



# DXplain Mobile: An Assessment of a Smartphone-Based Expert Diagnostic System

## Citation

Hamilton, Baker H. 2016. DXplain Mobile: An Assessment of a Smartphone-Based Expert Diagnostic System. Master's thesis, Harvard Medical School.

## Permanent link

<http://nrs.harvard.edu/urn-3:HUL.InstRepos:42061446>

## Terms of Use

This article was downloaded from Harvard University's DASH repository, and is made available under the terms and conditions applicable to Other Posted Material, as set forth at <http://nrs.harvard.edu/urn-3:HUL.InstRepos:dash.current.terms-of-use#LAA>

## Share Your Story

The Harvard community has made this article openly available.  
Please share how this access benefits you. [Submit a story](#).

[Accessibility](#)

## **DXplain Mobile: An Assessment of a Smartphone-Based Expert Diagnostic System**

Baker Hamilton MD, Mitchell Feldman MD, Edward Hoffer MD, Richard Kim, Kathleen  
Famiglietti, Adam Landman MD, Gordon Schiff MD, Henry Chueh MD

April 15, 2016

## **Abstract**

**Objectives:** Clinical decision support tools may help reduce diagnostic error, and improving the accessibility and usability of these tools may encourage more frequent use. We have developed a novel, self-contained, iPhone-based implementation of DXplain, a popular diagnostic decision support system. In this study we evaluated the diagnostic agreement of this new application compared to the standard web-based version of DXplain.

**Methods:** A native DXplain application for iOS was developed, with modifications made to DXplain's original database to maintain acceptable performance within the more limited form factor of a smartphone. Each of the 41 Clinical Pathological Cases (CPCs) from the New England Journal of Medicine (NEJM) in 2015 were entered into the smartphone application as well as the standard version of DXplain, and the ranking of each case's final diagnosis was compared.

**Results:** DXplain's database contained 52 of the 65 final diagnoses found within the CPCs (80%), and this set of final diagnoses appeared within the calculated differential of both versions of the software in 38 instances (73%). In 21 of these cases (55%) the iOS application and the web version of DXplain agreed exactly on the position of the final diagnosis, and the weighted kappa score for agreement between the 38 diagnoses was 0.83 (95% CI 0.76 - 0.90).

**Conclusions:** DXplain for iOS appears to have strong agreement with the traditional web version of DXplain. Diagnostic discrepancies against actual cases should be explored to improve the underlying algorithms and knowledge base. Additional usability testing should also be performed, with a possible pilot study of user interaction and satisfaction prior to an official release of the application.

**Keywords:** Decision Support Systems, Clinical; Medical Informatics Applications

## **1. Background and Significance**

Expert medical diagnostic systems have been an area of study and development in medical informatics for over 50 years [1]. Of existing diagnostic systems, DXplain is amongst the most widely used and longest running, having been developed and maintained since 1984 at the Massachusetts General Hospital Laboratory of Computer Science (MGH LCS) [2]. The project's founder, Octo Barnett, originally conceived of DXplain as an educational tool covering the scope of general internal medicine (and its related specialties) and pediatrics. The system would accept patient findings as entered by a clinician, and then output a differential diagnosis. At any point during a session, the user could request an explanation from DXplain of why any disease presented within the differential diagnosis should be considered [3].

The initial version of DXplain was released in 1986, and within a year supported 2000 diseases and was being used throughout the US. Its performance was assessed in the New England Journal of Medicine (NEJM) in 1994, and it compared favorably against Iliad, Meditel, and QMR – three other popular computer-based diagnostic systems at the time [4]. In 1996, DXplain was made available as a web-based application, which has since remained its exclusive format.

DXplain is used in similar measure among medical students, residents, and staff physicians, and has been shown to have a favorable impact on residents' differential diagnoses and management plans [5,6,7]. This effect of broadening the differential diagnosis and reducing the tendency to anchor on an incorrect initial principal diagnosis was hypothesized by Elkin et al. to be a likely explanation for the more efficient patient

workups and reduced charges observed with internal medicine residents using DXplain [8].

In a more recent comparison from 2012, DXplain again performed well against other leading differential diagnosis support systems [9]. Its knowledge base currently contains over 2400 diseases and over 5000 clinical findings, connected through approximately 230,000 data points describing disease/finding relationships [2]. DXplain is presently made available through institutional licensing, with all proceeds ultimately supporting the not-for-profit General Hospital Corporation. A no-cost evaluation license is available, as well as a demo of the software which can be accessed over the web [10].

### **1.1. Background – The Need for a Smartphone-Based DXplain**

While computer-aided diagnostic decision support systems have been developed to address problems related to diagnostic error, their widespread adoption has been limited by numerous challenges, including form factor, limited integration into the existing workflow, incompleteness or immaturity of the underlying database, and a lack of perceived utility for daily clinical work [11].

New form factors such as smartphones and tablets are now nearly ubiquitous in clinical settings, and these devices make clinical software readily accessible during rounds or from the patient's bedside. This increasingly prevalent mobile technology provides an opportunity to design a diagnostic support system that can be more easily integrated into clinicians' daily workflow, perhaps leading to increased usage [1].

There has already been some work done toward the development of mobile diagnostic support tools, most notably Isabel, which was evaluated alongside DXplain in

2012 by Bond, et al [9]. These apps tend to require Internet connections so that case processing can occur on remote servers, which can be problematic in hospitals containing many areas with unreliable cellular or Wi-Fi connectivity. Usability challenges such as this have been a significant hindrance to widespread adoption of clinical decision support tools [12], although the advent of more powerful smartphones that can locally store the large amount of data required for provision of quality diagnostic decision support might signify an opportunity for change.

## **1.2. Background – DXplain Mobile Development**

For these reasons, the team responsible for the development and maintenance of DXplain sought to create a mobile version of the software. This interdisciplinary team included a project manager, a pediatrician, an internist, and a web developer. The first author of this paper, with experience as a practicing emergency medicine physician as well as with iOS development, was added to the group to develop this application as part of his clinical informatics research fellowship at the MGH LCS.

Among the various smartphone operating systems, iOS was chosen as the platform for the initial release of DXplain due to the high degree of popularity of iPhones amongst clinicians [13]. Beta testing was established through TestFlight, which enabled the beta-stage app to simply be pushed to the team members' iPhones when an update was available. The different types of iPhones and varying versions of iOS among the team members was useful in providing information about backward or forward compatibility, as well as about the differences in time required to calculate differential diagnoses along a spectrum of older and newer devices. This allowed for a weekly cycle

of updates to the application, followed by a team meeting with feedback, which subsequently led to any needed bug fixes or optimizations, as well as implementation of new features.

Apple had released its language “Swift” only a few months prior to the start of this project’s development, and although it was new and unproven, the anticipation that this language would be the future of iOS development made it a clear choice. The developer learned the language as the new version of DXplain was written, although already having familiarity with the underlying development environment and related frameworks substantially reduced the learning curve. Swift was still undergoing significant updates following its official release, and updates in the language (particularly with releases of major versions, such as 2.0) would occasionally cause the project’s code to break. Fortunately, many of the errors ended up only being related to changes in syntax and were straightforward to fix.

Performance limitations due to the iPhone’s underlying hardware were occasionally in conflict with the twin goals of storing DXplain’s large dataset on the phone (rather than outsourcing case processing to a remote server) while still keeping the application quick and responsive. For example, the code for the algorithms to calculate differential diagnoses was implemented and evaluated for correctness using only a subset of the disease database. When the entire database was subsequently included, the application was initially rendered nearly unusable as the amount of time needed to process cases dramatically increased. Through careful code analysis, the issue was resolved as inefficiencies were reworked or eliminated, but instances like this motivated many modifications to streamline and simplify the mobile application’s database.



