

GutenTag Users' Manual

David L. Tabb

April 30, 2003

Abstract

GutenTag is software to automate the identification of peptides in proteomic spectral collections by the sequence tagging technique. The software infers partial sequences (“tags”) directly from the fragment ions present in each spectrum, examines a sequence database to assemble a list of candidate peptides, and evaluates the returned peptide sequences against the spectrum to determine which is the best match. The software, written in the Java programming language, runs equally well under Microsoft Windows, Linux, and other operating systems. GutenTag is specific to doubly-charged peptides. The program can analyze a twelve cycle MudPIT’s results in as little as six hours, though performance is heavily dependent upon database size.

1 Introduction

Typically, proteomic tandem mass spectra are identified by the database search technique, typified by the SEQUEST algorithm [1]. These algorithms search a sequence database for peptide sequences which would produce ions of the mass observed for a particular spectrum, then score these candidate sequences against the observed spectrum. The sequence tag technique for peptide identification was introduced by Matthias Mann in 1994 [2]. In this process, a partial sequence “tag” for the peptide is inferred directly from the observed spectrum, and then the database is searched to find peptides which include these partial sequences and which match the masses to either side of the partial sequence.

While these two techniques accomplish roughly the same thing, they go about it in complementary ways. The challenge is this: a spectrum can only be compared to a spectrum, and a

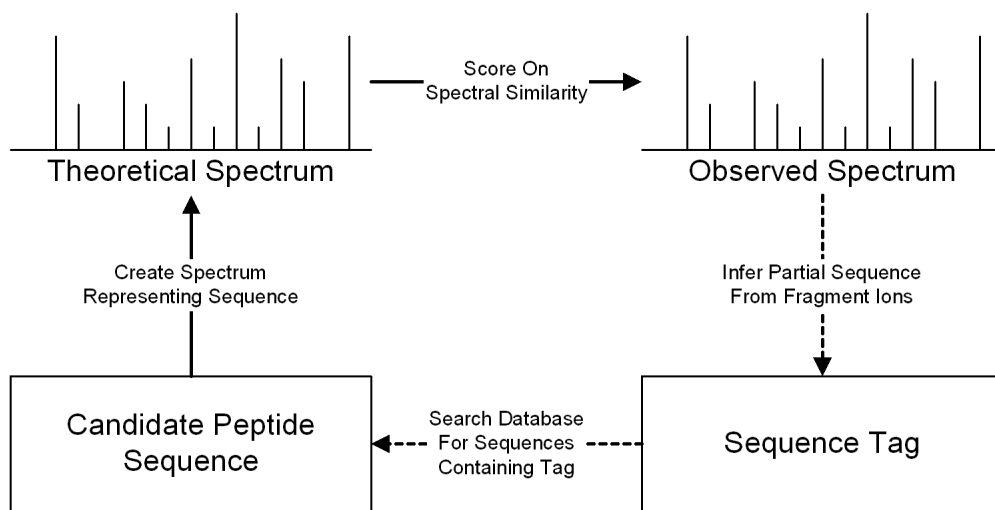


Figure 1: Database identification algorithms (solid arrows) construct theoretical spectra from database sequences, and then compare the theoretical spectra to the observed one. Sequence tag algorithms (dotted arrows) infer partial sequences from the spectrum and then search the database for these tag sequences. GutenTag runs the complete cycle, inferring a tag sequence, searching the database for the tag, constructing theoretical spectra for the candidate sequences, and then comparing the theoretical spectra to the observed spectrum.

sequence can only be compared to a sequence. To compare a sequence to a spectrum, one must either construct a theoretical spectrum from the sequence or infer a sequence from the spectrum. Database search algorithms use the former technique, while sequence tag algorithms use the latter (see Figure 1).

Sequence tag algorithms have not achieved the popularity of database search algorithms for several reasons. First, the process of inferring partial sequences from spectra is challenging; algorithmic models of fragmentation for peptides have been limited in their accuracies. In addition, manual inference of tag sequences takes too long for each spectrum to be practical for proteomic samples, which can generate thousands of spectra. Searching databases for tag sequences has not been optimized as well as possible, preventing the search of a database with multiple tags for each spectrum. Finally, when multiple candidate peptides are found from the tags for each spectrum, most sequence tag algorithms have not made provision for determining which candidate peptide sequence is the correct one.

GutenTag addresses all of these problems. Its model for inferring sequences from spectra

is based on a statistical analysis of reliable peptide identifications from SEQUEST, making accurate tag sequence inference possible. Its database search is optimized for searching multiple tags against the database sequence in a single pass [3]. The software can create theoretical spectra from candidate sequences and rapidly score them against the observed spectrum. GutenTag's performance and accuracy makes it a practical alternative to standard database search algorithms.

