Acta Cybernetica **22** (2015) 229–242.

Expanding Small Corpora to Aid People with Communication Impairment*

Gyula Vörös†

Abstract

Difficulties in the communication of people with various movement and cognitive disorders may be alleviated by means of pictorial symbols. Automatic transformation of symbol sequences to natural language is of high importance. Performing this task by defining all valid sentences manually would require a large amount of work. We show that a small initial seed corpus is sufficient, which can be expanded automatically by generating candidate sentences and filtering them using N-gram statistics from a much larger corpus. The method is evaluated on a seed corpus containing dialogues, collected from an English language learning website. The ratio of useful sentences in the expanded corpus is 3–4 times bigger than in the set of unfiltered candidate sentences. We also use a manually constructed corpus for further evaluation. To demonstrate the practical applicability of the method, we have implemented a sentence production prototype that performs the transcription of symbol sequences to natural language. The system produces new and meaningful sentences and thus it can considerably decrease the size of the corpus needed, while it can increase the variability of sentences.

Keywords: augmentative and alternative communication, natural language processing, corpus statistics

1 Introduction

People with severe communication impairment, caused by traumatic brain injury, cerebral palsy or other conditions face huge challenges in their daily lives. A significant portion of the affected people also has cognitive disorders which makes them unable to produce or recognize letters. Communication is often facilitated by selecting pictorial symbols from a list called *communication board*. Such boards may be

*This work was carried out as part of the EITKIC 12-1-2012-0001 project, which is supported by the Hungarian Government, managed by the National Development Agency, financed by the Research and Technology Innovation Fund and was performed in cooperation with the EIT ICT Labs Budapest Associate Partner Group.

†Eötvös Loránd University, Faculty of Informatics, Pázmány Péter sétány 1/C, H-1117, Budapest, Hungary, E-mail: vorosgy@gmail.com

DOI: 10.14232/actacyb.22.1.2015.14

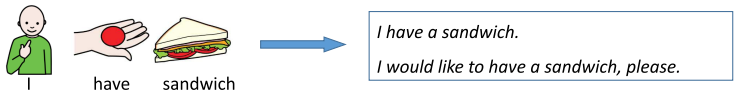

Figure 1: **Translating symbol sequences to sentences.** The message constructed from the symbols '*I*', '*have*' and '*sandwich*' is ambiguous, it can be translated to either '*I have a sandwich*' or '*I would like to have a sandwich*'.

low-tech tools (e.g., pictures on a piece of cardboard), or electronic devices. These techniques belong to the field of Augmentative and Alternative Communication (AAC).

A single selected communication symbol may be a message in itself, but more elaborate messages can also be composed from multiple symbols. In a sense, the users of symbol-based communication express themselves in their own language of symbol sequences. The sequences are often very brief and most of the words are omitted from them [3]. In some cases, the order of symbols within a message is also arbitrary [14]. These messages are often ambiguous, therefore the communication partner has to ask questions actively to find out the intended meaning. The lack of means to expressive communication forces many AAC users to interact only with a relatively narrow circle of people [2]. We aim to alleviate this problem by automatically transforming symbol sequences to sentences in natural language (Figure 1). This way, AAC users should be able to control their conversations better, and convey their wishes and thoughts more effectively.

There have been some attempts in the literature to generate sentences from symbol sequences in real time. These approaches are mostly rule-based, and language-dependent [3, 4, 7, 9].

A method that is satisfactory in practice is to define all possible sentences beforehand, for example, by storing them in a text corpus, and retrieving them during the conversation [1]. The manual construction of a such a corpus would require an enormous amount of work. The corpus would also need maintenance: for example, new sentences should be added as the vocabulary changes. The whole procedure should also be performed for every user separately, since they have different needs.

We propose a method for *corpus expansion* to help symbol-based communication. The idea is that only a small set of sentences is needed in a seed corpus, which is then automatically expanded, based on the available symbol set. The process of expanding a corpus automatically has been studied before [12], but, to the best of our knowledge, has not in the context of augmentative and alternative communication. In this paper, the method is evaluated on a seed corpus collected from an external source as well as on a manually constructed seed corpus.

