A User-Powered American Sign Language Dictionary

Danielle Bragg Kyle Rector Richard E. Ladner

Computer Science & Engineering
DUB Group, University of Washington
Seattle, WA 98195 USA
{dkbragg,rectorky,ladner}@cs.washington.edu

ABSTRACT
Students learning American Sign Language (ASL) have trouble searching for the meaning of unfamiliar signs. ASL signs can be differentiated by a small set of simple *features* including hand shape, orientation, location, and movement. In a feature-based ASL-to-English dictionary, users search for a sign by providing a query, which is a set of observed features. Because there is natural variability in the way signs are executed, and observations are error-prone, an approach other than exact matching of features is needed. We propose ASL-Search, an ASL-to-English dictionary entirely powered by its users. ASL-Search utilizes Latent Semantic Analysis (LSA) on a database of feature-based user queries to account for variability. To demonstrate ASL-Search's viability, we created ASL-Flash, a learning tool that presents online flashcards to ASL students and provides query data. Our simulations on this data serve as a proof of concept, demonstrating that our dictionary's performance improves with use and performs well for users with varied levels of ASL experience.

Author Keywords
American Sign Language (ASL); Dictionary; Education; Crowdsourcing; Latent Semantic Analysis (LSA); Information Retrieval (IR)

ACM Classification Keywords
H.3.3. Information Storage and Retrieval: Information Search and Retrieval; H.5.0 Information Interfaces and Presentation (e.g., HCI): General; K.3.1 Computers and Education: Computer Uses in Education

INTRODUCTION
Second language acquisition is an important human activity supported in K-12, higher, and continuing education institutions. According to the Modern Language Association, more than 1.6 million higher education students were enrolled in foreign language classes in 2009 [10]. The most popular languages taught in higher education are Spanish, French, German, and American Sign Language (ASL), in that order. ASL has the fastest growing enrollment among the four (16% compared to less than 5% between 2006 and 2009). Dictionaries that translate words from known to unknown languages and vice versa are important tools in learning a new language. For example, one might hear or read a word in the new language, but not know what it means. A dictionary provides the resource to find that meaning. Since ASL is a visual language with no standard written form, ASL-to-English dictionaries must take a different form than a text-based dictionary.

Current ASL-to-English resources attempt to help people search for unfamiliar signs, but fall short. For instance, online web search resources[1] accept text input and return videos or images of signs. Forming search queries for signs is not intuitive, as they require a text description or guess of the English meaning. Other tools allow people to naturally express a sign by demonstration, via computer vision or worn sensors. However, there are no guarantees that a user will replicate the sign accurately, or that the system will recognize the sign. Instead of demonstrating the sign, feature-based ASL-to-English dictionaries allow for people to use *features* of a sign (e.g. hand shape or location) to find the English meaning. Unfortunately, a person must use the features expected by existing dictionaries, or they may not find the translation. The limitations of each approach demonstrate a need for a more intuitive and flexible dictionary system.

In this paper, we propose ASL-Search and ASL-Flash, depicted in Figure 1. ASL-Search (Figure 1a) is a feature-based ASL-to-English dictionary powered entirely by its own users. Users form queries for unfamiliar signs by selecting sets of features (hand shapes, orientations, locations, or movements) based on their observations. The dictionary is backed by a search engine that uses information retrieval techniques adapted for ASL. Specifically, the search engine uses Latent Semantic Analysis (LSA), which traditionally models relationships between words and documents, to model relationships between features and signs. ASL-Flash (Figure 1b) is a learning tool for ASL students that provides an additional source of user queries for ASL-Search. ASL-Flash shows visitors online flashcards of ASL signs. Before providing the user with the definition, ASL-Flash requests that the user acts as if he/she is searching for the sign using the ASL-Search interface. ASL-Flash utilizes an established corpus of online videos from SigningSavvy, an English-to-ASL online dictionary. We built and deployed ASL-Flash, and used the query data gathered with ASL-Flash to provide a proof of concept of ASL-Search.

[1] e.g. Google, Bing, YouTube, and online ASL video resources like SigningSavvy

ACM 978-1-4503-2922-4/15/03…$15.00.
http://dx.doi.org/10.1145/2675133.2675226

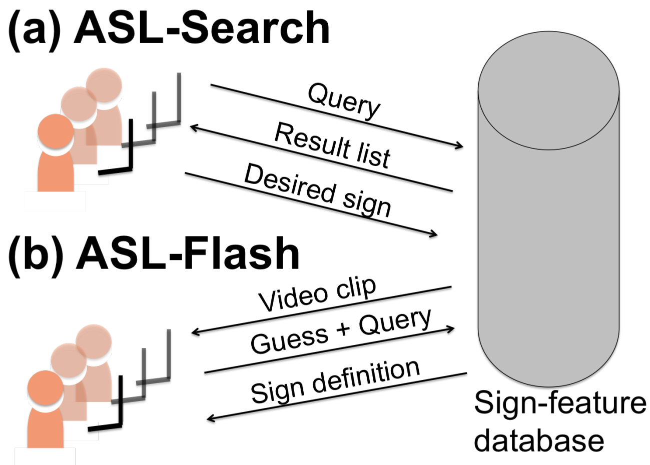

Figure 1: (a) ASL-Search: A feature-based dictionary that stores its users' queries in a database, and learns from that data to improve results. (b) ASL-Flash: An online learning tool that also contributes queries to the ASL-Search database. Users of both ASL-Search and ASL-Flash use the same query input interface, and contribute to the same database.

The core contributions of this work are:

- A survey of the methods that ASL students currently use to look up signs, and the difficulties they encounter.
- The design of ASL-Search, a user-powered ASL-to-English dictionary built by its users' queries.
- ASL-Flash, a learning tool for ASL students that serves as a source for user query data.
- ASL-Search proof of concept with data gathered from ASL-Flash.

BACKGROUND AND RELATED WORK

Computer-based sign language recognition, translation, and lookup systems accept two types of input: 1) a physical demonstration of the sign, or 2) a set of features describing the sign. Systems that require physical demonstrations often require expensive hardware, and rely on gesture recognition algorithms that cannot yet translate signs accurately. Accepting features as input simplifies the translation problem and gives users flexibility, but existing feature-based systems still do a poor job of matching input features to signs. Below, we explore existing sign language search methods further, describe their limitations, and how we improve upon their work by leveraging the crowd of users.

Search by Example

Some sign language recognition systems allow the user to search for a sign by demonstrating the motion of the sign. Video, the Microsoft Kinect, and sensory gloves have been used to record sign motions. These "search by example" systems rely on computer vision and gesture recognition to identify the sign.

Video Input

Recognizing signs from video recordings is difficult [7, 13, 26, 32, 38, 40]. Boston University [38] and Cooper et al. [7] have built systems that accept video as input and return lists of similar sign clips. Both perform well when trained and tested on the same users, but make no performance guarantees when trained and tested on different users. Other systems (e.g. [26]) attempt to recognize signs made by diverse users. While search results are accurate for small corpuses of signs, there are no guarantees about scalability. To recognize nuanced gestures, video-based systems often track pixel color (e.g. [13]), using Hidden Markov Models (e.g. [32]) or Time Delay Neural Networks (e.g. [40]). Researchers are actively working to improve these methods due to a conflict in visual perception; while humans perceive high-level features [30], features that computational methods extract from images and videos are typically low-level. Unlike these systems, ASL-Search accepts feature descriptions of signs from users and learns from these descriptions, making it both highly scalable and allowing for diverse search queries.