2 Installation and Execution

To install GutenTag, download the latest release of the program (the URL for doing should be in an email from The Scripps Research Institute). The Zip file should include several .class files, two .ini files, and a file titled GutenTag.bat. Extract this Zip into the directory C:\GutenTag. Copy GutenTag.bat to the C:\Winnt or C:\Windows directory. This batch file will let you start GutenTag from whatever directory you might be in because the directory where Windows is installed is almost always on the operating system path.

To set up for a run of GutenTag, you should collect the following into a directory on your machine:

- Spectra

Your spectra should be in either MS2¹ or DTA file format. These may, for example, be created as though preparing for a SEQUEST run with Thermo Finnigan's XCalibur software.

- GutenTag.ini

This configuration file determines which database is to be used for the search, the length of the tag sequences to be inferred, the number of tags to keep for each spectrum, and the masses and fragmentation behavior of amino acids. For more information on this file, see section 3.

- Isotoper.ini

This configuration file determines the isotope distribution to be used for each element. This file affects only the deisotoping during preprocessing of each spectrum.

¹<http://fields.scripps.edu/sequest/SQTFormat.html>

Next, open a command-line window (sometimes called MS-DOS Prompt and sometimes called Command Prompt). Switch to the drive and directory in which GutenTag is to run. If your directory contains DTA files, run the program with this invocation:

```
GutenTag --dta
```

but leave off the `--dta` option if your spectra are stored in MS2 files.

The software should start by loading the sequence database into memory. This may take time for particularly large databases. It's worth noting at this point that you should not try to run GutenTag on a database larger than the amount of memory you have available. Note that the database takes more space in memory than on disk (due to how Java stores String objects). In an test, a 24 MB database on disk resulted in a Java process of 110 MB (only part of which can be accounted for by the database). GutenTag performance scales with database size; each spectrum will take roughly a second for tag inference, and the remaining time is spent searching the database.

Once the database is in memory, the program will read the first spectrum from the disk, preprocess it, and infer sequence tags. Then the program will search the database for sequences which match a tag sequence as well as at least one of the flanking masses. The program will print any sequence matching the tag and both masses to the screen (see Figure 2).

As the software completes its analysis of each spectrum, it appends the top-ranked sequence tags to a TAG file and appends the best-matching peptide sequences to a SQT² file. It reports the number of spectra processed so far and then continues to the next. If DTA files are being processed, GutenTag will write its results to the file GutenTag.tag and GutenTag.sqt. If MS2 files are processed, GutenTag will use the base name of each MS2 file; if the file PfMrzP101.ms2 is being processed, for example, PfMrzP101.tag and PfMrzP101.sqt will be created. The other difference with MS2 files is that GutenTag will process each MS2 file found in the current directory, so multiple TAG and SQT files may be produced from MS2 files while only one TAG and SQT file result from DTA files.

3 Configuration

As alluded to before, two files allow configuration of GutenTag. Of these two, GutenTag.ini is likely to be of greater interest; Isotoper.ini is responsible only for setting the expected elemental

²<http://fields.scripps.edu/sequest/SQTFormat.html>

Preprocessing PfMrzP101.0372.0372.2...

Generating tag sequences...

Searching database for occurrence of tag sequences...

PFA0545c	158	171	PMYIQK RKD DKNNN
PFA0550w	301	316	QAVFFT PKK SVLNSSN
PF07_0058	489	502	EGFFKNSK ENY KEF
PF10_0044	710	723	NSDYIT PKY MIYIF
PF10_0110	350	363	VYYNIH NYK IVLEP
PF11_0239	611	625	RSHTDNNI ENY ISDS
PF11_0239	611	625	RSHTDNNIE NYI SDS
MAL13P1.295	796	809	PHVSYKIN ENY YLQ
PF14_0339	262	274	YDEMNRE EKF IKY

Scoring database sequences...

Appending to SQT file...

Appending to Tag file...

Spectra processed so far: 13

Figure 2: This text is an example of GutenTag's output while running. The lines showing the sequences assigned to the spectrum show the following information: the database ID for the matching protein, the positions in the protein sequence at which the peptide starts and stops, and the sequence matched. The sequence is divided into three portions, with the first representing the N-terminal sequence filled in from the database, the tag's sequence, and the C-terminal sequence filled in from the database. Note that RSHTDNNIENYISDS is matched to two different tags.

isotope ratios and so should require no changes. The options listed in GutenTag.ini, however, may require attention.