## **2. Objectives**

As a comprehensive, yet significantly more compact database was created for use with DXplain mobile, testing was needed to ensure that its results were in agreement with the standard web-based version of DXplain. Additional considerations for testing included the speed of case processing as it related to the number of findings entered, and the influence that mechanisms for more heavily weighting crucial findings would have on the system's overall diagnostic accuracy.

## **3. Methods**

All NEJM clinical pathological cases (CPCs) from 2015 were entered into both versions of DXplain, and the agreement between the two with regard to the position of the case's final diagnosis was compared. These CPCs represent diagnostically challenging cases related to the field of internal medicine, although they can also touch on unusual cases in other fields. In addition to the case content matching DXplain's stated scope (in most instances), the CPCs had the benefits of being recent, offering a complete account of the patient's history, and providing the final diagnosis against which DXplain's diagnoses can be compared. The CPCs were also managed at the same institution that develops DXplain.

The research fellow/developer was responsible for entering each of the cases. To reduce the potential for bias, all findings given for each case were entered to the point where the case discussants began to speak about the differential diagnosis. This approach had the drawback of not reflecting typical use of the application, as an average of 4 non-

demographic findings are typically entered on the web version of DXplain, whereas some NEJM cases required entry of over 100 findings [14].

The decision was made to proceed in this fashion for two reasons. First, to reduce bias that could be introduced by preferentially selecting findings known to support the final diagnosis, which was obviously known to the researcher working from the NEJM published cases. Second, it is anticipated that future use of DXplain may be through integration with an electronic health record, in which case it is envisioned that a potentially large number of case findings will be used as input.

The elements of each case's patient demographic makeup, history, physical exam, laboratory, and radiology workup were recorded into a spreadsheet, along with information about whether each finding was present or absent. These terms were then manually entered into the standard version of DXplain, requiring decisions to be made in some cases about which term in the DXplain vocabulary was the best representation of the case's term. The parsed cases with their mapped terms are available in the Appendix.

DXplain is designed in such a way that the temporality of findings is not a factor when entering cases. All pathological findings, regardless of whether they improved throughout the patient's course of illness, were entered. Additionally, not all terms from each case were present in the DXplain findings database, and these were omitted. A script was created to format the Excel spreadsheets containing case findings to a format that could be easily loaded by DXplain mobile, and once all 41 CPCs loaded, the cases were run and the final diagnosis rankings were compared.

DXplain also contains a feature called "focus" that gives the user the option of specifying which findings should be regarded as required by the application. If one or

more findings is selected for “focus”, then every disease in the resulting differential must contain each of those selected findings. This function was anticipated to be especially useful when scoring cases with large numbers of otherwise undifferentiated findings (such as NEJM CPCs), but the presence of multiple diagnoses in some cases suggested that the “focus” feature could also be severely limiting, as the patient’s second disease could be excluded if only the first disease contained all the “focus” findings. To address what in some instances could be overly exclusive behavior in the “focus” feature, we also created and tested a modified version in which each disease in the differential was only required to contain one of the “focus” findings.

The “focus” findings were selected from the titles of the CPCs, which provide symptoms, labs, or other findings felt by the case’s author to be of particular importance. For example, the title of the second NEJM CPC from 2015 is “A 25-year-old man with abdominal pain, syncope, and hypotension.” The findings “abdominal pain,” “syncope,” and “hypotension” were then chosen within the case as “focus” findings when evaluating the performance of this feature.

### **3.1. Methods - Application Description**

The application’s interface as described in this section relates to how it appeared at the time of this study, which was more intently focused on correctness and efficiency of the underlying algorithms rather than on usability. The interface will almost certainly be different after changes made subsequent to usability testing, although all of the features and functionality mentioned here will still be present.

DXplain mobile's initial screen displays an empty list to be populated by the user with the patient's findings. The user has the option to search for these findings, which might include symptoms, physical exam findings, and lab or radiographic results. If a particular finding is within a hierarchy that has more specific findings, a graphic will indicate this within the search interface, allowing the user to choose a more specific finding without being required to perform additional typing. For example, "chest pain" returned in the search function will have a disclosure indicator (Figure 1A) that reveals a list of more specific findings if tapped (Figure 1B). By swiping on a case finding, the user is presented with the "focus" option to inform DXplain that the finding is likely to be especially important in formulating the differential (Figure 1C).

Patient demographics, including approximate age, gender, and duration of symptoms, have a different mechanism for entry than other case findings. DXplain uses demographics as one of the chief determinants of what diseases to include or exclude from the differential. To emphasize their importance to the user, these values always remain prominently visible within a dedicated space on the findings entry screen (Figure 1C).

When the "DDx" button is tapped, the findings are scored and a differential diagnosis is presented to the user (Figure 1D). The diseases are sorted in descending order by the score calculated from the entered findings. Any disease that may require urgent action is indicated with a red scoring icon and an exclamation point. The user is initially shown the entire differential, and is given the option to view the results by disease prevalence or urgency through a tab bar located at the bottom of the screen (Figure 1E).

The “Refine” feature helps the user narrow the differential by asking about findings that likely be present based on the most highly ranked diseases in the differential (Figure 1F). This can expedite the entry of findings, as the user simply has to choose between the options of “Yes,” “No,” or “Unknown” in response to a list of possible findings, which is continually recalculated based on the user’s answers.

By tapping on a disease in the differential diagnosis list, the user can learn why it was included in the differential, based on the various demographics and findings that were entered. A separate differential diagnosis tailored to that disease is also presented, as well as a button that will automatically perform a Google search for the disease referencing only reputable medical websites (Figure 1G).

Analytics for the application are provided through the settings screen (available from the initial screen from which findings are entered), along with terms of service (Figure 1H). The user can optionally provide additional information about his/her medical role, specialty, and experience, which, along with case findings and other usage data, can then be reviewed by the application’s developers through the Google analytics cloud service. No identifying patient information is used by the application.

### **3.2. Methods - Data Analysis**

The data from the parsed CPCs, including patient demographics, case findings in both NEJM and DXplain terminology, and final diagnoses were stored in Microsoft Excel spreadsheets (Microsoft Corporation, Redmond, WA). A descriptive analysis was performed using these data, detailing characteristics about the set of cases such as the median number of findings, how many of these findings were present or absent, and the

median number of diagnoses per case. A determination of the number of CPC diagnoses present within the DXplain database was made, as well as the positions of the correct NEJM diagnoses between the two versions of DXplain.

A kappa coefficient was calculated to measure agreement between the results of the two software versions. The kappa was weighted to account for variations in the distance between the applications' rankings, and the weights were equally spaced. As a threshold was implemented within the iOS application to include a maximum of 45 diseases in the differential diagnosis, diseases that scored below this threshold were not included when calculating the kappa value. A Bland-Altman plot was generated to show differences in agreement. These statistical tests and plots were created using R and its "VCD" and "BlandAltmanLeh" packages [15, 16, 17], and were performed in consultation with a statistician from the Harvard Catalyst Biostatistics Program.

The time required for case processing was determined by placing timers within the application code that started when the user tapped the button to generate the differential diagnosis, and ended when the results were displayed on screen. These tests were performed on an iPhone 6 running iOS 9.3.1. The relationship between the number of seconds required for the case to be processed and the number of findings in the case were depicted on a scatter plot using Excel. A similar plot was also constructed comparing the number of findings per case and how the iOS application performed with regard to the final diagnosis' position.

To visually depict the performance of the "focus" function, a plot was created within Excel containing the positions of the final diagnoses run without "focus", and those with "focus" using both Boolean AND and Boolean OR.

#### 4. Results

In each of the 41 CPCs, there were a median of 40 present findings and 34 absent findings (with a median of 75 total findings for all cases). The median time required to process each case was 3.9 seconds, although for the five cases with 30 findings or less the average processing time was 1.04 seconds (median 0.93 seconds) (Figure 2).

There was a median of 1 diagnosis per CPC (average of 1.6), and the highest number of diagnoses within a single CPC was 4. A total of 65 diagnoses were found within all 41 CPCs, and of these, 52 were present in the DXplain knowledge base (80%). DXplain categorized the prevalence of 24 of the final diagnoses (46%) as being common.