Kinect Input

More recently, the Kinect has been explored for sign language recognition [9, 41]. The Kinect supports Skeletal Tracking, and can be used to track 20 joints in 3 dimensions[2]. As in gesture recognition for video, HMMs are commonly used to model these skeletal gestures over time (e.g. [41]). While the Kinect provides a rich data source, accurate sign language translation for diverse users with a complete sign language vocabulary is still an open problem. Recent attempts have tackled the problem of generalizing to diverse users by extracting universally relevant features from the Kinect data (e.g. [9]). While the Kinect collects more data than video alone, processing that data is computationally expensive, and system performance still depends on the accuracy of the user's sign demonstration. Our system removes these difficulties by allowing users to choose which features they use to describe signs, both reducing the complexity of and increasing confidence in the data provided.

Glove and Sensor Input

Gloves and other arm and hand sensors have also been used to input signs to a computer [20, 39, 25, 21]. A rich literature on the use of gloves and other sensors to detect movements exists beyond the scope of this related work section. These systems support detection of nuanced hand movements and positions that can be meaningful to signs. For example, Kim et al. studied the use of sensory gloves for depth detection [20]. In addition to sensing finger movement directly, research efforts have studied how to track the movement of fingers through video capture of a glove with differently colored fingers and areas [39]. Unfortunately, these systems require specialized hardware that can be expensive and uncomfortable. ASL-Search removes the need for additional hardware by allowing users to interact directly with a website.

Search by Feature Selection

Feature-based ASL-to-English dictionaries avoid the computational problems of "search by example" by allowing users

[2] http://www.xbox.com/en-US/kinect

to input features describing a sign. The features used by these systems are supported by linguistic analysis of ASL.

Features Grounded in Linguistics

Linguists have developed several ASL notation systems. They are not widely used for communication between people; it is easier to communicate in English or to sign [5, 31]. Stokoe notation is the best known in the Deaf community, as it propelled ASL to be recognized as a true language [33]. It defines a grammar based on five feature types: hand shape, location, orientation, movement, and relative position.

Printed ASL-to-English dictionaries use features from these notation systems to organize signs. Stokoe published his seminal notation with an ASL-to-English Dictionary, which sorts signs first by location followed by hand shape [33]. The Gallaudet ASL Handshape Dictionary also sorts signs by hand shape [34]. When a hand shape can be identified, it narrows the set of possible English translations. However, users may need to sift through a lengthy list of possibilities. Our dictionary builds on this prior linguistic work by using features taken directly from Stokoe notation.

Feature-Based Electronic Dictionaries

Electronic feature-based ASL-to-English dictionaries allow the user to select features to describe a sign (e.g. Handspeak[3], Jinkle[4], and SLinto[5]). For example, one commercial resource, "The Ultimate ASL to English Dictionary," uses hand shape, location, orientation, and movement for search [36]. Possible matching English words are returned in an alphabetized list, which is undesirable because the target word may be late in the alphabetical ordering. Existing feature-based ASL-to-English dictionaries have several limitations:

1. Poor matching of features to signs.[6]
2. Requirements on features that the user must select.[7]
3. Lack of support for feature omissions.[8]
4. Cumbersome search interfaces.[9]

The design of ASL-Search seeks to address each of these issues with a front front end that gives the user freedom in feature selection, and a backend that learns from user queries. The search engine that drives ASL-Search stores past user queries in a feature-sign matrix, and uses LSA to identify meaningful dimensions in feature space before matching new queries to signs.

[3]http://www.handspeak.com

[4]http://asl.jinkle.com/lsearch.php

[5]http://slinto.com/us/

[6]These dictionaries do not seem to be fully functional, and do not publish the algorithms used for returning search results. Based on the authors' experience using these systems, it is likely that they find matching signs by executing a strict database search.

[7]Handspeak requires hand shape, movement, and location for *only* one unspecified hand; Jinkle requires the starting hand shape, orientation, location, and movement for *only* the dominant hand; and SLinto requires *only* hand shape and location for both hands.

[8]SLinto is the only tool to allow for missing features.

[9]For example, Handspeak and Jinkle afford feature selection through drop-down lists of English words, which are poor descriptors of visual cues.

Crowdsourcing for Language and Search Tools

Crowdsourcing has been explored as a resource for language analysis (e.g. [11, 23, 27, 28]). For example, lexical applications include translating text between languages (e.g. [28]) and disambiguating word meanings (e.g. [27]). Crowdsourcing has also been explored as a means for real-time captioning of spoken words into written form [23] and collaboratively translating rich literature like Shakespeare's plays [11]. However, to the best of our knowledge, crowdsourcing has not been used to gather data in support of a sign language dictionary.

Crowdsourcing has also been used to build Duolingo [37], a program that teaches written languages through a progression of written and spoken exercises. Instead of paying language experts to translate the phrases used in the exercises, a crowd of volunteers translates them. Duolingo also looks for patterns in the translations that its users provide, allowing it to build language models. Like Duolingo, our ASL-Search dictionary learns from its users' queries to form a model of language. Our ASL-Flash "game" helps crowdsource the dictionary, simultaneously providing value to users by reinforcing vocabulary, and enriching the set of queries used to train the ASL-Search dictionary. Unlike Duolingo, ASL-Flash and ASL-Search are built specifically for signed languages, provide a dictionary, and do not require language competency of any contributors.

Search tools have also attempted to use crowdsourcing to improve search results (e.g. [6, 19, 24, 29, 35]). As the data sources for information retrieval systems become increasingly large and diverse, it becomes more difficult to use an expert to evaluate the relevance of these resources [17]. Fortunately, crowdsourcing offers a flexible, scalable resource for relevance evaluation. Workers from scalable platforms like Amazon's Mechanical Turk have been used to evaluate relevance of documents and other search results (e.g. [3, 12, 18, 24]). In search engines, *social search* uses explicit feedback from previous users to inform search results for subsequent users (e.g. [2]). Implicit user feedback, like click data, has also been leveraged to refine results (e.g. [1, 15, 16, 29]). Users can also contribute content including tags, reviews, and comments to improve search results for web search (e.g. [6]), library systems (e.g. [35]), and book searches (e.g. [19]). The use of crowdsourcing to improve search results in other domains suggests the potential of users to contribute to sign language dictionary search methods, as we do with ASL-Search and ASL-Flash.

SURVEY OF SEARCH METHODS

In order to motivate our system, we conducted a survey with current ASL students to determine methods used to search for unfamiliar signs. The survey used multiple choice questions to ask about the frequency of use for the resources in Figure 2, and free-form responses to gather more information on the students' lookup processes. We recruited 28 participants, 3 male and 25 female. The average (mean) age was 21, with a range of 18-41. All participants were either learning or already knew ASL, with a mean of 1.46 years of ASL experience. Two participants had experience with one addi-

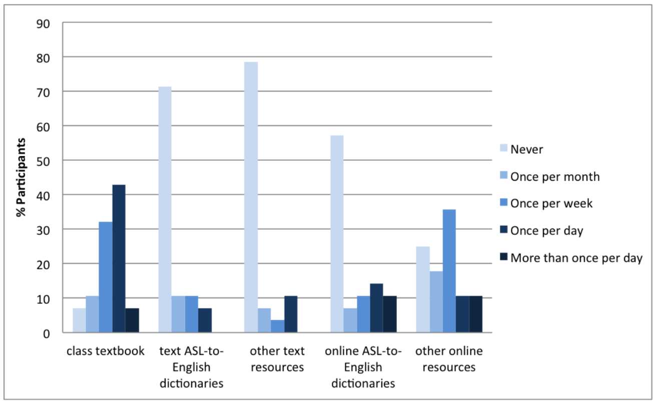

Figure 2: Use of existing sign search methods by ASL students.

tional sign language. The mean number of spoken languages including English was 1.89 per participant.