The first part of GutenTag.ini configures the way in which GutenTag functions. These options include:

- `DatabaseName`: This line specifies the path and filename of the FASTA database to be searched. For example, `c:\databases\yeast_orfs.fasta`.
- `SeqLenMin`: What is the length of the shortest tags to infer?
- `SeqLenMax`: What is the length of the longest tags to infer? Note that increasing this value even a small amount increases tag inference time significantly.
- `SeqQuota`: How many tags of each length should be retained?
- `FragmentTolerance`: How much tolerance between expected and observed m/z positions of fragment ions is permissible?
- `TrypticMinimum`: Do we require the ends of the sequence to be tryptic cleavage positions? If only partial identifications are found for a spectrum and this is set to two, no identification can be made because only one end is known. Note that this option will not increase the speed of the database search. DTASelect's `-y` option is preferable to GutenTag's `TrypticMinimum` option.

The remainder of the GutenTag.ini file should be modified only by folks who know what they're doing. These lines specify which amino acids are permissible in inferring tags and searching the database. Each line begins with the token `Res`. The next field is the character representing each amino acid. The third and fourth fields are the average and monoisotopic masses of the residues, respectively. The fifth and six fields are the N-bias values for y and b series ions, as described in a recent publication [4]. Note that cysteine's mass is recorded as 160.1448 in the default GutenTag.ini file. This mass assumes that one has reduced and alkylated the sample prior to tandem mass spectrometry; this treatment increases cysteine's mass by 57 Da.

4 Output interpretation

The first thing one should note about making sense of large masses of peptide identifications is that one shouldn't do it by hand. DTASelect and Contrast³ [5] are free of charge to academic and non-profit users and are capable of reading SEQUEST and GutenTag results interchangeably. If users need to examine the details of each spectrum's analysis, though, the formats of the TAG and SQT files produced will come in handy.

4.1 TAG files

The TAG file stores lists of tag sequences inferred from each spectrum. The format consists of S lines, describing spectra, and T lines, describing the tags derived from each spectrum. Each S line is followed by the T lines corresponding to the spectrum described in the S line. Each S line includes the following elements:

1. Low scan number: This is the first scan from which this spectrum was acquired.
2. High scan number: This is the last scan from which this spectrum was acquired.
3. Precursor charge state: Since GutenTag only processes spectra from doubly-charged precursors, this is always 2.
4. Time to process: How many seconds were required to process this spectrum?
5. Initial peak count: How many peaks were in the spectrum as read from the disk?
6. Final peak count: How many peaks remained after deisotoping?
7. Initial M+H⁺ mass: What was the observed precursor mass, given that this was a doubly-charged precursor ion?
8. Corrected M+H⁺ mass: What was the precursor mass estimated to be from fragment ion complementarity?
9. Sum of peak intensity: If we add the intensity of every peak in the tandem mass spectrum, what is the sum?

The T lines give data describing each tag found for each spectrum:

1. Tag score: Tags are listed from lowest-scoring to highest-scoring. A high score indicates a more likely sequence tag.

³<http://fields.scripps.edu/DTASelect>

2. Tag sequence: What residues comprise this tag?
3. Prefix mass: How much mass for this peptide is N-terminal to the highest m/z y series fragment ion for this tag?
4. Suffix mass: How much mass for this peptide is C-terminal to the lowest m/z y series fragment ion for this tag?

In essence, a TAG file gives the results for each spectrum prior to database lookup.

4.2 SQT Format

The results of the database search are stored in the SQT file. SQT files are more complex in format than TAG files because they store information specific to each spectrum (S lines), to each match for each spectrum (M lines), and for database sequences in which each match is found (L lines). M lines always “belong to” the preceding S line, and L lines always “belong to” the preceding M line. Figure 3 shows the system of objects implied by the SQT structure.

The SQT format was initially designed for the SEQUEST algorithm, and so many fields are used differently than originally intended. The values which GutenTag stores in each field of the S lines follow:

1. Low scan number: This is the first scan from which this spectrum was acquired.
2. High scan number: This is the last scan from which this spectrum was acquired.
3. Precursor charge state: Since GutenTag only processes spectra from doubly-charged precursors, this is always 2.
4. Time to process: How many seconds were required to process this spectrum?
5. “GutenTag”: This field is used for server name in SEQUEST, but GutenTag reports the algorithm used instead.
6. Corrected m/z of precursor: What precursor value corresponds best to complementarity among the fragment ions? This field holds observed precursor mass for SEQUEST.
7. Intensity sum: The sum of fragment ion intensities is stored here rather than the “total intensity” reported in SEQUEST OUT and SQT files.
8. Best tag score: What was the highest tag score for the best database sequence matched to this spectrum? In SEQUEST, this field holds the lowest preliminary score.

