N*-gram tables can be constructed. The first is the size of the object phrases. As mentioned previously, *N*-grams are usually either Bi-gram or Tri-gram, that is, two or three object phrases. While larger object-phrases might result in improved recognition performance, the complexity of the process is increased significantly as a larger corpus is needed.

3.2 Introducing direction into object adjacencies

The circuitry or schemas on graphical documents may be quite complex and addressed in application of SGLMs. Electrical Circuits can contain numerous components and wire connections, which could result in a lengthy phrase construction process. One approach to this problem is to introduce extra information about the domain in question into the *adjacency* definition stage in order to make adjacency criteria stricter. For example, with electrical circuits the fact that voltage is applied to the circuits and that current flows through the graphical objects can be used in adjacency definition of a document and also this can reduce the number of object phrases constructed resulting in faster computation and recognition. This is done by stipulating that object phrases can only be formed *in the direction of the current*. Not only will the number of phrases be decreased but the real world may also be modelled more accurately. The electrical symbols are now not only being treated as being part of a language, but they have also direction in the same way as words within natural language text also have a direction, for example, left to right in English language sentences.

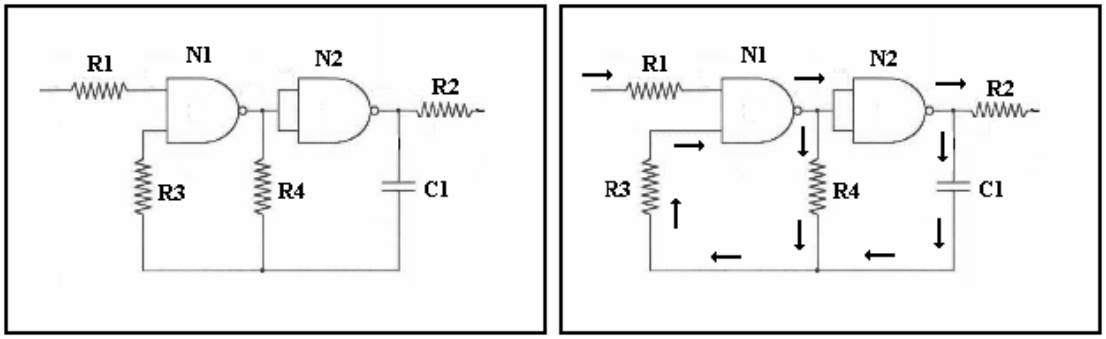

Figure 4 a) Sample electrical circuit and b) with current directions

Figure 4 a) shows a sample circuit with graphical objects labelled. Figure 4 b) shows the same circuit with the current direction indicated by the black arrows. Table 1 shows all the possible phrases

that commence with the object N1. The first two columns in the table represent 'Trigram Phrases' and 'Bigram Phrases' respectively with the phrases constructed using Figure 4 a). The last two columns of the table show the phrases constructed using the current direction in Figure 4 b) with a decrease in the number of phrases created.

Table 1. Object phrases, which start with the object N1, formed from Figure 1

Trigram Phrases	Bigram Phrases	Trigram Phrases with Direction	Bigram Phrases with Direction
N1 – N2 – R2	N1 – R1	N1 – N2 – R2	N1 – N2
N1 – N2 – C1	N1 – N2	N1 – N2 – C1	N1 – R4
N1 – N2 – R4	N1 – R3	N1 – R4 – R3	
N1 – R3 – R4	N1 – R4		
N1 – R3 – C1			
N1 – R4 – C1			
N1 – R4 – R3			

It must be noted however that by introducing extra information such as current direction into the process will increase the complexity of the task. A trade off must be made between how accurately the object relationships are modelled and how much time will be spent ascertaining the object relationships (via the adjacencies).

4. Counting object phrases

Another major problem with developing corpora for SGLMs is how to efficiently and effectively count all the object phrases within the graphical diagrams. Ideally the corpora would be made up of pre-labelled diagrams with the object connections, relationships or other relevant *adjacencies* also indicated. Finding a large enough source of such ready made documents may prove difficult so an efficient method of recording object relationships is desirable.

4.1 Graph representation

One method being tried by this research is to represent electrical circuits as graphs. By forming a graph with each node representing an object on graphical document, the object phrases can be quickly counted. Also, as the graph shows the full circuit, it may be easier to count higher-order object phrases that is, higher values of *N* within the *N*-grams.

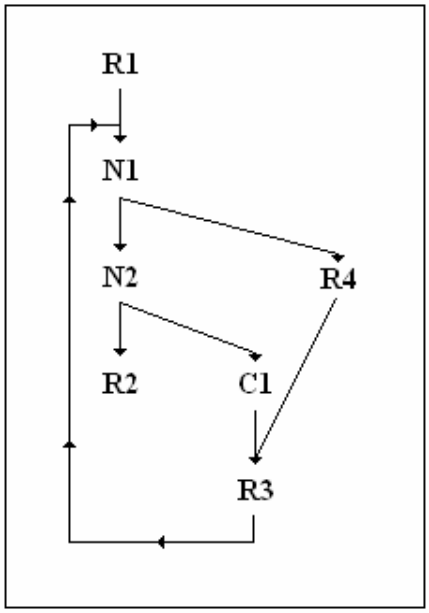

Figure 5. Graph representing the circuit from Figure 4 b)

Figure 5 shows the graph representation of the circuit in Figure 4 a). Such a graph can be easily formed and its structure is suitable for encompassing the notion of current direction. From a graph representation, object phrases can be formed. Figure 6 shows examples of object phrases formed from the graph in figure 5. Figure 6 a) shows a possible Bi-gram phrase, Figure 6 b) shows a possible Tri-gram phrase and Figure 6 c) shows a possible Quad-gram phrase, where $N = 4$.