2 Methods

Our method consists of two main stages. The first one is *corpus expansion* (Figure 2). This is a preparatory stage, which entails the following steps:

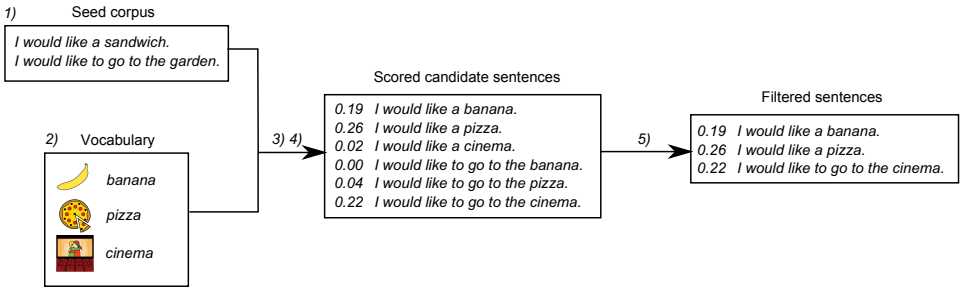

Figure 2: **Corpus expansion stage.** From a small seed corpus and a vocabulary, candidate sentences are generated, which are then scored and filtered to eliminate uncommon entries.

1. A *seed corpus* is defined.
2. The *vocabulary* (i.e., the collection of available symbols and the words or phrases they denote) is defined.
3. *Candidate sentences* are generated from the seed corpus and the vocabulary by performing *substitutions*.
4. *Scores* are associated to the candidate sentences based on how frequent their constituent N-grams are in a suitably large corpus.
5. The candidate sentences are *filtered* by selecting an appropriate score threshold.

The second stage is *symbol transformation*, in which the symbol sequences given by the impaired user are transformed to natural language:

1. A *symbol sequence* is defined by the user.
2. *Relevant sentences* are retrieved from the filtered corpus.
3. Sentences with the top K highest scores are presented to the user.

2.1 Candidate sentence generation

The goal of candidate sentence generation is to produce a large and diverse corpus. Optimally, the result should contain all sentences that correspond to some symbol sequences constructed from the vocabulary.

Our approach is based on replacing the *content words* (i.e., words that carry some semantic content) of the sentences in the seed corpus with words from the vocabulary. Using a part-of-speech tagger, open class words (nouns, verbs, adjectives and adverbs) can be marked as content words. We utilize this approach in Section 3.1. If the size of the seed corpus allows for it, the manual tagging of content words may also be feasible, as in Section 3.2.

Once the content words are known, candidate sentences are generated by simply replacing them with words in the vocabulary in every possible way. The new sentences will be similar to the initial ones in structure, but different in content. For example, from the sentence '*I would like a sandwich*' in the initial corpus, and the from the symbol for '*banana*' in the vocabulary, the sentence '*I would like a banana*' is generated.

Most of the candidate sentences are not useful in any situations. Consider the initial sentence '*I would like a soda from the fridge*', and the symbol '*cinema*'. The candidate sentence '*I would like a cinema from the fridge*' is generated. Including nonsense sentences in the expanded corpus has two disadvantages. First, it makes the corpus bigger which possibly slows searches down. Second, these sentences might be erroneously retrieved during the transformation stage: the symbol sequence '*like cinema*' might be transformed to the aforementioned example. Therefore, the candidate sentences need to be filtered in some way.

To reliably recognize inconsistencies (e.g., a cinema does not belong into the fridge), either an elaborate common sense database, or statistics from a huge corpus are necessary. In this paper, the latter approach is used. Corpus statistics can show that the second sentence has very low probability thus it can be discarded.

When the seed corpus is constructed manually, it might also be feasible to set up strict substitution rules for each sentence, instead of using corpus statistics. For example, a rule may specify that only food items can be substituted to the sentence '*I would like a ⟨something⟩ from the fridge*'. In Section 3.2, we use such rules to evaluate our method and thus demonstrate that using corpus statistics is sufficient.

