In *Proceedings of TALN 2000*, Lausanne, October 2000

Generic Techniques for Multilingual Speech Technology Applications*

Julie Carson-Berndsen and Michael Walsh

Department of Computer Science, University College Dublin, Ireland

Abstract

This paper is concerned with generic techniques for representing and evaluating phonological information in multilingual speech technology applications. A computational linguistic model of phonological interpretation is enhanced by a framework for constructing and evaluating phonotactic automata and by a generic lexicon model. The techniques make way for the extension of current speech technology to languages which have received little attention thus far.

1. Introduction

The success of commercial speech technology applications, in particular speech recognition, has led to a more widespread acceptance of spoken language interfaces but has created the impression that any work remaining to be done in this area is just a matter of tweaking the current algorithms. However, one of the major problems in the area of speech technology remains the treatment of new words. In general, a new word refers to any structure which is well-formed with respect to the phonological and morphological constraints of a particular language but which is not part of a lexicon of that language. For example, the word *stramp* is not found in any English lexicon, but a native speaker of the language would consider it to be well-formed (as opposed to a form such as *stnamp* which is considered ill-formed). Such forms point to idiosyncratic gaps in the lexicon and thus could potentially become words of the language in the future. In the context of speech recognition the term 'new word' is usually restricted to mean new with respect to a particular corpus. In order to recognise or generate new words, information about their internal structure is required. While it is now generally accepted that morphological and phonological constraints are required in speech technology applications, stochastic techniques, such as Hidden Markov models, incorporate these constraints only implicitly. The generalisations made by such models are based on concatenation only and there is no way, without consulting a lexicon, of distinguishing well-formed expressions from ill-formed ones. A computational linguistic approach allows such constraints to be used explicitly.

One of the main concerns of the work presented in this paper is to provide a framework for developing and testing phonological well-formedness constraints for speech technology applications, in principle for any language. The main motivation for this approach is that much work in the area of speech technology is on well-known languages such as English, French and German but minority languages have not been considered to any great extent. The approach presented in this paper offers multilingual functionality in that it facilitates the rapid construction and diagnostic evaluation of phonological constraints for any language. Furthermore, the representations used are declarative and therefore the constraints are relevant both in the domain of speech recognition and in the domain of speech synthesis. Directly related to multilingual functionality is the issue of reusability which is becoming increasingly important in speech technology applications. This requires the development of generic technologies for acquiring and representing phonological and morphological descriptions for reuse with existing technology.

This paper proposes a generic framework specifically designed to construct and evaluate phonological well-formedness constraints in a computational linguistic model for speech recognition. The next section presents the concepts of *phonotactic automata* and *multilinear representations,* which define the knowledge sources for the computational linguistic model. The model uses finite state techniques and an event logic to interpret these representations in the context of speech recognition; this is described in section 3. Section 4 presents a generic framework, based on the computational linguistic model, for construction and evaluation of phonotactic automata and section 5 discusses a generic lexicon model, which is used to distinguish actual (lexicalised) structures from 'new' forms. Section 6 concludes with some discussion of future work.

2. Phonotactic Automata

Constraints on phonological well-formedness can be represented declaratively in terms of networks which can be interpreted by finite state automata. Such constraints, known as phonotactics, represent the possible combinations of sounds of a language within a particular phonological domain, usually the syllable. An example of such an automaton representation, of CC- combinations in English syllable onsets, is depicted in the network of figure 1.

