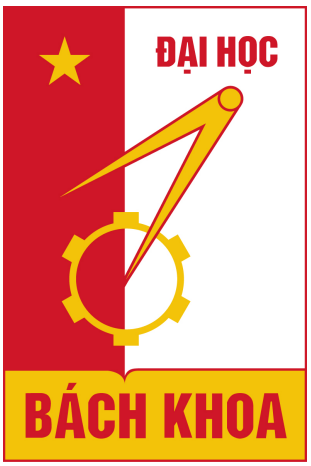
TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI
VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG

BÁO CÁO THU HOẠCH

BÀI BÁO

"DataTone: Managing Ambiguity in Natural Language Interfaces for Data Visualization"



Nhóm sinh viên thực hiện: **Nguyễn Quang Quý**
Tạ Bảo Thắng

Giáo viên hướng dẫn : **TS Nguyễn Kim Anh**

HÀ NỘI
Ngày 20 tháng 12 năm 2017

Mục lục

Danh sách bảng

Danh sách hình vẽ

1 Vấn đề được giới thiệu trong bài báo

Phân tích dữ liệu là một công việc khó, với sự hỗ trợ của các công cụ để biểu thị dữ liệu như Microsoft Exel, Tableau, ... Nhưng để trả lời cho những câu hỏi tinh vi, yêu cầu người dùng phải có hiểu biết về các công cụ dành cho chuyên gia phân tích. Nhưng có một cách dễ dàng hơn để giải quyết điều này đó là sử dụng giao diện ngôn ngữ tự nhiên. Người dùng có thể đặt câu hỏi trực tiếp mà không cần học các công cụ hoặc cách chuyển câu hỏi của họ thành trang tính hoặc các thao tác hiển thị dữ liệu. Tuy nhiên, ngôn ngữ tự nhiên thường tồn tại với những nhập nhằng, và sự nhập nhằng thể hiện ở nhiều mức :

- thứ nhất, câu hỏi của người dùng có thể sẽ không được xác định.
  Ví dụ: khi người trong câu hỏi có từ "product", và trong CSDL tồn tại "product category" và "product name" khi đó "product" sẽ tương ứng với cụm từ nào ?
- thứ hai, có thể có nhiều câu trả lời cho câu hỏi của người dùng.
  Ví dụ:"show revenue for New York City and Washington DC in 2012."
  Mục đích câu hỏi là so sánh tổng GDP của NYC các năm với WDC năm 2012 hay cả hai thành phố này đều trong năm 2012 ?

Thậm chí khi sự nhập nhằng về mặt ngôn ngữ được giải quyết, thì chúng ta cũng cần lựa chọn được cách biểu thị dữ liệu cho phù hợp:

- biểu đồ thanh chồng nhau: x- thời gian, y- mỗi khối ứng với 1 thành phố
- 2 biểu đồ cột, mỗi cột ứng với một thành phố
- biểu đồ đường - mỗi đường ứng với một thành phố
- ...

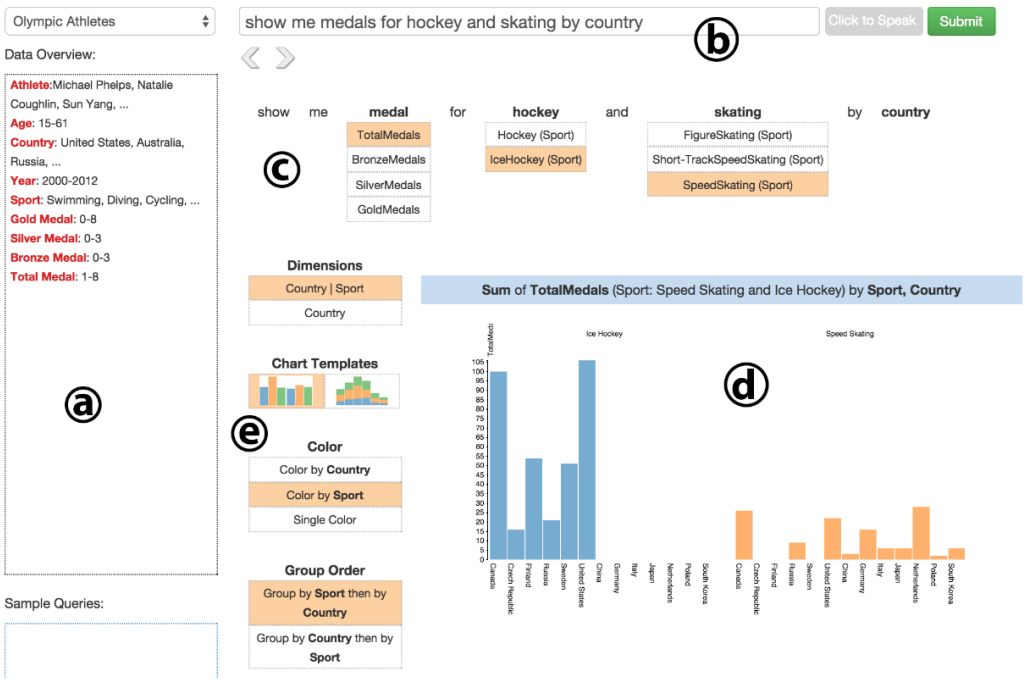
Có nhiều hệ thống đã được thiết kế để có thể giải quyết được vấn đề này như IBM Watson Analytics, Microsoft Power BI, ...Nhưng hạn chế của các hệ thống này là ràng buộc người dùng với một tập mẫu câu hỏi đã được lược bỏ đi sự nhập nhằng để cho hệ thống có thể hiểu được, điều đó làm mất đi sự toàn vẹn trong câu truy vấn của người dùng. Trong bài báo này, đã đề xuất một hệ thống mới có tên DataTone cho phép giữ lại sự toàn vẹn trong câu truy vấn của người dùng.

