

Self Organizing Maps for the Visual Analysis of Pitch Contours

Dominik Sacha Yuki Asano Christian Rohrdantz Felix Hamburg
Daniel Keim Bettina Braun Miriam Butt

Data Analysis and Visualization Group & Department of Linguistics
University of Konstanz

forename.lastname@uni-konstanz.de

Abstract

We present a novel interactive approach for the visual analysis of intonation contours. Audio data are processed algorithmically and presented to researchers through interactive visualizations. To this end, we automatically analyze the data using machine learning in order to find groups or patterns. These results are visualized with respect to meta-data. We present a flexible, interactive system for the analysis of prosodic data. Using real-world application examples, one containing preprocessed, the other raw data, we demonstrate that our system enables researchers to interact dynamically with the data at several levels and by means of different types of visualizations, thus arriving at a better understanding of the data via a cycle of hypothesis generation and testing that takes full advantage of our visual processing abilities.

1 Introduction and Related Work

Traditionally, linguistic research on F_0 contours has been conducted by manually annotating the data using an agreed-upon set of pitch accents and boundary tones such as the ToBI system (Beckman et al., 2005). However, the manual categorization of F_0 contours is open to subjectiveness in decision making. To overcome this disadvantage, recent research has focused on functional data analysis of F_0 contour data (Gubian et al., 2013). The F_0 contours are smoothed and normalized resulting in comparable pitch vectors for different utterances of the same structure. However, with this method, the original underlying data is abstracted away from and cannot be easily accessed (or visualized) for individual analysis.

One of the typical tasks in prosodic research is

to determine specific F_0 contours that signal certain functions. State of the art analysis is time intensive and not ideal, because statistics or projections are applied to the data leading to a possible loss of important aspects of original data. To overcome these problems, we offer a visual analytics system that allows for the use of preprocessed F_0 pitch vectors in data analysis as well as the ability to work with the original, individual data points. Moreover, the linguistic researcher is interactively involved in the visual analytics process by guiding the machine learning and by interacting with the visualization according to the visual analytics mantra “*Analyze first, Show the Important, Zoom, filter and analyze further, Details on demand*” (Keim et al., 2008).

Our system consists of three components. The *Data Input* where all input files are read and converted into the internal data model. The second part covers *Machine Learning* where we make use of Self Organizing Maps (SOM) in order to find clusters of similar pitch contours. The visualization based on the SOM result is realized within our last component, the *Interactive Visualization*. The researcher can interpret the data directly via this visualization, but may also interact with the system in order to steer the underlying model. The overall work flow is illustrated in Figure 1. This combination of human knowledge and reasoning with automated computational processing is the key idea of visual analytics (Thomas and Cook, 2006) and supports human knowledge generation processes (Sacha et al., 2014). Our contribution builds on existing previous work on SOM based visual analysis (Vesanto, 1999; Moehrmann et al., 2011), but also on previous attempts to visually investigate data from the domain of prosodic research (Ward and McCartney, 2010; Ward, 2014). Furthermore, we profit from approaches to analyze speech using the SOM algorithm (Mayer et al., 2009; Silva et al., 2011; Tadeusiewicz et al.,

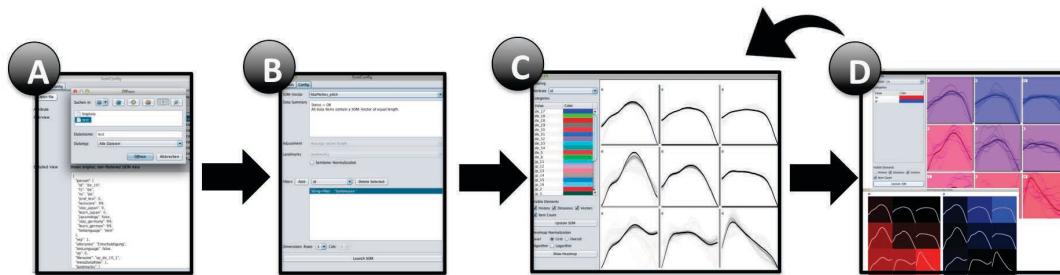


Figure 1: Work flow in four steps. A-Data Input, B-Configuration, C-Training, D-Visualization.

1999), but open up a new domain within this field as we allow for a visualization of pitch contours directly on a SOM-grid. We furthermore do not just produce one SOM, but also compute and visually present several dependent/derivative SOMs.