With regard to the concordance of the mobile application with the web application, there was complete agreement of which 38 of the 52 possible diagnoses appeared on the differentials. In these 38 instances, the iOS and web versions agreed exactly on the position of the diagnosis 53% of the time, and the weighted kappa was 0.83 (95% CI 0.76 – 0.90). A representation of these data in the form of a Bland-Altman plot is shown in Figure 3, which depicts a mean difference of -0.37, and upper and lower limits of 6.76 and -7.49, respectively.

Concerning overall validity, the final diagnosis appeared within the top 5 diseases in the differential of the iOS version in 20/52 instances (38%). Figure 4 depicts the relationship between the number of findings and diagnostic performance. Requiring at least one of a set of carefully chosen findings to be present in each of the diseases on the differential (via the “focus” feature using Boolean OR) resulted in the final diagnosis being present in the top 5 diseases in 17 instances (33%). Requiring all of the “focus”

findings, using Boolean AND, to be present in each disease, resulted in the final diagnosis being present in the top 5 in 9 instances (17%).

As mentioned, the final diagnosis did not always appear within the calculated differential diagnosis. If the success of appropriately ranking the final diagnosis is instead measured relative to the differentials that contain that diagnosis, the numbers better reflect the anticipated loss of sensitivity with a compensatory increase in specificity. In the 32 instances in which the final diagnosis appeared in the differential when using “focus” with Boolean OR, the correct diagnosis was in the top 5 17 times (53%). The number improved to 100% in the 9 instances in which the final diagnosis was present when using Boolean AND. This information is depicted in Figure 5.

## **5. Discussion**

The iOS and web versions of DXplain performed equally well with each diagnosis in terms of inclusion within the calculated differential diagnosis, and the high kappa coefficient indicates that our goal of generating similar output from separate databases differing greatly in terms of size and complexity was successful (Figure 3). The processing times appear to be acceptable and were linearly associated with the number of findings (Figure 2), and cases with fewer than 30 findings (most representative of typical cases) loaded within approximately 1 second [14].

The overall diagnostic performance of DXplain was not as good as has been reported in the literature during previous evaluations, and this is likely due to uncommon presentations of more exotic diagnoses being featured in current NEJM CPCs. It is possible that technologic advances related to detection of these diagnoses have not yet



been integrated into the DXplain knowledge base. Additionally, there are also a number of the NEJM CPCs that would not realistically involve the use of DXplain. For example, case #15 involving penetrating orbital trauma clearly fell outside of DXplain's intended scope (though interesting in other respects), but was included nonetheless to avoid selection bias.

Another factor that may have limited the performance of the system is the indiscriminate manner in which findings were added to each case. The number of findings alone did not appear to affect the diagnostic performance, as shown in Figure 4. Previous studies evaluating the diagnostic performance of DXplain have generally input a subset of case findings felt by investigators to be most important, which is a much closer approximation to how a standalone application would be used in an actual clinical setting [9]. Since this study was primarily concerned with assessing the concordance of DXplain mobile with the web version, case findings were not tailored for optimal diagnostic accuracy. Doing so would have likely been redundant, since if DXplain mobile's results are found to be consistent with the standard version, then the findings of previous studies with regard to the latter should logically extend to the mobile version as well.

The multitude of findings entered from the CPCs might mimic a user who is genuinely stumped, and reluctant to leave out any findings for fear of excluding something that may be relevant to the underlying disease entity. DXplain's "focus" function was assessed as a potential means of addressing the loss of accuracy that may be associated with entering all of a case's findings rather than only those deemed meaningful by a seasoned clinician. Using the "focus" feature, the user can feel free to enter every finding available and then explicitly clarify which ones DXplain should pay

particular attention to. When “focus” is applied to cases, a predictable tradeoff in sensitivity for specificity is seen. While the likelihood that the final diagnosis will appear in the differential decreases as the required number of findings increases, the ranking of the final diagnosis does appear to be improved if it is present (Figure 5).

There has been substantial demand from the DXplain user community for a mobile version of the software, and we feel that this successful implementation of a comprehensive, fast, and accurate version of DXplain on a smartphone platform represents significant progress toward meeting this demand. Built-in data collection for analytics purposes will be helpful in providing insights into user demographics and how the app is being used, which can then contribute toward improvement of the database and application design. Through the feedback obtained from DXplain’s users engaging with this new software, we hope to make real strides in tackling the long-standing problem of limited adoption of expert diagnostic systems.

Before releasing DXplain for iOS, more usability testing will need to be performed. The beta version of the application used for this study (seen in Figure 1) contains a practical interface that allows for full functionality of the range of features that are anticipated to be present in the released version. Input was obtained early on in the development process from an in-house graphic designer to help ensure adherence with Apple’s Human Interface Guidelines and overall best practices. The developer and other physicians on the project team also tested the app in clinical scenarios to help guide the design.

We sought a formal user experience review through consultation with a usability expert at Partners HealthCare Information Systems, which revealed many areas for

improvement in the user interface (UI) within broad categories like aesthetics, usability, and consistency. The recommendations provided by the consultant also served to highlight the importance of external testing, and once the UI has been revised, we plan to extend the group of beta testers to incorporate numerous additional clinicians with varying degrees of familiarity with DXplain.

Ultimately, our plan is to make DXplain for iOS readily accessible to clinicians around the world through the iTunes store. With this new application, we have focused on tackling many of the core issues that have historically limited adoption of similar tools by creating an application employing a mature database for a device that is easily integrated into the clinical workflow [1, 11]. Enhanced usability has been addressed by making the application functional and quick regardless of Internet availability, and analytics and feedback will be incorporated from users to continually improve interface design and both completeness and accuracy of the underlying database.

## **6. Conclusions**

DXplain for iOS is a diagnostic decision support tool that appears to be consistent with the standard web version of DXplain. In addition to providing similar diagnostic performance when tested against a range of complex cases, it succeeds in meeting the usability expectations of being both rapid and having functional independence of Internet connectivity. Preliminary user experiencing testing has been performed, although more testing in actual clinical scenarios will be required before the application is ready for release to the public.

## **7. Clinical Relevance Statement**

This study investigates the agreement between the results of a novel iOS-based expert diagnostic support system with its web-based predecessor. Ensuring comparable performance is a crucial step in establishing trust in this clinical decision support tool, which with its increased portability and ease of use may result in increased usage in a clinical setting, magnifying the studied benefits of the original system, including more efficient and less expensive patient care [6].

## **8. Conflict of Interest**

The first author declares he has no conflicts of interest in the research.