Cách thức hệ thống DataTone hoạt động: người dùng nhập vào câu truy vấn ngôn ngữ tự nhiên , hệ thống phân tích truy vấn và tạo một hoặc nhiều đặc tả dữ liệu (DSP) dùng để tạo ra truy vấn đối với CSDL. Từ DSP và truy vấn đối với CSDL, DataTone tạo các đặc tả trực quan (VSP) và tạo các biểu diễn tương ứng. Hệ thống sẽ xếp hạng các biểu diễn được tạo ra và trả về biểu diễn có rank cao nhất cho người dùng. Sự nhập nhằng được giải quyết nhờ vào sự kết hợp giữa điều khiển của người dùng và điều khiển của hệ thống. Bài báo đưa ra một khái niệm "ambiguity space" để nói đến mô hình của

sự nhập nhằng xuất hiện trong mỗi bước của đường ống trên. Tương tác với "ambiguity space" thông qua "ambiguity widget" - cho phép người dùng điều chỉnh quyết định của hệ thống trong quá trình tạo ra biểu thị.
Đóng góp của bài báo:

- Một phương pháp tiếp cận khởi tạo hỗn hợp cho phép người dùng tạo biểu thị dữ liệu của câu trả lời từ câu truy vấn tự nhiên,
- ambiguity space - mô hình của sự nhập nhằng trong truy vấn ngôn ngữ tự nhiên cho khám phá và biểu thị dữ liệu,
- thiết kế giao diện bao gồm các ambiguity widget, cho phép người dùng cung cấp các phản hồi tới hệ thống để xử lý nhập nhằng và điều hướng "ambiguity space"
- thuật toán quản lý các chỉnh sửa nhập nhằng theo thời gian qua các ràng buộc ưu tiên,
- thiết kế nghiên cứu mới để đánh giá các giao diện ngôn ngữ tự nhiên cho phân tích dữ liệu.