2.2 Scoring

Because of sparsity, the probabilities of whole sentences longer than 5-6 words cannot be estimated reliably by simply counting their occurrences in a large corpus. The number of occurrences of the individual N-grams (i.e., sequences of N words) in the sentences can be used instead. A number of widely used language models exist to perform this probability estimation, but they are usually suitable for estimating the conditional probability of a word given the previous ones.

To score the sentences, versions of the NgramSum method were used in this study, described in [11].

For a given N (typically $N = 2, 3, 4, 5$ are used), the NgramSum of a k long text fragment is defined as

$$N\text{gramSum}(w_1 \dots w_k) = \sum_{i=1}^{k-N+1} \text{Count}(w_i \dots w_{i+N-1}),$$

where $\text{Count}(s)$ is the number of times the word sequence s occurs in a given (usually very large) corpus. In other words, the NgramSum method sums the counts of all the N-grams in a text fragment.

When sentences generated by word substitution are compared to each other, the NgramSum method is used to capture how well the new word fits in the local con-

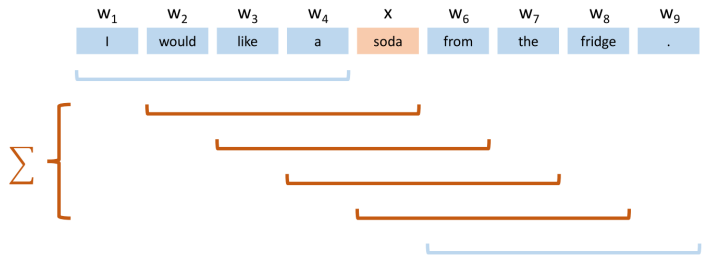

Figure 3: **Computation of the 4Score for a sentence generated by substitution.** The score is the sum of the counts of the 4-grams in the sentence that contains the substituted word.

text. N-grams that do not contain the substituted word are irrelevant (Figure 3). The score of a substitution where the word x is substituted to the i^{th} location can be computed as

$$N\text{Score}(w_1 \dots w_k \leftarrow^i x) = N\text{gramSum}(w_{i-N+1} \dots w_{i-1} x w_{i+1} \dots w_{i+N-1}).$$

The magnitude of the previous score (based on the raw count of occurrences) depends on many things, including the size of the corpus and the parameter N. For a given vocabulary V, the normalized score of a sentence generated by word substitution is

$$\text{Norm}N\text{Score}(w_1 \dots w_k \leftarrow^i x) = \frac{N\text{Score}(w_1 \dots w_k \leftarrow^i x)}{\sum_{v \in V} N\text{Score}(w_1 \dots w_k \leftarrow^i v)},$$

the score divided by the sum of scores for all possible substitutions.

The score functions above are only defined for single word substitutions. However, it is easy to generalize Nscore to work for multiple word substitutions. For example, when two words are substituted at the same time, into places i and j, respectively, NScore can be defined as:

$$N\text{Score}(w_1 \dots w_k \leftarrow^i x \leftarrow^j y) = N\text{gramSum}(w_{i-N+1} \dots w_{i-1} x w_{i+1} \dots w_{j-1} y w_{j+1} \dots w_{j+N-1}).$$

The normalized version is also straightforward:

$$\text{Norm}N\text{Score}(w_1 \dots w_k \leftarrow^i x \leftarrow^j y) = \frac{N\text{Score}(w_1 \dots w_k \leftarrow^i x \leftarrow^j y)}{\sum_{u,v \in V} N\text{Score}(w_1 \dots w_k \leftarrow^i u \leftarrow^j v)}.$$

Modified versions of these scores, written as Mod$N$gramSum, Mod$N$Score and ModNorm$N$Score were also used. They are similar to the respective original scores, with the only difference being that they are defined as zero if any of the N-gram counts used to compute them is zero.

These scores are used during corpus expansion to discard generated sentences that have lower scores than a certain threshold. The scores can also be used during the transformation stage rank the relevant sentences.

3 Results

3.1 Expansion of a natural corpus

We performed a corpus expansion test on a seed corpus collected from an external source, and estimated the ratio of correct sentences in the extended corpus.