Despite the existence of online ASL-to-English dictionary resources, students predominantly did *not* use them. Each participant provided frequency of use for: 1) class textbook, 2) text ASL-to-English dictionaries, 3) other text resources, 4) online ASL-to-English dictionaries, and 5) other online resources. Examples of online resources that are not ASL-to-English dictionaries include YouTube videos, online English-to-ASL dictionaries, and search engines like Google or Bing. We found that the class textbook (92.86%) and non-dictionary online resources (75.00%) were the only resources used by the majority of students (see Figure 2). However, only 42.86% of the surveyed students used online ASL-to-English dictionaries. This disparity suggests that the internet is an appealing place for students to search for signs, but ASL-to-English dictionary performance or discoverability is not yet competitive with other online resources. Because most students use the internet to search for unfamiliar signs, it is possible that improved online dictionaries will provide benefit to those learning ASL.

Another valuable resource for inquiring about unfamiliar signs is to ask another person, but only 17 of the 28 participants used this method regularly (60.71%). One reason that fewer students ask others[10] is the lack of availability to do so. For example, it might be inconvenient or embarrassing to stop a conversation and ask, particularly in a social setting. In addition, there are other scenarios when it is never possible, like when viewing a video alone. The other 11 students (39.29%) who did not ask others relied on text[11], online resources[12], and guessing the English representation.

Text and online resources as they exist today are not a viable solution to the problem of looking up signs for many students. In fact, 8 (28.57%) students said if they saw an unfamiliar sign, and had no one available to ask, they had no idea what actions to take. One possible reason is the difficulty of expressing a sign in text. In the words of one participant, *"describing the hand motions for a particular sign in words isn't always the easiest thing to do."* Another possible reason is that students must guess the meaning of the unknown sign using context in order to use English-to-ASL dictionaries, many text resources, or online videos. Other issues emerge when using ASL-to-English dictionaries; in particular, one participant alluded to inconsistent performance, explaining, *"I compare different dictionaries to make sure that I understand how the sign is formed or possible variations on the sign."* These responses strongly suggest that there is a need and opportunity to improve upon text and online resources when an ASL signer is not available.

In the face of these difficulties while searching for signs in online and text resources, 13 participants (46.43%) forego them altogether. If nobody is around to ask, 2 students (7.14%) ask somebody later and repeat the sign from memory. However, this task may be difficult because people may forget the need to ask, let alone the sign. 2 other students (7.14%) admitted that they were content with ignorance. *"I'd probably be content with just not knowing what the sign is,"* admitted one, while the other elaborated, *"It's such a visual language that trying to 'look' up how to sign a word can be more confusing than not having known it ever existed."* Several participants described a lineup of resources that they use. One participant explained, *"I first check my class textbook in the section where they go over vocabulary for the chapter. If it's not there I ask my twin brother who took the ASL series two years ago to see if he knows. If none of those things work I wait till it comes up in class to try and gain some more context for the sign."* The students' use of lists of resources highlights the difficultly and unreliability of searching for signs.

Implications from survey

Our survey demonstrated that current ASL learners not only struggle to search for the meanings of ASL signs, but stop their efforts altogether due to a lack of resources or motivation. Text and online resources are not conducive to inputting signs, and current ASL-to-English dictionaries are not complete or strong in performance. Therefore, our survey suggests that a usable, reliable, online ASL-to-English dictionary may remove an educational barrier: difficulty of searching for signs independently.

ASL-SEARCH

We propose ASL-Search, an ASL-to-English dictionary that is entirely powered by its users. The dictionary allows users to search for a sign by selecting a set of features that describe the sign. The system stores queries from previous users in a matrix of feature frequencies for each sign. As each user enters a query into ASL-Search, the system learns and improves the strength of its results using Latent Semantic Analysis (LSA). While the system's search interface allows an ASL sign to be described with many features spanning hand shape, orientation, location, and movement, LSA reduces the number of features needed (or dimensionality) when comparing a query to our database. By reducing the dimensionality

[10] in comparison to using a textbook or online resource that is not a dictionary
[11] class textbook, and other ASL books
[12] Google, Bing, YouTube, and AslPro [4]

of the feature space, LSA reduces noise in data while leveraging trends in the previously entered queries. For a feature-based sign language dictionary, there are several sources of variability in queries: 1) difficulty to create a comprehensive, unambiguous, and intuitive set of features that fully describe a signed language, 2) natural variability between different signers, 3) differences in perceptions between viewers, and 4) distorted memory of a sign. Instead of suffering from these sources of confusion, ASL-Search uses LSA to identify patterns in query variability and improve search results.

Search Interface

In order to design our search interface, we used the five feature types identified by Stokoe [33]. To be comprehensive, we used the hand shapes from the American Sign Language Handshape Dictionary (ASLHD) [34], while the rest of the features come directly from Stokoe notation. The features we chose are well grounded in linguistics, but further refinement of the features is an area for research outside the scope of this paper. The ASL-Search backend also compensates for imperfect design of the feature set by performing dimension reduction on the feature space. Below, we present each feature type, and the means to input features in a search query.

1. *Hand shape* (40 total): **What is the configuration of the hand and fingers?** Hand shapes are selected by clicking on pictures of the hand shapes. The pictures are of a fluent signer and organized in morphologic order, as in ASLHD.
2. *Location* (10 total): **Where relative to the body is the hand located?** The locations are selected by clicking on discrete regions of a picture of a person's torso and head.
3. *Orientation* (10 total): **Which direction is the palm facing? Are the fingers pointing up or down?** The orientations are presented by a series of pictures of a hand facing in the appropriate directions.
4. *Movement* (22 total): **What is the change in position over the sign duration?** The movements are presented by ball-and-arrow diagrams of the hand movement. We also provide a video of a person making the movement on demand.
5. *Relative position* (7 Total): **If there are two hands, where are they located with respect to one another?** The relative positions are presented by images showing the relation between the two hands.

Our user interface allows for the selection of features to describe and search for a sign (as in Figure 3). We allow users to omit feature types, or to select multiple features within a single type. By giving the user freedom, the ASL-Search interface allows the user to describe the sign as he or she remembers. Next, we discuss how our backend uses LSA to compensate for the variability in search queries.

Backend

Latent Semantic Analysis (LSA) is traditionally used in Natural Language Processing to analyze the similarity between documents and words based on word counts [8]. In this domain, the data is represented as a matrix X. Each row represents a document, and each column represents a word. Item

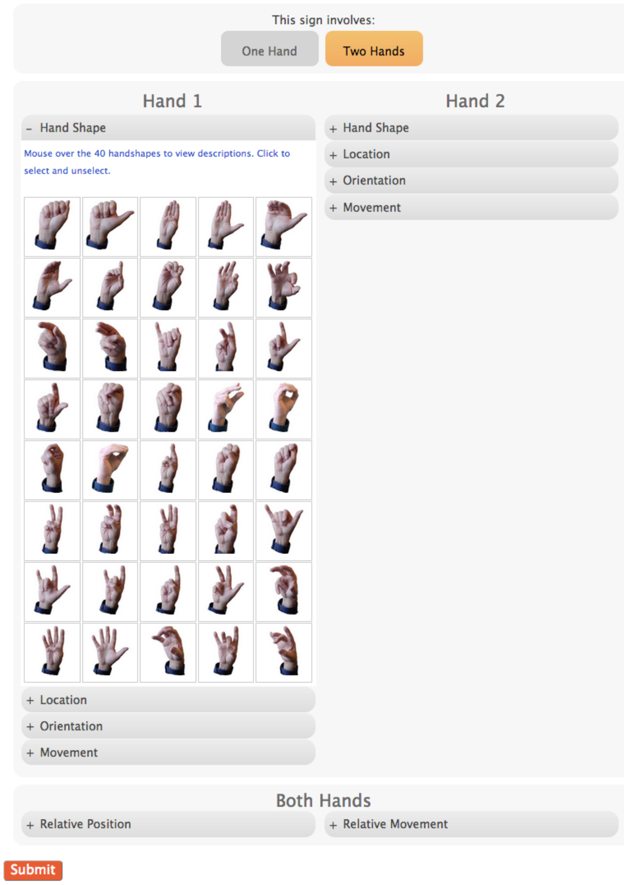

Figure 3: Screen shot of feature input interface as a user enters the hand shape for a two-handed sign. This interface is part of the ASL-Search design, and was deployed in ASL-Flash.