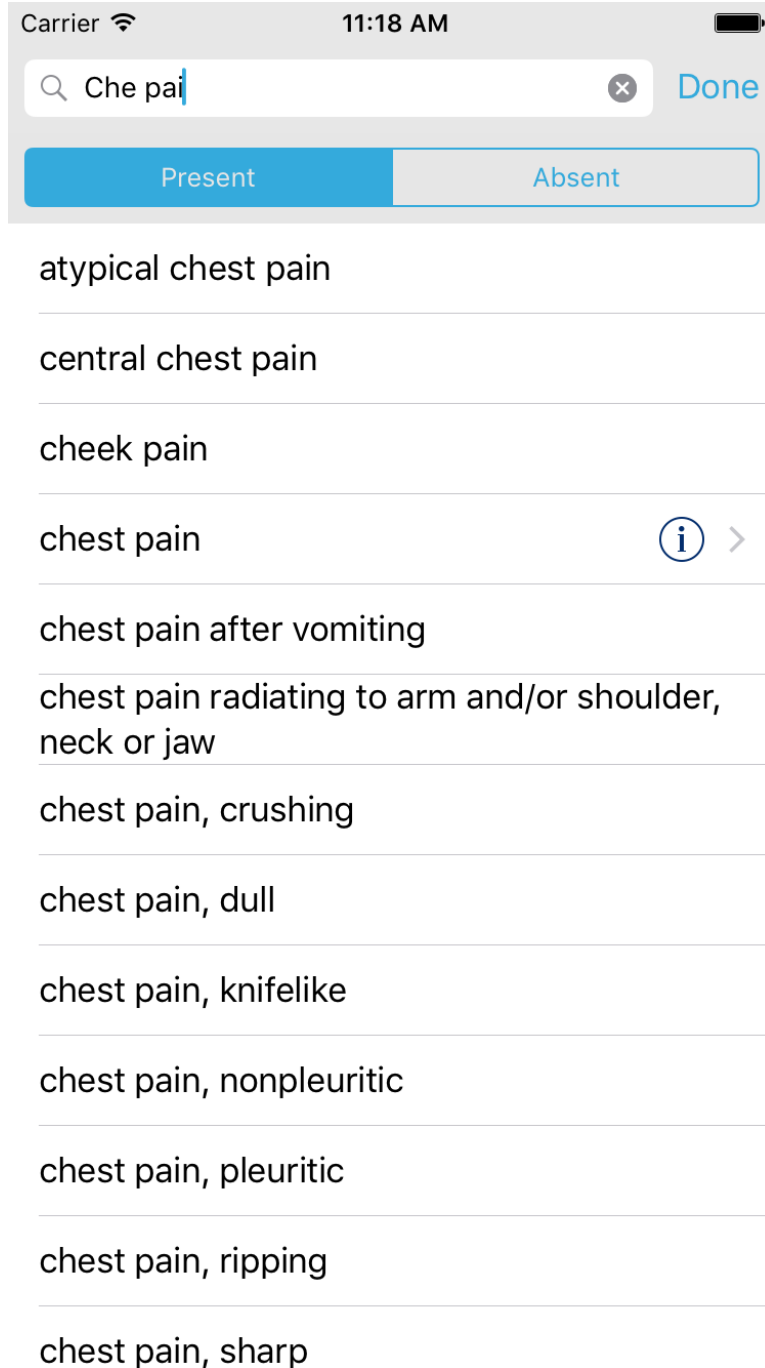
## **9. Human Subjects Protections**

No human and/or animal subjects were involved in this research.

## **10. Acknowledgements**

The author would like to express his gratitude toward and acknowledge the support of the Massachusetts General Hospital, the DXplain team including Drs. Feldman and Hoffer, Kathleen Famiglietti, Richard Kim, and Dr. Henry Chueh, as well as Drs. Schiff, Landman, Hoffer, and Feldman who graciously comprised my thesis committee, the Harvard Catalyst Biostatistics Program, the Division of Biomedical Informatics at Harvard Medical School, and the Boston Area Research Training Program in Biomedical Informatics (PI: Dr. Alexa T. McCray, Grant number T15LM007092).