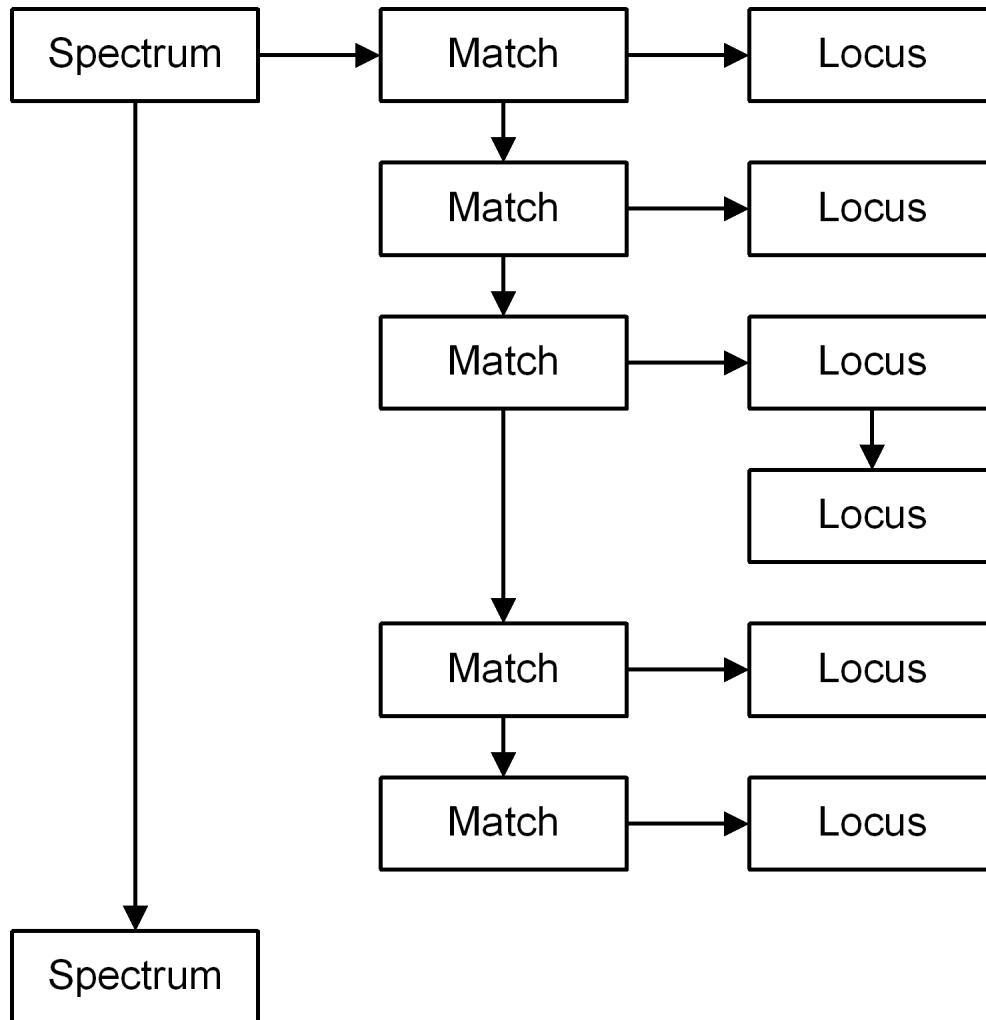


Figure 3: The region of a SQT file corresponding to each spectrum begins with an S (spectrum) line and continues until the next S line. The above diagram shows a collection of object implied by the following sequence of lines: SMLMLLLMLML. The described spectrum has five listed matches, with each matching sequence found in one locus, except for the third matching sequence, which is found in two loci. The next spectrum object is included only to emphasize that a series of spectrum objects are specified in a SQT file.

9. Sequence count: How many database sequences matched the tag sequence and at least one flanking mass? In SEQUEST, this field stores the number of database sequences matching the precursor mass.