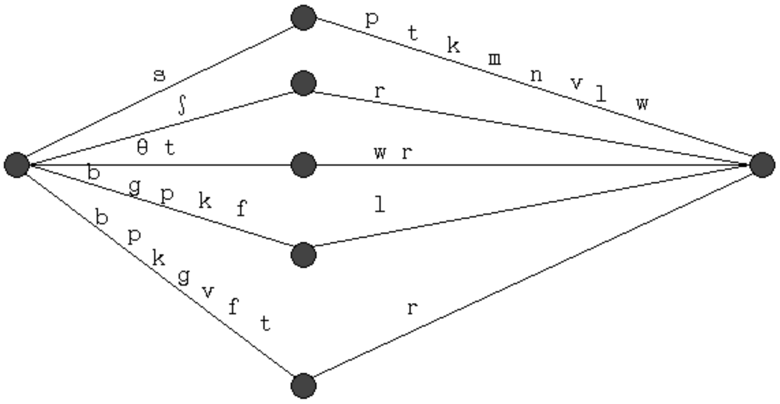


Figure 1: A phonotactic automaton of CC- combinations in English syllable onsets

This simple linear model of phonotactic constraints can be extended to a multilinear model where labels on the arcs of the automaton representation are no longer simple phonemes, but rather represent constraints on the temporal overlap relations occurring in each structural position. Examples of such constraints for two transitions are shown in figure 2. The constraint *voiceless* ° *plosive* specifies that a *voiceless* feature is expected to overlap with a *plosive* feature after an /s/ in an English syllable onset; this constraint generalises over /p/, /t/ and /k/. The arcs specify only those constraints required in the particular structural position, i.e. they are based on natural classes of features and are, in general, underspecified with respect to all the features needed to define any individual sound. Although only a substructure of the syllable onset is shown in figures 1 and 2, a complete set of phonological well-formedness constraints for a language can be represented in this way. The phonotactic automaton for each language will differ in its topology and in the overlap constraints it specifies.

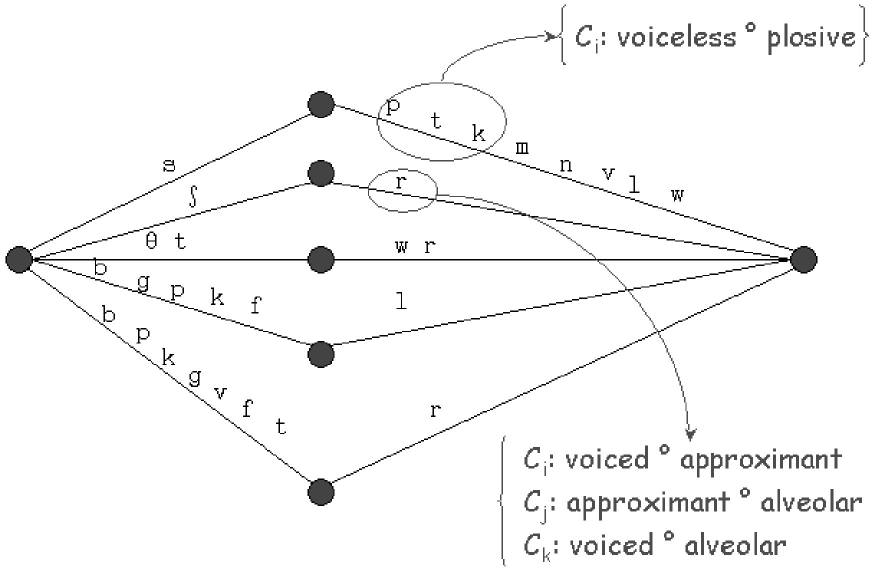

C_i: voiceless ° plosive
s
ʃ
θ t
b
g p k f
b
p
k
g
v
f
t
p t k
m n v l w
r
w r
l
r
C_i: voiced ° approximant
C_j: approximant ° alveolar
C_k: voiced ° alveolar

Figure 2: A phonotactic automaton of CC- combinations depicting temporal constraints

The advantage of this type of representation of phonotactic constraints is that an interpretation of multilinear phonological representations, as found in autosegmental phonology (cf. Goldsmith, 1990) and articulatory phonology (Browman & Goldstein, 1989), is made possible. A multilinear representation consists of a set of parallel tiers of features each of which has its own temporal pattern or *melody*. Each tier has its own segmentation and the start and end points of each of the features on the tiers may differ which makes the multilinear representation fundamentally different from a standard segmental phonological representation where all the features are subject to the same segmentation. In a multilinear representation, coarticulation can be modelled by overlap of information and since the overlap relations specified in the phonotactic automaton do not require features on different tiers to begin and end at the same time, a strict segmentation into non-overlapping units, as is usual in speech recognition, is not necessary. Phonological phenomena such as assimilations and elisions can, therefore, be represented in line with articulatory phonology in terms of feature (or gestural) overlap and magnitude. An example multilinear representation for the syllable *plant* is depicted in figure 3. This representation shows three tiers: the phonation tier with the features *voiced* and *voiceless*, the manner tier with the features *plosive, lateral, vowellike* and *nasal*, and the place tier with the features *labial, alveolar, back* and *apical*. Note that this is not a full specification; only three tiers are shown for illustration purposes. Features do not start and end at the same time and there are gaps between the features on some of the tiers. However, temporal relations can be defined between the features; the first *voiceless* feature overlaps the first *plosive* feature and the *labial* feature precedes (≺) the *alveolar* feature.

