Finding Similar Data Using SQL Server Integration Services

SQL Server 2005 Integration Services (SSIS) introduces two new tools designed for Data Warehousing, but their uses are far more than just warehousing. Fuzzy Lookups and Fuzzy Grouping can both improve quality of data. Fuzzy lookups is designed to correct errors in lookup tables such as misspelling in cities or states. Fuzzy grouping finds duplicates in datasets. The hospital where I work strives to identify and merge duplicates to improve and maintain one complete electronic medical record for each patient. We have been using the fuzzy grouping tool for the past 6 months.

The concept of fuzzy grouping is not new; many have used probabilistic linkage of datasets to find duplicates and related records in other datasets (for more info visit http://www.utcodes.org/Linkage/description.htm). Few commercial probabilistic linkage packages are available and they are expensive. With SSIS we now have a free tool to do probabilistic linkages.

Fix the SSIS executables first

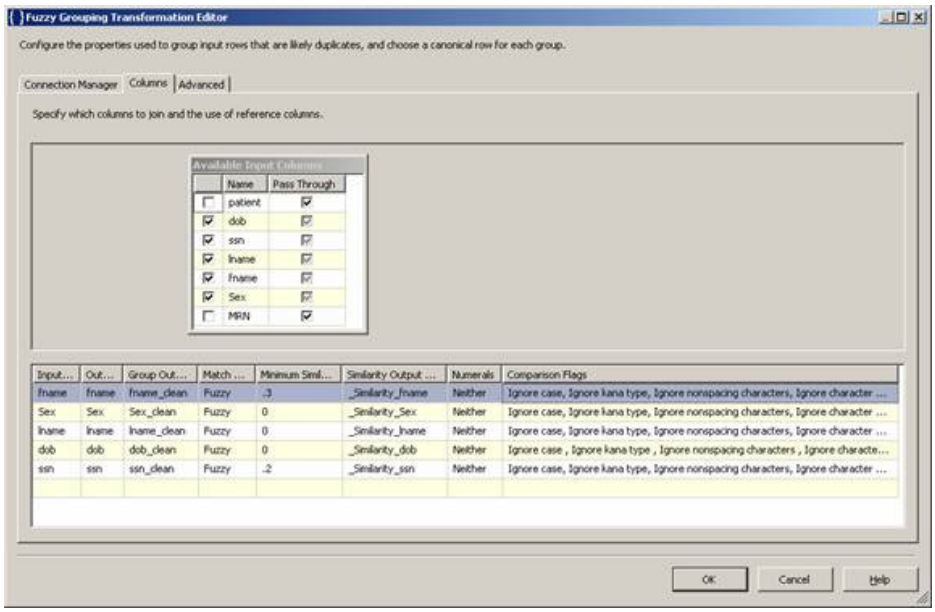
First install service pack 2 or the memory leak fix http://support.microsoft.com/kb/912423/. This fix resolves many memory issues when using record sets with over 1 million records, but there are still issues with more than 4 million records. Fuzzy grouping is memory and processor intensive so running SSIS on a server separate from the databases is recommended. On a 1.8 GHZ Pentium IV with 512 MB. My tests running an 800,000 record dataset took 5 hours. Adding another 512MB reduced the run time to 2 hours.

Define A Dataset

Open Visual Studio and start a BI project then define an OLE DB Source dataset. Any definable dataset may be grouped, but as many temporary tables are created, you must have a connection to a SQL Server and be a user with permission to create tables. Also limiting your dataset to the fields to be used for grouping and a primary key will increase performance. Be wary of data types as many are not supported and datetime fields must be transformed to varchar.

Add A Fuzzy Grouping Step

Next add a Fuzzy Grouping package. Define the fields that identify the duplicates. Fields such as First Name, Last Name, Gender, and Date of Birth will prove sufficient to find duplicates. Fields with high discerning power such as Social Security number will improve accuracy and should be given a higher weight. A primary key should also be used as a pass-through in the dataset to aid in future joins of the result set. Define the type of grouping, either exact (data must be equal) or fuzzy (data is compared to determine similarity). The fuzzy grouping allows you to choose further options to ignore cases, punctuation, kana, non-spacing characters, character width or symbols.