2 System

The system pipeline consists of three main components: 1) Data-Input; 2) Machine-Learning; 3) Interactive Visualizations.

2.1 Data Input

Our system is able to process and visualize any kind of data that satisfies the following restrictions. The data set needs to consist of a list of *data items*, where each item contains a set of key-value pairs, also called *data attributes*. The value of a data attribute must be a primitive, i.e., either a number, text string, or an array consisting of primitives. Except for primitive-arrays we do not allow nested data, thus we flatten the input data if necessary. Overall, data items should be comparable and contain attributes with equal keys (and different values).

The system also expects comparable feature vectors to which a distance measure can be applied. Furthermore, additional (meta) data can be part of the input. In the use cases presented here, each F_0 data is connected with speaker information such as the native language of the speaker, the level of second language (L2) proficiency and the context the data was produced in.

Vector Preprocessing After having loaded in the data, our system allows for the inspection of data prior to the actual analysis. Figure 1-A shows the inspection view that is typically used in the work flow at first. As part of the configuration work flow, the user selects an attribute as the *Input Vector* (Figure 1-B). This forms the basis of

the machine learning component.

Before entering the machine learning of training phase, our system performs a validation of the *Input Vector* and allows for its adjustment if necessary. Whereas normalized and smoothed data, i.e., data items with vectors of equal length, can be processed directly, our system also offers the functionality to perform basic preprocessing of raw *Input Vectors*. If it is found that not all vectors have equal length, we offer several preprocessing techniques from which one can be chosen: Besides simple approaches of adding mean-values (mean-padding) or 0s (zero-padding), we also offer an approach that makes use of linear interpolation (pair-wise). If time and landmark-information is available, it is also possible to divide the vectors into parts and adjust each of the parts separately. As a result, all the parts have equal length and are therefore better suited for comparison. The *Input Vectors* values can be normalized using *Semitone-Normalization*. The mean value can also be subtracted from each contour, in order to minimize gender effects.

In sum, we offer a very flexible preprocessing functionality for the *Input Vectors*. The available techniques can be combined flexibly and dynamically according to what is most suitable for the analysis task at hand. However, there are still methods that could be added. For example, one could additionally enhance the vector processing by a stronger leveraging of the time information in order to prepare the data for duration focused analysis tasks.

2.2 Machine Learning

We make use of Machine Learning (ML) for the detection of groups/clusters that are present in the data based on the *Input Vectors*. Additionally, the system detects correlations to the meta data. In our use cases this included information about the

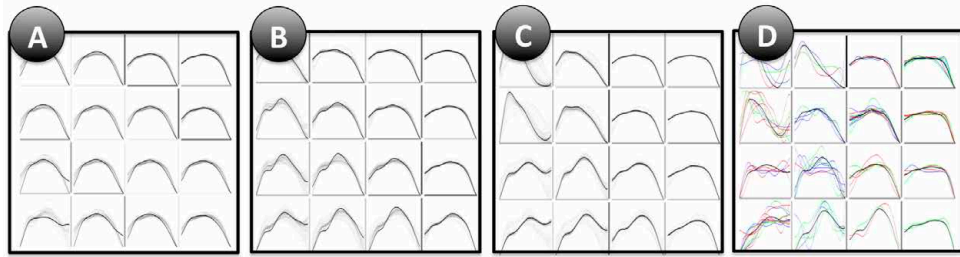


Figure 2: SOM-Training illustrated by 4 steps. For each cluster the prototype and the distances between adjacent cells are visualized by black lines in between. In step 4 the training has finished and the dedicated F_0 contours are also drawn in each cell.

native language of the speakers and the level of their language proficiency.

In principle, any distance function, projection or clustering method could be applied in our extensible framework. The central problem that needs to be resolved is that the high dimensional data from the *Input Vectors* needs to be reduced to a two-dimensional visualization that can be rendered on a computer screen or a piece of paper. We experimented with several different methods and found that SOMs, also known as Kohonen Maps (Kohonen, 2001), match the demands of this task best. SOMs are a well established ML technique that can be used for clustering or as a classifier based on feature vectors. SOMs are very suitable for our purpose for several reasons. First we can use SOMs as an unsupervised ML-technique to find a fixed number of clusters subsequent to a training phase. SOMs also provide a topology where similar clusters are adjacent. Finally, the algorithm adapts to the given input data depending on the amounts of desired clusters and data.