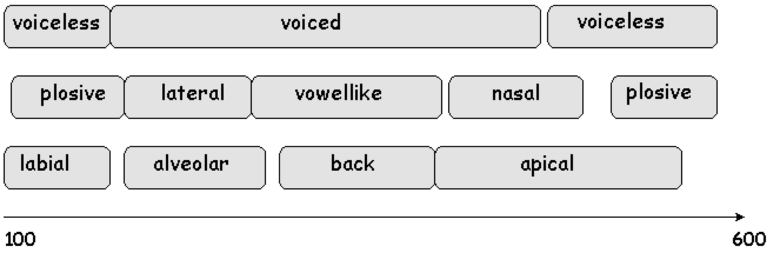


Figure 3: A multilinear representation of the syllable *plant*

To summarise, a phonotactic automaton defines a structured set of constraints which represents a complete phonotactics of a language and can distinguish between well-formed and ill-formed structures independent of any particular corpus and indeed of any particular speaker. (Note that the automaton is not intended to cater for proper names which do not conform to the phonotactics of the language although these can be modelled separately using the same finite state methodology). The temporal constraints of the phonotactics do not assume that a strict segmentation into non-overlapping units has taken place and therefore coarticulation phenomena and many speech variants can be modelled in a multilinear representation. The multilinear representation can then be interpreted using an event logic. This is the task of the computational linguistic model which is the topic of the next section.

3. The Computational Linguistic Model

The computational linguistic model for which the generic techniques are being developed is known as the *Time Map* model (cf. Carson-Berndsen, 1998, 2000). This model utilises finite state methodology and an event logic to demonstrate how declarative descriptions of phonological constraints can play a role in speech recognition. The main aim of this work has not been to build a speech recognition system which can compete with stochastic systems in terms of system performance but rather to design a knowledge-based multilinear component for a speech recognition system which utilises phonological well-formedness constraints and which is demonstrably of value in recognising new words, modelling and investigating coarticulation effects (temporal overlap of properties), and dealing with underspecified structures.

The *Time Map* model builds on the insights of multilinear finite state phonology which has dealt with the formalisation and implementation of nonlinear models such as autosegmental phonology and nonconcatenative morphology by Kay (1987), Bird & Ellison (1994), Kornai (1995) and others. It goes further than the above in that it has been tested and evaluated in the context of a speech recognition system for German. However, the model itself is not language-specific but uses language-specific components in the form of a phonotactic automaton and a corpus lexicon. These two components can be substituted with other phonotactic automata and lexica and in this way the methodology can be applied to other languages. Before discussing how this can be achieved, it is necessary to describe the basic functionality of the *Time Map* model.

In the *Time Map* model, the finite state techniques and the event logic together constitute a constraint-based approach to phonological parsing based on the temporal interpretation of phonological categories as events, and utilising a flexible notion of compositionality based on underspecified structures with autosegmental tiers of parallel phonetic and phonological events. The *Time Map* model distinguishes between two time domains: the *absolute* (or signal) *time* domain in which features are treated as events with temporal endpoints, and the *relative time* domain in which only the temporal relations (overlap and precedence) are relevant. The architecture of the Time Map model is depicted in figure 4.

Input to the *Time Map* model is a lattice of absolute-time events (i.e. acoustic features with time annotations as boundary points in milliseconds). These absolute-time events are mapped to the relative time domain using the axioms of the event logic which allow gaps to be interpreted in terms of precedence relations and overlapping properties in terms of overlap relations. Phonological parsing is then carried out entirely using the relations rather than the temporal annotations whereby the phonotactic automaton constrains the representation in the syllable domain to allow only well-formed structures to be accepted. Every time a final state in the automaton is reached, a possible syllable is found which is then passed to a syllable lexicon which tags the actual structures and provides them with a higher ranking than the ‘new’ forms. Since the syllable structures may be underspecified, the lexicon is feature-based and therefore full specifications can be provided for corpus syllables if required.