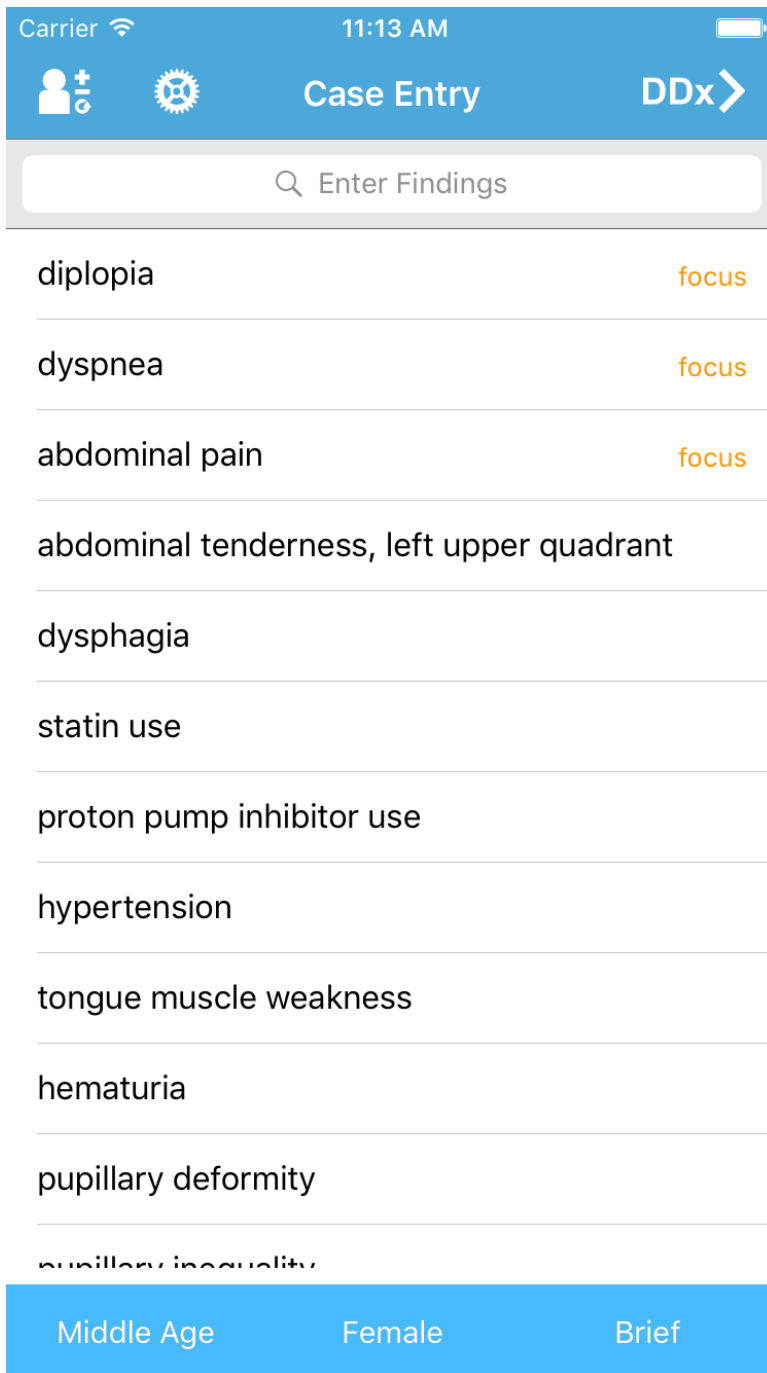
## 11. Tables and Figures



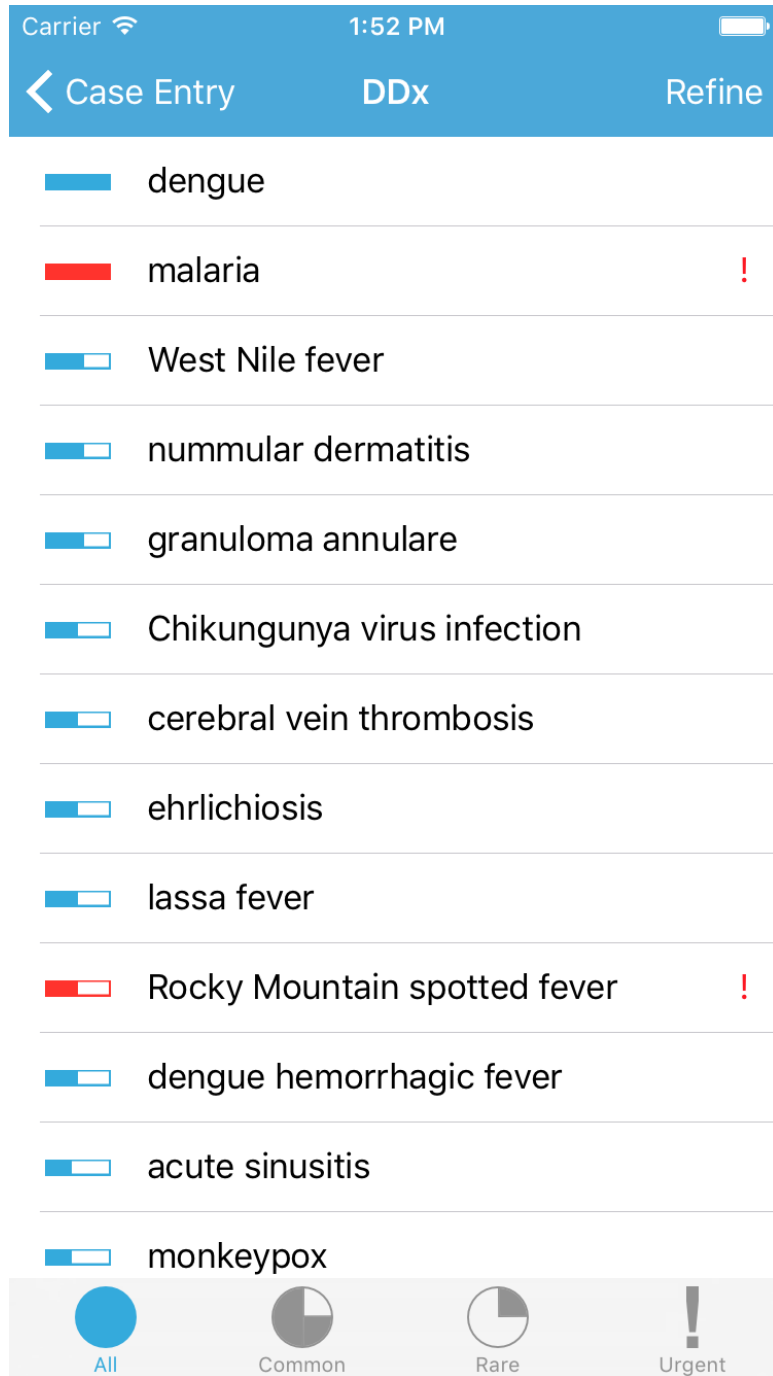
**Figure 1A.** Findings Entry. Any portion of entered text is matched, and findings with a hierarchy are denoted with a disclosure indicator



**Figure 1B.** Findings Entry. Tapping on a finding’s disclosure indicator reveals more specific findings, which can limit the amount of typing required by the user

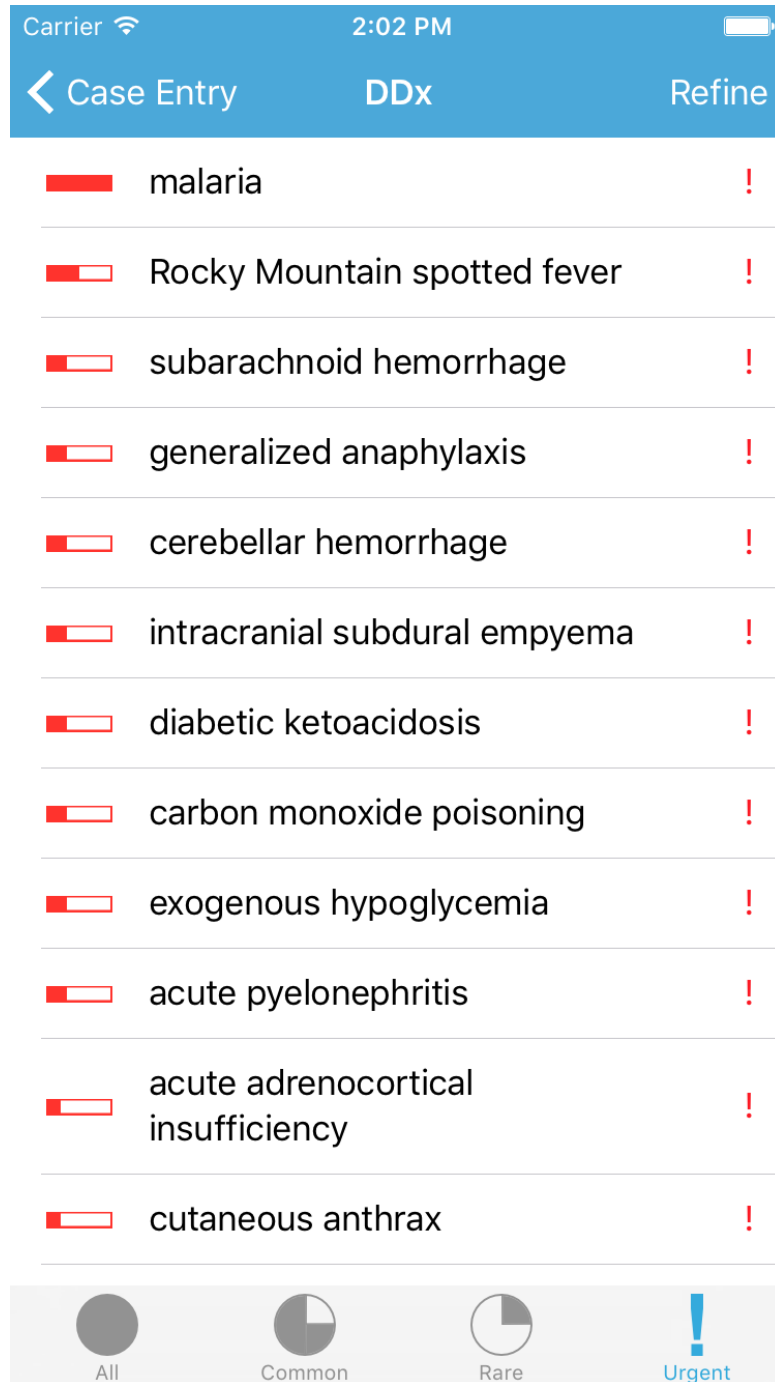


**Figure 1C.** List of Entered Case Findings. Swiping a finding will enable DXplain to “focus” on that finding, requiring it for every disease on the differential



**Figure 1D.** Differential Diagnosis. Diseases are ordered by likelihood based on entered findings. Red score bars and exclamation points denote diagnoses that may require urgent action





**Figure 1E.** Urgent Diseases. The tab bar at the bottom of the screen allows the user to view a subset of the differential, either by prevalence or by urgency (seen here)