(i, j) thus contains the frequency of word j in document i. Instead of documents and words, our ASL dictionary has signs and features, as demonstrated in Figure 4. Each row of X represents a sign, and each column represents a feature. A query is represented as a vector of 0's and 1's, where a 1 indicates that a particular feature was entered as part of that query.

LSA utilizes Singular Value Decomposition (SVD) to factorize $X = U\Sigma V^T$, where U and V are orthogonal matrices, and Σ is a diagonal matrix. More specifically, U and V are matrices whose rows comprise the eigenvectors of XX^T and X^TX, respectively, while Σ contains the eigenvalues of XX^T (and also of X^TX). We can then select the k largest eigenvalues and corresponding eigenvectors to yield a rank-k approximation for X, $X_k = U_k\Sigma_k V_k^T$. This is an optimal approximation for X,[13] and can be used to map each feature (word) or sign (document) to an item in k-dimensional space.

To compute the similarity between an incoming query q and each sign in our database, we first project both q and X onto our lower-dimensional space as shown in Figure 5, yielding

[13]optimal by the Frobenius norm

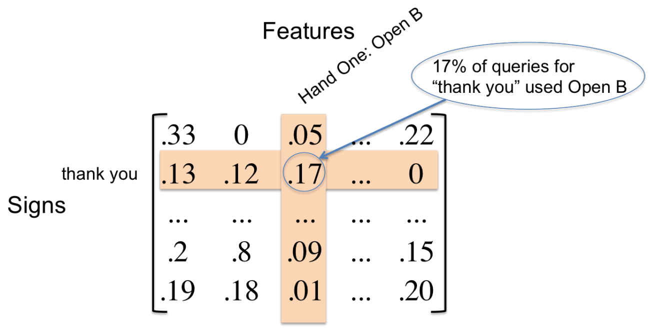

Figure 4: Matrix X of feature frequencies from user queries.

$q' = \Sigma_k^{-1} V_k^T q$ and $X' = \Sigma_k^{-1} V_k^T X$. We have now represented the query as well as each sign in our database as a vector in k-dimensional space. We want to identify the signs in our database that are closest to the query vector, so we take the angle between the incoming query and each sign vector in the space of reduced dimensionality. More specifically, we return the signs in our database sorted by cosine similarity with the incoming query, computed as $\frac{x' \cdot q'}{||x'||||q'||}$, where x' is a projected sign from our database and q' is the projected query.

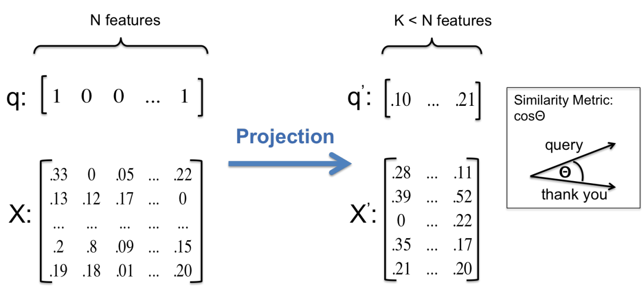

Figure 5: Dimension reduction from the original feature space to a lower-dimensional feature space.

We tailored the classic LSA algorithm for ASL-Search to account for ambiguity between hands. When entering a two-handed sign, each user determines which hand to input as "Hand 1" and which as "Hand 2" (as in Figure 3). Because people can make two choices, we replicate all entered data by switching "Hand 1" and "Hand 2." We store the flipped queries for each sign in a new row of the data matrix. A sign is a match for an incoming query when either the original or flipped row is a match. The intuition is that the flipped row more accurately represents the sign for users whose mental model of the two hands is opposite that of most users. By adding the flipped row, we can more accurately match queries for users who make the less popular hand choice.

ASL-FLASH

We also present ASL-Flash, a learning tool that teaches students ASL and contributes queries to the ASL-Search database. ASL-Flash presents a series of ASL "flashcards" to the user. The sign clips are taken with permission from SigningSavvy [14]. Each clip shows a single sign. Once the user views the clip, ASL-Flash presents a multiple choice question on the English meaning. Next, the person is instructed to use ASL-Search's interface as if they were searching for the sign. Finally, ASL-Flash provides the English meaning of the sign accompanied by the original sign video.

Query data collected by ASL-Flash has several uses: 1) it will provide seed data for ASL-Search when the dictionary is deployed, 2) it will provide ASL-Search with query data for new signs added to the dictionary and existing signs that are looked up infrequently, and 3) it can be used to demonstrate the viability of the ASL-Search design, as we do in this paper.

ASL-Flash Deployment

We used ASL-Flash to gather query data to verify ASL-Search. Because ASL-Search is of use to people learning ASL, we used signs from a textbook used in first-year ASL curricula [42]. The textbook contains 1101 signs, and we selected a random sample of 100 signs to use in our deployment. Each ASL-Flash user viewed 10 of these 100 signs, randomly selected[15] and ordered. Users were given the option to quit early, so some respondents completed fewer than 10. Overall, we collected 670 viable queries from 94 users: 52 (55.32%) female, 41 (43.62%) male, and 1 (1.06%) other. The mean age was 28.4 years, with a range of 14 to 67 years. Users had a wide range of ASL experience (9 with 0 years; 14 with 0-0.5 years; 18 with 0.5-1 years; 14 with 1-2 years; 6 with 2-3 years; 32 with over 3 years; 1 chose not to respond). The mean number of signed languages known was 1.7 with a standard deviation of 3.7. The mean number of spoken languages was 1.9 with a standard deviation of 1.4.

ASL-Flash is a sustainable source of query data for ASL-Search. The sustainability of ASL-Flash is important because we will use ASL-Flash to seed ASL-Search upon release, and to gather data for new or rare signs. 79.72% of users said they would use the system again (76.92% of those who answered all flash cards correctly, and 81.25% with at least one wrong answer). Regardless of the user's accuracy in identifying signs, the tool provides value in learning or reviewing signs.

A strong motivation for people to continue using ASL-Flash (and providing more data for ASL-Search) are the learning benefits. Out of users who incorrectly guessed at least one sign, 87.50% reported learning something new from ASL-Flash. In addition, those same users showed more activity viewing the sign videos that accompany the sign definitions on the answer pages. The count of video views was not normally distributed for those who guessed all signs correctly ($W = 0.4862$, p-value < 0.001) or their counterparts ($W = 0.7626$, p-value < 0.001). Users who made mistakes viewed videos significantly more times than those who guessed all of the signs correctly ($W = 158802$, p-value < 0.001, Mann-Whitney). This suggests the dual benefit of ASL-Flash users learning ASL, while providing valuable data for ASL-Search.

[14] http://www.signingsavvy.com
[15] from the least-viewed signs

During deployment, we asked users about their experience with the feature entry interface, which is identical to the interface proposed for ASL-Search. We found that our feature-based interface felt natural to most users. When asked the yes/no question: *"Did this survey give you a natural way to describe signs to look them up?"*, 75.68% said that the input mechanism was natural to use. Out of users who correctly guessed every sign, 73.08% found the interface intuitive; out of users who incorrectly guessed at least one sign, 77.08% agreed. This positive feedback suggests that the ASL-Search search interface will be suitable for looking up signs.

Feature Data from Experts

In addition to gathering features with ASL-Flash, we asked two experienced signers to provide features. They evaluated the same 100 signs used by ASL-Flash. Unlike in the ASL-Flash condition, we allowed the experts to replay the videos and complete the task in as much time as was needed. Their feature inputs were used to form baseline comparison search methods for ASL-Search.