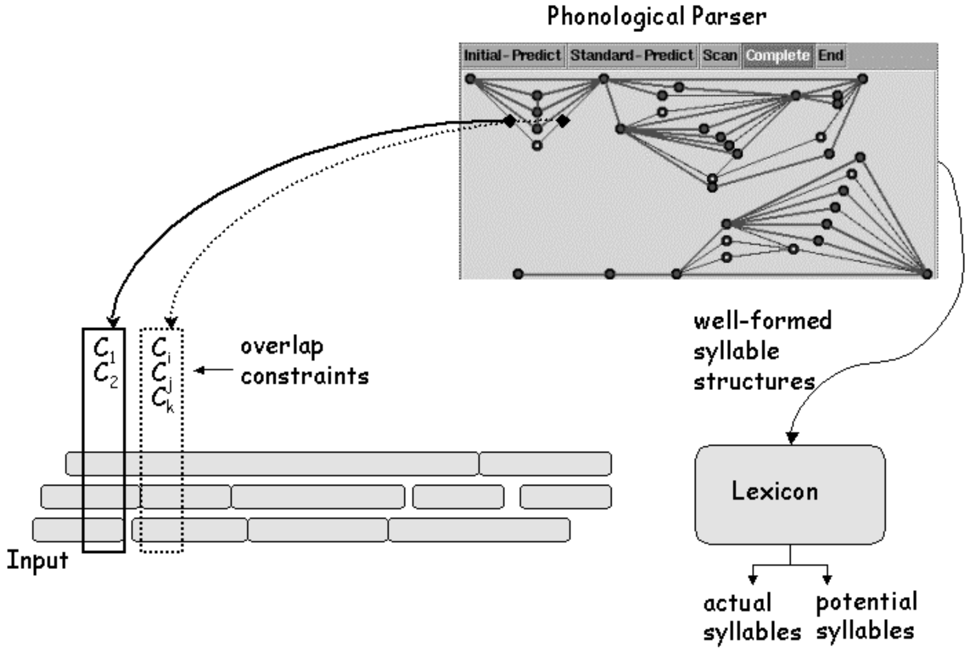


Figure 4: The *Time Map* Model

This concludes the description of the computational linguistic model. Further information on this approach including details of implementation and evaluation can be found in Carson-Berndsen (1998). The main focus of the research presented in the current paper is to provide the computational linguistic model with multilingual functionality by defining generic techniques for acquiring, representing and evaluating phonotactic automata and syllable lexica. These issues are addressed in the next two sections.

4. A Generic Framework for Developing Phonotactic Automata

Out of a desire to create generic technologies which could deal with speech recognition problems in a more robust way, the Language Independent Phonotactic System (LIPS) was designed. LIPS is a user-friendly system which enables users to build their own phonotactic automata by means of a graphical user interface. Its language independence or multilingual functionality stems from the users' ability to create a phonotactic automaton for any language, and furthermore, users can apply their own feature set when constructing a network, thus IPA-like features, such as plosive and fricative, feature geometries or any other feature set, can be applied. This generic approach permits the user to easily develop phonotactic descriptions and feature sets (or just add individual features) not only for languages such as English or German but also for minority languages, which have not been the focus of speech technology applications thus far.

LIPS provides a testbed for diagnostic evaluation of phonotactic descriptions and feature sets. It comprises two principal components, the network generator for constructing phonotactic automata and the parser (cf. figure 5). Other components under development within the LIPS architecture include a network-visualisation module, an input-visualisation module, a parse-status module, and an evaluation module.