The initial corpus was downloaded from the language learning website `eslfast.com`. Dialogues related to food were collected. Only sentences that were at most ten words long were retained, as longer ones tended to be overly specific, and they would have not been useful in a communication aid. After this filtering step, 429 sentences remained in the initial corpus.

To select the content words, the Stanford part-of-speech tagger was used with the English bidirectional model [13]. Words tagged as nouns were regarded as content words.

The vocabulary we used for the expansion was obtained the following way. We collected the filenames of the English ARASAAC symbol set [8]. Only files that were single words (i.e., contained only letters), longer than two characters were retained. Words that did not have a noun synset in WordNet [6] were discarded. We performed a further filtering step: we kept only those words that were present among the twenty thousand most frequent 1-grams according to the Google Books N-gram corpus [5]. After this, 2408 words remained in the vocabulary.

The corpus expansion was performed in two ways. A new corpus of the same size as the initial one was generated in both. As a *baseline*, we generated 429 new sentences by replacing content words with random words from the vocabulary. The generated sentences in the baseline were not filtered in any way. As a *filtered corpus*, again we generated sentences by substituting words in every possible way, but this time, we assigned scores to the resulting sentences and filtered them. The scores were computed using the Norm5Score method described in Section 2.2, using the BerkeleyLM software [10] with the Google Books N-gram corpus [5] as the basis of N-gram statistics. Two rounds of filtering steps were performed. First, to ensure that the resulting corpus will be diverse enough, for each sentence in the initial corpus, only the ten highest scoring sentences generated from them were retained. Then, the overall 429 highest scoring sentences were retained from them.

For evaluation, 100–100 sentences were sampled with uniform distribution from the original, the baseline, and the filtered corpora. Two independent human annotators received the sentences for evaluation. The annotators did not have communication impairment, and both of them were fluent in English. They indicated whether they can imagine a real life situation when the sentence would be potentially useful for communication. The 300 sentences were shuffled in a random order, so the annotators had no way to know the sources of the individual sentences. Both annotators received the same sentences, in order to compute their agreement.

The results and some example sentences are shown in Table 1. The judges agreed in their judgements 84% of the time. The inter-annotator agreement computed by Cohen's kappa was $\kappa = 67\%$, which is generally considered a moderate agreement.

Table 1: **Results of annotation from three corpora with some example sentences.** The numbers show how many sentences each of the annotators found useful (out of 100).

corpus	1st	2nd	examples
initial	100	95	So, you like red apples better? I want to buy some meat. You should just get the basics. What are you having for lunch? Thanks for your help.
baseline	24	15	I would like four sunflower of ground beef. I'm really not going to cook scarecrow. I bought my food from the herd today. I have a barrels for some chicken and potatoes. What would you like food?
filtered	70	60	All I usually have is some fruit for breakfast. Why don't you put it in your pocket? What kind of work do you want to make? Let me have a copy. What are you going to eat with your hands?

3.2 Expansion of a hand-crafted corpus

In the previous section, the expansion was performed on a corpus collected from an external source. Since the contents of the corpus were not controlled, manual evaluation was necessary. In this section, a very small corpus is constructed manually, which allows for semi-automatic evaluation.

For this experiment, we defined a vocabulary of 21 words (Table 3). Some of the words are different types of foods, but there are also drinks, liquid containers, and belongings of the speaker.

Seven simple sentences (Table 2) were written as an initial corpus that could be useful when an impaired person buys some food in a store or a cafeteria. We refer to these sentences as *templates*, because they contain one or two special symbols instead of content words, which are later replaced by the algorithm. The sentences allow the speaker to request a specific item, enquire about the price of a specific item, and ask the shop assistant to take out the proper amount of payment from the speaker's wallet (since the user is presumably disabled and unable to do this).