The experts did not completely agree on the features present in the 100 signs. Though they agreed on the number of hands used for each sign, they entered slightly different feature sets for each of the 100 signs. On average (mean), they disagreed on 7.21 out of 164 features (4.40%). The lack of agreement between experts confirms the ambiguity inherent to sign executions and viewer perceptions. This ambiguity supports our choice of LSA for ASL-Search's backend, which extracts meaningful feature dimensions through dimension reduction.

The disagreement between experts also highlights the necessity of collaborative work in building a sign language dictionary. Even if the dictionary backend is constructed by experts (as in previous work), it would be inappropriate for a single expert to evaluate all signs; rather, a group of experts would be required. The tasks required of them would be time-consuming and tedious, and difficult to scale. Instead of dealing with these problems surrounding the collection and synthesis of expert input, our system supports the collaboration required to build an ASL dictionary effortlessly. The dictionary is built as it is used, by the users themselves.

PROOF OF CONCEPT FOR ASL-SEARCH ALGORITHM

We used the data gathered by ASL-Flash to form the database backend of ASL-Search and simulate its performance looking up signs. We formed baseline search methods that mimic existing ASL-to-English dictionaries using our expert feature evaluations. Our comparisons find that ASL-Search outperforms these baselines.

Metric

Discounted Cumulative Gain (DCG) is a standard metric used to evaluate the performance of search engines [14]. Let rel_i be the relevance of the i-th result. The score for a list of p ordered results is computed as $DCG_p = \sum_{i=1}^{p} \frac{2^{rel_i}-1}{log_2(i+1)}$. For our problem, we are searching for one particular sign, so the relevance of each sign in the result list is binary $rel_i \in \{0, 1\}$. Consequently, the DCG reduces to $\frac{1}{log_2(i+1)}$, where i is the placement of the desired sign in the result list.

Normalized Discounted Cumulative Gain (NDCG) normalizes DCG across queries, since the range of DCG scores that are possible varies for different queries. To do this, it divides the DCG for a given query by the maximal possible DCG for that query if results were returned in the optimal order. This means that NDCG will take a value in $[0, 1]$ for all queries. For our problem, because relevance is binary and $rel_i = 1$ for exactly one sign i in the returned list, DCG and NDCG are equivalent.

Overall Performance

To validate the design of ASL-Search, we simulated its use with the query data gathered from ASL-Flash, as demonstrated in Figure 6. In the Testing Phase, we use leave-one-out cross-validation. The single held-out test query represents an incoming user query for a sign. We generate the sorted list of results that ASL-Search would return, using the rest of the data as its database of previous queries, and evaluate the quality of those results using DCG. In the Training Phase, we simulate ASL-Search learning the dimensionality to be used for reduction in the LSA algorithm. We run 10-fold cross-validation, computing a list of results for each query in the held-out part of the database, and average the DCG scores for each dimensionality. The dimensionality $K_{optimal}$ with the highest average DCG is chosen and used to generate results for the incoming test query.

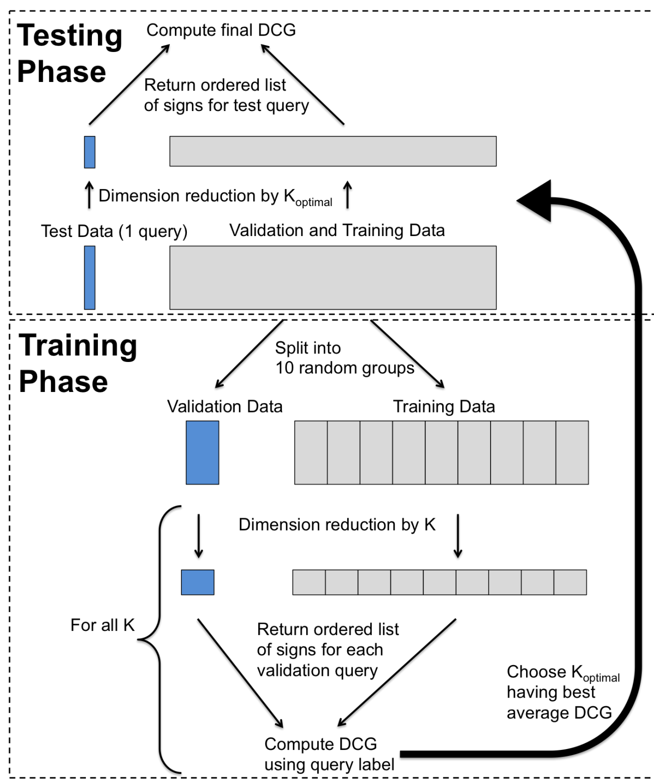

Figure 6: Simulated use and analysis of ASL-Search. The Training Phase replicates ASL-Search learning the dimensionality $K_{optimal}$ for reduction. The Testing Phase produces and evaluates the ordered list of signs that ASL-Search would return in response to incoming queries.

Because our online dictionary returns a sorted list of results, we can identify the position of the desired result in that list. Figure 7 provides a histogram of those placements for the Test Phase. 59.10% of the time, the desired result was the first result, and 84.93% of the time, it was in the top 10.

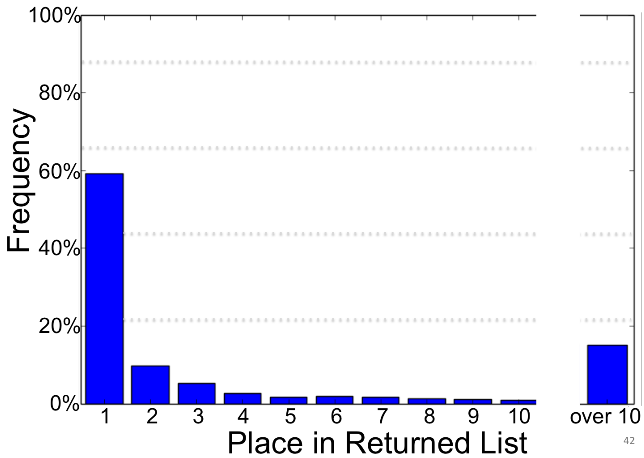

Figure 7: Place of desired result in the sorted result list. The "over 10" bucket summarizes the long tail of the distribution.

Tuning Dimensionality K

Choice of dimensionality to which the original feature space is reduced greatly impacts ASL-Search's performance, as demonstrated in Figure 8. Reducing to too few dimensions detracts from performance, as we lose meaningful information. Conversely, not reducing enough hurts performance, as we eliminate too little noise from the data. Because the choice of dimensionality impacts the quality of search results, ASL-Search uses cross-validation on the database of queries to learn the "best" choice of dimensionality for its users.

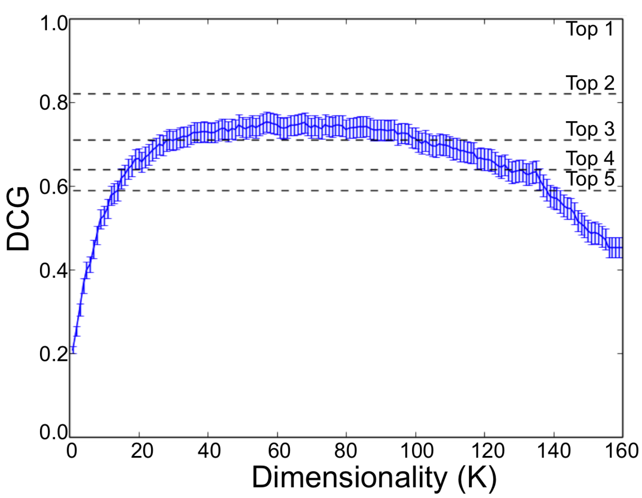

Figure 8: Effect of dimensionality K on performance, demonstrated by leave-one-out cross-validation on our entire dataset. The dotted lines show DCG when the desired sign is in the top 1, 2, 3, 4, or 5 results, with equal probability.