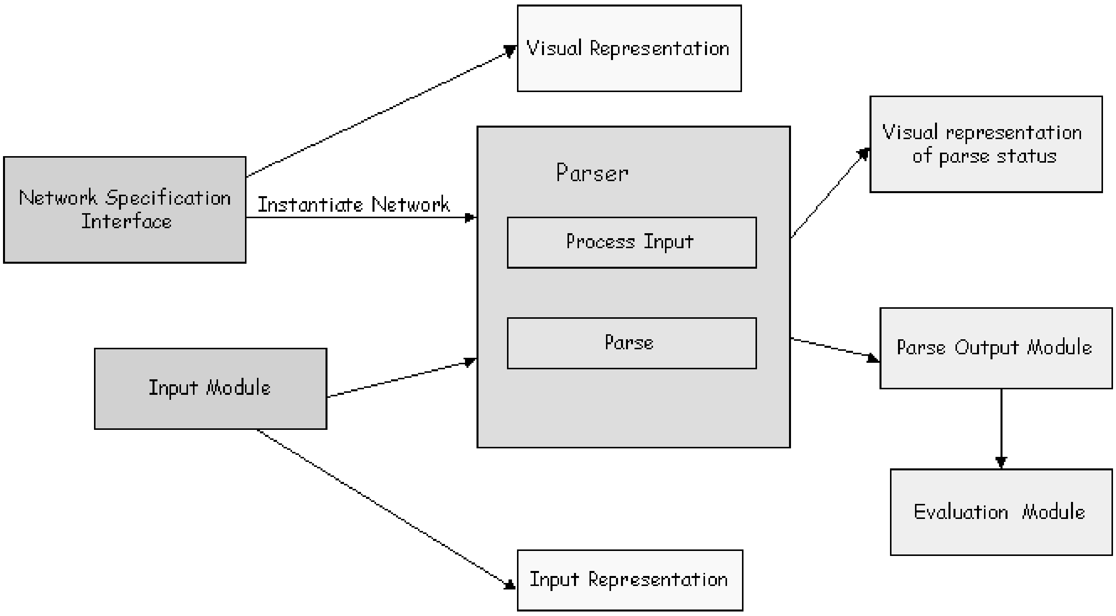
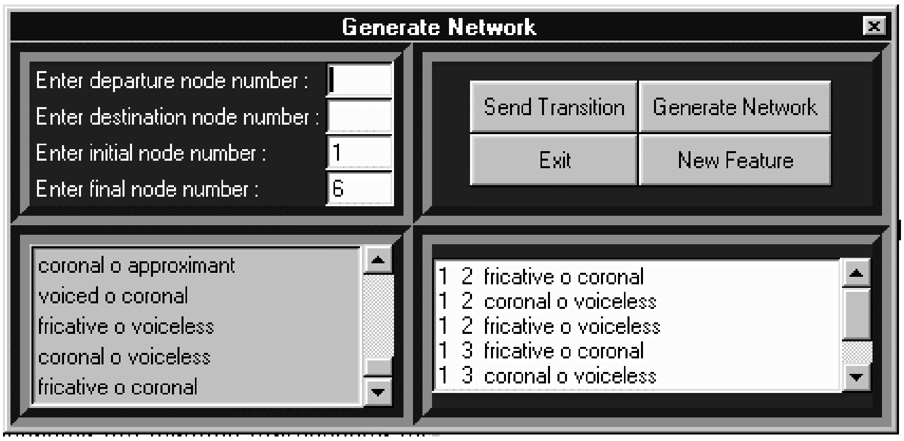


Figure 5: The LIPS Architecture

The network generator interface (see figure 6) allows the user to enter node values and to select from a list of feature overlap relations those that a given arc is to represent. Alternatively users can specify their own features. The network incorporates the characteristics of the *Time Map* model both implicitly, features precede other features in the network as arcs precede other arcs, and explicitly, features are selected in terms of overlap, thus a given transition might look like the following: < *1 2 labial* ° *nasal* >, which states that the arc between nodes 1 and 2 specifies the temporal constraint that a *labial* feature must overlap a *nasal* feature. When the user has completed his selection of nodes and feature overlaps the system generates a list of transitions representing the network.



Enter departure node number :
Enter destination node number :
Enter initial node number : 1
Enter final node number : 6

coronal o approximant
voiced o coronal
fricative o voiceless
coronal o voiceless
fricative o coronal

1 2 fricative o coronal
1 2 coronal o voiceless
1 2 fricative o voiceless
1 3 fricative o coronal
1 3 coronal o voiceless