From these templates and words, 1768 different sentences were generated by replacing the special symbols with words in every possible way. The '*a(n)*' indefinite article was resolved in every generated sentence (i.e., it was replaced by '*a*' before a consonant and by '*an*' before a vowel). The same thing could have been possibly handled using n-gram statistics, provided the distribution of the vowels and consonants for the first letters of the words were balanced, but that was not

Table 2: **The seven sentence templates of the hand-crafted initial corpus.** The word categories in the angled brackets mark the places of the content words that are filled by the corpus expansion algorithm. Word category information was not used during expansion, only during evaluation. The indefinite article '*a(n)*' was resolved automatically in each generated sentence.

purpose	#	template
request item	1	I would like to have a(n) ⟨*food*⟩.
	2	I would like to have a(n) ⟨*topping*⟩ ⟨*sandwich*⟩.
	3	I would like to have a(n) ⟨*container*⟩ of ⟨*drink*⟩.
enquire about price	4	How much is the ⟨*food*⟩?
	5	How much is the ⟨*topping*⟩ ⟨*sandwich*⟩?
	6	How much is the ⟨*container*⟩ of ⟨*drink*⟩?
ask for help	7	Please, take the money out of my ⟨*belonging*⟩!

Table 3: **Vocabulary for expanding the hand-crafted initial corpus.** Category information is not used by the expansion algorithm.

category		words
food	fruit	banana, strawberry, lemon, orange, apple, peach
	drink	water, milk, coffee
	topping	tuna, sausage, cheese, egg, chicken, ham
	sandwich	sandwich
other	container	cup, glass
	belonging	pocket, bag, wallet

the point, since grammatical rules like this are very easy to follow in a rule-based way.

The replacements were scored using the Norm$N$Score and ModNorm$N$Score methods for $N = 2, 3, 4$. As in our first experiment, the BerkeleyLM software with the Google Books N-gram corpus was used as the source of N-gram statistics.

The overwhelming majority of the generated sentences are nonsense (e.g., '*I would like to have a sausage lemon.*'), only a very small portion of them is appropriate. Annotating them one by one even by sampling would require a large amount of work. Fortunately, the controlled nature of the initial corpus and the vocabulary enables the annotation of the sentences by large groups. To this end, we designed the vocabulary and the sentence templates in such a way that makes possible to determine if a sentence is *valid* or not. A generated sentence is *valid* only if the words that are substituted in it correspond to the word categories shown in Table 2. The categories in the templates were defined with the following guidelines in mind:

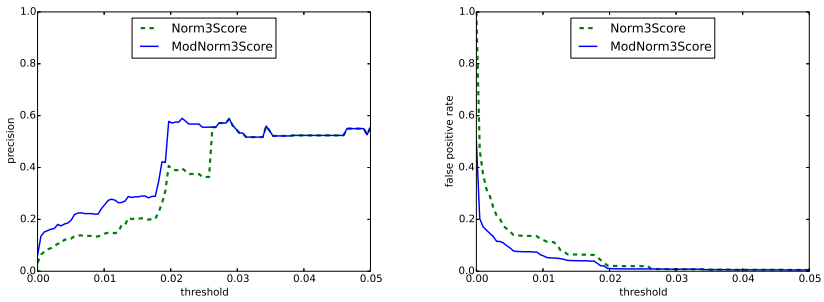

Figure 4: **Comparison of different thresholdings with the Norm3Score and the ModNorm3Score methods.** The experiments were performed based on the hand-crafted initial corpus. The left figure shows the precision values (higher is better). The right figure shows the false positive rates (lower is better).

- All valid sentences should be grammatically correct.
- All valid sentences should make sense in a food buying scenario.

These rules exclude some sentences which can be considered somewhat meaningful, for example, '*I would like to have an orange coffee*' or '*How much does the cup cost*', however, these are either unusual or not useful when someone wants to buy food.

According to these rules, only 59 of the 1827 generated sentences are valid, which corresponds to a 3.23% precision rate if all of them are included in the resulting corpus.

To score the sentences, two different methods were considered: Norm$N$Score and ModNorm$N$Score (the modified version is defined as zero if any of the N-gram counts in it are zero). A comparison between the two can be seen in Figure 4.

Statistics about different thresholdings based on the scores of sentences can be seen on Figure 5. Here, only ModNorm$N$Score is used as it proved to be clearly superior than Norm$N$Score before.

4 Discussion

4.1 Corpus expansion

The results described in Section 3.1 show that it is possible to automatically filter an expanded corpus, and achieve a significant increase in the ratio of useful sentences. Though the second judge was a bit stricter overall, the trend is clear: the filtered corpus contains 3–4 times as many correct sentences as the baseline (Table 1).