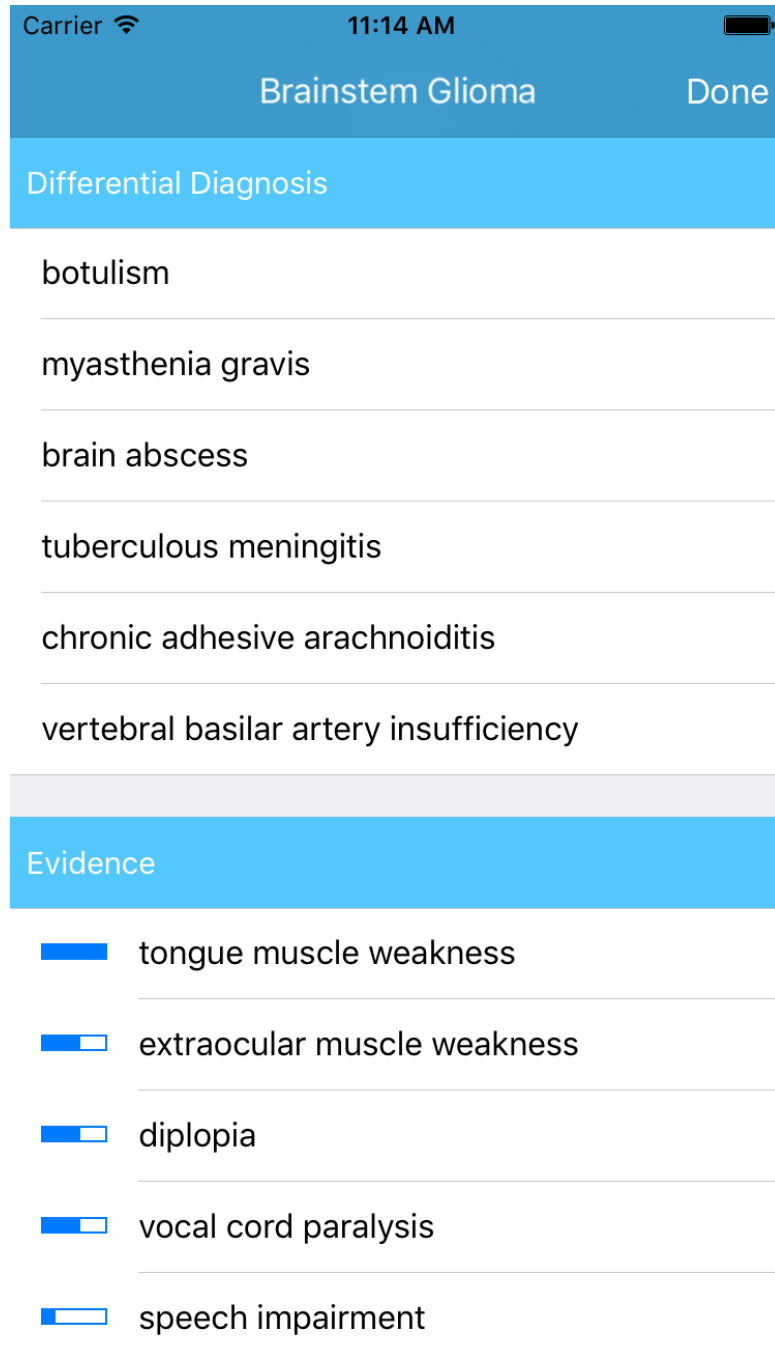
Carrier 11:14 AM

Cancel Refine Done

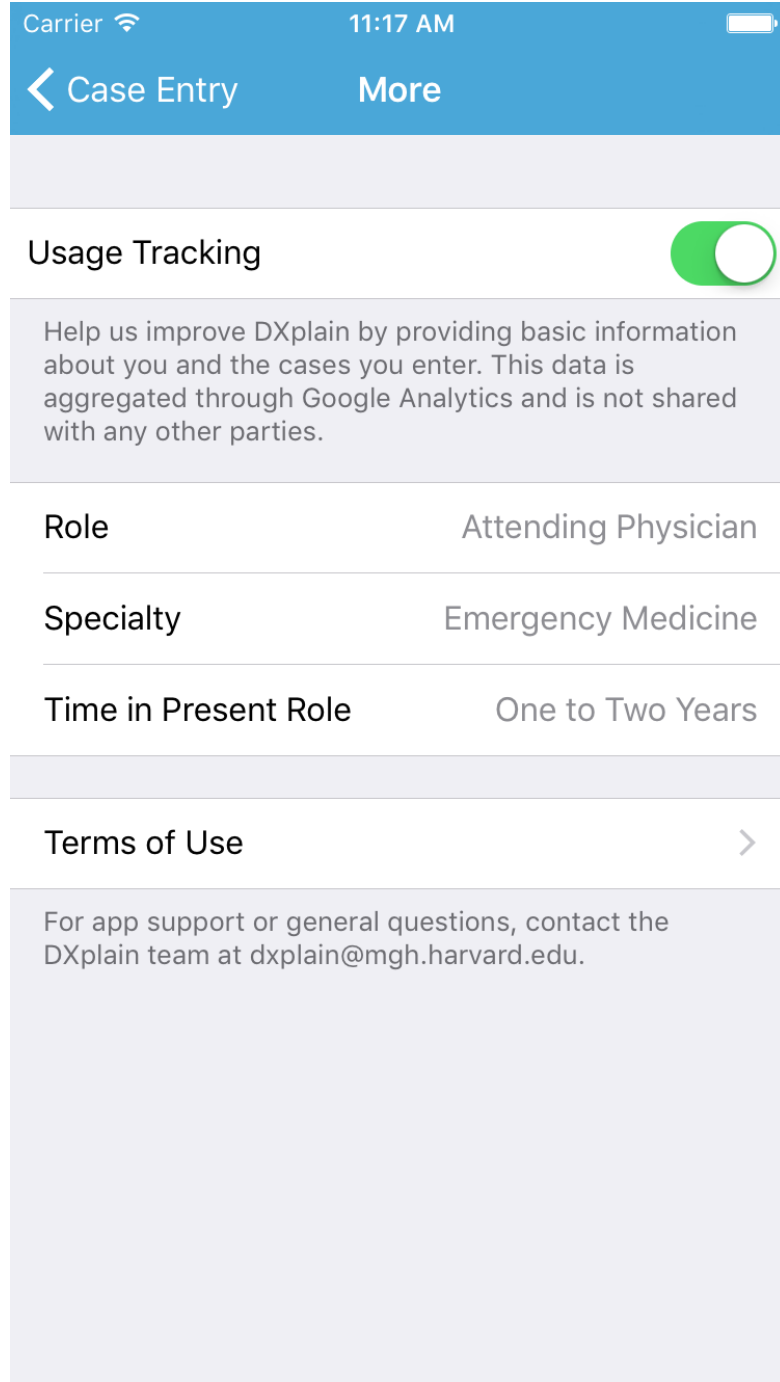
Are any of these findings present?

bulbar palsy	<input type="radio"/> Yes <input checked="" type="radio"/> Unk. <input type="radio"/> No
paralysis, progressive	<input type="radio"/> Yes <input checked="" type="radio"/> Unk. <input type="radio"/> No
toe paresthesia	<input type="radio"/> Yes <input checked="" type="radio"/> Unk. <input type="radio"/> No
glucose intolerance	<input type="radio"/> Yes <input checked="" type="radio"/> Unk. <input type="radio"/> No

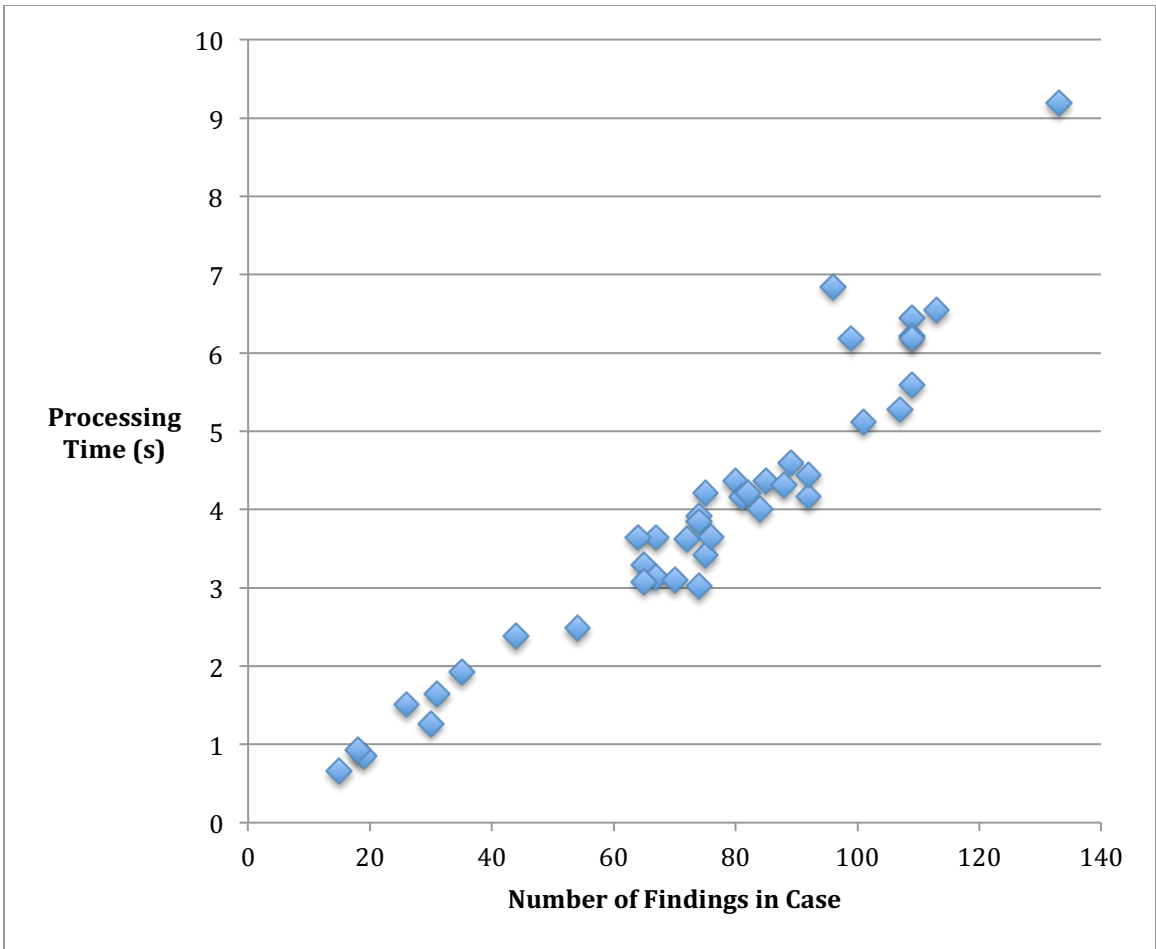
**Figure 1F.** Refine Findings. Using this option, DXplain will try to improve the accuracy of its differential by asking the user about findings that may be present based on the most highly ranked diseases