Furthermore, in our system, the clustering and dimensionality reduction are integrated in one step. This stands in contrast to other clustering and dimensionality reduction techniques like Multi Dimensional Scaling (MDS), Principal Component Analysis (PCA) or Non-negative Matrix Factorization (NMF). A disadvantage found with these other methods is that they tend to lead to clutter in the two-dimensional space (when there is high degree of overlap in the data). It is also unclear when to perform the clustering: in the high dimensional space before projection or in the two-dimensional space afterwards.

Our system proceeds as follows. First, the SOM-grid is initialized with random cluster centroids, which are feature vector prototypes for each cluster. Afterwards each feature vector is

used to train the SOM in a random order. For each vector the SOM algorithm determines the best matching unit (BMU) and adjusts the BMU and adjacent clusters prototypes based on the input vector. This process is repeated n -times until the SOM is in a stable state (Figure 2, steps A-C). After the training phase the resulting grid can be used for clustering. Each vector is assigned to the cluster with the least distance to the cluster prototype (BMU). In Step D of Figure 2 each cell represents a cluster containing the cluster prototype (black vector) and the cluster members (colored vectors).

Note that we did not rely on existing software libraries like the SOM-toolbox, but instead implemented the algorithm from scratch. The reason for this is that we aim at being able to visualize and steer the algorithm at every step (see Section 2.4).

2.3 Visualizations

We build on Schreck et al.’s work on SOM-based visual analysis (Schreck, 2010). Within the basic SOM-grid, we provide several different ways of visualizing the information of interest to the researcher. As shown in Figure 3-A, we provide an overview visualization which shows the SOM-grid (Figure 3-A) filled by the clustered pitch contours. The individual cells also show the cluster centroid and the vectors (contours) that belong to that cell in relation to the centroid (Figure 3-F). We also visualize the training history of a cluster in the background in each cell (Figure 3-A) in order to keep track of the training phase.

Beyond the clustered contours, we furthermore provide possible visualizations (these can be selected or not), which add in simple highlighters or bar charts to the SOM result (Figure 3-C). We also experimented with heatmaps,¹ which turned out

¹In our approach a color overlay for the SOM grid

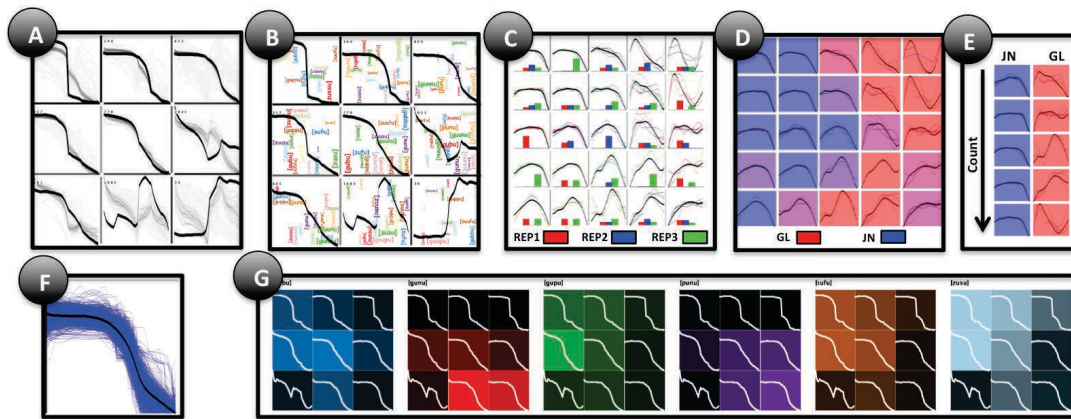


Figure 3: Different approaches to visualize SOM-results according to available meta data. (A) Grid visualization, (B) word cloud, (C) bar charts, (D) mixed color cells, (E) ranked group clusters, (F) one single cell that visualizes contained vectors and the cluster prototype, (G) separated heatmaps for all values of a categorical attribute.

to be good for visualizing the distribution of data attributes among the SOM-grid (3-G). The color-intensity of a node depends on the number of data items it contains; the more data items, the stronger the intensity.

We offer several normalization options. One approach takes the global maximum (of all groups/grids), whereas the other one takes a local maximum for each single group/grid. Different kinds of normalizations can also be chosen in order to handle outliers or small variabilities in the data. Depending on the underlying data, an adequate normalization technique is needed to obtain visible patterns in the data.