2 Giao diện của hệ thống DataTone



Hình 1: Giao diện người dùng của hệ thống DataTone

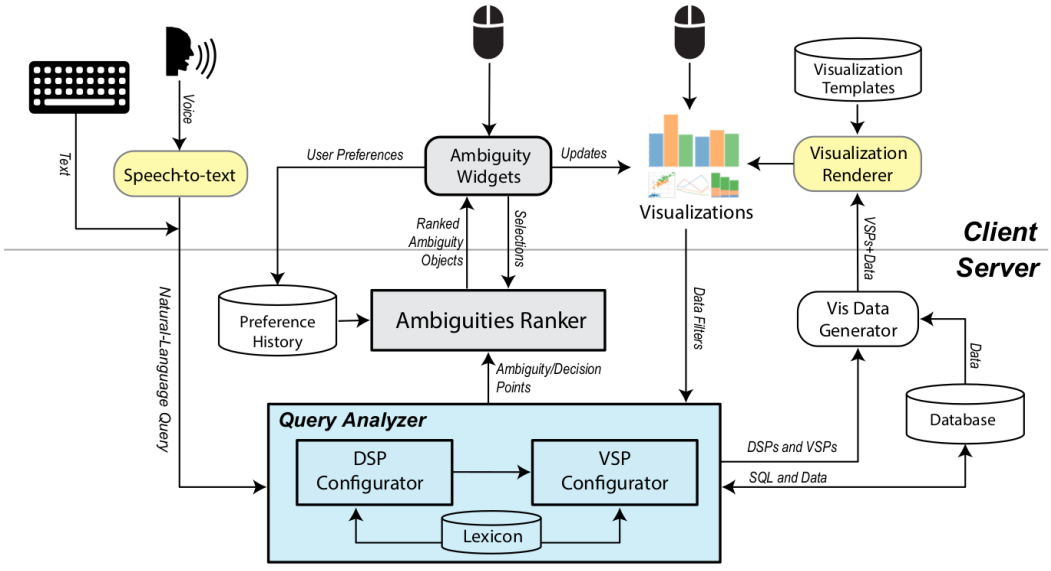
Trên hình là giao diện người dùng của hệ thống DataTone , bao gồm 4 phần :

- Tổng quan dữ liệu: mô tả khái quát các thuộc tính có trong tập dữ liệu được chọn.
- Hộp nhập truy vấn: người dùng có thể nhập từ bàn phím hoặc bằng giọng nói.
- Biểu thị: Biểu đồ mô tả câu trả lời cho truy vấn.
- Widget: Những option hệ thống đề xuất cho người dùng để tăng tính đúng đắn của câu trả lời mà người dùng mong muốn.

Ví dụ : truy vấn "show me medals for hockey and skating by country"
Hệ thống ghi nhận được 3 nhập nhằng dữ liệu tồn tại:

- Medals: tổng số hay chỉ gold hoặc chỉ silver hay chỉ bronze
- hockey: Hockey(Sport) hay IceHockey(Sport)
- Skating

3 Hệ thống DataTone



Hình 2: Kiến trúc hệ thống DataTone

Trong hệ thống DataTone, Query Analyzer có chức năng là chuyển đổi từ câu truy vấn sử dụng ngôn ngữ tự nhiên sang hai định dạng trung gian:

- Data Specification (DSP): capture những khía cạnh liên quan tới dữ liệu của truy vấn đầu vào và truy vấn đối với CSDL.

- Visual Specification (VSP): là một ngữ pháp đồ họa dùng để ánh xạ các thành phần của DSP với những thông tin cần thiết trong việc render các view.

3.1 Natural language interpretation

3.1.1 Tokenization and similarity mapping

Để chuyển đổi giữa ngôn ngữ tự nhiên và biểu thị trực quan , DataTone thực hiện một số chuyển đổi , đầu tiên sẽ ánh xạ ngôn ngữ tự nhiên với DSP , sau đó là DSP với VSP.
Ban đầu cần xác định được các đặc trưng bậc thấp của ngôn ngữ tự nhiên (ví dụ :từ , cụm từ, ...) có nghĩa trong ngữ cảnh của tập dữ liệu và các tác vụ phân tích. Tập các cụm từ đó được tạo ra bằng cách lấy tất cả n-gram của câu. Sau đó xác định những n-gram liên quan tới tập dữ liệu và định nghĩa tác vụ bằng cách so sánh mỗi n-gram với tập biểu thức chính quy và một từ điển gồm các cụm từ phổ biến (ví dụ: "compare", "average", "less than"). Cụ thể,hệ thống thực hiện gắn mỗi n-gram với một trong 8 nhãn :

- database attributes
- database cell values
- numerical values
- time expressions
- data operators and functions (ví dụ: greater than, less than, equal, sum, average, sort)
- visualization key phrases (ví dụ: trend, correclation, relationship, distribution, time series , bars, stacked bars, line graph)
- conjunction and disjunction terms (ví dụ and , or)
- “direct manipulation” terms (ví dụ : add, color)

Những n-gram được gắn chính xác với một từ vựng hoặc mẫu biểu thức chính quy được gắn nhãn tự động. Ngược lại, việc gắn nhãn sẽ dựa trên sự tương đương về mặt ngữ nghĩa thông qua việc đánh giá t = Sim($ngram_i$, $lexicon_j$). Nếu t $>$ 0.8 thì cặp ($ngram_i$, $lexicon_j$) sẽ được lựa chọn:
Sim($ngram_i$, $lexicon_j$)=MAX{$Sim_{wordnet}$ (. . .), $Sim_{spelling}$ (. . .)}.