The results of Section 3.2 demonstrate that corpus statistics can be used instead of manually defined substitution rules. In the comparison between the two scoring methods, seen in Figure 4, the modified version clearly outperforms the other. The

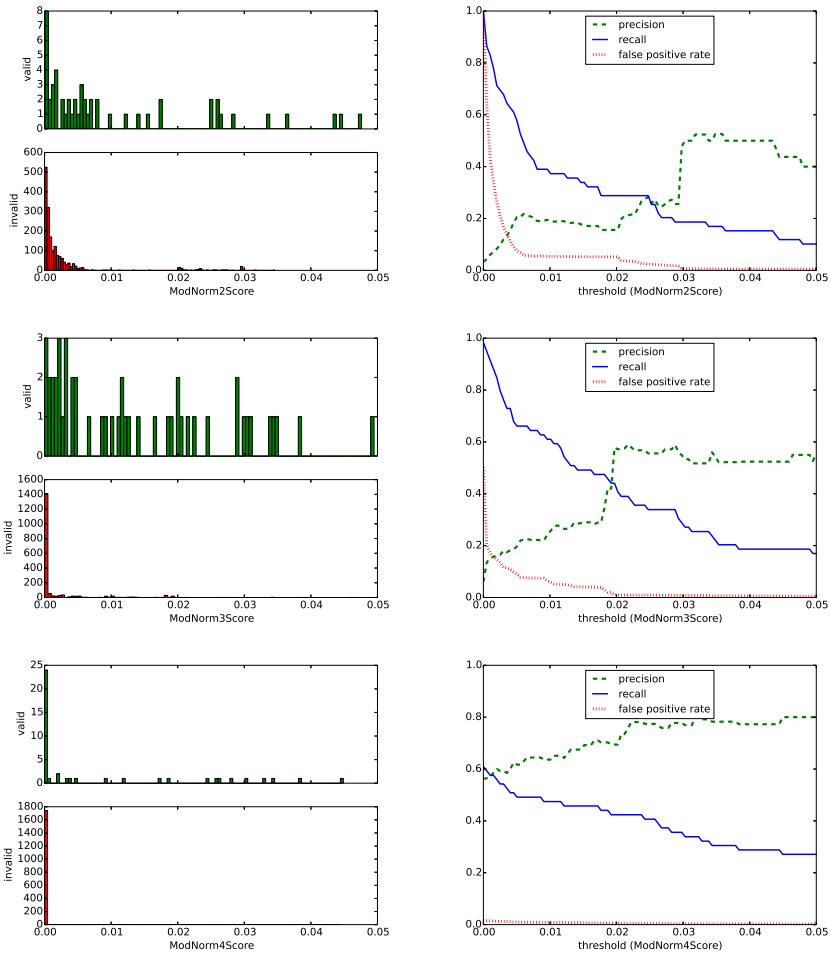

Figure 5: **Distribution of scores for valid and invalid sentences (left), and precision, recall and false positive rates of different thresholds (right).** From top to bottom, the ModNorm$N$Score method is used with different N-gram lengths ($N = 2, 3, 4$). On the left hand side, histograms of the scores of valid (top) and invalid (bottom) sentences can be seen. The diagrams on the right hand side show statistics about the corpora generated using different thresholds.

reason is that it eliminates possibilities like the following sentence: '*I would like to have a sandwich tuna*'. This sentence receives a relatively high Norm3Score, because the 3-gram '*have a sandwich*' is very common, even though the 3-gram '*a sandwich tuna*' has a zero count.

The histograms of the scores (Figure 5) are very different for valid and invalid sentences, which suggests that thresholding can be used effectively to separate them. The histograms for ModNorm3Score are particularly convincing: here, the large majority of invalid sentences have zero or very low scores, while only 3 of the valid sentences are rated as very improbable. Other statistics computed with different thresholds are also promising: for example, using the ModNorm3Score values, approximately 80% of the invalid sentences can be filtered out while retaining almost all of the valid sentences. The precision (i.e., the rate of valid sentences among the accepted ones) can be easily improved from 3.23% to 15% or better by raising the threshold higher, depending on how high the recall has to be.