A drawback of the heatmaps is that it is not easy to detect if cells are homogeneous or heterogeneous. That means that it is hard to determine whether a cell contains only vectors of a specific group (i.e., in our use cases just native Japanese or native Germans) or if it is a mixed cell. For that reason we also offer another visualization. For each cell we derive the color depending on the number of group members. Therefore we assign a color (e.g. red vs. blue) to each group and mix them accordingly. As a result homogeneous (red vs. blue) and heterogeneous (purple) clusters are easy to detect (see Figure 3-D, where GL stands for “German learner” and “JN” for Japanese native). Finally, we also offer word cloud visualizations for each cell (Figure 3-B). These allow the user an overview of the values contained in a cell if the selected attribute has many categories/values.

Each of these visualizations offers different per-

spectives on the data and the user is able to interact dynamically with each of the different visualization possibilities.

2.4 Interaction

The system offers various possibilities for interaction: 1) Configuration/Encoding Interactions; 2) SOM Interactions; 3) Selection Interactions.

Configuration/Encoding Interactions: The algorithm and the visualization techniques offer many possibilities for individualized configuration, e.g., the grid dimensions of the SOM or the normalization techniques that are applied by the visualization techniques. Furthermore the cell layout can be toggled interactively from the SOM-grid to a grouped alignment. An advantage of the grouped alignment is that the typical feature clusters for each group can be determined by their position. In combination with our coloring approach, the analysts are thus able to locate the top group clusters and detect if they are homogeneous or heterogeneous (Figure 3-E). Users may also define and change visual mappings like the colors that are assigned to the attribute values.

SOM Interactions: We incorporate the idea that the analyst should be able to steer the training phase of the algorithm as well (Schreck et al., 2009). The analyst is able to enter into an iterative process that refines the analysis in each step. In each step the SOM result can be manipulated and serves as an input for the next iteration. For one, it is possible to delete cells directly on the grid. Another interactive possibility is to move cells to a

desired position and to “pin” them to this position. That means that for the next SOM training this cell is fixed. We make use of these interactions to steer the SOM-algorithm to deliver visually similar outputs. For example, if we fix a cell near the upper right corner, in the next round of training this cell and the cells similar to it will be in the same corner (e.g., in Figure 4-E the two gray cells are fixed). Finally, it is possible to break off the current training process and to restart or to investigate the current state in more detail if the analyst already perceives a pattern or discovers problems.

Selection Interactions: These interactions help to filter and select the data during the analysis process. The data that are contained in the current SOM visualization serve as input for the next iteration of the analysis work flow. Besides removing data elements directly on the SOM grid, data can be selected to be removed directly in the attribute table (Figure 4-D). This feature allows the analyst to drill down into selected data subspaces. *Details on Demand* operations also enable the user to inspect subsets of clusters. Furthermore, single cells can be selected and investigated in a separate linked detail view.

By enabling these interactions we present the analyst with the flexible possibilities for an iterative analysis process. The system first provides an overview of the data, the analyst is able to interact with the data in iterations of hypothesis formation and testing. The hypothesis testing can be done with respect to the entire data set, or with respect to a selected subset. In order to keep track of the various visualizations and interactions conducted by the analyst, we offer a visualization history that displays the developed SOM grids next to one another (e.g., Figure 5). Clicking on one of these grids will automatically bring the selected SOM to the front of the screen.

3 Use Cases

We demonstrate the added value that our approach brings to prosodic research with respect to two linguistic experiments that were originally conducted independently of this work. We take a “paired analytics” approach for an evaluation of the potential of our system (Arias-Hernandez et al., 2011). In this approach, an expert for visual analytics collaborates with a domain expert. The domain expert places their focus on tasks, hypotheses and ideas while an analysis expert operates the system.

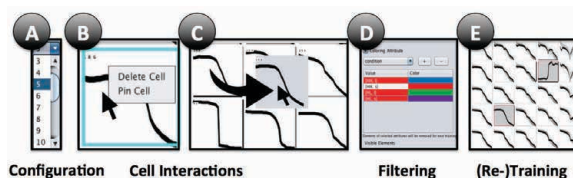


Figure 4: Interaction techniques that enable for an iterative data exploration. *Configuration Interactions* can be used to define parameters like the grid dimensions or visual mappings (e.g., selecting the attribute colors). *SOM Interactions* include the direct manipulation of the SOM-visualization (move, delete, or pin cells, begin or stop SOM training). *Selection Interactions* enable the analyst to dismiss data in each step in order to drill down into interesting data subspaces.