Figure 6: The LIPS Network Generator

LIPS has been designed as a generic framework for the *Time Map* model. Therefore, it encompasses both the finite state methodology, by allowing the definition of phonotactic automata for individual languages, and the event logic which defines the mapping from absolute time to the temporal relations between the features. LIPS employs a top-down parsing strategy. Rather than mapping all absolute-time events to events in the relative time domain, the parser only checks for temporal overlaps which are anticipated by the phonotactic automaton, in this way minimising the search space and processing time. A speech data file containing temporally annotated events serves as the input to the parser. These events might look like the following:

```
449 s fricative
452 s coronal
573 e fricative
583 e coronal
573 s apical
```

indicating that a *fricative* begins at 449ms and ends at 573ms, a *coronal* begins at 452ms and ends at 583ms and an *apical* begins at 573ms. The parser, beginning at the initial node in the phonotactic automaton imposes a window, of parametrisable size, on the input file and scans for the overlap relations specified on the arcs leading out of the initial node. For example, if the arc from node 1 to node 2 specifies a constraint that a *fricative* feature must overlap a *coronal* feature then the parser searches within the window for these features and determines whether they overlap in time by means of the axioms of the event logic. A successful scan (i.e. all constraints are satisfied) results in the completion of an arc. The nodes from which to search next are then predicted according to rules of precedence. The start-point of the next window is based upon the end-points of the features found. Searching continues in a breadth-first manner, gradually moving through the network and incrementing the window, until no more alternative paths are available to the parser. All paths which result in a final state constitute the syllable hypotheses which are then passed to an evaluation module.

The evaluation module requires two additional components in order to carry out a diagnostic evaluation of the phonotactic constraints: a reference file, to which the output hypotheses must be compared, and a lexicon, which distinguishes lexicalised forms from potentially new forms. For this purpose, a generic lexicon model has been proposed which is the topic of the next section.

5. The Generic Lexicon Model

The main motivation for the development of the generic lexicon model was to be able to generate many different output representations on-the-fly. The LIPS diagnostic evaluation is, therefore, only one application of the generic lexicon model, which can be used to generate output formats for other speech technology and multimodal applications. On the basis of a phonological feature description of syllable structure, individual lexica with varying degrees of granularity (e.g. features, phonemes, syllables, etc.) for recognition and synthesis can be generated in an application-specific format. The generic lexicon model uses either individual finite state transducers or cascades of finite state transducers to generate other segmental and multilinear representations from syllable models. The syllable models are represented in DATR, a simple language designed specifically for lexical knowledge representation that allows the definition of nonmonotonic inheritance networks with path/value equations (cf. Evans & Gazdar, 1996).

The specific interaction between LIPS and the generic lexicon model is depicted in figure 7. Input to the generic lexicon model is phonemically labelled syllable data. The generic lexicon model allows syllable representations to be generated for a lexicon consistency test, a soundness and completeness check of the LIPS phonotactic automaton, and for diagnostic evaluation of the LIPS model with other data. In parallel, reference files have been generated on the basis of the same phonemically labelled syllable data for use in the evaluation procedure.