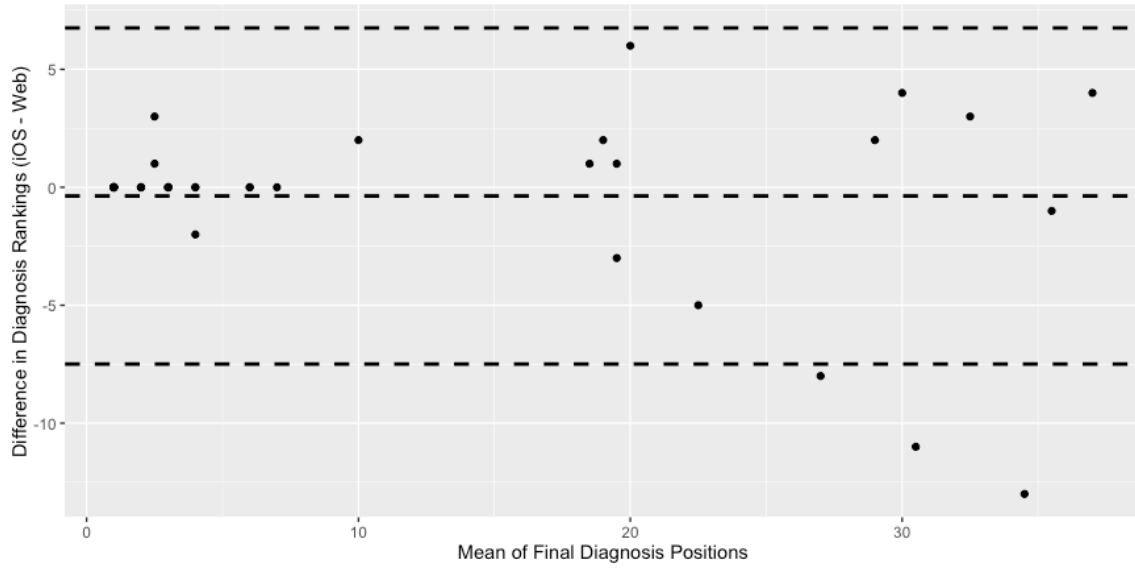
**Figure 1G.** Disease Information. Tapping on a disease will present a differential diagnosis for that disease, along with the entered findings that support the disease in the present case. A button is also provided that will perform a Google search on the disease, although it is not captured in this screenshot



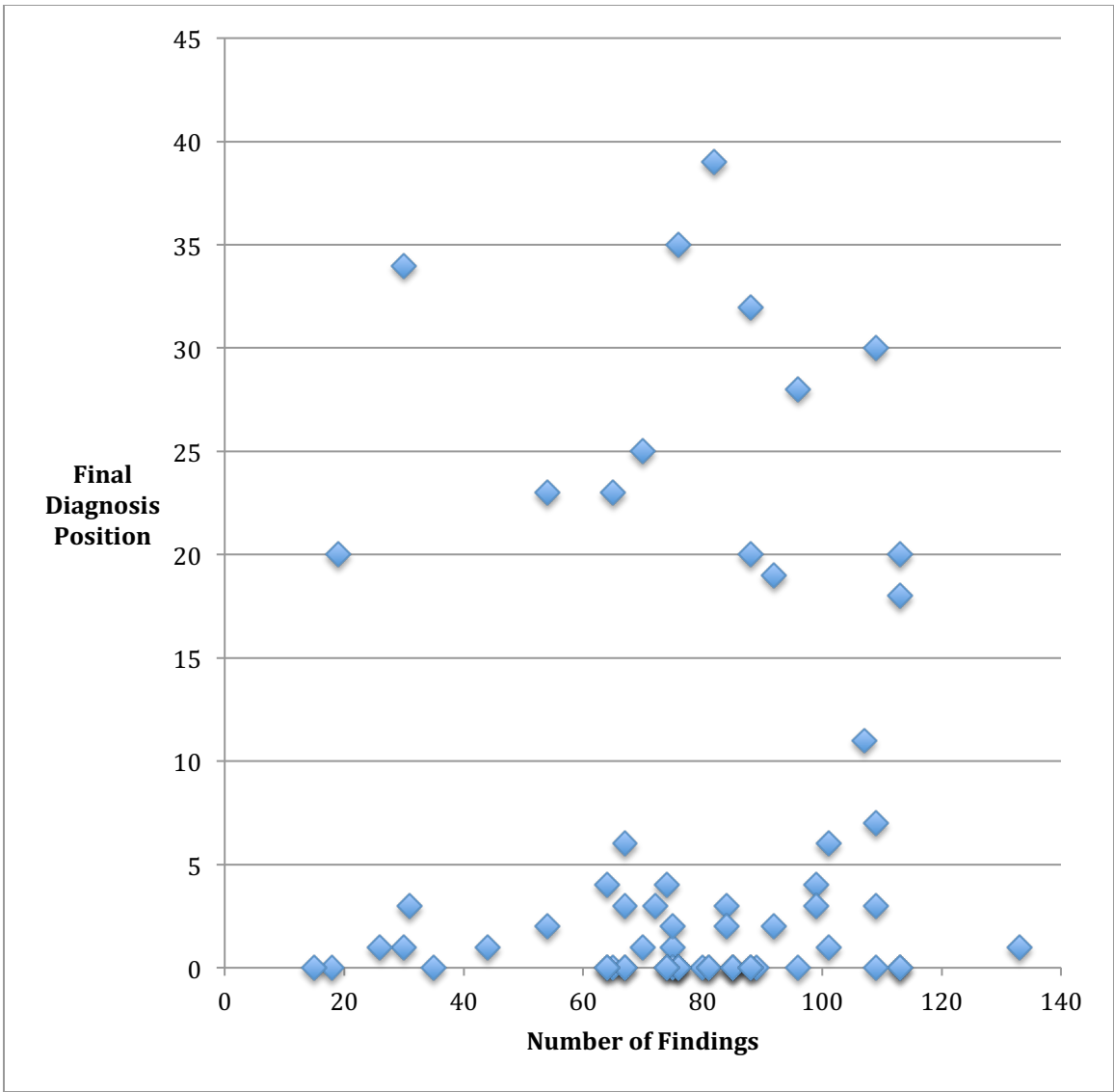
**Figure 1H.** User Options. The user can elect to opt in or out of providing information to Google analytics about entered cases, as well as information about their clinical practice