We are well aware that the standards for evaluation in natural language processing are quantitative in nature. There is an inherent conflict between quantitative evaluation and the rationale for using a visual analytics system in the first place. Visual analytics has the overall aim of allowing an interactive, exploratory access to an underlying complex data set. It is very difficult to quantify data exploration and cycles of hypothesis testing in the absence of a bench mark or gold standard. This is a known problem within visual analytics (Keim et al., 2010; Sacha et al., 2014), but one which cannot be addressed within the scope of this paper. The two use cases presented here should be seen as an initial test as to the added value of our system. An application to other scenarios and other use cases is planned as future work.

The use cases discussed below consist of experiments that were concerned with whether linguistic structures of a native language (henceforth L1) influence second language (henceforth L2) learning. The experiments involved Japanese native speakers vs. German learners of Japanese. The latter group had varying degrees of L2 competence. The data set consists of F_0 contours and meta data about the speakers.

3.1 Experiment 1

The first experiment investigated how native speakers of an intonation language (German) produce attitudinal differences in an L2 that has lexically specified pitch movement (Japanese).

Methods

15 Japanese native speakers and 15 German native speakers, who were proficient in the respective languages participated in the experiment. They produced the German word *Entschuldigung* and the Japanese word *sumimasen*, which both mean ‘excuse me’. The Japanese word contains a lexically specified pitch fall associated with the penultimate mora in the word, /se/. Materials were presented with descriptions of short scenes. The task was to produce the target word three times in order to attract an imaginary waiter’s attention in a crowded and noisy bar.

Our hypotheses were that Japanese native speakers would not change the F_0 contours across the three attempts, because the Japanese falling pitch accent is lexically fixed. German learners would change them, because German F_0 can be changed in order to convey attitude or emotion.

Segmental boundary annotation was carried out on the recorded raw data using Praat (Boersma and Weenink, 2011) as the first step. In Experiment 1, segmental boundaries were put between the Japanese smallest segmental unit, morae, which resulted in *—su—mi—ma—se—n—* (the straight lines signal the segmental boundaries). Then, F_0 contours were computed from the annotated data using the F_0 tracking algorithm in the Praat toolkit with the default range of 70-350 Hz for males and 100-500 Hz for females. Following the procedures of Functional Data Analysis (Ramsay and Silverman, 2009), we first smoothed the sampled F_0 contours into a continuous curve represented by a mathematical function of time $f(t)$ adopting B-splines (de Boor, 2001). Values of F_0 were expressed in semitones (=st) and the mean value was subtracted from each value, in order to minimize gender effects. After smoothing the curves we automatically carried out *landmark registration* in order to align corresponding segmental boundaries in time (Gubian et al., 2013; Ramsay et al., 2009). After these steps, the smoothed F_0 data all had the same duration.

Analysis

The analysis process of analyzing Experiment 1 is shown in Figure 5. The first SOM offers an overview for the whole dataset. The word cloud visualization additionally shows the utterances that occur in the cells (*sumimasen*, *Entschuldigung*). In a next step the data set was filtered to show only the data for *sumimasen* (Fig-

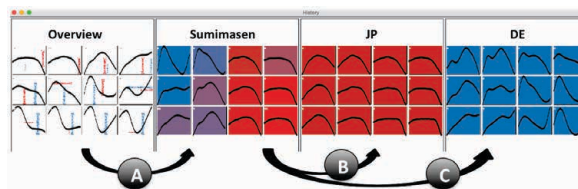


Figure 5: Experiment 1 work flow history: An overview is shown first. In the following steps data is filtered and the analysis refines stepwise into an interesting subspace. First, only the utterances *sumimasen* ‘excuse me’ are selected (A). These are then further subdivided according to speaker group (B/C): Japanese Native (JN) vs. German (DE).

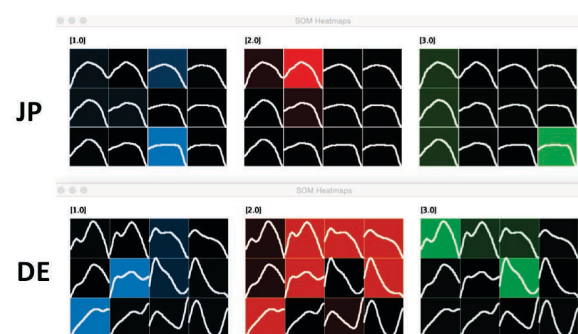


Figure 6: Experiment 1: Heatmaps for the repetition attribute for each speaker group. German learner contours clearly include more variations compared to native speaker contours.

ure 5-A) and a second SOM with only this data was trained. In the 2nd SOM in Figure 5 the cells are coloured according to the number of speaker groups in each cell. Our analyst was able to discover different pitch contours per group (blue-German cells on the left-hand side and red-Japanese cells on the right-hand side).