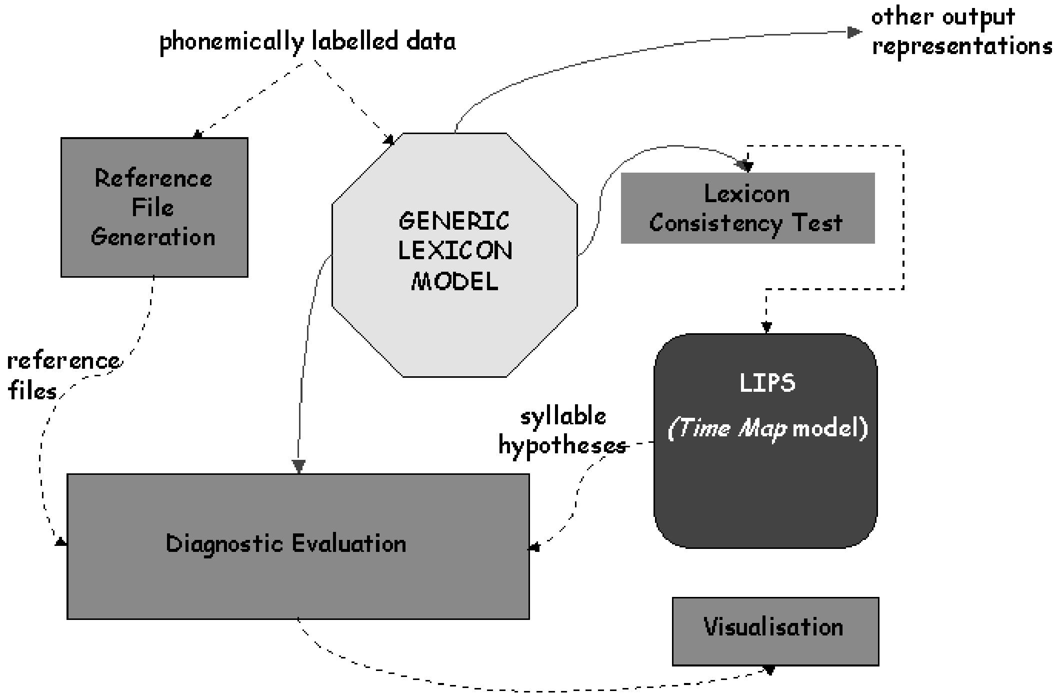


Figure 7: Interaction between LIPS and the Generic Lexicon Model

The generic lexicon model consists of syllable and feature template definitions which specify the patterns for the phonotactics and the default structures for consonants (based on the unmarked consonant /t/) and vowels (based on the neutral vowel /▶ /) using a feature geometry classification. Specific phonological segments inherit the defaults from the general consonant or vowel nodes but are marked with respect to particular features (i.e. exceptions to the defaults). For example, the segment /d/ would inherit all its feature specifications from /t/ with the exception of voicing. Specific syllable entries then inherit their feature structure from the individual phonological segments and their pattern from the syllable template definitions. A syllable template definition based solely on the individual segments will only allow a segmental representation of syllable structures to be generated in which the segmentation function across the tiers is the same. However, for the speech application discussed in the previous sections, a more detailed temporal representation is required from which information about the individual tiers and their melodies can be generated independently. Rather than representing the somewhat arbitrary splitting of events necessitated by segmentation into phonemic segments (e.g. the German syllable /mIt/ consists of two *voiced* segments followed by a *voiceless* segment), it is possible with this type of representation to refer to a single event, which spreads (or is smoothed) over more than one phonemic segment (e.g. the German syllable /mIt/ consists of a *voiced* feature, which spreads over two phonological segments, followed by a *voiceless* feature). The individual tier melodies are defined in the generic lexicon using finite state transducers which, although conceptually separate from the syllable and feature templates and the specific entries, are also modelled in DATR.

Given an input query, the generic lexicon tool generates the output format for LIPS on the basis of the specific syllable entries by inheriting feature information and average durations for each phonological segment via the syllable structure templates, and the finite state transducers, enhanced by arithmetic operations to calculate the temporal annotations, perform spreading in line with the obligatory contour principle (cf. figure 8). The tier melody then describes the precedence relations that exist between individual features on the tier. Similarly overlap information describing the temporal relation between events on different tiers can be generated.

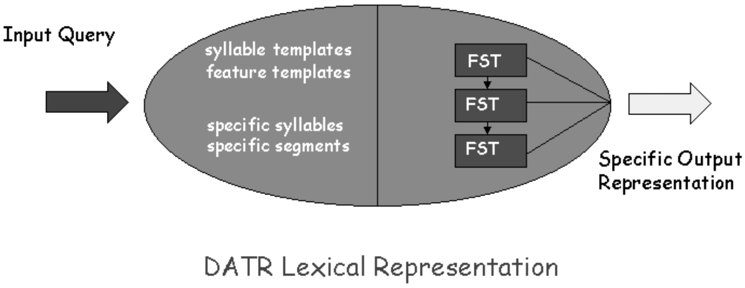


Figure 8: The Generic Lexicon Model

For example, the generic lexicon model can generate the following information about the German syllable /mIt/:

- Tier melody information (here phonation tier as example) with a precedence relation and temporal annotations (average duration in ms); the *voiced* feature has spread across two phonemic segments: event([voiced], 142) ≺ event([voiceless], 79)
- Overlap information (for /m/): event([voiced]) ° event([nasal]) ° event([labial])

The multilinear output representations generated by the generic lexicon model consist either of individual tier melody representations or complete phonological event structures (cf. figure 3) with parallel information describing temporal relations between all features in the representation, which are used by LIPS to distinguish between syllables that are in the corpus and those that are not. Other output representations generated by the generic lexicon tool are phonemic representations with and without temporal information, segmental feature representations also with or without temporal annotations, and graph visualisations of the phonological structures of the syllable models.