Using N value larger than 3, the precision can be improved to much higher levels, although this results in a very low recall rate. The explanation is that longer N-grams are sparse in the Google N-gram corpus, but are also more reliable if they are present.

4.2 Symbol transformation prototype

We demonstrate the practical applicability of the method by describing a system that retrieves relevant sentences for symbol sequences from the automatically expanded corpus described in Section 3.2. We computed the Mod3NormScore of the generated sentences, however, we did not filter the resulting sentences, to evade the problem of setting a somewhat arbitrary threshold.

The transformation process is best explained by an example. Suppose the user has constructed a symbol sequence from the vocabulary. The words that belong to the symbols are '*I*', '*have*', '*chicken*' and '*sandwich*'. First, the relevant sentences in expanded corpus are retrieved: only those sentences are considered that contain all the input words. These may be '*I would like to have a chicken of sandwich*', '*I would like to have a sandwich chicken*', etc. From the filtered sentences, the one that had the highest score during the automatic expansion is returned to the user, which is '*I would like to have a chicken sandwich*'.

A consideration is that usually the most general sentence should be retrieved. For example, if the input words are '*I*', '*have*' and '*sandwich*', then the sentence '*I would like to have a sandwich*' is preferred to '*I would like to have a chicken sandwich*', even though the two sentences may both be very probable in themselves. This consideration can be taken into account by dividing the score with the length of the current sentence. This way, shorter sentences are favoured, which are usually more general.

Some sentences that were retrieved with this simple method from the corpus are shown on Figure 6. These examples show that the system produces meaningful sentences, which were not present in the seed corpus. Moreover, this is performed without using hand-crafted substitution rules or any extensive fine-tuning of the

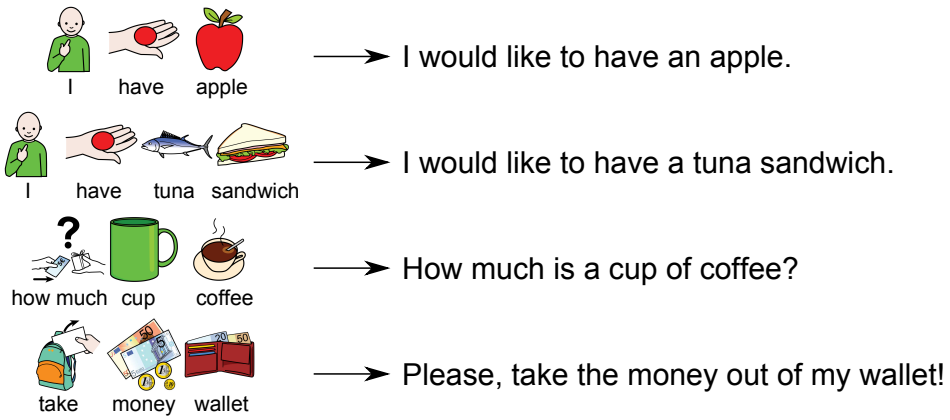

Figure 6: **Transforming symbol sequences to natural language.** Some examples of automatically generated sentences, retrieved from the corpus based on symbol sequences.

system.

It should be noted, however, that the pure statistical approach is not perfect. With some testing, we have found a few cases where the highest scoring sentence is not the best one. For example, the score of the incorrect sentence '*How much is an egg of sandwich?*' is slightly higher than the correct one: '*How much is the egg sandwich?*'. Some of these inconsistencies might be eliminated by careful tuning, such as trying different scoring schemes and thresholds. This would, however, negate one of the main advantage of our method: the reduced amount of work required to prepare a communication aid. A much more promising approach is to allow the impaired AAC user to give feedback upon the quality of the produced sentences, and enable the system to adapt to the choices of the user.

5 Conclusion

The method presented in this paper can be used to expand either a natural corpus or a manually defined set of sentence templates. In the either case, corpus statistics is used to determine which words of the vocabulary should be substituted to the templates. Without corpus statistics, the manual definition of substitution rules would be necessary.