In order to get more details we decided to train an additional SOM for each speaker group. We simply added the relevant filters and began a new SOM training for each group (Figure 5-B/C). As a result the two visualizations now clearly show that the F_0 produced by the groups look different. For further analysis, we also opened a heatmap visualization for another attribute for each group based on the SOM-grids B and C. In Figure 6 the repetitions (1st, 2nd, or 3rd) are shown for each group. One can clearly discover that the Japanese native speakers’ (top) F_0 contours rarely vary in comparison with the German speakers (bottom).

3.2 Experiment 2

In Experiment 1 we were able to determine that German learners did not produce typical Japanese F_0 contours, namely flat F_0 followed by a drastic pitch fall, just on the basis of unannotated F_0 data. The second experiment examined whether German learners can produce this typical Japanese F_0 phonetic form in an imitation experiment. The experiment was originally conducted independently of Experiment 1.

Methods

24 Japanese native speakers and 48 German learners were asked to imitate Japanese disyllabic non-words consisting of three-morae ($/CV:CV/$) with a long-vowel. All stimuli were recorded either with a flat pitch (high-high, HH) or with a falling pitch (high-low, HL) that occurs after the long-vowel. F_0 contours produced by Japanese native speakers are expected to imitate the stimuli correctly by realizing the typical phonetic form of a Japanese pitch accent, namely a drastic pitch fall preceded by a flat F_0 . In contrast, as per the results of Experiment 1, German learners are expected not to produce this phonetic form.

In analogy to Experiment 1, segmental annotation was carried out. Segmental boundaries were put between consonants and vowels, which resulted in $-c-v-(c)c-(v)v-$. Then, F_0 contours were computed as in Experiment 1. The data contained the raw Hertz values of F_0 and additional information included data about segments, speaker information, time and landmark information for the produced pitch contour. In total 2393 data records were put into the SOM system.

Analysis

The analysis workflow for Experiment 2 is shown in Figure 7. The first SOM offers an overview for the whole dataset. This overview clearly shows two clusters for flat and falling F_0 contours (“HH”-blue and “HL”-red). On the lower most right corner, there is a red cell in the blue cluster. This type of pattern could be indicative of an error or noise in the data set.

Note that the SOM system did not know which experimental conditions the data contained. Without any information about the experimental variables, SOM detected differences across conditions. Furthermore, no other current analysis techniques enable an overview of F_0 data in this manner. Since we were interested in the phonetic re-

alization of Japanese pitch accent, we further analyzed only the data of the falling F_0 condition.

As a consequence, a second SOM containing only the “HL” contours was trained (Figure 7-A). The next step was to remove the noise from the data (Figure 7-2nd SOM). In the second SOM we discovered one cell that contains non falling F_0 contours (lower left corner). We deleted this cell and fixed/pinned the other corner cells in order to steer the SOM algorithm to produce a similar SOM in the next training phase (Figure 7-B). In the next SOM the cells are colored according to the number of speaker groups in each cell (blue-German, red-Japanese). The three cells in the lower left corner were the most frequent F_0 contours produced only by German learners of Japanese. To analyze this further, we also changed the grid based layout to the ranked group layout to show the three most frequent F_0 contours in each language group (Figure 7-C). As a result, the last SOM visualization now clearly shows that the F_0 produced by the groups look different: Japanese native speakers produced typical Japanese F_0 contours consisting of a flat F_0 before a drastic F_0 fall (Gussenhoven, 2004). The third cells from above in both of the language groups show the same F_0 forms, suggesting that some German native speakers produced F_0 contours that were very similar to those of Japanese native speakers. Note however, that the most frequent contours produced by German learners clearly differed from the Japanese contours. Finally, one of the most important contributions of the SOM system was that it delivered us the findings without the necessity of having first manually annotated a large amount of data, saving personell costs.