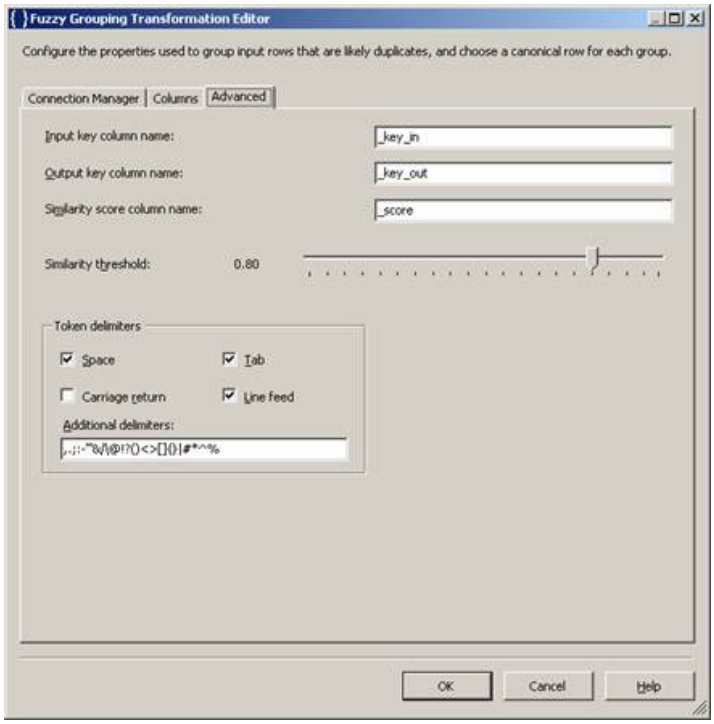


Minimum Similarities

Minimum Similarities should be defined for each field. These thresholds define how closely you want each of the values to correspond. Minimum similarities are between 0 and 1. Where 1 means the values are exact and 0 means review everything. Setting a higher minimum similarity will speed the matching process. Minimum similarities are very dependent on data and the number of matches, and match quality will depend on these values. Setting minimum similarities too high will result in only closer to exact records being found. So you may need some experimentation to determine the best thresholds to use. For example, I like my dates of birth (dob) to be close, but I realize that data entry errors skew dates, so I set a threshold of .20 for dob. In the example below May 12 1972 is compared against May 12 1971 and the similarity score is .84 so it would make the cut. If unsure run some tests with no minimum similarities and review your resulting _similarity fields for each field. Take some averages and review the matches to determine what a good match is, then set your threshold at this level.

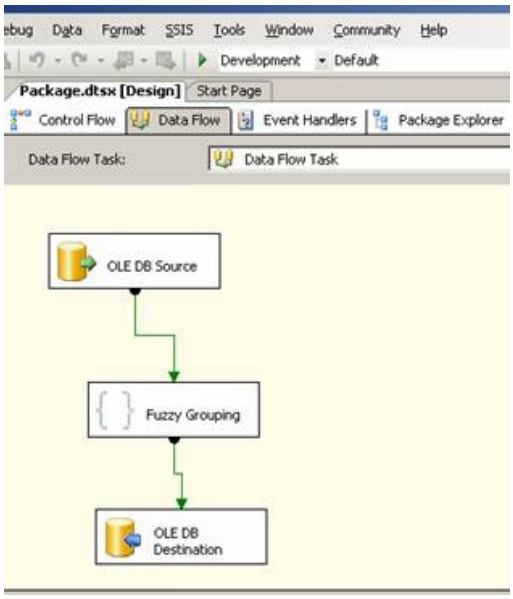
Minimum Thresholds

An overall minimum threshold must also be set. I have found that matches with my data get weak after .80. So I set my minimum overall threshold at .80. Again this is dependent on your data, accuracy of entry staff and how closely you want/need your records to match.



Define the Destination

The last step sets an OLE DB destination and either creates a new table or uses existing and matches the output fields in the fuzzy grouping step.



The outcome of the grouping will give a _key_in which is the unique key of the first table, a key_out field that identifies the unique groups, the _similarity_fieldname and the _score field that is the average of the _similarity_fieldname. The _score field is between 0 and 1 where a 0 is no similarity and 1 is exact.

Running the following dataset through the grouping:

Here is the sample input data:

patient	dob	ssn	Sex	lname	fname
12345	May 12 1971	5451545615	m	Nordberg	Brian
4564	May 12 1972	55661545615	M	Nordberger	Brian
11	May 12 1972	5151545615	m	Nordberg	Brain
51651	June 12 1968	45646541156	F	Biggs	Strong
51625	June 21 1968	45646541156	F	Bigs	Stong
51625	June 21 1968	45646541156	F	Biggs	Stong

And here are the outputs:

_key_in	_key_out	_score	patient	dob	dob_clean	_Similarity_dob	ssn	ssn_clean	_Similarity_ssn	Sex	Sex_clean	_Similarity_Sex	Lname	Lname_clean	_Similarity_Lname	fname	fname_clean	_Similarity_fname
2	2	1	4564	May 12 1972	May 12 1972	1	55661545615	55661545615	1	M	M	1	Nordberger	Nordberger	1	Brian	Brian	1
3	2	0.7919	11	May 12 1972	May 12 1972	1	5151545615	55661545615	0.709274	m	M	1	Nordberg	Nordberger	0.799934	Brain	Brian	0.53378
1	2	0.7622	12345	May 12 1971	May 12 1972	0.840515	545154516	55661545615	0.375	m	M	1	Nordberg	Nordberger	0.799934	Brian	Brian	1
6	6	1	51625	June 21 1968	June 21 1968	1	45646541156	45646541156	1	F	F	1	Biggs	Biggs	1	Stong	Stong	1
5	6	0.9463	51625	June 21 1968	June 21 1968	1	45646541156	45646541156	1	F	F	1	Bigs	Biggs	0.797414	Stong	Stong	1
4	6	0.8157	51651	June 12 1968	June 21 1968	0.607913	45646541156	45646541156	1	F	F	1	Biggs	Biggs	1	Strong	Stong	0.832197

Here it grouped the first 3 records together and the last 3 together. Note it did not choose the correct name in grouping one, it simply noted that they are all the same. So if you want the "correct" person, as we did for the hospital, you will need a human to review the results. In our hospital database of over 1.4 million records we found over 5000 duplicates. Unfortunately the nature of the business requires us to review each of the dataset and pull medical records, so an automated merge process is out of the question. But Fuzzy Grouping has proved to be a very valuable tool and has given us many useful reports.