The fields on the M lines include:

1. Rank by score: How did this database sequence rank by score (ions matched times normalized dot product)?
2. Rank by tags matched: The number of tags which matched to the most commonly tagged database sequence is p . The number of tags matched to this sequence is t . The number reported here is $(p + 1) - t$. In other words, the sequence matching the most tags gets a "1," and the others will have higher values. This is intended to correspond to the rank by preliminary score produced by SEQUEST.
3. Calculated sequence mass: What is the molecular weight of the sequence shown here? If this sequence is partial rather than complete, a substantial gulf may exist between this and the precursor mass.
4. DeltCN: If h is the highest score for any match on this spectrum and t is the score of this match, $DeltCN = (h - t)/h$. This is the same way DeltCN is calculated for SEQUEST.
5. Score: Let m be the number of ions in this spectrum matching to this sequence, and let d be the normalized dot product of the predicted ions against matching ions in the spectrum (d ranges from 0-1). $Score = m * d$. This is where one would find XCorr in SEQUEST results.
6. Percent of TIC accounted for: What percentage of the intensity in this spectrum does this sequence explain? SEQUEST puts its preliminary score here.
7. Matched ions: How many of the peaks in this spectrum correspond to this sequence?
8. Expected ions: How many peaks would be expected within the scan range of this spectrum if this were the correct sequence? Note that SEQUEST does not take scan range into account when reporting this value.
9. Sequence: What is this database sequence? GutenTag always reports the flanking amino acids as "-", while SEQUEST includes the flanking characters from the database. DTASelect will fill in the correct letters when it reads each identification.

10. Validation State: GutenTag and SEQUEST both report “U” (unknown) when writing a SQT file.

Finally, the L lines for each identification include:

1. Protein ID: What is the name of this protein in which the above sequence is found?
2. Context info: Where within this protein’s sequence does this peptide appear? Were the N- and C-termini tryptic cleavage sites or protein sequence termini?

5 DTASelect and GutenTag

Using DTASelect with GutenTag rather than SEQUEST results requires a bit of extra care.

The following tips may help you use these programs together more successfully:

- sequest.params: DTASelect can only read algorithm configurations from sequest.params files. To make a sequest.params for GutenTag results, specify the correct database, report monoisotopic masses for fragment ion calculation and precursor mass calculation, and ensure that the mass of cysteine is statically modified to 160 Da.
- Labels for fields: DTASelect knows that the values in GutenTag-produced SQT files are not the same values as would be present in SEQUEST-produced SQT files. Different labels appear above the columns for each peptide as a result.
- Score threshold: The primary score in GutenTag is a normalized dot product multiplied by a number of ions rather than an XCorr. While XCorrs for correctly identified doubly-charged precursor spectra are generally above 2.5, a comparable GutenTag identification’s score is generally above 10. Shorter peptides will tend to score lower than longer peptides.
- Isobaric differentiation: GutenTag models the intensities differently for isoleucine and leucine, so if the peptide ADLTAEGR will receive a different score than ADITAEGR for the same spectrum. The difference between these scores may be very small, resulting in a low DeltCN value. Use DTASelect’s `-d 0` option to prevent filtering on the basis of DeltCN. Glutamine and lysine are also roughly isobaric, but they are rarely substituted for each other in real-world sequence variations.

- Partial sequence scores: Partial sequences may be too short to score above a cutoff of 10 due to a smaller number of ions calculated. You may want to set the score cutoff to zero and show only the peptides for the protein of interest, as in `DTASelect -2 0 -E KERATIN`.
- Partial sequence extents: If you have multiple copies of a modified spectrum, different copies may receive different identifications. For example, the modified sequence `LEQINVGM*R` may be identified as `LEQINVG-`, `LEQINV-`, `LEQIN-`, etc. As a result, the same spectrum may receive different sequence identifications from GutenTag. On the converse, spectra may represent different modified peptides but receive the same identification from GutenTag. If you are looking for modification spectra, you may want to use DTASelect's `-t 0` option to prevent these spectra from being grouped together with only one shown.

References

- [1] Yates, J. R. III; Eng, J. K.; McCormack, A. L., Schieltz, D. Method to correlate tandem mass spectra of modified peptides to amino acid sequences in the protein database. *Anal. Chem.* 1995, 67: 1426-1436.
- [2] Mann, M.; Wilm, M. Error-tolerant identification of peptides in sequence databases by peptide sequence tags. *Anal. Chem.* 1994, 66: 4390-4399.
- [3] Aho, A. V.; Corasick, M. J. Efficient string matching: an aid to bibliographic search. *Comm. ACM* 1975, 18: 333-340.
- [4] Tabb, D. L.; Smith, L. L.; Brechi, L. A.; Wysocki, V. H.; Lin, D.; Yates, J. R. III. Statistical characterization of ion trap tandem mass spectra from doubly charged tryptic peptides. *Anal. Chem.* 2003, 75: 1155-1163.
- [5] Tabb, DL; McDonald, WH; Yates, JR 3rd. DTASelect and Contrast: Tools for assembling and comparing protein identifications from shotgun proteomics. *J. Proteome Res.* 2002 1:21-26.