4 Conclusion

We provide an interactive system for the analysis of prosodic feature vectors. To complement other state of the art techniques we make use of machine learning in combination with interactive visualizations. We implemented an iterative process using chains of SOM-trainings for a step-by-step refinement of the analysis. We show with real experiment data that the system supports linguistic research. Importantly, the analysis allows for a clustering of F_0 contours that works without time-intensive and possibly subjective manual intonational analysis. The clustered contours can be subjected to intensive phonological analy-

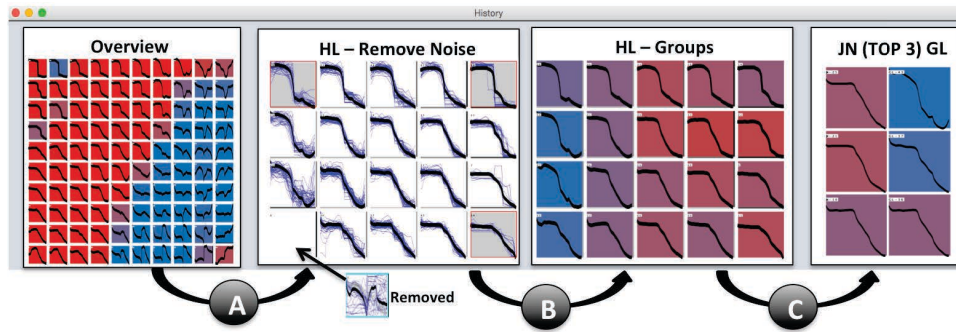


Figure 7: Experiment 2 work flow history: An overview is shown first. Then only “HL” F_0 contours (red colored cells) are selected (A). The researcher interacted directly with the second SOM: The noise cluster was removed (bottom left corner) and the other corner cells were fixed in order to steer the SOM algorithm to produce a visually similar SOM for the next training (B). The resulting SOM reveals blue clusters on the left hand side. Changing the layout to the top clusters per group (C) allows for a better comparison.

sis and furthermore allow the potential detection of more fine-grained phonetic differences across conditions. The analyses hence provide an important first step that the linguist can then focus on for subsequent analysis. For example, it is very easy to filter the data (e.g., examine only a subset of the data) or to adjust the grid size. More importantly, the approach is advantageous for an analysis of L2 data, since the learners’ language has a dynamic character (Selinker, 1972) and it is difficult to determine intonational categories beforehand. Our SOM approach is generalizable to all kinds of data for which feature vectors can be derived, including other linguistic features as intensity, amplitude or duration.

We learned that the visualization of F_0 contours provides the most intuitive access for an understanding of the underlying data. One reason is that the F_0 contour can be visually inspected and directly related to meta data (e.g., through colors). Even without time-intensive manual annotation of F_0 contours, we could clearly see the differences between L1 and L2 performance despite the different characteristics of the two experimental data sets. We visualized and animated the SOM training phase and presented this to the researcher as well. This may seem unnecessary, but experience has shown that it helps users that are not experienced with ML to better understand the processes.

We applied our technique to two different datasets. A comparison of the achieved results shows that our approach works very well “out of the box” with preprocessed data and also with less effort on the preprocessing. To overcome the prob-

lem of handling less preprocessed data we added simple methods that turned out to be sufficient in order to reveal new insights. The system helped us to handle unexpected outliers or noise in the data. All the F_0 contours that do not match the major clusters of the SOM-algorithm are assigned to a few single cells. The data in these cells could easily be removed.

We plan to make the system available for other researchers in the future and are considering several expansions as well. For one, other machine learning and visualization techniques could be added for additional or further tasks. We also could try to support the user more in detecting interesting subspaces in the data. It is possible, for instance to visualize an overview of attribute-heatmaps that enables the human to detect patterns in each iteration.

In sum, this paper has presented an innovative and promising new approach for the automatic analysis of prosodic data. Key components are that prosodic data is translated into vectors that can be processed and analyzed further by SOM techniques and presented to the user as an interactive visual analytic system.

Acknowledgments

We gratefully acknowledge the funding for this work, which comes from the Research Initiative LingVisAnn and the research project “Visual Analytics of Text Data in Business Applications” of the University of Konstanz.