- $Sim_{wordnet}(ngram_i, lexicon_j)$: khoảng cách đồ thị trong đồ thi worldNet
- $Sim_{spelling}(ngram_i, lexicon_j) = \frac{\vec{ngram_i}.\vec{lexicon_j}}{den}$: xử lý các lỗi chính tả chính bằng cách kiểm tra sự tương tự "bag of words" trong 2 token

3.1.2 Relation identification

Sau khi đã gắn nhãn, xây dựng bộ lọc bằng Stanford Core NLP Parser để tạo cây phân tích ngữ nghĩa và ngữ pháp.

Ví dụ: "show me the states that had total sales greater than than 20000 and less than 100000"
Ngữ pháp : "total sales" :NP (noun phrase), "greater than 20000 " :ADJP (adjective phrase)
Ngữ nghĩa: "greater than" and "less than" are connected to total sales.

3.2 Data specification (DSP)

Từ sự phân tích và gắn nhãn có thể tạo ra một hoặc nhiều DSP. Các thuộc tính trong DSP bao gồm:

- Attributes: giữ lại các thuộc tính được xác đinh trong câu gốc
- Values: dạng string, number, time, ... được xác định trong câu
- Filters
- Order: thứ tự xắp sếp dữ liệu được xác trong câu truy vấn
- Aggregates: sums, averages, maximum values, counts, ...
- Dimensions: biến độc lập - cách phân vùng dữ liệu
- Measures: đơn vị tính toán

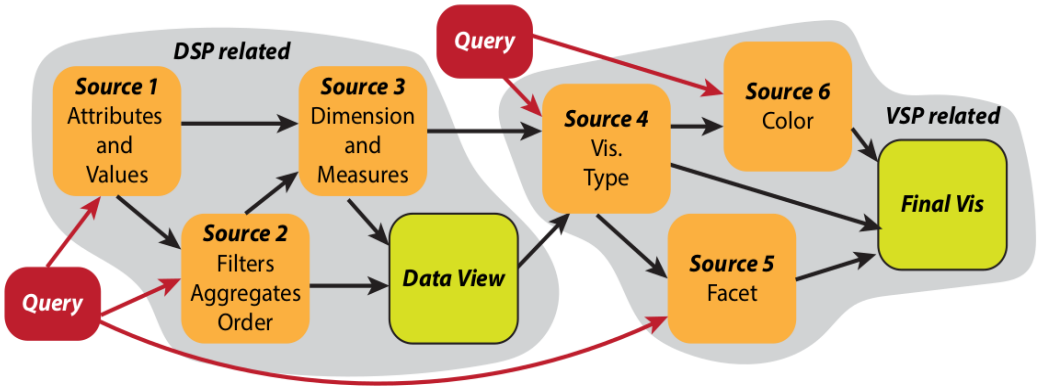
Sau đó câu lệnh truy vấn CSDL được tạo theo dạng:
SELECT {Aggregates}, {Dimension Attributes} FROM Table WHERE {Implicit Filters} GROUP BY {Dimension Attributes} HAVING {Explicit Filters} ORDER BY {Order}

3.3 Visualization generation

Kết quả thu được từ truy vấn SQL cùng với DSP đã tạo ra sẽ được sử dụng để tạo ra VSP (visual specification) cho mỗi DSP. Nếu truy vấn SQL không trả về kết quả hoặc dữ liệu trong kết quả quá ít thì hệ thống sẽ bỏ qua DSP đó. DataTone hiện tại hỗ trợ một số lượng hữu hạn loại biểu thị (scatter plots, various bar and line chart formats). Mỗi loại được đại diện bởi VSP "template" khác nhau ứng với các cấu hình khác nhau. Hệ thống sử dụng thư viện D3.js để render VSP được trả về cho client.

DataTone cố gắng ánh xạ mỗi DSP với mẫu VSP ứng với cấu hình cụ thể của DSP (dựa trên measure, dimension và loại DSP). Điều này có nghĩa là nhiều VSP có thể phù hợp với DSP đã cho. VSP được xếp hạng từ cao tới thấp và giúp cho người dùng dễ dàng xử lý sự nhập nhằng thông qua sự tương tác đơn giản với widget.