The dimensionality learned by the Training Phase was relatively stable throughout our simulation. Figure 9 shows the distribution of the chosen dimensionality. The mean optimal dimensionality found was 69.26. The feature dimensions produced by the dimension reduction are difficult to interpret intuitively. Each resulting dimension was a linear combination of all the original features.These results are characteristic of LSA, which is known to create dimensions that are difficult to interpret [22].

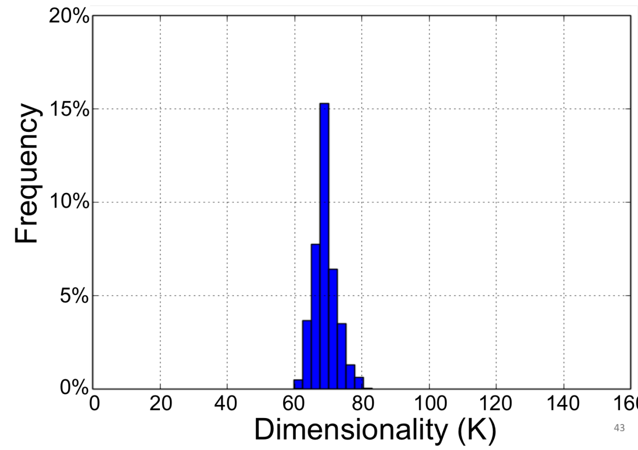

Figure 9: Dimensionality learned by ASL-Search ($K_{optimal}$ in Figure 6) from the database of past queries.

Baseline Methods

We compared ASL-Search to three baseline methods that return an ordered list of signs in response to a feature-based query. Because existing online ASL-to-English dictionaries do not seem to be fully functional and are not guaranteed to cover a vocabulary that matches the signs used in ASL-Flash, we generated our own baseline methods. These baselines are explained below:

- *Random*: Returns all signs in the dictionary in a completely random order.
- *Expert Or*: Compares the incoming query to the union (logical "or") of our expert feature vectors. It returns all signs for which the incoming features are a subset of the expert "or" vector, sorted by the number of matching features. We suspect that existing feature-based ASL-to-English electronic dictionaries use similar methods, since their results vary in length, and sometimes return no results at all. We chose the union of expert features, rather than their intersection (logical "and") because the union allows for more successful feature matches and better performance.
- *Expert Hamming*: Computes the hamming distance between the incoming query and the union (logical "or") of our expert feature vectors for each sign in the dictionary. It returns all signs in the dictionary, ordered by increasing Hamming distance so that the closest signs are returned first. *Expert Hamming* serves as a more robust alternative to *Expert Or*.

Performance as the ASL-Search Database Grows

The simulated longitudinal performance of our dictionary demonstrates that ASL-Search significantly outperforms existing baselines. ASL-Search improves as more users contribute data, suggesting that ASL-Search will further outpace other methods in the future.

We simulated the performance of our dictionary over time using the query data gathered by ASL-Flash. We repeated the following two steps 20 times: 1) We held out a set of 100 test queries, one chosen randomly for each of our 100 signs. 2) We simulated the growth of the database by incrementally adding queries to the database, and evaluated performance on the test queries. Specifically, we divided the remaining query data into 10 equal sized and random groups, and added each group to the database to simulate database growth. Performance after each addition was evaluated by computing the DCG for the results generated for the 100 test queries. Figure 10 displays the average DCG from the 20 trials, with error bars of the standard error over the 20 trials.

Perhaps surprisingly, *Random* and *Expert Or* had almost the same average performance, with *Expert Or* outperforming *Random* by less than 0.001. The average performance of *Expert Or* is comparable to *Random* because it fails to return the desired sign for some queries. *Expert Or* only returns signs that the experts have determined to have all features selected in the query. Because there is variation in the features that viewers perceive, some queries contain features that the experts considered absent from the desired sign. In these cases, *Expert Or* does not return the desired sign (and in fact might not return any results at all), and receives a DCG score of 0. For these queries, *Random* outperforms *Expert Or* since it always returns the desired sign at some position in the result list. Conversely, for queries whose features are considered to be present in the desired sign by the experts, *Expert Or* typically outperforms *Random*. While *Expert Or* is a weak baseline, it is representative of existing ASL-to-English dictionaries.

The *Expert Hamming* baseline better leverages the valuable signal in the expert data, and consequently outperforms *Expert Or*. Unlike *Expert Or*, *Expert Hamming* does not restrict the signs returned to those that possess all features present in the query. Instead, it uses Hamming distance to evaluate similarity between the incoming query and every expert evaluation of a sign in the dictionary. Using Hamming distance produces more nuanced comparisons, and allows *Expert Hamming* to always return the desired sign at some rank in the result list.

Overall, our system returned more accurate results than the *Random*, *Expert Or*, and *Expert Hamming* baseline methods. The crowd provides data that experts do not: the features commonly seen by real users who are unfamiliar with the signs they look up. ASL-Search's use of dimension reduction through LSA allows ASL-Search to leverage this signal. Furthermore, our system improves with additional data from users, which is not possible for *Random*, *Expert Or*, *Expert Hamming*, or variants of our baselines employed by existing online ASL-to-English dictionaries.

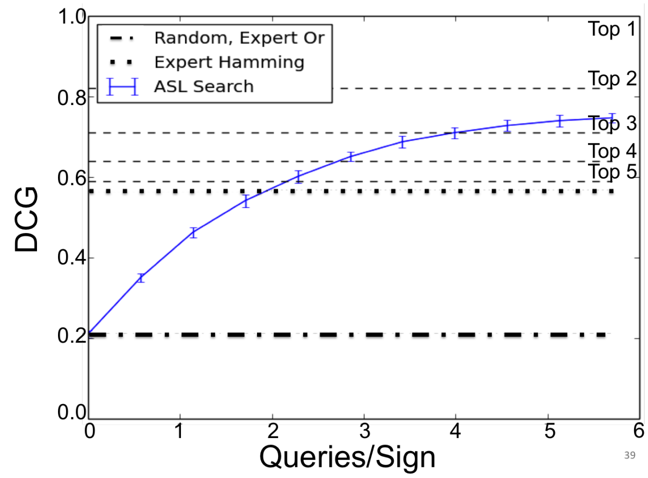

Figure 10: Performance of ASL-Search with use. Results were averaged over 20 simulations of dictionary growth.

Performance by ASL Experience

Our system performs well for target users with varying levels of ASL experience, as seen in Figure 11. To generate these results, we set the dimensionality to 70, which was the mean dimensionality chosen in the Training Phase, and held out one query at a time for testing. Our performance is relatively stable for target users of varying levels of ASL experience, but was slightly worse for those with absolutely no ASL experience. One possible reason for this lower performance is that brand new signers may have trouble identifying the visual queues in a sign. They may have suffered from fatigue as well, with decreased quality as they progressed through ASL-Flash's flash cards.

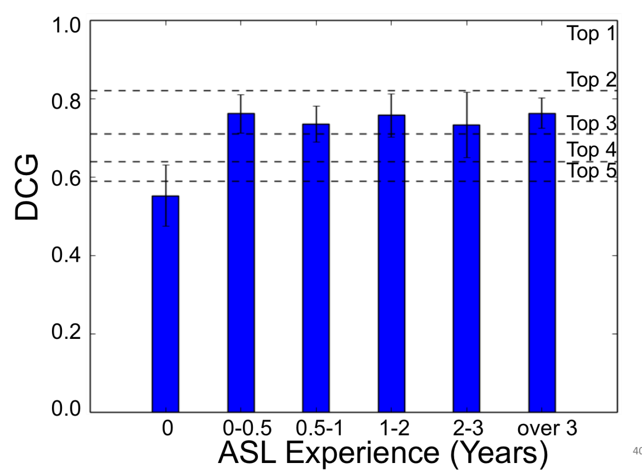

Figure 11: Performance for queries entered by users with varying ASL experience.