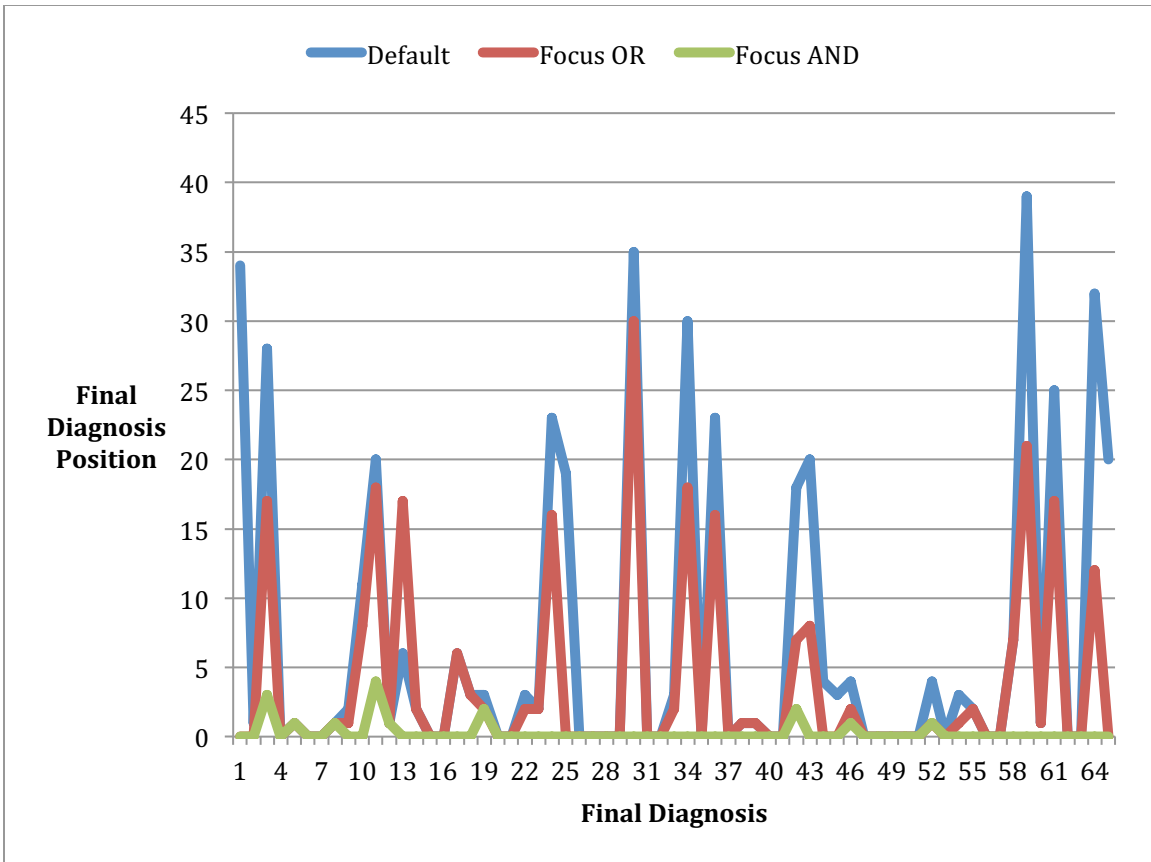
**Figure 2.** Relationship between number of findings and time required (in seconds) for DXplain mobile to process the case



**Figure 3.** A Bland-Altman plot showing the differences in positioning of final diagnoses between iOS and web versions of DXplain. The mean difference was -0.37, with an upper limit of 6.76, and a lower limit of -7.49



**Figure 4.** Relationship between the number of findings in a case and DXplain's ranking of the final diagnosis (ideal ranking being #1)



**Figure 5.** Ranking of final diagnosis without “focus” feature, and with “focus” as both Boolean AND and Boolean OR. The former requires all focused findings to support each disease on the differential, while the latter requires only one of the focused findings to support each disease. Note that a value of “0” on the Y-axis is used if the diagnosis was not present on the differential. The numerical representation of the diagnoses along the X axis are mapped to their corresponding names in the Appendix



## 12. References

1. Shortliffe EH, Cimino JJ, editors. Clinical decision-support systems. Biomedical informatics: computer applications in health care and biomedicine. 4<sup>th</sup> ed. London: Springer-Verlag; 2014.
2. DXplain [Internet]. Boston: Massachusetts General Hospital Laboratory of Computer Science; c1987-2012 [cited 2016 Apr 16]. Available from: <http://www.mghlcs.org/projects/dxplain>
3. Barnett GO, Cimino JJ, Hupp JA, Hoffer EP. DXplain. An evolving diagnostic decision-support system. JAMA. 1987 Jul 3;258(1):67-74.
4. Berner ES, Webster GD, Shugerman AA, Jackson JR, Algina J, Baker AL, et al. Performance of four computer-based diagnostic systems. N Engl J Med. 1994 Jun 23;330(25):1792-6.
5. Barnett GO, Famiglietti KT, Kim RJ, Hoffer EP, Feldman MJ. DXplain on the Internet. Proceedings of the AMIA Annual Symposium; 1998 Nov 7-11; Lake Buena Vista, FL. p. 607-11.
6. Bauer BA, Lee M, Bergstrom L, Wahner-Roedler DL, Bundrick J, Litin S, et al. Internal medicine resident satisfaction with a diagnostic decision support system (DXplain) introduced on a teaching hospital service. Proceedings of the AMIA Annual Symposium; 2002 Nov 9-13; San Antonio, TX. p. 31-5.
7. Feldman MJ, Hoffer EP, Barnett GO, Kim RJ, Famiglietti KT, Chueh HC. Impact of a computer-based diagnostic decision support tool on the differential diagnosis of medicine residents. J Grad Med Educ. 2012 Jun;4(2):227-31.

8. Elkin PL, Liebow M, Bauer BA, Chaliki S, Wahner-Roedler D, Bundrick J, et al. The introduction of a diagnostic decision support system (DXplain) into the workflow of a teaching hospital service can decrease the cost of service for diagnostically challenging Diagnostic Related Groups (DRGs). *Int J Med Inform.* 2010 Nov;79(11):772-7.
9. Bond WF, Schwartz LM, Weaver KR, Levick D, Giuliano M, Graber ML. Differential diagnosis generators: an evaluation of currently available computer programs. *J Gen Intern Med.* 2012 Feb;27(2):213-9.
10. DXplain [Internet]. Boston: Massachusetts General Hospital Laboratory of Computer Science; c1987-2016 [cited 2016 Apr 16]. DXplain Annotated Demo. Available from:  
<http://dxplain.org/dxp2/dxp.sdemo.asp?login=demslehome>
11. Miller RA. Computer-assisted diagnostic decision support: history, challenges, and possible paths forward. *Adv Health Sci Educ Theory Pract.* 2009 Sep;14 Suppl 1:89-106.
12. National Academies of Sciences, Engineering, and Medicine. Improving diagnosis in health care. Washington, DC: The National Academies Press; 2015.
13. Ventola CL. Mobile devices and apps for health care professionals: uses and benefits. *P T.* 2014 May; 39(5): 356-364.
14. Hoffer EP, Feldman MJ, Kim RJ, Famiglietti KT, Barnett GO. DXplain: Patterns of use of a mature expert system. *Proceedings of the AMIA Annual Symposium* 2005 Oct 22-26; Washington, DC. p. 321-5.

15. R Core Team. R: A language and environment for statistical computing. Vienna, Austria: R Foundation for Statistical Computing; 2013.
16. David Meyer, Achim Zeileis, and Kurt Hornik (2015). vcd: Visualizing Categorical Data. R package version 1.4-1.
17. Bernhard Lehnert (2015). BlandAltmanLeh: Plots (Slightly Extended). Bland-Altman Plots. R package version 0.3.1. <https://CRAN.R-project.org/package=BlandAltmanLeh>.