4 Xử lý sự nhập nhằng



Hình 3: Những sự phụ thuộc trong "Ambiguity space"

Các phần tử đường ống khác nhau trong DataTone tạo ra các loại nhập nhằng khác nhau, từ đó có thể dẫn đến nhiều sự biểu diễn khác nhau cho một truy vấn. Để biểu thị kết quả cho người dùng, lựa chọn đơn giản là hiển thị tất cả các biểu diễn khác nhau cho truy vấn đó. Tuy nhiên, sự nhập nhằng tăng theo cấp số nhân, do vậy không thể giải quyết như vậy được. DataTone xử lý bằng cách xếp hạng các VSP và cung cấp cho người dùng cuối một kĩ thuật để chuyển đổi nhanh chóng giữa các biểu thị.

4.1 Sự nhập nhằng dữ liệu

Sự nhập nhằng ngôn ngữ có thể xuất hiện ở mức từ vựng hoặc cú pháp.Sự mơ hồ từ vựng phát sinh do đa nghĩa và sự mơ hồ cú pháp do các cấu trúc có thể của câu .Việc tạo thành nhiều lời giải thích có thể của truy vấn dẫn đến nhiều nguồn nhập nhằng .

Source 1: Nhận diện thuộc tính cơ sở dữ liệu và giá trị văn bản

Người dùng cuối thường liên hệ tới các thực thể và các thuộc tính trong nhiều cách mơ hồ.
Ví dụ: khi người dùng hỏi về “product" và trong CSDL đang có thông tin về “product category“ và “product sub-category”, vậy thì thuộc tính nào sẽ được ánh xạ ?
DataTone xử lý vấn đề này bằng cách tính toán hàm Sim đã nhắc tới ở trên.

Source 2: Nhận diện các bộ lọc, phân loại và tổng hợp

Một câu truy vấn có thể được phân tích theo nhiều dạng cú pháp, dẫn đến nhiều dạng cấu trúc của câu đối với cùng một truy vấn.
Ví dụ: "population in Michigan and California in 2012" có thể được phân tích như một yêu cầu cho dân số của Michigan hiện tại và California năm 2012 thay vì cả 2 bang đều năm 2012. DataTone xử lý bằng cách dùng các bộ lọc tường minh.

Source 3: Lựa chọn Dimension và measure

Khi người dùng xác định bộ lọc, không rõ liệu có nên xử lý các thuộc tính của một bộ lọc như là một thuộc tính dimension hay không. Hệ thống có thể xếp hạng các option một cách heuristic, những thuộc tính dạng số thường là measure và những thuộc tính phân loại thường là dimension. Với một số trường hợp mà biến phân loại gồm quá nhiều trường hợp duy nhất, DataTone tính toán các giá trị duy nhất khác biệt chia cho số hàng. Kết quả thu được là cao thì sẽ quyết định xử lý cột đó như một measure (theo tính toán giá trị ngưỡng nhỏ nhất là 0.7)

4.2 Sự nhập nhằng trong quyết định thiết kế

Source 4: Lựa chọn mẫu biểu thị

Cho một DSP, có thể sẽ tồn tại nhiều mẫu VSP. DataTone sử dụng trực tiếp các lệnh trực tiếp hoặc các định nghĩa nhiệm vụ.Thực hiện điều này thông qua việc sử dụng từ điển. Tập từ điển của DataTone bao gồm các từ liên quan tới 4 tác vụ : comparison, correlation, distribution analysis và trends.

Source 5: Faceting data for small multiples

Các hiển thị small-multiple của thông tin giúp cho việc so sánh dữ liệu và tìm mẫu thông quan nhiều chiều. Với truy vấn đề cập đến nhiều thuộc tính dữ liệu, sự nhập nhằng đến từ quyết định thuộc tính nào để sử dụng như các tham số khía cạnh cho các bội số nhỏ và cách tổ chức chúng. Từ tập VSP đã xếp hạng, hệ thống ưu tiên xếp hạng nhóm phù hợp với thứ tự trong các thuộc tính được đề cập trong truy vấn.