Our proposed system typically returned the most appropriate results when the English meaning of the sign was not known, as shown in Figure 12. For all levels of experience, except for users with 1-2 years of ASL, queries for unknown signs produced better results than those for known signs. Even for users with 1-2 years of ASL, the difference in performance is negligible. It is likely that LSA handles queries for unfamiliar signs better because these queries are characterized by feature variabilities that LSA expects and compensates for. Our results demonstrate that ASL-Search should fulfill its purpose of helping users to look up ASL signs, and especially unfamiliar ones.

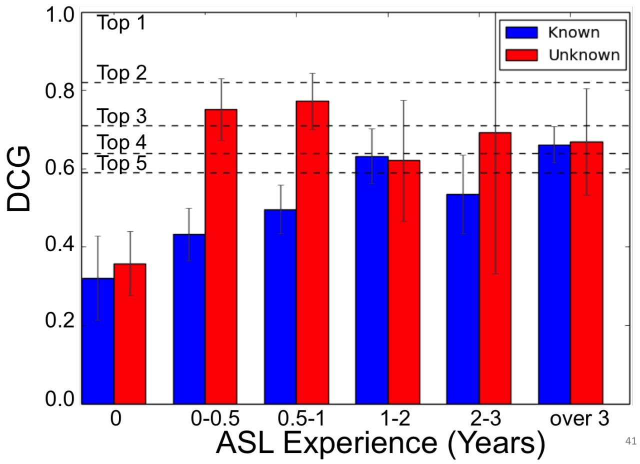

Figure 12: Performance for queries entered by target users with varying ASL experience, separated by whether or not the sign was known.

DISCUSSION AND FUTURE WORK

Our proof of concept for ASL-Search suggests that implementing our design will be effective. The survey we ran on existing lookup methods exposed difficulties that ASL students encounter when searching for a sign definition, and highlighted the need for a tool that allows users to easily and accurately look up ASL signs. Our proof of concept demonstrates that ASL-Search can accurately match its users' queries to signs and fill this need.

Furthermore, our proof of concept demonstrated that ASL-Search's performance will only improve with use. Even with a small amount of user data in our proof of concept, ASL-Search outperformed existing baseline methods. Unlike ASL-Search, these baseline methods do not improve with use, indicating that ASL-Search will further outpace them when deployed. Because ASL-Search learns from its users' queries, it has the ability to evolve with the language. For example, if the hand shape used for a particular sign changes over time, ASL-Search will detect that shift in its users' queries, and adapt its results accordingly. Because ASL-Search is built entirely upon input from the user community, its results are tailored to its users. Furthermore, because its users build the dictionary as they use it, crowdsourcing through ASL-Search happens effortlessly.

Our deployment of ASL-Flash with a small corpus of signs demonstrated that ASL-Flash is both a sustainable resource of supplementary data for ASL-Search, and a valuable learning tool in its own right. In our deployment, the majority of ASL-Flash users reported they would use ASL-Flash again, and that the flash cards helped them learn. Additionally, the majority of knowledgeable signers reported that they would use the tool again and that it helped them learn. We found that less knowledgeable signers devoted more attention to the answer videos. These results suggest that users of all experience levels can benefit from using ASL-Flash, and will likely continue to use it and provide data for ASL-Search.

ASL-Search can be adapted to solve problems besides sign language lookup, and serve communities of users in other domains. For example, a version of ASL-Search can be released to support bird watchers looking up the species of birds that they see. The bird can be described by a discrete set of features, like feather color and beak shape. Latent Semantic Analysis is well suited for this domain because the birds can be described by a set of features, and there will be natural variability in people's descriptions of the birds they see, just as there is natural variability in people's descriptions of signs. The ASL-Search design serves as a model for domains outside of sign language.

We are excited to explore a number of future directions. We plan to investigate improvements to the lookup algorithm. Alternative topic modeling methods could be used, like probabilistic latent semantic analysis (PLSA) or latent Dirichlet allocation (LDA), which would allow us to model the co-occurrence probability of queries and signs. Alternate feature-sign weights, like term frequency-inverse document frequency (TF-IDF), and other dimension reduction techniques, like principal component analysis (PCA) or group sparsity methods, could also be used. With or without dimensionality reduction, classification methods like decision trees, support vector machines (SVMs), and even multinomial logistic regression could also be used to determine the sign class for each incoming query, instead of LSA's cosine similarity. Unsupervised or semi-supervised methods could leverage unlabeled data, for example queries from ASL-Search where the user did not provide information about which sign they sought. Relevant methods can also be combined in various ways, for example by applying a sequence of operations, or weighting results from methods run in parallel. We also plan to investigate the scalability of these search methods as we expand the ASL-Search dictionary in preparation for release. We look forward to seeing the system deployed for public use.

Another interesting direction is to make the collaboration with other users more transparent. Our current ASL-Search design supports *implicit* cooperation between users. As the users enter their queries, they might be unaware of the fact that they are contributing to the success of the dictionary along with their fellow users. Making this collaboration more transparent may introduce future research questions to explore.

CONCLUSION

In this work, we present the design of ASL-Search, an ASL-to-English dictionary system entirely powered by its users. We motivated our system with a survey on methods used by ASL students to search for signs they do not understand. Our survey demonstrated the inadequacy of existing methods and the potential for our system to help mitigate this issue. ASL-Search's system design allows the user to search for a sign by entering visual features that describe the sign (such as hand shape and movement), and stores these queries in a database. Unlike existing systems, it novelly applies Latent Semantic Analysis (LSA) to match incoming queries to signs in the database, thereby mitigating noise in the data and accounting for observation variability.

We also built ASL-Flash, a learning tool that presents a series of ASL flashcards and simultaneously gathers additional query data for ASL-Search. We used ASL-Flash to gather query data from real users to demonstrate ASL-Search's performance in a proof of concept for the system design. The proof of concept demonstrates that ASL-Search outperforms comparable existing dictionaries, and will only improve over time with use.

This work supports learning ASL in two distinct but related ways: 1) through the design of ASL-Search, an electronic ASL-to-English dictionary that is both easy to use and accurate and 2) with the development of ASL-Flash, a set of online flash cards that help students learn ASL. The two systems work together to support the community of sign language learners. As ASL-Flash users reinforce and acquire ASL knowledge, they also contribute query data that improves the performance of the ASL-Search dictionary. ASL-Search has the potential to become a valuable resource for a large community, supporting sign language students and fluent signers alike.

When we look beyond ASL to other signed languages, the potential impact of ASL-Search and ASL-Flash grows. The difficulties encountered in learning ASL are not unique to ASL, but pervasive to all signed languages. Regardless of the language from which they come, signs are three-dimensional motions not easily described with written words, and there is natural variability in their execution and perception. ASL-Search can provide a dictionary resource for an international community of signers with releases in sign languages besides ASL and target languages other than English. Similarly, ASL-Flash can help students learn and reinforce sign meanings in other languages, and provide a sustainable datasource for the ASL-Search release in those languages.

ACKNOWLEDGMENTS

We thank SigningSavvy for granting us permission to use their sign videos in our deployment of ASL-Flash. This work is based in part upon work supported by the National Science Foundation under Grant No. 0915268. Any opinions, findings, and conclusions or recommendations expressed in this work are those of the authors and do not necessarily reflect the views of the National Science Foundation.