In this paper, scoring rules that sum the counts of constituent N-grams are used. More work seems to be necessary to explore other scoring schemes, and determine their characteristics. The simple elimination of substitutions where some of the counts are zero clearly improve the results (Figure 4), which indicates that multiplicative approaches are particularly promising.

The scoring methods themselves are language-independent. The experiments described in this paper did not use explicit grammatical rules (apart from the form

of indefinite articles). This was mainly possible because the substituted words were nouns, and in English, the form of nouns changes relatively rarely. However, if other parts of speech (e.g., verbs) are substituted, or the method is adapted to agglutinative languages such as Hungarian, then more elaborate morphological processing is necessary.

When sentences are constructed from symbol sequences (either by retrieving them from a previously generated corpus or by assembling them in real time), very often there are multiple possibilities (Figure 1). The user should have the option to select from the sentences. The selection process is complicated by the users' illiteracy. Displaying a menu with the selectable sentences can not be used, because the user is unable to read them. A possible solution might be to generate speech from the sentences and play them to the user, who can indicate whether the current sentence is what he meant or not. To overcome the inherent errors of the purely statistical approach, the feedback received from the user should be utilized to improve the accuracy of future queries.

Acknowledgements

I would like to thank the work of the people involved with the project, especially András Lőrincz, András Sárkány, Anita Verő, Balázs Pintér and Brigitta Miksztai-Réthey.

References

[1] Arnott, J. L. and Alm, N. Towards the improvement of augmentative and alternative communication through the modelling of conversation. *Comput. Speech Lang.*, 27(6):1194–1211, 2013.

[2] Blackstone, S. W., Dowden, P., Berg, M. Hunt, Soto, G., Kingsbury, E., Wrenn, M., and Liborin, N. Augmented communicators and their communication partners: A paradigm for successful outcomes. In *Conference Proceedings CSUN*, 2001.

[3] Karberis, G. and Kouroupetroglou, G. Transforming spontaneous telegraphic language to well-formed Greek sentences for alternative and augmentative communication. In *Methods and Applications of Artificial Intelligence*, pages 155–166. Springer, 2002.

[4] McCoy, K. F., Pennington, C. A., and Badman, A. Luberoff. Compansion: From research prototype to practical integration. *Natural Language Engineering*, 4(01):73–95, 1998.

[5] Michel, J-B., Shen, Y. K., Aiden, A. P., Veres, A., Gray, M. K., Pickett, J. P., Hoiberg, D., Clancy, D., Norvig, P., Orwant, J., et al. Quantitative analysis of culture using millions of digitized books. *Science*, 331(6014):176–182, 2011.

[6] Miller, G. A. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41, 1995.

[7] Pahisa-Solé, J. and Herrera-Joancomartí, J. Pictogram AAC prototype software that expands telegraphic language into natural language in Catalan and Spanish. In *ISAAC*, 2014.

[8] Palao, S. ARASAAC. http://catedu.es/arasaac, 2014.

[9] Patel, R., Pilato, S., and Roy, D. Beyond linear syntax: An image-oriented communication aid. *Assistive Technology Outcomes and Benefits*, 1(1):57–66, 2004.

[10] Pauls, A. and Klein, D. Faster and smaller n-gram language models. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 258–267. Association for Computational Linguistics, 2011.

[11] Sinha, R. and Mihalcea, R. Combining Lexical Resources for Contextual Synonym Expansion. In *Proceedings of the International Conference RANLP*, pages 404–410, 2009.

[12] Tao, T., Wang, X., Mei, Q., and Zhai, CX. Language model information retrieval with document expansion. In *Proceedings of the main conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics*, pages 407–414. Association for Computational Linguistics, 2006.

[13] Toutanova, K., Klein, D., Manning, C. D., and Singer, Y. Feature-rich part-of-speech tagging with a cyclic dependency network. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1*, pages 173–180. Association for Computational Linguistics, 2003.

[14] Wiegand, K. and Patel, R. Non-syntactic word prediction for AAC. In *Proceedings of the Third Workshop on Speech and Language Processing for Assistive Technologies*, pages 28–36. Association for Computational Linguistics, 2012.