The generic lexicon model has also been designed with multilingual functionality in mind. The syllable and feature templates and the specific syllable entries are language dependent but the finite state transducers, which define the mapping to other output representations, are not. Therefore, once the syllable and feature templates have been specified, new syllable entries can be generated automatically from corpora and the existing finite state transducers will allow the varying output representations to be generated.

6. Conclusions and Future Work

This paper has presented generic technologies for computational linguistic models in speech technology applications. Phonotactic automata and multilinear representations were introduced and their application in a computational linguistic model of phonological interpretation in speech recognition was discussed. A framework for developing and testing phonotactic descriptions and a generic lexicon model for speech technology applications were presented and the multilingual functionality of these technologies was highlighted. Although the paper focussed on generic techniques for developing multilingual components for a specific computational linguistic model, the techniques proposed are relevant for speech technology applications in general. It has already been demonstrated, for example, that phonotactic automata provide useful information for fine tuning of stochastic models (cf. Jusek et al, 1994) and the generic lexicon model has been shown to generate other representation formats relevant to speech technology and multimodal applications (cf. Carson-Berndsen, 1999a,b).

Future work in this area concerns demonstrating the application of the generic technologies to other languages, such as Irish, which has a fundamentally different phonology from that of English or German. Currently, the work has focussed on representation and evaluation but another possible direction for future investigation would be the integration of the generic framework with ongoing work in the area of automatic acquisition of phonotactic constraints (cf. Belz, 1998).

The work described in this paper has specifically addressed speech recognition applications. However, since the phonotactic descriptions are declarative, they can also be used in speech synthesis applications. On the basis of the temporally annotated feature-based syllable models, finite state transducers can be defined in the generic lexicon model to generate parameters for synthesis. This is a topic which is currently under investigation in an ongoing research project.

References

BIRD S. & T. M. ELLISON (1994), One-level phonology: autosegmental representations and rules as finite state automata. In: *Computational Linguistics* 20, pp.55-90.

BELZ, A (1998), An approach to the automatic acquisition of phonotactic constraints. In: ELLISON, T. M. (ed.): *Proceedings of SIGPHON 98: The Computation of Phonological Constraints*, pp. 34-55.

BROWMAN C.P. & L. GOLDSTEIN (1989), Articulatory gestures as phonological units. *Phonology 6*, Cambridge University Press, Cambridge, pp.201-251.

CARSON-BERNDSEN J. (1998), *Time Map Phonology: Finite State Models and Event Logics in Speech Recognition.* Kluwer Academic Publishers, Dordrecht.

CARSON-BERNDSEN J. (1999a), A Generic Lexicon Tool for Word Model Definition in Multimodal Applications. In: *Proceedings of EUROSPEECH 99*, 6th European Conference on Speech Communication and Technology, Budapest, September 1999.

CARSON-BERNDSEN J. (1999b), A Feature Geometry Based Lexicon Model for Speech Applications. In: *Proceedings of IDS 99*, Interactive Dialogue in Multimodal Systems, ESCA Tutorial and Research Workshop, Kloster Irsee, June 1999.

CARSON-BERNDSEN J. (2000), Finite State Models, Event Logics and Statistics in Speech Recognition. In: GAZDAR, G.; K. SPARCK JONES & R. NEEDHAM (eds.): *Computers, Language and Speech: Integrating formal theories and statistical data*. Philosophical Transactions of the Royal Society, Series A, Volume 358, issue no. 1770.

EVANS R. & G. GAZDAR (1996), DATR: A language for lexical knowledge representation. In: *Computational Linguistics* 22, 2, pp. 167-216.

GOLDSMITH J. (1990), *Autosegmental and Metrical Phonology*. Cambridge Mass: Basil Blackwell Inc.

JUSEK A. et al. (1994), Detektion unbekannter Wörter mit Hilfe phonotaktischer Modelle. In: *Mustererkennung 94*, 16. DAGM-Symposium Wien, Berlin: Springer-Verlag, pp.238-245.

KAY M. (1987), Nonconcatenative Finite-State Morphology. In: *Proceedings EACL '87*, Copenhagen, pp.2-10.

KORNAI A. (1995), *Formal Phonology*. Levittown, PA: Garland Publishing.

[*] This research is funded by Enterprise Ireland Basic Research Grant Scheme SC/99/069