REFERENCES

1. Agichtein, E., Brill, E., and Dumais, S. Improving web search ranking by incorporating user behavior information. In *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, ACM (2006), 19–26.
2. Agrahri, A. K., Manickam, D. A. T., and Riedl, J. Can people collaborate to improve the relevance of search results? In *Proceedings of the 2008 ACM conference on Recommender systems*, ACM (2008), 283–286.
3. Alonso, O., Rose, D. E., and Stewart, B. Crowdsourcing for relevance evaluation. In *ACM SIGIR Forum*, vol. 42, ACM (2008), 9–15.
4. Aslpro, May 2014. `http://www.aslpro.com`.
5. Aslwrite - support american sign language through writing!, May 2013. `http://www.aslwrite.com/`.
6. Bao, S., Xue, G., Wu, X., Yu, Y., Fei, B., and Su, Z. Optimizing web search using social annotations. In *Proceedings of the 16th international conference on World Wide Web*, ACM (2007), 501–510.
7. Cooper, H., Pugeault, N., and Bowden, R. Reading the signs: A video based sign dictionary. In *Computer Vision Workshops (ICCV Workshops), 2011 IEEE International Conference on*, IEEE (2011), 914–919.
8. Deerwester, S. C., Dumais, S. T., Landauer, T. K., Furnas, G. W., and Harshman, R. A. Indexing by latent semantic analysis. *Journal of the Association for Information Science and Technology (JASIS) 41*, 6 (1990), 391–407.
9. Elliott, R., Cooper, H., Ong, E.-J., Glauert, J., Bowden, R., and Lefebvre-Albaret, F. Search-by-example in multilingual sign language databases. In *Procs. of Second International Workshop on Sign Language Translation and Avatar Technology (SLTAT)* (2011).
10. Furman, N., Goldberg, D., and Lusin, N. Enrollments in languages other than english in united states institutions of higher education, fall 2009. In *Modern Language Association*, ERIC (2010).
11. Giordano, T. *Toward a crowdsourced model for ASL translations of Shakespeare's works*. PhD thesis, California State University, 2013.
12. Grady, C., and Lease, M. Crowdsourcing document relevance assessment with mechanical turk. In *Proceedings of the NAACL HLT 2010 workshop on creating speech and language data with Amazon's mechanical turk*, Association for Computational Linguistics (2010), 172–179.
13. Imagawa, K., Lu, S., and Igi, S. Color-based hands tracking system for sign language recognition. In *Proceedings of the Third IEEE International Conference on Automatic Face and Gesture Recognition*, IEEE (1998), 462–467.
14. Järvelin, K., and Kekäläinen, J. Cumulated gain-based evaluation of ir techniques. *ACM Transactions on Information Systems (TOIS) 20*, 4 (2002), 422–446.
15. Joachims, T. Optimizing search engines using clickthrough data. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, ACM (2002), 133–142.

16. Joachims, T., and Radlinski, F. Search engines that learn from implicit feedback. *IEEE Computer 40*, 8 (2007), 34–40.
17. Kamps, J., Geva, S., Peters, C., Sakai, T., Trotman, A., and Voorhees, E. Report on the sigir 2009 workshop on the future of ir evaluation. In *ACM SIGIR Forum*, vol. 43, ACM (2009), 13–23.
18. Kazai, G. In search of quality in crowdsourcing for search engine evaluation. In *Advances in information retrieval*, vol. 6611 of *Lecture Notes in Computer Science*. Springer, 2011, 165–176.
19. Kazai, G., Kamps, J., Koolen, M., and Milic-Frayling, N. Crowdsourcing for book search evaluation: impact of hit design on comparative system ranking. In *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*, ACM (2011), 205–214.
20. Kim, J.-H., Thang, N. D., and Kim, T.-S. 3-d hand motion tracking and gesture recognition using a data glove. In *Industrial Electronics, 2009. ISIE 2009. IEEE International Symposium on*, IEEE (2009), 1013–1018.
21. Kim, J.-S., Jang, W., and Bien, Z. A dynamic gesture recognition system for the korean sign language (ksl). *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on 26*, 2 (1996), 354–359.
22. Landauer, T. K., Foltz, P. W., and Laham, D. An introduction to latent semantic analysis. *Discourse processes 25*, 2-3 (1998), 259–284.
23. Lasecki, W., Miller, C., Sadilek, A., Abumoussa, A., Borrello, D., Kushalnagar, R., and Bigham, J. Real-time captioning by groups of non-experts. In *Proceedings of the 25th annual ACM symposium on User interface software and technology*, ACM (2012), 23–34.
24. Le, J., Edmonds, A., Hester, V., and Biewald, L. Ensuring quality in crowdsourced search relevance evaluation: The effects of training question distribution. In *SIGIR 2010 workshop on crowdsourcing for search evaluation* (2010), 21–26.
25. Liang, R.-H., and Ouhyoung, M. A real-time continuous gesture recognition system for sign language. In *Automatic Face and Gesture Recognition, 1998. Proceedings. Third IEEE International Conference on*, IEEE (1998), 558–567.
26. Ong, E.-J., Cooper, H., Pugeault, N., and Bowden, R. Sign language recognition using sequential pattern trees. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, IEEE (2012), 2200–2207.
27. Parent, G., and Eskenazi, M. Clustering dictionary definitions using amazon mechanical turk. In *Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk*, Association for Computational Linguistics (2010), 21–29.
28. Post, M., Callison-Burch, C., and Osborne, M. Constructing parallel corpora for six indian languages via crowdsourcing. In *Proceedings of the Seventh Workshop on Statistical Machine Translation*, Association for Computational Linguistics (2012), 401–409.
29. Radlinski, F., and Joachims, T. Query chains: learning to rank from implicit feedback. In *Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining*, ACM (2005), 239–248.
30. Rui, Y., Huang, T. S., and Chang, S.-F. Image retrieval: Current techniques, promising directions, and open issues. *Journal of visual communication and image representation 10*, 1 (1999), 39–62.
31. Signwriting for sign languages, May 2013. `http://www.signwriting.org/`.
32. Starner, T., and Pentland, A. Real-time american sign language recognition from video using hidden markov models. In *Motion-Based Recognition*. Springer, 1997, 227–243.
33. Stokoe, W. C., Casterline, D. C., and Croneberg, C. G. *A dictionary of American Sign Language on linguistic principles*. Linstok Press Silver Spring, 1976.
34. Tennant, R. A., and Brown, M. G. *The American sign language handshape dictionary*. Gallaudet University Press, 1998.
35. Trainor, C. Open source, crowd source: harnessing the power of the people behind our libraries. *Program: electronic library and information systems 43*, 3 (2009), 288–298.
36. The ultimate american sign language dictionary, May 2013. `http://idrt.com/store/the-ultimate-american-sign-language-dictionary`.
37. von Ahn, L. Duolingo: learn a language for free while helping to translate the web. In *Proceedings of the 2013 international conference on Intelligent user interfaces (IUI)*, ACM Press (2013), 1–2.
38. Wang, H., Stefan, A., Moradi, S., Athitsos, V., Neidle, C., and Kamangar, F. A system for large vocabulary sign search. In *Trends and Topics in Computer Vision*. Springer, 2012, 342–353.
39. Wang, R. Y., and Popović, J. Real-time hand-tracking with a color glove. In *ACM Transactions on Graphics (TOG)*, vol. 28, ACM (2009), 63.
40. Yang, M.-H., and Ahuja, N. Recognizing hand gestures using motion trajectories. In *Face Detection and Gesture Recognition for Human-Computer Interaction*. Springer, 2001, 53–81.
41. Zafrulla, Z., Brashear, H., Starner, T., Hamilton, H., and Presti, P. American sign language recognition with the kinect. In *Proceedings of the 13th international conference on multimodal interfaces*, ACM (2011), 279–286.
42. Zinza, J. E. *Master ASL!* Sign Media Inc, 2006.