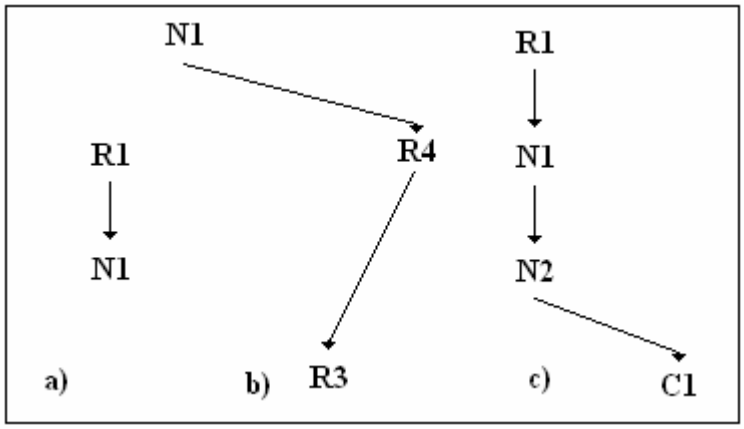

Figure 6. Possible Phrases: a) Bi-gram, b) Tri-gram and c) Quad-gram

4.2 Corpora size

A significant decision to be made while developing corpora is the size. Corpora used in SLMs for natural language processing can consist of millions of words, taken from a wide variety of sources. The Brown Corpus for example contains one million words from 15 different text sources [4]. With SGLMs it is also clearly beneficial to have large training corpora although the complexity of counting the object phrases can make the process an extremely lengthy one, depending on the number of documents that will make up the corpora.

One advantage graphical data has however, is that for a particular domain the vocabulary size could be significantly smaller than for a natural language. The vocabulary size is the number of distinct tokens, whether they are words for SLMs or electrical symbols for SGLMs. This means that the number of graphical documents needed will be also significantly smaller than for a natural language.

4.3 Part – of Speech Tagging

A variant of the *N*-gram model is the Part-of-Speech model. In SLMs it is used to assign tags such as verb, noun, adverb or adjective to words within text data. A similar variant is being undertaken by this research where similar graphical objects are defined as being part of a set (or denoted by a tag). These super-sets can then be assigned to unknown objects, again based on their neighbouring object's identities but also on their sets. For example, a super set identity in an electrical domain could be a Resistor. The Resistor set can include all types of resistor: fixed-value, rheostat, potentiometer and so on. An unknown object can then be identified as being part of a set, even if the exact object type is still not known.

5. Conclusion:

This paper describes different approaches for developing corpora of data for use with Statistical Graphical Language Models. In this work SGLMs are used and applied as a component of a graphics recognition system which uses shape and structural techniques for the labelling and recognition of objects within graphical documents. SGLMs are designed to be combined with other classifiers to improve the performance of recognition system. Statistical Graphical Language Models applied to the graphical data are adapted from Statistical Language Models used in Natural Language Processing.

The relationships between graphical objects is domain dependant and often very complex. Developing corpora and constructing phrases for this data is a difficult problem. There are however solutions which use object adjacencies to define these relationships. Previous work used object

adjacencies based on wire connection between objects on drawing. Previous work applied SGLMs using this adjacency definition and results indicate that this approach can improve recognition performance.

There are other considerations to be made however when developing the corpora. Future experiments will implement and evaluate the approaches discussed in this paper such as using direction in the definition of adjacencies to investigate whether recognition performance is further improved by its use. The use of graph methods to represent object relationships within diagrams will also be tested to assess their suitability and usefulness. Part-of-speech tagging will also be applied and evaluated. All of these methods will be evaluated on a variety of graphical domains to ascertain which methods suit best to each domain.

6. References:

[1] Andrew O'Sullivan, Laura Keyes and Adam Winstanley **"**An Extended System for Labeling Graphical Documents using Statistical Language Models", in proc. of the 6th IAPR International Workshop on Graphics Recognition (*GREC '05*), City University of Hong Kong, August 2005, 77-86

[2] F. Jelinek, *Statistical Methods for Speech Recognition*. MIT Press 1997

[3] J.M. Ponte, W.B. Croft, "A language modeling approach to information retrieval", in proc. of the 6th international conference on research and development in information retrieval *(SIGIR'98)* 1998, 276-281

[4] Jurafsky, D. and .Martin, J.H., *Speech and Language Processing*, Prentice-Hall, 2000.

[5] Manning, C.,D., and Schutz, H., *Foundations of Statistical Natural Language Processing*, MIT Press, Cambridge, 2001.

[6] Bahl, L R., Brown, P. F., Peter V. de Souza and R. L. Mercer., "A tree-based statistical language model for natural language speech recognition." *IEEE Transactions on Acoustics, Speech and SignalProcessing*, 37:1001-1008, July 1989.

[7] Shilman, M., Pasula, H., Russell, S. and Newton, R., "Statistical Visual Language Models for Ink Parsing." *AAAI Spring 2002 Symposium on Sketch Understanding*, 2002.

[8] Rosenfeld, R., "Two Decades of Statistical Language Modeling: Where Do We Go From Here?", *Proceedings of the IEEE*, 88 (8), pp 1270-1278, 2000.