Source 6: Chọn phương pháp mã hóa

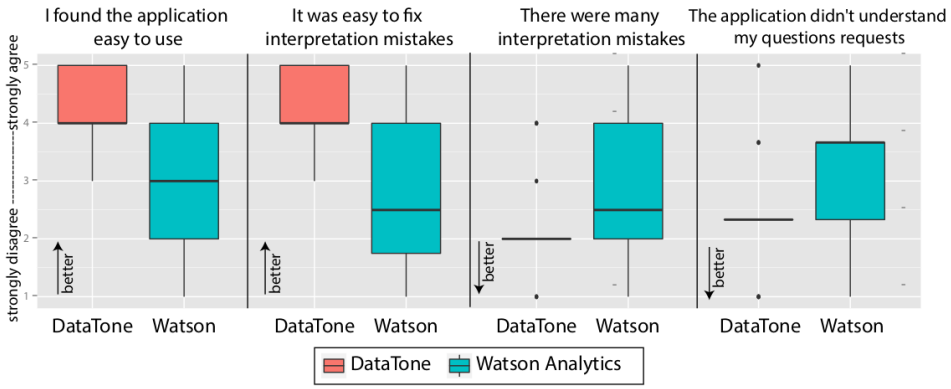
Màu sắc, hình dạng, và kích thước là các biến phổ biến trong một biệu thị. Trong mẫu thử nghiệm hiện tại, DataTone chỉ hỗ trợ mã hóa màu. Tuy nhiên, có thể vẫn còn mơ hồ trong việc sử dụng màu hay không. Mặc dù màu sắc có thể không được gọi rõ ràng, hoặc có thể dẫn đến mã hoá kép, nó vẫn có thể được mong muốn.

5 Đánh giá

- Để đánh giá DataTone, tác giả đã tiến hành một nghiên cứu so sánh đánh giá của người dùng với hệ thông Watson của IBM.
- So sánh 2 hệ thống Watson và DataTone:
  - Giống: cho phép người dùng xây dựng các hình ảnh hóa thông qua sự kết hợp giữa:
    - Tương tác ngôn ngữ tự nhiên.
    - Thao tác trực tiếp.
  - Khác: Cách chúng tiếp cận sự mơ hồ:
    - DataTone: xây dựng một hình ảnh trực quan trực tiếp từ truy vấn của người dùng.
    - Watson: Đáp ứng truy vấn của người dùng bằng 1 danh sách các câu hỏi được đề xuất (các câu hỏi mà hệ thống có thể trả lời bằng hình ảnh trực quan dữ liệu), sau đó mới cho người dùng điều chỉnh hình ảnh tạo ra thông qua thao tác trực tiếp.
- Do 2 hệ thống khác nhau về thiết kế giải quyết sự mơ hồ trong ngôn ngữ tự nhiên, nên tác giả quyết định sẽ so sánh toàn diện 2 hệ thống thay vì so sánh từng thành phần riêng lẻ.
- Phương pháp đánh giá: Đánh giá Jeopardy :
  - Đánh giá Jeopardy lấy cảm hứng từ một gameshow trên truyền hình của Mỹ. Ớ đó người dùng sẽ phải đưa ra câu trả lời dưới định dạng 1 câu hỏi.
  - Người chơi phải đưa ra câu trả lời bằng các hình ảnh trực quan để chứng minh đúng đắn hay phủ định điều đó.
  - Ví dụ: Chúng ta có thể nói rằng: “North Vietnam has the fewest number of people without jobs”
  - Người chơi phải đưa ra sự đúng sai của nhận định đó bằng cách sử dụng hệ thống để chứng minh nó bằng hình ảnh trực quan. Người chơi có thể sử dụng DataTone hoặc Watson cho đến khi họ tin rằng họ đã đạt được một câu trả lời thỏa đáng.
  - Tác giả chọn các chủ đề sao cho nếu người chơi chỉ đơn giản là lặp lại các từ ngữ trong chủ đề 1 cách đơn giản sẽ không thể tạo ra được 1 view chính xác.
  - Một câu trả lời hợp lí cho chủ đề nêu ở ví dụ trên là “Show me a sorted view of unemployment by state”

- Trong khảo sát, tác giả khảo sát 16 người tham gia, trong thời gian 1h với 3 bộ Dataset, 10 chủ đề /dataset
- 14/16 người thích sử dụng DataTone hơn.

Kết quả khảo sát



Hình 4: So sánh DataTone và IBM Waston

6 Thảo luận và công việc trong tương lai

- Hệ thống DataTone tốt hơn các sản phẩm thương mại hiện đại trên rất nhiều mẫu.
- Hệ thống có 1 vài hạn chế như:
  - Sự phân giải mơ hồ dựa vào heuristic
  - Linh hoạt hơn các widget trừu tượng
  - Hiện tại, tác giả chỉ hỗ trợ các bộ dữ liệu bảng đơn.
  - Cuối cùng, mặc dù DataTone cung cấp đầu vào giọng nói, nhưng nó chưa khuyến khích được dùng do chưa hoàn thiện