References

- Richard Arias-Hernandez, Linda T Kaastra, Tera Marie Green, and Brian Fisher. 2011. Pair analytics: Capturing reasoning processes in collaborative visual analytics. In *System Sciences (HICSS), 2011 44th Hawaii International Conference on*, pages 1–10. IEEE.
- Mary E. Beckman, Julia Hirschberg, and Stefanie Shattuck-Hufnagel. 2005. The original ToBI system and the evolution of the ToBI framework. In S.-A. Jun, editor, *Prosodic Typology – The Phonology of Intonation and Phrasing*. Oxford University Press.
- Paul Boersma and David Weenink. 2011. Praat: Doing phonetics by computer [computer program] version 5.2.20.
- Carl de Boor. 2001. *A Practical Guide to Splines*. Springer, New York.
- Michele Gubian, Yuki Asano, Salomi Asaridou, and Francesco Cangemi. 2013. Rapid and smooth pitch contour manipulation. In *Proceedings of the 14th Annual Conference of the International Speech Communication Association, Lyon, France*, pages 31–35.
- Carlos Gussenhoven. 2004. *The Phonology of Tone and Intonation*. Research surveys in linguistics. Cambridge University Press, Cambridge. 2003065202 Carlos Gussenhoven. ill. ; 24 cm. Includes bibliographical references (p. 321-344) and index.
- Daniel Keim, Gennady Andrienko, Jean-Daniel Fekete, Carsten Görg, Jörn Kohlhammer, and Guy Melançon. 2008. *Visual analytics: Definition, process, and challenges*. Springer.
- Daniel A. Keim, Jörn Kohlhammer, Geoffrey P. Ellis, and Florian Mansmann. 2010. *Mastering the Information Age - Solving Problems with Visual Analytics*. Eurographics Association.
- Teuvo Kohonen. 2001. *Self-organizing Maps*, volume 30. Springer.
- Rudolf Mayer, Jakob Frank, and Andreas Rauber. 2009. Analytic comparison of audio feature sets using self-organising maps. In *Proceedings of the Workshop on Exploring Musical Information Spaces, in Conjunction with ECDL*, pages 62–67.
- Julia Moehrmann, Andre Burkovski, Evgeny Baranovskiy, Geoffrey-Alexej Heinze, Andrej Rapoport, and Gunther Heidemann. 2011. A discussion on visual interactive data exploration using self-organizing maps. In *Advances in Self-Organizing Maps - 8th International Workshop, WSOM 2011, Espoo, Finland, June 13-15, 2011. Proceedings*, pages 178–187.
- James Ramsay and Bernard. W. Silverman. 2009. *Functional Data Analysis*. Springer.
- James Ramsay, Giles Hookers, and Spencer Graves. 2009. *Functional Data Analysis with R and MATLAB*. Springer.
- Dominik Sacha, Andreas Stoffel, Florian Stoffel, Bum Chul Kwon, Geoffrey P. Ellis, and Daniel A. Keim. 2014. Knowledge generation model for visual analytics. *IEEE Transactions on Visualization and Computer Graphics*, 20(12):1604–1613.
- Tobias Schreck, Jürgen Bernard, Tatiana Tekuov, and Jörn Kohlhammer. 2009. Visual cluster analysis of trajectory data with interactive Kohonen maps. *Palgrave Macmillan Information Visualization*, 8:14–29.
- Tobias Schreck, 2010. *Visual-Interactive Analysis With Self- Organizing Maps — Advances and Research Challenges*, pages 83–96. Intech.
- Larry Selinker. 1972. Interlanguage. *IRAL-International Review of Applied Linguistics in Language Teaching*, 10(1-4):209–232.
- Ana Cristina C Silva, Ana Cristina P Macedo, and Guilherme A Barreto. 2011. A SOM-based analysis of early prosodic acquisition of English by Brazilian learners: preliminary results. In *Advances in Self-Organizing Maps*, pages 267–276. Springer.
- Ryszard Tadeusiewicz, Wieslaw Wszolek, Antoni Izworski, and Tadeusz Wszolek. 1999. The methods of pathological speech visualization [using Kohonen neural networks]. In *Engineering in Medicine and Biology, 1999. 21st Annual Conference and the 1999 Annual Fall Meeting of the Biomedical Engineering Society, BMES/EMBS Conference, 1999. Proceedings of the First Joint*, volume 2, pages 980 vol.2–, Oct.
- James J. Thomas and Kristin A. Cook. 2006. A visual analytics agenda. *IEEE Computer Graphics and Applications*, 26(1):10–13.
- Juha Vesanto. 1999. SOM-based data visualization methods. *Intelligent Data Analysis*, 3(2):111–126.
- Nigel G Ward and Joshua L Mccartney. 2010. Visualization to support the discovery of prosodic contours related to turn-taking.
- Nigel G Ward. 2014. Automatic discovery of simply-composable prosodic elements. In *Speech Prosody*, volume 2014, pages 915–919.